



**AÇIKLANABİLİR YAPAY ZEKÂ KULLANARAK SALDIRI TESPİT
SİSTEMİ GELİŞTİRME**

Samed AL

DOKTORA TEZİ
BİLGİ GÜVENLİĞİ MÜHENDİSLİĞİ ANA BİLİM DALI

GAZİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

ARALIK 2025

ETİK BEYAN

Gazi Üniversitesi Fen Bilimleri Enstitüsü Tez Yazım Kurallarına uygun olarak hazırladığım bu tez çalışmada;

- Tez içinde sunduğum verileri, bilgileri ve dokümanları akademik ve etik kurallar çerçevesinde elde ettiğimi,
- Tüm bilgi, belge, değerlendirme ve sonuçları bilimsel etik ve ahlak kurallarına uygun olarak sunduğumu,
- Tez çalışmada yararlandığım eserlerin tümüne uygun atıfta bulunarak kaynak gösterdiğimi,
- Kullanılan verilerde herhangi bir değişiklik yapmadığımı,
- Bu tezde sunduğum çalışmanın özgün olduğunu,

bildirir, aksi bir durumda aleyhime doğabilecek tüm hak kayıplarını kabullendiğimi beyan ederim.

Samed AL

29/12/2025

AÇIKLANABİLİR YAPAY ZEKÂ KULLANARAK SALDIRI TESPİT SİSTEMİ GELİŞTİRME

(Doktora Tezi)

Samed AL

GAZİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

Aralık 2025

ÖZET

Saldırı Tespit Sistemleri (STS), yetkisiz erişimleri ve zararlı aktiviteleri tespit ederek bilgi sistemlerinin korunmasında önemli bir rol oynamaktadır. Her geçen gün karmaşıklaşan siber tehditlere karşı, geleneksel saldırı tespit sistemleri gelişmiş Yapay Zekâ (YZ) tekniklerini kullanacak şekilde evrimleşmiştir. Ancak, "kara kutular" olarak adlandırılan özellikle derin öğrenme temelli birçok YZ modelinin karmaşık yapısı, zeki sistemlerin karar alma süreçlerini anlama ve bunlara güvenme konusunda önemli zorluklar ortaya koymaktadır. Açıklanabilir Yapay Zekâ (AYZ), karmaşık modellere şeffaflık ve açıklanabilirlik sağlayarak bu soruna bir çözüm olarak ortaya çıkmaktadır. STS'ler YZ ile birlikte daha büyük önem kazanmakta, YZ modellerinin aldığı kararların nedenlerini bilmek ise bu kritik alanda en önemli çalışma alanlarının başında gelmekte, AYZ çözümlerine olan ihtiyaç ise giderek artmaktadır. YZ modelleri ve çıktıları daha açıklanabilir ve şeffaf hale getirmek, artan endişeleri gidermek, kara kutu problemini çözmek, modellere ve kararlara güveni artırmak amacıyla açıklanabilir STS konusu bu tez çalışmasında ele alınmış; STS ve AYZ ilişkisi detaylı olarak incelenmiş; 2018-2025 yılları arasında yapılan 386 adet AYZ destekli STS çalışması farklı perspektiflerden gözden geçirilmiş; literatürdeki en kapsamlı çalışma yapılmış; YZ temelli geliştirilen STS modellerinin kararlarını güvenle açıklayan ve üç adımdan oluşan (veri ön işleme, hibrit sınıflandırma ve açıklanabilirlik) özgün bir STS geliştirilmiştir. Ayrıca, tez çalışmasında; öznitelik seçimi, veri dengeleme, LightGBM, GCN, SHAP ve GNNExplainer yaklaşımlarından oluşan üç özgün algoritma önerisi sunulmuş; büyük dil modelleri (BDM) kullanılarak mevcut sistemin kararlarını daha açıklanabilir hale getirmek için dördüncü bir BDM algoritması sisteme entegre edilmiştir. Önerilen birinci algoritma; veri ön işleme adımında rasgele orman özellik önemi, Pearson korelasyon matrisi, SMOTE ve Tomek-Links tekniklerinden oluşmaktadır. İkinci algoritma olarak; hibrit sınıflandırma adımında LightGBM ve GCN yaklaşımlarının birleşiminden oluşan hibrit bir sınıflandırma algoritması geliştirilmiştir. Üçüncü algoritma; SHAP ve GNNExplainer AYZ yaklaşımlarından oluşan yeni ve çok boyutlu bir açıklanabilir YZ önerisidir. Dördüncü algortimada ise BDM entegrasyonu sonucunda elde edilen açıklanabilirlik çıktıları 2 BDM sohbet robotu kullanılarak yorumlatılmış ve kullanıcılara geliştirilen sistemin verdiği sonuçlar daha kapsamlı açıklanmıştır. Önerilen kompakt sistemin yalnızca yüksek performansla çalışması değil, aynı zamanda güvenilir ve anlaşılabilir kararlar ürettiği yapılan deneysel çalışmalar ile doğrulanmıştır. Deneysel çalışmalarda, iki farklı veri kümesi kullanılarak farklı metriklerle ölçülmüş, literatürdeki 19 farklı YZ modeli ile eğitilerek karşılaştırılmış ve sonuçlar değerlendirilmiştir. Önerilen sistem; ikili ve çoklu sınıflandırma sonuçlarında veri kümelerine bağlı olarak 0,8444 ile 0,9999 arasında doğruluk performansına ulaşmıştır. Elde edilen sonuçlar; literatürdeki mevcut modellere kıyasla oldukça başarılı olup özniteliklerin model çıktıları üzerindeki etkileri ile birlikte alt-graf seviyesinde de açıklanabilirlik sağlanmış ve böylece, önerilen sistem hem özellik düzeyinde hem de ilişkisel düzeyde çok boyutlu çözümler üretmiştir. Bunlara ilave olarak, veri kümeleri, Zeek açık kaynak STS üzerinde gerçek bir STS ortamında çalıştırılarak elde edilen performanslar değerlendirilmiştir. Sonuç olarak; bu tez çalışmasında geliştirilen çözümlerin siber güvenlik gibi kritik bir alanda sadece STS'ler için değil diğer alanlarda da kullanılabileceği, YZ modellerinde "kara kutu" problemlerinin çözümüne katkılar sağlayacağı, AYZ'nin STS'lere entegrasyonu sonucunda açıklanabilir modellerin saldırı tespitinin etkinliğini, güvenilirliğini ve hesap verebilirliğini artırabileceği, AYZ'yi modern STS'lerin temel bir bileşeni haline getireceği öngörülmektedir.

Bilim Kodu : 92438

Anahtar Kelimeler : Saldırı tespit sistemi, YZ, açıklanabilir YZ, derin öğrenme, graf sinir ağları, büyük dil modelleri, LLM, XAI

Sayfa Adedi : 284

Danışman : Prof. Dr. Şeref SAĞIROĞLU

DEVELOPING AN INTRUSION DETECTION SYSTEM USING EXPLAINABLE ARTIFICIAL INTELLIGENCE

(Ph. D. Thesis)

Samed AL

GAZİ UNIVERSITY

GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

December 2025

ABSTRACT

Intrusion Detection Systems (IDS) play a crucial role in protecting information systems by detecting unauthorized access and malicious activity. To combat increasingly complex cyber threats, traditional intrusion detection systems have evolved to utilize advanced Artificial Intelligence (AI) techniques. However, the complex nature of many AI models, particularly those based on deep learning and referred to as "black boxes," presents significant challenges in understanding and trusting the decision-making processes of these intelligent systems. Explainable Artificial Intelligence (AI) emerges as a solution to this problem by providing transparency and explainability to complex models. IDSs are gaining greater importance with AI, and understanding the reasons behind the decisions made by AI models is one of the most important areas of study in this critical field, while the need for AI solutions is steadily increasing. This thesis addresses the topic of explainable IDS with the aim of making AI models and their outputs more explainable and transparent, alleviating growing concerns, solving the black box problem, and increasing confidence in models and decisions; the relationship between IDS and AI is examined in detail; 386 AI-supported IDS studies conducted between 2018 and 2025 are reviewed from different perspectives; the most comprehensive study in the literature is conducted; and a unique IDS that reliably explains the decisions of AI-based IDS models and consists of three steps (data preprocessing, hybrid classification, and explainability) is developed. Furthermore, in this thesis, three original algorithm proposals consisting of feature selection, data balancing, LightGBM, GCN, SHAP, and GNNExplainer approaches are presented; a fourth LLM algorithm is integrated into the system to make the decisions of the existing system more explainable using large language models (LLMs). The first proposed algorithm consists of random forest feature importance, Pearson correlation matrix, SMOTE, and Tomek-Links techniques in the data preprocessing step. The second algorithm is a hybrid classification algorithm developed by combining LightGBM and GCN approaches in the hybrid classification step. The third algorithm is a new and multidimensional explainable AI proposal consisting of SHAP and GNNExplainer AI approaches. In the fourth algorithm, the explainability outputs obtained as a result of LLM integration are interpreted using 2 LLM chatbots, and the results given by the developed system are explained to the users in a more comprehensive way. Experimental studies have confirmed that the proposed compact system not only operates with high performance but also produces reliable and understandable decisions. In experimental studies, two datasets were used, measured with different metrics, trained with 19 different AI models in the literature, compared, and the results were evaluated. The proposed system achieved accuracy performance between 0.8444 and 0.9999 in binary and multiple classification results, depending on the datasets. The obtained results are quite successful compared to existing models in the literature, and explainability was provided at the sub-graph level as well as the effects of features on model outputs, thus the proposed system produced multidimensional solutions at both the feature and relational levels. In addition, the performance of the developed models was evaluated by running them in a real IDS environment on the Zeek open-source IDS. In conclusion, it is predicted that the solutions developed in this thesis can be used not only for IDS in a critical field such as cybersecurity, but also in other areas; that they will contribute to solving "black box" problems in AI models; that the integration of AI into IDS can increase the effectiveness, reliability, and accountability of explainable models in attack detection; and that AI will become a fundamental component of modern IDS.

Science Code : 92438

Key Words : Intrusion detection system, artificial intelligence, explainable artificial intelligence, deep learning, graphical neural networks, LLM, XAI

Page Number : 284

Supervisor : Prof. Dr. Şeref SAĞIROĞLU

TEŞEKKÜR

Öncelikle, bugün bağımsız, özgür ve güvenli bir ortamda akademik çalışmalar yürütebilmemizi mümkün kılan aziz şehitlerimize en derin minnet ve teşekkürlerimi sunuyorum. Doktora eğitimim boyunca hem akademik hem sosyal anlamda engin tecrübe ve bilgi birikiminden daima faydalandığım değerli danışman hocam Sayın Prof. Dr. Şeref SAĞIROĞLU'na, tez izleme komitemde yer alıp çalışmalarım boyunca değerli katkı ve desteklerini esirgemeyen Sayın Prof. Dr. Murat DENER ile Sayın Prof. Dr. Ertan ONUR hocalarıma teşekkür ve saygılarımı sunuyorum. Bilgi güvenliği alanı ile tanışmama ve bu alanda çalışmama vesile olan Sayın Hamid AKKUŞ'a, lisansüstü eğitim hayatım boyunca beni sürekli teşvik eden ve destekleyen değerli büyüğüm, amcam Sayın Salih KÜÇÜKKAYA'ya teşekkür ve saygılarımı sunuyorum. Doktora eğitimim boyunca sabırla bana destek olan ve anlayışını eksik etmeyen sevgili eşim Şadiye'ye, bu süreç içerisinde yeterince vakit ayıramadığım kızlarım Miray ve Gökçe'ye ve son olarak beni yetiştirerek bu günlere gelmemi sağlayan Anneme ve Babama sonsuz teşekkürlerimi sunuyorum.

Bu tez çalışması; Gazi Üniversitesi BAP birimi tarafından ADEP programı kapsamında (FGA-2022-8043) desteklenmiştir. Gazi Üniversitesi Bilimsel Araştırma Projeleri Birimine ve Yüksek Öğretim Kurumu Başkanlığına destekleri için şükranlarımı sunarım. Aynı zamanda; deneysel çalışmaların gerçekleştirilmesindeki katkılardan dolayı TÜBİTAK TRUBA'ya ve çalışmalarım boyunca karşılaştığım sorunlarda sağladıkları çözüm olanakları ve cevapları için destek ve yardımları için ilgili birimlere ve yetkililerine teşekkürlerimi bir borç bilirim.

İÇİNDEKİLER

	Sayfa
ÖZET	iv
ABSTRACT.....	vi
TEŞEKKÜR.....	vii
İÇİNDEKİLER	ix
ÇİZELGELERİN LİSTESİ.....	xiv
ŞEKİLLERİN LİSTESİ.....	xvi
RESİMLERİN LİSTESİ	xix
SİMGELER VE KISALTMALAR.....	xx
1. GİRİŞ.....	1
2. MEVCUT ÇALIŞMALARIN İNCELENMESİ	6
2.1. Literatürdeki Derleme Çalışmalarının İncelenmesi	8
2.2. Literatürdeki AYZ İçeren STS Çalışmalarının İncelenmesi	12
3. SALDIRI TESPİT SİSTEMLERİ	27
3.1. Saldırı Tespit Sistemlerinin Sınıflandırılması	35
3.1.1. Zaman davranışına göre sınıflandırma	35
3.1.2. Mimari yapıya göre sınıflandırma.....	36
3.1.3. Bilgi kaynağına göre sınıflandırma	37
3.1.4. Saldırı tespit yöntemine göre sınıflandırma	39
3.1.5. Hedef sistemlere göre sınıflandırma	40
3.2. Anormallik Temelli Saldırı Tespit Sistemleri.....	42
3.3. Anormallik Tespitinde Kullanılan Yöntemler	44
3.3.1. İstatistiksel temelli yöntemler	45
3.3.2. Bilgi temelli yöntemler	48
3.3.3. Veri madenciliği temelli yöntemler	49
3.3.4. Makine öğrenmesi temelli yöntemler.....	51

	Sayfa
3.3.5. Kullanıcı davranışı belirleme	54
3.3.6. Bilgisayar immünolojisi	54
3.4. Saldırı Tespit Sistemlerinde Sorun ve Zorluklar	54
3.5. Saldırı Tespit Sistemi Veri Kümeleri	56
3.6. Zeki Saldırı Tespit Sistemleri	57
4. AÇIKLANABİLİR YAPAY ZEKÂ	62
4.1. YZ ve Kara Kutu Problemi	63
4.2. Açıklanabilir YZ Terminolojisi	66
4.2.1. Açıklanabilirlik	67
4.2.2. Yorumlanabilirlik	67
4.3. Açıklanabilir YZ Hedefleri	68
4.4. Açıklama Türleri	70
4.5. Açıklanabilir YZ Taksonomisi	72
4.5.1. Kapsam bakımından açıklanabilir YZ	72
4.5.2. Uygulama aşaması bakımından açıklanabilir YZ	73
4.5.3. Model bakımından açıklanabilir YZ	74
4.5.4. Veri türü bakımından açıklanabilir YZ	75
4.6. Açıklanabilir YZ Teknikleri	75
4.6.1. Görsel açıklamalar	76
4.6.2. Özellik temelli açıklamalar	76
4.6.3. Örnek temelli açıklamalar	76
4.6.4. Pertürbasyon temelli açıklamalar	77
4.6.5. Gradyan/geri-yayılım temelli açıklamalar	77
4.6.6. İçsel açıklanabilir metotlar	77
4.7. Açıklanabilir YZ Araç Kutuları	79
4.8. Açıklama Değerlendirme Kriterleri	83

	Sayfa
4.8.1. Hesaplamalı metrikler	84
4.8.2. Bilişsel metrikler	86
4.9. Büyük Dil Modellerinde Açıklanabilir YZ.....	87
4.10. Saldırı Tespit Sistemlerinde Açıklanabilir YZ'nin Önemi	89
4.11. Saldırı Tespit Sistemlerinde Açıklanabilir YZ'nin Zorlukları	91
4.12. Açık Konular ve Gelecek Yönler	93
4.12.1. AYZ özellikli STS'lere yönelik saldırılar	94
4.12.2. Yasal uyumluluklar ve mahremiyet	94
4.12.3. Performans ve açıklanabilirlik arasındaki denge	94
4.12.4. Değerlendirme ölçütleri.....	95
5. MATERYAL VE METOT	96
5.1. Veri Kümeleri	96
5.1.1. CIC-IDS2017 veri kümesi.....	96
5.1.2. UNSW-NB15 veri kümesi	100
5.2. Veri Kümesi Temizleme	102
5.3. Öznitelik Seçimi	103
5.4. Veri Dengeleme	106
5.4.1. SMOTE	106
5.4.2. Tomek-links	108
5.5. LightGBM Algoritması	108
5.6. Graf Sinir Ağları	112
5.6.1. GCN	114
5.6.2. GraphSAGE	115
5.6.3. GAT.....	116
5.6.4. GIN.....	117
5.7. SHAP Algoritması.....	118

	Sayfa
5.8. GNNExplainer Algoritması	120
5.9. Zeek Açık Kaynak Saldırı Tespit Sistemi.....	123
5.10. BDM Tabanlı Açıklama	127
5.11. TRUBA Platformu	129
5.12. Google Colab Platformu.....	131
5.13. Değerlendirme Metrikleri	133
6. ÖNERİLEN MODEL.....	135
6.1. Önerilen Veri Önişleme Adımı	135
6.2. Önerilen Sınıflandırma Adımı	138
6.3. Önerilen Açıklanabilirlik Adımı.....	145
6.4. BDM Yaklaşımı ve Entegrasyonu.....	145
7. DENEYSEL ÇALIŞMALAR	148
7.1. Veri Önişleme Adımı Sonuçları	148
7.2. Model Parametreleri	156
7.3. Sınıflandırma Adımı Sonuçları.....	158
7.3.1. CIC-IDS2017 veri kümesi sınıflandırma sonuçları.....	159
7.3.2. UNSW-NB15 veri kümesi sınıflandırma sonuçları	162
7.4. Açıklanabilirlik Adımı Sonuçları	165
7.4.1. UNSW-NB15 veri kümesi açıklama katmanı sonuçları	165
7.4.2. CIC-IDS2017 veri kümesi açıklama katmanı sonuçları.....	174
7.5. Zeek Performans Sonuçları	178
7.6. LLM Tabanlı Açıklamalar.....	182
8. SONUÇ VE DEĞERLENDİRMELER	199
KAYNAKLAR	209
EKLER.....	255

Sayfa

EK-1. UNSW-NB15 veri kümesi çok sınıflı sınıflandırma karmaşıklık matrisleri ve ROC eğrileri	256
EK-2. UNSW-NB15 veri kümesi ikili sınıflandırma karmaşıklık matrisleri ve ROC eğrileri.....	263
EK-3. CIC-IDS2017 veri kümesi çok sınıflı sınıflandırma karmaşıklık matrisleri ve ROC eğrileri	270
EK-4. CIC-IDS2017 veri kümesi ikili sınıflandırma karmaşıklık matrisleri ve ROC eğrileri.....	277
ÖZGEÇMİŞ	284



ÇİZELGELERİN LİSTESİ

Çizelge	Sayfa
Çizelge 2.1. Literatürdeki çalışmalarla kıyaslama	11
Çizelge 2.2. AYZ içeren STS çalışmaları.....	15
Çizelge 2.3. AYZ içeren STS çalışmalarının değerlendirilmesi.....	22
Çizelge 3.1. STS veri kümeleri.....	57
Çizelge 4.1. AYZ teknikleri sınıflandırması.....	77
Çizelge 4.2. AYZ araç kutuları.....	82
Çizelge 5.1. CIC-IDS2017 veri kümesi günlere göre trafik içerikleri.....	97
Çizelge 5.2. CIC-IDS2017 veri kümesinin sınıflara göre dağılımı	97
Çizelge 5.3. CIC-IDS2017 veri kümesi öznitelik açıklamaları	98
Çizelge 5.4. UNSW-NB15 test ve eğitim veri kümeleri veri dağılımları.....	100
Çizelge 5.5. UNSW-NB15 tüm veri kümesine ilişkin veri dağılımı	101
Çizelge 5.6. UNSW-NB15 veri kümesi öznitelik açıklamaları	101
Çizelge 5.7. Zeek kurulum yapılan politika ve paketler	125
Çizelge 7.1. CIC-IDS2017 veri kümesi sınıf-etiket eşleştirmeleri	149
Çizelge 7.2. UNSW-NB15 veri kümesi sınıf-etiket eşleştirmeleri.....	149
Çizelge 7.3. LightGBM hiperparametreleri	158
Çizelge 7.4. GCN hiperparametreleri	158
Çizelge 7.5. CIC-IDS2017 çok sınıflı sınıflandırma sonuçları.....	160
Çizelge 7.6. CIC-IDS2017 ikili sınıflandırma sonuçları	160
Çizelge 7.7. UNSW-NB15 çok sınıflı sınıflandırma sonuçları	163
Çizelge 7.8. UNSW-NB15 ikili sınıflandırma sonuçları	164
Çizelge 7.9. CIC-IDS2017 veri kümesi için Zeek tarafından üretilen günlere göre trafik ve saldırı sayıları	181

Çizelge	Sayfa
Çizelge 7.10. UNSW-NB15 veri kümesi için Zeek tarafından üretilen trafik ve saldırı sayıları.....	181
Çizelge 7.11. CIC-IDS2017 ve UNSW-NB15 veri kümeleri için Zeek performans sonuçları.....	181



ŞEKİLLERİN LİSTESİ

Şekil	Sayfa
Şekil 2.1. 2018-2025 yılları arasındaki AYZ çalışmaları	6
Şekil 2.2. Araştırma metodolojisi	8
Şekil 3.1. Saldırıların hedefi ve boyutu	28
Şekil 3.2. Saldırı tespit sistemleri zaman çizelgesi	30
Şekil 3.3. Temel STS yapısı.....	32
Şekil 3.4. Saldırı türleri.....	34
Şekil 3.5. STS sınıflandırması	36
Şekil 3.6. STS mimari yapısı	37
Şekil 3.7. Anormallik tespiti yaklaşımının temel yapısı	44
Şekil 3.8. Normal ve saldırı içeren dağılımın teorik ayrışımı.....	45
Şekil 3.9. Anormallik temelli saldırı tespit sistemlerinin sınıflandırılması	46
Şekil 3.10. STS’lerin zorlukları	55
Şekil 3.11. STS blok şeması	58
Şekil 3.12. Sınıflandırma akış şeması	60
Şekil 4.1. Açıklanabilirlik ile performans arasındaki ilişki	63
Şekil 4.2. AYZ modellerinin YZ alanı ile ilişkisi.....	64
Şekil 4.3. Kara kutu modeli	65
Şekil 4.4. AYZ hedefleri.....	69
Şekil 4.5. Açıklama türleri.....	71
Şekil 4.6. AYZ taksonomisi.....	72
Şekil 4.7. Aşamalarına göre açıklanabilir YZ.....	74
Şekil 4.8. AYZ teknikleri.....	75
Şekil 5.1. CIC-IDS2017 veri kümesi sınıf dağılımı.....	107

Şekil	Sayfa
Şekil 5.2. UNSW-NB15 veri kümesi sınıf dağılımı	107
Şekil 5.3. LightGBM ve diğer gradyan artırma algoritmaları arasındaki ağaç büyüme yapısının karşılaştırılması	109
Şekil 5.4. Graf sinir ağları mimarisi.....	113
Şekil 5.5. Temel Zeek mimarisi	125
Şekil 5.6. Temel LLM akış diyagramı	128
Şekil 5.7. TRUBA mimarisi.....	130
Şekil 5.8. Karmaşıklık matris sonuçları	133
Şekil 6.1. Önerilen sistemin çalışma süreçleri	136
Şekil 6.2. Önerilen veri ön işleme adımı.....	141
Şekil 6.3. Önerilen sınıflandırma adımı	144
Şekil 6.4. Önerilen açıklanabilirlik adımı	152
Şekil 7.1. Deneysel akış şeması	148
Şekil 7.2. CIC-IDS2017 çok sınıflı veri kümesi RF özellik önemlerinin gösterimi.....	150
Şekil 7.3. CIC-IDS2017 ikili veri kümesi RF özellik önemlerinin gösterimi	150
Şekil 7.4. UNSW-NB15 çok sınıflı veri kümesi RF özellik önemlerinin gösterimi.....	151
Şekil 7.5. UNSW-NB15 ikili veri kümesi RF özellik önemlerinin gösterimi	152
Şekil 7.6. CIC-IDS2017 çok sınıflı veri kümesi Pearson korelasyonlarının gösterimi	152
Şekil 7.7. CIC-IDS2017 ikili veri kümesi Pearson korelasyonlarının gösterimi	153
Şekil 7.8. UNSW-NB15 çok sınıflı veri kümesi Pearson korelasyonlarının gösterimi	154
Şekil 7.9. UNSW-NB15 ikili veri kümesi Pearson korelasyonlarının gösterimi.....	155
Şekil 7.10. CIC-IDS2017 veri kümesi için veri dengeleme işlemi sonrası veri dağılımı	156
Şekil 7.11. UNSW-NB15 veri kümesi için veri dengeleme işlemi sonrası veri dağılımı	156

Şekil	Sayfa
Şekil 7.12. UNSW-NB15 veri kümesi örnek bir saldırı trafiği için GNNExplainer açıklama çıktısı	166
Şekil 7.13. UNSW-NB veri kümesi örnek bir saldırı trafiği için SHAP waterfall açıklama çıktısı	170
Şekil 7.14. UNSW-NB veri kümesi örnek bir saldırı trafiği için SHAP force açıklama çıktısı	171
Şekil 7.15. UNSW-NB15 çok sınıflı sınıflandırmada özniteliklerin sınıflara etkisinin gösterimi.....	171
Şekil 7.16. UNSW-NB veri kümesi örnek bir normal trafik için GNNExplainer açıklama çıktısı	172
Şekil 7.17. UNSW-NB veri kümesi örnek bir normal trafik için SHAP force açıklama çıktısı	172
Şekil 7.18. UNSW-NB veri kümesi örnek bir normal trafik için SHAP waterfall açıklama çıktısı	173
Şekil 7.19. CIC-IDS2017 veri kümesi örnek bir saldırı trafiği için GNNExplainer açıklama çıktısı	174
Şekil 7.20. CIC-IDS2017 veri kümesi örnek bir saldırı trafiği için SHAP waterfall açıklama çıktısı	177
Şekil 7.21. CIC-IDS2017 veri kümesi örnek bir saldırı trafiği için SHAP force açıklama çıktısı	178
Şekil 7.22. CIC-IDS2017 çok sınıflı sınıflandırmada özniteliklerin sınıflara etkisinin gösterimi.....	179

RESİMLERİN LİSTESİ

Resim	Sayfa
Resim 5.1. Akya-cuda kuyruğu için örnek SLURM betiği.....	131
Resim 5.2. Orfoz kuyruğu için örnek SLURM betiği	132
Resim 7.1. CIC-IDS2017 veri kümesi örnek conn.log dosyası	180
Resim 7.2. CIC-IDS2017 veri kümesi için Zeek tarafından üretilen log dosyaları	180
Resim 7.3. CIC-IDS2017 veri kümesi örnek notice.log dosyası	180
Resim 7.4. Açıklanabilirlik katmanı web arayüzü özet bölümü.....	182
Resim 7.5. Açıklanabilirlik katmanı web arayüzü ChatGPT bölümü	183
Resim 7.6. Resim 7.5'in genişletilmiş versiyonu	183

SİMGELER VE KISALTMALAR

Bu çalışmada kullanılmış simgeler ve kısaltmalar, açıklamaları ile birlikte aşağıda sunulmuştur.

Simgeler

db

Desibel

hz

Hertz

m³

Açıklamalar bir satırdan uzun olmamalıdır

m²

Metrekare

Kısaltmalar

Açıklamalar

AYZ

Açıklanabilir YZ

BDM

Büyük dil modelleri

CAGR

Birleşik yıllık büyüme oranı

CNN

Konvolüsyonel sinir ağı

CPS

Siber fiziksel sistem

DL

Derin öğrenme

DNN

Derin sinir ağları

DT

Karar ağacı

DVM

Destek vektör makineleri

GAT

Graf dikkat ağları

GCN

Graf konvolüsyonel ağlar

GenAI

Üretken yapay zeka

GIN

Graf izomorfizm ağları

IoT

Nesnelerin interneti

LightGBM

Light gradyan artırma makinesi

LR

Lojistik regresyon

LSTM

Uzun-kısa süreli bellek

ML

Makine öğrenmesi

MIB

Yönetim bilgi tabanı

Kısaltmalar**Açıklamalar****MLP**

Çok katmanlı algılayıcı ağ

NB

Naive Bayes

NLP

Doğal dil işleme

RF

Rasgele orman

SHAP

Shapley eklemeli açıklamalar

SOC

Siber operasyon merkezi

STS

Saldırı tespit sistemi

SVM

Destek vektör makinesi

YSA

Yapay sinir ağları

YZ

YZ

1. GİRİŞ

Son yıllarda ağ ve internet teknolojileri, bulut hesaplama ve özellikle Nesnelerin İnterneti (IoT) teknolojilerinin hızla gelişmesiyle birlikte birbirine bağlı bilgi sistemlerinin boyutu giderek artmaktadır. Günümüzde milyarlarca insan birçok farklı bilgi sistemi üzerinden internete erişerek çeşitli bilgilerini transfer etmektedir. Giderek büyüyen ağlarda taşınan kişi ve kurumlara özgü hassas verilerin miktarı da her geçen gün artmaktadır. Bununla birlikte, siber saldırılarda kullanılan araçların giderek çeşitlenmesi, gelişmesi ve kullanımı için gereken bilgi seviyesinin azalmasıyla birlikte her geçen gün birçok farklı kişi, kurum ve kuruluşa çok çeşitli ölçeklerde siber saldırılar yapılmaktadır. Bunun sonucunda da saldırı ve savunma arasındaki amansız mücadelenin dozu her geçen gün artarken bilgi güvenliğinin önemi de her geçen gün giderek daha önem kazanmaktadır.

Bilgi güvenliğinin sağlanmasında çeşitli güvenlik yaklaşımları bulunmaktadır. Bilgi gizliliği ve bütünlüğünün korunmasında genel olarak kriptografik yaklaşımlar kullanılırken bilgi erişilebilirliğinin korunmasına yönelik güvenlik duvarları, antivirüsler, DoS mitigation çözümleri ve saldırı tespit sistemleri (STS) gibi birçok farklı güvenlik yaklaşımı kullanılmaktadır. Bu araçların en güncel teknikler kullanılarak geliştirilmesi, özelleştirilmiş kural ve politikalara göre yapılandırılması, yapılan saldırıların büyük ölçüde önlenmesine ve bilgi güvenliğinin sağlanmasında bizlere yardımcı olmaktadır. STS'ler, bilgi sistemlerine yapılan saldırılara karşı önlem almamızı sağlayan, saldırıları tespit edip bunu uyarı veya alarm üreterek bildiren veya raporlayan yazılım ve donanımlardır. Bilgi sistemlerinde STS'lerin kullanılması ile sistemlere yapılan yetkisiz erişimler ve kötüye kullanımlar tespit edilerek, bunların yol açabileceği zararlar mümkün olduğunca erken ve otomatik olarak engellenmiş olur. Siber ortamdaki tehdit ve tehlikelerin sistemlere çoğunlukla dışarıdan gelmesinden dolayı, tehditleri ve saldırıları bu ortamların girişinde tespit etmek çok önemlidir. Bu yüzden, STS'ler her geçen gün daha fazla önem kazanmaktadır. Özellikle Yapay Zeka (YZ) ve derin öğrenme gibi gelişen teknolojilerle donatılan STS'ler bilgi ve bilgi sistemlerinin güvenliğinin sağlanmasında her geçen gün önemli bir aktör olmaktadır. Zeki saldırı tespit sistemleri olarak adlandırabileceğimiz YZ tekniklerini kullanan STS'ler giderek daha popüler ve etkin bir savunma mekanizması olarak siber güvenliğin sağlanmasına önemli katkı sunmaktadır.

Geçtiğimiz on yıl boyunca YZ ve makine öğrenmesinin popülaritesinde ve kullanımında önemli bir artış olmuştur. YZ günlük hayatta birçok farklı alanda her geçen gün daha yaygın olarak kullanılmaya ve gelişmeye devam etmektedir. Grand View Research tarafından yapılan bir araştırmada 2022'de küresel YZ pazarının boyutunu 136,55 milyar ABD doları olarak değerlendirilmiş ve 2023'den 2030'a kadar %37,3'lik bir bileşik yıllık büyüme oranı (CAGR) tahmin edilmiştir [1]. Next Move Strategy Consulting tarafından yapılan araştırmada ise 2021'de küresel açıklanabilir YZ pazarının boyutunu 4,4 milyar ABD doları olarak değerlendirilmiş ve 2022'den 2030'a kadar %18,4'lik bir bileşik yıllık büyüme oranı (CAGR) ile 21 milyar ABD dolarına çıkacağını tahmin etmiştir [2]. Bu gelişimlere paralel olarak saldırı tespit sistemlerinde de YZ kullanımı önemli ölçüde artmakta ve bunun sonucunda da STS'ler her geçen gün daha fazla gelişmekte, yaygın olarak kullanılmakta ve siber güvenlik stratejilerinin vazgeçilmez bir parçası haline gelmektedir. Bu eğilimin sonucunda derin öğrenme gibi şaşırtıcı sonuçlar elde eden çok çeşitli yeni yaklaşımlar ve teknikler ortaya çıkmıştır. Son yıllarda ortaya atılan bu modeller, tahmin performansı (örneğin, Accuracy, Precision, Recall, F1-Score) açısından önceki modellere göre önemli ölçüde gelişim göstermiştir. Bununla birlikte, bu modellerin karmaşıklık, yorumlanabilirlik, açıklanabilirlik ve adillik gibi fazla ilgi görmemiş çeşitli özellikleri de bulunmaktadır. YZ teknolojilerinin, özellikle de sinir ağları modelleri gibi popüler, anlaşılabilir, karmaşık ve opak makine öğrenmesi (ML) modellerinin giderek daha fazla benimsenmesiyle, anlamak giderek zorlaşmaktadır. Bu durum siber güvenlik gibi bir alanda sorunu daha da kötüleştirmekte ve önemli hale getirmektedir çünkü önemli kararların nedenini açıklayamayan bir sisteme güvenmek ve kendi haline bırakmak bariz tehlikeler içermektedir. Sonuç olarak, güncel tekniklerin karmaşıklığı giderek artmakta ve bunun sonucunda da kolayca incelenemeyen, doğrulanamayan ve herhangi bir insanın kavrayamayacağı kadar karmaşık olan YZ/ML kara kutu modelleri ortaya çıkmaktadır. Bu tür modeller, çeşitli karmaşık sorunlarını çözme yeteneklerine sahiptir, ancak tahminlerin olumsuz etkilerinin olduğu durumlar nedeniyle güvenilmez kalırlar. Açıklanabilir Yapay Zeka (AYZ), bu sorunun üstesinden gelmek için daha yorumlanabilir, açıklanabilir ve anlaşılabilir YZ'ye geçişi önermektedir. Çözüm için bütünsel bir yaklaşım olan AYZ, YZ modellerini anlamak, kavramak ve yorumlamak için çeşitli yöntemler kullanmaktadır. Açıklanabilir YZ araştırma alanı, YZ'nin açıklanabilirlik ve yorumlanabilirlik sorununu çözmeyi ve YZ'nin kararlarını kullanıcılara kullanıcıların anlayabileceği terimlerle açıklamayı amaçlamaktadır. Uzmanların model tahminlerini doğrulamasına yardımcı olan karmaşık makine öğrenmesi modellerinin davranışı hakkında daha fazla ayrıntı ortaya

çıkarmaktadır. Bununla birlikte, YZ siber güvenlik, sağlık, finans, ulaşım, otonom sürüş, askeri ve hukuk gibi yüksek riskli alanlarda sıklıkla kullanılmaya başlandığından dolayı neden olabilecekleri tehlikeli sonuçlar nedeniyle paydaşların YZ'yi anlama ihtiyacı giderek artmaktadır. Bu anlama ihtiyacı, insanların YZ teknolojilerine güvenip benimsemeleri için aşılması gereken bir sorun olarak da araştırmacıların karşısında durmaktadır.

Son zamanlarda açıklanabilir YZ teknikleri diğer siber güvenlik araştırma alanlarında olduğu gibi STS'lerde de sıklıkla kullanılmaya başlamıştır. Siber güvenlik araştırmacıları, kullanıcıları, politika yapımcıları ve bu alandaki her türlü paydaşın, YZ'den uygun şekilde yararlanması için önemini anlayabilmesi gerekmektedir. AYZ, siber güvenlik alanındaki YZ modellerinin anlaşılabilirliğini ortaya koymada önemli bir rol oynamaktadır. Bu tez çalışmasının odak noktası, AYZ tekniklerini kullanan bir saldırı tespit sistemi mimarisi ortaya koymaktır. Bu amaçla özgün bir açıklanabilir saldırı tespit sistemi önerilmiştir. Ayrıca, çalışmanın bir diğer odak noktası, literatür araştırması neticesinde AYZ ve STS alanlarını kapsamlı ve detaylı olarak ele alarak daha fazla tartışma ve gelecek çalışmalar için sağlam bir temel sunmasıdır. Sonuç olarak çalışma kapsamında, özgün bir STS tasarımının yanı sıra AYZ ve STS çalışma alanları, AYZ'ye dayalı STS çözümleri, AYZ'ye olan ihtiyaç ve AYZ'nin STS'ler için önemini de içine alan geniş yelpazede bir kavramsal bilgi de sunulmuştur.

Bu tez çalışmasının temel amacı; yüksek performanslı, açıklanabilir ve güvenilir bir STS geliştirmek ve bu sistemin hem özellik düzeyinde hem de ilişkisel (graf) düzeyde karar mekanizmalarını açıklayabildiğini ortaya koymaktır. Bu doğrultuda belirlenen araştırma soruları şu şekildedir;

Araştırma Sorusu-1: Saldırı Tespit Sistemlerinde (STS) yapay zekâ kullanımının literatürdeki mevcut durumu nedir?

Son yıllarda hangi yapay zeka modelleri kullanılmaktadır?

Yapay zeka modellerinde ve bu çalışma kapsamında odaklanılan STS modellerinde karşılaşılan kara kutu problemi çözülebilir mi?

Tekil güçlü modeller mi yoksa hibrit mimariler mi, performans–kararlılık dengesini daha iyi sağlamaktadır? Hibrit sınıflandırma yaklaşımları nasıl tasarlanmaktadır ve hangi yöntemsel zayıflıklar tekrarlanmaktadır?

Hangi veri kümeleri ve değerlendirme metrikleri tercih edilmekte, kullanım eğilimleri nasıl değişmektedir?

STS problemlerinde tablosal veri yapısı için boosting (LightGBM), derin öğrenme ve graf tabanlı öğrenme yaklaşımlarından hangisi kararlı ve genellenebilir bir şekilde en uygun performansı sağlamaktadır?

Araştırma Sorusu-2: STS alanında Açıklanabilir Yapay Zeka (AYZ) kullanımının mevcut durumu nedir?

AYZ yöntemleri hangi düzeylerde (öznitelik-seviyesi, model-seviyesi, karar-seviyesi) uygulanmaktadır?

Hangi AYZ teknikleri (SHAP, LIME, GNNExplainer vb.) daha yaygın kullanılmaktadır?

Çok boyutlu (öznitelik + graf seviyesi) AYZ yaklaşımı kullanan çalışmaların sayısı ve niteliği nedir; temel boşluklar nelerdir?

Araştırma Sorusu-3: Büyük Dil Modelleri (BDM) STS süreçlerine nasıl entegre edilmektedir?

Literatürde BDM entegrasyonu hangi biçimlerde (ön-işleme, açıklama yorumlama, alarm filtreleme) yapılmaktadır?

BDM destekli açıklama yorumlamanın analist deneyimine katkısı nedir?

BDM kullanılarak yorumlanan AYZ çıktıları, teknik açıklamaların güvenlik analistleri açısından daha anlaşılır, bağlamsal ve operasyonel hâle gelmesini sağlayabilmekte midir?

Bu yaklaşım, mevcut STS çalışmalarındaki kullanıcı değerlendirmesi boşluğunu doldurmakta mıdır?

Araştırma Sorusu-4: YZ Modelleri ile açıklanabilirlik STS tasarımında güvenle kullanılabilir mi?

Literatürde ayrı ayrı ele alınan hibrit sınıflandırma, graf tabanlı öğrenme, çok boyutlu AYZ ve BDM entegrasyonunun tek bir STS çerçevesinde bütünleştirilebilir mi? Bu yeni bir çözüm olabilir mi?

Performans ve açıklanabilirliği birlikte optimize edebilir mi?

Ön-işleme ve veri sınıf dengeleme yaklaşımları yeni STS tasarımında etkili midir?

Graf tabanlı öğrenme (GCN) STS'lerde kullanılabilir mi?

LightGBM ve GCN'in birleştirildiği hibrit yaklaşım, hem örnek içi (feature-level) hem de örnekler arası (graph-level) bilgiyi bütünleştirerek tekil modellere kıyasla daha dengeli ve güvenilir STS kararları üretebilir mi?

Özellik düzeyinde SHAP ve graf düzeyinde GNNExplainer kullanan çok boyutlu açıklanabilirlik yaklaşımları daha etkin bir çözüm haline getirilebilir mi?

Araştırma Sorusu-5: YZ Modeli + AYZ Modeli + BDM + STS bütünleşik sistem tasarımı başarılı olarak yapılabilir mi?

Bütünleşik Sistem tasarımı ve performansı, kural ve politika tabanlı STS'lere (örneğin Zeek) kıyasla farklı veri kümelerinde daha yüksek başarımlar sağlar mı? Özellikle de farklı özellik dağılımlarına ve saldırı taksonomilerine sahip CIC-IDS2017 ve UNSW-NB15 veri kümeleri üzerinde yüksek performans sunar mı? Gerçek ortam simülasyonlarında aynı başarıyı sergiler mi?

Önerilen sistem veri ön-işleme adımı, hibrit sınıflandırma adımı ve açıklanabilirlik adımı olmak üzere üç temel adımdan oluşmaktadır. Çalışma kapsamında altı farklı yaklaşımdan

oluşan üç özgün algoritma geliştirilmiştir. Önerilen veri ön işleme adımı, öz nitelik seçimi ve veri dengeleme yaklaşımlarından oluşmaktadır. Bu adım sonucunda veri kümeleri sınıflandırma adımına hazır hale getirilmiştir. Önerilen hibrit sınıflandırma adımında LightGBM ve Graf Konvolüsyonel Ağ (GCN) yaklaşımlarının birleşiminden oluşan hibrit bir sınıflandırma algoritması geliştirilmiştir. Sınıflandırma adımı sonucunda yüksek performans elde edilmiştir. Açıklanabilirlik adımında önerilen sistemin açıklanabilirliğini ve güvenilirliğini artırmak için SHAP ve GNNExplainer AYZ yaklaşımlarından oluşan bir açıklanabilirlik algoritması geliştirilmiştir. Bununla birlikte, güncel teknoloji olan büyük dil modelleri(BDM) dördüncü bir algoritma olarak önerilen sisteme entegre edilmiştir. BDM entegrasyonu ile elde edilen açıklanabilirlik çıktıları BDM kullanılarak yorumlatılmış ve kullanıcılara daha etkin açıklamalar üretilmiştir. Önerilen sistemin yalnızca yüksek performansla çalışması değil, aynı zamanda güvenilir ve anlaşılabilir kararlar üretmesi de sağlanmıştır. Deneysel çalışmalar, CIC-IDS2017 ve UNSW-NB15 veri kümeleri üzerinde yürütülmüş, sonuçlar doğruluk, precision, recall, specificity, ROC-AUC ve F1-Skoru gibi metriklerle ölçülmüştür. Sınıflandırmalar; RF (Rastgele Orman), DT (Karar Ağacı), NB (Naive Bayes), LR (Lojistik Regresyon), MLP (Çok Katmanlı Algılayıcı), LSTM (Uzun Kısa Süreli Bellek), CNN (Evrişimsel Sinir Ağı), Gated Recurrent Unit (GRU), CNN-LSTM, CNN-GRU, eXtreme Gradient Boosting (XGBoost), LightGBM, Graph Convolutional Network (GCN), Graph Attention Networks (GAT), Graph Isomorphism Network (GIN), GraphSAGE, LightGBM-GIN, XGBoost-GCN, XGBoost-GIN ve önerilen model (LightGBM-GCN) olmak üzere literatürdeki yirmi farklı makine öğrenmesi, derin öğrenme ve hibrit yöntem kullanılarak gerçekleştirilmiştir. Bununla birlikte, çalışmada kullanılan veri kümeleri Zeek açık kaynak STS üzerinde çalıştırılarak gerçek bir STS ortamındaki performansları değerlendirilmiştir. Sonuç olarak, bu tez çalışmasında yalnızca performans odaklı değil; aynı zamanda güvenlik uzmanlarının karar süreçlerini denetleyebileceği, güvenilir ve açıklanabilir YZ sistemlerine geçişin önünü açan bütünsel bir çözüm sunulmuştur. Bu tez çalışmasında özgün bir açıklanabilir STS geliştirilmesinin yanı sıra, tanımlarını, terminolojilerini ve tekniklerini inceleyerek AYZ'ye kapsamlı bir genel bakış sunulmuştur. Bununla birlikte, bu tez çalışması kapsamında literatürdeki AYZ destekli STS çalışmalarını inceleyen en kapsamlı çalışma yapılmıştır. Bu çalışmanın temel katkıları şu şekilde özetlenebilir:

- 1) Bu tez çalışmasında belirlenen 5 ana araştırma sorusu ile bunlara bağlı olan alt araştırma sorunlarına cevaplar kapsamlı bir literatür çalışması sonucunda ve yapılan analizler ve

elde edilen bulgular kapsamında geliştirilen yöntemler, algoritmalar ve sistemden elde edilen sonuçlar dikkate alınarak cevaplandırılmıştır.

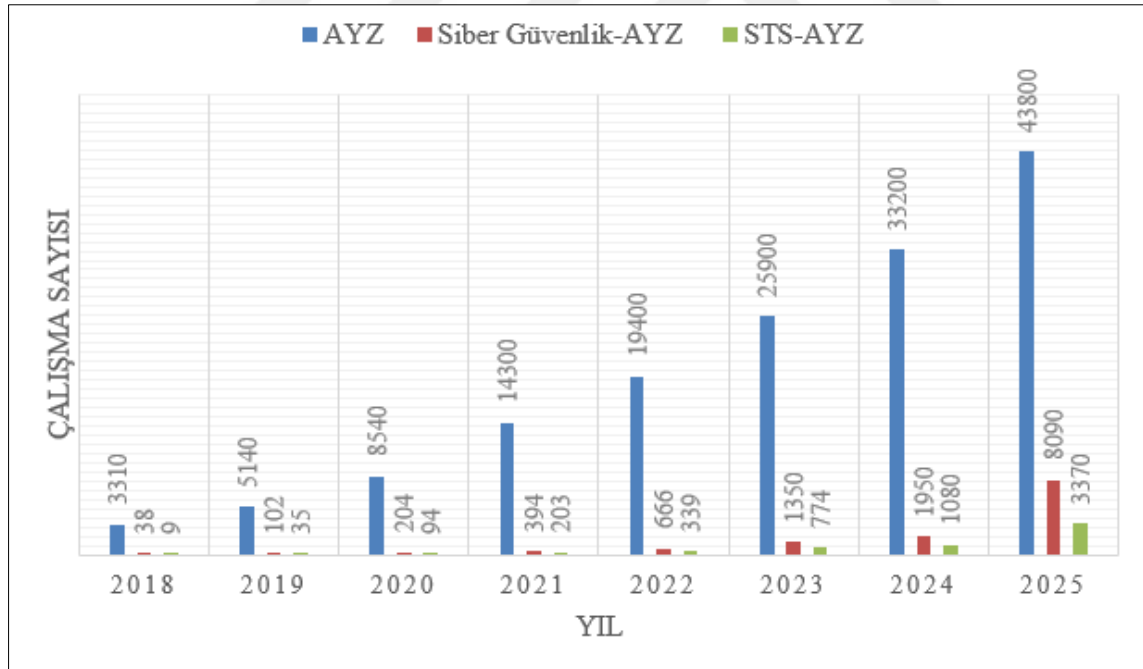
- 2) Literatürde AYZ destekli STS çalışmalarını kapsamlı olarak ele alan bir çalışma bulunmamaktadır. Bu tez çalışması kapsamında, özgün bir açıklanabilir STS modeli geliştirilmesinin yanı sıra 2018-2025 yılları arasındaki AYZ içeren STS çalışmaları kapsamlı olarak ele alınmıştır. Böylece, alandaki ilk kapsamlı çalışma yapılarak literatüre yani bir bakış açısı kazandırılmıştır.
- 3) Çalışma kapsamında YZ modellerindeki kara kutu problemine çözüm sunulmuştur.
- 4) Bu tez çalışmasında 3 temel adım, 4 farklı algoritma ve 6 farklı yaklaşımdan oluşan özgün ve çok boyutlu bir açıklanabilir saldırı tespit sistemi geliştirilmiştir.
- 5) Önerilen STS ile bir ağ altyapısındaki hem istatistiksel özellikleri hem de ilişkisel/topolojik yapıyı eş zamanlı öğrenebilen bir STS mimarisi geliştirilmiştir.
- 6) Önerilen sistemde hem hesaplama karmaşıklığını azaltmak hem de sınıflandırma doğruluğunu artırmak için çalışma kapsamında bir öznitelik seçme yaklaşımı geliştirilmiştir. Geliştirilen yaklaşım, Rastgele Orman ve Pearson Korelasyon Matrisini içeren iki farklı teknikten oluşmaktadır. CIC-IDS2017 ve UNSW-NB15 veri kümeleri üzerinde geliştirilen yaklaşım kullanılarak öznitelik seçimi yapılmıştır. Bu işlem sonucunda, sınıflandırma adımı için daha anlamlı öznitelikler seçilmiştir.
- 7) Önerilen sistemde veri dengesizliği problemini ele almak için veri dengeleme adımında SMOTE aşırı örnekleme ve Tomek-Links alt örnekleme tekniklerinden oluşan bir yaklaşım geliştirilmiştir. Bu yaklaşım sayesinde, hem aşırı örnekleme hem de yetersiz örnekleme tekniklerinin dezavantajları ortadan kaldırılmıştır.
- 8) Bu tez çalışmasında iki farklı yaklaşımdan oluşan hibrit bir sınıflandırma algoritması geliştirilmiştir. Önerilen hibrit sınıflandırma algoritması LightGBM ve Graf Konvolüsyonel Ağ (GCN) yaklaşımlarının birleşiminden oluşmaktadır. Böylece modelin, yalnızca örnek içi (feature-level) bilgiye değil, örnekler arası yapısal ilişkilere de dayalı karar vermesi sağlanmıştır.
- 9) Açıklanabilirlik adımında önerilen sistemin açıklanabilirliğini ve güvenilirliğini artırmak için SHAP ve GNNExplainer AYZ yaklaşımlarından oluşan çok katmanlı bir açıklanabilirlik algoritması geliştirilmiştir. Geliştirilen algoritma ile sisteme çok boyutlu bir açıklama yeteneği kazandırılmıştır. Böylece STS kararları şeffaflaştırılmıştır. Önerilen algoritma ile hem grafik tabanlı hem de öznitelik tabanlı açıklamalar üretilmiştir. Bu sayede hem verinin yapısal ilişkileri öğrenilmekte hem de geleneksel modellere göre daha derinlemesine saldırı desenleri tanımlanabilmektedir.

- 10) Geliştirilen sisteme güncel BDM teknolojileri entegre edilerek sistemin açıklanabilirliği daha da artırılmıştır. BDM desteği ile elde edilen çıktılar yorumlatılmış ve kullanıcılara daha etkin açıklamalar üretilerek bir karar destek mekanizması oluşturulmuştur.
- 11) Geliştirilen sistem, 19 farklı makine öğrenmesi ve derin öğrenme tekniğiyle karşılaştırılmıştır. Deneysel sonuçlar, önerilen modelin literatürdeki mevcut ve hibrit yöntemlerden daha iyi performans sergilediğini göstermiştir.
- 12) Literatürdeki çalışmalardan farklı olarak bu tez çalışmasında önerilen sistemde kullanılan veri kümeleri gerçek dünya uygulamalarında kullanılan Zeek açık kaynak STS üzerinde çalıştırılarak gerçek bir STS ortamındaki performansları değerlendirilmiştir.
- 13) Gelecekteki STS, AYZ ve BDM tabanlı güvenlik çalışmalarına referans olacak bir mimari tasarım gerçekleştirilmiştir.

Bu tez çalışmasında, Bölüm 2’de literatürde son 5 yıldaki AYZ destekli saldırı tespit sistemlerini içeren çalışmalar kapsamlı olarak ele alınmıştır, Bölüm 3’de saldırı tespit sistemleri Bölüm 4’de ise açıklanabilir YZ detaylı olarak ele alınmıştır. Bölüm 5’de kullanılan materyal ve metotlar hakkında detaylı bilgiler sunulmuştur. Bölüm 6’da geliştirilen yöntem detaylı olarak anlatılmıştır. Bölüm 7’de elde edilen deneysel sonuçlar sunulmuştur. Son olarak, Bölüm 8’de elde edilen sonuçlar tartışılmış, değerlendirilmiş ve gelecekte yapılacak olan çalışmalar için öneriler sunulmuştur.

2. MEVCUT ÇALIŞMALARIN İNCELENMESİ

Literatürde AYZ üzerine çok az çalışma yayınlanmıştır. Özellikle siber güvenlik alanında çok az sayıda çalışma bulunmaktadır. Bununla birlikte, AYZ ve siber güvenlik birçok çalışmada ayrı ayrı ele alınmıştır. Ayrıca, literatürde STS'lerde AYZ kullanımına özel olarak odaklanan kapsamlı bir araştırma bulunmamaktadır. Genellikle siber güvenliği ele alan çalışmalarda kısa bir alt başlık olarak değerlendirilmiştir. Şekil 2.1, 2018-2025 yılları arasında Google Akademik’de tüm AYZ çalışmalarının, siber güvenlik alanındaki AYZ çalışmalarının ve STS alanındaki AYZ çalışmalarının arama sonuçlarını göstermektedir. Şekil 2.1 incelendiğinde hem AYZ hem de STS alanlarını birlikte ele alan çok az çalışma olduğu görülebilir. Örneğin, 2023 yılında tüm AYZ çalışmaları arasında STS'yi içeren çalışmaların yüzdesi yaklaşık %3'tür. Bu tez çalışması, açıklanabilir YZ teknikleri kullanarak bir STS sisteminin geliştirilmesinin yanı sıra STS'lerde uygulanan çeşitli AYZ tekniklerine kapsamlı bir giriş sağlayarak literatürdeki bu boşluğu da doldurmaktadır.

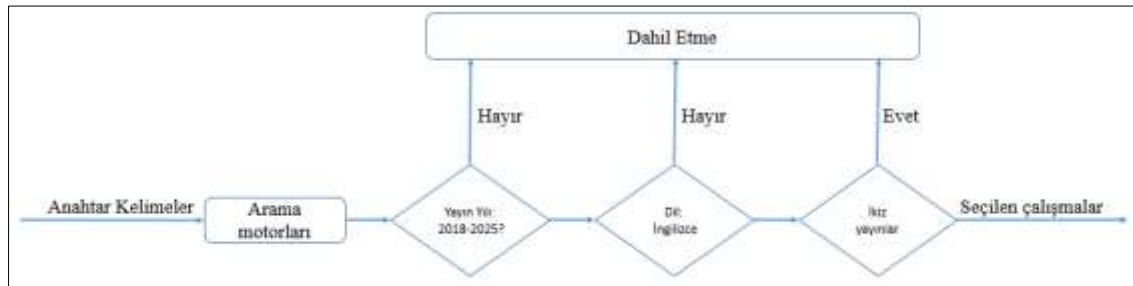


Şekil 2.1. 2018-2025 yılları arasındaki AYZ çalışmaları

Bu tez çalışması kapsamında, AYZ hakkında kapsamlı ve güncel bir arka plan bilgisi oluşturularak saldırı tespit sistemlerinde AYZ uygulamalarına ilişkin literatürdeki mevcut durum ve çalışmalar da incelenmiştir. Bu amaçla, Şekil 2.2’de gösterilen araştırma metodolojisi izlenerek, hakemli konferanslarda, dergilerde, ön baskılarda ve web sitelerinde

yayınlanmış saygın makaleler ve çalışmalar dikkatlice seçilerek referans seti oluşturulmuştur. İncelenen araştırma makalelerinin toplanması için aşağıdaki kriterler belirlenmiştir:

- 1) İlk olarak, Google Scholar [3], IEEE Xplorer [4], Springer [5] ve ScienceDirect [6] gibi arama motorlarında temel anahtar kelimeler olan "XAI" ve "IDS" kullanılmıştır. Ayrıca, "Explainable Artificial Intelligence", "Explainable AI", "Explainable ML", "Interpretable ML", "Interpretable Artificial Intelligence", "Interpretable AI" ve "Intrusion Detection System" gibi eşanlamlı anahtar kelimeler kullanılarak arama sonuçları elde edilmiştir.
- 2) Belirtilen anahtar kelimeler kullanılarak, arama motorlarında ("Explainable Artificial Intelligence" OR "XAI") AND ("IDS" OR "Intrusion Detection System") gibi gelişmiş arama yöntemleri kullanılmış ve sonuçlar elde edilmiştir. Seçim kriterleri, hem AYZ hem de STS'yi birlikte ele almak ve açıklanabilirlik özelliklerine sahip bir STS tasarım çalışması olmak üzere tanımlanmıştır.
- 3) Saldırı tespit sistemlerinde AYZ tekniklerinin uygulanmasındaki en son eğilimler hakkında bir rapor oluşturmak üzere 2018-2025 yılları arasında yayınlanmış çalışmalar seçilmiştir.
- 4) Akademik arama motorları, Medium ve GitHub gibi birçok web sitesi, özellikle güncel AYZ teknikleri ve araç kutuları açısından incelenmiştir.
- 5) Çalışma kapsamında yalnızca İngilizce yayınlar tercih edilmiştir.
- 6) İlgili araştırmaların kapsamlı bir şekilde kapsanmasını sağlamak için incelenen çalışmaların atıfları da takip edilmiştir.
- 7) Sonuçlar, en saygın çalışmaları vurgulamak için dikkatlice filtrelenmiştir. Filtrelenen çalışmaların başlıkları ve özetleri tarandı ve seçim kriterlerine göre seçildi ve seçilen çalışmalar kapsamlı bir şekilde incelendi.



Şekil 2.2. Araştırma metodolojisi

2.1. Literatürdeki Derleme Çalışmalarının İncelenmesi

Literatürde, siber güvenlik alanında YZ kullanımını inceleyerek araştırmacılara rehberlik etmeyi amaçlayan birtakım çalışmalar bulunmaktadır. Örneğin, Veiga vd. [7], ağ güvenliğinde YZ'nin sunduğu yenilikçi çözümleri ve avantajları ayrıntılı olarak açıklayarak, YZ'nin mevcut güvenlik sistemlerinin etkinliğini artırabileceğini göstermektedir. Çalışma, ağ güvenliğinde YZ'nin potansiyelini daha iyi anlamak ve bu alandaki gelecekteki yeniliklere rehberlik etmek için önemli bilgiler sağlamaktadır. [8]'te Wickramasinghe vd. siber-fiziksel sistemlerin (CPS) güvenliğini sağlamada derin öğrenmenin genel uygulamalarını incelemiştir. Derin öğrenmenin CPS güvenliğine nasıl entegre edilebileceğine, mevcut yaklaşımlara ve bu yaklaşımların geliştirilebilirliğine odaklanılmışlardır. Siber güvenlikte YZ kullanımına ilişkin ayrıntılı bir çalışma olarak Shaukat vd. [9], son on yılda siber güvenlik alanında kullanılan ML tekniklerinin kapsamlı bir incelemesini sunmaktadır. Bu çalışma, ML tekniklerinin siber güvenlikte nasıl uygulandığını ve bu alandaki gelişmeleri inceleyerek, denetlenen ve denetlenmeyen öğrenme tekniklerinin saldırı tespiti, kötü amaçlı yazılım tespiti ve dolandırıcılık tespiti gibi birçok alanda kullanıldığını göstermektedir.

YZ tabanlı STS'lerin siber güvenlik alanında kaydettiği önemli ilerlemeler nedeniyle, STS'lerde YZ kullanımını yalnızca incelemeye ve analiz etmeye odaklanan birçok çalışma bulunmaktadır. Örneğin, Mishra vd. [10], ağ güvenliğinin kritik bileşenleri olan STS'lerde ML tekniklerinin nasıl kullanıldığına odaklanmıştır. Çalışma, özellikle denetlenen, denetlenmeyen ve yarı denetlenen öğrenme olmak üzere ML tekniklerinin STS sistemlerine nasıl entegre edilebileceğini ve bu yaklaşımların avantajlarını ve sınırlamalarını ayrıntılı olarak analiz etmektedir. ML yöntemlerinin STS'lerde başarıyla uygulanabileceği ve bu sistemlerin saldırı tespitini otomatikleştirebileceği ve iyileştirebileceği gösterilmiştir. Gümüşbaş vd. [11], siber güvenlik ve STS için DL yöntemleri ve veritabanları hakkında kapsamlı bir inceleme sunmaktadır. STS'ler çalışmada ayrıntılı olarak ele alınmasa da, DL yöntemlerinin uygulanmasında kullanılan veri kümeleri daha kapsamlı bir şekilde ele alınmaktadır. STS'lerde YZ kullanımına ilişkin bir diğer çalışma Muneer vd. [12] tarafından yürütülmüştür. Bu çalışma, YZ tabanlı yaklaşımların STS'lerde nasıl kullanıldığını ayrıntılı olarak analiz ederek, YZ'nin STS'lerin etkinliğini artırmak için güçlü bir araç olduğunu vurgulamaktadır.

Siber güvenlik alanındaki kararların kritik doğası nedeniyle, birçok çalışma AYZ'yi siber güvenlik perspektifinden araştırmanın önemini vurgulamıştır. Kuppa ve Le-Khac [13], AYZ yöntemlerinin YZ sistemlerinin güvenilirliğini artırmak için nasıl kullanılabileceğini ve düşmanca saldırılara karşı nasıl savunma geliştirilebileceğini incelemiştir. Çalışma, AYZ'nin siber güvenlikteki potansiyelini ortaya koyarken aynı zamanda bu alandaki mevcut zorlukları ve gelecekteki iyileştirmeleri de vurgulamaktadır. [14]'de sunulan bir başka çalışma, siber güvenlikte AYZ'nin mevcut durumunu, karşılaşılan zorlukları, çözülmemiş sorunları ve gelecekteki araştırma yönlerini kapsamlı bir şekilde ele almaktadır. Capuano vd. [15], siber güvenlikte AYZ'nin önemini vurgulamakta ve mevcut teknikleri, uygulama alanlarını, avantajlarını ve sınırlamalarını analiz etmektedir. Çalışma ayrıca, siber güvenlikte AYZ kullanımını daha da teşvik etmek için gelecekteki araştırmalar için öneriler sunmaktadır. Benzer şekilde, Charmet vd. [16], siber güvenlikte AYZ hakkında kapsamlı bir literatür taraması gerçekleştirerek, AYZ 'nin yapay zeka tabanlı siber güvenlik sistemlerinde nasıl kullanıldığını, avantajlarını, zorluklarını ve gelecekteki araştırma yönlerini incelemiştir. Siber güvenlikte AYZ kullanımına ilişkin bir başka çalışma da Zhang ve arkadaşları tarafından yürütülmüştür [17]. Bu çalışma, siber güvenlikte kullanılan AYZ tekniklerini, bu tekniklerin sunduğu avantajları ve karşılaştıkları zorlukları ele alarak bu alandaki mevcut araştırma durumuna genel bir bakış sunmaktadır. Ayrıca, AYZ 'nin siber güvenlikte nasıl daha da geliştirilebileceğine dair öneriler de sunmaktadır. Yukarıda bahsedilen çalışmalara benzer şekilde, Rjoub ve arkadaşları [18] da siber güvenlikte AYZ'nin mevcut durumu ve bu alandaki en son gelişmeler hakkında kapsamlı bir tartışma sunmaktadır.

Birçok farklı kritik uygulama alanında açıklanabilirliğe olan ihtiyacın artması nedeniyle, açıklanabilirliği kavramsal olarak kapsamlı bir şekilde ele alan ve derinlemesine inceleyen çok sayıda çalışma yürütülmüştür. Rudin [19], çalışmasında kara kutu modellerine göre açıklanabilir modellerin tercih edilmesinin neden kritik olduğunu göstermekte ve açıklanabilir modellere dayalı yaklaşımlar önererek bu modellerin özellikle yüksek riskli karar süreçlerinde tercih edilmesi gerektiğini savunmaktadır. Açıklanabilir modellerin şeffaflık, güven ve doğruluk sağladığını, aynı zamanda denetim ve hata düzeltme süreçlerini kolaylaştırdığını belirtmektedir. Sahakyan vd. [20], özellikle tablo verileri için AYZ yöntemlerinin kullanımı hakkında kapsamlı bir inceleme sunmuştur. Çalışma, tablo verileri için mevcut AYZ tekniklerini inceleyerek hangi yöntemlerin kullanıldığını, avantajlarını ve sınırlamalarını analiz etmiştir. Alicioglu ve Sun [21], AYZ yöntemlerinin görsel analitiklerle

nasıl desteklendiği hakkında kapsamlı bir inceleme sunmuştur. Çalışma, AYZ yöntemleriyle entegre edilmiş görsel analitik tekniklerini keşfetmekte, bu tekniklerin avantajlarını, karşılaşılan zorlukları ve potansiyel gelecekteki araştırma alanlarını analiz etmektedir. [22], Endüstri 4.0'daki yapay zeka uygulamalarının nasıl açıklanabilir hale getirilebileceğini inceleyen bir literatür taraması yürütmüştür. Endüstri 4.0'daki YZ uygulamalarının AYZ 'ye dönüştürülmesinde hangi AYZ tekniklerinin kullanılabileceğini, bunların nasıl uygulanabileceğini ve bu alandaki fırsatları ayrıntılı olarak açıklamaktadır. Kök vd. [23], AYZ yöntemlerinin Nesnelerin İnterneti'nde (IoT) kullanılan YZ modellerini daha şeffaf hale getirmek için nasıl uygulanabileceği konusunda kapsamlı bir literatür taraması sunmuş, AYZ'nin IoT'de karşılaştığı zorluklar vurgulanmış ve bu alanda gelecekteki araştırmalar ile ilgili önerilerde bulunmuştur. Chamola vd. [24], güvenilir ve açıklanabilir YZ hakkında sunulan kapsamlı incelemede, güvenilir YZ oluşturma sürecinde AYZ 'nin neden kritik olduğunu, güvenilirliğin bileşenlerini ve AYZ 'nin YZ sistemlerinde nasıl kullanılabileceğini ana hatlarıyla açıklamışlar, özellikle de sağlık, finans, hukuk ve otonom sistemler gibi alanlarda şeffaflık, güvenlik ve adalet sağlamada AYZ 'nin rolünü vurgulamışlardır. Lettrache ve Ramdani [25], modelden bağımsız AYZ yöntemleri hakkında kapsamlı bir literatür taraması ve vaka çalışmasında, modelden bağımsız AYZ yöntemlerini tanıtmış, bu yöntemlerin nasıl çalıştığını açıklamış, farklı YZ modellerine uygulanabilirliklerini incelemişlerdir. Ayrıca, bu tekniklerin bir vaka çalışması yoluyla gerçek dünya senaryolarında nasıl uygulanabileceğini göstermişlerdir. Mohale vd. [26], saldırı tespit sistemlerinin kara kutu hâline gelen YZ ve ML modelleriyle çalışmasının yarattığı şeffaflık ve yorumlanabilirlik sorunlarını ele almışlardır. Çalışmada, bu sorunun çözümünde öne çıkan yaklaşım olarak AYZ yöntemlerinin STS'e entegrasyonunun nasıl gerçekleştirildiği ve hangi etkilere sahip olduğuna dair literatürdeki durum irdeleniyor. Mamta ve Gaurav [27], IoT (Internet of Things) ağlarında kullanılan STS bağlamında teknoloji füzyonu kavramını ele alıyor. Yani heterojen IoT cihazlarının, çeşitli iletişim protokollerinin ve çok yönlü veri kaynaklarının birleştiği ortamda, STS çözümlerinin nasıl daha güçlü, ölçeklenebilir ve güvenilir hâle getirilebileceğini araştırıyor. Stanek vd. [28], AYZ alanında güncel durumu ve ileri yönelimleri ele almayı amaçlıyor. Yazarlar, özellikle güvenlik, savunma ve kritik altyapılar bağlamında AYZ'nin önemine dikkat çekiyorlar. Çalışma, AYZ'nin hem teknik hem de uygulamalı boyutlarını incelemişlerdir. Riba vd. [29], siber güvenlik alanında giderek artan saldırı çeşitliliği ve trafik hacmi nedeniyle modern STS çözümlerinde makine öğrenimi (ML) ve derin öğrenme (DL) yöntemlerinin kullanımını sistematik biçimde inceleyen kapsamlı bir derleme çalışması yapmışlardır. Saeed ve Mehmood [30], kurumsal

2.2. Literatürdeki AYZ İçeren STS Çalışmalarının İncelenmesi

Bu bölümde, literatürdeki AYZ destekli STS çalışmaları ayrıntılı olarak ele alınmış ve bu çalışmalardan elde edilen bilgiler Çizelge 2.2'de sunulmuştur. Ayrıca, çalışmalar araştırmacılar tarafından kullanılan AYZ teknikleri, YZ modelleri, veri kümeleri, AYZ tekniklerinin modele özgü olup olmadığı ve uygulama aşaması gibi çeşitli açılardan incelenmiştir. Bu bakış açılarından mevcut çalışmaların bir özeti Çizelge 2.3'de sunulmuştur.

SHAP, bireysel özelliklere önem puanları atayarak model kararları için hem genel hem de yerel açıklamalar sağlama becerisi nedeniyle STS çalışmalarında yaygın olarak benimsenen bir AYZ yöntemidir. Çizelge 2.3'de belirtildiği gibi, SHAP 213 farklı STS çalışmasında tercih edilmiştir. SHAP'ın STS çalışmalarında diğer modellere kıyasla daha fazla kullanılması, hem yerel hem de genel özellik önem açıklamaları sunmadaki esnekliğini ve gücünü vurgulamaktadır. SHAP'ın bu özellikleri, özellikle derin sinir ağları gibi karmaşık modeller kullanıldığında, STS kararlarına olan güveni artırmak için güçlü bir çözüm olduğu kanıtlanmıştır.

LIME, karmaşık bir modeli yerel olarak yorumlanabilir bir modelle yaklaştırarak bireysel tahminleri açıklayan bir diğer popüler AYZ yöntemidir. LIME'ın bireysel tahminleri açıklama yeteneği, güvenlik uzmanlarının belirli örneklerin neden saldırı olarak sınıflandırıldığını anlamaları gereken durumlarda özellikle faydalıdır. Bu gelişmiş yetenek, onu SHAP'tan sonra literatürde en çok tercih edilen yöntem haline getirmiştir.

Özellikle, Rastgele Orman ve Karar Ağacı gibi kendinden açıklanabilen yöntemler, STS çalışmalarında açıklanabilirlik için de kullanılmıştır. Karar ağaçları doğası gereği yorumlanabilir oldukları için SHAP veya LIME gibi sonradan açıklanabilirlik yöntemlerine ihtiyaç duymazlar. Ancak, birden fazla ağaçtan alınan kararları birleştiren rastgele ormanlar için, SHAP gibi sonradan açıklanabilirlik yöntemleri, topluluktaki her bir özelliğin katkısını açıklamak için kullanılabilir. Çizelge 2.2'de görüldüğü gibi, literatürde SHAP, LIME ve kendinden açıklayabilen yöntemlerin yanı sıra, STS sistemlerinde kullanılan birçok farklı AYZ tekniği bulunmaktadır.

Literatürdeki çalışmaların çoğu, sonuçları açıklanabilir kılmak için daha önceden geliştirilmiş AYZ yöntemlerini kullanmıştır ve bu durumun bir sonucu olarak, çoğu

çalışmada yapılan açıklamalar sonradan yapılmıştır. Çalışmalarda en çok benimsenen yöntem SHAP'dir. SHAP'den sonra, çalışmalarda kullanılan bir diğer AYZ yöntemi LIME'dir. Ancak bazı çalışmalarda, doğası gereği beyaz kutu olan ve gerekli açıklamaları sağlayan karar ağacı ve rastgele orman gibi çeşitli yöntemler de kullanılmıştır.

Literatürde bulunan çalışmalarda, en çok benimsenen yöntemler sırasıyla RF, DT, DNN, XGBoost, ANN ve CNN teknikleri olmak üzere çok çeşitli YZ yöntemleri kullanılmıştır. Çizelge 2.3'de görüldüğü gibi, karar ağacı ve rastgele orman algoritmaları AYZ destekli STS çalışmalarında sıklıkla tercih edilmiştir. Karar ağaçları ve rastgele ormanlar, makine öğrenimindeki en yorumlanabilir YZ modelleri arasındadır ve bu nedenle AYZ çalışmalarında yaygın olarak kullanılmaktadırlar. Ayrıca, bu algoritmalar genellikle STS çalışmalarında tahmin performansını değerlendirmek için SVM, MLP, NB, KNN ve LR gibi geleneksel makine öğrenimi algoritmalarıyla birlikte kullanılır.

Geleneksel makine öğrenmesi algoritmalarının ardından, YSA, DNN, CNN, LSTM, GBM ve Autoencoder gibi derin öğrenme tabanlı teknikler, özellikle son yıllarda AYZ destekli STS sistemlerinde yaygın olarak tercih edilmektedir. Bu teknikler, tahmin performansı açısından geleneksel makine öğrenmesi tekniklerinden önemli ölçüde daha iyi performans gösterdikleri için tercih edilmektedir. DNN ve YSA algoritmaları, derin öğrenme tabanlı tekniklerin temelini oluştururken, CNN ve LSTM gibi özelleşmiş derin öğrenme modelleri de literatürde sıklıkla kullanılmaktadır.

XGBoost ve LightGBM gibi popüler varyantları da içeren Gradient Boosting Machines (GBM), tablo verileri için en iyi performans gösteren makine öğrenimi modelleri arasındadır. Bu modeller esasen karar ağaçlarına dayanmaktadır, ancak GBM modelleri doğası gereği karar ağaçlarından daha karmaşık olduğundan, SHAP veya LIME gibi sonradan açıklama yöntemleri gerektirirler.

CNN'ler, STS'de sıklıkla kullanılan bir diğer YZ tekniğidir. CNN'ler doğası gereği karmaşık ve yorumlanması zordur. SHAP ve LIME gibi algoritmalar genellikle CNN'lerin açıklanabilirliğini artırmak için kullanılırken, Grad-CAM (Gradyan Ağırlıklı Sınıf Etkinleştirme Eşlemesi) ve LRP (Katman Tabanlı İlgililik Yayılımı) gibi CNN'lere özgü yöntemler görsel açıklamalar oluşturmak için yaygın olarak kullanılır.

RNN'ler ve Uzun Kısa Süreli Bellek (LSTM) ağları gibi türevleri, sıralı verileri işlemek üzere tasarlanmıştır. Zaman serisi tahmini, konuşma tanıma, doğal dil işleme ve siber güvenlikte yaygın olarak kullanılırlar. Sıralı yapıları nedeniyle, RNN ve LSTM modellerini açıklamak zor olabilir. LIME gibi post-hoc yöntemler genellikle bir dizideki bireysel zaman adımlarının veya kelimelerin etkisini açıklamak için kullanılır. Ayrıca, RNN'lere sıklıkla entegre edilen dikkat mekanizmaları, modelin giriş dizisinin hangi kısımlarına daha fazla odaklandığını belirterek yerleşik bir yorumlanabilirlik biçimi sağlar.

Autoencoders, boyut azaltma, anomali tespiti ve veri sıkıştırma gibi görevler için kullanılan gözetimsiz sinir ağlarıdır. Giriş verilerini daha düşük boyutlu bir alana kodlayıp ardından orijinal haline geri döndürerek çalışırlar. Anomali tespitinde, autoencoders, anormal veriler için genellikle daha yüksek olan yeniden oluşturma hatasını ölçerek aykırı değerleri belirlemek için kullanılır. SHAP ve LIME, belirli veri noktalarının neden anormal olarak işaretlendiğini açıklamak ve modelin öğrendiği temel kalıplara dair iç görüler sağlamak için kullanılabilir.

Graf sinir ağları son yıllarda giderek popüler hale gelen bir YZ tekniğidir. Buna paralel olarak STS tasarımlarında da kullanılmaya başlanmıştır. Veriler arasında mesaj iletimi yöntemiyle çalışarak bağlamsal ilişkileri ortaya koymaktadır. GNNExplainer ve PGExplainer gibi graf sinir ağlarına özgü açıklanabilirlik teknikleri kullanılmaktadır.

Bölüm 3.5'de detaylı olarak açıklandığı üzere, literatürde STS çalışmalarında kullanılabilecek birçok veri seti bulunmaktadır. STS çalışmalarında en sık kullanılan veri seti NSL-KDD'dir. Bu veri seti daha eski bir veri seti olması ve yapısının birçok YZ modeline uygun olması nedeniyle daha çok tercih edilmektedir. NSL-KDD veri setini, son yıllarda araştırmacılar arasında giderek daha popüler hale gelen UNSW-NB15 ve CIC-IDS2017 veri setleri yaygın olarak kullanılmaktadır. Bu veri setlerine ek olarak, Çizelge 3.1'de gösterildiği gibi, literatürde uzmanlaşmış görevler için kullanılan birçok başka veri seti de kullanılmıştır. Çizelge incelendiğinde, en eski veri seti olmasına rağmen KDD99 veri setinin artık birçok çalışmada tercih edilmediği görülmektedir. Bunun nedeni, güncelliğini yitirmiş olması ve birçok modern saldırı türü ve yöntemini içermemesidir.

Çizelge 2.2 incelendiğinde, literatürdeki çalışmaların çoğunun SHAP ve LIME gibi modelden bağımsız yaklaşımlar kullandığı görülmektedir. Bununla birlikte, DT ve RF gibi

özünde açıklanabilir modellere ek olarak, CNN, DNN ve autoencoders gibi özellikle DL modelleri için geliştirilmiş özel açıklama yöntemleri kullanan çeşitli çalışmalar da mevcuttur.

Ayrıca, literatür incelendiğinde, IoT ağlarındaki saldırı tespit sistemlerinin açıklanabilirliği üzerine yapılan çalışma sayısının, IoT teknolojilerinin ve ağlarının hızla gelişmesine paralel olarak son yıllarda arttığı dikkat çekmektedir. Ayrıca, Çizelge 2.2'de açıkça görüldüğü gibi, AYZ tabanlı STS çalışmaları giderek daha fazla ilgi görmekte ve araştırmacılar tarafından her yıl daha fazla araştırma yürütülmektedir.

Saldırı tespit sistemlerinde YZ ve AYZ kullanımına ilişkin literatür, kullanılan model aileleri, tercih edilen veri kümeleri, değerlendirme metrikleri, AYZ yaklaşımlarının hangi düzeylerde uygulandığı ve tekrar eden yöntemsel zayıflıklar açısından Araştırma Sorusu-1 ve Araştırma Sorusu-2 kapsamında ayrıntılı biçimde incelenmiştir. Literatürdeki çalışmalar incelendiğinde AYZ-STs alanında bulunan bazı kritik boşluklar bu çalışma sayesinde ortadan kaldırılmaktadır. Literatürde LightGBM+AYZ yaygın olarak bulunmaktadır. Az da olsa GNN+AYZ yaklaşımları bulunmaktadır. Bununla birlikte önerdiğimiz LightGBM+GNN mimarisi literatürde bulunmamaktadır. Bu çalışma heterojen iki farklı model ailesini (tree-based + graph-based) birleştiren ilk AYZ-STs mimarilerinden biridir. Bununla birlikte, çalışmaların hemen hemen tamamı SHAP/LIME analizine odaklanmakla birlikte sınırlı sayıda GNNExplainer ile açıklanabilirlik sağlanmıştır. Bizim çalışmamız tree-based SHAP açıklamalarını + GNN temelli alt-graf açıklamalarını birleşik bir şekilde yorumlayan ilk sistematik yaklaşımı sunmaktadır. Bununla birlikte, bu tez çalışması AYZ-STs çıktılarının BDM destekli semantik açıklamalarını sunarak yeni bir araştırma yönü açmaktadır

Çizelge 2.2. AYZ içeren STS çalışmaları

Kaynaklar	Yıl	ML/DL Modelleri	Veri kümesi	AYZ Modeli	Lokal/Global	Modelle Özgü/Bağımsız	Post-hoc/İçsel
Mahmoud et al.[31]	2025	LightGBM	UNSW-NB15 CIC-IDS2017	SHAP	Global	Agnostik	Post-hoc
Shoukat et al.[32]	2025	LSTM-AE	CIC-IDS2017	SHAP	Global	Agnostik	Post-hoc
Fatema et al.[33]	2025	ANN	CICIoT2023	SHAP	Global	Agnostik	Post-hoc
Hulayyil et al.[34]	2025	DNN, NB, LR, RF, XGBoost, DT	CuSmartHome, IoT23 dataset	SHAP	Lokal, Global	Agnostik	Post-hoc
Hosayin et al.[35]	2025	XGBoost	WUSTL-EHMS-2020	SHAP, LIME	Lokal, Global	Agnostik	Post-hoc
Muhammad et al.[36]	2025	Decision Tree	UNSW-NB15	LIME, ELI5	Lokal, Global	Agnostik	Post-hoc

Çizelge 2.2. (devam) AYZ içeren STS çalışmaları

Wali et al.[37]	2025	Random Forest	CSE-CIC IDS 2018 CIC-IoT 23	SHAP	Lokal, Global	Agnostik	Post-hoc
Wang et al.[38]	2025	DT, LR	UNSW-NB15 CIC-IDS2017	SHAP, LIME	Lokal, Global	Agnostik	Post-hoc
Yu et al.[177]	2025	GNN	CADETS E3, THEIA E3	GNNE explainer	Lokal, Global	Spesifik	Post-hoc
Salloum and Norozpour [39]	2025	CNN	NSL-KDD	SHAP, LIME	Lokal, Global	Agnostik	Post-hoc
Nugraha et al.[40]	2025	XGBoost	CICIoT2023 CIC-DDoS2019 5 G PFCP	SHAP, LIME	Lokal	Agnostik	Post-hoc
Alamro et al.[41]	2025	CNN-BiGRU-AM	NSL-KDD CIC-IDS2017	SHAP	Lokal, Global	Agnostik	Post-hoc
Boubertakh et al.[42]	2025	DNN	IoTID20	SHAP	Global	Agnostik	Post-hoc
Harshitha et al.[43]	2025	DNN	CICIDS2017 CICIDS2018	SHAP, LIME	Lokal, Global	Agnostik	Post-hoc
Wang et al.[182]	2025	GAT	NSL-KDD, KDD99	GNNE explainer	Global	Spesifik	Post-hoc
Al-Hchaimi et al.[44]	2025	KNN, DT, RF, NB	-	SHAP, LIME	Lokal, Global	Agnostik	Post-hoc
SeonWoo et al.[183]	2025	GraphSAGE, AE	CIC-IDS2017 CIC-Edge-IIoT	GNNE explainer	Lokal, Global	Spesifik	Post-hoc
Tserenkhuu et al.[45]	2025	MLP, LSTM, CNN	X-IIoTID InSDN	SHAP, LIME	Global	Agnostik	Post-hoc
Osswald et al.[184]	2025	GCN	-	GNNE explainer	Lokal, Global	Spesifik	Post-hoc
Alatawi[46]	2025	LR	UNSW-NB15, NSL-KDD, CICIDS2019	SHAP, LIME	Lokal, Global	Agnostik	Post-hoc
Wu et al.[185]	2025	GCN, GAT, GraphSAGE	DARPA TC DARPA OpTC	Self Explainable	Global	Spesifik	İçsel
Shaker et al.[47]	2025	XGBoost	SCVIC-APT-2021	SHAP	Lokal, Global	Agnostik	Post-hoc
Alabbadi and Bajaber[48]	2025	CNN, DNN, TabNet	ToN_IoT	SHAP, LIME	Lokal, Global	Agnostik	Post-hoc
Mohale and Obagbuwa [49]	2025	DT, MLP, XGBoost, RF, CatBoost, LR, NB	UNSW-NB15	SHAP, LIME, ELI5	Lokal, Global	Agnostik	Post-hoc
Dasari et al.[50]	2025	DT, XGBoost	UNRID 2023	SHAP, LIME	Lokal, Global	Agnostik	Post-hoc
Uccello et al.[51]	2025	RF	CICIDS2017 WUSTL-IIOT-2021	SHAP	Global	Agnostik	Post-hoc
Ebrahimi et al.[52]	2025	CNN	ToN_IoT	SHAP, LIME	Lokal, Global	Agnostik	Post-hoc
Rahman[53]	2025	RF	Edge-IIoTset	SHAP	Lokal, Global	Agnostik	Post-hoc
Alabdulatif[54]	2025	ANN, SVM	NSL-KDD	SHAP	Lokal, Global	Agnostik	Post-hoc
Mohammed and Nema[55]	2025	XGBoost, RF, DT	UNSW-NB15	SHAP, LIME, ELI5	Lokal, Global	Agnostik	Post-hoc
Fernandes-Morales et al.[56]	2025	DT, RF, MLP, LSTM	NF-UQ-NIDS-v2	SHAP, LIME	Lokal, Global	Agnostik	Post-hoc
Boubertakh[57]	2025	DNN	IoTID20	SHAP	Global	Agnostik	Post-hoc
Garikapati et al.[58]	2025	RF, XGBoost, MLP	CIC-IDS2017	SHAP	Global	Agnostik	Post-hoc
Kalakoti et al.[59]	2025	LSTM	-	SHAP, LIME, IG, DeepLift	Lokal, Global	Agnostik	Post-hoc
Tewari and Singal[60]	2025	CNN, BiLSTM	CICIoT2023	SHAP, LIME	Lokal, Global	Agnostik	Post-hoc
Hermosilla et al.[61]	2025	XGBoost, TabNet	UNSW-NB15	SHAP, LIME	Lokal, Global	Agnostik	Post-hoc
Sadhwani et al.[62]	2025	CNN, LSTM, BiLSTM	UNSW-NB15 NSL-KDD ToN_IoT X-IIoTID	SHAP	Lokal, Global	Agnostik	Post-hoc
Dhruv et al.[63]	2025	DT, XGBoost, LR, KNN, RF, SVM	-	SHAP, LIME	Lokal, Global	Agnostik	Post-hoc

Çizelge 2.2. (devam) AYZ içeren STS çalışmaları

Almadhor et al.[64]	2025	DNN	NF-TON-IoT UNSW-NB15	SHAP	Global	Agnostik	Post-hoc
Saleh et al.[65]	2025	XGBoost	RTIoT2022	SHAP	Lokal, Global	Agnostik	Post-hoc
Xue et al.[66]	2025	NN, LSTM, Autoencoder	TON_IoT Gas Pipeline	SHAP, LIME, LEMNA	Lokal, Global	Agnostik	Post-hoc
Purshottam et al.[67]	2025	LightGBM	UNSW-NB15 CIC-IDS2017	SHAP	Lokal, Global	Agnostik	Post-hoc
Chatzimiltis et al.[68]	2025	LSTM	-	SHAP, LIME	Lokal, Global	Agnostik	Post-hoc
Kılınçer[69]	2025	CNN, BiLSTM	CL2-IDS	SHAP	Global	Agnostik	Post-hoc
Oyetodun et al.[70]	2025	CNN, LSTM, Autoencoder	ICS Dataset	SHAP, LIME	Lokal, Global	Agnostik	Post-hoc
Mohammed[71]	2025	SVM, RF, Autoencoders, CNN	SDN_Intrusion	SHAP	Lokal, Global	Agnostik	Post-hoc
Fatima et al.[72]	2025	RF,DT,DNN,MLP, XGBoost	IoTID20	SHAP, LIME	Lokal, Global	Agnostik	Post-hoc
Ahmad[73]	2025	LSTM	CICADA- IIoT2024	SHAP, LIME	Lokal, Global	Agnostik	Post-hoc
Shaker et al.[74]	2025	XGBoost	SCVIC-APT- 2021	SHAP	Global	Agnostik	Post-hoc
Eddine et al.[75]	2025	LSTM	X-IIoTID	SHAP, LIME	Lokal, Global	Agnostik	Post-hoc
Rawajbeh et al.[76]	2025	CNN-BiLSTM	ToN_IoT, Bot- IoT	SHAP	Lokal, Global	Agnostik	Post-hoc
Bilal et al.[77]	2025	Bi-GRU	CIC-IDS2017	SHAP	Lokal, Global	Agnostik	Post-hoc
Alabbadi and Bajaber [78]	2025	DNN, LSTM	TON_IoT NSL-KDD CICIoMT 2024	SHAP, LIME	Lokal, Global	Agnostik	Post-hoc
Toluwaleke et al.[79]	2025	Bi-GRU	CICIDS-2017 NSL-KDD 5G Network Dataset	SHAP, LIME	Lokal, Global	Agnostik	Post-hoc
Kumar et al.[80]	2025	IENN	CICIDS-2017 NSL-KDD	LIME	Lokal	Agnostik	Post-hoc
Khandan et al.[81]	2025	RF	X-IIoTID WUSTL-IIoT- 2021	SHAP, LIME	Lokal, Global	Agnostik	Post-hoc
Zhang et al.[82]	2025	DNN, NB, LR, LightGBM	NSL-KDD UNSW-NB15 ToN-IoT	SHAP	Lokal, Global	Agnostik	Post-hoc
Metwally and Zamzam[83]	2025	LR, KNN, RF, MLP, BiLSTM	UNSW-NB15	SHAP, LIME	Lokal, Global	Agnostik	Post-hoc
Bilal et al.[84]	2025	DNN	Edge-IIoT CIC-IoT2023 TII-SSRC-23,	SHAP, LIME	Lokal, Global	Agnostik	Post-hoc
Hemalatha et al.[85]	2025	RF, SVM, CNN	CIC-IDS2017	SHAP, LIME	Lokal, Global	Agnostik	Post-hoc
Wan et al.[86]	2025	NN	KDD99, NSL- KDD, UNSW- NB15, MQTT, Wustl-IIoT, UNR-IDD, CICIoT2023	SHAP	Global	Agnostik	Post-hoc
Bibers et al.[87]	2025	RF, LR, KNN, AdaBoost	CIC-IDS2017	SHAP, LOCO, PFI, DALEX, ProfWeight	Lokal, Global	Agnostik	Post-hoc
Krishnaveni et al.[87]	2025	BiLSTM	NSL-KDD, CICIDS-2018 ToN_IoT	SHAP	Global	Agnostik	Post-hoc
Nalinipriya et al.[89]	2025	Autoencoder	CICIDS-2017 NSL-KDD	LIME	Lokal	Agnostik	Post-hoc
Ahmed et al.[90]	2025	RF, XGBoost	NSL-KDD	SHAP, LIME	Lokal, Global	Agnostik	Post-hoc
Arreche et al.[91]	2025	DNN	NSL-KDD CICIDS-2017 RoEduNet- SIMARGL2021	LRP, IG, DeepLIFT	Lokal, Global	Agnostik	Post-hoc

Çizelge 2.2. (devam) AYZ içeren STS çalışmaları

Kumar et al.[92]	2025	DT, RF, LR, KNN, SVM	WSN-DS KDD99	SHAP, LIME	Lokal, Global	Agnostik	Post-hoc
Bacevicius et al.[93]	2025	DT, RF, KNN	CIC-IDS2017 CSE-CIC-IDS-2018	LIME	Lokal	Agnostik	Post-hoc
Ogunseyi and Thiagarajan[94]	2025	LSTM	NF-BoT-IoT-v2 IoTID20	SHAP, LIME	Lokal, Global	Agnostik	Post-hoc
Kumar and Achary[95]	2025	RL, GNN	CIC-IDS2017	SHAP	Global	Agnostik	Post-hoc
Yağız and Göktaş[96]	2025	VAE	Edge-IIoTset UKM-IDS20 CTU-13 NSL-KDD	Self Explainable	Global	Spesifik	İçsel
Alohali et al.[97]	2025	LSTM-BiGRU	NSLKDD CICIDS 2017	SHAP	Global	Agnostik	Post-hoc
Hoang et al.[98]	2025	RF, MLP, LSTM, GRU, RNN	CICIoT2023	SHAP, LIME	Lokal, Global	Agnostik	Post-hoc
Holubenko et al.[99]	2025	DT, RF, KNN, NB, MLP	ADFA-LD	SHAP, LIME	Lokal, Global	Agnostik	Post-hoc
Ahmad et al.[100]	2025	ETC, GRU	CICIoT2023	SHAP	Global	Agnostik	Post-hoc
Asal et al.[101]	2025	CVAE, DT, RF	WUSTL-IIoT-2021	SHAP, LIME	Lokal, Global	Agnostik	Post-hoc
Patel[102]	2025	KNN-RF	CICIDS 2017 CTU-NCC	SHAP, LIME	Lokal, Global	Agnostik	Post-hoc
Narkedimilli et al.[103]	2025	RF, XGBoost	CIC-IoV-2024 CIC-APT-IIoT-2024 EDGE-IIoT	LIME	Lokal	Agnostik	Post-hoc
Hewa et al.[104]	2025	XGBoost	UNSW-NB15	SHAP	Global	Agnostik	Post-hoc
Vishwakarma and Kesswani[105]		CNN, LSTM, RF	CIC IoT 2022 UNSW-NB15 CICIDS2018	SHAP, LIME	Lokal, Global	Agnostik	Post-hoc
Ericson et al.[106]	2025	CNN, LSTM, GCN	UNSW-NB15	SHAP	Global	Agnostik	Post-hoc
Kalutharage et al.[107]	2025	NN	USBIDS	SHAP	Global	Agnostik	Post-hoc
Hasan and Tasnim[108]	2025	DT, XGBoost, AdaBoost, LightGBM, KNN, LSTM, TabNet, LDA, QDA	ToN_IoT, DS2OS	SHAP, LIME	Lokal, Global	Agnostik	Post-hoc
Obidiagha et al.[109]	2025	Isolation Forest	CICIoT2023	SHAP, LIME	Lokal, Global	Agnostik	Post-hoc
Farooq et al.[110]	2025	RF	KDD99	SHAP, LIME	Lokal, Global	Agnostik	Post-hoc
Singh and Rathee[111]	2025	RF, AdaBoost, DNN, LightGBM, KNN, SVM	CICIDS-2017 NSL-KDD	SHAP, LIME	Lokal, Global	Agnostik	Post-hoc
Farhat et al.[112]	2025	XGBoost, CatBoost, RF, ExtraTrees	CICIDS2017, CICIDS2018, CIC-ToN-IoT, NF-UNSW-NB15-v2, NF-ToN-IoT-v3, NF-BoT-IoT-v3	SHAP, LIME	Lokal, Global	Agnostik	Post-hoc
Saied and Guirguis[113]	2025	DT, RF, XGBoost, AdaBoost	N-BaIoT	SHAP, LIME	Lokal, Global	Agnostik	Post-hoc
Soni et al.[114]	2025	RF, MLP, KNN, NB, LR, SVM	False data injection attack dataset	SHAP, LIME	Lokal, Global	Agnostik	Post-hoc
Hariharan et al.[115]	2025	CNN-LSTM	CIC-IDS 2017, CIC-ToN-IoT, UNSW-NB15,	LIME	Lokal	Agnostik	Post-hoc
Saravanan et al.[116]	2025	RF	IoT-23, LITNET-2020	Self Explainable	Global	Spesifik	İçsel
Idima et al.[117]	2025	XGBoost	-	SHAP, LIME	Lokal, Global	Agnostik	Post-hoc
Rajkumar and Shalinie[118]	2025	NN	CIC-IoT2022 SDN-DDoS24	SHAP	Lokal, Global	Agnostik	Post-hoc

Çizelge 2.2. (devam) AYZ içeren STS çalışmaları

Izuazu et al.[119]	2025	DNN	WUSTL-IIoT 2021 Edge-IIoT	SHAP, LIME	Lokal, Global	Agnostik	Post- hoc
Krishnan et al.[120]	2025	BiLSTM	Edge-IIoT	SHAP, LIME	Lokal, Global	Agnostik	Post- hoc
Sharma and Shambharkar[121]	2025	CNN, BiLSTM, GRU	CICIoMT 2024	SHAP	Lokal, Global	Agnostik	Post- hoc
Mittal et al.[122]	2025	CNN, LSTM, ANN, DT	Aposemat IoT-23 AIoT Malware detection dataset	LIME	Lokal	Agnostik	Post- hoc
Thulitha et al.[123]	2025	NB	MAWI	SHAP	Global	Agnostik	Post- hoc
Nazreen and Sangeetha[124]	2025	NB	CICIDS 2017, Honeypots, NSL- KDD	SHAP	Global	Agnostik	Post- hoc
Ahsan et al.[125]	2025	RF, CatBoost, LR	VeReMi	SHAP	Lokal, Global	Agnostik	Post- hoc
Ambritta et al.[126]	2025	RF	NSL-KDD, CICIDS 2017	SHAP	Global	Agnostik	Post- hoc
Sunkara[127]	2025	RF, XGBoost, DNN	CICIDS 2017 TON_IoT	SHAP, LIME	Lokal, Global	Agnostik	Post- hoc
Wakili and Bakkali[128]	2025	RF, LightGBM, CatBoost, Extra Trees, XGBoost	CICIoT2023	SHAP	Global	Agnostik	Post- hoc
Patel et al.[129]	2025	RF, XGBoost, SVM, DT, LR	-	LIME	Lokal	Agnostik	Post- hoc
Imtiaz et al.[130]	2025	CNN	UNSW-NB15, KDD99, Bot-IoT, TON_IoT	Grad-CAM, SHAP	Lokal, Global	Agnostik	Post- hoc
Khan et al.[131]	2025	CNN-GRU	CICIoV2024	SHAP	Lokal, Global	Agnostik	Post- hoc
Melendrez et al.[132]	2025	XGBoost, CatBoost	CICIDS 2017	SHAP	Lokal, Global	Agnostik	Post- hoc
Garoppo et al.[133]	2025	CNN	ToN-IoT	Grad-CAM	Lokal	Agnostik	Post- hoc
Saxena and Mittal[134]	2025	XGBoost, RF, DT, Extra Trees	Edge-IIoT	SHAP	Global	Agnostik	Post- hoc
Lau et al.[135]	2025	BiLSTM	5G-NIDD	SHAP	Lokal, Global	Agnostik	Post- hoc
Patel et al.[136]	2025	XGBoost, KNN, RF	CIC-IDS2017, CTU-NCC,	SHAP, LIME	Lokal, Global	Agnostik	Post- hoc
Gupta and Sahu[137]	2025	Autoencoder, BiGRU	Edge-IIoT	SHAP	Lokal, Global	Agnostik	Post- hoc
Randika et al.[138]	2025	RF, XGBoost, LightGBM	Edge-IIoT	SHAP	Lokal, Global	Agnostik	Post- hoc
Turgut and Basarslan[139]	2025	RNN, LSTM, GRU, BiLSTM, BiGRU	CICIoMT2024, IoMT- TrafficData, ECU-IoHT	SHAP, LIME	Lokal, Global	Agnostik	Post- hoc
Habibi and Sukarno[140]	2025	XGBoost	Edge-IIoT	SHAP	Lokal, Global	Agnostik	Post- hoc
Saheed et al.[141]	2025	BiLSTM	UNSW-NB15, X-IIoTID	SHAP	Lokal, Global	Agnostik	Post- hoc
Wazid et al.[142]	2025	ANN, CNN, CNN-LSTM, CNN-GRU	CICIoT2023	SHAP	Lokal, Global	Agnostik	Post- hoc
Chettier et al.[143]	2025	AE-RF-CNN-LSTM	ADFA-LD	SHAP	Global	Agnostik	Post- hoc
Shiddiqi et al.[144]	2025	RF, DT	N-BaloT	SHAP	Lokal, Global	Agnostik	Post- hoc
Gogoi et al.[145]	2025	SVM, RF, XGBoost	-	SHAP	Global	Agnostik	Post- hoc
Somayajula et al.[146]	2025	DT, SVM, ANN, FIS	NSL-KDD, CIC-IDS2017	SHAP, LIME	Global	Agnostik	Post- hoc
Mathew and Babu[147]	2025	GAT	NF-UNSW-NB15	SHAP, LIME	Global	Agnostik	Post- hoc
Ullah et al.[148]	2025	BiLSTM	NSL-KDD	SHAP, PFI	Lokal, Global	Agnostik	Post- hoc
Senthamizh et al.[149]	2025	CNN, LSTM	NSL-KDD, CICIDS2017 UNSW-NB15	SHAP, LIME	Global	Agnostik	Post- hoc

Çizelge 2.2. (devam) AYZ içeren STS çalışmaları

Ahmad et al.[150]	2025	LSTM	ToN-IoT	SHAP, PDP, ICE	Lokal, Global	Agnostik	Post-hoc
Fan et al.[151]	2025	Autoencoder	CIC IoT Dataset 2022 UNSW-NB15	Self Explainable	Global	Spesifik	İçsel
Alzakari et al.[152]	2025	CNN-BiGRU	Edge-IIoT, BoT-IoT	SHAP	Global	Agnostik	Post-hoc
Kumar et al.[153]	2025	LR, SVM, NB, RF, KNN, XGBoost	NF-ToN-IoT-V2	SHAP	Global	Agnostik	Post-hoc
Kaur and Gupta[154]	2025	LR, XGBoost, CNN, RF, KNN	CICIoT2023	SHAP, LIME	Lokal, Global	Agnostik	Post-hoc
Alsharaiah et al.[155]	2025	NN	CIC IoMT2024	SHAP	Lokal, Global	Agnostik	Post-hoc
Qian et al.[186]	2024	GAT, GraphSAGE, GCN	C2, P2P, Chord	GNNEExplainer	Lokal, Global	Spesifik	Post-hoc
Sun et al.[178]	2024	GCN, GAT	CIC-IDS2017	GNNEExplainer	Global	Spesifik	Post-hoc
Kaya et al.[179]	2024	CatBoost,	NF-UNSW-NB15-v2 NF-CSE-CIC-IDS2018-v2	PGExplainer	Global	Spesifik	Post-hoc
Hooshmand et al.[157]	2024	XGBoost	NSL-KDD UNSW-NB15	SHAP	Lokal, Global	Agnostik	Post-hoc
Ahmed et al.[156]	2024	SEM, BMA, CEM	CIC-IDS2017	SHAP, LIME	Lokal, Global	Agnostik	Post-hoc
Bellegdi et al.[158]	2024	RF, AdaBoost, XGBoost, LightGBM	CSE-CIC-IDS2018	SHAP, LIME	Lokal, Global	Agnostik	Post-hoc
Arreche et al.[159]	2024	RF, DNN, AdaBoost, MLP, KNN, SVM, LightGBM	CICIDS-2017 NSL-KDD RoEduNet-SIMARGL2021	SHAP, LIME	Lokal, Global	Agnostik	Post-hoc
Attique et al.[160]	2024	DNN	CIC-IDS2017	SHAP	Lokal, Global	Agnostik	Post-hoc
Kaya et al.[161]	2024	CatBoost	NF-UNSW-NB15-v2 NF-CSE-CIC-IDS2018-v2	PGExplainer	Lokal, Global	Agnostik	Post-hoc
Kumar et al.[162]	2024	Autoencoder	Kitsune attack dataset CIC-IDS2017	TRUSTEE	Global	Agnostik	Post-hoc
Gyawali et al.[163]	2024	DNN, XGBoost	CICIoT2023	LIME, SHAP, and counterfactual explanations	Lokal, Global	Agnostik	Post-hoc
Gummadi et al.[164]	2024	DT, DNN, SVM, AdaBoost, MLP	N-BaIoT, MEMS	SHAP, LIME, CEM, ALE, PFI, Profweight, LOCO	Lokal, Global	Agnostik	Post-hoc
Uysal and Kose[165]	2024	DT, RF, ET, SVM, XGBoost	CIC-IDS2017	SHAP, LIME	Lokal, Global	Agnostik	Post-hoc
Ahakonye et al.[166]	2024	DT, RF, DNN	WUSTL-IIoT	SHAP, LIME	Lokal, Global	Agnostik	Post-hoc
Mahmoud et al.[167]	2024	RF, DT, XGBoost, SGD, LinearSVC	CIC-IDS2017	SHAP	Lokal, Global	Agnostik	Post-hoc
Xu et al.[168]	2024	DNN	NSL-KDD, UNSW-NB15	SHAP, LIME, LEMNA	Lokal, Global	Agnostik	Post-hoc
Nugraha et al.[169]	2024	XGBoost	CIC-DDoS2019 CICIoT2023 d	SHAP, LIME	Lokal, Global	Agnostik	Post-hoc
Tai et al.[170]	2024	GAN	UNSW-NB15 CIC-IDS2017	SHAP	Global	Agnostik	Post-hoc
Haripriya and Jagadeesh[171]	2024	DNN	CSE-CIC-2018	SHAP, LIME	Lokal, Global	Agnostik	Post-hoc
Ananya et al.[172]		LR, KNN, DT, SVM	CIDDS-001	SHAP, LIME	Lokal, Global	Agnostik	Post-hoc
Atagun et al.[173]	2024	XGB, AdaBoost, CatBoost, DT	KDD99	SHAP	Lokal, Global	Agnostik	Post-hoc
Komarchesqui et al.[174]	2024	GAN-GRU	CIC-DDoS2019	SHAP	Lokal, Global	Agnostik	Post-hoc

Çizelge 2.2. (devam) AYZ içeren STS çalışmaları

Khediri et al.[175]	2024	RF	CIC-IDS2017	SHAP	Lokal, Global	Agnostik	Post-hoc
Mia et al.[176]	2024	DT, XGB, RF	CIC-IoT-2023 NF-UQ-NIDS-v2 HIKARI 2021	SHAP	Global	Agnostik	Post-hoc
Gaspar et al.[191]	2024	MLP	ADFA-LD	SHAP, LIME	Lokal, Global	Agnostik	Post-hoc
Hoosmand et al.[192]	2024	XGBoost	NSL-KDD, UNSW-NB15	SHAP	Global	Agnostik	Post-hoc
Maricar et al.[193]	2024	XGBoost	CIC-IDS2017	LIME	Lokal	Agnostik	Post-hoc
Mallampati and Seetha[194]	2024	LGBM	CIC-IDS2017 UNSW-NB15	SHAP	Lokal, Global	Agnostik	Post-hoc
Son et al.[180]	2023	GraphSAGE	DARPA TCE3	GNNEexplainer	Lokal, Global	Spesifik	Post-hoc
Sivamohan and Sridhar [195]	2023	BiLSTM	NSL-KDD, HoneyPot	LIME, SHAP, KHO	Lokal, Global	Agnostik	Post-hoc
Keshk et al.[196]	2023	LSTM	NSL-KDD, UNSW-NB15 and TON_IoT	SHAP, ICE, PDP, PFI	Lokal, Global	Agnostik	Post-hoc
Lanfer et al.[197]	2023	RF	NSL-KDD	SHAP	Global	Agnostik	Post-hoc
Siganos et al.[198]	2023	-	CIC-IoT-Dataset-2022, IEC 69870-5-104	SHAP	Lokal, Global	Agnostik	Post-hoc
Nguyen et al.[199]	2023	CNN, SAE	CSE-CIC-IDS2018	SHAP	Lokal, Global	Agnostik	Post-hoc
Hussein and Dosuky[200]	2023	RF, SVM, NB, MLP, CNN	CPS	SHAP	Lokal	Agnostik	Post-hoc
Almuqren et al.[201]	2023	IENN-based classification	CIC-IDS2017 NSL-KDD	LIME	Lokal	Agnostik	Post-hoc
Sharma et al.[202]	2023	DNN	NSL-KDD	LIME	Lokal	Agnostik	Post-hoc
Rehman et al.[203]	2023	CNN	NSL-KDD	LIME	Lokal	Agnostik	Post-hoc
Djenouri et al.[204]	2023	LSTM	NSL-KDD	SHAP	Lokal	Agnostik	Post-hoc
Pande and Khamparia[205]	2023	DNN	NSL-KDD, UNSW-NB15	SHAP	Lokal, Global	Agnostik	Post-hoc
Sharma et al.[206]	2023	DNN, CNN	NSL-KDD, UNSW-NB15	LIME, SHAP	Lokal, Global	Agnostik	Post-hoc
Sevri and Karacan[207]	2023	DNN	-	LIME,SHAP	Lokal, Global	Agnostik	Post-hoc
Novo et al.[208]	2023	DT, XGBoost, LR, KNN, RF, MLP	UNSW-NB15	SHAP	Lokal	Agnostik	Post-hoc
Gurbuz et al.[209]	2023	KNN, DT, RF, NB, SVM, MLP, ANN	Intensive Care Unit (ICU)	LIME, SHAP, ELI5, IG	Lokal, Global	Agnostik	Post-hoc
Lavieille and Atlas[210]	2023	Isolation Forest, GMM, SVM, AE, LOF	Microsoft Defender's logs	SHAP	Lokal, Global	Agnostik	Post-hoc
Nkoro et al.[211]	2023	LGBBoost	Edge-IIoTset	SHAP	Lokal	Agnostik	Post-hoc
Garcia et al.[212]	2023	DT, AdaBoost, LR, NB, RF, MLP	SQL injection dataset	FI, PI, SHAP	Lokal	Agnostik	Post-hoc
Kalutharage et al.[213]	2023	Autoencoder	USB-IDS	SHAP	Lokal, Global	Agnostik	Post-hoc
Javeed et al.[214]	2023	BiLSTM, Bi-GRU	CICIDDoS2019	SHAP	Global	Agnostik	Post-hoc
Bacevicius and Taraseviciene[224]	2023	LR, DT, RF, CNN, ANN	CIC-IDS2017 CSE-CIC-IDS2018	LIME, SHAP	Lokal, Global	Agnostik	Post-hoc
Patil et al.[215]	2023	DT, SVM, KNN, NN	CPS dataset	SHAP	Lokal	Agnostik	Post-hoc
Muna et al.[216]	2023	XGBoost	IoTID20	ELI5, SHAP, LIME	Lokal, Global	Agnostik	Post-hoc
Hassan et al.[217]	2023	RF	DoS dataset	LIME	Lokal	Agnostik	Post-hoc
Lanvin et al.[218]	2023	Autoencoder	CIC-IDS2017	Self explainable	Lokal, Global	Spesifik	İçsel

Çizelge 2.2. (devam) AYZ içeren STS çalışmaları

Pawlicki et al.[219]	2023	DNN	DDoS Backscatter	LIME, SHAP	Lokal, Global	Agnostik	Post-hoc
Zhou et al.[220]	2023	DT	CIC-AWS-2018	Self Explainable	Global	Agnostik	Post-hoc
Arreche et al.[221]	2023	RF, DNN, MLP, AdaBosst, KNN, LightGBM	RoEduNet-SIMARGL2021 CICIDS-2017 NSL-KDD	LIME, SHAP	Lokal, Global	Agnostik	Post-hoc
Sivamohan et al.[222]	2023	Bi-LSTM	CICIDS-2018 NSL-KDD	SHAP	Lokal, Global	Agnostik	Post-hoc
Houda et al.[223]	2023	Federated learning	UNSW-NB15	RuleFit, SHAP	Lokal, Global	Agnostik	Post-hoc
Lopes et al.[224]	2023	ANN	LYCOS-IDS2017	SHAP	Lokal	Agnostik	Post-hoc
Islam et al.[225]	2023	RF, DT, ANN, CNN	UNSW-NB15	LIME, SHAP, ELI5, ProtoDash	Lokal, Global	Agnostik	Post-hoc
Hasan et al.[226]	2023	Boosting algorithms	SCVIC-APT-2021	SHAP	Lokal	Agnostik	Post-hoc
Kostopoulos et al.[227]	2023	XGBoost, MLP	DGArchive	SHAP	Lokal, Global	Agnostik	Post-hoc
Thi et al.[228]	2023	GNN, CNN-GRU	NF-ToN-IoT, DARPA TCE3	SHAP	Lokal	Agnostik	Post-hoc
Wei et al.[229]	2023	MLP	CICDDoS2019	SHAP	Lokal, Global	Agnostik	Post-hoc
Irénée et al.[230]	2023	XGBoost	CIRA-CIC-DoHBrw-2020	SHAP	Lokal, Global	Agnostik	Post-hoc
Rumesh et al.[231]	2023	DNN	UNSW-NB15 NSL-KDD	SHAP	Global	Agnostik	Post-hoc
Gebrehans et al.[232]	2023	CNN	CSIRO Exfiltration Netlab DGA	SHAP	Lokal, Global	Agnostik	Post-hoc
Alani et al.[233]	2023	RF, LR, DT, GNB, XGBoost	IoT-ID, TON_IoT	SHAP	Lokal	Agnostik	Post-hoc
Bashaiwth et al.[234]	2023	LSTM	CICIDS2017 CSECICIDS2018 CICDDoS2019	LIME, SHAP, Anchor, LORE	Lokal	Agnostik	Post-hoc
Feng et al.[235]	2023	KNN	DARPA 2009 DDoS CAIDA 2007 DDoS FRGP NTP Flow Data DDoS Chargen 2016 FRGP Colorado Traffic	KNN	Lokal, Global	Spesifik	İçsel
Himmelhuber et al.[181]	2022	GraphSAGE	-	GNNExplainer	Lokal, Global	Spesifik	Post-hoc
Miao et al.[236]	2022	CNN	CIC-IDS2017	SHAP	Lokal	Agnostik	Post-hoc
Zhu et al.[237]	2022	GNN	Botnet graph dataset	Self Explainable	Global	Spesifik	İçsel
Mowla et al.[238]	2022	KNN, MLP, RF	SAD, ORNL	LIME	Lokal	Agnostik	Post-hoc
Alani and Miri[239]	2022	DT, GNB, RF, LR	TON_IoT, IoT-23, IoT-ID	SHAP	Lokal	Agnostik	Post-hoc
Szczepański et al.[240]	2022	ANN	IoT-23	Oracle-Explainer	Lokal	Agnostik	Post-hoc
El-Mihoub et al.[241]	2022	NB, LR, DT, SVM	ASNM-NPBO	EBM	Lokal, Global	Agnostik	Post-hoc
Huong et al.[242]	2022	Autoencoder	SCADA dataset	SHAP	Global	Agnostik	Post-hoc
AL-Essa et al.[243]	2022	NN	MalDroid20 CIC-IDS2017	DALEX	Global	Agnostik	Post-hoc
Chawla et al.[244]	2022	Autoencoder	Network cell trace dataset	SHAP	Lokal	Agnostik	Post-hoc
Carrera et al.[245]	2022	Autoencoder	KDD99 NSL-KDD CIC-IDS2017	SHAP	Lokal	Agnostik	Post-hoc

Çizelge 2.2. (devam) AYZ içeren STS çalışmaları

Sauka et al.[246]	2022	DT, RF,SVM, MLP	NSL-KDD	SHAP	Lokal, Global	Agnostik	Post-hoc
Eriksson and Grov[247]	2022	DNN	UNSW-NB15	LIME, SHAP	Lokal, Global	Agnostik	Post-hoc
Roy et al.[248]	2022	DNN	CIC-IDS2017	LIME, SHAP	Lokal, Global	Agnostik	Post-hoc
Roshan and Zafar[249]	2022	Autoencoder	CIC-IDS2017	KernelSHAP	Lokal, Global	Agnostik	Post-hoc
Tabassum et al.[250]	2022	DT, RF, ANN AdaBoost, XGBoost, MLP	ToN_IoT	LIME, SHAP, ELI5	Lokal, Global	Agnostik	Post-hoc
Ha et al.[251]	2022	LSTM based autoencoder	SCADA dataset	Gradient SHAP	Lokal, Global	Agnostik	Post-hoc
Aslam et al.[252]	2022	DT, NB, RF, XGB, AB, CB	DNS dataset	SHAP, LIME	Lokal, Global	Agnostik	Post-hoc
Zhou et al.[253]	2022	DT	CIC-AWS-2018	Self explainable	Lokal, Global	Spesifik	İçsel
Ramyavarshini et al.[254]	2022	-	NSL-KDD	LIME, SHAP	Lokal, Global	Agnostik	Post-hoc
Patil et al.[255]	2022	DT, RF, SVM	CIC-IDS2017	LIME	Lokal	Agnostik	Post-hoc
Younisse et al.[256]	2022	CNN	KDD99, Distilled Kitsune-2018	SHAP	Lokal	Agnostik	Post-hoc
Kalutharage et al.[257]	2022	Autoencoder	USB-IDS	SHAP	Lokal, Global	Agnostik	Post-hoc
Hutchison et al.[258]	2022	Autoencoder	NSL-KDD, CIC2017	Self explainable	Lokal, Global	Spesifik	İçsel
Yılmaz and Bardak[259]	2022	XGBoost, CatBoost, LightGBM	CIC-IDS2017, CSECIC-IDS2018, INSDN	SHAP	Global	Agnostik	Post-hoc
Barnard et al. [260]	2022	XGBoost AutoEncoder	NSL-KDD	SHAP	Lokal	Agnostik	Post-hoc
Andresini et al. [261]	2022	Neural Network and Attention	NSL-KDD UNSW-NB15	Self Explainable	Lokal	Spesifik	İçsel
Houda et al. [262]	2022	DNN	NSL-KDD, UNSW-NB15	RuleFit, LIME, SHAP	Lokal, Global	Agnostik	Post-hoc
Yiwen et al. [263]	2022	CNN+LSTM, XGBoost	KDD99	LIME, SHAP	Lokal	Agnostik	Post-hoc
Zebin et al.[264]	2022	Random Forest	CIRA-CIC-DoHBrw-2020	SHAP	Lokal	Agnostik	Post-hoc
Hsupeng et al.[265]	2022	XGBoost	CIC-IDS2017, CSTI-10	SHAP	Lokal	Agnostik	Post-hoc
Le et al.[316]	2022	Random Forest	NF-BoT-IoT-v2, NF-ToN-IoT-v2	SHAP	Lokal, Global	Agnostik	Post-hoc
Dias et al.[266]	2022	Decision Tree	-	Self Explainable	Global	Spesifik	İçsel
Oseni et al.[267]	2022	DNN	ToN_IoT	SHAP	Lokal, Global	Agnostik	Post-hoc
Lundberg et al.[268]	2022	DNN	Survival	SHAP	Lokal	Agnostik	Post-hoc
Houda et al.[269]	2022	DNN	UNSW-NB15	RuleFit, SHAP	Lokal, Global	Agnostik	Post-hoc
Ho et al.[270]	2022	DT, CNN	KDD99, CICIDS2017	SHAP	Lokal, Global	Agnostik	Post-hoc
Khan et al.[271]	2022	CNN, LSTM	GSP dataset	LIME	Global	Agnostik	Post-hoc
Ables et al.[272]	2022	SOM	NSL-KDD CICIDS2017	Self Explainable	Lokal, Global	Spesifik	Post-hoc
Batchu et al.[273]	2022	DES techniques	CICDDoS2019	SHAP, LIME	Lokal, Global	Agnostik	Post-hoc
Al-Essa et al.[274]	2022	DNN	MalDroid20 CICIDS2017	SHAP	Lokal, Global	Agnostik	Post-hoc
Liu et al.[275]	2021	DNN	UGR'16	CEM, BRCC, EDA	Lokal, Global	Agnostik	Post-hoc, İçsel
Mane and Rao[276]	2021	DNN	NSL-KDD	SHAP, LIME, BRCC, CEM, ProtoDash	Lokal, Global	Agnostik	Post-hoc, İçsel
Mahbooba et al.[277]	2021	Decision Tree	NSL-KDD	Self Explainable	Lokal	Spesifik	Intrinsic

Çizelge 2.2. (devam) AYZ içeren STS çalışmaları

Wali and Khan[278]	2021	Random Forest	CSE-CIC-IDS 2018	SHAP	Lokal	Agnostik	Post-hoc
Prasad et al.[279]	2021	ML	KDD99	TCAV	Global	Agnostik	Post-hoc
Sarhan et al.[280]	2021	DFF/RF	CSE-CIC-IDS2018, BoT-IoT, ve ToN-IoT	SHAP	Lokal	Agnostik	Post-hoc
Tcydenova et al.[281]	2021	SVM	NSL-KDD	LIME	Lokal	Agnostik	Post-hoc
Wang et al.[282]	2021	Multiclass/Binary Classifiers	CICIDS2017	SHAP	Lokal, Global	Agnostik	Post-hoc
Kim et al.[283]	2021	XGBoost	NSL-KDD	SHAP	Global	Agnostik	Post-hoc
Gu et al.[284]	2021	DNN	ADFA-LD, UNM	SHAP	Lokal, Global	Agnostik	Post-hoc
İslam et al.[285]	2021	RF, ET, SVM, GB, ANN, NB	CICIDS2017	Domain Knowledge Infusion	Lokal	Spesifik	İçsel
Andresini et al.[286]	2021	DNN	CICIDS2017	DALEX	Lokal, Global	Spesifik	Post-hoc
Zolanvari et al.[287]	2021	ANN	NSL-KDD, UNSW-NB15	TRUST	Lokal	Agnostik	Post-hoc
Roshan et al.[288]	2021	Autoencoder	CICIDS2017	SHAP	Lokal	Agnostik	Post-hoc
Rao and Mane[189]	2021	Binary classifier	NSL-KDD	SHAP, LIME	Lokal	Agnostik	Post-hoc
Das et al.[289]	2021	RF, DT, LR	NSL-KDD	SHAP, LIME	Lokal, Global	Agnostik	Post-hoc
Wierzbicki et al.[290]	2021	GBM, GLM, RF, NN	RegSOC-KES2021	SHAP	Lokal	Agnostik	Post-hoc
Sarhan et al.[291]	2021	DFF/RF	CSE-CIC-IDS2018, BoTIoT, and ToN-IoT	SHAP	Lokal	Agnostik	Post-hoc
Dang[292]	2021	XGBoost	-	PDP, SHAP	Lokal, Global	Agnostik	Post-hoc
Szczepanski et al.[293]	2021	ANN	IoT-23	LIME Oracle-Explainer	Lokal	Agnostik	Post-hoc
Al-Imran et al.[294]	2021	LSTM	Kyoto 2013	LIME	Lokal	Agnostik	Post-hoc
Caforio et al.[295]	2021	CNN	KDD99, NSL-KDD, UNSW-NB15	GradCAM	Lokal	Spesifik	Post-hoc
Wickramasinghe et al.[296]	2021	SOM	KDD99, DoHBrw-2020	Self Explainable	Lokal, Global	Spesifik	Post-hoc
Alenezi and Ludwig[297]	2021	XGBoost, RF	ISCX-URL2016, CICMalDroid 2020	SHAP	Lokal, Global	Agnostik	Post-hoc
AL-Essa[298]	2021	XGBoost	NSL-KDD CICIDS2017	DALEX	Lokal, Global	Agnostik	Post-hoc
Botana et al.[299]	2020	DT	KDD99 ISCXIDS 2012	Self Explainable	Lokal	Spesifik	Ante-hoc
Rathod et al.[300]	2020	Autoencoder	Gas Pipeline system dataset	LIME, LRP	Lokal, Global	Agnostik	Post-hoc
Sejr et al.[301]	2020	KNN	ECML PKDD 2007 dataset	KNN	Lokal, Global	Spesifik	İçsel
Karn et al.[302]	2020	DT, EML, ANN, LSTM	Syscalls	SHAP, LIME	Lokal	Agnostik	Post-hoc
Wang et al.[187]	2020	Multiclass/Binary Classifiers	NSL-KDD	SHAP	Lokal, Global	Agnostik	Post-hoc
Reyes et al.[303]	2020	RF/NB/XGB	AWID	SHAP	Lokal, Global	Agnostik	Post-hoc
Szczepanski et al.[304]	2020	ANN/DT	CICIDS2017	Clustering	Lokal	Agnostik	Post-hoc
İslam et al.[305]	2020	RF, ET, SVM, GB, ANN, NB	CICIDS2017	Domain Knowledge Infusion	Lokal	Spesifik	İçsel
Antwarg et al.[306]	2020	Autoencoder	KDD99	SHAP	Lokal	Agnostik	Post-hoc

Çizelge 2.2. (devam) AYZ içeren STS çalışmaları

Ahn et al.[307]	2020	CNN	ISCX VPN-nonVPN	Genetic algorithm	Lokal	Spesifik	İçsel
Wu et al. [308]	2020	CNN	NSL-KDD	LIME	Lokal	Agnostik	Post-hoc
Sarica and Angin[190]	2020	RF	SDN dataset	Self Explainable	Lokal, Global	Spesifik	İçsel
Dang[309]	2020	XGBoost	CICIDS2017	PDP, SHAP	Lokal, Global	Agnostik	Post-hoc
Guerra-Manzanares et al.[310]	2019	DT, RF, KNN	IoT Botnet dataset	LIME	Lokal	Agnostik	Post-hoc
Morichetta et al.[311]	2019	SVM	Youtube QoE	LIME	Lokal	Agnostik	Post-hoc
Li et al.[312]	2019	RNN	NSL-KDD	Regression+Fused Lasso	Lokal	Spesifik	İçsel
Amarasinghe et al.[315]	2018	DNN	NSL-KDD	DNN Explanation Generator	Global	Spesifik	Post-hoc
Amarasinghe and Manic[313]	2018	MLP	NSL-KDD	Key Feature Extraction	Lokal	Agnostik	Post-hoc
Marino et al.[314]	2018	MLP	NSL-KDD	Adversarial	Lokal	Agnostik	Post-hoc

Çizelge 2.3. AYZ içeren STS çalışmalarının değerlendirilmesi

Perspektif	Türler	Çalışmalar	Sayı	Özet
AYZ Modelleri	LIME	[35],[36],[38],[39],[40],[43],[44],[45],[46],[48],[49],[50],[52],[55],[56],[59],[60],[61],[63],[66],[68],[70],[72],[73],[75],[78],[79],[80],[81],[83],[84],[85],[89],[90],[92],[93],[94],[98],[99],[101],[102],[103],[105],[108],[109],[110],[111],[112],[113],[114],[115],[117],[119],[120],[122],[127],[129],[136],[139],[146],[147],[149],[154],[156],[158],[159],[163],[164],[165],[166],[168],[169],[171],[172],[189],[191],[193],[195],[201],[202],[203],[206],[207],[209],[216],[217],[219],[221],[224],[225],[234],[238],[247],[248],[250],[252],[254],[255],[262],[263],[271],[273],[276],[281],[289],[293],[294],[300],[302],[308],[310],[311],[171],[173],[152],[180],[181],[183],[185],[189],[198],[202],[211],[212],[214],[216],[218],[219],[226],[227],[235],[237],[240],[245],[153],[253],[257],[258],[264],[266],[272],[274],[275]	112	SHAP, model kararları için hem genel hem de yerel açıklamalar sunabilmesi ve her bir özelliğe önem puanları atayabilmesi nedeniyle STS çalışmalarında yaygın olarak kullanılan bir AYZ yöntemidir. LIME'in bireysel tahminleri açıklama yeteneği, belirli örneklerin neden saldırı olarak sınıflandırıldığını yorumlaması gereken güvenlik uzmanları için özellikle faydalıdır. Bu gelişmiş yetenek, onu literatürde SHAP'tan sonra en çok tercih edilen yöntem haline getirmiştir. Karar ağaçları, doğası gereği yorumlanabilir oldukları için SHAP veya LIME gibi sonradan açıklanabilirlik yöntemlerine ihtiyaç duymaz.

Çizelge 2.3. (devam) AYZ içeren STS çalışmalarının değerlendirilmesi

SHAP	[31],[32],[33],[34],[35],[37],[38],[39],[40], [41],[42],[43],[44],[45],[46],[47],[48],[49], [50],[51],[52],[53],[54],[55],[56],[57],[58], [59],[60],[61],[62],[63],[64],[65],[66],[67], [68],[69],[70],[71],[72],[73],[74],[75],[76], [77],[78],[79],[81],[82],[83],[84],[85],[86], [87],[90],[92],[94],[95],[97],[98],[99],[100],[101],[102],[104],[105],[106],[107],[108], [109],[110],[111],[112],[113],[114],[117],[118],[119],[120],[121],[123],[124],[125],[126],[127],[128],[130],[131],[132],[134],[135],[136],[137],[138],[139],[140],[141],[142],[143],[144],[145],[146],[147],[148],[149],[150],[152],[153],[154],[155],[156],[157],[158],[159],[160],[163],[164],[165],[166],[167],[168],[169],[170],[171],[172],[173],[174],[175],[176],[187],[189],[191],[192],[194],[195],[196],[197],[198],[199],[200],[204],[205],[206],[207],[208],[209],[210],[211],[212],[213],[214],[215],[216],[219],[221],[222],[223],[224],[225],[226],[227],[228],[229],[230],[231],[232],[233],[234],[236],[239],[242],[244],[245],[246],[247],[248],[249],[250],[251],[252],[254],[256],[257],[259],[260],[262],[263],[264],[265],[267],[268],[269],[270],[273],[274],[276],[278],[280],[282],[283],[284],[288],[289],[290],[291],[292],[297],[302],[303],[306],[309],[316],[163],[164],[168],[169],[170],[171],[172],[173],[174],[175],[176],[177],[178],[152],[179],[180],[183],[185],[186],[187],[188],[189],[190],[191],[193],[192],[194],[195],[196],[197],[198],[200],[203],[206],[208],[209],[210],[211],[212],[214],[216],[218],[220],[221],[223],[224],[226],[227],[228],[229],[146],[231],[232],[233],[234],[237],[238],[240],[242],[244],[246],[247],[248],[252],[153],[253],[254],[255],[256],[261],[266],[151],[267],[270],[273]	213
GNExplainer	[177], [178], [180], [181], [182], [183], [184], [186]	8
ICE	[150], [196]	2
PDP	[150],[196],[292],[309]	4
PFI	[87],[148],[164],[196]	4
IG	[59],[91],[209]	3
FI, PI	[212]	1
RuleFit	[226],[262], [269]	3
Anchor, LORE	[234]	1
LRP	[91],[300]	2
KNN	[235],[301]	2
Oracle-Explainer	[240],[293]	2
KernelSHAP	[249]	1
Gradient SHAP	[251]	1
CEM, BRCCG	[164],[275],[276]	3
TCAV	[279]	1
TRUST	[287]	1
GradCAM	[295]	1
Domain Knowledge Infusion	[285], [305]	2
Clustering	[304]	1
Genetic algorithm	[307]	1

Çizelge 2.3. (devam) AYZ içeren STS çalışmalarının değerlendirilmesi

	Regression+Fused Lasso	[312]	1	
	DNN Explanation Generator	[315]	1	
	Key Feature Extraction	[313]	1	
	Adversarial	[314]	1	
	Self Explainable	[96],[116],[151],[185],[190],[218],[220],[237],[253],[258],[261],[266],[272],[277],[296],[299]	16	
Modele Özgü		[96],[116],[151],[177],[178],[179],[180],[181],[182],[183],[184],[185],[186],[190],[218],[235],[237],[253],[258],[261],[266],[272],[277],[285],[286],[295],[296],[299],[301],[305],[307],[312],[315]	33	Literatürdeki çalışmaların çoğu, SHAP ve LIME gibi modelden bağımsız yaklaşımlar kullanmıştır. Ancak, DT ve RF gibi içsel olarak açıklanabilir modellere ek olarak, CNN, DNN ve otokodlayıcılar gibi DL modelleri için özel olarak geliştirilmiş belirli açıklama yöntemleri kullanan birçok çalışma da mevcuttur.
İçsel Açıklama		[96],[116],[151],[185],[190],[218],[235],[237],[253],[258],[261],[266],[275],[276],[277],[285],[301],[305],[307],[312]	20	Literatürdeki çalışmaların büyük çoğunluğunda sonuçların açıklanabilir olması için daha önce geliştirilen AYZ yöntemleri kullanılmış olup, bu nedenle çalışmaların çoğunda yapılan açıklamalar post-hoc niteliktedir.
Veri Kümesi	NSL-KDD	[39],[41],[46],[54],[62],[78],[79],[80],[82],[86],[87],[89],[90],[91],[96],[97],[111],[124],[126],[146],[148],[149],[157],[159],[168],[182],[187],[189],[192],[195],[196],[197],[201],[202],[203],[204],[205],[206],[221],[222],[231],[245],[246],[254],[258],[260],[261],[262],[272],[276],[277],[281],[283],[287],[289],[295],[298],[308],[312],[313],[314],[315]	62	NSL-KDD veri seti, daha eski bir veri seti olması ve yapısının birçok yapay zeka modeli için uygun olması nedeniyle daha sık tercih edilmektedir. NSL-KDD veri setini, son yıllarda araştırmacılar arasında giderek daha popüler hale gelen UNSW-NB15 ve CIC-IDS2017 veri setleri takip etmektedir. En eski veri seti olmasına rağmen KDD99 veri setinin artık birçok çalışmada tercih edilmemesi dikkat çekicidir. Bunun nedeni, güncelliğini yitirmiş olması ve birçok modern saldırı türü ve yöntemini içermemesidir.
	UNSW-NB15	[31],[36],[38],[46],[49],[55],[61],[62],[64],[67],[82],[83],[86],[104],[105],[106],[112],[115],[130],[141],[147],[149],[151],[157],[161],[168],[170],[179],[192],[194],[196],[205],[206],[208],[223],[225],[231],[247],[261],[262],[269],[287],[295]	43	
	CIC-IDS2017	[31],[32],[38],[41],[43],[51],[58],[67],[77],[79],[80],[85],[87],[89],[91],[93],[95],[111],[112],[136],[146],[149],[156],[159],[160],[162],[165],[167],[170],[175],[178],[183],[193],[194],[201],[218],[221],[224],[234],[236],[243],[245],[248],[249],[255],[259],[265],[270],[272],[274],[282],[285],[286],[288],[298],[304],[305],[309]	58	
	ADFA-LD	[99],[143],[191],[284]	4	
	Honeypot	[124],[195]	2	
	TON_IoT	[48],[52],[62],[66],[76],[78],[87],[108],[127],[130],[196],[233],[239],[250],[267]	15	
	CIC-IoT-Dataset-2022, IEC 69870-5-104	[198]	1	
	CSE-CIC-IDS2018	[158],[161],[179],[224],[280],[291]	6	
	CIC-AWS-2018	[220],[253]	2	
	CPS Dataset	[200],[215]	2	
	Intensive Care Unit (ICU)	[209]	1	
	Microsoft Defender's logs	[210]	1	
	Edge-IIoTset	[53],[96],[211]	3	
	SQL injection dataset	[212]	1	
	USB-IDS	[213],[257]	2	
	CICDDoS2019	[229],[234],[273]	3	
	IoTID20	[42],[57],[72],[94],[216]	5	
	DoS dataset	[217]	1	

Çizelge 2.3. (devam) AYZ içeren STS çalışmalarının değerlendirilmesi

DDoS Backscatter	[219]	1
RoEduNet-SIMARGL2021	[91],[159],[221]	3
LYCOS-IDS2017	[224]	1
SCVIC-APT-2021	[47],[74],[226]	3
DGArchive	[227]	1
NF-ToN-IoT, DARPA TCE3	[64],[112],[153],[180],[228],[316]	6
CIRA-CIC-DoHBrw-2020	[230],[264]	2
CSIRO Exfiltration Netlab DGA	[232]	1
IoT-ID	[233],[239]	2
IoT-23	[116],[122],[239],[240],[293]	5
DARPA 2009 DDoS CAIDA 2007 DDoS FRGP NTP Flow Data FRGP Colorado Traffic DDoS Chargin 2016	[235]	1
Botnet graph dataset	[237]	1
SAD, ORNL	[238]	1
ASNM-NPBO	[241]	1
SCADA dataset	[251]	1
MalDroid20	[243],[274]	2
Network cell trace dataset	[244]	1
DNS dataset	[252]	1
KDD99	[86],[92],[110],[130],[173],[182],[245],[256],[263],[270],[279],[295],[296],[299],[306],[263],[270],[209]	18
Distilled Kitsune-2018	[256]	1
CSTI-10	[265]	1
NF-BoT-IoT-v2,	[94],[316]	2
NF-ToN-IoT-v2	[153],[316]	2
Survival	[268]	1
GSP dataset	[271]	1
UGR'16	[275]	1
RegSOC-KES2021	[290]	1
Kyoto 2013	[294]	1
ISCX-URL2016	[297]	1
ISCXIDS 2012	[299]	1
ISCX VPN-nonVPN	[307]	1
Gas Pipeline system dataset	[300]	1
ECML PKDD 2007 dataset	[301]	1
AWID	[303]	1
SDN dataset	[190]	1
IoT Botnet dataset	[310]	1

Çizelge 2.3. (devam) AYZ içeren STS çalışmalarının değerlendirilmesi

YZ Modell eri	CNN	[39],[41],[45],[48],[52],[60],[62],[69],[70] , [71],[76],[85],[105],[106],[115],[121],[122],[130],[131],[133],[142],[143],[149],[152], [154],[199],[200],[203],[206],[224],[225],[228],[232],[236],[256],[263],[270],[271],[295],[307],[308]	41	Karar ağacı ve rastgele orman algoritmaları, AYZ destekli STS çalışmalarında yaygın olarak tercih edilmiştir. Karar ağaçları ve rastgele ormanlar, makine öğrenmesinde kullanılabilen en yorumlanabilir YZ modelleri arasındadır. Bu nedenle, bu yöntemler AYZ çalışmalarında sıklıkla kullanılmıştır. Ayrıca, bu algoritmalar SVM, MLP, NB, KNN ve LR gibi geleneksel MÖ algoritmalarıyla birlikte STS çalışmalarında tahmin performansını değerlendirmek için yaygın olarak kullanılmaktadır. Geleneksel MÖ algoritmalarının ardından, özellikle son yıllarda AYZ destekli STS sistemlerinde ANN, DNN, CNN, LSTM, GBM ve Autoencoder gibi derin öğrenme tabanlı teknikler giderek daha fazla tercih edilmektedir. Bu teknikler, geleneksel MÖ tekniklerine kıyasla önemli ölçüde daha iyi tahmin performansı sundukları için birçok STS sisteminde tercih edilmektedir.
	LSTM	[32],[45],[56],[59],[60],[62],[66],[68],[69] , [70],[73],[75],[76],[78],[83],[87],[94],[97] , [98],[105],[106],[108],[115],[120],[121],[122],[135],[139],[141],[142],[143],[148],[149],[150],[195],[196],[204],[214],[222],[234],[251],[263],[271],[294],[302]	45	
	DNN	[42],[43],[48],[57],[64],[72],[78],[82],[84] , [91],[111],[119],[127],[159],[160],[163],[164],[166],[168],[171],[202],[205],[206],[207],[219],[221],[231],[247],[248],[262],[267],[268],[269],[274],[275],[276],[284],[286],[315]	39	
	ANN	[33],[54],[122],[142],[146],[209],[224],[225],[240],[250],[285],[287],[293],[302],[304],[305]	16	
	XGBoost	[35],[40],[47],[49],[50],[55],[58],[61],[63] , [65],[72],[74],[90],[103],[104],[108],[112] , [113],[117],[127],[128],[129],[132],[134],[136],[138],[140],[145],[153],[154],[157],[158],[163],[165],[167],[169],[192],[193],[208],[216],[227],[230],[233],[250],[259],[260],[263],[265],[283],[292],[297],[298],[309]	53	
	RF	[44],[49],[51],[53],[55],[56],[58],[63],[71] , [72],[81],[83],[85],[87],[90],[92],[93],[98] , [99],[101],[102],[103],[105],[110],[111],[112],[113],[114],[116],[125],[126],[127],[128],[129],[134],[136],[138],[143],[144],[145],[153],[154],[158],[159],[165],[166],[167],[175],[176],[190],[197],[200],[208],[209],[212],[217],[221],[224],[225],[233],[238],[239],[246],[250],[252],[255],[280],[285],[289],[290],[291],[297],[303],[305],[310]	75	
	DT	[38],[44],[49],[50],[55],[56],[63],[72],[92] , [93],[99],[101],[108],[113],[122],[129],[134],[144],[146],[164],[165],[166],[167],[172],[173],[176],[208],[209],[212],[215],[220],[224],[225],[233],[239],[241],[246],[250],[252],[253],[255],[270],[289],[299],[302],[304],[310]	47	

Çizelge 2.3. (devam) AYZ içeren STS çalışmalarının değerlendirilmesi

MLP	[45],[49],[56],[58],[72],[83],[98],[99],[114],[159],[164],[191],[200],[208],[209],[212],[221],[227],[229],[238],[246],[250],[313],[314]	24
KNN	[44],[63],[83],[87],[92],[93],[99],[102],[108],[111],[114],[136],[153],[154],[159],[172],[208],[209],[215],[221],[235],[238],[301],[310]	24
SVM	[54],[63],[71],[85],[92],[111],[114],[129],[145],[146],[153],[159],[164],[165],[172],[200],[209],[210],[215],[241],[246],[255],[281],[285],[305],[311]	26
LR	[38],[46],[49],[63],[82],[83],[87],[92],[114],[125],[129],[153],[154],[172],[208],[212],[224],[233],[239],[241],[289]	21
BiLSTM	[60],[62],[69],[76],[83],[87],[120],[121],[135],[139],[141],[148],[195],[214]	14
Bi-GRU	[77],[79],[214]	3
NB	[31],[36],[38],[46],[49],[55],[61],[62],[64],[67],[82],[83],[86],[104],[105],[106],[112],[115],[130],[141],[147],[149],[151],[157],[161],[168],[170],[179],[192],[194],[196],[205],[206],[208],[223],[225],[231],[247],[261],[262],[269],[287],[295]	43
SAE	[199]	1
Autoencoder	[66],[70],[71],[89],[137],[151],[162],[213],[218],[242],[244],[245],[249],[251],[257],[258],[260],[288],[300],[306]	20
LGBost	[211]	1
AdaBoost	[87],[108],[111],[113],[158],[159],[164],[173],[212],[250]	10
CatBoost	[49],[112],[125],[128],[132],[161],[173],[179],[259]	9
LightGBM	[31],[67],[82],[108],[111],[128],[138],[158],[159],[221],[259]	11
IENN-based classification	[201]	1
Isolation Forest	[109],[210]	2
Federated learning	[223]	1
GNN	[95],[106],[147],[177],[178],[180],[181],[182],[183],[184],[185],[186],[228],[237]	14
LSTM based autoencoder	[251]	1
Neural Network and Attention	[261]	1
CNN-LSTM	[115],[142],[143],[263]	4
CNN-GRU	[131],[142],[228]	3
DES techniques	[273]	1
RNN	[273]	1
SOM	[272],[296]	2



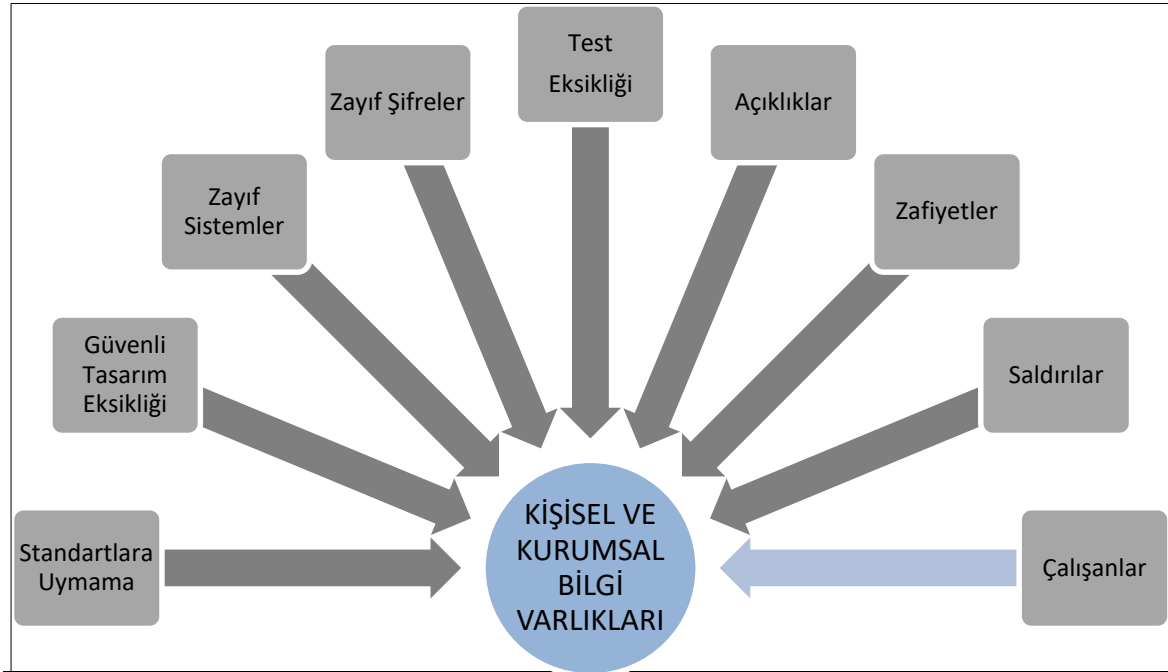
3. SALDIRI TESPİT SİSTEMLERİ

Güncel bilgiye sahip olan ve bu bilgileri iyi işleyen, analiz eden, değerlendiren ve organize eden toplumlar tarihin başarılı medeniyetlerini kurmuşlardır. Gerek sosyal ve ekonomik gerekse teknolojik alanlar olsun hayatımızın her kademesinde BT teknolojilerinden faydalanan ve verilerden değer elde edenler, sektörlerle, teknolojilere hatta toplumlar üzerinde hâkimiyet kurmakta ve kendilerini kabul ettirmektedirler. Bilginin öneminden dolayı devletler tarafından ulusal bilgi politikaları ve stratejileri geliştirilmekte, altyapı yatırımları yapılmakta, sektörler desteklenmekte, yüksek yetkinliğe sahip uzmanlar yetiştirilmektedir. Bu strateji ve politikalara uygun olarak; veriye erişen, bunu bilgiye, değere ve faydaya dönüştüren ve kullanan kişilerden, bilgiye yön veren uzmanlardan, bilgi teknolojilerinden, bilginin maliyetinden bu bilgi politikalarını oluşturacak kuruma kadar birçok farklı alanda düzenlemeler yapılmaktadır [317]. John Naisbitt bir ifadesinde “*yeni güç kaynağı; azınlığın elindeki para değil, çoğunluğun elindeki bilgidir*” diyerek bilginin önemini ortaya koymaktadır [318]. Bilgi çağını yaşayan günümüz toplumları ve bireyleri, veriyi ve bilgiyi önemli bir anahtar işlevi olarak görmekte, kalkınmışlığın göstergesi olarak nitelendirilen bir ölçüt olarak kabul etmektedirler.

Bilgi kavramının önemi anlaşıldıkça, bilginin bir varlık olarak korunması ve güvenliğinin sağlanmasına verilen önemde artmaktadır. Günümüz bilgi çağında gerek kişisel gerekse kurumsal bilgi varlıkları dijital ortamlarda depolanmakta, işlenmekte, tutulmakta ve korunmaktadır. Bu ortamlarda saklanan verilerin korunması, fiziksel olarak korunmanın yanı sıra dijital olarak ta birtakım önlemlerin alınması açısından birçok sorumluluğu da beraberinde getirmekte, bilginin korunmasına yönelik yöntem, yaklaşım ve eylemler kısaca bilgi güvenliği olarak adlandırılmaktadır [319]. Diğer bir ifadeyle bilgi güvenliği, bilgi varlıklarının başına bir iş gelmeden (izinsiz erişim veya kullanım, ifşa, yok edilme, değiştirilme, silinme veya kopyalanma gibi) önce gelmesi muhtemel her türlü riski önceden ön görme ve önlem alma girişimidir. Temel olarak; gizlilik, bütünlük ve erişilebilirliğin sağlanması için birçok güvenlik teknolojileri ve çözümleri geliştirilmiş olup kullanılmaktadır. Değişen ve gelişen saldırı yüzeyi ve boyutuna göre de yeni çözümler ve teknolojiler geliştirilmektedir.

İnsan hayatını kolaylaştıran birçok bilgi ve bilgisayar sistemleri internetin de sağladığı

küresel büyüme ölçeği içerisinde giderek büyük boyutlara ulaşmaktadır. Bilgisayar ağ sistemlerinin yaygın olarak kullanılmaya başlanmasıyla birlikte ağ üzerinden yapılan saldırılarda da artışlar yaşanmaya başlamıştır. Bu sistemlerde oluşabilecek çeşitli güvenlik açıklıkları ile bilgisayar güvenliğinin tehdit edilerek kişisel, kurumsal ve değerli bilgilerin yetkisiz kişilerin eline geçmesi ve kötüye kullanılması tehlikesi ortaya çıkmaktadır. İnternetin ve bu bilgi sistemlerinin her geçen gün yaygınlaşması ve bu sistemlerin giderek daha fazla kullanılması bilgisayar ağlarının güvenliğini hayati öneme taşımıştır. Saldırı olaylarının bir kısmına bilinçsiz kullanıcılar neden olurken bir kısmına da bilerek sisteme zarar vermek ve bilgi varlıklarını ele geçirmek isteyen kötü niyetli kişiler tarafından yapılmaktadır. İnternetin yaygınlaşması ile bilgisayar haberleşmelerindeki saldırıların sayısı ve çeşidi de artmıştır [320, 321]. Bu saldırılar karşısında kimlik doğrulama, yetkilendirme, anti-viral yazılımlar ve çözümler geliştirilmeye çalışılmıştır. İlk çıkan saldırılar daha çok basit kod yürütme, parola tahminleri gibi etkisi ve olasılığı düşük saldırılar olmasına karşın süreç içerisinde saldırıların karmaşıklığı ve etkileri artmıştır. Buna rağmen bu saldırıları kullanabilmek için gereken bilgi düzeyi düşmüştür. Şekil 3.1’den de görülebileceği gibi saldırı boyutu ve yüzeyi zaman içerisinde kullanılan teknoloji ve uygulamalar ile gelişmekte ve değişmektedir.



Şekil 3.1. Saldırıların hedefi ve boyutu

BT'lerin gelişmesi, kullanımının yaygınlaşması ve devasa veri denizlerine erişilmesi ile bilginin işlenmesi, saklanması, taşınması ve korunması daha da önem kazanmıştır. Casus ve zararlı yazılımlar, mobil tehditler, saldırganların ve saldırıların sayısındaki artışlardan dolayı bilgi varlıklarına verilen zararın boyutunun da hızla yükselmesi, bilgi güvenliğinin önemini arttırmış, kurumların bilgi güvenliğini sağlayabilmeleri konusunda geliştirilen pek çok yaklaşım kullanılmaya başlanmıştır. Teknolojik önlemlerin geliştirilmesi sırasında bir varlık olarak bilgiyi, tehdit ve saldırılara karşı korumak için güvenlik duvarları, anti-viral yazılım çözümleri, saldırı tespit sistemleri (STS), saldırı engelleme sistemleri (SES) gibi araçlar geliştirilmiştir. Bu araçların en güncel çözümler kullanılarak geliştirilmesi, belirlenen kural ve politikalara göre yapılandırılması, yapılan saldırıların büyük ölçüde önlenmesine ve bilgi güvenliğimizin sağlanmasında bizlere yardımcı olmaktadır [321]. STS'ler, sistemlere yapılan saldırılara karşı önlem almamızı kolaylaştıran, saldırıları tespit edip bunu uyarı veya alarm üreterek bildiren veya raporlayan yazılım ve donanımlardır. STS'lerin kullanılması ile sistemlere yapılan yetkisiz erişimler ve kötüye kullanımlar tespit edilerek, bunların yol açabileceği zararlar mümkün olduğunca otomatik olarak engellenmiş olur. Bilgisayar sistemlerinde, STS'lerin kullanılması ile birlikte, sisteme ne tür saldırıların daha çok yapıldığı, sistemdeki mevcut açıklar ve saldırganlar hakkında daha detaylı bilgiler elde edilebilir [321]. Elektronik ortamdaki tehdit ve tehlikelerin sistemlere çoğunlukla dışarıdan gelmesinden dolayı, bu ortamlara giren tehditleri ve saldırıları önceden tespit etmek çok önemli olduğundan, STS'ler bu amaçla kullanılmaktadır [322].

STS teknolojileri, saldırganlar ile saldırılara karşı çeşitli savunma mekanizmaları geliştirmeye çalışan siber savunma ekipleri arasındaki devam eden mücadelenin sonucu olarak Şekil 3.2'de gösterildiği gibi zaman içerisinde önemli gelişmeler kaydetmiştir. James Anderson tarafından 1980 yılında otomatik olarak bilgi sistemlerinin izlenmesi fikri ortaya atılmış ve güvenlik ihlallerini tespit etmek için denetim izlerinin kullanılabileceği ana hatlarıyla açıklanmıştır [323]. Bu temel çalışma, bu alandaki ilk çalışmalardan biri olmakla birlikte imza tabanlı sistemler olan ilk nesil STS'lerin geliştirilmesine yol açmıştır. 1980'lerin sonu ve 1990'ların başında ortaya çıkan birinci nesil STS'ler, büyük ölçüde imza tabanlı tespit yöntemlerine dayanmaktaydı. Denning tarafından 1987'de geliştirilen Ağ Güvenlik İzleyicisi (NSM) [324] birinci nesil STS'lerin en eski örneklerinden biridir ve modern STS'lerin temelini oluşturmaktadır. 1990'larda anomali tabanlı STS'ler geliştirilmeye başlanmıştır. Bu sistemler, bilinen imzalara güvenmek yerine bir bilgi sisteminin normal davranışından sapmasını tespit etmeyi amaçlamaktadır. 2000'li yıllarda,

siber tehditlerin artan karmaşıklığı ve daha uyarlanabilir ve akıllı tespit mekanizmalarına duyulan ihtiyaç nedeniyle STS'lerin makine öğrenmesi (ML) ve YZ (YZ) teknolojileri ile entegrasyonu sağlanarak zeki STS'ler oluşturulmaya başlanmıştır. YZ tabanlı STS'ler, verilerden öğrenme, kalıpları belirleme ve geleneksel yöntemlerin kaçırabileceği karmaşık korelasyonlara dayalı kararlar alma gibi önemli yeteneklere sahiptir [325]. Son on yılda ise YZ tabanlı STS'ler gelişmiş tespit yetenekleri sunarken, bu modeller tarafından alınan kararların açıklanabilirliği ile ilgili olarak da yeni zorlukları beraberinde getirmiştir. STS'lerde kullanılan YZ modellerinin giderek karmaşık hale gelmesi ile birlikte ortaya çıkan açıklanabilirlik ve anlaşılabilirlik ihtiyaçlarından dolayı açıklanabilir STS'ler geliştirilmeye başlanmıştır. Açıklanabilir STS'ler ile yüksek doğruluk performansının yanı sıra kararları hakkında net açıklamalar sunan yeni nesil STS'ler geliştirilmektedir. Son birkaç yıldır ise üretken yapay zekâ alanındaki gelişmelere paralel olarak üretken YZ içeren STS'ler geliştirilmeye başlanmıştır.



Şekil 3.2. Saldırı tespit sistemleri zaman çizelgesi

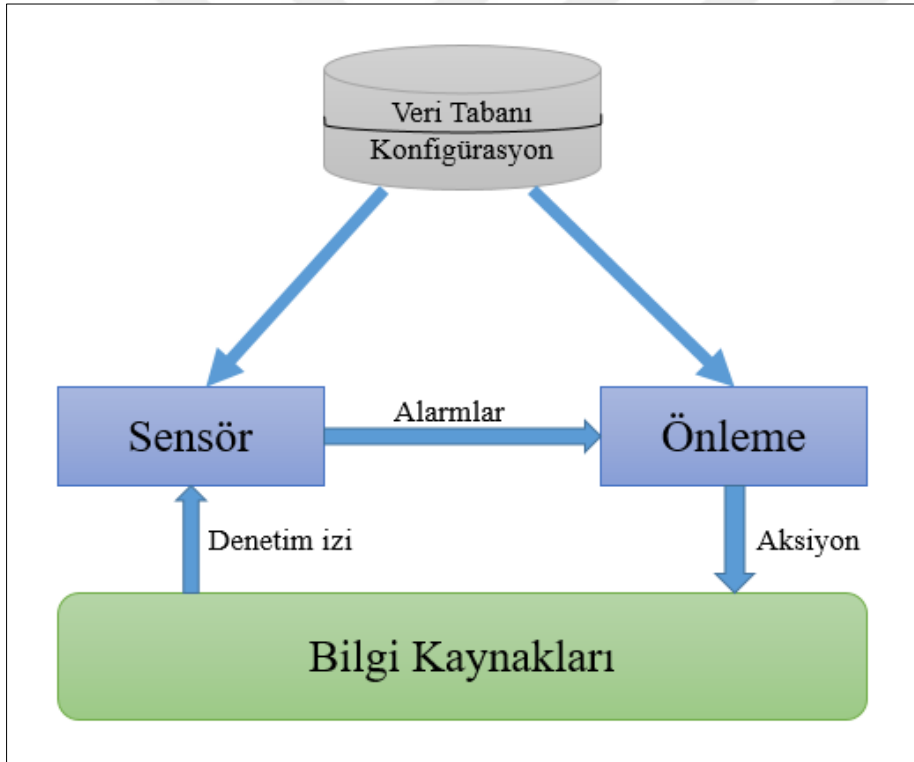
Bilgiyi korurken, sistemlerin koruma sürekliliğinin sağlanması da hayati önem taşımaktadır. Sürekliliği korumak için, yapılan saldırılara karşı alınan önlemlerin güncelliğini koruması gerekmektedir. Bu güncellik de ancak değişen saldırı ve yöntemlerin bilinmesi ve var olan sisteme adapte edilmesi ile sağlanabilir. Yeni bir saldırının veya yöntemin tespitine geçmeden önce “saldırının” ne anlama geldiğini bilmek gerekir. Bu nedenle güvenlik konusunda yapılan çalışmalarda saldırının birçok tanımı yapılmıştır. Anderson’a göre “*bir saldırı, izin almadan bilgiye ulaşım, değiştirme, sistemi kullanılmaz veya güvenilmez hale getirmektir*” [323].

Saldırılar, sezgisel olarak bir bilgi sistemi içerisinde, sistemin güvenlik politikalarını ihlal eden etkinliklerdir. Saldırı tespiti ise saldırıların türünü, şiddetini, ihlal ettiği zafiyet vb. durumları tespit etmeye, tanımlamaya yönelik olarak geliştirilen işlemlerdir [323]. Saldırgan, bilişim sistemlerindeki zafiyetlerden yararlanarak sistem üzerindeki emellerini gerçekleştirmek üzere saldırıyı planlayan ve gerçekleştiren kişidir. Saldırı tespit sistemleri ise, bilgisayar sistemlerine yapılan saldırıları gerçekleştirdiği anda veya gerçekleştikten sonra tespit etmek, korunacak sistemlerin dışındaki ortamlardan gelebilecek çeşitli paket ve verilerden oluşan bu saldırıları fark etmek üzere tasarlanmış sistemlerdir. Temel bir STS yapısı Şekil 3.3’de gösterilmektedir. Anderson tarafından [323], konunun temelini oluşturan ve 1980 yılında bir rapor olarak sunulan bu tanımından günümüze kadar literatürde birçok tanım yapılmış ve konuyla ilgili olarak birçok yazılım gerçekleştirilmiştir. Günümüzde ise saldırı, “bilginin mahremiyetini, bütünlüğünü ve erişilebilirliğini tehlikeye atabilecek girişimlerin kümesi” olarak tanımlanmaktadır [321].

Saldırlara verilecek yanıtlar ise Şekil 3.4’de özetlenen saldırılar dikkate alınmadan değerlendirilmemelidir. Saldırıların incelenmesi ve ilgili hedeflerin belirlenmesi saldırılara karşı etkin bir savunma yönteminin belirlenmesinde önemlidir. Bu nedenle saldırıların sınıflandırılması ve saldırı-savunma uyumunun sağlanması gerekmektedir. Örneğin, sunucu tabanlı saldırılar çoğunlukla sistem kullanılabilirliğine, işletim sistemi performansına ve web hizmeti işlemlerine yönelik saldırılardır. Ağ tabanlı saldırılar ise, ağ kullanılabilirliğini ve performansını etkilemeyi amaçlayan ağlara yönelik saldırıları içermektedir. Bununla birlikte, pasif saldırılarda saldırganlar sistemdeki herhangi bir bilgiyi yalnızca gizlice dinler ve değiştirmezken, aktif saldırılarda saldırganlar tüm ağı yok etmek amacıyla yetkisiz erişim elde etmeye ve sistemdeki bilgileri değiştirmeye çalışmaktadır. Sistemler, bu tür saldırılardan hızla kurtulabilmelidir. Bu amaçla saldırıların çoğu Şekil 3.4’de sınıflandırılmıştır.

İnternetin ve iletişim olanaklarının artmasıyla birlikte saldırganlar tarafından saldırılabilecek daha çok sistem ortaya çıkmıştır. Bu saldırıların büyük bir bölümü kullanılan sistemin kusurları veya eksiklerinden faydalanılarak yapılır. Bu tür saldırıları engellemenin iki yolu vardır; ilki tamamen güvenli bir sistem ve ortam oluşturmak, ikincisi ise saldırıları tespit edip gerekli önlemleri almaktır. Bunlardan ilki pratik açıdan mümkün olmamaktadır. Bunların gerekçeleri ise [326] nolu kaynaktan özetlenmiş ve aşağıda verilmiştir. Bunlar;

- Kullanılan işletim sisteminde var olan açıkların genellikle ilk olarak saldırganlar tarafından fark edilmesi ve önlem alınana kadar bu açıkların kullanılabilmesi,
- Veri iletişimde kullanılan protokollerin yapısında var olan bazı kuralların saldırı amaçlı kullanılabilmesi,
- Kriptografik metotların ve anahtarlarının kırılabilmesi, kullanıcıların şifrelerini unutulması veya kriptosistemin kırılabilmesi gibi nedenlerle yüksek seviyede bir güvenlik sağlanamaması,
- Dış ortama karşı güvenliği sağlanan sistemin, iç ortamlardan suistimal edilerek güvenliğin ortadan kaldırılması,
- Güvenlik amacıyla kullanıcı yetkilerinin minimuma indirilmesi sonucu kullanıcı verimliliğinin düşmesi gibi nedenlerdir.



Şekil 3.3. Temel STS yapısı

Belirtilen bu gerekçelerden dolayı, tamamen güvenli bir sistem oluşturmanın mümkün olmadığı görüldüğüne de üst düzey güvenliğe sahip bir sistem oluşturmak mümkündür. Bunun için maliyet göz önünde bulundurulduğunda, gerekli olan tüm güvenlik araçlarından faydalanılmalıdır. Ancak saldırganların, kapatılan açıkların ardından, saldıracak yeni açıklar bulabilmesinin ve önceden tahmin edilemeyen yollara başvurmalarının, tamamen güvenli bir

ortamın oluşturulamamasına neden olduğu unutulmamalıdır. STS'lerin bu konudaki önemi, kullanılan yeni teknikler sayesinde önceden bilinmeyen saldırıların tespit edilebilmesini kolaylaştırmaktadır [321]. Saldırı tespit sistemlerinin güvenlik yönünden sağladığı üstün özelliklerinin yanı sıra tüm güvenlik sorunlarının üstesinden gelememektedir. STS'ler güvenlik seviyesinin artırılmasına yardımcı olan araçlar olarak karşımıza çıkmaktadır.

Bir bilgi sisteminde üst düzey güvenlik seviyelerine ulaşılabilmesi için STS'ler birçok farklı güvenlik çözümü ile entegre olarak bütüncül bir güvenlik anlayışının bir parçası olmalıdır. Bir bilgi ekosisteminde uygulanması gereken diğer güvenlik önerileri bu bölümün konusu olmadığı için burada değinilmeyecektir. Ancak, bir saldırı tespit sisteminden beklenmesi gerekenler veya bu sistemlerin yapabilecekleri hakkında özet bilgiler verilerek STS'lerin kabiliyetleri hakkında genel bir fikir oluşturulması amaçlanmaktadır. STS'lerin kabiliyetleri aşağıda kısaca özetlenmiş olup bunlar [327];

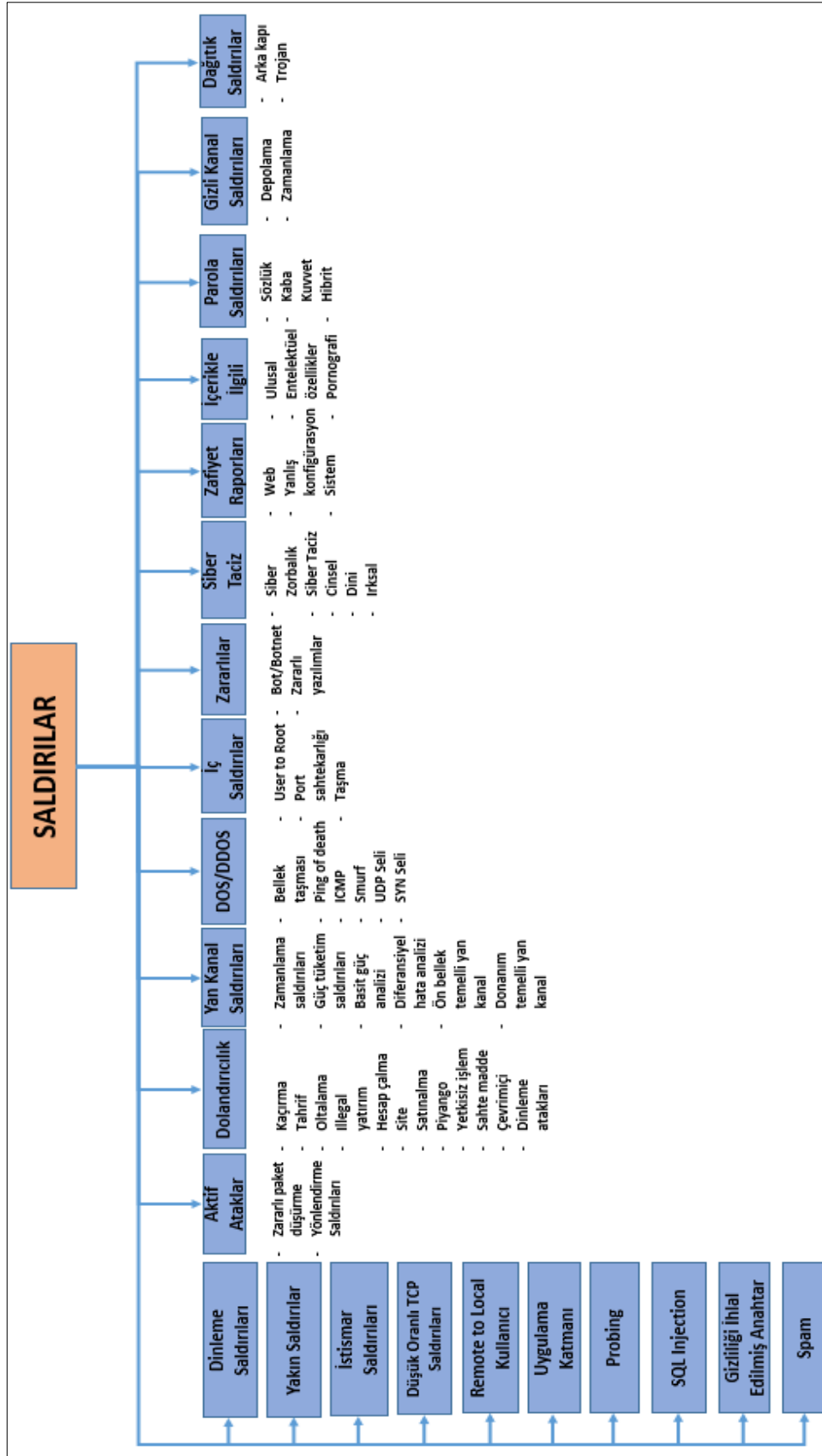
- Bilgi sistemleri altyapılarına bütüncül bir akış açısı sağlamaktır.
- Kullanıcı aktivitelerini bilgi sistemlerine giriş noktasından etki noktasına kadar takip edebilmektir.
- Verilerdeki değişiklikleri tanıyabilmekte ve bildirebilmektedir.
- Güncel saldırıları aramak için internet ortamı izleme görevini otomatikleştirebilmektedir.
- Bilgi sistemlerinin saldırı altında olduğunu tespit edebilmektedir.
- Sistem konfigürasyonlarındaki hataları tespit edebilmektedir.
- Bilgi varlıklarına yönelik plan, politika ve prosedür oluşturulmasında ağ, sistem ve güvenlik yöneticilerine rehberlik etmektedir.
- Uzman personel eksikliğinden kaynaklı oluşabilecek güvenlik yönetimi sorunlarında süreçlerin takip edilmesini kolaylaştırmaktadır.
- Yoğun ve hızlı bir ağda trafiğin mümkün olduğunca izlenmesini ve incelemesini kolaylaştırmaktadır.

Saldırı tespit sistemlerinin kabiliyetlerinin yanı sıra bir STS'de bulunması gereken ideal ve güncel özellikler ise [328];

- Sıfıncı gün saldırılarını algılayabilme,
- İnsan müdahalesine yok denecek kadar az gereksinim duyma,

- Herhangi bir uygulama ya da servise yönelik geliştirilebilir yapıya sahip olma,
- Taklit saldırıları (mimicry) saldırıları algılayabilmeli,
- Düşük ya da kontrol edilebilir düzeyde yanlış-pozitif ve yanlış-negatif sayısına sahip olmalı,
- Yüksek bant genişliği bulunan ortamlarda etkin ve yönetilebilir bir yapıya sahip olmalı ve
- Gerçek zamanlı ya da buna yakın sayılabilecek bir gecikme ile çalışmalıdır.





Şekil 3.4. Saldırı türleri ([329] nolu kaynaktan alıntılanarak Türkçeleştirilmiştir.)

3.1. Saldırı Tespit Sistemlerinin Sınıflandırılması

STS'ler birçok farklı kritere göre sınıflandırılmaktadır. Şekil 3.5'de gösterildiği gibi en genel sınıflandırma kriterleri; saldırı tespit yöntemi, bilgi kaynağı, mimari yapısı, hedef sistemin türü ve zaman davranışıdır. Bir STS, sınıflandırmanın hangi kritere göre yapıldığına bağlı olarak Şekil 3.5'de gösterilen farklı sınıflarda yer alabilmektedir. Bir STS, veri işleme zamanına göre gerçek zamanlı, mimari yapısı bakımından merkezi sistemli, kullanılan bilgi kaynağı olarak sunucu tabanlı, saldırı tespit yöntemi olarak anormallik tespiti yaklaşımı, koruduğu sisteme göre sunucu temelli sınıfına dâhil olabilmektedir. Örneğin, belirlenen sınıflandırma kriteri sadece saldırı tespit yöntemi ise anormallik tespiti veya imza tabanlı yaklaşım kullanan sınıfının bir üyesidir. Bu durumda diğer kriterlerden bahsedilmez. Bununla birlikte, bu kriterler bir STS'nin karakteristiğini ortaya koymaktadır.

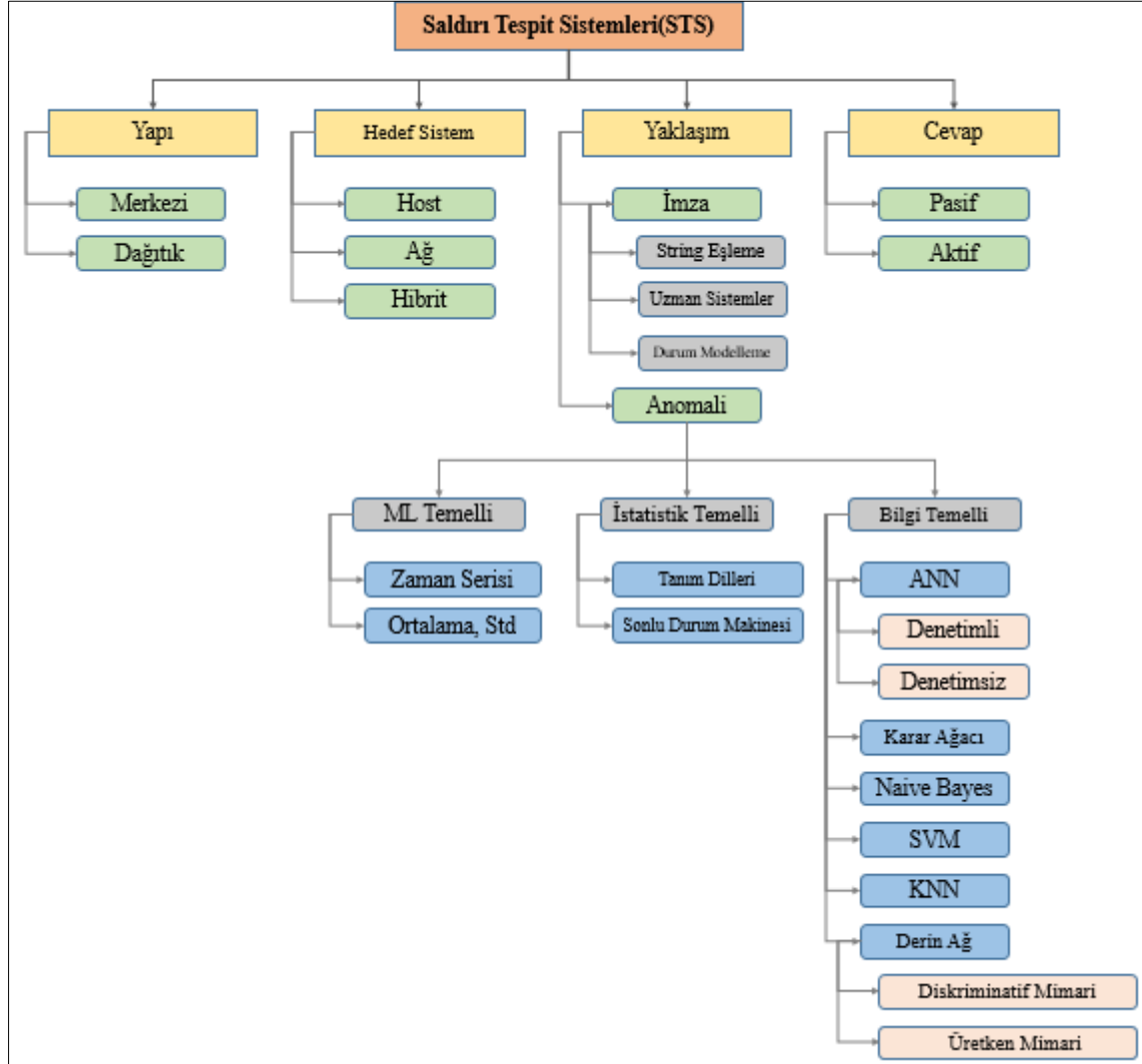
3.1.1. Zaman davranışına göre sınıflandırma

Saldırı tespit sistemleri zaman davranışına göre “gerçek zamanlı” ve “gerçek zamanlı olmayan” şeklinde ikiye ayrılmaktadır. Burada bahsedilen zaman kavramı, bir STS'nin saldırıları tespit edebilmesi amacıyla izlediği veriler ile bu verilerin analizi arasında geçen süreyi ifade etmektedir.

Gerçek zamanlı saldırı tespit sistemlerinde; veriler iletişim anında analiz edilmektedir ve bir saldırı tespit edilmesi durumunda oluşturulması gereken tepki aktif olarak anında verilmektedir. Bu tür STS'lerin yoğun veri akışı olan sistemlerde kullanılması zor ve maliyetli olabilmektedir. Ancak, saldırılara aktif olarak cevap üretilmesi gereken sistemler için de vazgeçilmez bir çözüm olmaya devam etmektedir. Bu yaklaşım, olası saldırıların ağ üzerinden yayılımını hızlı bir şekilde engelleyebilmek amacıyla genellikle ağ-temelli STS'lerde daha fazla tercih edilmektedir.

Gerçek zamanlı olmayan saldırı tespit sistemlerinde ise; elde edilen veriler depolanmakta ve belirli periyotlarla analiz edilmesi için STS'lere gönderilmektedir. Bu yöntemde veri izleme noktalarından STS analiz motorlarına veri akışı sürekli değildir. Saldırı profilinin çıkarılması ve bu tür saldırılara bir daha maruz kalınmaması için, geçmişe yönelik depolanan veriler üzerinde analizler yapılmaktadır. Bu tür STS'ler daha çok saldırının etki analizi veya

raporlama amacıyla kullanılmaktadır. Bundan dolayı, koruyucu bir yaklaşım olarak değil saldırı izlerini tespit edici bir yaklaşım olarak tanımlanabilir.



Şekil 3.5. STS sınıflandırması [330] numaralı kaynaktan alıntılanmış ve Türkçeleştirilmiştir.

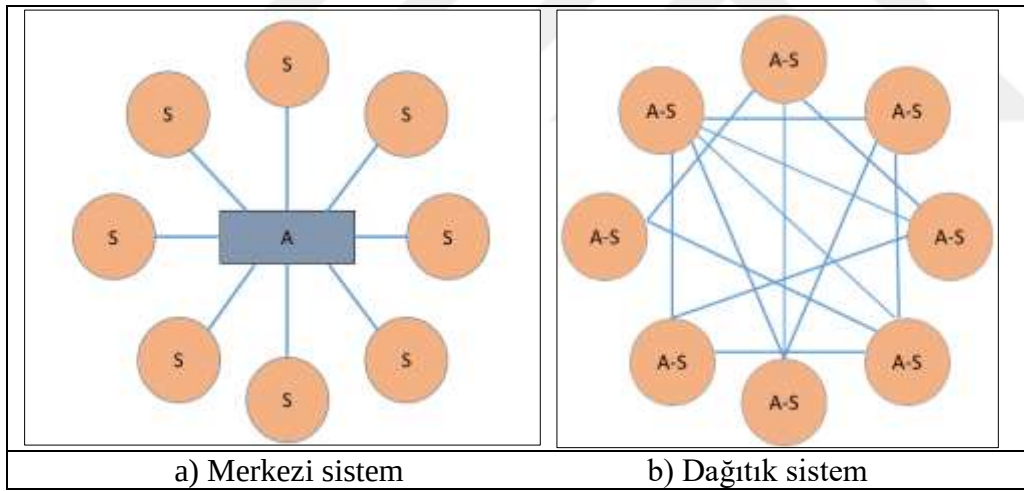
3.1.2. Mimari yapıya göre sınıflandırma

STS'lerin mimari açıdan sınıflandırılması, konumlandırılan analiz noktalarının nasıl yerleştirildiğine bağlı olarak oluşturulmaktadır. Bu tür sınıflandırma merkezi sistem ve dağıtık sistem olmak üzere ikiye ayrılmaktadır.

Merkezi sistem saldırı tespit sistemleri veri toplayan sensörler ve merkezi analiz birimi olmak üzere iki bileşenden oluşmaktadır. Bu tür STS'lerde; tüm izleme, tespit ve raporlama

işlemleri Şekil 3.6'da gösterildiği gibi merkeze yerleştirilen bir merkezi analiz biriminde (A) sistem tarafından kontrol edilmektedir. Bilinen STS'lerin çoğu bu kategoriye girmektedir. Merkezi sistem STS'lerde analiz birimi farklı ataklara karşı tek bir hata noktası (single point of failure) olmaktadır. Bu durumda, merkezi analiz birimine karşı yapılacak başarılı bir saldırıda tüm sistem savunmasız hale gelecektir.

Dağıtık sistemlerde analiz ve veri toplama kabiliyetine sahip tek tür bileşenler bulunmaktadır. Bu tür sistemlerde; izleme, tespit ve cevap işlemleri Şekil 3.6'da gösterildiği gibi ağ üzerinde birçok noktada konumlandırılan hem sensör hem analiz yeteneği (A-S) bulunan ajanlar (agent) yoluyla yapılmaktadır. Bir sensör diğer bir sensörde bulunan bilgiye ihtiyaç duyduğunda bir istek göndererek o bilgiyi alır ve kendi üzerinde işleyerek istenilen sonucu çıkarmaktadır. Genellikle geniş noktalara yayılmış ağlarda kullanılan bir yöntemdir. Bu sistemler, tek bir hata noktası olmaması ve daha kolay ölçeklenebilir olmalarından dolayı bazı amaçlar için merkezi sistemlere göre daha avantajlı olmaktadır.



Şekil 3.6. STS mimari yapısı ([331] nolu kaynaktan alıntılanmıştır.)

3.1.3. Bilgi kaynağına göre sınıflandırma

Saldırı tespit sistemleri, analiz, karşılaştırma, tespit ve önleme yapmak için bilgi kaynaklarını kullanmaktadırlar. Bilgi kaynakları, çeşitli yollarla bilgisayar veya ağ paketlerinin dinlenmesinden elde edilebildiği gibi, bilgi sistemi kullanıcılarının davranış modellerinden de elde edilebilmektedir. Saldırı tespitinde kullanılacak olan bilginin nasıl ve nereden toplanacağı, geliştirilecek olan STS'nin amaçlarına göre değişmektedir. Başlıca

bilgi kaynakları olarak denetleme izleri, ağ paketleri ve uygulama kayıt dosyaları sıralanabilir.

Denetleme izleri sistem davranışına ilişkin en eski bilgi kaynaklarından biridir. İşlemci süresi, bellek, disk veya ağ kullanımı gibi paylaşılan kaynakların tüketimi ve sistem kullanıcıları tarafından başlatılan uygulamalar hakkında bilgi sağlamaktadır. Bu kayıtlar, ağ cihazlarında, sunucularda ve çeşitli iş istasyonları gibi hemen hemen her bilgi sisteminde bulunmaktadır. Denetleme izlerinin her bilgi sisteminde bulunması, saldırı tespit sistemlerinin onu bir tespit kaynağı olarak kullanmaya çalışmasına neden olmuştur. Bu kayıtlar, işletim sistemlerine göre farklılık göstermekle birlikte belirli dosya konumlarında tutulmaktadır. Denetim izleri, işletim sistemlerine iyi entegre edilmiştir ve kullanımı kolaydır. Bununla birlikte, denetleme izi bilgilerinin de bir takım dezavantajları vardır. Bilgi sistemi üzerinde yapılan işlemlerin doğru bir şekilde tespit edilebilmesi, zaman damgalarının doğru olarak işlenmesine bağlıdır. Zaman bilgisi yanlış olan kayıtlar üzerinden saldırılar etkili bir şekilde tespit edilememektedir.

Bir diğer denetim kaydı sağlayan Syslog ise, işletim sistemleri tarafından uygulamalara sağlanan bir denetim hizmetidir. Syslog, bilgi sisteminden bir metin dizesi olarak bir zaman damgası ve bilgi sisteminin çalıştığı sistemin adını ekler. Ardından bu kayıtlar, yerel olarak bilgi sisteminin kendi üzerinde veya harici olarak istenilen bir sunucuya arşivlenebilmektedir. Syslog saldırı tespit sistemlerinde veya bilgi sistemlerinin izlenmesinde önemli ölçüde kullanışlı olmasına rağmen çeşitli saldırılara da konu olabilmektedir. Syslog hizmetinde bulunan zafiyetler bilgi sistemlerinde de güvenlik açıklıklarına sebep olabilmektedir. Syslog'un kullanımı çok kolaydır. Bu yüzden, birçok bilgi sistemi geliştiricisi bu hizmeti denetim izi olarak kullanmaya başlamıştır. Swatch gibi Syslog tarafından sağlanan bilgileri kullanan birkaç saldırı tespit aracı geliştirilmiştir.

Basit Ağ Yönetimi Protokolü (SNMP) Yönetim Bilgi Tabanı (MIB), ağ yönetimi amacıyla kullanılan bir bilgi kaynağıdır. Ağ cihazları üzerinde yönlendirme tabloları ve adresler gibi yapılandırma bilgilerini ve çeşitli ağ arayüzlerinde ve ağın farklı katmanlarında trafiği ölçmek için sayaçlar kullanarak performans verilerini elde etmektedir. SECURENET projesi [332], SNMP MIB'lerin saldırı tespit sistemleri için bir denetim kaynağı olarak kullanılabileceğini göstermektedir.

Saldırı tespit sistemlerinde başlıca bilgi kaynaklarından birisi de ağ paketleridir. Ağ dinleyicileri (network sniffers) bilgi toplamak amacıyla saldırganlar tarafından oldukça fazla kullanılmaktadır. Saldırganlar tarafından kullanılan bu yöntem ağ tabanlı saldırı tespit sistemlerinde ağ üzerinde meydana gelen olaylar hakkında bilgi toplamak için de etkili bir araç olarak kullanılmaktadır. Bilgi sistemlerine erişimler genellikle ağ üzerinden gerçekleşmektedir ve bu nedenle paketleri sunucuya girmeden önce yakalamak sunucuyu izlemenin en etkili yollarından biridir. Hizmet reddi veya aksattırma (DoS) saldırıları gibi ağ üzerinden erişimi engellemeyi amaçlayan saldırıların ağ düzeyinde tespit edilerek saldırılar engellenmelidir. Bunu başarabilmenin yolu ise ağ paketlerinin analiz edilerek saldırıların tespit edilmesi ve önlenmesidir. Bu sebepten ağ paketleri saldırı tespit sistemlerinde önemli bilgi kaynaklarından biridir.

Saldırı tespit sistemlerinde önemli bilgi kaynaklarından biri de uygulama günlük dosyalarıdır. Bilgisayar ve internet altyapısının her geçen gün daha fazla kullanımı ile bu ağlar üzerinden hizmet veren uygulamaların sayısı da önemli ölçüde artış göstermiştir. Uygulamalara yönelik olası saldırıların tespiti ve önlenmesinde uygulama günlük dosyaları büyük önem arz etmektedir. Uygulama günlük dosyalarında uygulamanın kullanım istatistikleri, kullanıcı-uygulama etkileşimine ait bilgiler ve uygulamaya özgü sistem kayıtları bulunmaktadır. Bu bilgilerin takip edilmesi ve raporlanması uygulama geliştiricilerine gelecek çalışmalarla ilgili yol gösterirken saldırı tespit sistemlerinde de önemli bir bilgi kaynağı olmaktadır.

3.1.4. Saldırı tespit yöntemine göre sınıflandırma

STS'lerde saldırı tespit yöntemi olarak genellikle davranış modellemeye dayalı anormallik tespiti ve imza kayıtlarına dayalı kötüye kullanım tespiti olmak üzere iki farklı yaklaşım kullanılmaktadır. Anormallik tespitine dayanan yaklaşım, sistemdeki kullanıcı davranışlarını modellerken, kötüye kullanım (imza) tespitine dayanan yaklaşım, saldırganların davranışlarını modellemektedir.

Kötüye kullanım tespiti (misuse detection) yaklaşımında; saldırganların davranışlarının ve sistem açıklıklarının modellerinin çıkarılması esastır. Saldırının veya zafiyetin imzası olarak da nitelendirilen bu modeller, daha önce karşılaşılmış saldırıların analiz edilerek kendine has karakteristik özelliklerinin çıkarılması ile elde edilmektedir. Saldırılara ve açıklıklara

yönelik imza veri tabanları oluşturulduktan sonra, bu imzalarla eşleşen hareketler saldırı olarak tespit edilmektedir. Bu yöntemde, açık bir şekilde saldırı olarak tanınmayan herhangi bir davranış normal davranış olarak kabul edilmektedir. Bu nedenle, bu tür saldırı tespit sistemlerinin doğruluğu yüksek olmakta ve yanlış pozitif oranı oldukça düşük olmaktadır. Bununla birlikte bu STS'lerin en önemli dezavantajı, saldırılar ve zafiyetler hakkındaki imzaların düzenli olarak güncellenmesi gerekliliğidir. Saldırıları veya zafiyetleri işletim sistemlerine, sürüm bilgisine, platforma, uygulamalara ve protokollere bağlı olduğundan imza veri tabanlarının güncel tutulması ve mümkün olan her türlü saldırı ve zafiyet imzasını içermesi ciddi bir iş yükü getirmektedir. Düzenli güncellenme yapılmayan STS'lerde güncel saldırıların tespit edilmesi imkânsız hale gelmektedir. Bu STS'lerin bir diğer dezavantajı ise bilgi sistemi içerisinde çeşitli güvenlik açıklıklarını kullanmadan yapılan saldırıları tanıyamamaktadır.

Anormallik tespiti yaklaşımı; temel olarak, sistemde meydana gelen anormal olayları, normal olaylardan ayırt etme yaklaşımıdır. Bu yöntemde, sistemin veya kullanıcıların normal veya beklenen davranışından bir sapmanın gözlemlenmesiyle bir saldırının tespit edilebileceği varsayılmaktadır. Normal davranış modeli, bilgi sistemlerinden çeşitli yollarla toplanan bilgilerden çıkarılmaktadır. Saldırı tespit sistemleri, normal davranış modelini mevcut aktivite ile karşılaştırır. Modelden bir sapma gözlemlendiğinde, STS tarafından bir alarm üretilir. Özetle, önceden öğrenilmiş bir davranışa karşılık gelmeyen herhangi bir sistem davranışı saldırı olarak kabul edilir. Bu özet bilgilerin ardından anormallik tespiti yaklaşımı ilerleyen bölümlerde daha detaylı anlatılacağı için burada daha fazla detay verilmemiştir.

3.1.5. Hedef sistemlere göre sınıflandırma

STS'ler konumlandırıldıkları veya izledikleri bilgi sistemlerine göre üç gruba ayrılmaktadırlar. Bu yöntemlerde temel ayrım korunan veya izlenen bilgi sistemlerine göre yapılmıştır. Bundan dolayı, ağ, sunucu ve uygulama temelli STS'ler olarak sınıflandırılmaktadırlar.

Ağ temelli STS'ler; bilgi kaynağı olarak ağ paketlerini izleyerek ağ üzerinden yapılan saldırıları tespit etmektedirler. Ağ paketlerinin yakalanarak bunların analiz edilmesi ile saldırı tespitini gerçekleştirirler. Bir ağ segmentini veya ağ anahtarını izleyen bir ağ tabanlı

STS, ağ segmentine bağlı birden fazla sunucuyu etkileyen ağ trafiğini izleyebilir ve ağdaki sunucuların tamamını koruyabilir. Ağ tabanlı STS'ler genellikle bir ağdaki çeşitli noktalara yerleştirilmiş bir dizi sensör veya sunucudan oluşmaktadır. Bu sensörler, ağ trafiğini izler, yerel analizini gerçekleştirir ve saldırıları merkezi bir yönetim konsoluna bildirir. Ticari saldırı tespit sistemleri genellikle ağ tabanlı çalışmaktadır. Bu STS'lerin en önemli avantajı büyük ölçekteki ağları bile izleyerek ağda bulunan birçok bilgi sistemini koruyabilmeleridir. Bir diğer avantajları ise pasif yapıda olduklarından dolayı ağ topolojisinde değişikliğe gerek kalmadan kolay entegre edilebilmeleridir. Ağ tabanlı STS'ler bu avantajlarının yanı sıra bazı dezavantajlara da sahiptir. En önemli dezavantajları, ağ trafiğinin çok fazla olduğu bir ağdaki tüm paketleri işlemede zorluk yaşayabilir ve ağ üzerinde yoğun trafik oluşan dönemlerde başlatılan bir saldırıyı algılayamayabilir. Bir diğer dezavantaj olarak ağ tabanlı STS'ler ağ üzerindeki şifreli trafiği analiz edememektedir. Bunun yanı sıra, bu STS'ler saldırıların sadece başladığı bilgisini elde edebilmektedir, bilgi sistemleri üzerindeki etkileri veya sonucu hakkında bilgi sağlayamazlar. Bu bilgiler sunucu veya uygulamalar üzerinden elde edilecek verilerle belirlenebilmektedir.

Sunucu temelli STS'ler; bir bilgisayar sistemi içerisinde işletim sistemi hesap izleri ve sistem kaydı gibi verileri depolamakta ve toplanan bu veriler üzerinde çalışmaktadırlar. Bu şekilde toplanan veriler ile işletim sistemine yönelen saldırılar için hangi sistem çağrılarının ve hangi kullanıcıların sorumlu olduğu tespit edilebilmektedir. Sunucu temelli STS'lerin saldırı tespitinde araştırılan ilk alan olduğu belirtilmektedir [333]. Ağ tabanlı saldırı tespit sistemlerinden farklı olarak, tüm süreç sunucunun kendi üzerinde işletilmektedir. Bu sistemler her bir sunucuya yerel olarak dağıtılır ve yalnızca kurulu olduğu sunucuyu izlemektedir. Bu tür STS'ler genellikle saldırıya daha fazla maruz kalabilecek veya kritik iş ve veri bulunduran sunucular için kullanılmaktadır. Sunucu-temelli STS'ler, denetim izi ve sistem günlük dosyaları olmak üzere iki tür bilgi kaynağı kullanmaktadır. Bunların ağ temelli sistemlere göre bir avantajı, yerel sunucuyu izleme yetenekleri sayesinde ağ temelli STS'lerin yakalayamayacağı saldırıları tespit edebilmeleridir. Bununla birlikte, yapılan saldırıların sonuçları ve etkileri tespit edilebilmektedir. Ağ temelli sistemlerden bir diğer avantajı ise, ağ paketlerinin şifrelenmeden veya şifresi hemen çözüldükten sonra izlenerek saldırıları tespit edebilmesidir. Bu avantajlarının yanı sıra, izlenmesi istenen her sunucu için kurulum ve yapılandırma işlemlerinin yapılması gerekmektedir. Bu durum sistemin işletilmesini ve yönetilmesini zorlaştırmaktadır. Ayrıca, sunucu temelli STS'ler DoS atakları devre dışı bırakılabilmektedirler.

Uygulama temelli STS'ler; olayları bir yazılım uygulaması ile analiz eder ve sunucu tabanlı STS'lerin özel bir alt kümesidir. Uygulama temelli STS'lerde bilgi kaynağı olarak genellikle uygulamaya ait günlük dosyalar kullanılmaktadır. Analiz motoruna, belirli uygulamalar için o uygulamalara has özellikler bildirildiği takdirde, yetkisini aşan kullanıcıların gerçekleştirdiği saldırılar uygulama temelli STS'lerce tespit edilebilmektedir [334]. Uygulama temelli STS'ler, kullanıcı ve uygulama arasındaki etkileşimi izleyebilmektedir, bu durum da genellikle yetkisiz etkinlikleri bireysel kullanıcılara kadar izlemelerine olanak tanımaktadır. Bunun yanı sıra, şifreli ortamlarda çalışabilmeleri de bir diğer avantajlarıdır. Bu tür STS'lerin başarılı olabilmeleri için uygulama kayıt dosyaları özenli bir şekilde korunmalıdır.

3.2. Anormallik (Aykırılık) Temelli Saldırı Tespit Sistemleri

Davranış kavramı temel olarak bir bilgi sisteminde bulunan uygulamalar, ağ cihazları ve kullanıcılar gibi çeşitli bilgi kaynakları tarafından gerçekleştirilen olaylar veya oluşturulan trafikler olarak tanımlanabilir. Davranış modelleme ise bir bilgisayar ağında bilgi kaynakları tarafından oluşturulan davranışları kullanarak incelenmek üzere sistemin bir özetinin oluşturulması olarak tanımlanabilir. Bir bilgisayar ağında bulunan bilgi kaynaklarının oluşturduğu davranışların tanımlanması, modellenmesi ve takip edilmesi saldırıların tespit edilmesinde etkin bir yöntem olarak kullanılmaktadır.

Bir bilgi sisteminin normal davranışlarının belirlenmesi ile sistemde olağan dışı davranışlara sebep olan saldırılar aykırı olarak işaretlenerek tespit edilmektedir. Burada normal davranışın bilinmesi ve modellenmesi esastır [335]. Bu tür saldırı tespit sistemleri anormallik temelli saldırı tespit sistemleri olarak tanımlanmaktadır. Anormallik temelli saldırı tespiti literatürde yaygın olarak davranış modelleme olarak da bilinmektedir. Anomali tespiti, kullanıcının ve/veya uygulamanın davranışına dayanmaktadır; buna kullanıcı profili veya uygulamanın davranışı denir. 1980'de Anderson tarafından önerilmiş ve 1987'de Denning [336] tarafından devralınmıştır. Anderson, kullanıcı profilini, daha önce öğrenilmiş olan olağan davranıştan sonraki sapmayı tespit etmek için davranışını modelleyen bir dizi ilgili önlemlerle açıklamayı önermiştir. Anomali tespit sistemleri şu soruyu yanıtlamaya çalışmaktadır: "Kullanıcının ve/veya uygulamanın mevcut davranışı geçmiş davranışıyla tutarlı mıdır?"

Anormallik veya adı üstünde aykırılık normal davranıştan sapma anlamına gelmektedir. Anormallik tespiti yaklaşımı, temel olarak sistemde meydana gelen anormal olayları, normal olaylardan ayırt etme mantığı ile çalışmaktadır. Anormallik tespiti temelli bir saldırı tespit sisteminin temel yapısı Şekil 3.7’de gösterilmektedir. Bir bilgi güvenliği sistemi içerisinde yer alan STS için anormallik, normal davranışlardan herhangi bir sapma durumuna karşılık gelmektedir. Anormallik dedektörleri; kullanıcıların, sunucuların veya ağ bağlantılarının temsil edilen normal davranışlarının profillerini oluşturmaktadırlar. Normal davranış Şekil 3.8’de gösterilen bir dağılım ile veya dağılımın momentleriyle modellenilebilir. Belirli kurallar zinciri ile tanımlanabilir veya başka yöntemler uygulanabilir ama sonuçta bu tanımlı ya da öğrenilen davranıştan sapma anormallik olarak değerlendirilir ve sapmanın şiddetine göre saldırı olarak sınıflandırılabilir [338]. Sistemin normal davranışı, uzun süren bir analiz sonucunda elde edilebilmektedir. Sistemdeki kullanıcı veya kullanıcı gruplarının davranış profillerinin belirlenmesi anormallik tespiti için temel işittir. Normal davranış profili belirlendikten sonra, farklılık gösteren davranışlar saldırı olarak tespit edilmektedir. Anormallik tespitinde saldırıların doğru tespit edilmesi, normal davranış profilinin ne kadar doğru belirlendiğiyle doğru orantılı olarak değişmektedir.

Anormallik tespiti yapan sistemlere öğrenen sistemlerde denmektedir. Çünkü sistemdeki normal faaliyetler sürekli güncellenmektedir. Sistemdeki bütün faaliyetler sürekli izlenmekte ve tespit edilen normal davranışlar ile kıyaslanmaktadır. Aykırılık tespitinde bütün sızma faaliyetlerinin mutlaka alışılmışın dışında gerçekleşeceği varsayılmaktadır. Örneğin, sistem üzerindeki istatistiklere bakarak normal kullanım modelleri (CPU kullanımı, çalışma saatleri vb.) belirlenir ve bu normal davranışların dışındakiler saldırı olarak kabul edilir [338]. Bu yöntemin zorluğu, sistemin normal davranış özelliklerinin belirlenmesindeki zorluklardır. Bu yöntemde yanlış veya yetersiz modelleme nedeniyle normal işlemler yanlışlıkla saldırı olarak kabul edilebilir. Bu yöntemin avantajı ise sıfırcı gün atakları gibi yeni saldırıların tespit edilebilmesidir. Anormal davranışların belirlenmesinde genellikle aşağıda belirtilen davranış profilleri kullanılmaktadır [338].



Şekil 3.7. Anormallik tespiti yaklaşımının temel yapısı ([319] nolu kaynaktan alıntılanarak Türkçeleştirilmiştir.)

Bireysel Profiller: Bir kullanıcının yapması beklenen genel faaliyetleri kapsamaktadır. Bu faaliyetler, çalışma saatleri ve çalışma biçimi gibi bireysel aktivite örüntüleridir.

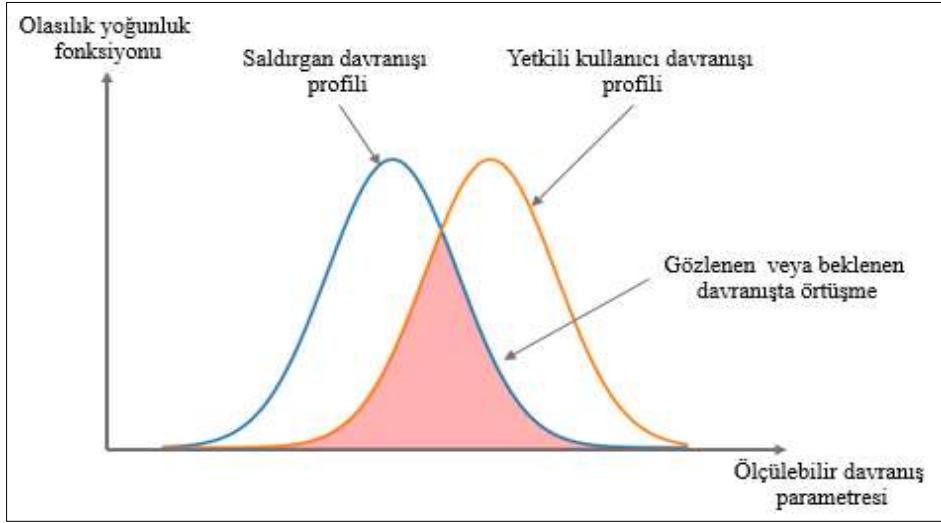
Grup Profiller: Bir gruptaki kullanıcıların profilini kapsamaktadır. Kullanıcıların çalışma biçimi, kullandıkları kaynaklar, tarihsel aktiviteleri kapsar.

Kaynak Profili: Uygulamalar, kullanıcı hesapları, iletişim portları gibi kaynakların nasıl kullanıldığı gözlemlenmektedir.

Diğer Profiller: Çalıştırılabilir programların sistem kaynaklarını nasıl kullandığı gözlemlenmektedir.

3.3. Anormallik Tespitinde Kullanılan Yöntemler

Anormallik tespitinde istatistiksel yöntemlerden veri madenciliğine kadar birçok yaklaşım uygulanabilir [340]. Şekil 3.9, anormallik temelli saldırı tespit tekniklerinin taksonomisini göstermektedir. Bunlar; istatistiksel, bilgiye dayalı, makine öğrenmesi tabanlı, veri madenciliği tabanlı, kullanıcı niyet tanımlama ve bilgisayar immünolojisi olarak sınıflandırılmaktadır [341].

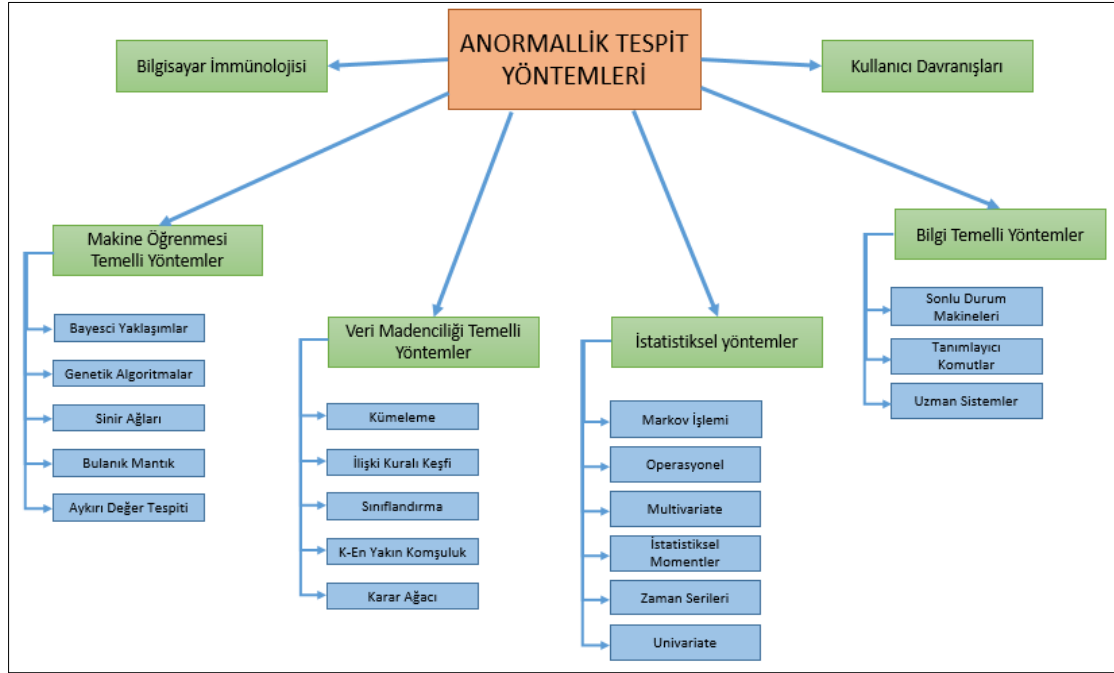


Şekil 3.8. Normal ve saldırı içeren dağılımın teorik ayrışımı [339] nolu kaynaktan alıntılanarak Türkçeleştirilmiştir

3.3.1. İstatistiksel-temelli yöntemler

İstatistiksel temelli yöntemler, normal profili oluşturmak için normal işlemler üzerinde ortalama ve varyans gibi istatistiksel özellikleri kullanmaktadır [342]. İstatistiksel testler, gözlemlenen işlemin normal profilden sapıp saptadığını belirlemek için kullanılmaktadır. Bu yöntemlerde saldırı tespit sistemleri, profili normalden farklı olan işlemlere bir puan atar. Eğer bu atanan skor eşik değerine ulaşırsa, alarm verilmektedir. Eşik değeri, bir süre boyunca meydana gelen olayların sayısına göre ayarlanmaktadır. Eşik değerinin ne olması gerektiği önemli bir konudur. Çünkü bu değer tanıma doğruluğunu etkileyen konulardan biridir. Eşik değeri verilirken genellikle uzman görüşünden faydalanılır. Yani sistemin kurulması esnasında o sistem için hangi eşik değerinin normali verdiği bilinerek sabit bir değer verilir. Daha sonra bu değer doğru tanıma oranına bakılarak aşağı yukarı hareket ettirilir. Ayrıca her anormallik tespiti sistemi için eşik değeri farklı olacaktır [343].

İstatistiksel temelli teknikler operasyonel model veya eşik ölçümü, zaman serisi modeli, Markov süreç modeli veya Marker modeli, parametrik yaklaşımlar, istatistiksel momentler veya ortalama ve standart sapma modeli, çok değişkenli model ve parametrik olmayan yaklaşımlar olarak sınıflandırılmaktadır [341].



Şekil 3.9. Anormallik temelli saldırı tespit sistemlerinin sınıflandırılması [341] nolu kaynaktan alıntılanarak Türkçeleştirilmiştir

İstatistiksel tabanlı yöntemler, verinin dağılımı ile ilgili ön kabulde bulunarak (parametrik yaklaşım) veya ön kabulde bulunmayarak (parametrik olmayan yaklaşım) oluşturduğu olasılıksal modeller ile anomalileri tespit etmeye çalışmaktadır. Parametrik yaklaşımlar analiz edilmiş verinin dağıtılmasına temel oluşturan analitik metotlarla ilgilidir. Parametrik olmayan istatistiksel anormallik tespitinde, kullanıcı aktiviteleri verisi iki farklı kümeye alınmaktadır. Biri normal olmayanlar, diğeri normal aktiviteleri göstermektedir. Parametrik olmayan yaklaşımın avantajı; güvenilir olay verisi indirgemeleri yapabilmesi, tespit hızının arttırması, parametrik istatistiksel analizdeki doğruluğu geliştirmesidir.

Zaman serileri, veri noktalarının sıklığını ifade eder ve düzenli zaman aralıklarında, ardışık zaman alanlarında tipik olarak ölçülmektedir. Zaman serisi tahmini önceden bilinen olayları baz alarak gelecek olayları tahmin etmenin kavramsal modelidir. Zaman serisi verisi analizlerinde model kurmak yerine serinin kendisi analiz edilir. Her veri seti, zaman serisi değildir. Bir veri setinin zaman serisi olabilmesi için değerlerinin zamana bağlı olarak değişmesi gerekir. Bir zaman serisinde analizlere başlamadan önce kontrol edilmesi gereken durum, verinin durağan olup olmamasıdır. Zamanla varyansın ve ortalamanın sabit olması gerekir. Kısaca şöyle tanımlayabilmektedir; zaman içinde varyansın ve ortalamanın sabit olması ve gecikmeli iki zaman periyodundaki değişkenlerin kovaryansının değişkenler

arasındaki gecikmeye bağlı olup, zamana bağlı olmamasıdır. Zaman serileri analizinin içerdiği yöntemler nicel yöntemlerdir. Dolayısıyla, zaman serileri analizi zaman içinde düzenli aralıklarla gözlemlenen verilerin istatistiksel olarak incelenmesini ve gelecek dönemlerde elde edilebilecek verilerin öngörüsünün güvenilir bir şekilde yapılabilmesini içermektedir.

İstatistik tabanlı tekniklerin temel avantajları şunlardır [342,343,344]:

- i. Saldırıların imzaları hakkında önceden bilgi gerektirmezler. Böylece sıfıncı gün saldırılarını tespit edebilirler.
- ii. Veri dağılımının altında yatan varsayımlar doğruysa, istatistiksel teknikler anormallik tespiti için bir çözüm sunmaktadır.
- iii. Sistem imzaların hiçbirine bağlı olmadığından güncelleme gerekli değildir. Bu nedenle bakımı kolaydır.
- iv. Uzun süre boyunca meydana gelen izinsiz giriş faaliyetleri doğru bir şekilde tanımlanabilir ve DoS saldırılarını tespit etmede iyidir.

İstatistik tabanlı tekniklerin dezavantajları aşağıdaki gibidir [342,343,345]:

- i. Doğru istatistiksel dağılımlara ihtiyaçları vardır.
- ii. İstatistik tabanlı tekniklerin öğrenme sürecinin doğru ve etkili olması günler veya haftalar almaktadır.
- iii. İstatistiksel temelli sistemler, saldırganların zararlı davranışlarını normal gibi "öğretebilme" yeteneğine karşı savunmasızdır. Örneğin, bir saldırgan Nmap gibi bir program kullanabilir ve ağda çok sayıda SYN taraması gönderebilir. Zamanla bu faaliyet sistem tarafından normal kabul edilecektir.
- iv. İstatistiksel temelli sistemler, operatörlerin paket analizi konusunda oldukça yetenekli olmalarını gerektirir. Ayrıca, SBID sistemleri kural bazlı sistemler gibi sorunu açıkça belirten uyarılar üretmemektedir. Veri analisti, örneğin "EXPLOIT dtspcd istismar girişimi" başlıklı uyarılar almayacaktır.

3.3.2. Bilgi temelli yöntemler

Bilgi temelli teknikler, belirli saldırılardan ve sistem açıklıklarından bilgi çıkarmak için kullanılmaktadırlar. Bu bilgi, ağda veya sistemde meydana gelen saldırıları veya atakları tanımlamak için kullanılabilir. Bilgi tabanlı tekniklerde, ağ veya ana bilgisayar olayları önceden tanımlanmış saldırı kurallarına veya kalıplarına göre kontrol edilir. Amaç, bilinen saldırıları genel bir şekilde temsil etmek, böylece gerçek olayların ele alınmasını kolaylaştırmaktır. Bu teknikler, önceden belirlenmiş saldırı gösterimleriyle eşleşmeye çalışarak bilinen saldırı örneklerini aramaktadır. Arama işlemi, diğer saldırı tespit tekniklerinde olduğu gibi tam bir bilgi eksikliği ile başlar [346]. Bu yöntemlerde de bir saldırı tespit edilir edilmez alarm üretilmektedir. Bunlar hem kötüye kullanım hem de anomali temelli tespit için kullanılabilir [347].

Bilgi tabanlı teknikler genel olarak durum geçiş analizi, uzman sistemler ve imza analizi olarak sınıflandırılmaktadır. Bilgi tabanlı teknikler iyi doğruluk ve çok düşük yanlış alarm oranlarına sahiptir. Toplanan bilgi, güvenlik analistinin önleyici veya düzeltici önlem almasını kolaylaştırmaktadır.

Bilgiye dayalı teknikler, gerçekleştirilen dikkatli ve ayrıntılı analizlere dayanarak her bir saldırının bilgisini korur; bu zaman alıcı bir işittir. Her saldırıyı güncellemek için önceden bilgi sahibi olmak zor bir işittir.

Bilgi tabanlı tekniklerden uzman sistem yaklaşımı en yaygın kullanılan bilgi tabanlı yöntemlerden biridir [348]. Uzman sistem, geleneksel anlamda, ilişkili bir bilgi tabanı olan veya olmayan kural tabanlı bir sistemdir. Uzman bir sistem, kuralları sistemin mevcut durumuna göre eşleştiren bir kural motoruna sahiptir ve eşleştirme sonuçlarına bağlı olarak bir veya daha fazla kural tetiklemektedir.

Bilgi tabanlı diğer bir örnek sistem, durum geçişi analizidir. Durum geçişi analizi, durum geçiş diyagramlarına dayalı bir dizi hedef ve geçiş içeren saldırıları açıklar. Bir saldırı durumunu tetikleyen her olay izinsiz giriş olarak kabul edilmektedir. Trafik verilerini, güvenli durumdan güvenliği ihlal edilmiş bir hedef duruma götüren bir dizi durum değişikliği olarak modeller. Durum geçiş analizi üç ana bileşenden oluşmaktadır. Bunlar; bilgi tabanı, çıkarım motoru ve karar motorudur. Denetim (audit) verisi ön işlemcisi, çıkarım

motoruna girdi olarak gönderilecek ham denetim verilerini yeniden biçimlendirir. Çıkarım motoru, önceden işlenmiş denetim verilerinden çıkarılan durum geçişlerini izler ve daha sonra bu durumları bilgi tabanında bulunan durumlarla karşılaştırır. Karar motoru, durum geçişlerinin eşleşmesi için çıkarım motorunun gelişimini izler. Ayrıca, çıkarım motorunun ve karar tablosunun sonuçlarına dayanarak yapılacak eylemleri de belirtir. Son olarak, karar sonuçları eylem için TOA (Site Güvenlik Görevlisi) arayüzüne gönderilir. Koordineli ve kullanıcı oturumlarına yönelik saldırıları iyi tespit edebilmektedir [348].

İmza analizi yönteminde ise, saldırıların semantik tanımı denetim yolunda kolayca bulunabilen bilgilere dönüştürülmektedir. Bu tür bilgilere örnek olarak, saldırıların ürettiği denetim olaylarının dizilerini veya denetim izinde aranabilecek veri örüntüleri verilebilir [349]. İmza analizi yöntemi uzman sistemlerle aynı bilgi edinme yaklaşımına sahiptir fakat bilgi edinme yöntemleri farklı şekillerdedir. Bu yöntemde atakların semantik tanımları, içinde açık bir şekilde denetim izleri bulunan bilgiye dönüştürülür. Örneğin, saldırı senaryoları, ürettikleri denetim (audit) olaylarının dizilerine veya sistem tarafından oluşturulan denetim izlerinde aranabilecek veri örüntülerine çevrilebilir. Bu yöntem atak tanımlarının semantik seviyelerini azaltmaktadır [350].

Bilgi tabanlı yöntemlerin temel avantajları şunlardır [351]:

- i. Denetimsiz bir ortamda çalışabilirler.
- ii. Verilerin istatistiksel dağılımı hakkında herhangi bir varsayımda bulunmazlar.

Bilgi tabanlı yöntemlerin temel dezavantajları ise şu şekilde sıralanabilir [351]:

- i. Anormalliklerin varlığını, yalnızca verilerde çok fazla sayıda anormallik mevcut olduğunda tespit edebilir.
- ii. Bilgi teorik tekniği kullanılarak test örneği ile anomali puanını ilişkilendirmek zordur.

3.3.3. Veri madenciliği temelli yöntemler

Veri madenciliği teknikleri, istatistiksel ve makine öğrenme yöntemlerini veritabanı yönetimi ile birleştirerek, büyük veri kümelerinden motiflerin çıkarılması için kullanılan bir dizi tekniktir [349]. Veri madenciliğinin doğasında yer alan özetlemeler yardımıyla veri

indirgeme sayesinde gerçek zamanlı saldırı tespiti de yapılması mümkün olabilecektir. Ağ trafiğinden elde edilen normal trafik profilleri veri tabanlarında tutulmaktadır. Yeni girişimlerle normal trafik karşılaştırılarak saldırı tespiti yapılmaktadır [343].

Bilgi tabanlı STS'ler, örüntüleri bilinen saldırıları tespit edebilir, ancak iç taraftan yapılan saldırıları tespit etmek zordur. Bu zorluğu aşabilmenin yöntemlerinden biri veri madenciliği teknikleridir. Bu tekniklerin ana fikri, yararlı örüntüleri ve ayrıca daha önce göz ardı edilen örüntüleri veri kümesinden çıkarmaktır [352].

Veri madenciliği tabanlı teknikler kümeleme, birliktelik kuralı analizi, sınıflandırma, K en yakın komşu ve karar ağacı yöntemleri olarak sınıflandırılmaktadır.

Olayların birlikte gerçekleşme durumlarını çözümleyen veri madenciliği yöntemlerine Birliktelik kuralları denir. Bu yöntemler, birlikte olma kurallarını belirli olasılıklarla ortaya koymaktadır. Birliktelik kuralları analizi insanların veri madenciliğini anlamak için kafalarında tasarladıkları veri madenciliğine en yakın olanıdır. Bu yöntem bilim veritabanındaki tüm ilginç örüntüleri bulmaktadır. Her taşın altına bakılmaktadır. Bu aynı zamanda modelin zayıflığıdır. Kullanıcı ortaya çıkan bilginin büyüklüğü karşısında zorlanabilir ve bu miktardaki bilginin kullanılabilirliğinin analizi zor ve zaman alıcıdır. Veri madenciliği sonucu ortaya çıkarılan ilişkiler birliktelik kuralları olarak gösterilebilir. Birliktelik kuralı, elde edilen geçmiş verilerin analiz edilmesiyle verilerdeki nenelerin birliktelik davranışlarının incelenmesi ve yeni veriler üzerinde çalışmalar yapılmasını destekleyen bir yaklaşımdır.

Sınıflandırma, yeni bir nesnenin niteliklerini inceleme ve bu nesneyi önceden tanımlanmış bir sınıfa atamaktır. Burada önemli olan, her bir sınıfın özelliklerinin önceden net bir şekilde belirlenmiş olmasıdır. Karar ağaçları veya sınıflandırma ağaçları, istatistiksel öğrenme ve veri madenciliğinde çok bilinen sınıflandırma metodudur. Veri madenciliğinde çok kullanıldığından dolayı, Veri madenciliğinin yük beygiri olarak ta adlandırılmıştır. Sınıflandırma problemlerinde en çok kullanılan algoritmalarından biri karar ağaçlarıdır. Diğer sınıflandırma algoritmalarıyla kıyaslandığında karar ağaçlarının yapılandırılması ve anlaşılması daha kolaydır. Karar ağaçları kullanılarak sınıflandırma 2 aşamada gerçekleştirilir. İlk adımda ağaç oluşturulur. İkinci adımda ise veriler tek tek ağaca uygulanarak sınıflandırma gerçekleştirilir. Karar ağaçları oluşturulurken kullanılan

algoritmanın ne olduğu önemlidir. Kullanılan algoritmaya göre ağacın şekli değişebilir. Değişik ağaç yapıları da farklı sınıflandırma sonuçları verebilir. Karar ağaçlarına dayalı olarak geliştirilen birçok algoritma vardır. Bu algoritmalar birbirlerinden kök, düğüm ve dallanma kriteri seçimlerinde izledikleri yol açısından ayrılırlar. Literatürde en yaygın olarak bilinen algoritmalar ID3, C4.5 ve C5'dir. Karar ağaçlarının eğitimi ve test edilmesi çok hızlıdır, sonuçları daha rahat görselleştirilebilir, sonuçları kural çıkarımında kullanılabilir.

Veri madenciliği tabanlı tekniklerin temel avantajları şunlardır:

- i. Yüksek boyutlu verileri işleyebilirler.
- ii. Önceden hesaplanmış modeller eğitim aşamasında tasarlandığından, test aşamasındaki her bir örneğin karşılaştırılması daha hızlı bir şekilde yapılabilmektedir.
- iii. Örüntüler, denetimsiz modda üretebilirler.

Veri madenciliği tabanlı tekniklerin temel dezavantajları şunlardır:

- i. Bu yöntemler, anormallikleri kümelemenin bir yan ürünü olarak tanımlar ve anomali tespiti için optimize edilmezler.
- ii. Yüksek depolama alanı gerektirirler ve yüksek boyutları nedeniyle sınıflandırmada yavaşlırlar.

3.3.4. Makine öğrenmesi temelli yöntemler

Makine öğrenimi, bir programın kapasitesi ya da belirli bir görev ya da görev grubundaki zaman içindeki performanslarını öğrenmek ve geliştirmek için bir çerçeve olarak tanımlanabilir [353]. Makine öğrenmesi, veriden karmaşık örüntünün tespit edilmesi ve akılcı karar verme için istatistik ve bilgisayarın hesaplama gücünden faydalanmaktadır [354]. Makine öğrenim stratejileri, önceki sonuçlara göre gelişen bir çerçeve oluşturmaya vurgu yapmaktadır, yani son zamanlarda elde edilen verilere göre işletim stratejilerini değiştirebilirler. Anomali tespitinde, genellikle normal davranışlara karşılık gelen etiketli veriler kullanılabilirken, anormal davranışlar için etiketli veriler mevcut değildir. Denetimli makine öğrenmesi yöntemleri saldırısız eğitim verilerine ihtiyaç duymaktadır. Ancak, bu tür eğitim verilerinin gerçek dünyadaki ağ ortamlarında elde edilmesi zordur. Bu eğitim verisi eksikliği, makine öğreniminde iyi bilinen dengesiz veri dağıtımına yol açmaktadır.

Denetimsiz anormallik tespiti, denetimli anormallik tespitinin dezavantajlarının üstesinden gelebilmektedir [355].

Makine öğrenmesine dayalı teknikler genel olarak Bayesci yaklaşımlar, destek vektör makineleri, yapay sinir ağları, bulanık mantık ve genetik algoritmalar olarak sınıflandırılmaktadır.

Bayesci yaklaşımlar, çok sınıflı ağ ortamlarında anormallik tespiti için kullanılmaktadır. Bir sınıfa verilen test örneğinin kullanımından önce eğitim veri kümesinden tahmin edilir. Bayes teoreminin basitleştirilmiş halini kullanan algoritmalar. Algoritma hesaplamada koşullu olasılık kullanmaktadır. Eğitim veri seti üzerinde sistem eğitilmekte ve test veri setinin hangi sınıfa dahil olduğu tespit edilmeye çalışılmaktadır. Bayes yaklaşımı basit bir yapıya sahip olmasına rağmen, oldukça başarılı sonuçlar üretmektedir [356]. Bayes ağını kullanan tek değişkenli bir kategorik veri seti için temel bir teknik, bir test verisi örneği verildiğinde, bir sınıf etiketini (normal sınıf etiketlerinden ve anomali sınıf etiketinden) gözlemlemenin sonraki (posterior) olasılığını tahmin eder. En büyük sonraki (posterior) sınıf etiketi, verilen test örneği için tahmin edilen sınıf olarak seçilir. Bir sınıfa verilen test örneğini ve önceki sınıf olasılıklarını gözlemleme olasılığı, eğitim veri kümesinden tahmin edilir. Sıfır olasılıkları, özellikle anomali sınıfı için, Laplace yumuşatma kullanılarak yumuşatılır. Temel teknik, her test örneği için öznitelik başına posterior olasılıkları toplanarak ve test örneğine bir sınıf etiketi atamak için toplanan değeri kullanarak çok değişkenli kategorik veri kümesini genelleştirmek için genelleştirilebilir [357].

Destek Vektör Makineleri (DVM), yüksek performans gösteren ve oldukça basit bir fikir ile oluşturulmuş bir öğrenme algoritmasıdır. Temelini İstatiksel Öğrenme teorilerinden alan, DVM teori olarak iki sınıfa ait örnekleri doğrusal olarak en iyi ayıran destek vektörlerine dayanmaktadır. DVM; doğru, düzlem ya da hiper düzlem yardımı ile verilerin sınıflara ayrıldığı bir makine öğrenmesi yöntemidir. Tek sınıflı ortamlarda da anormallik tespiti için DVM kullanılmaktadır. Doğrusal olarak ayrışabilen veriler için sıkça kullanılan bu yöntem, çekirdek fonksiyonları yardımı ile verileri doğrusal olarak ayrıştırılabilir duruma getirebildiği için doğrusal olarak ayrışamayan veriler içinde kullanılabilir. Bu yöntemdeki asıl amaç, hataların karesini en aza indirecek ayırıcı belirlemektir [344].

Yapay sinir ağı (YSA), insan beyninden esinlenerek geliştirilmiş, ağırlıklı bağlantılar aracılığıyla birbirine bağlanan ve her biri kendi belleğine sahip işlem elemanlarından oluşan paralel ve dağıtılmış bilgi işleme elemanlarıdır, bir başka deyişle, biyolojik sinir ağını taklit eden bilgisayar programlarıdır [358]. En temel yapı taşı olarak nöronları kullanırlar. Yapay bir nöronun girişler, toplama birimi ve transfer fonksiyonu olmak üzere üç özelliği vardır. Yapay nöronlar biyolojik nöronlara benzer olması için tasarlanmıştır. Biyolojik sistemlerde öğrenme, nöronlar arasındaki sinaptik (synaptic) bağlantıların ayarlanması ile olur. Yani, insanlar doğumlarından itibaren bir yaşayarak öğrenme süreci içerisine girerler. Bu süreç içinde beyin sürekli bir gelişme göstermektedir. Yaşayıp tecrübe ettikçe yani öğrendikçe sinaptik bağlantılar ayarlanır ve hatta yeni bağlantılar oluşur veya ağırlıklar düzenlenir veya değiştirilir. Bu sayede öğrenme gerçekleşir. Bu durum YSA için de geçerlidir. YSA’larda öğrenme, eğitme yoluyla örnekler kullanarak olur; başka bir deyişle, gerçekleşme girdi/çıkıtlı verilerinin işlenmesiyle, yani eğitme algoritmasının bu verileri kullanarak bağlantı ağırlıklarını (weights of the synapses) bir yakınsama sağlanana kadar, tekrar tekrar ayarlamasıyla olur [359]. YSA’lar tek sınıflı ortamların yanı sıra çok sınıflı ortamlarda da kullanılmaktadır. Sinir ağı kullanan temel bir çok sınıflı anormali algılama tekniği iki adımda çalışmaktadır. İlk olarak, farklı normal sınıfları öğrenmek için normal eğitim verileri üzerinde bir sinir ağı eğitilir. İkinci olarak, her test örneği sinir ağına bir girdi olarak sağlanır. Ağ test girişini kabul ederse, normaldir ve ağ bir test girişini reddederse, bir anormallik olarak algılanmaktadır [357].

ML temelli yöntemlerde tercih edilen sınıflandırıcı, geliştirilen STS’nin başarısında çok önemli rol oynamaktadır. Yapay sinir ağı en sık kullanılan ve aynı zamanda en yüksek başarı oranını elde eden sınıflandırıcılardır. Ancak Destek vektör makineleri de bazı saldırı tiplerinde etkin çözümler üretebilmektedir. Sistem eğitiminde ve test aşamasında kullanılan veri kümesi önemlidir. Oluşturulan STS’nin başarısını ölçmek için yeteri kadar test verisine ihtiyaç bulunmaktadır. Yeterli miktarda test verisi, sistem başarısının doğru ölçülmesinde önemli rol oynamaktadır [354]. Temel avantajları esneklik, uyarlanabilirlik ve karşılıklı bağımlılıkların yakalanmasıdır. Dezavantajı ise, yüksek algoritmik karmaşıklık ve uzun eğitim süreleri olarak belirtilebilir.

3.3.5. Kullanıcı davranışı belirleme

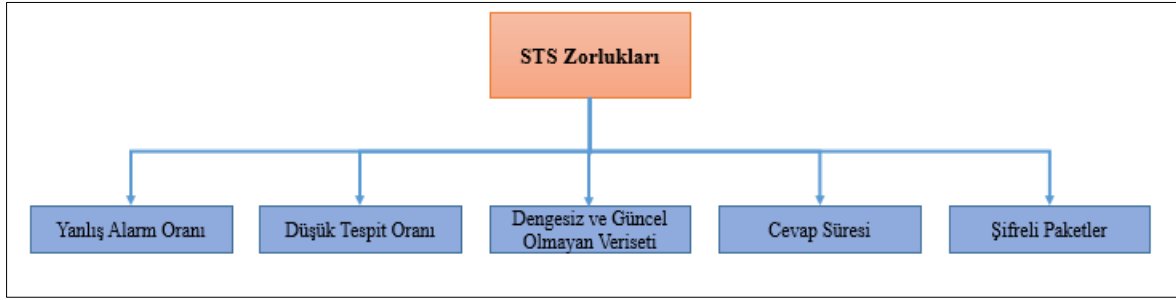
Bu yöntemde, saldırı tespit sistemleri, anormal etkinlikleri normal etkinliklerden ayırt etmek için kullanıcıyı veya sistem kullanımını kategorize eden özelliklere dayanarak oluşturulabilir. Anomali tespitindeki temel nokta, izlenen sistem günlüğü veya hesap günlüğü verilerinden profil oluşturma sistemi veya kullanıcı davranışı üzerinedir. Log verileri veya sistem günlüğü, UNIX kabuk komutlarını, sistem çağrılarını, tuş vuruşlarını, denetim olaylarını ve kullanılan ağ paketlerini içerebilmektedir [341].

3.3.6. Bilgisayar immünolojisi

Bağışıklık sistemi, insan vücudunu koruyan karmaşık bir sistemdir. İmmünoloji araştırmaları, insan vücudunun savunmasının altında yatan mekanizmaları anlayarak immünolojik hastalıklar için ilaç geliştirmek ve sağlığı korumak için oldukça önemlidir. Bu araştırmalardan yola çıkılarak insanlardaki bağışıklık sistemine benzer bir önleme sisteminin bilgi sistemlerinde de uygulanabileceği fikri ortaya çıkmıştır. Bu yöntem çok saldırı tespit sistemlerinde çok fazla kullanılmamakla birlikte bir yöntem türü olarak burada sınıflandırmaya dâhil edilmiştir.

3.4. Saldırı Tespit Sistemlerinde Sorun ve Zorluklar

İlk STS'nin önerilmesinden beri araştırmacılar tarafından birçok farklı STS geliştirilmiş ve uygulanmıştır. Ayrıca bu amaç için sürekli olarak daha doğru, daha hızlı ve ölçeklenebilir yöntemler araştırılmıştır. IoT ve büyük veri çağına gelmesi ve her geçen gün gelişmesiyle birlikte saldırı tespit sistemlerinin konusu ve sayısı da artmaktadır. Araştırma toplulukları tarafından birçok yöntem ve sistem geliştirilmiş olmasına rağmen, davranış tabanlı saldırı tespit sistemleri için hala bir dizi açık araştırma konusu ve zorluğu bulunmaktadır. Şekil 3.10, STS'lerdeki sorun ve zorlukları özetlemektedir.



Şekil 3.10. STS’lerin zorlukları

STS’lerde karşılaşılabilecek sorun ve zorluklar STS amaç ve yöntemlerine göre değişmekle birlikte bir STS’nin tasarımında genel olarak karşılaşılabilecek bazı temel zorluklar bulunmaktadır. Bunlardan bazıları şunlardır:

- i. Ağ anomalisine dayalı bir STS, yanlış alarm oranını azaltmalıdır. Ancak, yanlış alarmı tamamen azaltmak mümkün değildir. Ortamdan bağımsız bir saldırı tespit sistemi geliştirmek, ağ anomalisine dayalı saldırı tespit sistemi geliştirme topluluğu için bir başka zorluktur [360-364].
- ii. Saldırı tespit sistemini değerlendirmek için kullanılabilecek genel bir metodoloji veya bir dizi parametre geliştirmek bir diğer zorlu görevdir [363,364].
- iii. Saldırı tespit sistemlerinde yeni örüntüler tanımlandığında, performanstan ödün vermeden veri tabanının güncellenmesi bir başka zorlu görevdir [360,364].
- iv. Ele alınması gereken diğer bir görev, eğitim aşamasında ve ayrıca dağıtım aşamasında veri ön işlemenin hesaplama karmaşıklıklarını azaltmaktır [360,361].
- v. Her saldırı kategorisi için öznetelikleri seçmek için uygun bir yöntem geliştirmek bir diğer önemli görevdir [360-362].
- vi. Anomali tespiti için etkili bir topluluk yaklaşımı oluşturmak üzere ilişkilendirilmemiş ve tarafsız bir grup sınıflandırıcıdan en iyi sınıflandırıcıyı belirlemek başka bir zorluktur [360-362].
- vii. Gürültü, bir saldırı tespit sisteminin etkinliğini ciddi bir şekilde sınırlandırabilir. Yazılım hatalarından, bozuk DNS verilerinden ve kaçmış yerel paketlerden üretilen kötü paketler, hatalı alarm oranı önemli ölçüde yüksek olabilmektedir [365].
- viii. Şifreli paketler, saldırı tespit sistemlerinin çoğu tarafından işlenemez. Bu nedenle, şifreli paket daha önemli ağ saldırıları oluşana kadar keşfedilmemiş bir ağa izin verebilir [365].

3.5. Saldırı Tespit Sistemi Veri Kümeleri

Siber güvenlik alanında, özellikle STS'lerde, veri kümelerinin akıllıca seçilmesi ve kullanılması, çeşitli makine öğrenimi ve istatistiksel modellerin geliştirilmesinde, eğitilmesinde ve değerlendirilmesinde önemli bir rol oynar [366]. Veri kümeleri, yöntemin davranışları tespit etme yeteneğini değerlendirmemize olanak sağlayarak herhangi bir STS yaklaşımını doğrulamak için gereklidir. Benzer şekilde, verilerin kalitesi ve kapasitesi, AYZ modellerinin kararlarını önemli ölçüde etkiler. Wireshark veya WinDump gibi çok sayıda yazılım aracı siber güvenlik verileri oluşturmak için kullanılsa da, bu yöntemler genellikle dar veya düşük hacimli verileri toplamak için daha uygundur. Ancak, saldırı tespit sistemleri için gereken veri kümeleri genellikle yüksek edinim süresi ve maliyetleri gerektirir [367]. Bu nedenle, literatürden kamuya açık siber güvenlik veri kümelerini kullanmak, veri toplamaya harcanan süreyi azaltabilir ve çalışma verimliliğini artırabilir. Araştırmacılar, bu veri kümelerini kullanarak AYZ tabanlı siber güvenlik çözümlerini eğitebilir, doğrulayabilir ve değerlendirebilirler. STS araştırmalarında kullanılan önemli veri kümelerinden bazılarının ayrıntılı açıklamaları Çizelge 3.1'de sunulmaktadır. Her veri kümesi kaynağı, boyutu ve bileşimi açısından analiz edilmektedir.

Farklı saldırı tespit veri kümelerini karşılaştırmak ve araştırmacıların kendi özel senaryoları için uygun veri kümelerini bulmalarına yardımcı olmak amacıyla, yıl, saldırı türleri, normal ve saldırı trafiği, örnek sayısı ve özellik sayısı gibi özellikler değerlendirmenin temeli olarak kullanılmıştır. Her gün yeni saldırı senaryoları ortaya çıktığından, bir saldırı tespit veri kümesinin güncelliği, kullanılabilirliğinde önemli bir rol oynamaktadır. Veri kümesinin güncelliğini belirtmek için "yıl" özelliği kullanılmıştır. Normal trafik verilerinin varlığı, "evet" veya "hayır" değerleriyle başka bir karşılaştırma parametresi olarak kullanılmıştır. Normal kullanıcı davranışının dâhil edilmesi, bir STS'yi değerlendirmek için temel parametrelerden biridir. STS veri kümeleri, veri kümesindeki kötü amaçlı ağ trafiğinin türünü gösteren çeşitli saldırı senaryoları veya türleri içermelidir. "Etiketli veriler" parametresi, veri kümelerinin etiketli olup olmadığını gösterir. Etiketli veri kümeleri, gözetimli makine öğrenimi yöntemlerini eğitmek ve saldırı tespit tekniklerini değerlendirmek için gereklidir. Bu parametrelerin yanı sıra veri setlerinde yer alan örnek sayısı ve özellik sayısı da araştırmacıların seçiminde yol gösterici olarak kullanılmıştır.

Çizelge 3.1. STS veri kümeleri

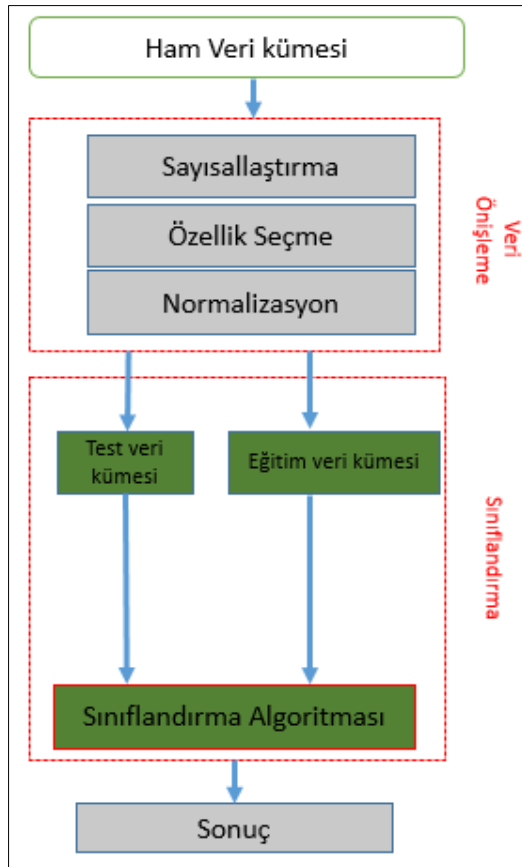
Veri kümesi	Referans	Yıl	Örnek sayısı	Özellik sayısı	Etiketli Data	Normal Trafik	Saldırı Trafik	Saldırı Türleri
NSL-KDD	[368]	2009	150K	41	√	√	√	DoS, R2L, U2R, Prob
UNSW-NB15	[369]	2015	2M	49	√	√	√	DoS, solucan, arka kapı, ve fuzzers
CIC-IDS2017	[370]	2017	3.1M	80	√	√	√	Kaba kuvvet, Portscan, Botnet, DoS, DDoS, web, sızma(Infiltration)
ADFA-LD	[371]	2013	2.7M	26	√	√	√	Sıfıncı-gün saldırıları, Stealth saldırısı, C100 Webshell saldırısı
ToN_IoT	[372]	2020	22.3M	132	√	√	√	XSS, DDoS, DoS, parola kırma saldırıları, keşif veya doğrulama, MITM, fidye yazılım, arka kapı ve enjeksiyon saldırıları
CIC-IoT-Dataset-2022	[373]	2022	1M	48	√	√	√	Flood ve RTSP- kaba kuvvet
CSE-CIC-IDS2018	[374]	2018	8.3M	80	√	√	√	Kaba kuvvet, Portscan, Botnet, Dos, DDoS, web, sızma(Infiltration)
CPS Dataset	[375]	2021	24.3M	14	√	√	√	DoS, tarama ve MITM saldırıları
CICDDoS2019	[376]	2019	50M	80	√	√	√	12 farklı DDoS saldırısı
IoT-23	[377]	2020	325M	19	√	√	√	20 farklı zararlı yazılım saldırısı
MalDroid20	[378]		11K	470	√	√	√	Reklam zararlı yazılımı, bankacılık ve SMS zararlı yazılımı, riskware
KDD99	[379]	1999	5M	41	√	√	√	DoS, R2L, U2R, Prob
BoT-IoT	[380]	2019	1.5M	34	√	√	√	DoS, DDoS, Port scanning, Brute force, Information theft, CC
UGR'16	[381]	2016	16M	12	√	√	√	DoS, Scan, Botnet, SPAM
Kyoto	[382]	2006	93M	24	√	√	√	Shellcode ve zararlı yazılım
ISCX-URL2016	[383]	2016	114K	17	√	√	√	Spam, zararlı yazılım, oltalama ve tahrif(defacement)
ISCXIDS 2012	[384]	2012	2M	20	√	√	√	Kaba kuvvet, sızma(infiltrating), HttpDos, DDoS
ISCX VPN-nonVPN	[385]	2016	45K	63	√	√	√	Chat, VoIP, streaming servis grupları
AWID	[386]	2016	37M	15	√	√	√	Kimlik doğrulama isteği, ARP seli(flooding), enjeksiyon, prob isteği
CIDDS-001	[387]	2017	32M	81	√	√	√	DoS, port tarama (ping-tarama, SYN-tarama), SSH kaba kuvvet
Botnet	[388]	2010	19K	25	√	√	√	Menti, Murlo, Neris, NSIS, Rbot, Sogou, Strom, Virut, Zeus saldırıları
CDX	[389]	2009	500K	75	√	√	√	Bellek taşması

3.6. Zeki Saldırı Tespit Sistemleri

Saldırı tespit sistemleri saldırıların dinamik yapısına bağlı olarak tarihsel süreçte sürekli olarak gelişim göstermiştir. Güvenlik ihtiyaçlarının dinamik ve sürekli hale gelmesi yapay sinir ağları, yapay bağışıklık sistemleri, bulanık mantık ve derin öğrenme yaklaşımları gibi YZ tekniklerini kullanan zeki saldırı tespit sistemlerine olan ihtiyacı ortaya çıkarmıştır. STS'lerin geliştirilmesinde YZ teknikleri kullanılarak STS performanslarının iyileştirilmesi hedeflenmiş ve zeki yöntemlerin kullanılması ile saldırı tespit sistemlerinin başarı oranlarının klasik yöntemlere göre arttığı gözlenmiştir. Özellikle, sıfıncı gün saldırıları olarak adlandırılan yeni saldırıların tespit edilebilmesi için kullanılan anormallik tespiti yaklaşımlarında zeki tekniklerin kullanılması başarı oranlarını önemli ölçüde artırmaktadır. Bu bölümde zeki bir STS gerçekleştirilmesi için gereken temel adımlar açıklanmıştır. Zeki

STS sistemi, genel hatları ile Şekil 3.11’de gösterildiği gibi veri ön işleme, sınıflandırma ve sonuçların değerlendirilmesi adımlarından oluşmaktadır.

Zeki STS’lerin bir bilgi sisteminde saldırgan davranışları yakalayabilmesi için sistemin normal davranış örüntülerini öğrenmesi gerekmektedir. Bu amaç doğrultusunda, zeki STS’ler bilgi sistemlerine ait çeşitli veri kümeleri kullanılarak eğitilmektedir. Zeki STS tasarımı saldırganların başarılı bir şekilde tespit edilebilmesi için eğitim veri kümeleri önemli bir yer tutmaktadır. Eğitim veri kümelerinin miktarı, güncel saldırıları içermesi, doğru etiketlenmesi ve dengeli olmaları zeki STS tasarımı başarıyı etkileyen faktörlerdir.



Şekil 3.11. STS blok şeması

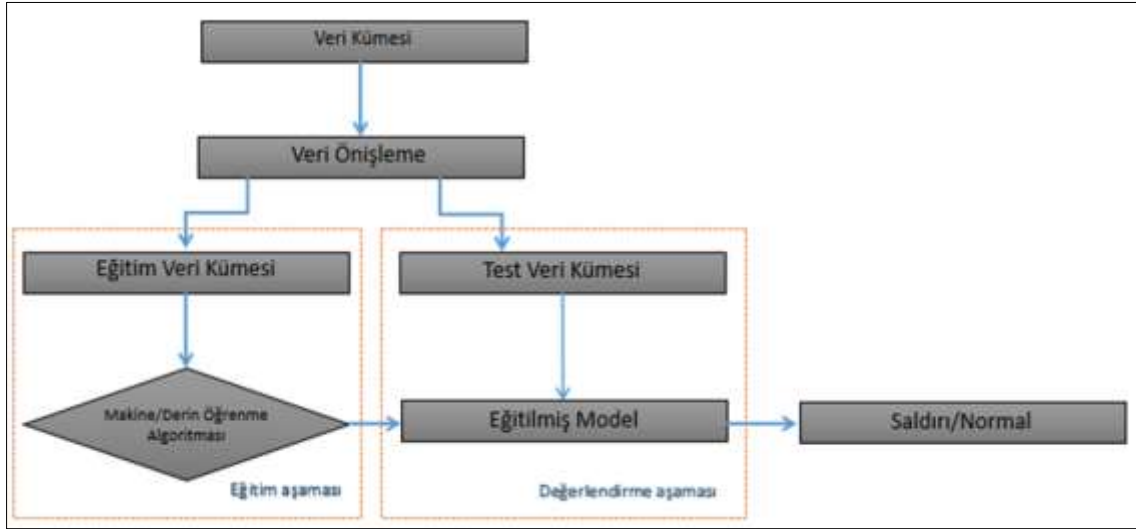
Veri ön işleme, ham verileri sınıflandırma aşamasına yönelik kullanışlı ve verimli bir biçime dönüştürmek için kullanılan tekniklerin bütünüdür. Bu sebepten veri ön işleme, zeki saldırı tespit sistemi sürecinde önemli bir adımdır. Veri ön işleme ham verileri daha sonraki işlemler için hazırlayan bir hazırlık aşaması olarak değerlendirilebilir. Gerçek dünya verileri genellikle eksik, düzensiz ve birçok hata içermektedir. Veri ön işleme, bu tür

sorunları çözmek için kullanılan yöntemler bütünüdür. Veri ön işleme sayesinde kullanılacak YZ algoritmasının performansı artırılmaktadır.

Zeki STS tasarımı verinin kümesi üzerinde YZ teknikleri uygulanmadan önce verinin kümesinde sayısal olmayan değerler sayısal değerlere dönüştürülerek YZ algoritmalarına hazır hale getirilmeli ve normalizasyon işlemi gerçekleştirilmelidir. Normalizasyon işlemi sonucunda verinin kümesi içinde bulunan tüm sayısal değerler belirli aralıktaki değerlere dönüştürülecektir. Bu sayede yüksek sayısal değerlere sahip bazı özelliklerin algoritma sonucunu etkileyerek performansı olumsuz etkilemesinin önüne geçilmiş olacaktır. Bir diğer önemli adım olarak, tanımlayıcı özellikler olan ve sınıflandırma algoritmalarının sonucunu etkilemeyen özellikler özellik seçme yöntemleri ile verinin kümesinden çıkarılmalıdır.

Zeki STS'lerin ikinci temel adımı olan sınıflandırma aşamasında sisteme adını veren zeki yaklaşımlar ile saldırı ve normal trafikler sınıflandırılmaktadır. Sınıflandırma algoritması olarak birçok farklı algoritma kullanılmaktadır. Kullanılan algoritmalar amaca ve uygulanacak verinin kümesinin özelliklerine göre değişiklik göstermektedir. STS'lerde kullanılacak DL ve ML yaklaşımları Şekil 3.12'de gösterildiği gibi öğrenme ve tahmin etme adımları olmak üzere iki adımdan oluşmaktadır. Zeki STS'de kullanılan sınıflandırıcının eğitim aşamasında, her sınıftaki nesnelerin özellikleri çıkarılır, böylece sınıflandırıcı bazında bu ilişkiler üzerine bir model oluşturulur. Bu modeller daha sonra yeni verinin nesnelerinin sınıflarını tahmin etmek için kullanılmaktadır. Bu kararlar, verinin içinde olacağı tahmin edilen kategorinin özelliklerine dayanmaktadır.

Zeki STS'lerin sınıflandırma sonuçlarının değerlendirilmesinde doğruluk, hassasiyet, F-skoru, recall, ROC ve Precision-Recall eğrileri gibi parametreler literatürde yaygın olarak kullanılmaktadır. Bu parametreler hata matrisi verilerinden elde edilmektedir. Karmaşıklık matrisinin temel elemanları doğru-pozitif (TP), doğru-negatif (TN), yanlış-pozitif (FP) ve yanlış-negatif (FN)'tir.



Şekil 3.12. Sınıflandırma akış şeması

Zeki STS’lerde kullanılan YZ algoritmalarına göre elde edilen başarı oranı değişebilmektedir. Başarılı bir zeki STS tasarımı için araştırmacıların YZ algoritmalarına da hâkim olması gerekmektedir.

Zeki STS’ler genel olarak değerlendirildiğinde;

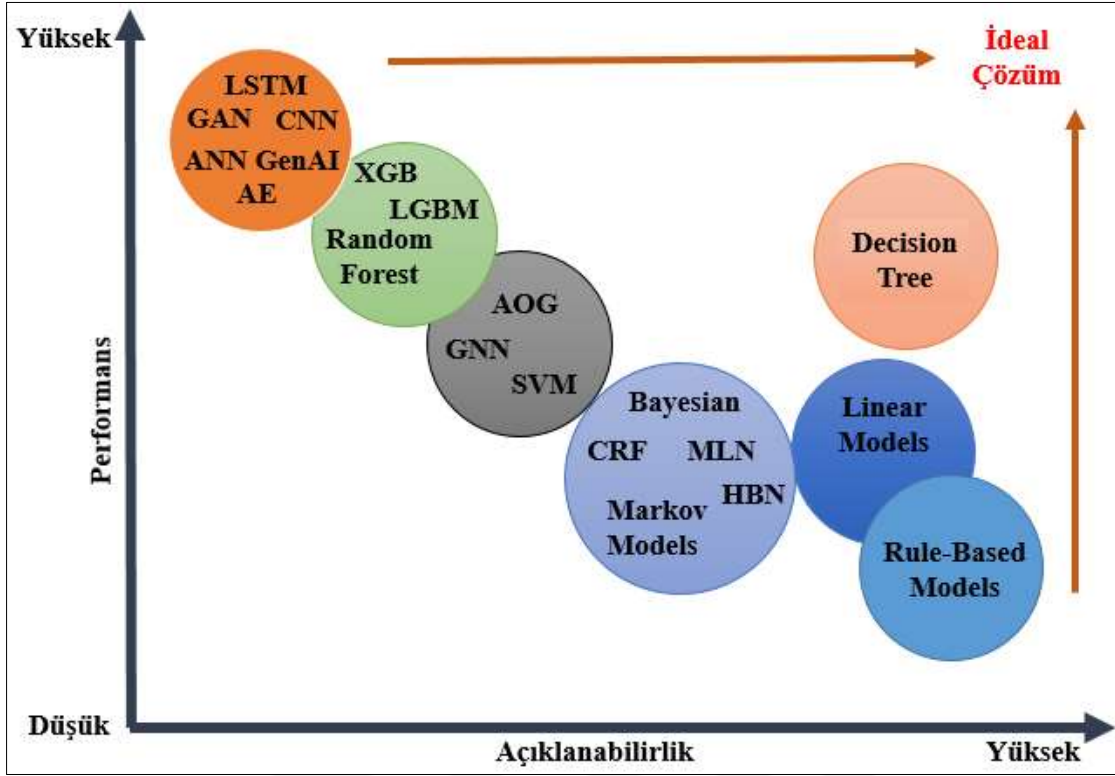
- her geçen gün büyüyen ve gelişen bilgi ve bilgisayar sistemlerine paralel olarak siber saldırılar da daha karmaşık hale gelen bir dünyada, ekonomik, siyasi ve politik gibi çeşitli motivasyonlarla hareket eden saldırganların güvenlik önlemlerini aşabilmek için kapsamlı ve karmaşık yöntemler kullanmakta, saldırganlar tarafından kullanılan karmaşık yöntemlere karşı imza tabanlı olan çözümler başarısız olduğu,
- DL mimarilerindeki gelişmeler, yeni ve dağıtık öğrenme algoritmalarının performansının yükselmesi, federe öğrenme yaklaşımlarıyla verilere erişmeden eğitilmiş modellere erişme, öğrenilmiş yapılarıdaki birikimin yeni modellere transfer edilebilmesi gibi yeni yaklaşımların kullanılmasıyla STS’lerin performansının hızla artacağı,
- Zeki STS uygulamalarının ve ürünlerinin son yıllarda artış göstermesinin arkasında her ne kadar YZ yaklaşımları, mimarileri ve algoritmalarında artış var ise de erişilen verilerin hem boyut hem de çeşitliliğinin artması, bu veri kümelerine erişimin kolaylaşması, etiketli verilerin çoğalması, açık veri yaklaşımlarının gelişmesi, güncel saldırıları ve ağ verilerini içeren veri kümelerinde artış ve özellikle de veri kümelerindeki özniteliklerin sayısının da çoğalmasının etkili olduğu,

- Siber güvenlik alanında etkili bir çözüm olduğu ve gelecek saldırıların hem tespitinde hem de durdurulmasında çok etkili bir çözüm olacağı,
- ML, DL ve YZ algoritmalarının hızla gelişimine paralel olarak zeki STS'lerin geliştirilmesi de artık siber güvenlik ve savunmada etkili çözümler içerisinde oldukları,
- Öğrenme mimarileri ve algoritmalarındaki her türlü iyileştirme ve geliştirmenin, sadece STS'ler değil diğer çözümlerinde zekileştirilmesine büyük katkılar sağlayacağı,
- Zeki algoritmaların yanı sıra büyük veri teknolojilerinin gelişimi ile birlikte bilgi sistemlerine ait büyük miktardaki veriler işlenerek saldırı tespit sistemlerinin daha kullanışlı, etkili ve başarılı çözümler haline geleceği, tüm bu gelişmelere paralel olarak STS'lerin işlevselliği, önemi ve siber güvenlik sektöründeki payının da giderek artacağı, daha kapsamlı STS'lerin geliştirilmesinin kolaylaşacağı,
- Zeki STS'ler etkili bir çözüm olarak karşımıza çıkmakta, YZ ve makine öğrenmesi algoritmalarını kullanan zeki saldırı tespit sistemleri bilgi sistemlerinde normal ve anormal davranışları tanımlayarak güvenlik seviyesinin artırılmasında önemli katkılar sunmaktadırlar.

4. AÇIKLANABİLİR YAPAY ZEKÂ

Son yıllarda, YZ ve ML uygulamaları çeşitli araştırmalarda giderek daha popüler hale gelmiştir ve siber saldırı tespiti, görüntü sınıflandırma, video tavsiyesi, sosyal ağ analizi, metin madenciliği ve doğal dil işleme gibi birçok alanda kullanılmaktadır. Çeşitli YZ algoritmaları arasında, temsil öğrenme [390] olarak da bilinen "derin öğrenme" algoritmaları bu uygulamalarda yaygın olarak kullanılmaktadır. Verilerin giderek büyümesi ve kullanılabilirliği ve donanım teknolojilerindeki önemli gelişmeler derin öğrenme alanında yeni araştırmaların ortaya çıkmasına yol açmıştır. Köklerini geleneksel sinir ağlarından alan derin öğrenme yaklaşımları, önceki YZ algoritmalarından önemli ölçüde daha iyi performans göstermektedir. En yeni derin öğrenme tekniklerinin çoğu doğal dil işleme, görsel veri işleme, konuşma ve ses işleme, siber güvenlik ve diğer birçok uygulama türlerinde umut verici sonuçlar göstermiştir [391,392].

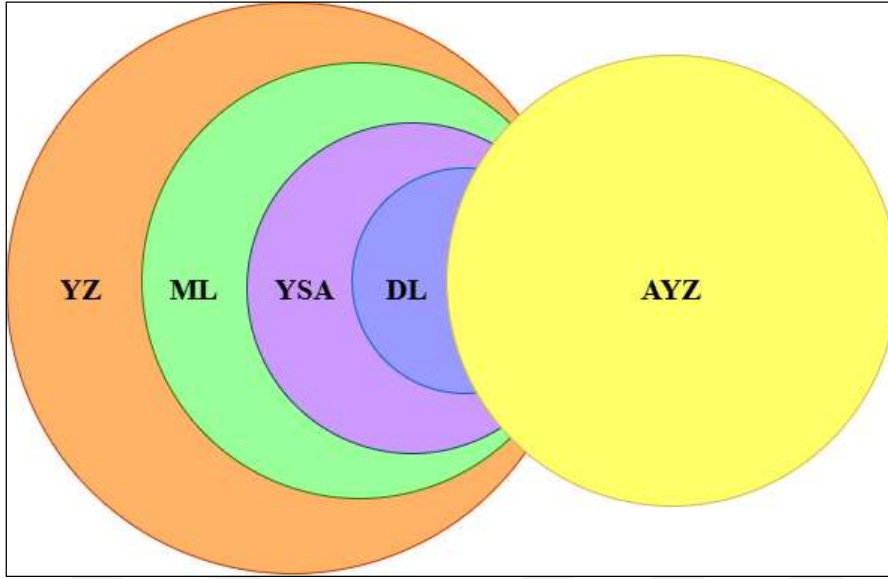
Açıklanabilir bir YZ, işleyişini açık veya kolay anlaşılır hale getirmek için ayrıntılar, sonuçlar veya nedenler üreten zekâdır. Açıklanabilir YZ, YZ ile gerçek zekâ arasında bir köprü işlevi görmektedir. AYZ alanı, günümüzde ML ve DL algoritmalarının en büyük eksiklikleri olan modellerin yorumlanabilirliği ve açıklanabilirliği problemini ele almaktadır. Şekil 4.1’de gösterildiği gibi algoritmalar daha güçlü hale geldikçe ve daha iyi bir doğrulukla daha iyi tahminde bulundukça, bir tahminin nasıl ve neden yapıldığını anlamak giderek daha önemli hale gelmektedir. ML ve DL algoritmalarının gelişimi çeşitli uygulamaların tahmin ve doğruluk performanslarının artırılmasında önemli ilerlemelere yol açmakla birlikte giderek daha karmaşık yapıları da beraberinde getirmiştir. Bu durumun sonucu olarak algoritmaların gelişmişliği ve şeffaflığı arasındaki bir ödünleşim ortaya çıkmaktadır. Yorumlanabilirlik ve açıklanabilirlik olmadan, YZ’nin gerçek hayattaki uygulamalarının tahminlerine güvenilmesi pek mümkün görünmemektedir. İnsan tarafından anlaşılabilir açıklamalar, sistem güvenliğini artırmanın yanı sıra ML sistemlerinin güvenliğini ve sürekli benimsenmesini teşvik etmektedir. Gelişmekte olan bir alan olarak, AYZ’nin önümüzdeki yıllarda araştırmacılar ve uygulayıcılar için hayati önem taşıyacağı değerlendirilmektedir. Bu bölüm, açıklanabilir YZ’nin temellerini, terminolojisini ve sınıflandırmasını kapsamlı şekilde sunmaya yönelik bir yaklaşım benimsemektedir.



Şekil 4.1. Açıklanabilirlik ile performans arasındaki ilişki

4.1. YZ ve Kara Kutu Problemi

YZ, 1950'lerde Alan Turing'in [393] "Makinalar düşünebilir mi?" sorusuyla başlayarak günümüzde bilgisayar sistemlerinin insan gibi düşünebilmesi, öğrenebilmesi, karar verebilmesi, problemleri çözebilmesi ve insanlara özgü olan yetenekleri taklit edebilmesine yönelik kapsamlı çalışmalar yürüten bir bilim ve mühendislik alanı haline gelmiştir. Bu alanda yürütülen çalışmalar, bilgisayar bilimi, psikoloji, felsefe, nöro-bilim ve dil bilimi gibi birçok farklı disiplinden beslenmektedir [394]. YZ, algoritmalar ve yazılımlar geliştirerek bilgisayarların insan zekâsının bazı yönlerini taklit etmesini sağlamayı amaçlamaktadır. YZ, Şekil 4.2'de gösterildiği gibi ML ve DL gibi çalışma alanlarını da kapsayan geniş bir çalışma alanıdır. ML, çeşitli algoritmaların veri setlerinden öğrenmelerini sağlayarak belirli görevler için insan benzeri performans göstermelerini sağlamaya çalışmaktadır. DL ise, insan beynindeki nöronlar arası etkileşimleri modelleyen yapay sinir ağları temelinde yükselmekte ve genellikle görüntü, dil ve ses işleme gibi karmaşık görevlerde sıklıkla tercih edilmektedir.



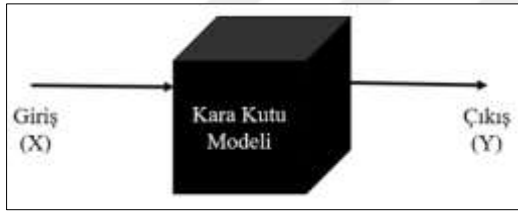
Şekil 4.2. AYZ modellerinin YZ alanı ile ilişkisi

YZ, sağlıktan savunmaya, finanstan otonom araçlara kadar birçok alanda devrim yaratan günümüz teknolojilerinden en etkileyici ve ilgi çekici olanıdır. Birçok farklı alanda zorlu görevler karşısında başarılı sonuçlar üretebilmek için özellikle derin öğrenme algoritmaları olmak üzere karmaşık YZ algoritmaları geliştirilmiştir. Milyonlarca parametre ve karmaşık bağlantılardan oluşan derin öğrenme algoritmalarının nasıl çalıştığını ve kararlarını nasıl verdiğini anlamak da giderek zorlaşmaktadır. YZ algoritmalarının iç dinamiklerinin ve karar verme mekanizmalarının şeffaf olmaması ve anlaşılabilmesi durumu Şekil 4.3’de gösterildiği gibi "kara kutu" problemi olarak tanımlanmaktadır. Kara kutu problemi, özellikle derin sinir ağları gibi ileri YZ modellerinin popülerleşmesiyle ve yaygın olarak kullanılmasıyla daha belirgin bir sorun haline dönüşmüştür. YZ modellerinin karmaşıklık düzeyleri, büyük miktardaki verilerin modeller üzerindeki etkileri ve öğrenme süreçlerindeki çok miktardaki parametrenin güncellenmesi ve takip edilmesi gibi etkenler kara kutu probleminin en önemli sebepleri olarak gösterilebilir. Kara kutu problemi, kararların açıkça anlaşılabilmesi ve hatalı kararların sorgulanabilmesinden dolayı özellikle etik ve yasal sorumluluklar bağlamında önemli sonuçlar doğurabilmektedir. Örneğin, önyargılı verilerle eğitilmiş kara kutu modelleri ayrımcı sonuçlar üretebilir veya yargı kararlarının adil olmadığı kanısını doğurabilir.

Kara kutu modelleri, yalnızca parametrelerine bakılarak anlaşılabilen makine öğrenmesi modelleridir. Doğrusal modellerden ve karar ağaçlarından farklı olarak, makine öğrenmesi

ve özellikle derin öğrenme modelleri, karmaşıklıkları nedeniyle genellikle kara kutu modelleri olarak tanımlanmaktadır. Kara kutu modelleri doğası gereği şeffaf olmayan modellerdir. Yetenekli uzmanların bile bu 'kara kutu' modellerini tamamen kavraması zor olabilmektedir. Derin öğrenme teknikleri geniş bir ölçekte uygulama alanına sahip olduğundan, araştırmacılar, uygulayıcılar ve politika yapıcılar, algoritmanın karar verme sürecinde hangi faktörlerin daha önemli olduğunu anlamak isteyebilmektedir.

Bir ML ve DL modelinde, modelin dâhili işleyiş mekanizmasını bilmeden veya anlamadan herhangi bir giriş verisine karşılık gelen çıkış verisi kolaylıkla gözlenebilmektedir. Matematiksel modellerin aksine, ML modelleri, girdilerin çıktılara eşlenmesini doğrudan verilerden öğrenebilmektedir. Özellikle karar ağaçları gibi bazı modeller için bu eşleme kolayca fark edilebilirken DL modelleri gibi modeller için tahminlerin nasıl yapıldığını anlamak neredeyse imkânsız hale gelmektedir. Birçok ML ve DL modeli, esasen iç mekanizmaları ve tahminlerinin detaylarını belirtmeyen “kara kutular” olarak karşımıza çıkmaktadır.



Şekil 4.3. Kara kutu modeli

ML ve DL algoritmalarındaki bu şeffaflık ve anlama eksikliği, modellere olan güven ve benimsenme açısından ciddi sonuçlar doğurabilmektedir. Örneğin, model tahminlerinin yanlış olup olmadığı nasıl bilinebilmektedir? Bu, özellikle sağlık ve siber güvenlik gibi yüksek riskli alanlarda hayati önem taşımaktadır. Eğitilmiş bir model %99 doğruluğa sahipse, bir siber güvenlik uzmanı saldırı tahminine güvenebilir mi? Ya model en önemli saldırı olaylarını kaçırırsa? Ya yüksek doğruluk, test verilerindeki veri sızıntısından kaynaklanıyorsa ve örnek dışı performans çok daha kötüyse? Bu gibi sorulardan dolayı açıklanabilir YZ, makine öğrenmesinin benimsenmesi için hayati önem taşımaktadır. Finans, kefalet ve şartlı tahliye başvurularında ayırım gözetme, tıbbi teşhis, siber güvenlik vb. gibi yüksek riskli kararlar için ML modellerinin açıklanabilir olması zorunlu hale gelmektedir [395,396].

Kara kutu modellerinin faydası giderek daha sorgulanır hale gelmektedir ve bazı araştırmacılar tarafından bunların yüksek riskli kararlar için kullanılmaması gerektiği savunulmaktadır [397]. Doğası gereği açıklanabilir modelleri kullanmanın önemli avantajları olsa da, yüksek doğruluk nedeniyle kara kutu modelleri tartışmalara rağmen hala yaygın olarak kullanılmaktadır. Bunun yanı sıra, kara kutu modellerini şeffaf ve yorumlanabilir algoritmalara dönüştürme arayışı hem akademik hem de iş dünyasında giderek daha fazla ilgi görmektedir.

YZ'nin güvenilirliği, şeffaflığı, etik ve yasal kullanımı ve benimsenmesi için kara kutu problemine çözüm bulmak kritik öneme sahiptir. AYZ alanı, bu soruna çözüm bulmak için son yıllarda umut verici adımlar atmaktadır. Açıklanabilirlik, YZ sistemlerinin iç işleyişini daha anlaşılır hale getirerek, kullanıcıların, uzmanların, politika yapımcıların ve karar vericilerin bu sistemlere güvenmesini sağlamaktadır. Ayrıca, AYZ, modellerin hatalarının tespit edilmesinde ve düzeltilmesinde önemli bir görev üstlenerek YZ sistemlerinin genel performansını ve güvenilirliğini artırmaktadır.

4.2. Açıklanabilir YZ Terminolojisi

Açıklanabilirlik ifadesi farklı kişi, toplum veya topluluklar için farklı anlamlar ifade edebilmektedir. Ayrıca, uygulamaların farklılığına bağlı olarak açıklama kavramı da değişebilmektedir. Açıklanabilir AI sistemleri açısından üretilen çıktılar birçok çalışmada Açıklanabilir ve Yorumlanabilir olmak üzere iki farklı şekilde ifade edilmekte ve birbirinin yerine kullanılabilir. Ancak, literatürde birbirinin yerine kullanılsa da kapsamlı değerlendirildiğinde bu terimler çok yakından ilişkili olmakla birlikte, bazı çalışmalar farklılıklarını tespit etmekte ve bu iki kavramı birbirinden ayırmaktadır. Yorumlanabilirlik veya açıklanabilirlik için somut bir matematiksel tanım yoktur ve bunlar bir ölçü ile ölçülmemiştir; ancak, yalnızca bu iki terimi değil, aynı zamanda anlaşılabilirlik gibi ilgili kavramları da açıklığa kavuşturmak için bir dizi girişimde bulunulmuştur [398-400]. Ancak, tüm bu tanımlar matematiksel formaliteden yoksundur [401].

4.2.1. Açıklanabilirlik

Açıklamalar resmi bir tanım olarak "iç işlevlerini sınıflandırmak veya detaylandırmak amacıyla bir model tarafından gerçekleştirilen herhangi bir eylemi veya prosedürü ifade

eden bir modelin aktif özelliği" olarak ifade edilmiştir [402,403]. Bunun yanı sıra açıklamalar, daha fazla bilgi sağlayabilen ve insan kavrayışı için sistem tarafından üretilen anlamlı bir dizi çıktıyı ifade etmektedir. Bu açıklamalar belirli bir kararın neden verildiğini açıklamaktadır. Örnek olarak bir siber savunma sisteminin, belirli bir davranışın neden saldırgan olarak etiketlendiğini açıklayabilmesi verilebilir. Açıklanabilirlik, bir makine öğrenmesi sisteminin içindeki dahili mantık ve mekanikle ilişkilidir. Bir model ne kadar açıklanabilirse, model eğitilirken veya kararlar alınırken gerçekleşen dahili prosedürler açısından insanların elde ettiği anlayış o kadar derin olur [404].

4.2.2. Yorumlanabilirlik

Yorumlanabilirliğin en popüler tanımlarından biri, çalışmalarında [399] onu “bir insana anlaşılır terimlerle açıklama veya sunma yeteneği” olarak tanımlayan Doshi-Velez ve Kim'in tanımıdır. Bir başka popüler tanım da Miller'ın [405] çalışması olup yorumlanabilirlik "bir insanın bir kararın nedenini anlama derecesi" olarak tanımlanmıştır. Sezgisel olmasına rağmen, bu tanımlar matematiksel formaliteden ve titizlikten yoksundur [401].

Yorumlanabilirlik, belirli bir çıktıya hangi özelliklerin katkıda bulunduğunu ve sistemi simüle ederek sistemin bu kararı nasıl aldığını ortaya çıkarabildiği anlamına gelmektedir. Genel olarak yorum, kararları etkileyen özellikleri konu edinmektedir. Yorumlar ile insanlar veya alan uzmanları tarafından sistem tam anlamıyla anlaşılamamaktadır; ancak, makine öğrenmesi veya veri bilimcileri, verilerin hangi kısmının belirli sonuçlara katkıda bulunduğunu anlayabilmektedir. Bu yorumlar, sistemin belirli kararlar alması için ne tür verilerin gerekli olduğunu anlamamıza yardımcı olabilmektedir. Yorumlanabilirlik çoğunlukla bir modelin çıktılarının arkasındaki sezgiyle bağlantılıdır. Bir ML sistemi ne kadar yorumlanabilir olursa, sistemin girdileri ve çıktıları içindeki neden-sonuç ilişkilerini tanımlamanın o kadar kolay olduğu değerlendirilebilir [401]. Yorumlanabilir bir model, insanların iç mantığını veya altında yatan süreçleri anlayabilecekleri anlamına gelmemektedir.

4.3. Açıklanabilir YZ Hedefleri

YZ sağladığı kolaylıklarla her ne kadar önemli ölçüde fayda sağlasa da etik, gizlilik,

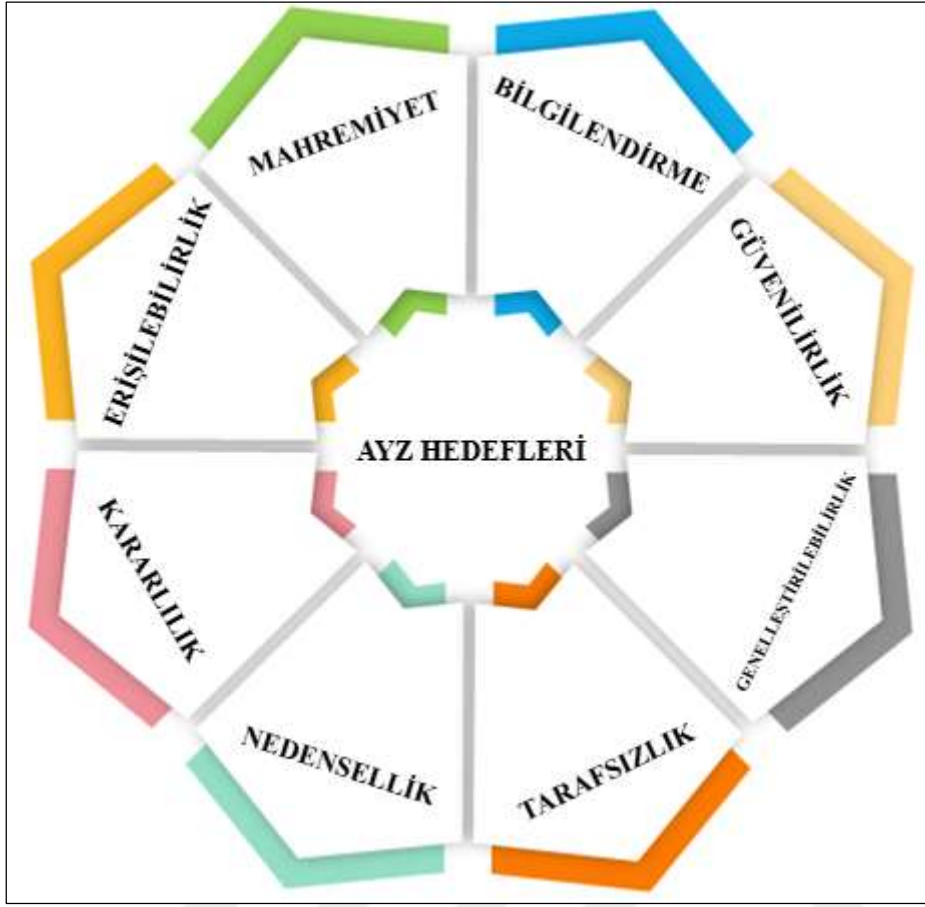
güvenilirlik, yanlılık, opaklık ve eşitlik gibi bir dizi önemli sorunu da içermektedir. Açıklanabilir YZ teknikleri bu sorunları birçok yönden ele alarak azaltabilmektedir. YZ uygulamalarının gelecekte daha geniş kitleler ve uygulama alanları tarafından güvenle kullanılabilmesi ve kabul edilebilmesi açıklanabilir YZ yöntemlerinin Şekil 4.4’de belirtilen hedeflere ulaşmasıyla mümkün olabilecektir.

Bilgilendirme: Açıklanabilir YZ; YZ kullanıcılarını, araştırmacılarını veya uzmanlarını bilgilendirmesi AYZ’nin en temel hedeflerinden birisidir. Bir açıklanabilirlik tekniği YZ modelinin sonuçlara nasıl, ne zaman ve ne koşullarda ulaştığı konusunda bilgi verebilmelidir.

Güvenilirlik: Bir AYZ algoritması belirgin, anlaşılması kolay ve doğru tanımlanan açıklamalarla YZ algoritmalarına güveni artırmalıdır. Bunu başarabilmenin yolu da NIST’de belirtildiği üzere [406], bir YZ uygulamasının güvenilirliği açıklanabilirliği ile elde edilmektedir. Güvenilir açıklamalar güvenilir YZ’yi doğurmaktadır. Bu yüzden bir AYZ algoritması güvenilir olmayı hedeflemelidir. AYZ’ye olan güven seviyesinin artması doğal olarak insanlar tarafından daha iyi benimsenmesine yol açacaktır.

Tarafsızlık: Son birkaç yılda, hem veri yanlılığı (bias) [407,408] hem de algoritmik yanlılık [409] ile ilgili önemli araştırmalarla YZ’de tarafsızlık kavramı ön plana çıkmıştır. Açıklanabilirlik, YZ modellerinde yanlılığın olup olmadığını veya ne zaman olduğunu belirlememize imkân tanımalıdır.

Nedensellik: Günümüzde YZ’nin temel sınırlamalarından biri modellemenin doğasında var olan nedenselliğin olmamasıdır [410]. Açıklanabilir YZ alanında, verilerdeki ilişkileri belirlemeye ve açıklamaya yönelik olarak giderek daha fazla çalışma yapılmaktadır [45,46]. Nedenselliği kanıtlamak için genellikle önemli ölçüde alan ve arka plan bilgisi gerekli olsa da, neden ve sonucu araştırmak için açıklanabilirlik kullanılabilir [410].



Şekil 4.4. AYZ hedefleri

Genelleştirilebilirlik: Genelleştirebilme bir görev üzerinde eğitilen bir AYZ modelinin farklı modellere de uygulanabileceği ve farklı görevler için bir başlangıç noktası olarak kullanılabileceğini ifade etmektedir. AYZ, bir modelin genelleştirilme sürecinde hangi sınırların ve sınırlamaların olabileceğini belirlememize ve anlamamıza da olanak tanımaktadır [411].

Kararlılık: Bir YZ modelinden aynı koşullarda aynı kararı vermesi beklenmektedir. Açıklanabilir YZ, bir modelin çeşitli koşullar altında ne kadar kararlı veya stabil çalışacağına dair bize fikir verebilmelidir.

Erişilebilirlik: YZ uygulamalarının teknik bilgisi olmayan kişiler tarafından erişilebilirliği, popülerliğin ve benimsenmenin artmasında önemli bir rol oynamaktadır. Açıklanabilir YZ, karmaşık YZ modellerinin bilgisini ve anlaşılmasını kolaylaştırabilir ve böylece sıradan insanların bunlarla uğraşırken yükünü azaltabilir [412].

Mahremiyet: YZ uygulamalarıyla birlikte gizlilik ve güvenliğin önemi artarken, açıklanabilir YZ'nin sağladığı avantajlardan biri de gizlilik seviyesini değerlendirme yeteneğidir [410]. Model açıklanabilirliği ile şifrelenmiş verilerde veya algoritmalarda gizliliğin ihlal edilip edilmediği daha kolay değerlendirebilmektedir [413]. Ayrıca, AYZ veriler hakkında bilgi sahibi olmadan modellerin ve algoritmaların bütünlüğünü sağlayarak açıklamalar üretebilmelidir.

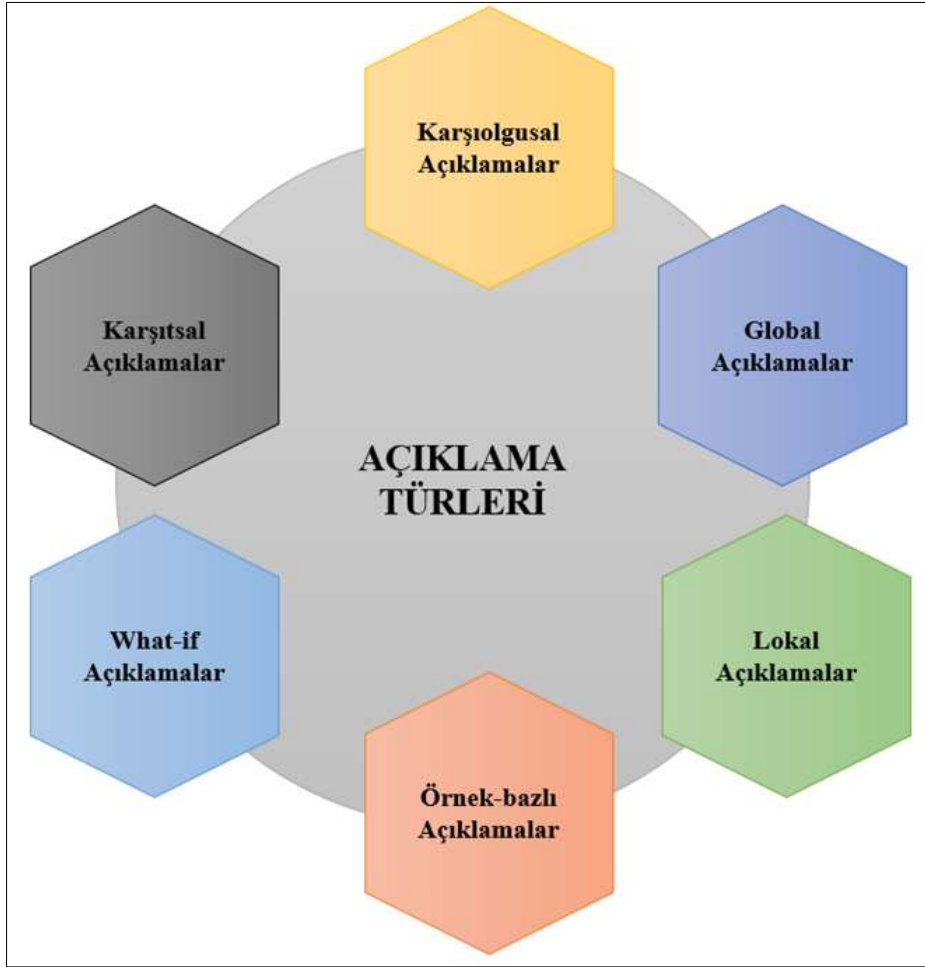
4.4. Açıklama Türleri

Açıklanabilir YZ yöntemleri, karmaşık sistemleri yorumlamamıza ve anlamamıza yardımcı olmak için farklı türde açıklamalar sağlayabilmektedir. Bu bölümde Şekil 4.5’de gösterildiği gibi açıklanabilir YZ tarafından sağlanan altı tür açıklama türü kapsamlı olarak sunulmuştur.

Global Açıklamalar: YZ sistemlerinde en sık karşılaştığımız soru “bir model nasıl çalışır?” sorusudur. Global açıklamalar, bu soruya yanıt vererek modellerin tahminlerine nasıl ulaştığını açıklamaya çalışmaktadır. Global açıklamalar görsel çizelgeler, matematiksel formüller veya model grafikleri şeklinde olabilmektedir. Global açıklamalar, bize modelin davranışının temsilini sağlaması açısından bütünsel bir açıklama türüdür.

Lokal Açıklamalar: Global açıklamalarla nasıl sorusuna cevap bulunduğunda “neden” sorusunu sorma ihtiyacı ortaya çıkmaktadır. Lokal açıklamalar bir modelin belirli bir girdi için neden bir tahmine ulaştığı sorusuna yanıt aramaktadır. Veri veya model algoritmasının belirli özellikleri ile bunlara karşılık gelen tahminleri eşleştirebilmektedir.

Karşıtsal Açıklamalar: Karşıtsal açıklamalar, belirli bir girdi için bir modelin neden başka bir tahmin yerine mevcuttaki tahmin sonucunu ürettiğini anlamamıza yardımcı olmaktadır. “Neden-değil” veya “neden X ve Y değil” sorularına yanıt verirler. Genellikle bir modelin tahminini ve beklenen davranışını anlamak için “neden” açıklamalarıyla birlikte kullanılırlar. Modelin farklı bir tahmin yapmasına neden olmak için girdilerde veya model parametrelerinde hangi minimum değişikliklerin gerekli olduğunu belirlemede yararlıdırlar.



Şekil 4.5. Açıklama türleri

What-if Açıklamaları: Bir YZ modelinin girdilerini veya model parametrelerini değiştirdiğimizde model çıktısındaki değişikliklerin varsayımsal açıklamalarını ifade etmektedir. Model tahminleri ve model özellikleri arasındaki ilişkileri anlamamıza yardımcı olmak için çok kullanışlı bir açıklama türüdür.

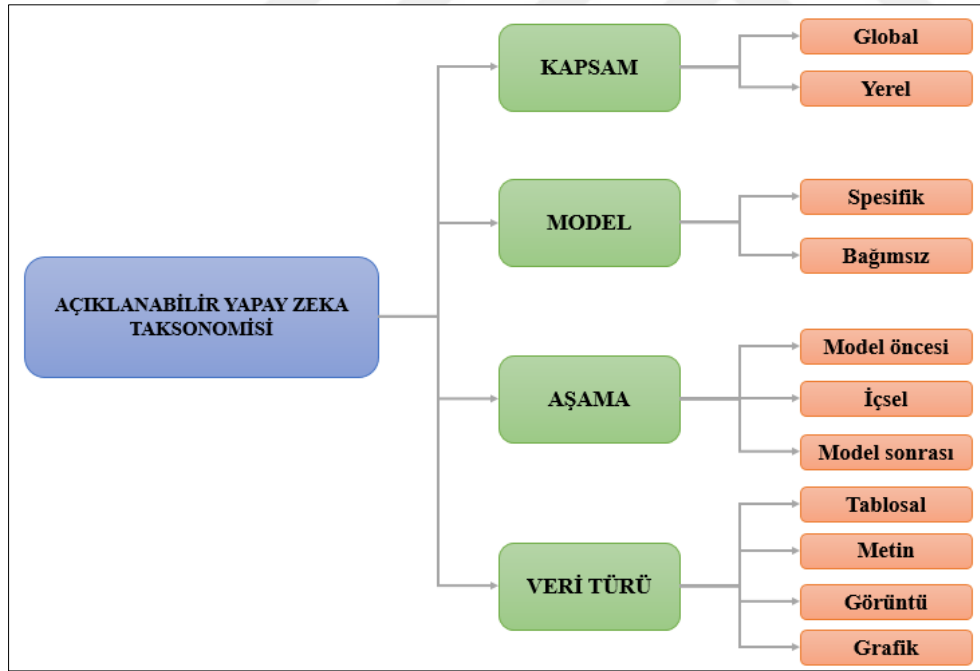
Karşı Olgusal Açıklamalar: Bir modelin girdi veya parametrelerinde, modelin istenen bir çıktı yapmasına neden olacak varsayımsal değişiklikleri anlatmaktadır. Modelin içyapısını anlamaya gerek kalmadan modelde yapılabilecek en küçük değişiklikleri tanımlayarak istenen sonuca nasıl ulaşılır sorusuna cevap verirler.

Örnek-Bazlı Açıklamalar: Örnek temelli açıklamalar, bir modelin davranışını veya temel alınan veri dağılımını basitçe verilerin belirli örneklerini vurgulayarak ifade eden bir

açıklama türüdür. Modelin benzer çıktıları tahmin edeceği benzer girdi örnekleri sunmak yaygın bir kullanım tercihidir.

4.5. Açıklanabilir YZ Taksonomisi

AYZ taksonomisi, farklı bakış açılarına göre [402,414] farklı sınıflandırmalara ayrılabilir. Gelişen açıklanabilir YZ alanına paralel olarak sınıflandırmalarında değişebileceği göz önünde bulundurulmalıdır. Ancak, AYZ teknikleri farklı bakış açıları ile değerlendirildiğinde kapsamlı bir AYZ taksonomisi Şekil 4.6'da sunulmaktadır. Bir AYZ tekniği bir veya daha fazla kategori altında sınıflandırılabilir. Örneğin, bir yöntem post-hoc, model-spesifik veya global olarak sınıflandırılabilir. Bundan dolayı her tekniğin kendi alt sınıflandırması altında incelenmesinin daha doğru bir yöntem olduğu değerlendirilmektedir. Bu bölümde çeşitli yaklaşımlarına ve özelliklerine göre AYZ yöntem ailesinin bir sınıflandırması sunulmuştur.



Şekil 4.6. AYZ taksonomisi

4.5.1. Kapsam bakımından açıklanabilir YZ

Açıklanabilir YZ yöntemleri, kapsam olarak global veya lokal olabilmektedir. Bazı yöntemler her iki kapsama da dâhil olabilmektedir. Global yöntemler, modellerin

davranışına bütüncül bir açıklama sağlarken lokal yöntemler, örnek seviyesinde davranışların açıklamasında kullanılmaktadır.

Global yöntemler, kapsamlı, tüm modelin tahminlerini açıklamaya çalışmaktadır. Global açıklamalar, model çıktılarında model yapılarının ve parametrelerinin nasıl bir etkiye sahip olduğunun anlaşılmasını sağlamaktadır. Bu yöntem, modelin girdi verilerinin özelliklere ve çıktılara nasıl eşlendiğine dair bir anlayış sağlayarak tüm modelin tek seferde anlaşılmasını sağlamaktadır [410]. Lokal yöntemler ise belirli bir örneğin çıkış verisiyle nasıl eşleştirildiğini açıklamaya çalışmaktadır. Bu yöntemler bir örnek üzerinden modelin çıktısına nasıl ulaşıldığını açıklamaktadır. Bu yöntemler yapıyı daha basit bir model kullanarak küçük bir ilgi alanındaki modele indirgemektedir.

4.5.2. Uygulama aşaması bakımından açıklanabilir YZ

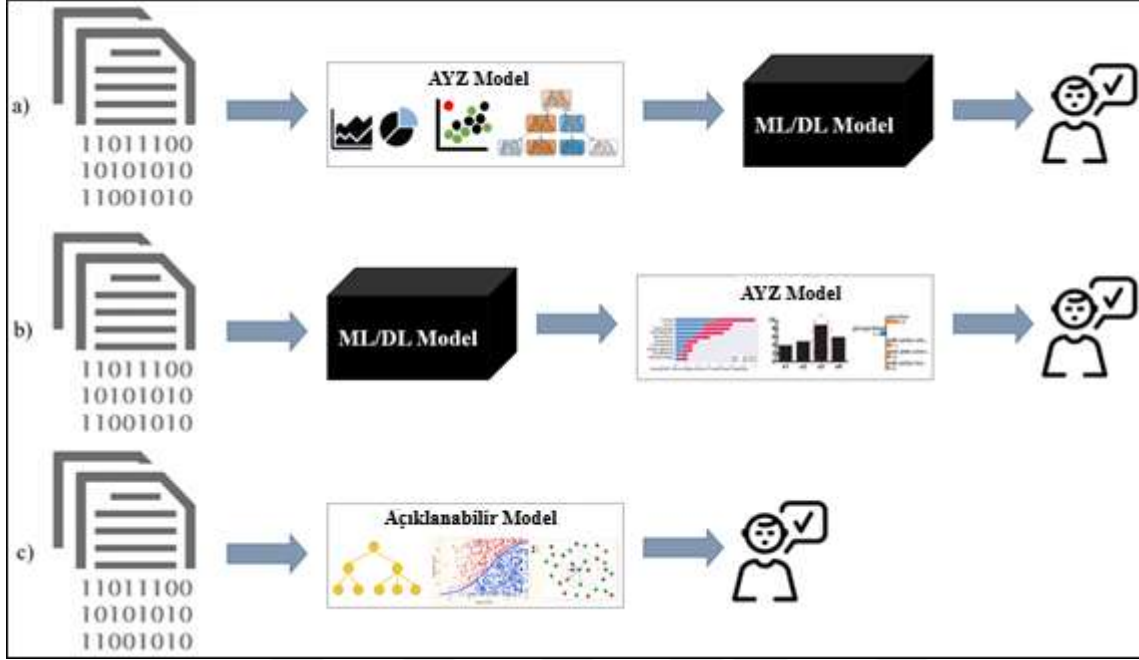
AYZ yöntemleri, Şekil 4.7’de gösterildiği gibi bir modelin tahminini yapmadan önce, tahmini sırasında veya tahmin sonrasında uygulanma durumlarındaki aşamalara göre açıklamalar sağlayabilmektedir.

Ante-hoc yöntemler, eğitim aşamasından önce harici olarak yorumlamayı içermektedir. Model öncesi açıklanabilirlik teknikleri, yalnızca verinin kendisine uygulanabildikleri için modelden bağımsızdırlar. Veri görselleştirme, keşifsel veri analizi tekniklerinden oluşan model öncesi teknikler açıklanabilirlik için kritik öneme sahiptir. Model öncesi açıklanabilirlik teknikleri ile modeli düşünmeden önce verilerin keşfedilmesi ve anlamlandırılması amaçlanmaktadır. Verilerdeki anlamlı özellikler, örüntüler ve seyreklik model öncesi veri yorumlanabilirliğine yardımcı olan bazı özelliklerdir.

İçsel yöntemler, açıklanabilirlik sağlamak için modelin içyapısından yararlanan ve kendi kendini açıklayan modelleri ifade etmektedir. Karar ağaçları ve Sparse Linear Regression modelleri temel intrinsic yöntemlere örneklerdir. Bununla birlikte, intrinsic teknikler model doğruluğu açısından dezavantajlara da sahiptir.

Post-hoc yöntemler, modellerin içyapılarını anlamaya gerek kalmadan herhangi bir eğitilmiş kara kutu modeline uygulanabilen teknikleri ifade etmektedir. Giriş örnekleri ve tahminleri arasındaki ilişkileri çözerek modellerin global veya lokal davranışlarının açıklamalarını

sağlamaktadır. Post-hoc yöntemler karar sistemlerinin eğitim sürecinden sonra haricen uygulanmaktadır. Post-hoc yöntemler, içsel modellere bile uygulanabilmektedir.



Şekil 4.7. Aşamalarına göre açıklanabilir YZ: a) Ante-hoc, b) Post-hoc, c) İçsel

4.5.3. Model bakımından açıklanabilir YZ

AYZ, belirli model sınıflarına uygulanmaya dayalı olacak şekilde modele özgü ve modelden bağımsız açıklamalar olmak üzere gruplandırılabilir.

Model-agnostik yöntemler, model sınıflarını sınırlandırmadan her tür modele uygulanabilen yöntemlerdir. Bu yöntemler açıklamaları ve model sınıflarını birbirinden ayırmaktadır. Böylece, açıklamalar model türünden bağımsız hale gelmektedir. Model öncesi ve sonrası açıklanabilirlik yöntemlerinin çoğu, geniş bir model topluluğuna uygulanabilir oldukları için model-bağımsız yöntemlerdir.

Modele özgü yöntemler, özellikle derin sinir ağları gibi karmaşık ve opak yapıya sahip tipik kara kutunun iyi bilinen örneklerine özgü oluşturulan yöntemlerdir. Bu yöntemler adından da anlaşılacağı üzere yalnızca oluşturulan modellere özgü olarak uygulanmaktadır. Modele özgü yöntemler, modelin belirli özelliklerinden veya mimarisinden yararlandıkları için modelden bağımsız yöntemlere göre daha avantajlı olabilmektedir. Modele özgü

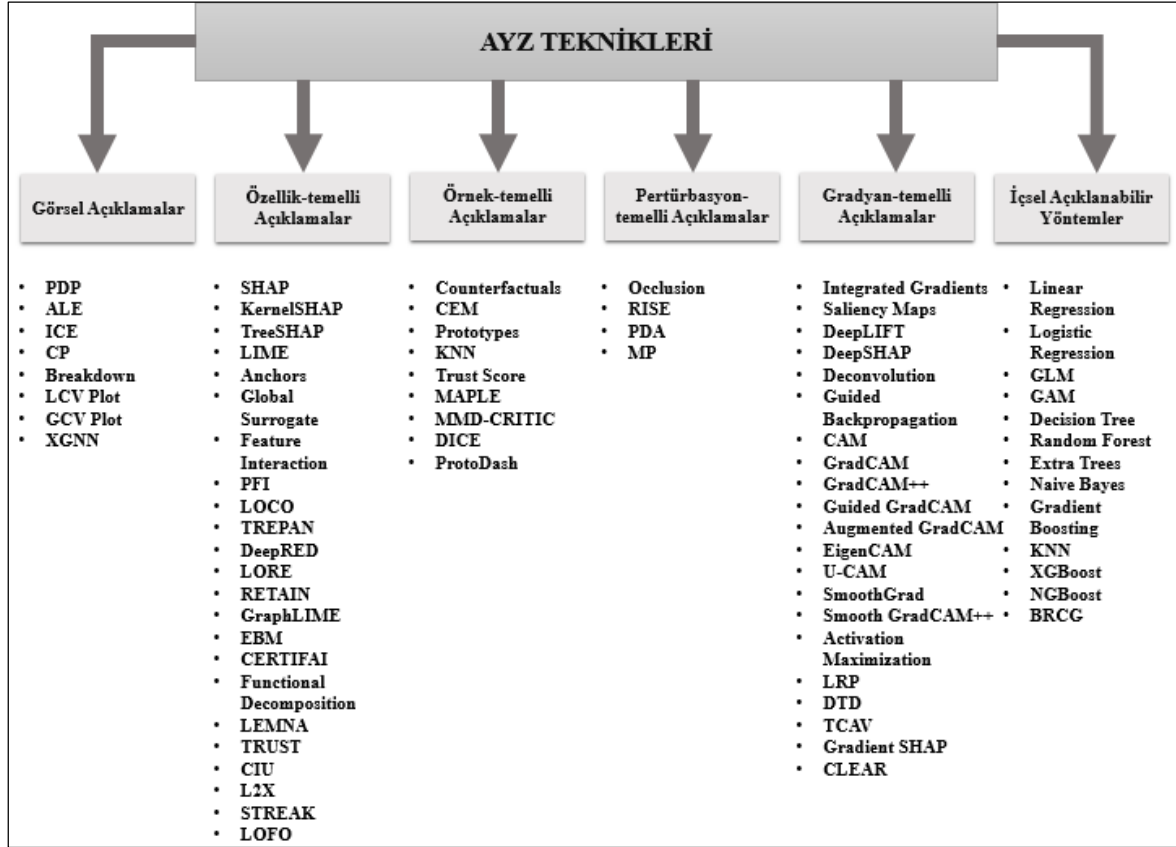
açıklamaların dezavantajı ise, belirli bir açıklama türüne ihtiyaç duyulduğunda model belirlemede bir sınırlama olmasıdır.

4.5.4. Veri türü bakımından açıklanabilir YZ

AYZ, oluşturulan açıklamanın çıkış formatına göre tablosal, metin, görüntü ve grafik olmak üzere sınıflara ayrılabilir.

4.6. Açıklanabilir YZ Teknikleri

YZ alanında açıklanabilirlik kavramının popülerlik kazanmasıyla birlikte son on yılda farklı AYZ yöntemleri geliştirilmeye başlandı. Literatürde, kara kutu modellerini açıklamaya ve yorumlamaya yardımcı olmak için birçok farklı AYZ yöntemleri geliştirilmiştir. Bu yöntemler farklı yaklaşımları benimsemekte ve farklı yorumlar sağlamaktadır [410]. Bu bölümde, Şekil 4.8’de görüldüğü gibi literatürdeki AYZ teknikleri altı farklı gruba ayrılarak kapsamlı olarak ele alınmıştır. Bu bölümde tüm teknikler detaylı olarak açıklanmamakla birlikte okuyucular için tüm teknikler hakkında detaylı bilgilere ulaşabilecekleri referanslar sağlanmıştır. Bununla birlikte, literatürdeki AYZ tekniklerinin Çizelge 4.1’de detaylı bir sınıflandırması ve özeti sunulmuştur.



Şekil 4.8. AYZ teknikleri

4.6.1. Görsel açıklamalar

Görsel açıklamalar, bir şeyleri açıklamanın en doğal ve kolay yollarından biri olarak karşımıza birçok farklı alanda çözüm olarak çıkmaktadır. Araştırmacılar, uzmanlar ve uygulamacılar tarafından kara kutu sorununun üstesinden gelmek için çeşitli görsel açıklama teknikleri geliştirilmiştir. Görsel açıklamalar, bir modelin girdi ve çıktısı arasındaki veya girdi nitelikleri arasındaki ilişkileri incelemek için birçok yöntem içermektedir. Bu teknikler genellikle kullanıcıların her bir girdinin çıktıya nasıl katkı sağladığını anlamasını kolaylaştırmaktadır. Özellik kümesinin küçük olması durumunda, bu korelasyonları yorumlamak kullanıcılar için daha kolay olmakla birlikte özellik kümesi büyüdükçe açıklamalar daha karmaşık hale gelmekte ve kullanıcıların bunları yanlış yorumlamasına neden olabilmektedir. Hem lokal hem de global açıklamalar görsel olarak desteklenmektedir.

4.6.2. Özellik temelli açıklamalar

Özellik temelli açıklama teknikleri de görsel açıklama yöntemleri gibi özelliklerin model tahminine katkısını değerlendirmeyi amaçlamaktadır. Bunun yanı sıra, özellik temelli açıklamalar tip, sağlamlık ve anlaşılabilirlik gibi bazı faktörleri de göz önünde bulundururlar. Bu tür yöntemler, her özelliğin bir tahmine ne kadar katkıda bulunduğunu gösteren özellik katkı vektörleri üretmektedir.

4.6.3. Örnek temelli açıklamalar

Örnek temelli açıklamalar genellikle modelden bağımsız yaklaşımlardır. Bu yaklaşımlar veri kümesinin belirli örneklerini seçerek bir modelin genel davranışını, yerel davranışını veya belirli veri örnekleri üzerindeki temel veri dağılımını açıklamayı amaçlamaktadır. Görsel açıklama veya özellik öneminden farklı olarak, kullanıcıların karar modelinin kompakt modellerini oluşturmaya yardımcı olurlar [23]. Genel olarak, örnek tabanlı yöntemler, bir örneğin özellik değerleri daha fazla bağlam taşıyorsa, yani verilerin, resimler veya metinler gibi bir yapıya sahip olduğu anlamına geliyorsa iyi çalışmaktadır.

4.6.4. Perturbasyon temelli açıklamalar

Perturbasyon temelli yöntemler, görüntüdeki pikseller, metindeki sözcükler veya diğer bazı veri türlerinin benzer öğeleri gibi bir modelin girdisini farklı varyasyonlarıyla yinelenmeli olarak değiştirerek ve çıktıdaki değişiklikleri gözlemleyerek yapay zeka modellerini açıklamayı amaçlamaktadır. Çıktıdaki değişikliklerin gözlemlerinin, girdinin hangi bölümlerinin çıkarım için özellikle önemli olduğunu göstermesi beklenir. Bozulmuş öğenin önemi, öğenin mevcut olduğu çıktı ile yokluğundaki çıktının karşılaştırılmasıyla tahmin edilir ve çıkarılmasının çıktıyı önemli ölçüde değiştirmesi önemli kabul edilir [415]. Perturbasyon temelli yöntemler genellikle görüntü temelli makine öğrenmesi modellerinde kullanılmaktadır. Bu yöntemler için kullanılabilecek bozulma türlerine açıklama ekleme (annotating), bulanıklaştırma, kaydırma ve maskeleyme örnek verilebilir. Ayrıca, belirli özellikler için sıfır veya rastgele karşı-olgusal örneklerin ikame edilmesiyle veya bir piksel koleksiyonunun (süper pikseller) gruplandırılmasıyla özellik seviyesinde yapılabilir [416].

4.6.5. Gradyan/Geri-yayılım temelli açıklamalar

Gradyan temelli açıklama yöntemleri bilginin akışına dayanmaktadır. Bu yöntemler girdi özellikleri ile çıktı arasındaki ilişkiyi belirlemek için geri yayılım sırasındaki bilgi akışını kullanırlar. Gradyan tabanlı yöntemler, görsel açıklamalar sağlamak için genellikle nöronların ısı haritalarını veya özellik katkılarını kullanmaktadır.

4.6.6. İçsel açıklanabilir metotlar

Bu sınıflandırma kriteri altında sunulan modeller, harici bir AYZ yöntemi gerektirmeden kullanıcılara çeşitli açıklamalar sağlayabilmektedir. Bu yöntemler ile modelin içsel mekanizmaları çıkarılabilir ve model, kullanıcıların anlayabileceği bir şekilde bilgi sağlayabilmektedir.

Çizelge 4.1. AYZ teknikleri sınıflandırması

AYZ Tekniği	Referans	Lokal/Global	Agnostik/ Spesifik	Post-Hoc/ İçsel	Veri Türü
PDP	[417]	Global	Agnostik	Post-hoc	Tablosal
ALE	[418]	Global	Agnostik	Post-hoc	Tablosal
ICE	[419]	Lokal/Global	Agnostik	Post-hoc	Tablosal
CP	[420]	Lokal	Agnostik	Post-hoc	Tablosal
Breakdown	[421]	Lokal	Agnostik	Post-hoc	Tablosal
LCV Plot	[422]	Lokal	Agnostik	Post-hoc	Tablosal
GCV Plot	[422]	Global	Agnostik	Post-hoc	Tablosal
XGNN	[423]	Lokal/Global	Spesifik	Post-hoc	Resim
SHAP	[424]	Lokal/Global	Agnostik	Post-hoc	Hepsi
KernelSHAP	[424]	Lokal/Global	Agnostik	Post-hoc	Hepsi
TreeSHAP	[425]	Lokal/Global	Spesifik	Post-hoc	Hepsi
LIME	[426]	Lokal	Agnostik	Post-hoc	Hepsi
Anchors	[427]	Lokal/Global	Agnostik	Post-hoc	Tablosal
Global Surrogate	[428]	Global	Agnostik	Post-hoc	Hepsi
Feature Interaction	[428]	Global	Agnostik	Post-hoc	Tablosal
PFI	[429]	Global	Agnostik	Post-hoc	Tablosal
LOCO	[430]	Lokal	Agnostik	Post-hoc	Tablosal
TREPAN	[431]	Global	Spesifik	Post-hoc	Tablosal
DeepRED	[432]	Global	Spesifik	Post-hoc	Tablosal, Resim
LORE	[433]	Lokal	Agnostik	Post-hoc	Tablosal
RETAIN	[434]	Global	Spesifik	Post-hoc	Tablosal
GraphLIME	[435]	Lokal	Agnostik	Post-hoc	Resim
EBM	[436]	Lokal/Global	Spesifik	Post-hoc	Tablosal
CERTIFAI	[437]	Lokal	Agnostik	Post-hoc	Hepsi
Functional Decomposition	[438]	Global	Agnostik	Post-hoc	Tablosal
LEMNA	[439]	Global	Spesifik	Post-hoc	Hepsi
TRUST	[440]	Lokal	Agnostik	Post-hoc	Tablosal
CIU	[441]	Lokal	Agnostik	Post-hoc	Hepsi

Çizelge 4.1. (devam) AYZ teknikleri sınıflandırması

L2X	[442]	Lokal	Agnostik	Post-hoc	Hepsi
STREAK	[443]	Lokal	Agnostik	Post-hoc	Resim
LOFO	[444]	Lokal/Global	Agnostik	Post-hoc	Tablosal
Counterfactuals	[428]	Lokal	Agnostik	Post-hoc	Tablosal, Resim
CEM	[445]	Lokal	Agnostik	Post-hoc	Tablosal, Resim
Prototypes	[428]	Global	Agnostik	Post-hoc	Resim
Trust Score	[446]	Lokal	Agnostik	Post-hoc	Resim
MAPLE	[447]	Lokal	Agnostik	Post-hoc	Tablosal
MMD-CRITIC	[448]	Global	Agnostik	Post-hoc	Hepsi
DICE	[449]	Lokal	Agnostik	Post-hoc	Hepsi
ProtoDash	[450]	Lokal	Agnostik	Post-hoc	Tablosal
Occlusion	[451]	Lokal	Agnostik	Post-hoc	Resim
RISE	[452]	Lokal	Spesifik	Post-hoc	Resim
PDA	[453]	Lokal	Agnostik	Post-hoc	Hepsi
MP	[454]	Lokal	Agnostik	Post-hoc	Resim
Integrated Gradients	[455]	Lokal	Spesifik	Post-hoc	Hepsi
Saliency Maps	[456]	Lokal	Agnostik	Post-hoc	Resim
DeepLIFT	[457]	Lokal	Spesifik	Post-hoc	Hepsi
DeepSHAP	[424]	Lokal	Spesifik	Post-hoc	Hepsi
Deconvolution	[458]	Lokal	Spesifik	Post-hoc	Resim
Guided Backpropagation	[459]	Lokal	Spesifik	Post-hoc	Resim
CAM	[460]	Lokal	Spesifik	Post-hoc	Resim
GradCAM	[461]	Lokal	Spesifik	Post-hoc	Resim
GradCAM++	[462]	Lokal	Spesifik	Post-hoc	Resim
Guided GradCAM	[461]	Lokal	Spesifik	Post-hoc	Resim
Augmented GradCAM	[463]	Lokal	Spesifik	Post-hoc	Resim
EigenCAM	[464]	Lokal	Spesifik	Post-hoc	Resim
U-CAM	[465]	Lokal	Spesifik	Post-hoc	Resim
SmoothGrad	[466]	Lokal	Spesifik	Post-hoc	Resim
Smooth GradCAM++	[467]	Lokal	Spesifik	Post-hoc	Resim
Activation Maximization	[468]	Lokal/Global	Spesifik	Post-hoc	Resim
LRP	[469]	Lokal	Agnostik	Post-hoc	Hepsi
DTD	[470]	Lokal	Spesifik	Post-hoc	Resim
TCAV	[471]	Lokal	Spesifik	Post-hoc	Resim
Gradient SHAP	[424]	Lokal/Global	Agnostik	Post-hoc	Resim
CLEAR	[472]	Lokal	Spesifik	Post-hoc	Tablosal, Resim
Linear Regression	[473]	Lokal/Global	Spesifik	İçsel	Hepsi
Logistic Regression	[474]	Lokal/Global	Spesifik	İçsel	Hepsi
GLM	[475]	Global	Spesifik	İçsel	Tablosal
GAM	[476]	Global	Spesifik	İçsel	Hepsi
Decision Tree	[477]	Lokal/Global	Spesifik	İçsel	Hepsi
Random Forest	[478]	Lokal/Global	Spesifik	İçsel	Hepsi
Extra Trees	[479]	Lokal/Global	Spesifik	İçsel	Hepsi
Naive Bayes	[480]	Lokal/Global	Spesifik	İçsel	Hepsi
Gradient Boosting	[481]	Lokal/Global	Spesifik	İçsel	Hepsi
KNN	[482]	Lokal/Global	Spesifik	İçsel	Hepsi
XGBoost	[483]	Lokal/Global	Spesifik	İçsel	Hepsi
NGBoost	[484]	Lokal/Global	Spesifik	İçsel	Hepsi
BRCG	[485]	Global	Spesifik	İçsel	Tablosal

4.7. Açıklanabilir YZ Araç Kutuları

Tek bir AYZ yöntemi, açıklanacak kişi için tüm ilgili model özelliklerini netleştirme işini genellikle tam olarak yapmamaktadır. Bu nedenle, ortak bir ara yüze sahip tek bir kütüphanede birden fazla açıklanabilirlik yöntemi uygulayan bu tür araç kutuları bu sorunun çözümü için ortaya çıkmıştır. Bu bölümde, Çizelge 4.2'de özetlenerek 2018 yılından beri yayınlanan bazı araç kutuları hakkında bilgiler sunulmuştur.

Skater: Skater, tüm ML modeli biçimlerine model açıklanabilirliği sağlamaktadır. Hem global hem de yerel açıklamalar sağlayarak herhangi bir kara kutu modeline açıklanabilirlik sağlayan açık kaynaklı bir python kütüphanesidir. Skater, görsel ve metinsel girdiler için DNN'ler de dâhil olmak üzere farklı türden eğitilmiş modellerin hem küresel hem de yerel açıklaması için toplam yedi AYZ yöntemi desteklemektedir. Modelden bağımsız çalışan bir AYZ araç kutusudur.

InterpretML: Açık kaynaklı bir python kütüphanesidir. Hem doğrusal regresyon, kural tabanlı, karar ağaçları vb. gibi beyaz kutu algoritmaların açıklamalarını hem de PDP, LIME, SHAP vb. teknikler aracılığıyla kara kutu algoritmaların açıklamalarını sağlamaktadır. Birleşik bir API kullanarak birden fazla açıklanabilirlik yöntemini yerleşik bir API ile karşılaştırmaktadır. Yeni başlayanlar için uygun olup Microsoft tarafından geliştirilen InterpretML beş modelden bağımsız ve dört şeffaf AYZ yöntemini uygulamaktadır.

iNNvestigate: Çeşitli sinir ağları mimarisine açıklanabilirlik sağlamaya yardımcı olan bir kütüphanedir. Birçok analiz yönteminin kullanıcılara kolay erişim ve basit uygulamasını sağlayan ortak bir arayüz ile tasarlanmıştır.

AI Explainability 360 (AIX 360): IBM tarafından geliştirilen AI Explainability 360 sekiz farklı ve son teknoloji açıklanabilirlik yöntemini ve iki değerlendirme ölçümünü içeren açık kaynaklı bir yazılım araç setidir. Uygulanan iş akışı, AYZ yöntemlerinin önerilen pratik, ağaç benzeri bir taksonomisini izlemektedir. Uygulanan yöntemler, verilerin açıklanabilirliği, doğası gereği açıklanabilir modeller ve hem küresel hem de yerel olarak post-hoc açıklanabilirlik dâhil olmak üzere çok çeşitli açıklanabilirlik ihtiyaçlarını kapsar.

explAIner: TensorFlow derin öğrenme çerçevesi için mevcut TensorBoard araç takımına bir eklenti olarak gerçekleştirilmiştir. Pek çok post-hoc DNN açıklaması ve analitik aracı, önerilen model anlama, teşhis ve iyileştirme boru hattı aşamalarıyla uyumludur. Hem araştırmacılar, hem uygulayıcılar hem de yeni başlayanlar için ideal bir araçtır.

FAT Forensic: Scikit-learn'e dayalı olarak, model ve veri kalite kontrollerine ve yerel ve küresel açıklanabilirliğe yönelik çeşitli uygulamalar içermektedir. Fairness ölçümleri üzerine araştırma yapan araştırmacılar için ve makine öğrenmesi modellerinin ön üretim sırasında izlenmesi isteyen uygulayıcılar için olmak üzere iki kullanım durumu için genel bir arayüz sağlayan kara kutu yöntemlerini içermektedir. Görüntü ve tablo tipindeki veriler için yöntemler desteklenir.

Alibi: Açık kaynaklı bir Python kütüphanesidir. Sınıflandırma ve regresyon için dokuz farklı AYZ yöntemi içermektedir. Bu, bazı ek destekleyici metinsel girdilerle birlikte görüntü veya tablo girdi verileri için hem yerel hem de küresel analiz yöntemlerini içermektedir.

PDPbox: PDPbox amacı kısmi bağımlılık grafiklerini kullanarak, denetimli makine öğrenmesi algoritmaları için belirli özelliklerin model tahminleri üzerindeki etkisini görselleştirmektir. PDPbox başlangıçta tüm scikit-learn modelleriyle uyumlu olacak şekilde tasarlanmış olsa da, farklı modellerin farklı tahmin arayüzlerine sahip olabileceği belirtilmiştir.

ICEbox: Herhangi bir denetimli öğrenme algoritması tarafından tahmin edilen modeli görselleştirmeye yönelik bir araç olan Bireysel Koşullu Beklenti (ICE) grafiklerini uygular. PyCEbox'da aynı şekilde ICEbox gibi ICE grafiklerini kullanan bir araç kutusudur.

Captum: PyTorch için açık kaynaklı bir model açıklanabilirlik kütüphanesidir. Çok modluluk, genişletilebilirlik ve kullanım kolaylığı bu kütüphanenin üç temel özelliğidir. Çok modluluk, metin, resim, video veya ses gibi farklı giriş türlerini destekler. Genişletilebilirlik, yeni özelliklerin ve algoritmaların eklenmesine olanak tanır ve böylece kitaplığı günceller. Kütüphane içindeki Captum Insights adlı ayıklamaya yardımcı olan bir görselleştirme aracı da kullanım kolaylığı sağlamaktadır.

H2O: H2O'daki çeşitli açıklanabilirlik yöntemleri ve görselleştirmeleri için uygun bir sarmalayıcıdır. Ana işlevler, `h2o.explain()` (genel açıklama) ve `h2o.explain_row()` (yerel açıklama), bireysel H2O modelleri ve diğer bazı modeller için çalışır.

Ethical ML: AYZ kütüphanesi, elde edilen sonuçların çıkarımlarını ve girdi özelliklerini analiz ederek veri analizine ve modellerin değerlendirilmesine olanak tanıyan çeşitli araçlar içerir.

Tf-Explain: Sinir ağının anlaşılmasını kolaylaştırmak amacıyla Tensorflow 2.0 için açıklanabilirlik yöntemleri sunar. Hem çekirdek API'si hem de `tf.keras.callbacks`'leri ile modellerin eğitimine ilişkin geri bildirim alınmaktadır.

What-If Tool: Regresyon ML modellerini açıklamak için kullanımı kolay, görsel bir arayüz sağlar. Açıklanabilirlik, farklı veri özelliklerinin önemini analiz ederek ve birden fazla model ve girdi verilerinin alt kümeleri genelinde model davranışını görselleştirerek ölçülebilir. Çok az kodlama gerektirir ve Jupyter Notebook, JupyterHub, Colab ve TensorBoard ile entegre edilebilir.

ELI5: ELI5, bir makine öğrenmesi modelinin tahminlerinin açıklanmasını ve görselleştirilmesini sağlayan ve bir makine öğrenmesi sınıflandırıcısında hata ayıklamaya yardımcı olan popüler bir Python paketi olan "Could Like I'm 5" in kısaltılmış bir biçimidir. ELI5, uzman olmayanların anlayabileceği basit ve sezgisel bir dil kullanarak modelin tahminlerinde en alakalı ve etkili olan faktörlere ilişkin yorumlanabilir ve açıklanabilir içgörüler sağlayan bir AYZ yaklaşımıdır. ELI5, bir özelliğin makine öğrenmesi modelindeki önemini göstermek için her özellik ile ilişkili ağırlıkları vermektedir. ELI5 tekniği, Rastgele Orman özellik ağırlık değerlendirmesine çok benzer. Ağırlıklar, ağaç topluluğundaki karar bölmeleri takip edilerek hesaplanır. Katkı puanı, her bir düğümün çıktı puanının ebeveynden alt düğüme farklılık derecesi ile hesaplanır. Tahmin, özellik katkılarının ve "yanlılık" puanının toplamına dayanır. ELI5, permütasyon önemini dahil ederek, dikkatini özelliklerin seçimine yönlendirir. ELI5 hem lokal hem de global açıklamalar sağlamaktadır. Ayrıca, ELI5 modelden bağımsız değildir ve kendi XGBoost uygulamasına sahiptir.

DALEX: moDeL Agnostic Language for Exploration and eXplanation (DALEX), makine öğrenmesi modellerini görselleştirmek ve açıklamak için kullanılabilecek çok yönlü bir R

paketidir. DALEX makine öğrenmesi modellerini etkili bir şekilde görselleştirmek için oluşturulmuştur. DALEX, özelliklerin ne olduğu ve modelin tahmini sonucunu etkilemeye yönelik katkılarının ne kadar ağır olduğu konusunda bize görsel bir anlayış sağlamamıza yardımcı olur. Makine öğrenmesinde hem regresyon hem de sınıflandırma görevleri için kullanılabilir. DALEX ile özelliğin önemini, özellikler ile sonuç değişkeni arasındaki ilişkiyi ve gelecek tahminlerinin doğruluğunu inceleyebiliriz.

Çizelge 4.2. AYZ araç kutuları

Araçlar	Referans	Giriş Türü	Lokal/ Global	Spesifik/ Agnostik	Kod Deposu
Skater	[486]	Tablosal Görüntü	Global	Agnostik	https://github.com/oracle/Skater
InterpretML	[487]	Tablosal	Lokal Global	Agnostik	https://github.com/interpretml/interpret
iNNvestigate	[488]	Tablosal Görüntü	Lokal Global	Agnostik	https://github.com/albermax/innvestigate
AI Explainability 360 (AIX 360)	[489]	Tablosal Görüntü	Lokal Global	Agnostik	https://github.com/Trusted-AI/AIF360
explAIner	[490]	Görüntü	Global	Agnostik	https://github.com/dbvis-ukon/explainer
FAT Forensic	[491]	Görüntü Tablosal	Lokal	Agnostik	https://github.com/fat-forensics/fat-forensics
Alibi	[492]	Tablosal Görüntü	Lokal Global	Agnostik	https://github.com/SeldonIO/alibi
PDPbox	[493]	Tablosal	Global	Agnostik	https://github.com/SauceCat/PDPbox
ICEbox	[494]	Tablosal	Lokal	Agnostik	https://cran.r-project.org/web/packages/ICEbox/index.html
PyCEbox	[495]	Tablosal	Lokal	Agnostik	https://github.com/AustinRochford/PyCEbox
Captum	[496]	Tablosal Görüntü	Global Lokal	Agnostik	https://github.com/pytorch/captum
H2O Explainability	[497]	Tablosal İstatistik	Global	Agnostik	https://docs.h2o.ai/h2o/latest-stable/h2o-docs/explain.html
Ethical ML	[498]	Tabular	Local Global	Agnostik	https://ethicalml.github.io/xai/index.html
Tf-Explain	[499]	Görüntü	Lokal	Agnostik	https://tf-explain.readthedocs.io/en/latest/
What-If Tool	[500]	Tablosal Görüntü	Lokal Global	Agnostik	https://pair-code.github.io/what-if-tool/
ELI5	[501]	Tablosal Görüntü	Lokal Global	Spesifik	https://eli5.readthedocs.io/en/latest/
DALEX	[502]	Tabular İstatistik	Global	Agnostik	https://github.com/ModelOriented/DALEX

AYZ araç kutuları, farklı uygulama senaryolarına ve gereksinimlere bağlı olarak belirli avantajlar ve sınırlamalar sunar. Örneğin, Skater modelden bağımsız bir yaklaşım sunarken, InterpretML daha kapsamlı yöntemlerle daha sistematik bir çerçeve sunar. Skater ve explAIner kullanıcı dostu arayüzleriyle öne çıkarken, FAT Forensics ve AIX 360 gibi araçlar daha fazla uzmanlık gerektirebilir. Örneğin Alibi, kullanıcıların açıklamaları yorumlayabilmesi için belirli bir teknik bilgi düzeyi gerektirebilir. Bu nedenle, araç kutuları

seçilirken proje gereksinimleri ve model türü gibi faktörler dikkate alınmalıdır. Gelecekte, bu araçların entegrasyonu ve kullanıcı deneyimlerinin iyileştirilmesi yoluyla AYZ'de daha fazla ilerleme beklenmektedir.

4.8. Açıklama Değerlendirme Kriterleri

Önceki bölümlerde sunulduğu gibi birçok açıklama yöntemi mevcuttur, ancak uygun bir açıklama çerçevesine sahip olmak için hala yanıtlanması gereken bazı sorular vardır. Açıklamanın iyi olup olmadığını nasıl anlayabiliriz? sorusu önemli sorulardan biridir. Eğer açıklamanın iyi olduğu belirtilmişse buna nasıl güvenebiliriz sorusu da bir diğer önemli sorudur. Ayrıca, herhangi bir ML/DL modeli açıklanabilirlik yöntemlerinden herhangi biri kullanılarak açıklanabilir bundan dolayı hangi AYZ yönteminin kullanılacağı veya hangi AYZ yönteminin temel alınan model ve kullanılan veriler için en uygun olduğu konusunda bir belirsizlik vardır. Bu soruların cevapları için açıklanabilirlik sonuçlarını değerlendirebilecek kriterlere ihtiyaç duyulmaktadır. Bu bölümde açıklama değerlendirme kriterleri sunulmuştur.

Literatürde çeşitli yorumlanabilirlik değerlendirme taksonomileri geliştirilmiştir. Mantık ve karmaşıklık açısından, Doshi-Velez ve Kim [399] açıklayıcı değerlendirmeleri uygulamaya dayalı değerlendirme, insana dayalı değerlendirme ve işlevsel temelli değerlendirme olmak üzere üç gruba ayırmıştır. Hoffman ve diğerleri [503] goodness, satisfaction, understanding ve verimlilik ölçümü olmak üzere dört açıdan ortak ölçüm yöntemlerinden bahsetmişlerdir. Mohseni ve diğerleri tarafından [504] açıklamaların "verimliliğinin" doğrulanması için hedeflenen kitle (ör. YZ uzmanları) ve metrikler aracılığıyla açıklamaların değerlendirilmesi için çalışma yapmışlardır. Honegger ve diğerleri [505] ise açıklanabilirliğin değerlendirilebilmesi için üç kriter önermişlerdir. Bu kriterler, özdeş (identical) örneklerin aynı açıklamalara sahip olmasının gerekliliğini belirten Identity kriteri, özdeş olmayan örneklerin açıklamalarının aynı olamayacağı belirten "Separability" kriteri ve benzer örneklerin benzer açıklamalarının olması gerekliliğini belirten Stability kriteridir. Son olarak, DARPA'nın AYZ programı, Kullanıcı memnuniyeti, Zihinsel model, Görev performansı ve Güven değerlendirmesi kriterlerini kullanarak AYZ sistemlerini değerlendirmiştir [506]. Bu çalışmada biz açıklamaların değerlendirilmesine yardımcı olacak kriterleri hesaplama ölçütleri ve bilişsel ölçütler olmak üzere iki temel yöntem olarak kategorize ettik. Bununla birlikte, DARPA tarafından belirtilen Kullanıcı memnuniyeti,

Zihinsel model, Görev performansı ve Güven değerlendirmesi gibi değerlendirme yöntemleri bilişsel ölçütlerin alt kısımları olarak değerlendirilmiştir. Benzer şekilde Fidelity, Stability, Sparsity, Comprehensibility, Faithfulness, Local Accuracy, Robustness, Consistency, Completeness ve Causality gibi değerlendirme yöntemleri hesaplamalı metriklerin alt kısımları olarak değerlendirilmiştir.

4.8.1. Hesaplamalı metrikler

Bu metrikler, AYZ sisteminde üretilen açıklamaların kalitesinin sayısal olarak ifade edilmesini belirtmektedir [504]. Ölçümler genellikle uygun şekilde geliştirilmiş sayısal denklemler kullanılarak yapılmaktadır. Sayısal hesaplamalara dayalı olduğu için hesaplamalı metrikler açıklama teknikleri oluşturmak için kılavuz olarak insan faktörü olmadan kullanılabilir [507]. Yani, bu metrikler, insan denekleri üzerinde çalışma yapmak yerine, gelişmiş açıklamaların doğruluğunu değerlendirmek için matematiksel hesaplamalara dayalı yollar oluşturmaktadır. Bu metrikler, modellerin güvenilirliğini standart performans ölçütleriyle açıklamasından dolayı YZ topluluğunda yaygın olarak kullanılmaktadır.

Doğruluk (Fidelity) [424], açıklamanın modelin gerçek tahminleriyle ne kadar yakından eşleştiğini ölçer. Başka bir deyişle, modelin tahmin edilen sonucunun açıklamanın tahminine ne kadar benzediğini değerlendirir. Amaç, modelin tahminleri ile açıklamanın çıktısı arasındaki farkı en aza indirmektir. Doğruluk şu şekilde formüle edilir:

$$Fidelity = 1 - \frac{\sum_{i=1}^n |\hat{y}_i - y_i|}{n} \quad (4.1)$$

Burada, $\hat{y}_i$ modelin orijinal tahminini, y_i açıklama yoluyla yapılan tahminini n ise toplam veri noktası sayısını ifade etmektedir.

Kararlılık (Stability) [508] metriği, benzer veri noktaları için açıklamaların ne kadar tutarlı olduğunu ölçmektedir. İki benzer veri noktasının çok farklı açıklamaları varsa, kararlılık düşük kabul edilir. Kararlılık şu şekilde hesaplanabilir:

$$Stability = 1 - \frac{|Explanation(x_1) - Explanation(x_2)|}{d(x_1, x_2)} \quad (4.2)$$

Burada $Explanation(x_1)$ ve $Explanation(x_2)$ iki benzer veri noktasının açıklamalarını, $d(x_1, x_2)$ iki veri noktası arasındaki mesafeyi ifade etmektedir.

Seyreklik (Sparsity) [426], bir açıklamanın ne kadar basit olduğunu ölçmektedir. Daha az özellik kullanan açıklamalar genellikle daha anlaşılır kabul edilmektedir.

$$Sparsity = 1 - \frac{k}{m} \quad (4.3)$$

Burada k , açıklamada kullanılan özelliklerin sayısı, m ise modelde kullanılan toplam özelliklerin sayısını ifade etmektedir.

Anlaşılabilirlik (Comprehensibility) [399], açıklamanın insanlar tarafından ne kadar kolay anlaşılabilirliğini ölçer. Bu genellikle açıklamanın uzunluğu ve kullanılan terimlerin basitliğiyle ilişkilidir. Bu ölçüt için matematiksel bir formül yerine, metin analiz yöntemleri kullanılabilir. Örneğin, anlaşılabilirlik, açıklamanın kelime sayısı ve karmaşıklığı hesaplanarak ölçülebilir. Daha kısa ve daha basit açıklamalar genellikle daha anlaşılır kabul edilir.

Sadakat (Faithfulness) [398], açıklamanın modelin gerçek işleyişini ne kadar doğru yansıttığını ölçer. Açıklama, modelin karar alma mekanizmalarıyla uyumluysa, sadakat yüksek kabul edilir. Sadakat ve sadakat terimleri bazen birbirinin yerine kullanılsa da, farklı amaçlara hizmet eder ve genellikle farklı şekilde değerlendirilir. İlk bakışta benzer görünebilirler, ancak aralarındaki fark, açıklamanın modelin çıktılarıyla ne kadar tutarlı olduğu (sadakat) ve açıklamanın modelin iç karar alma süreçlerini ne kadar doğru yansıttığı (sadakat) noktasında yatmaktadır.

4.8.2. Bilişsel metrikler

AYZ uzmanı olmayan insanların dahi kolaylıkla anlayabileceği açıklamaların değerlendirilmesi için bilişsel ölçümler kullanılmaktadır. AYZ'nin temel amacı, makinenin arkasındaki mantığı ve süreci insanlar için anlaşılır kılmak olduğundan, insanların anlayışı üzerinden açıklamaların değerlendirilmesi, açıklamalar için açık bir değerlendirme göstergesidir. Belirli görevlerde veya uygulamalarda hesaplamalı metriklerin dikkatli bir

şekilde alınması gerekmesine rağmen, katılımcıların kendi bakış açılarına göre en iyi açıklamayı belirlemelerini açıkça isteyerek konuların değerlendirilmesi birçok duruma genellenebilir [509].

Kullanıcı memnuniyeti, son kullanıcının görüşlerine dayalı olarak açıklamanın doğruluğunu ve kullanışlılığını ölçmektedir [506]. Kullanıcı memnuniyetini ölçmek için hem öznel hem de nesnel bazı yaklaşımlar araştırılmıştır. Literatürde bulunan yaygın yaklaşımlar kullanıcı görüşmeleri, Likert ölçekli anketler ve uzman vaka çalışmalarıdır. Gedikli ve ark. [510] memnuniyetlerini ve açıklama sağlamanın en etkili yollarını araştırmak için kullanıcı görüşmeleri kullanmıştır. [511] ve [512]'deki çalışmalar ise kullanıcının memnuniyetini ölçmek için Likert ölçekli bir anket kullanmışlardır.

Bilişsel model, bireylerin işlerin nasıl yürüdüğü hakkında nasıl düşündüklerini açıklayan bilişsel psikolojiden alınan bir kavramı ifade etmektedir [513]. Bilişsel model, kullanıcıların sistemi nasıl kavradığına dair bir tasvirdir [504]. Bilişsel modeller, kullanıcının sistemi anlaması ve farklı durumlarda kararlarını tahmin etme yeteneği olan bir açıklamanın etkinliğini ölçmek için insanın açıklayıcı akıl yürütmesinin felsefi, psikolojik ve doğal modellerinden türetilir [506]. Bu modeller, kullanıcıların sistemi ne kadar anladıklarına ve belirli bir karara nasıl vardıklarını değerlendirerek YZ sistem kararlarına güvenme veya şüphe etme konusunda fikir vermektedir. Bu ölçümler bireysel kararların, genel modelin, sistemin güçlü ve zayıf yönlerinin ve tahminlerinin nasıl üretildiğinin anlaşılmasına odaklanır. Bu modellere yönelik literatürde çeşitli çalışmalar yapılmıştır [514].

Görev performansı, açıklamanın kullanıcının karar verme sürecini iyileştirip iyileştirmediğini ve ayrıca kullanıcının AYZ sistemlerini ne kadar iyi anladığını ölçmektedir. Kullanıcı görev performansı, sistem tarafından desteklenen belirlenen bir görev için tamamlama süresi ve başarı oranı yoluyla kullanıcının performansının değerlendirilmesidir [506]. Bu metriğe yönelik literatürde çeşitli çalışmalar yapılmıştır [515,516].

Güven değerlendirmesi, kullanıcının AYZ sistemleri tarafından alınan kararlara ne zaman güveneceğini veya şüphe duyacağını bilme yeteneğinin değerlendirilmesidir [506]. Bir AYZ sisteminde güven değerlendirmesi son derece önemlidir. Zeki bir sisteme duyulan güven sistemin olumlu ya da olumsuz olarak anlaşılmasını etkileyen önemli bir faktördür. Bu aynı

zamanda sistemin etkinliğinin bir ölçüsünü de ifade etmektedir. Kullanıcı güveni bir sistemin kullanımı sırasında ve sonrasında anketler ve görüşmeler yoluyla doğrudan kullanıcı geri bildirimi istenerek değerlendirilebilmektedir. Bu metriğe yönelik literatürde çeşitli çalışmalar yapılmıştır [517,518].

4.9. Büyük Dil Modellerinde Açıklanabilir YZ

Bu bölüm, son zamanlarda giderek popülerlik kazanan AYZ ve Büyük Dil Modelleri (BDM)'ler arasındaki ilişkiyi ele almaktadır. Bu iki önemli ve gelişen çalışma alanının birbirleri üzerindeki etkileri hakkında bazı değerlendirmeler yapılmıştır. Bu konu, AYZ'nin önümüzdeki dönemdeki gelişimini ve değişim yönünü etkileyecek önemli gelişmelerden biri olduğu için bu çalışmaya dâhil edilmiştir.

BDM'ler, doğal dili anlamak ve analiz etmek için kullanılan DL algoritmalarını içeren modelleri ifade eder. BDM'ler, önceki kelimelerin bağlamı göz önüne alındığında bir sonraki kelimenin veya kelime dizisinin olasılık dağılımını tahmin ederek insan benzeri metinler üretebilir. Bu modeller genellikle yinelemeli sinir ağları (RNN'ler) veya dönüştürücüler gibi DL algoritmaları kullanılarak büyük miktarda metin verisi üzerinde eğitilir. BDM'ler, dil çevirisi, duygu analizi, sohbet robotu diyalogları ve daha fazlası gibi çeşitli dil ile ilgili işlevleri gerçekleştirme yetenekleri bakımından benzersizdir. BDM'ler, metin özetleme, dil çevirisi, soru cevaplama ve metin tamamlama gibi çeşitli doğal dil işleme faaliyetlerini gerçekleştirme potansiyelleri nedeniyle son derece popülerdir. OpenAI'nin GPT (Generative Pre-trained Transformer) serisi en bilinen BDM'lerden biridir.

BDM'ler, makinelerin insan benzeri metinleri nasıl anladığı ve ürettiği konusunda devrim niteliğinde bir başarı elde etti. Ancak, olağanüstü yeteneklerinin yanı sıra, bu modellerin iç mekanizmaları hala belirsizdir ve bu şeffaflık eksikliği uygulamalar için istenmeyen riskler oluşturmaktadır. Bu nedenle, bu modelleri anlamak ve açıklamak, davranışlarını, sınırlamalarını ve sosyal etkilerini açıklamak açısından büyük önem taşımaktadır [519]. BDM'lerin gizlilik, halüsinasyon ve güven açısından etik ve yorumlanabilirlik konusunda bazı zorlukları vardır. BDM'ler büyük miktarda veri işlediğinden, gizlilik endişeleri önemli bir zorluk olarak ortaya çıkmıştır. Kişisel ve hassas bilgilerin sorumlu bir şekilde işlenmesini ve istenmeyen sonuçların önlenmesini sağlamak zorunludur. Modellerin performansı ile kullanıcı gizliliğinin korunması arasında denge kuran etik yaklaşımların oluşturulması

gerekmektedir. Halüsinasyon, BDM'ler tarafından yanlış veya uydurma bilgi üretilmesini ifade eder. Bu modeller, halüsinasyon etkisi ile yanlış veya yanıltıcı içerik üreterek yanlış bilgiye neden olabilir. Son olarak, kullanıcılar oluşturulan içeriğin güvenilir ve tarafsız olduğuna güvenmelidir. Bu nedenle, BDM programlarının karar alma süreçlerinde şeffaflığının sağlanması ve üretilen çıktıların anlaşılır şekilde açıklanması güvenin oluşturulmasında önemlidir.

Son yıllarda BDM modellerinin artan popülaritesi, bu modellere olan güven sorununu beraberinde getirmiştir. Bu nedenle, AYZ'nin odağı son zamanlarda şeffaflık ve açıklanabilirlik eksiklikleri nedeniyle sıklıkla eleştirilen BDM'lere doğru kaymıştır [520]. BDM'lerin açıklanabilirlik ihtiyacını karşılamak, hem AYZ hem de BDM alanlarında önemli bir gelişme ve dönüşüm sağlayacaktır. Karmaşık yapıları nedeniyle, mevcut AYZ yöntemlerinin çoğu doğrudan BDM'lere uygulanamaz ve bu da yeni ve alana özgü açıklanabilirlik yaklaşımlarının geliştirilmesini gerekli kılar. Ayrıca, AYZ'nin BDM'lere sağlayacağı güven ve şeffaflık yetenekleri, üretkenliklerini ve uygulanabilirliklerini aktif olarak artırmaya devam edecektir. Sonuç olarak, son yıllarda popüler olan iki alan olan AYZ ve BDM, karşılıklı olarak gelişmeye ve birbirlerini değiştirmeye devam edecektir. BDM'lerin açıklanabilirliği için literatürde The Learning Interpretability Tool [521], Phoenix [522] ve Comgra [523] gibi çeşitli açıklanabilirlik yaklaşımları geliştirilmiştir. BDM'lerin AYZ ile açıklanabilirlik ihtiyaçlarını karşılamının yanı sıra, BDM'lerin AYZ için kullanılmasının önemli katkılar sağlayabileceği düşünülmektedir. BDM'lerin AYZ uygulamaları için kullanılmasının temel katkıları aşağıdaki gibi olabilir [524-530, 175]:

- Geleneksel açıklamalar modelin matematiksel ve teknik yönlerine odaklanırken, BDM'ler karmaşık bir modelin çıktılarını daha geniş bir kitleye daha akıcı ve derinlemesine açıklamalarla açıklayabilir.
- BDM, özellikle DL modelinde, karmaşık bir modelin karar alma süreçlerini basitleştirebilir, ancak AYZ modeli milyonlarca parametre ve katmanda sistem oluşturmanın karmaşıklığını artırır.
- BDM, yalnızca sonucu değil, aynı zamanda modelin bu sonuca nasıl ulaştığını da açıklayarak, neden-sonuç ilişkilerine dayalı açıklamaların doğrudan üretilmesini kolaylaştırır, ancak AYZ modelleri hala statiktir.

- Geleneksel açıklamalar girdi verilerine, katmanlara, bağlantılara ve kararlara odaklanırken, BDM, uzmanlar için daha derinlemesine teknik açıklamalar, daha basit ve daha anlaşılır açıklamalar sağlayabilir.
- AYZ modelleri modellerin parametrelerine, verilerine, matematiksel ifadelerine veya kurallarına bağlıken, BDM kullanıcılarla etkileşimlidir veya derinlemesine açıklamalar sunar.

BDM tekniklerinin AYZ uygulamalarında kullanımı önemli avantajlar sunabilir, ancak beraberinde bazı zorluklar ve sınırlamalar da getirir [530-534]. Bu zorluklar ve sınırlamalar şunlardır:

- Modeldeki eksik bilgiler nedeniyle doğru çıktılar üretmede halüsinasyon riski.
- Belirli parametreleri açıklamada ve belirli bir kullanım senaryosu veya bağlam için uygun açıklamalar üretmede BDM şeffaflığının eksikliği.
- Siber güvenlik, tıp, hukuk vb. gibi riskli veya kritik konular için BDM'lerin iç mekanizmalarına dair eksik açıklamalar.
- Dinamik BDM'ler göz önüne alındığında, literatürde esnek ve uyarlanabilir açıklanabilirlik için birçok olanak ve fırsat mevcuttur, ancak AYZ modelleri hala statiktir.

4.10. Saldırı Tespit Sistemlerinde Açıklanabilir YZ'nin Önemi

Yüksek performans, geliştirilebilirlik, esneklik ve sıfırıncı gün saldırılarını tespit etme yeteneği, yapay zeka tekniklerinin siber güvenlik alanındaki önemli avantajlarıdır. Ayrıca, yapay zeka tabanlı yaklaşımlar, zaman içinde değişen dinamik verileri işleyebilir, büyük ölçekli ve çok boyutlu verileri tanıyabilir ve verilerdeki anormallikleri belirleyebilir. Ancak, DL tabanlı yaklaşımların akış bilgisi eksikliği, kaçınma girişimlerine karşı savunmasızlık, ilgili özellikleri tasarlamak için gereken sınırlı veri bilgisi ve uygulamayı inceleyecek nitelikli alan uzmanlarının eksikliği gibi birçok dezavantajı vardır [15]. Özellikle son iki dezavantaj, açıklanabilirlik ihtiyacını ortaya çıkarmaktadır. Ek olarak, siber güvenlik uzmanlarının, araştırmacıların ve geliştiricilerin bu modelleri, değerlerini ve bulgularının doğruluğunu anlamaları için yapay zeka modellerinin açıklanabilir ve yorumlanabilir olması kritik öneme sahiptir. Yanlış alarmları araştırmak, sistem önyargılarını ve hatalarını belirlemek ve gelecekteki sistem iyileştirmeleri için bilinçli kararlar almak için açıklamalara ihtiyaç vardır [119].

Zeki STS modellerinde kullanılan tekniklerin giderek gelişen ve karmaşık yapıları, modellerin başarısını artırırken aynı zamanda şeffaflıklarını da azaltarak kara kutulara dönüşen akıllı STS'lerin benimsenmesini zorlaştırmaktadır. Birçok STS modeli, sonuçları siber güvenlik uzmanlarının kararlarını optimize etmelerini sağlayacak açıklamalar sunmadığı için pratik uygulamalarda kullanılma şansı bulamamıştır. Bu nedenle, siber güvenlik uzmanları, analistler ve karar vericiler, YZ modellerine dayalı akıllı STS'ler tarafından alınan kararlara güvenmek konusunda isteksizdir [315]. Herhangi bir STS tespitine olan güveni etkileyen en önemli faktörlerden biri, kötü amaçlı verilerin sisteme yönelik herhangi bir saldırıyı tespit etmedeki etkisini anlamaktır. Literatürdeki çoğu çalışma, STS'ye güvenmek yerine çeşitli sınıflandırma modellerinin doğruluğuna odaklanmıştır. AYZ teknikleri, siber güvenlik uzmanlarının YZ tabanlı akıllı STS'lerin kararlarının altında yatan veri kanıtlarını ve nedensel akıl yürütmeyi anlamalarını sağlayarak STS'lere olan güveni artırır [316].

Siber güvenlik ihlali durumunda, bir uzmanın YZ sisteminin mantığını anlaması genellikle zordur. Ayrıca, böylesine kara kutu bir ortamda karar vermek de zordur. YZ sistemleri bazen yanlış pozitif ve yanlış negatif sonuçlar şeklinde yanlış bulgular sunarak güvenlik uzmanlarını yanıltabilir ve tüm sistemin bütünlüğünü tehlikeye atabilir. Bu gibi durumlarda, ihlal senaryolarını veya anormal olayları daha doğru bir şekilde ele almak için açıklanabilirlik yeteneklerine ciddi bir ihtiyaç vardır. AYZ, güvenlik uzmanlarına ihlal riskini azaltmak ve genel güvenliği iyileştirmek için temel akıl yürütme ve tehdit tespit yetenekleri sağlar. AYZ ayrıca risk keşfi ve önceliklendirme, olay müdahale koordinasyonu ve kötü amaçlı yazılım tehdidi tespiti konularında da yardımcı olabilir. AYZ, açıklanabilirlik, yorumlanabilirlik ve hesap verebilirlik gerektiren durumlarda umut verici bir çözüm gibi görünmektedir [14]. Bir STS'nin, saldırıların bilgi sistemi altyapısının hangi bölümünü hedef aldığı, bilgi sistemlerinin hangi güvenlik açıklarından yararlandığı veya hangi güvenlik politikalarının ihlal edildiği gibi sorulara basit açıklamalar sunması gerekir. Bu ihtiyacı karşılayabilecek temel çözüm, sonuçları mantıksal olarak açıklayabilen modellere sahip STS'lerdir. Özetle, STS'lerdeki açıklamalar, kullanıcıların akıllı bir STS'deki örüntünün ne olduğunu, bir örneğin hangi özelliklerinin kötü amaçlı olduğunu ve neden bir saldırı olarak sınıflandırıldığını ve STS'nin kararlarına nasıl güvenilebileceğini daha iyi anlamalarına yardımcı olur.

4.11. Saldırı Tespit Sistemlerinde Açıklanabilir YZ'nin Zorlukları

Diğer uygulama alanlarının aksine, siber güvenlik, açıklama yöntemlerine duyulan ihtiyaç konusunda belirli zorluklar ortaya koymaktadır [536]. İlk olarak, siber güvenlik sistemleri tarafından üretilen sonuçların önemi nedeniyle, bu sistemlere tam bir güven duyulması gerekmektedir. Bir diğer önemli sorun ise güvenlik uygulamalarındaki çok sayıda öğrenilmiş model önyargısı ve insan önyargısıdır [537]. Öğrenilmiş model önyargıları genellikle dengesiz verilerden, tarihsel eşitsizliklerden veya hatta verilerin toplanma ve etiketlenme biçimindeki sistematik sorunlardan kaynaklanmaktadır. Örneğin, belirli coğrafi bölgelerden gelen daha fazla siber saldırı örneği konusunda eğitilmiş bir IDS, diğerlerinin risklerini haksız yere önceliklendirebilir veya göz ardı edebilir. Bir YZ modeli bu önyargıları yansıtıyorsa, bu yanlışlıkları pekiştiren açıklamalar sağlayabilir. Bu durum, YZ'nin karar alma sürecine olan güveni zedeler çünkü açıklamalar, belirli güvenlik tehditlerinin belirli bölgelerde, demografik özelliklerde veya davranışlarda haklı bir sebep olmaksızın daha yaygın olduğunu öne sürebilir. AYZ sistemleri bu önyargıları açıklamakta zorlanabilir ve istemeden zararlı veya yanlış güvenlik sonuçlarını güçlendirebilir. İnsan önyargıları, güvenlik analistleri veya operatörleri YZ sistemlerini yorumlarken veya onlarla etkileşim kurarken ortaya çıkar. Bunlar, doğrulama önyargısı (önceden var olan inançları destekleyen bilgileri tercih etme), bağlama önyargısı (karşılaşılan ilk bilgiye aşırı güvenme) ve diğer öznel etkiler gibi bilişsel önyargıları içerebilir. İnsanlar AYZ sistemlerinin çıktılarını incelemeye veya doğrulamaya dâhil olduklarında, önyargıları açıklamaları nasıl yorumladıklarını etkileyebilir. Örneğin, bir insan operatör belirli bir tehdit türünden şüpheleniyorsa, çelişkili bilgileri göz ardı ederek, YZ'nin açıklamasının şüphelerini doğrulayan kısımlarına seçici olarak odaklanabilir. Bu, insan ve makine önyargılarının birbirini güçlendirdiği çarpık karar almaya yol açabilir. Veriye dayalı karar alma modelleri bu önyargıları devralabilir ve bu tür modellerden gelen sonraki tahminler adil hatta yanlış olabilir. Güvenlik uygulamalarındaki açıklamalar, modeldeki temel nedensellikleri gösterebilir [538].

Tüm yeni teknolojilerde olduğu gibi, STS'lerde AYZ'nin uygulanması, avantajlarının yanı sıra bazı dezavantajlar da getirir. AYZ, STS uygulamalarına sağladığı açıklamalar sayesinde önemli iyileştirmeler sağlayabilir, ancak aynı zamanda saldırganlar için YZ uygulamalarına yönelik yeni saldırıları da kolaylaştırabilir; bu da STS'lerin açıklanabilir olması nedeniyle ciddi güvenlik tehditleri oluşturabilir [539]. Ancak, açık kaynaklı tartışmalarda olduğu gibi,

avantajlar ve dezavantajlar birlikte değerlendirildiğinde, AYZ'nin YZ'nin benimsenmesine yönelik riskleri azaltarak dezavantajlardan daha fazla avantaja sahip olduğu düşünülmektedir. Ayrıca, AYZ ilkeleriyle uyumlu YZ modellerinin tanımlanması veya modelden bağımsız AYZ çerçevelerinin geliştirilmesi, güvenlik açıklarının zamanında tespit edilmesi yeteneklerini artıracak ve endüstriyel ve insan senaryolarında büyük ölçekli YZ kullanımına olanak tanıyacaktır [15].

Bir diğer zorluk ise doğruluk ve açıklanabilirlik arasındaki dengedir. Doğrusal modeller ve karar ağaçları gibi daha basit modeller, karmaşık modellere göre daha yorumlanabilir, ancak istenen doğruluğu sağlayamayabilirler. Bu, hem yüksek doğruluğun hem de açıklanabilirliğin kritik öneme sahip olduğu STS'lerde önemli bir zorluktur.

Ölçeklenebilirlik ve hesaplama yükü, STS'lerde AYZ kullanımında da önemli zorluklar yaratabilir. SHAP ve LIME gibi bazı STS teknikleri, özellikle büyük veri kümelerine veya karmaşık modellere uygulandığında hesaplama açısından verimli olmayabilir. Bu durum, tehditlere anında yanıt verilmesi gereken gerçek zamanlı STS ortamlarında bir sınırlama olabilir.

AYZ'yi STS'lerde kullanırken, bağlama özgü açıklamalar üretme zorluğu da ortaya çıkar. AYZ tekniklerinin etkinliği ve verimliliği, uygulandıkları bağlama göre değişebilir. Bir bağlamda açıklanabilir olan bir model, başka bir bağlamda anlamlı açıklamalar sağlamayabilir.

Açıklamaların kullanışlılığı da kullanıcıların uzmanlığına göre değişebilir. Açıklamalardaki ayrıntı düzeyi ve teknik karmaşıklık, hedef kitleye göre uyarlanmalıdır. Bu durum, farklı kullanıcıların uzmanlık seviyelerine uygun açıklamalar üretme zorluğunu beraberinde getirmektedir.

Siber tehditlerin sürekli olarak gelişen ve değişen yapısı bir AYZ sistemi tarafından üretilen açıklamaların güncelliğini yitirebileceği anlamına gelebilmektedir. Bu durum, YZ modellerinin ve açıklama mekanizmalarının sürekli olarak güncellenmesini ve yeniden eğitilmesini zorunlu kılmaktadır.

Saldırganlar tarafından açıklanabilirliğin potansiyel olarak kötüye kullanılması AYZ'nin sağladığı önemli faydaların yanı sıra yeni bir güvenlik açığı riskini de ortaya çıkarmaktadır. AYZ'nin STS'ler için sağladığı içgörüler istismar edilerek STS'lerin zayıflıkları ortaya çıkarılabilmektedir. Saldırganlar, belirli davranışların nasıl saldırı olarak işaretlendiğini sistematik olarak inceleyerek, STS'yi atlatmak için yeni stratejiler geliştirebilirler. Yani açıklanabilirlik bir saldırı aracı olarak kullanılabilmektedir. Bu tür saldırılardan korunmak için açıklamalara sınırlı ve kontrollü erişim sağlanması, katmanlı güvenlik uygulanması veya eğitim sürecine saldırgan örneklerinin dâhil edilerek sürecin işletilmesi gibi önlemler alınabilmektedir [396].

4.12. Açık Konular ve Gelecek Yönler

Bu bölüm, STS'ler için AYZ metodolojilerinin incelenmesinden elde edilen içgörüler ve dersleri ele almaktadır. Ayrıca, bu tez çalışmasındaki ayrıntılı incelemelere dayanarak, açık konuları ele aldık ve gelecekteki yönler için olası araştırma zorluklarını vurguladık. Bu çalışmada, AYZ teknikleri ayrıntılı olarak ele alınmaktadır. Literatürdeki neredeyse tüm açıklanabilirlik yaklaşımları, okuyuculara rehberlik etmek ve araştırma sorularının yanıtlarını bulmak için bu çalışmada sunulmaktadır.

AYZ birçok kolaylık ve zorluk sunsa da, YZ araştırmacıları, kullanıcıları, profesyonelleri ve politika yapıcıların giderek artan ilgisini çekmektedir. AYZ yaklaşımlarının yardımıyla, YZ modellerinin güven ve şeffaflığını, hataların veya ML/DL mimarilerinin kolayca benimsenmesini ve ayıklanmasını sağlayarak "sorumlu YZ"ya doğru önemli adımlar atılmıştır. Dahası, AYZ önemli kazanımların yanı sıra bazı zorluklar da getirmiştir. AYZ tabanlı STS'ler kazanımları ve etkileri açısından güçlü bir araç olabilse de, AYZ'yi STS'lere uygulamada bazı zorluklar bulunmaktadır. Başlıca zorluklar aşağıda farklı alt başlıklar altında sunulmakta ve tartışılmaktadır.

4.12.1. AYZ özellikli STS'lere yönelik saldırılar

STS'ler, esas olarak güvenlik tehditlerini tespit ederek bilgi sistemlerinin güvenliğini sağlamaya odaklanır. Ancak, AYZ tabanlı STS'ler içyapıları ve karar mekanizmaları konusunda daha şeffaf olduklarından, saldırganlar STS'lerin güvenlik açıklarını kolayca tespit edebilir ve bunlara özgü gelişmiş saldırılar gerçekleştirebilirler.

Eğitim verilerini manipüle etmek veya bunlara zarar vermek için yapılan saldırılar (zehirleme gibi), kimlik doğrulama mekanizmalarını atlatmak için yapılan saldırılar (yüz doğrulama sistemi gibi), savunma mekanizmalarını manipüle etmek için yapılan saldırılar ve karşıt oligusal oyun saldırıları [13] son zamanlarda ortaya çıkan saldırılardır.

AYZ özellikli STS'lerin açıklanabilirlik özelliklerinin kötüye kullanılması ciddi sosyal, ekonomik ve etik sorunlara yol açabilir.

AYZ destekli STS'lerin kötüye kullanım durumlarında yapıcı kullanımını sağlamak için uygun kural ve düzenlemelerin oluşturulması, korunan sistemlere ve saldırılardan etkilenen kişilerin güvenliğine zarar verebilir [22].

4.12.2. Yasal uyumluluklar ve mahremiyet

Belirli koşullar altında kullanıcı verilerini belirtmek için protokol odaklı olsa bile [14] ve tasarımı gizlilik dikkate alınmalı ve modelin geri döndürülemez olduğundan emin olunmalıdır [402]. YZ modellerinin sistem yaşam döngüsü boyunca gizlilik sorunlarını dikkate alması gerekir; ancak bazı durumlarda daha fazla açıklanabilirlik, gizliliğin ihlal edilmesine neden olabilir.

4.12.3. Performans ve açıklanabilirlik arasındaki denge

Pratik güvenlik uygulamalarında AYZ algoritmaları kullanıldığında, algoritmaların zaman karmaşıklığının ele alınması gerekir [14] ve veri çeşitliliği, veri hızı ve veri kalitesi [402] dikkate alınarak hesaplama kaynakları, model doğruluğu ve hesaplama gücü, performans ve açıklanabilirlik arasında bir denge oluşturur. Bu sorunları ele alırken, birden fazla kısıtlama göz önünde bulundurularak optimize edilmiş AYZ algoritmaları geliştirilmelidir.

AYZ tabanlı STS'ler değerlendirildiğinde, performansta ve açıklanabilirlikte önemli iyileştirme sağlamada bazı zorluklarla karşılaşılabilir. Bazı kendi kendini açıklayan AYZ yaklaşımları şeffaflık sağlasa ve başvuranlar karar verme sürecini kolayca anlayabilse de, birçok çalışma her zaman tatmin edici sonuçlar vermemektedir. Örneğin, en iyi performans gösteren DL modelleri en az açıklanabilir olanlardır ve özellikle yüksek trafik oranlarına sahip karmaşık ve büyük veriler içeren sistemlerde yüksek performanslı modeller için en

kötü açıklanabilirliği oluşturlar. Bu nedenle, performans ve açıklanabilirliği aynı anda sağlamak da önemli zorluklardan biridir.

4.12.4. Değerlendirme ölçütleri

Değerlendirme ölçütleri, STS'lerde AYZ yaklaşımlarının uygulanmasındaki diğer önemli faktörler ve zorluklardır. Birçok AYZ yönteminin açıklamaların eksiksizliğini değerlendirmek için farklı yaklaşımları, performansları, doğrulukları ve ölçümleri vardır (Bkz. Çizelge 2.2).

AYZ yaklaşımları ve uygulamaları için bir dizi standart açıklanabilirlik değerlendirme ölçütü de bir diğer zorluktur. Temeldeki gerçek verilerden öğrenilen gerçekleri yansıtmaya yetenekleriyle ilgili farklı kriterlere göre, bazı durumlarda belirli tutarlılık seviyelerine ulaşmak için AYZ yöntemlerinin cezalandırılması ve ödüllendirilmesi gereklidir [540].

Açıklanabilirliklere sahip yapay zeka teknikleri geliştirmek, gelecekte hem yüksek performanslı hem de daha tanımlayıcı modeller, özellikle güvenilir, itibarlı, adil, istikrarlı, tutarlı ve kullanıcı dostu yaklaşımlar veya modeller gerektirecektir.

AYZ modelinin güvenilir performans göstermesini sağlamak için gereken yasal altyapılar, yalnızca STS'ler için değil, aynı zamanda daha somut ve evrensel kavramlarla tüm kritik alanlar için ve güvenlik ve gizlilik konularını daha fazla endişeyle ele almak için de zorluklar oluşturmaktadır.



5. MATERYAL VE METOT

Bu bölümde tez çalışması kapsamında önerilen yöntemde kullanılan materyal ve metotlar kapsamlı olarak ele alınmıştır. Bu kapsamda, kullanılan geliştirme altyapıları, veri kümeleri, veri ön işleme adımları, öznitelik seçme, veri dengeleme ve geliştirilen açıklanabilir STS mimarisine ilişkin bileşenler detaylı olarak açıklanmıştır. Açıklanabilir bir saldırı tespit sistemine yönelik gerçekleştirilen tez çalışması kapsamında sistem mimarisi içerisinde yer alan her bir adımda kullanılan materyal, metot ve teknikler ilerleyen alt başlıklarda detaylı olarak ele alınmıştır.

5.1. Veri Kümeleri

STS sistemleri tasarımında güvenilir test ve doğrulama veri kümeleri önemli bir bileşendir. Literatürdeki veri kümelerinin çoğunluğu güncel değildir. Bu veri kümelerinden bazıları trafik çeşitliliği ve hacimleri bakımından uygun değilken bazıları ise bilinen saldırıların çeşitliliğini kapsamamaktadır. Bununla birlikte, bazı veri kümelerinde özellik seti ve meta veriler eksiktir.

Bu tez çalışmasında STS tasarımının vazgeçilmez parçası olan veri kümeleri seçilirken güncel olması, saldırı çeşitliliğine sahip olması, gerçek dünya verilerine benzemesi ve literatürde yaygın olarak tercih edilmesi gibi parametreler dikkate alınmıştır. Bu doğrultuda, çalışmalar sırasında CIC-IDS2017 ve UNSW-NB15 veri kümeleri tercih edilmiştir. Deneysel çalışmalarda her iki veri kümesinden sınıf dağılımlarını koruyacak şekilde alt veri kümeleri oluşturularak bu veri kümeleri üzerinden sonuçlar elde edilmiştir.

5.1.1. CIC-IDS2017 veri kümesi

CIC-IDS2017 veri kümesi [370], gerçek dünya verilerine benzeyen normal trafik ve en yaygın saldırı türlerini içermektedir. Bununla birlikte, zaman damgası, kaynak ve hedef IP bilgileri, kaynak ve hedef portları, protokoller ve saldırılar temelinde akış etiketleri CICFlowMeter [545] aracı kullanılarak yapılan ağ trafiği analizinin sonuçlarını da içermektedir. Veri toplama süreci 3 Temmuz 2017 Pazartesi günü saat 09:00'da başlamış ve 7 Temmuz 2017 Cuma günü saat 17:00'de sona ererek toplam 5 gün sürmüştür. Bu veri seti

oluşturulurken gerçekçi arka plan trafiklerin oluşturulması hedeflenmiştir. İnsan etkileşimlerinin soyut davranışını profillemek ve doğal normal trafik arka plan trafiği oluşturmak için önerilen B-Profil sistemi [546] kullanılmıştır. Bu veri seti için, HTTP, HTTPS, FTP, SSH ve e-posta protokollerine dayalı olarak 25 kullanıcının soyut davranışı oluşturulmuştur. Veri yakalama dönemi toplam 5 gün sürmüştür. Pazartesi yalnızca normal trafikleri içeren gündür. Diğer günlerde ise hem normal trafikler hem de saldırı trafikleri üretilmiştir. Uygulanan saldırılar arasında Brute Force FTP, Brute Force SSH, DoS, Heartbleed, Web Attack, Infiltration, Botnet ve DDoS yer almaktadır. CICIDS-2017 veri kümesinin günlere göre trafik içerikleri Çizelge 5.1’de sunulmuştur. CIC-IDS2017 veri kümesi 2.830.743 satır ve 80 öznitelikten oluşmaktadır. Öznitelikler ve açıklamaları Çizelge 5.3’de sunulmuştur. Veri seti 14 saldırı türü içermektedir. Saldırıların ver kümesi içerisindeki dağılımı Çizelge 5.2’de gösterilmektedir. Bununla birlikte, pcap formatındaki ağ akış kayıtları günlere göre ayrılarak araştırmacıların hizmetine sunulmuştur.

Çizelge 5.1. CIC-IDS2017 veri kümesi günlere göre trafik içerikleri

Günler	Trafik içerikleri	Veri miktarı
Pazartesi	Normal Trafik	11 GB
Salı	Brute Force, FTP ve SSH saldırıları ve normal trafik	11 GB
Çarşamba	Dos/DDos, Dos slowloris, dos slowhttptest, Dos Hulk, Dos GoldenEye, Heartbleed saldırıları ve normal trafik	13 GB
Perşembe	Web and Infiltration Attack- Brute Force Web Attack-XSS, Sql Injection, DropBox Download saldırıları ve normal trafik	7,8 GB
Cuma	Botnet Ares, Port scan, firewall rules off, DDoS LOIT saldırıları ve normal trafik	8,3 GB

Çizelge 5.2. CIC-IDS2017 veri kümesinin sınıflara göre dağılımı

Sınıflar	Örnek sayısı
Normal	2,096,134
DoS Hulk	172,846
DDoS	128,016
PortScan	90,819
DoS GoldenEye	10,286
FTP-Patator	5,933
DoS slowloris	5,385
DoS Slowhttptest	5,228
SSH-Patator	3,219
Bot	1,953
Web Attack Brute Force	1,470
Web Attack XSS	652
Infiltration	36
Web Attack Sql Injection	21
Heartbleed	11
TOPLAM	2,830,743

Çizelge 5.3. CIC-IDS2017 veri kümesi öznitelik açıklamaları

No	Öznitelik ismi	Açıklama
1	Destination Port	Hedef port numarası.
2	Flow Duration	Akışın süresi (mikrosaniye cinsinden).
3	Total Fwd Packets	İleri yönde (source to destination) toplam paket sayısı.
4	Total Backward Packets	Geri yönde (destination to source) toplam paket sayısı.
5	Total Length of Fwd Packets	İleri yönlü paketlerin toplam uzunluğu (byte cinsinden).
6	Total Length of Bwd Packets	Geri yönlü paketlerin toplam uzunluğu (byte cinsinden).
7	Fwd Packet Length Max	İleri yönlü en uzun paket uzunluğu.
8	Fwd Packet Length Min	İleri yönlü en kısa paket uzunluğu.
9	Fwd Packet Length Mean	İleri yönlü ortalama paket uzunluğu.
10	Fwd Packet Length Std	İleri yönlü paket uzunluklarının standart sapması.
11	Bwd Packet Length Max	Geri yönlü en uzun paket uzunluğu.
12	Bwd Packet Length Min	Geri yönlü en kısa paket uzunluğu.
13	Bwd Packet Length Mean	Geri yönlü ortalama paket uzunluğu.
14	Bwd Packet Length Std	Geri yönlü paket uzunluklarının standart sapması.
15	Flow Bytes/s	Akış başına saniyede byte oranı.
16	Flow Packets/s	Akış başına saniyede paket oranı.
17	Flow IAT Mean	Akışlar arası ortalama zaman farkı (Inter-Arrival Time).
18	Flow IAT Std	Akışlar arası zaman farklarının standart sapması.
19	Flow IAT Max	En uzun akışlar arası zaman farkı.
20	Flow IAT Min	En kısa akışlar arası zaman farkı.
21	Fwd IAT Total	İleri yönde toplam zaman farkı.
22	Fwd IAT Mean	İleri yönlü ortalama zaman farkı.
23	Fwd IAT Std	İleri yönlü zaman farklarının standart sapması.
24	Fwd IAT Max	İleri yönlü en uzun zaman farkı.
25	Fwd IAT Min	İleri yönlü en kısa zaman farkı.
26	Bwd IAT Total	Geri yönde toplam zaman farkı.
27	Bwd IAT Mean	Geri yönlü ortalama zaman farkı.
28	Bwd IAT Std	Geri yönlü zaman farklarının standart sapması.
29	Bwd IAT Max	Geri yönlü en uzun zaman farkı.
30	Bwd IAT Min	Geri yönlü en kısa zaman farkı.
31	Fwd PSH Flags	İleri yönlü PSH bayrak sayısı.
32	Bwd PSH Flags	Geri yönlü PSH bayrak sayısı.
33	Fwd URG Flags	İleri yönlü URG bayrak sayısı.
34	Bwd URG Flags	Geri yönlü URG bayrak sayısı.
35	Fwd Header Length	İleri yönlü başlık uzunluğu.
36	Bwd Header Length	Geri yönlü başlık uzunluğu.
37	Fwd Packets/s	İleri yönlü paketlerin saniyelik oranı.
38	Bwd Packets/s	Geri yönlü paketlerin saniyelik oranı.
39	Min Packet Length	Minimum paket uzunluğu (byte).
40	Max Packet Length	Maksimum paket uzunluğu (byte).
41	Packet Length Mean	Paket uzunluklarının ortalaması.
42	Packet Length Std	Paket uzunluklarının standart sapması.
43	Packet Length Variance	Paket uzunluklarının varyansı.
44	FIN Flag Count	FIN bayrak sayısı.
45	SYN Flag Count	SYN bayrak sayısı.
46	RST Flag Count	RST bayrak sayısı.
47	PSH Flag Count	PSH bayrak sayısı.
48	ACK Flag Count	ACK bayrak sayısı.
49	URG Flag Count	URG bayrak sayısı.
50	CWE Flag Count	CWE bayrak sayısı (Congestion Window Reduced).
51	ECE Flag Count	ECE bayrak sayısı (Explicit Congestion Notification).
52	Down/Up Ratio	İndirilen ve yüklenen veri oranı.
53	Average Packet Size	Ortalama paket boyutu.
54	Avg Fwd Segment Size	İleri yönlü ortalama segment boyutu.

Çizelge 5.3. (devam) CIC-IDS2017 veri kümesi öznitelik açıklamaları

55	Avg Bwd Segment Size	Geri yönlü ortalama segment boyutu.
56	Fwd Header Length.1	İleri yönlü başlık uzunluğu (tekrar sütun).
57	Fwd Avg Bytes/Bulk	İleri yönlü ortalama byte/bulk oranı.
58	Fwd Avg Packets/Bulk	İleri yönlü ortalama paket/bulk oranı.
59	Fwd Avg Bulk Rate	İleri yönlü ortalama bulk oranı.
60	Bwd Avg Bytes/Bulk	Geri yönlü ortalama byte/bulk oranı.
61	Bwd Avg Packets/Bulk	Geri yönlü ortalama paket/bulk oranı.
62	Bwd Avg Bulk Rate	Geri yönlü ortalama bulk oranı.
63	Subflow Fwd Packets	Alt akış ileri yönlü paket sayısı.
64	Subflow Fwd Bytes	Alt akış ileri yönlü byte miktarı.
65	Subflow Bwd Packets	Alt akış geri yönlü paket sayısı.
66	Subflow Bwd Bytes	Alt akış geri yönlü byte miktarı.
67	Init_Win_bytes_forward	İleri yönlü başlangıç pencere boyutu.
68	Init_Win_bytes_backward	Geri yönlü başlangıç pencere boyutu.
69	act_data_pkt_fwd	İleri yönlü aktif veri paket sayısı.
70	min_seg_size_forward	İleri yönlü minimum segment boyutu.
71	Active Mean	Aktif dönemlerin ortalaması.
72	Active Std	Aktif dönemlerin standart sapması.
73	Active Max	Aktif dönemlerin maksimum süresi.
74	Active Min	Aktif dönemlerin minimum süresi.
75	Idle Mean	Boş (idle) dönemlerin ortalaması.
76	Idle Std	Boş dönemlerin standart sapması.
77	Idle Max	Boş dönemlerin maksimum süresi.
78	Idle Min	Boş dönemlerin minimum süresi.
79	Flow ID	Her akış için benzersiz kimlik (source-dest tuple).
80	Label	Trafığın normal veya saldırı etiketi (Benign/Attack).

5.1.2. UNSW-NB15 veri kümesi

Çalışma sırasında kullanılan bir diğer veri kümesi UNSW-NB15 [369] veri kümesidir. UNSW-NB15 veri kümesi, Avustralya Siber Güvenlik Merkezi tarafından saldırı tespit sistemlerini değerlendirmek amacıyla oluşturulmuştur. IXIA Perfect-Storm aracı kullanılarak oluşturulmuştur. Veri kümesi 100 GB'dan fazla ağ trafik verisi içermektedir. 100 GB'lık ham trafiği yakalamak için tcpdump aracı kullanılmıştır. Fuzzers, Analysis, Backdoors, DoS, Exploits, Generic, Reconnaissance, Shellcode ve Worms olmak üzere 9 saldırı kategorisi içermektedir. Veri kümesi 49 öznitelik ve 2,540,044 adet ağ akış verisinden oluşmaktadır. Çalışma kapsamında Çizelge 5.6'da gösterilen öznitelikler UNSW-NB15_features.csv dosyası ile araştırmacılara sunulmuştur. Bununla birlikte, çıkarılan tüm trafik kayıtları UNSW-NB15_1.csv, UNSW-NB15_2.csv, UNSW-NB15_3.csv and UNSW-NB15_4.csv olmak üzere 4 farklı dosya olarak araştırmacıların hizmetine sunulmuştur. Ayrıca, özellikle yapay zeka çalışmalarında kullanılmaları amacıyla oluşturulan veri kümesinden bir bölüm UNSW_NB15_training-set.csv ve UNSW_NB15_testing-set.csv olmak üzere bir eğitim ve test kümesi olarak

yapılandırılmıştır. Saldırı ve normal trafikler olmak üzere farklı türlerden ağ trafiklerini içeren eğitim kümesi 175.341, test kümesi ise 82.332 trafik kayıt sayısından oluşmaktadır. Bununla birlikte, pcap formatındaki ağ akış kayıtları araştırmacıların hizmetine sunulmuştur. UNSW-NB15 tüm veri kümesinin ve eğitim ve test veri kümelerinin veri dağılımlarına ilişkin bilgiler Çizelge 5.4 ve Çizelge 5.5’de sunulmuştur.

Çizelge 5.4. UNSW-NB15 test ve eğitim veri kümeleri veri dağılımları

Trafik içeriği	Eğitim seti veri miktarı	Test seti veri miktarı
Normal trafik	56,000	37,000
Analysis saldırısı	2,000	677
Backdoor saldırısı	1,746	583
DoS saldırısı	12,264	4,089
Exploits saldırısı	33,393	11,132
Fuzzers saldırısı	18,184	6,062
Generic saldırısı	40,000	18,871
Reconnaissance saldırısı	10,491	3,496
Shell Code saldırısı	1,133	378
Worms saldırısı	130	44
Toplam	175,341	82,332

Çizelge 5.5. UNSW-NB15 tüm veri kümesine ilişkin veri dağılımı

Trafik içeriği	Veri miktarı
Normal trafik	2,218,761
Analysis saldırısı	2,677
Backdoor saldırısı	2,329
DoS saldırısı	16,353
Exploits saldırısı	44,525
Fuzzers saldırısı	24,246
Generic saldırısı	215,481
Reconnaissance saldırısı	13,987
Shell Code saldırısı	1,511
Worms saldırısı	174
Toplam	2,540,044

Çizelge 5.6. UNSW-NB15 veri kümesi öznitelik açıklamaları

No	Öznitelik ismi	Açıklama
1	srcip	Kaynak IP adresi (Source IP address).
2	sport	Kaynak port numarası (Source port number).
3	dstip	Hedef IP adresi (Destination IP address).
4	dsport	Hedef port numarası (Destination port number).
5	proto	Kullanılan protokol türü (e.g., TCP, UDP, ICMP).
6	state	Bağlantının durum bilgisi (e.g., CON, FIN, INT, ACC, CLO).
7	dur	Bağlantının süresi (Duration of the connection in seconds).
8	sbytes	Kaynaktan hedefe gönderilen bayt sayısı.
9	dbytes	Hedekten kaynağa gönderilen bayt sayısı.
10	sttl	Kaynak TTL (Time To Live) değeri.

Çizelge 5.6. (devam) UNSW-NB15 veri kümesi öznitelik açıklamaları

11	dttl	Hedef TTL değeri.
12	sloss	Kaynak tarafından kaybedilen paket sayısı.
13	dloss	Hedef tarafından kaybedilen paket sayısı.
14	service	Uygulama katmanı servisi (ör. http, ftp, smtp, dns, ssh).
15	Sload	Kaynaktan hedefe veri yükleme hızı (bytes/s).
16	Dload	Hedekten kaynağa veri yükleme hızı (bytes/s).
17	Spkts	Kaynaktan hedefe gönderilen paket sayısı.
18	Dpkts	Hedekten kaynağa gönderilen paket sayısı.
19	swin	Kaynak TCP pencere boyutu.
20	stcpb	Kaynağın TCP base sequence numarası.
21	dtcpb	Hedefin TCP base sequence numarası.
22	dwin	Hedef TCP pencere boyutu.
23	tcprtt	TCP round-trip time (RTT) ölçümü.
24	synack	SYN-ACK zaman farkı (kaynak ile hedef arasındaki el sıkışma süresi).
25	ackdat	ACK veri gecikme zamanı.
26	smean	Kaynaktan hedefe ortalama paket boyutu.
27	dmean	Hedekten kaynağa ortalama paket boyutu.
28	trans_depth	HTTP istek zinciri derinliği.
29	response_body_len	HTTP yanıt gövdesi uzunluğu.
30	ct_srv_src	Aynı kaynak IP'den gelen bağlantıların aynı servise oranı.
31	ct_state_ttl	Aynı bağlantı durumuna sahip akışların sayısı.
32	ct_dst_ltm	Aynı hedef IP'ye yapılan son bağlantı sayısı (last time measure).
33	ct_src_dport_ltm	Kaynak IP'nin aynı hedef porta yaptığı son bağlantı sayısı.
34	ct_dst_sport_ltm	Hedef IP'nin aynı kaynak portundan aldığı son bağlantı sayısı.
35	ct_dst_src_ltm	Hedef ve kaynak çiftine ait son bağlantı sayısı.
36	is_ftp_login	FTP oturumu sırasında giriş yapılıp yapılmadığı (0 veya 1).
37	ct_ftp_cmd	FTP komutu sayısı.
38	ct_flw_http_mthd	HTTP metod (GET, POST, HEAD vb.) sayısı.
39	ct_src_ltm	Aynı kaynak IP tarafından yapılan son bağlantı sayısı.
40	ct_srv_dst	Aynı hedef IP'nin aynı servise yaptığı bağlantı sayısı.
41	is_sm_ips_ports	Kaynak ve hedef IP/port kombinasyonunun simetrik olup olmadığı (0/1).
42	ct_src_dport_mean	Kaynak IP'nin hedef port başına ortalama bağlantı sayısı.
43	ct_dst_sport_mean	Hedef IP'nin kaynak port başına ortalama bağlantı sayısı.
44	ct_srv_src_mean	Servis başına ortalama kaynak bağlantı sayısı.
45	ct_dst_src_mean	Hedef-kaynak çifti başına ortalama bağlantı sayısı.
46	ct_dst_ltm_mean	Hedef IP'nin son bağlantı ortalaması.
47	attack_cat	Saldırı kategorisi (Fuzzers, DoS, Backdoor, Exploits vb.).
48	label	Saldırı durumu (0 = Normal, 1 = Attack).
49	id	Bağlantı kimliği veya kayıt tanımlayıcısı (record ID).

5.2. Veri Kümesi Temizleme

Veri kümesi temizleme işleminde gerçek dünya verilerinde bulunan ve YZ algoritmalarını olumsuz etkileyen durumlar ele alınarak veri kümelerinde iyileştirmeler yapılmaktadır. Veri kümesi üzerinde YZ teknikleri uygulanmadan önce veri kümesi incelenmeli, veri kümesinde sayısal olmayan değerler sayısal değerlere dönüştürülmeli, eksik veya çift kayıtlar varsa işlenmeli, sonuca katkı sağlamayacak kayıtlar varsa silinmeli ve normalizasyon işlemi gerçekleştirilmelidir. Normalizasyon işlemi sonucunda veri kümesi içinde bulunan tüm

sayısal değerler 0 ile 1 arasında bir değere dönüştürülmektedir. Bu sayede yüksek sayısal değerlere sahip bazı özniteliklerin algoritma sonucunu etkileyerek performansı olumsuz etkilemesinin önüne geçilmiş olacaktır. Bununla birlikte, tanımlayıcı öznitelikler olan ve sınıflandırma algoritmalarının sonucunu etkilemeyen öznitelikler özellik seçme yöntemleri ile veri kümesinden çıkarılmalıdır.

Veri kümesi üzerinde sınıflandırma algoritmaları uygulanmadan önce bir veri sayısallaştırma tekniği olan Label Encoding işlemi ile veri kümesindeki kategorik değerler sayısal değerlere atanmakta ve ardından Eş. 5.1'de gösterilen normalizasyon işlemi gerçekleştirilmektedir. Normalizasyon işlemi sonucunda veri kümesi içinde bulunan tüm sayısal değerler 0 ile 1 arasında bir değere dönüştürülmektedir.

$$X' = \frac{X - X_{min}}{X_{max} - X_{min}} \quad (5.1)$$

burada, X orijinal değeri, X' normalize edilmiş değeri, X_{max} ve X_{min} ise sırası X 'in maksimum ve minimum değerlerini ifade etmektedir.

5.3. Öznitelik Seçimi

Öznitelik seçimi, gereksiz özellikleri kaldırarak sınıflandırma için en ilgili ve uygun öznitelik kümesini belirleme işlemidir. Öznitelik seçimi, YZ algoritmalarında işlem yükünün azaltılması ve modellerin performanslarının iyileştirilmesi için önemli bir süreçtir. Bu nedenle, öznitelik seçimi çok sayıda öznitelik içeren ağ trafik verilerinin sınıflandırılmasında kritik bir faktördür. Bunun yanı sıra, öznitelik seçimi depolama alanından tasarruf edebilir, hesaplama yükünü azaltarak hesaplama hızını artırabilir, gereksiz özellikleri kaldırabilir, gürültüyü azaltabilir ve aşırı öğrenme problemini önleyebilir. Öznitelik seçimi işlemi özellikle kısıtlı kaynaklara sahip altyapılar için enerji gereksinimini ve hesaplama yükünü hafifletmesi bakımından daha da önemli olmaktadır. Bu tez çalışmasında, öznitelik seçimi için Random Forest algoritması ve korelasyon matrisinden oluşan iki aşamalı bir öznitelik seçim prosedürü uygulanmıştır.

Random Forest [429], sınıflandırma ve regresyon için kullanılan popüler bir denetimli topluluk algoritmasıdır. Random Forest algoritması veri setinin alt kümelerinden oluşan çok

sayıda karar ağacının çoğunluk oylaması ile nihai sonuca varmasına dayanır. Karar ağaçları dallanmaya düğümlerdeki en yüksek bilgi kazancına sahip öznitelik ile başlamaktadır. Dallanmanın gerçekleşeceği özniteliğin seçimi ise özniteliğin önem derecesine bağlıdır. Veri setinde özniteliklerin önem derecelerinin hesaplanması, öznitelik seçimi ve nispeten daha az etki düzeyine sahip özniteliklerin azaltılmasında etkilidir. Özniteliklerin önem derecesine göre sıralanabilmesi Random Forest (RF) algoritmasının öznitelik seçiminde kullanılmasını mümkün kılmaktadır. Karar ağacındaki bir özniteliğin önemi, ağaçtaki düğümlerin ormandaki diğer ağaçların saflığını ne kadar artırdığına bakılarak belirlenmektedir. Öznitelik seçiminde Gini İndeksi safsızlık oranının ölçüsüdür. Buna göre Gini İndeksi küçük olan öznitelik en saf ve en yüksek öneme sahip öznitelik olarak kabul edilmektedir. Bir v düğümünün saflık oranını belirlemek için Gini İndeksi aşağıdaki şekilde hesaplanmaktadır [548]:

$$Gini(v) = \sum_{i=1}^l f_i(1 - f_i) \quad (5.2)$$

Burada l toplam sınıf sayısını, f_i ise i sınıfının sıklığını temsil etmektedir. v düğümünü dallandırmak için X_i özniteliğinin kazancı aşağıdaki şekilde hesaplanır:

$$gain(X_i, v) = Gini(X_i, v) - W_L Gini(X_i, v^L) - W_R Gini(X_i, v^R) \quad (5.3)$$

Burada v^L ve v^R v düğümünün sol ve sağ alt düğümlerini, W_L ve W_R ise alt düğümlere ulaşan örnek sayısını temsil etmektedir. X_i özniteliği için önem derecesi aşağıdaki şekilde hesaplanır:

$$Importance_i = \frac{1}{n_{tree}} \sum_{k \in S_{X_i}} gain(X_i, v) \quad (5.4)$$

Burada n_{tree} ağaç sayısını, $k \in S_{X_i}$ ise dallanmış düğümlerin kümesini ifade eder. Elde edilen önem derecesi/ skor normalize edilerek $[0,1]$ aralığına indirgenir.

Bu tez çalışmasında öznitelikler arasındaki korelasyonun gözlemlenmesi ve öznitelik seçimi için bir diğer yöntem olarak Pearson korelasyon katsayısı yöntemi kullanılmıştır. Pearson korelasyon katsayısı yöntemi iki sürekli değişken arasındaki istatistiksel ilişkiyi ölçen test istatistiklerini ifade etmektedir. Bir diğer tanım olarak, iki veri seti arasındaki doğrusal

korelasyonun bir ölçüsüdür [548]. Kovaryans yöntemine dayandığından, ilgilenilen değişkenler arasındaki ilişkiyi ölçmenin en iyi yöntemi olarak bilinmektedir. İlişkinin büyüklüğünün veya korelasyonun yanı sıra ilişkinin yönü hakkında da bilgi vermektedir. Her zaman -1 ile 1 arasında bir değere sahip olacak şekilde sonuçlar üretmektedir. Esasen kovaryansın normalleştirilmiş bir ölçümünü ifade etmektedir. Şu şekilde formüle edilmektedir:

$$r = \frac{n(\sum xy) - (\sum x)(\sum y)}{\sqrt{[n \sum x^2 - (\sum x)^2][n \sum y^2 - (\sum y)^2]}} \quad (5.5)$$

Burada,

- r = Pearson katsayısını
- n = örnek sayısını
- $\sum xy$ = eşleştirilmiş örneklerin toplamını
- $\sum x$ = x skorlarının toplamını
- $\sum y$ = y skorlarının toplamını
- $\sum x^2$ = x skorlarının karelerinin toplamını
- $\sum y^2$ = y skorlarının karelerinin toplamını ifade etmektedir.

5.4. Veri Dengeleme

Herhangi bir ağda olası saldırı trafiklerinin oranı normal trafıklere göre oldukça azdır. Bundan dolayı, saldırı tespit sistemlerinde kullanılan veri kümeleri oldukça az sayıda saldırı trafiği içermektedir. Bu tez çalışmasında kullanılan CIC-IDS2017 ve UNSW-NB15 veri kümeleri Şekil 5.1 ve Şekil 5.2’de gösterildiği dengesiz sınıflardan oluşan bir veri kümeleridir. Sınıflar arasındaki dengesiz dağılım sınıflandırma performansını olumsuz etkilemektedir. Özellikle azınlık sınıflar tespit oranını önemli ölçüde olumsuz etkilemektedir. Dengesiz sınıf problemi geleneksel makine öğrenmesi modellerinde sıklıkla ele alınmakla birlikte derin öğrenme yaklaşımlarında yeterince dikkate alınmamaktadır. Veri dengesizliği için yalnızca alt örnekleme (undersampling) tekniklerini kullanmak kullanışlı normal ağ trafiğinin elenmesine ve eğitim amaçlı kullanılan veri miktarının önemli ölçüde azalmasına neden olmaktadır. Tek başına aşırı örnekleme (oversampling) tekniklerini kullanmak ise gereksiz veri boyutu artışına ve gürültülere sebep olmaktadır. Bu tez çalışmasında, dengesiz sınıf probleminin üstesinden gelebilmek için STL (Smote+Tomek-

Link) [549] olarak adlandırdığımız Synthetic Minority Oversampling Technique (SMOTE) ve Tomek-Links aşırı örnekleme ve alt örnekleme yaklaşımlarının birlikte kullanılmasından oluşan bir yaklaşım önerilmiştir.

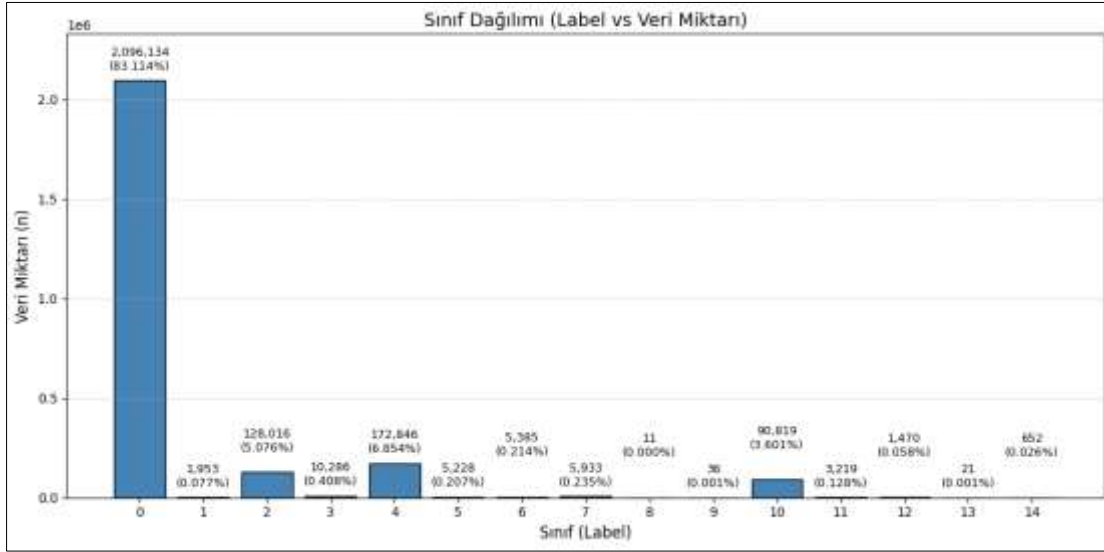
5.4.1. SMOTE

SMOTE(Synthetic Minority Over-sampling Technique), adından da anlaşılacağı üzere bir aşırı örnekleme tekniğidir. Bu yöntemde azınlık sınıftaki veriler yukarı yönlü örnekleme yapılarak sentetik veriler üretilmektedir. SMOTE, sentetik veriler üreterek rasgele aşırı örnekleme yöntemlerinin sebep olduğu aşırı uyum probleminin de üstesinden gelmektedir. SMOTE, özellik uzayına yakın örnekleri seçerek çalışmaktadır. Azınlık sınıftan rasgele bir örnek seçilmektedir daha sonra seçilen örneğin k tane en yakın komşu bulunur. En yakın komşulardan rasgele bir tane seçildikten sonra iki örnek özellik arasındaki fark 0 ile 1 arasında bir sayı ile çarpılır ve seçilen örnek değerine eklenir. Daha sonra iki örnek özellik arasına bir çizgi çizilir ve bu çizgi boyunca sentetik örnekler üretilmektedir. Gereken aşırı örnekleme miktarına bağlı olarak, k -en yakın komşulardan komşular rastgele seçilir [550].

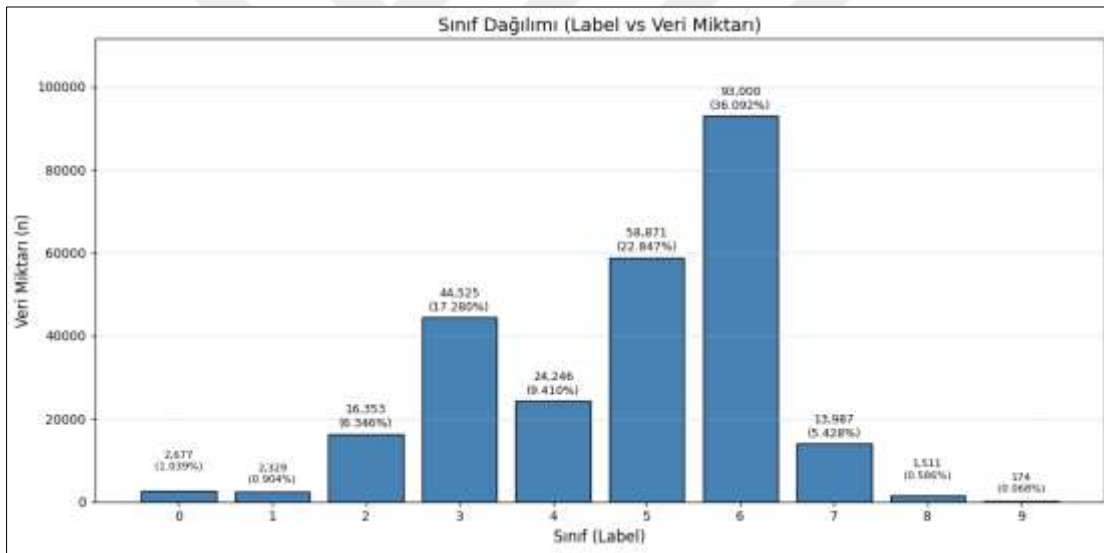
SMOTE örnekleri, azınlık sınıftan iki benzer örneğin (s, s^R) linear kombinasyonlarıdır ve şu şekilde tanımlanmaktadır;

$$n = s + d \cdot (s^R - s), 0 \leq d \leq 1 \quad (5.6)$$

burada, s^R k en yakın komşuluk sayısına göre s 'nin rasgele seçilmiş örneğini ve d , iki örnek arasındaki farkı ifade etmektedir.



Şekil 5.1. CIC-IDS2017 veri kümesi sınıf dağılımı



Şekil 5.2. UNSW-NB15 veri kümesi sınıf dağılımı

5.4.2. Tomek-Links

Tomek-Links, Tomek tarafından geliştirilmiş dengesiz veri setlerine uygulanan bir alt örnekleme tekniğidir [551]. En yakın komşuluk kuralı (NNR)'nin geliştirilmiş bir versiyonu olarak değerlendirilebilir. Bu yaklaşımda Tomek link üzerindeki örnekler veri kümesinden kaldırılmaktadır. Veri kümesinde birbirine en yakın fakat farklı sınıflara ait olan veri örnek çiftlerini oluşturur. Bu veri çiftlerine Tomek link adı verilmektedir. Temel fikir azınlık ve çoğunluk sınıfları birbirinden ayırmaktır.

x bir sınıfa ait y de diğer sınıfa ait örnek olsun, x ve y en yakın komşu ve $d(x,y)$; x ve y arasındaki mesafe olması koşuluyla;

$$T(x,y) \text{ is a Tomek - Link, if for any instance } i, \quad (5.7)$$

$$d(x,y) < d(x,i) \text{ or } d(x,y) < d(y,i)$$

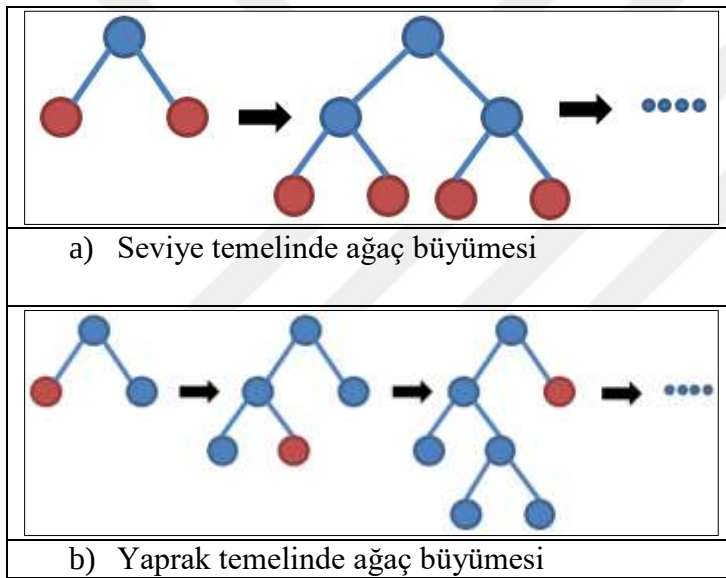
T-links iki sınıfı birbirinden ayırmaktadır. Bu link üzerindeki veri örnekleri gürültü olarak kabul edilmektedir. Çoğunluk sınıfa ait gürültüleri silmek sınıf ayrımını artırır ve veri dağılımını dengeli hale getirmektedir. Burada gürültü örneklerinin çoğunluk sınıftan silindiğine dikkat edilmelidir.

5.5. LightGBM Algoritması

Light Gradient Boosting Machine (LightGBM) Microsoft tarafından geliştirilen, makine öğrenmesi uygulamaları için ücretsiz ve açık kaynaklı dağıtık bir gradyan artırma (gradient boosting) çerçevesidir. LightGBM, hızlı, dağıtık ve yüksek performanslı ağaç bazlı öğrenme algoritmasını kullanan bir algoritmadır. Çeşitli bilgi sistemleri aracılığıyla üretilen verilerin boyutu her geçen gün artmaktadır. Bu durum verilerin hızlı işlenmesi gerekliliğini ortaya çıkarmakla birlikte geleneksel veri bilimi algoritmalarının daha hızlı sonuç vermesi zorlaşmaktadır. LightGBM yüksek hızı nedeniyle Light olarak isimlendirilmiştir. Bu özelliği sayesinde yüksek boyuttaki verileri hem hızlı bir şekilde işleyebilmekte hem de daha az belleğe ihtiyaç duymaktadır. LightGBM'nin bir diğer önemli özelliği de üretilen sonuçların doğruluğuna odaklanmasıdır. LightGBM, GPU öğrenimini desteklemekte ve bu nedenle veri bilimcileri, veri bilimi uygulama geliştirme için LGBM'yi yaygın olarak kullanmaktadır. LightGBM'nin bir başka avantajı da, kategorik özelliklerin en uygun şekilde bölünmesini desteklemesidir. LightGBM, bir grüplama yöntemiyle kategorik özelliklerin optimal ayrılmasını desteklemektedir. Bu şekilde sayısal dönüşümün neden olduğu seyrek veriler önlenmektedir. Bu avantajlarının yanı sıra küçük boyutlu veri kümelerinde overfitting problemine karşı hassas oluşu önemli bir dezavantajdır. Şekil 5.3, LightGBM'in diğer ağaç bazlı algoritmalarından farkını göstermektedir. Diğer algoritmalar ağaçları yatay olarak (level-wise) büyütürken, LightGBM ağacı dikey olarak (leaf-wise) büyütmektedir. Ağaç yapısının büyümesi için maksimum delta kaybı olan yaprak seçilmektedir. Aynı yaprağı büyütürken, yaprak temelinde (leaf-wise) algoritma, seviye temelinde (level-wise) bir algoritmaya göre kaybı daha fazla azaltabilir. LightGBM tipik olarak daha az sayıda karar ağacından ve karar

ağacı başına daha az yapraktan oluşmaktadır. Bu durum, LightGBM’i zaman açısından verimli hale getirmektedir.

LightGBM, Gradient-based One-Side Sampling (GOSS) ve Exclusive Feature Bundling (EFB) olmak üzere iki algoritmadan oluşmaktadır. LightGBM, karar ağacının özellik seçiminde gelişmiş bir histogram algoritmasını benimsemektedir. Burada EFB algoritması ile özelliklerin sayısı azaltırken GOSS algoritması tarafından eğitim aşamasındaki örnek sayısı azaltılmaktadır. Bu iki algoritma LightGBM’in özelliklerini oluşturmaktadır ve XGBoost [483], pGBRT [552] gibi diğer Gradient Boosting Decision Tree (GBDT) algoritmalarına üstünlük sağlamak için bir araya getirilmiştir.



Şekil 5.3. LightGBM ve diğer gradyan artırma algoritmaları arasındaki ağaç büyüme yapısının karşılaştırılması

GOSS temel olarak bir “data downsampling” tekniğidir. Model eğitimi aşamasında, büyük gradyanlı örnekler information gain üzerinde daha büyük bir etkiye sahiptir. Bundan dolayı, GOSS algoritması veri örneklerini downsampling yaparak büyük gradyanlı örnekleri tutar ve bilgi kazanımında yardımcı olmayan küçük gradyanlara sahip olanları rastgele düşürmektedir.

$\{x_1, x_2, \dots, x_n\}$ olmak üzere n boyutlu bağımsız ve özdeş olarak dağıtılmış bir veri kümemiz olduğunu varsayalım. Burada her x_i x^s uzayı içindeki s boyutlu vektörleri temsil etmektedir. Her gradient boosting iterasyonunda loss fonksiyonunun modelin çıkışına göre

negatif gradyanı $\{g_1, g_2, \dots, g_n\}$ ile ifade edilmektedir. GOSS yönteminde, eğitim örnekleri, gradyanlarının mutlak değerlerine göre azalan düzende sıralanır. Ardından, daha büyük gradyanlara sahip örnekler tutulur ve bir örnek alt kümesi olan A elde edilir. Daha küçük gradyanlara sahip örneklerden rasgele örnekleme yapılarak B kümesi oluşturulmaktadır. Böylece, GOSS algoritması ile her iterasyonda düşük etkiye sahip örnek sayısı azaltılarak tahmin kabiliyeti artırılmaktadır. Bir önceki turda eğitilen örnekler için, her örneğin gradyanı, örneğin tahminindeki hata derecesini göstermektedir.

O, her örneğin gradyan hesaplaması için karar ağacında bulunan sabit bir düğümdeki (fixed node) eğitim verisini ifade etmektedir. j segmentasyon özelliğinin d segmentasyon noktasındaki bilgi kazancı Eş. 5.8'de gösterilmektedir:

$$V_j(d) = \frac{1}{n_o} \frac{(\sum_{x_i \in A_l} g_i)^2}{n_l^j(d)} + \frac{1}{n_o} \frac{(\sum_{x_i \in A_r} g_i)^2}{n_r^j(d)} \quad (5.8)$$

Burada,

$$x_i \in A_l = x_i \leq d$$

$$x_i \in A_r = x_i > d$$

$$n_o = \sum I[x_i \in O], n_l^j(d) = \sum I[x_i \in O: x_{ij} \leq d]$$

$$n_r^j(d) = \sum I[x_i \in O: x_{ij} > d]$$

$$A_l = \{x_i \in A: x_{ij} \leq d\}$$

$$A_r = \{x_i \in A: x_{ij} > d\} \text{ ifade etmektedir.}$$

GOSS algoritması için a, daha büyük gradyan örneklerinin oranını ve $b \in (0, 1-a)$ rastgele seçilecek daha küçük gradyan örneklerinin oranını ifade etmektedir. a ve b'nin değerleri önceden belirlenmektedir. GOSS, veri dağılımındaki küçük gradyanlara sahip bu veri örneklerini $\frac{(1-a)}{b}$ sabit bir çarpanı ile rasgele örnekleme yapmaktadır. GOSS bu sayede orijinal veri kümesinin dağılımını çok fazla değiştirmeden doğruluğu yüksek tutarak veri boyutunu azaltmaktadır. Böylece, nihai bilgi kazancı Eş. 5.9 ile hesaplanmaktadır:

$$V_j(d) = \frac{1}{n_o} \frac{(\sum_{x_i \in A_l} g_i + \frac{1-a}{b} \sum_{x_i \in B_l} g_i)^2}{n_l^j(d)} + \frac{1}{n_o} \frac{(\sum_{x_i \in A_r} g_i + \frac{1-a}{b} \sum_{x_i \in B_r} g_i)^2}{n_r^j(d)} \quad (5.9)$$

Burada,

$$B_l = \{x_i \in B : x_{ij} \leq d\}$$

$B_r = \{x_i \in B : x_{ij} > d\}$ ifade etmektedir.

Sonuç olarak, split point'i belirlemek için tüm veri kümesinin bilgi kazancı yerine daha küçük bir veri alt kümesinin bilgi kazancı ($V_j(d)$) hesaplanmaktadır. Bunu sonucu olarak da hesaplama yükü önemli ölçüde azalmaktadır.

EFB esas olarak veri örnekleme ve özellik sayısının etkin bir şekilde azaltılması için kullanılmaktadır. EFB, doğruluk oranına zarar vermeden özellik sayısını azaltmayı ve buna bağlı olarak model eğitiminin verimliliğini arttırmayı amaçlamaktadır. EFB'nin iki temel işlem adımı vardır. Bunlar, bundle oluşturmak ve özellikleri aynı bundle'da birleştirmektir. Yüksek boyutlu veriler genellikle çok seyrek olmaktadır. Seyrek bir özellik alanında, birçok özellik mutually exclusive'dir. EFB, exclusive özellikleri tek bir özellikte güvenli bir şekilde toplayabilmektedir. Böylece, EFB ile seyrek özellikler birleştirilip daha yoğun özellikler oluşturulur. İki özellik tamamen mutually exclusive değilse, özellikler arasındaki non-mutual exclusion derecesini ölçmek için conflict ratio kullanılır. Değer küçük olduğunda, nihai doğruluğu etkilemeden iki özellik birleştirilebilmektedir. EFB, elde edilen özellik bundle'larından bireysel özelliklerle aynı özelliklere sahip histogramları oluşturmaktadır. Buna bağlı olarak karmaşıklık azalmakta, doğruluk seviyesi korunmakta ve daha düşük bellek tüketimi ile birlikte daha hızlı eğitim süreci olmaktadır.

5.6. Graf Sinir Ağları

Graf Sinir Ağları (GSA), düzensiz (non-Euclidean) veri yapılarında öğrenme gerçekleştiren derin öğrenme modelleridir. Geleneksel sinir ağları, vektör, matris veya düzenli grid yapılar üzerinde çalışırken, GSA'lar düğümler (nodes), kenarlar (edges) ve özniteliklerden (features) oluşan graf yapılı veriler üzerinde öğrenme yapmaktadır. Bu tür veriler, düğümler arasındaki ilişkisellik ve bağımlılıkları modellemeyi mümkün kılmaktadır. Bir düğümün etiketi veya davranışı yalnızca kendi özniteliklerinden değil, komşularının durumlarından da etkilenebilmektedir. Bu nedenle GSA'lar, bir veri örneğinin komşuluk bağlamında durumunu yakalayıp yapısal bilgiye duyarlı temsiller (structure-aware embeddings) üretmektedir [553].

GSA'ların temel çalışma ilkesi, mesaj iletimi (message passing) paradigmasına dayanmaktadır. Bu yaklaşımda her düğüm, komşularından bilgi ("mesaj") toplar, bu bilgileri bir toplulaştırma (aggregation) fonksiyonu aracılığıyla özetler ve ardından kendi temsilini günceller. Böylece düğüm temsilleri yalnızca kendi özniteliklerine değil, çevresindeki düğümlerin durumlarına da bağlı hâle gelir.

Matematiksel olarak bir graf $G = (V, E)$ olarak tanımlanmaktadır. Burada V düğümler (vertices) kümesini, E ise kenarlar (edges) kümesini temsil eder. Bu basit tanım, son derece karmaşık sistemleri modellemek için güçlü bir temel oluşturur. Bir graf mimarisindeki düğüm, kenar, öznitelik, düğüm öznitelikleri ve kenar öznitelikleri temel kavramlardır. Düğümler, bir graf içindeki temel varlıkları veya nesneleri temsil ederler. Kenarlar, düğümler arasındaki ilişkileri, bağlantıları veya etkileşimleri temsil ederler. Öznitelikler, gerçek dünya grafları genellikle sadece yapısal bilgiden daha fazlasını içerir. Hem düğümler hem de kenarlar, onları daha ayrıntılı tanımlayan ek bilgiler veya öznitelikler taşıyabilir. Düğüm öznitelikleri, bir düğümü tanımlayan vektörel bilgilerdir. Kenar öznitelikleri, iki düğüm arasındaki ilişkinin doğasını tanımlayan bilgilerdir.

Her düğüm v_i bir öznitelik vektörüne $x_i \in \mathbb{R}^d$ sahiptir. GSA'lar bu düğüm özniteliklerini ve bağlantı bilgilerini birleştirerek katman bazlı bir mesaj iletimi (message passing) mekanizması uygular. Her katmanda düğümler komşularından bilgi alır, bu bilgiyi birleştirir ve kendi temsilini günceller. l -inci katmanda bir düğüm için güncelleme şu şekilde formüle edilir:

$$h_v^{(l+1)} = \sigma(W^{(l)} \cdot AGG_{u \in \mathcal{N}(v)} f(h_v^{(l)}, h_u^{(l)}, e_{uv})) \quad (5.10)$$

Burada:

$h_v^{(l)}$: v düğümünün l -inci katmandaki gizli temsili,

$\mathcal{N}(v)$: v 'nin komşular kümesi,

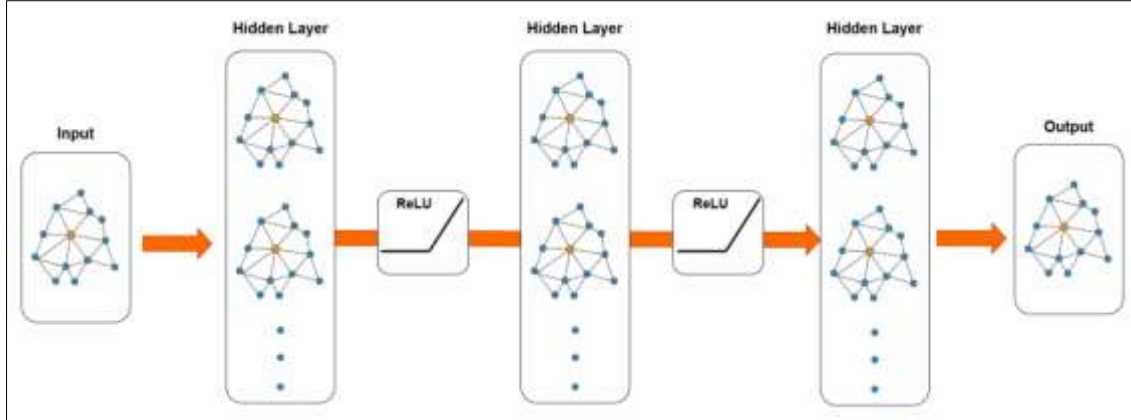
$f(\cdot)$: komşuluk ilişkisini modelleyen fonksiyon,

AGG : toplulaştırma (ortalama, toplam, dikkat ağırlıklı vb.)

$W^{(l)}$: öğrenilebilir ağırlık matrisi,

σ : aktivasyon fonksiyonunu ifade etmektedir.

Bu genel çerçeve, Message Passing Neural Networks (MPNN) olarak da adlandırılan Şekil 5.4’de gösterilen GSA mimarilerinin temelini oluşturmaktadır. Mesaj iletimi, düğümler arası bağımlılıkları, yapısal örüntüleri ve komşuluk bağlamını öğrenmeye imkân tanıyarak, GSA’ları klasik CNN/MLP modellerinden ayırır. Bu sayede öğrenilen temsiller “structure-aware embeddings” olarak adlandırılmaktadır [554,555].



Şekil 5.4. Graf sinir ağı mimarisi

Zaman içerisinde, GCN [556], GraphSAGE [557], GAT [558], GIN [559], ChebNet [560] gibi birçok varyasyon geliştirilmiştir. Her model bu çerçeveyi farklı bir toplulaştırma/güncelleme stratejisiyle genişletmektedir. GCN, normalize edilmiş ortalama mesaj yayılımı (mean propagation) uygular ve komşuluk bilgisini spektral uzayda sadeleştirilmiş biçimde işler. GraphSAGE, büyük ölçekli graflarda örnekleme (sampling) mekanizması kullanarak induktif öğrenmeyi mümkün kılar; böylece daha önce görülmemiş düğümler üzerinde genelleme yapılabilir. GAT, “dikkat katsayıları (attention coefficients)” ekleyerek her komşunun katkısını dinamik olarak öğrenir ve heterojen komşuluklarda esneklik sağlar. GIN, Weisfeiler–Leman (WL) isomorfizm testine denk bir ayırt edicilik gücü elde ederek düğüm sınıflandırmasında yüksek ifade kapasitesine ulaşır. ChebNet (Chebyshev Spectral CNN) ise GNN’lerin erken dönem spektral yaklaşımıdır; burada graf konvolüsyonu, Laplasyen özdeğer ayrışımı yerine Chebyshev polinomlarıyla yaklaşık olarak ifade edilir. Böylece yerelleştirilmiş filtreler elde edilir ve maliyetli özvektör hesaplaması ortadan kaldırılır.

Bu modeller, grafın doğasına göre (yoğun/seyrekle, homofili/heterofili, dinamik/statik) farklı avantajlar sağlar. GSA’ların temel avantajları:

- Topolojik bağımlılıkları yakalayarak ilişkisel öğrenme sağlar.
- Düşük etiketli veri durumlarında yarı-denetimli öğrenme için uygundur.
- Heterojen veri türlerini (ör. IP–port–protokol) tek bir grafik bütünlüğü içinde işleyebilir.
- Görselleştirilebilir ve açıklanabilir (örneğin GNNExplainer, PGExplainer, GraphLIME).

Bu avantajların yanı sıra birtakım zorlukları da beraberinde getirmektedir. Bunlar:

- Katman sayısı arttıkça düğüm temsilleri birbirine yakınsar (oversmoothing).
- Uzak bağımlılıkların bilgi sıkışması problem (over-squashing).
- Büyük graf verilerinde bellek ve zaman maliyeti yüksektir.

olarak ifade edilebilir.

5.6.1. GCN

Kipf & Welling [556] tarafından geliştirilen Graf Konvolüsyonel Ağlar (GCN) modeli, graf üzerindeki konvolüsyonun basitleştirilmiş formunu kullanmaktadır:

$$H^{(l+1)} = \sigma(\tilde{D}^{-1/2} \tilde{A} \tilde{D}^{-1/2} H^{(l)} W^{(l)}) \quad (5.11)$$

Burada, $\tilde{A} = A + I$ öz-döngüler eklenmiş bitişiklik matrisi ve $\tilde{D}$ karşılık gelen derece matrisini ifade etmektedir. Bu yapı, bilgi yayılımını (propagation) normalize ederek öğrenme stabilitesini artırmaktadır.

GCN'ler düğüm sınıflandırması, grafik sınıflandırması gibi görevlerde, homofili graf yapısına sahip, yani benzer düğümlerin birbirine bağlandığı durumlarda ve etiketli düğüm verilerinin olduğu ve grafik bağlantılarının anlamlı olduğu senaryolarda kullanılabilir. Hesaplama açısından nispeten basit ve verimli olmaları ve düğüm özelliklerini ve graf yapısını birlikte öğrenebilmeleri başlıca avantajlarıdır. İndüktif genelleme (yeni düğümler) yönünden sınırlı olabilmeleri, çok katman kullanıldığında tüm düğüm temsillerinin birbirine yaklaşması problemi yaşayabilmeleri ise başlıca dezavantajlarıdır. Ayrıca, GCN'lerde dikkat mekanizması yoktur ve tüm komşular eşit ağırlıklı olabilmektedir.

5.6.2. GraphSAGE

GraphSAGE, özellikle büyük ölçekli grafiklerde indüktif öğrenmeyi (yeni düğümler ve graf yapılarıyla genelleme) hedefleyen bir GNN çerçevesidir [557]. Modelin temel yeniliği, her düğüm için komşuluktan bilgi toplama ve bu bilgilere göre birleştirme fonksiyonları öğrenmesidir. Böylece doğrudan her düğüm için bir gömme vektörü değil, öğrenilmiş bir embedding fonksiyonu elde edilmektedir. GraphSAGE matematiksel olarak şu şekilde ifade edilir:

$$\begin{aligned} h_{\mathcal{N}(v)}^{(l)} &= AGGREGATE^{(l)} (\{h_u^{(l)}, u \in \mathcal{N}(v)\}), \\ h_v^{(l+1)} &= \sigma (W^{(l)} [h_v^{(l)} \parallel h_{\mathcal{N}(v)}^{(l)}]) \end{aligned} \quad (5.12)$$

Burada AGGREGATE fonksiyonu örnekleme ve havuzlamaya göre belirlenmekle birlikte ortalama, maksimum ya da LSTM-tabanlı olabilir.

GraphSAGE, dinamik graf yapılarında, sürekli yeni düğümlerin eklendiği veya çıkarıldığı senaryolarda, etiketlenmemiş düğümlere genelleme yapmak gerektiğinde ve büyük ölçekli grafiklerde örnekleme stratejilerinin önemli olduğu durumlarda kullanılabilir.

İndüktif genelleme yeteneği yani eğitim sırasında görülmemiş düğümler için embedding üretilmesi ve komşu örnekleme sayesinde ölçeklenebilirlik sağlaması başlıca avantajlarıdır. Bunun yanı sıra, örnekleme işleminde bazı bilgileri kaçırabilmesi, Aggregator fonksiyonunun seçiminin performansını ciddi etkilemesi ve GCN gibi topolojik yapının tüm detaylarını öğrenmede sınırlı olabilmesi başlıca dezavantajları olarak sıralanabilir.

5.6.3. GAT

Graph Attention Network (GAT) [558], düğüm komşuluklarından gelen mesajların ağırlıklandırılarak toplanmasını sağlayan dikkat (attention) mekanizması içeren bir GSA modelidir. GAT modelinde her düğüm, komşularının önem derecesini öğrenebilir; bu da heterojen komşuluklara sahip grafiklerde bilgiyi daha seçici kullanmayı sağlamaktadır. GCN’de tüm komşular eşit önemdeyken, GAT her komşuya öğrenilebilir bir α_{ij} ağırlık atar. Bu durum, özellikle heterojen veya yoğun graflarda esnek öğrenmeyi sağlamaktadır.

Bir GAT katmanının başlıca matematiksel ifadesi şu şekildedir:

$$\begin{aligned}
 z_i &= Wh_i, z_j = Wh_j \forall j \in \mathcal{N}(i) \\
 e_{ij} &= \text{LeakyReLU}(a^T [z_i \parallel z_j]) \\
 \alpha_{ij} &= \text{softmax}_j(e_{ij}) = \frac{\exp(e_{ij})}{\sum_{k \in \mathcal{N}(i)} \exp(e_{ik})} \\
 h'_i &= \sigma(\sum_{j \in \mathcal{N}(i)} \alpha_{ij} z_j)
 \end{aligned} \tag{5.13}$$

Burada W öğrenilir ağırlık matrisini, a dikkat mekanizması için öğrenilir vektörü temsil etmektedir. $\parallel$ ile vektörlerin birleştirilmesi yapılmaktadır. Ayrıca, bu matematiksel ifade çok başlıklı dikkat için aşağıdaki gibi genişletilebilmektedir:

$$h'_i = \parallel_{k=1}^K \sigma(\sum_{j \in \mathcal{N}(i)} \alpha_{ij}^k W^k h_j) \tag{5.14}$$

GAT modeli, komşu düğümler arasında eşit olmayan önemli/önemsiz ilişkilerin bulunduğu graf yapılarında, heterojen veri tipleri veya değişken bağlantı türleri olduğunda, indüktif öğrenme senaryolarında ve “hangi komşu daha önemli?” sorusunun yanıtının öğrenilmesi istendiğinde kullanışlı olmaktadır.

Komşulara farklı ağırlık verme yeteneği sayesinde daha esnek modelleme yapabilmesi ve indüktif ve transduktif öğrenmeye uygun olması başlıca avantajlarıdır. Bunun yanı sıra, dikkat katsayılarının öğrenilmesinin ek parametre getirmesi, dolayısıyla aşırı öğrenmeye veya hesaplama yüküne neden olabilmesi ve büyük graf-komşuluklarda dikkat hesaplamasının maliyetli olabilmesi başlıca dezavantajları olarak sıralanabilir.

5.6.4. GIN

Graph Isomorphism Network (GIN) [559] modeli, GSA modellerinin ayırt edicilik gücünü artırmayı amaçlayan bir mimaridir. GIN, klasik GSA modellerinin bazı graf yapıları arasındaki farklılıkları ayırt edemediğini teorik olarak göstererek bu sınırlamaların aşılmasını sağlamıştır. GIN modelinin temel matematiksel ifadesi şu şekildedir:

$$h_v^{(k)} = \text{MLP}^{(k)}\left((1 + \epsilon^{(k)}) h_v^{(k-1)} + \sum_{u \in \mathcal{N}(v)} h_u^{(k-1)}\right) \tag{5.15}$$

Burada $\in^{(k)}$ katman başına öğrenilen ya da sabit olabilir $h_v^{(0)} = x_v$ başlangıç düğüm özelliğini ifade etmektedir. Bu matematiksel ifade ile GIN, her düğümün önceki katmandaki kendi temsili ve komşuların toplamını birlikte kullanılmaktadır. Graf düzeyinde okuma için genellikle tüm düğüm temsilleri toplanır ya da katmanlar boyunca birleştirilme işlemi yapılmaktadır.

GIN modeli, grafik sınıflandırması, motif tanıma, kimyasal graf veri setleri gibi yapısal farklılıkların belirleyici olduğu görevlerde ve karmaşık topolojiye sahip graf yapılarında modelin ayırt edicilik gücünün önemli olduğu senaryolarda kullanışlı olabilmektedir.

Teorik olarak Weisfeiler–Leman test kadar güçlü olması ve topolojik yapıdan kaynaklanan örüntüleri klasik GSA'lara göre daha iyi öğrenebilmesi başlıca avantajlarıdır. Bunun yanı sıra, diğer modeller kadar yaygın ve optimizasyon açısından olgun olmamaları ve büyük graflarda toplama agregatörü nedeniyle hesaplama/bellek açısından daha maliyetli olmaları başlıca dezavantajlarıdır.

Önerilen STS modeli mimarisinde graf sinir ağı olarak GCN kullanılmıştır. Bunun yanı sıra, GIN, GAT ve GraphSAGE modelleri de çalıştırılmış ve performans değerleri karşılaştırılmıştır.

5.7. SHAP Algoritması

Lundberg ve Lee [424] tarafından yazılan SHAP (SHapley Additive exPlanations), bireysel tahminleri açıklamaya yönelik bir yöntemdir. Tüm olası özellik kombinasyonlarını ve bunların YZ modelinin tahminine katkılarını hesaplayarak her bir özelliğin önemini belirlemektedir. SHAP yöntemi, kooperatif oyun teorisi (cooperative game theory) kapsamında Lloyd Shapley [561] tarafından tanımlanan Shapley değerlerine dayanmaktadır. Temel olarak, SHAP, Shapley değerine dayalı hem global hem yerel ve model-agnostik bir açıklama yöntemidir. SHAP, herhangi bir makine öğrenmesi modelinin çıktısını açıklamaya yönelik oyun kuramsal ve birleşik bir çerçevedir.

Shapley değeri oyun teorisinden gelmektedir. Oyun teorisinde oyun, işbirlikçi bir yapıdadır. Oyun içerisindeki görev tahmindir, kazanç ise tahmin ile temel tahmin arasındaki mesafeyi

(genellikle gözlemlerin ortalaması) belirtmektedir. Oyundaki oyuncular ise özelliklerdir. Ardından, Shapley değeri, özellikler arasında tahmin kaymasını temel tahminden ayırmak için kullanılır. Dolayısıyla, her bir özelliğin gerçekleşmesi, tahminin olumlu ya da olumsuz bir varyasyonunu ifade etmektedir. Bir oyun ortamında her oyuncu (özellik) oyunun sonucuna (model çıktısı) belli bir katkı yapar. Amaç, her oyuncunun adil katkısını matematiksel olarak belirlemektir.

Bir özellik seti $N = \{1,2,3 \dots, n\}$ ve bir çıktı fonksiyonu $v(S)$ için Shapley değeri matematiksel olarak şu şekilde ifade edilir:

$$\phi_i(v) = \sum_{S \subseteq N \setminus \{i\}} \frac{|S|!(n-|S|-1)!}{n!} [v(S \cup \{i\}) - v(S)] \quad (5.16)$$

Burada:

$\phi_i(v)$: i. özelliğin katkısını,

$v(S)$: yalnızca S alt kümesiyle yapılan model tahminini ifade etmektedir.

$\frac{|S|!(n-|S|-1)!}{n!}$ kombinasyonel terimi ise adil ağırlıkların sağlanması için kullanılmaktadır.

Bu matematiksel tanım her özelliğin katkısının tüm olası iş birliği kombinasyonlarında ortalamasının alındığını ifade etmektedir.

SHAP'ın amacı, her bir özelliğin tahmine katkısını hesaplayarak bir x örneğinin tahminini açıklamaktır. SHAP'ın arkasındaki fikir, Shapley değer açıklamasını bir özellik ilişkilendirme yöntemi olarak ifade etmektedir [562]. Yani, bir modelin çıktısının özellik katkılarına ayrıştırılmasını sağlamaktadır. Matematiksel olarak şu şekilde ifade edilmektedir:

$$f(x) = \phi_0 + \sum_{i=1}^M \phi_i \quad (5.17)$$

Burada, $f(x)$ model tahminini, ϕ_0 ortalama model çıktısını (baseline), ϕ_i ise her özelliğin tahmine katkısını (Shapley değeri) ifade etmektedir.

SHAP, Shapley değerlerini YZ modellerine uyarlayan bir açıklama çerçevesidir. Shapley değerlerini hesapladığından, Shapley değerlerinin tüm avantajları SHAP için de

geçerlidir[74]. SHAP değerinin önemli bir avantajı, şeffaflığı ve yerel olarak yorumlanabilirliğidir; bu, her gözlemin kendi SHAP değer kümesine sahip olabileceği anlamına gelir. Bir örneğin neden tahminini aldığını ve tahmin edicilerinin katkılarını açıklamak mümkündür [563]. Shapley değeri, teorik çıkarımlarla açıklama dâhil olmak üzere yerel ve küresel yorumlama sağlamaktadır. SHAP hızlı, güvenilir değerlendirmeler sağlamaktadır. Bu yöntem özellikle ağaç tabanlı modeller için hızlı bir uygulamaya sahiptir. Bu durum SHAP'ın popüleritesinin artmasında önemli rol oynamaktadır, çünkü Shapley değerlerinin benimsenmesinin önündeki en büyük engel, yavaş hesaplama. Bu avantajların yanı sıra, SHAP, LIME ve Shapley değerlerini birbirine bağlamaktadır. Bu, her iki yöntemi de daha iyi anlamak için çok yararlıdır. Ayrıca, yorumlanabilir makine öğrenimi alanını birleştirmeye yardımcı olur [564].

SHAP algoritmasının temel çalışma adımları özetle şu şekildedir:

1. Model belirleme aşamasında LightGBM, Random Forest, GCN gibi bir $f(x)$ tahmin fonksiyonu seçilmektedir.
2. Referans (baseline) tanımlama aşamasında modelin ortalama çıktısı veya referans girdisi belirlenmektedir. Bu aşamada modelin tüm özellikler devre dışı bırakıldığında yani hiç bilgi verilmediğinde yaptığı tahmin olan referans değeri hesaplanır.
3. Öznitelik alt kümelerinin belirlenmesi aşamasında tüm olası özellik alt kümeleri oluşturulmaktadır. Her bir i özelliği için modelin o özelliği dâhil edip etmemesine göre çıktısındaki değişim ölçülmektedir.
4. Önceki aşamadaki çıktılara göre her özelliğin marjinal katkısı hesaplanmaktadır.
5. Her özellik için tüm alt kümelerdeki katkılar Shapley ağırlıklarıyla çarpılarak ortalaması alınmaktadır. Bu adım, SHAP algoritmasının adil katkı dağılımı prensibini (Shapley axioms) garanti eden kısımdır.
6. Bu aşamada SHAP algoritması KernelSHAP, TreeSHAP, DeepSHAP gibi model türüne göre farklı hesaplama yaklaşımlarını benimser.
7. SHAP değerleri normalize edilir ve birleştirilir. Tüm özelliklerin katkı değerleri toplandığında model çıktısındaki fark tam olarak ortaya çıkmaktadır. Bu durum, SHAP additivity özelliği olarak bilinmektedir.
8. Görselleştirme aşamasında ise özellik katkıları genellikle summary plot, waterfall plot, force plot gibi biçimlerde sunulmaktadır. SHAP algoritması ile hem yerel hem global açıklamalar üretilebilmektedir.

Shapley değerlerinin dezavantajları SHAP için de geçerlidir. Shapley değerleri yanlış yorumlanabilir ve yeni veriler için bunları hesaplamak için verilere erişim gerekmektedir. SHAP ile önyargıları gizleyebilen kasıtlı olarak yanıltıcı yorumlar oluşturmak mümkündür. Bu durum açıklamalara olan güveni zedelemektedir.

5.8. GNNExplainer Algoritması

GNNExplainer, Ying ve arkadaşları [565] tarafından önerilen, GSA modellerinin kararlarını açıklamak üzere geliştirilmiş bir model-agnostik açıklama yöntemidir. Temel mantık, bir GSA'nın belirli bir tahmin üretirken hangi alt-graf yapısına ve hangi özelliklere dayandığını ortaya çıkarmaktır. Yani, bir GSA modeli verildiğinde GNNExplainer; model bu tahmini yaparken grafin hangi kısmı ve hangi öznitelikler kritik rol oynadı? sorusuna yanıt aramaktadır. Bu yaklaşım, SHAP veya LIME gibi geleneksel AYZ tekniklerinden farklı olarak grafin topolojik doğasını korumaktadır. Böylece, üretilen açıklamalar, yalnızca özellik düzeyinde değil kenar ve düğüm ilişkileri düzeyinde de yapılmaktadır.

Bir GSA modelinin tahminini $\hat{y} = f(G, X)$ olarak varsayarsak GNNExplainer bu tahminin arkasındaki en açıklayıcı altgraf olan $G_S = (V_S, E_S)$ 'yi ve öznitelik alt kümesi olan X_S 'yi bulmayı hedeflemektedir. GNNExplainer, açıklamayı bilgi teorisine dayalı bir optimizasyon problemi olarak formüle etmektedir. Amaç, orijinal modelin çıktısını maksimum oranda korurken, minimum bilgiyle açıklama sağlamaktır. GNNExplainer yönteminin optimizasyon problemi ve kayıp fonksiyonu temel olarak şu şekilde ifade edilmektedir:

$$\max_{G_S, X_S} I(Y; G_S, X_S) \quad (5.18)$$

Eşitlik 5.18'de belirtilen optimizasyon probleminde $I(Y; G_S, X_S)$ hedef değişkeni, Y altgraf/özellik altkümesi arasındaki ortak bilgi değerini ifade etmektedir. Buradaki amaç G_S ve X_S seçimlerinin model çıktısı hakkında mümkün olduğunca fazla bilgi taşımasıdır. Bununla birlikte, G_S doğrudan bir maske vektörü (kenar maskesi) $M_E \in [0,1]^{|E|}$ ile temsil edilirken X_S ise bir öznitelik maskesi $M_F \in [0,1]^d$ ile temsil edilmektedir. GNNExplainer bu maskeleri öğrenerek hangi kenarların ve özelliklerin tahmin üzerinde ne kadar etkili olduğunu belirlemektedir.

Kayıp fonksiyonunun hesaplanmasında ise modelin orijinal tahmini $\hat{y} = f(G.X)$ ile açıklayıcı altgraf tahmini $\hat{y}' = f(G \odot M_E, X \odot M_F)$ ile karşılaştırılmaktadır. Buradaki amaç $\hat{y}$ ile $\hat{y}'$ arasındaki bilgi farkını minimize etmektir. Bunun için:

$$\mathcal{L}_{expl} = -\mathbb{E}_{G_S}[log P_\theta(Y | G_S, X_S)] + \alpha \|M_E\|_1 + \beta H(M_E) \quad (5.19)$$

Burada, $P_\theta(Y | G_S, X_S)$ açıklayıcı altgrafın model çıktısını koruma olasılığını ifade etmekte, $\|M_E\|_1$ maskenin seyrekliğini (sparsity) teşvik etmekte, $H(M_E)$ entropi düzenleyerek maskeyi ikili değerlere yakınlaştırır ve α ve β ise düzenleme katsayılarını temsil etmektedir. Sonuç olarak M_E ve M_F maskeleri optimize edilerek, model kararını en iyi açıklayan altgraf G_S elde edilmektedir.

GNNExplainer algoritmasının temel çalışma adımları özetle şu şekildedir:

- (1) Hedef tahmin belirlenmesi aşamasında açıklanacak düğüm, kenar veya graf etiketi seçilmektedir.
- (2) Maske parametrelerinin başlatılması aşamasında rastgele başlatılan M_E ve M_F maske tensörleri oluşturulmaktadır.
- (3) GSA modeli üzerine ileri yayılım aşamasında maskelenmiş graf $(A \odot M_E, X \odot M_F)$ ile model çalıştırılır.
- (4) Modelin orijinal tahmini ile maskeli tahmin arasındaki fark ve düzenleme terimleri hesaplanır. Yani, kayıp hesabı yapılır.
- (5) Geri yayılım aşamasında maskeler optimize edilir.
- (6) Altgraf ve öznetelik çıkarım aşamasında $M_E > \text{eşik değeri}$ sahip kenarlar seçilerek açıklayıcı altgraf olan G_S oluşturulur.
- (7) Elde edilen altgraf genellikle renk ve kalınlıklarla gösterilerek görselleştirilmiş olur.

GNNExplainer, diğer AYZ yöntemlerinden farklı olarak, kararın yalnızca özellik değil graf yapısıyla da ilişkisini açıklamaktadır. Her tür GSA mimarisiyle (GCN, GAT, GIN, GraphSAGE vb.) uyumlu çalışabilen model-agnostik bir çözümdür. GNNExplainer ile tek bir düğümün veya tüm ağıın davranışı açıklanabilmektedir. Böylece, hem yerel hem de global açıklamalar sağlayabilme yeteneğine sahiptir. Altgraf temelli açıklamalar özellikle siber güvenlik gibi kritik alanlar için sezgisel açıklamalar sağlar. GNNExplainer yönteminin bu

avantajlarının yanı sıra, özellikle büyük graflarda maske optimizasyonunun yavaş olabilmesi, farklı rastgele başlatmalarda farklı altgrafların oluşabilmesi ve modeldeki genelleme hatalarını düzeltmemesi gibi dezavantajları da bulunmaktadır [566].

Güncel saldırı senaryolarında davranışlar çoğu zaman ilişkisel bir yapı sergilemektedir. Bu nedenle ağ trafiğinin iletişim grafi biçiminde modellenmesi ve GNN tabanlı STS'lerin kullanımı literatürde giderek yaygınlaşmaya başlamıştır. Bununla birlikte, GNN tabanlı STS'ler, “kara kutu” problemi nedeniyle güvenilmez kalmaktadır. Bu noktada Graf-AYZ yaklaşımları, alarmın “neden” üretildiğini kanıtlayabilir şekilde sunarak STS çıktılarının güvenilirliğini artırmayı hedefler. GNNExplainer, klasik özellik önemi yaklaşımının ötesine geçerek topoloji + özellik birlikteliğini açıklayabilmesiyle STS'ler için önemli bir bileşen haline gelmektedir. GNNExplainer'ın alt-graf çıktısı; tarama ve yayılım gibi saldırıların doğal graf motiflerini görünür kılarak adli ve olay analiz süreçlerini hızlandırabilmektedir.

5.9. Zeek Açık Kaynak Saldırı Tespit Sistemi

Zeek (eski adıyla Bro), ağ trafiğini derinlemesine analiz eden açık kaynak bir ağ izleme ve saldırı tespit sistemidir. İlk olarak Vern Paxson tarafından 1995 yılında geliştirilmiş olup, modern ağ güvenliğinde sadece bir STS değil aynı zamanda kapsamlı bir ağ davranış analizi ve olay korelasyon platformu olarak konumlanmaktadır [567].

Zeek'in temel amacı yalnızca saldırı tespiti yapmak değil, aynı zamanda ağda gerçekleşen tüm olayları anlamlandırmak ve detaylı kayıt altına almaktır. Bu özellik, sistemin hem adli analiz hem de siber tehdit istihbaratı süreçlerinde etkin biçimde kullanılmasını sağlamaktadır [568]. Zeek, çok katmanlı bir analiz mimarisine sahiptir ve ağ trafiğini uygulama katmanına kadar ayırıştırabilir. Bu sayede, yalnızca IP veya TCP düzeyinde değil, HTTP, DNS, SSL/TLS, FTP, SSH, SMTP gibi protokollerde de davranış analizi yapılabilir [568].

Zeek, protokol farkındalığı yüksek bir sistemdir. Her bir bağlantı oturumu, ilgili protokolün kuralları çerçevesinde ayrıştırılır. Zeek Script dili, belirli olayların (örneğin bir HTTP isteği, TLS handshake veya DNS sorgusu) gerçekleşmesiyle tetiklenir ve kullanıcıya özel aksiyonların tanımlanmasına olanak tanır [567]. Zeek, her protokol ve olay türü için ayrı log dosyaları oluşturur (conn.log, http.log, ssl.log, notice.log vb.). Bu kayıtlar hem JSON hem ASCII formatında tutulabilir [568]. Zeek Package Manager (ZKG) aracılığıyla yüzlerce

eklenti (ör. JA3/JA4 TLS fingerprinting, ICSNPP endüstriyel protokol analizleri, HTTP saldırı modülleri) kolayca yüklenebilir [569]. Zeek yalnızca imza tabanlı değil, aynı zamanda davranışsal ve istatistiksel yöntemlerle de anormallikleri tespit eder [570].

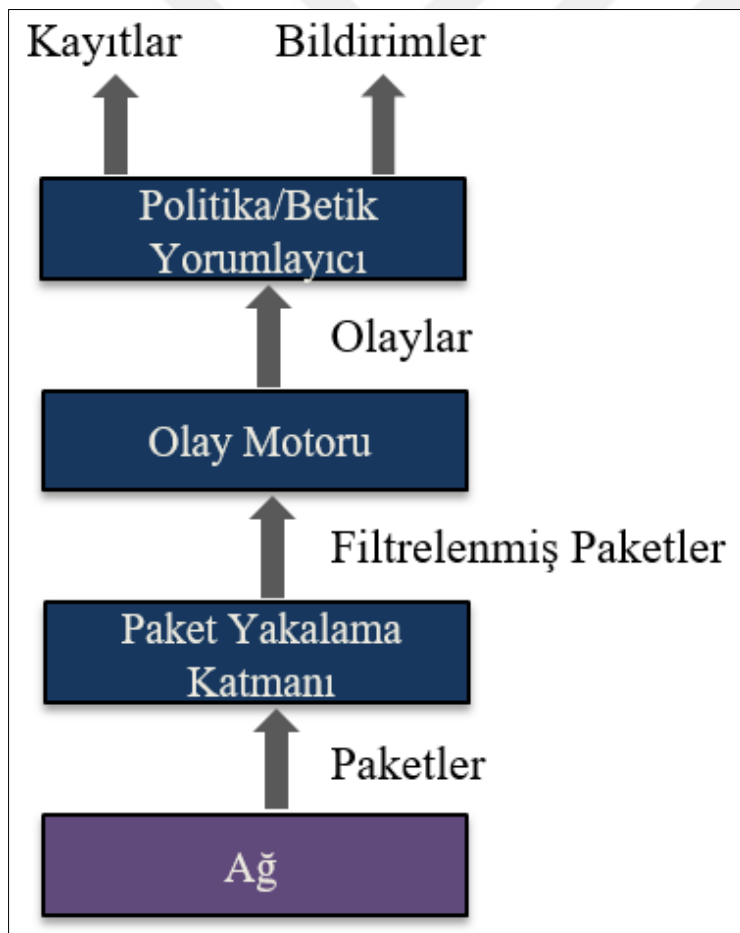
Temel Zeek mimarisi Şekil 5.5’de gösterildiği modüler, olay tabanlı ve dağıtık bir yapıya sahiptir [568]. Paket Yakalama Katmanı (Packet Capture Layer) ağ ara yüzünden (ör. AF_Packet, PCAP, Netmap) gelen trafiği yakalar ve üst katmanlara iletir. Buradaki görev, trafiği minimum kayıpla Olay Motoru’na aktarmaktır. Olay Motoru (Event Engine), paketleri protokollere göre ayrıştırarak olaylara dönüştürür. Örneğin, yeni bir TCP bağlantısı veya DNS isteği birer olay olarak tanımlanmaktadır. Zeek Script dili, bu olaylara tepki olarak güvenlik politikalarını çalıştırmaktadır. Örneğin, çok sayıda başarısız SSH girişimi tespit edilirse bir “notice” oluşturulabilir [567]. Bu betiklerin çalıştırılması işlemlerini Betik Yorumlayıcı (Script Interpreter) bileşeni yapmaktadır. Günlük Çerçevesi (Logging Framework) olayların, bağlantıların ve tespitlerin kayıt altına alındığı yapı olup LogAscii veya JSON formatında çıktı üretebilmektedir. Dağıtık yapılarda yer alan Haberleşme Katmanında (Connection Layer) ise, farklı sensörler arasında veri senkronizasyonu sağlanmaktadır. Bu özellik, yüksek trafiğe sahip ağlarda ölçeklenebilirliği mümkün kılmaktadır [568].

Zeek’in başlıca avantajları şu şekildedir:

- Derin paket analizi ile Katman 7’ye kadar inceleme imkânı sunmaktadır [568].
- Zeek Script dili sayesinde ağ politikaları kullanıcıya özel yazılabilmektedir [567].
- DNS, HTTP, SSL, SMTP, FTP gibi birçok protokol için detaylı log oluşturarak kapsamlı bir loglama mekanizması sunmaktadır [568].
- Çok çekirdekli sistemlerde paralel analiz desteği sunmaktadır.
- Zeek çıktıları Splunk, ELK veya Kafka gibi sistemlerle kolaylıkla entegre edilebilmektedir [571].
- ICSNPP eklentileri sayesinde Modbus, DNP3, BACnet gibi protokoller analiz edilebilmektedir [569].
- Corelight ve Zeek topluluğu tarafından sürekli güncellenmektedir.

Zeek’in başlıca dezavantajları ise şu şekilde sıralanabilir:

- Snort veya Suricata gibi doğrudan imza tabanlı sistemlerden farklı olarak, Zeek davranışsal tespit yapar ve bu bazı durumlarda belirli saldırı tiplerini kaçırabilmektedir [570].
- Zeek Script dili karmaşıktır ve etkin politika yazımı deneyim ve uzmanlık gerektirmektedir.
- Zeek trafik üzerinde yalnızca gözlem ve inceleme yapmaktadır, trafik üzerinde engelleme veya aktif müdahale yapmamaktadır.
- Özellikle yüksek hacimli ağlarda kullanılması durumunda detaylı loglama nedeniyle yüksek disk alanı kullanabilmektedir.
- Dağıtık mimari uygulanan büyük ağ altyapılarında önemli bir yönetim yükü oluşturabilmektedir.



Şekil 5.5. Temel Zeek mimarisi

Bu tez çalışması kapsamında Zeek açık kaynak STS sistemine CIC-IDS2017 ve UNSW-NB15 veri kümesinin pcap dosyaları girdi olarak verilmiş ve elde edilen sonuçlar

değerlendirilmiştir. Saldırıların tespit edilebilmesi amacıyla Zeek üzerinde ön tanımlı bazı politikalar varsayılan olarak gelmektedir. Bununla birlikte daha iyi bir saldırı tespiti için local.zeek dosyasına yüklenen paketler Çizelge 5.7’de verilmiştir.

Bu aşamada gerçek dünya ortamlarında kullanılabilen bir açık kaynak STS sisteminin ağ akış verileri karşısındaki davranışı ve tespit süreçleri analiz edilmiştir.

Çizelge 5.7. Zeek kurulum yapılan politika ve paketler

Satır / Paket	Tür	İşlev
/opt/zeek/share/zeek/site/compat_log_reporter.zeek	Compat shim / araç	Eski/uyumsuz log formatlarını veya eski paketleri yeni Zeek sürümüne göre uyarlayan yardımcı betik; log uyumluluğu/raporlama iyileştirmesi sağlar.
@load frameworks/notice	Framework	Zeek’in olay/uyarı (notice) altyapısını yükler; diğer policy’lerin notice üretmesini yönetir.
@load policy/protocols/conn/community-id-logging	Policy	Akışlar için Community ID (flow hash) üretir; farklı sensör/araçlar arasında aynı akışı ilişkilendirmek için.
redef LogAscii::use_json = T;	Konfigürasyon (redef)	Tüm Zeek ASCII loglarını JSON formatında yazmaya zorlar.
redef LogAscii::json_timestamps = JSON::TS_ISO8601;	Konfigürasyon (redef)	JSON zaman damgası formatını ISO8601 olarak ayarlar.
@load policy/protocols/ssl/heartbleed	Policy (SSL)	Heartbleed (CVE-2014-0160) benzeri TLS heartbeat istismarlarını tespit etmeye çalışır.
@load policy/protocols/ssh/detect-bruteforcing	Policy (SSH)	SSH için brute-force/çoklu başarısız oturum denemelerini tespit eder.
@load policy/protocols/ftp/detect-bruteforcing	Policy (FTP)	FTP brute-force denemelerini işaretler.
@load policy/protocols/http/detect-sql-injection	Policy (HTTP)	HTTP isteklerinde basit SQLi imzaları/heuristikleri ile şüpheli isteği tespit eder.
@load frameworks/software/vulnerable	Framework (software)	Yazılım envanterine/binary signature’larına göre bilinen zafiyetlere karşı tespit/işaret yapmaya yardımcı olur.
@load frameworks/software/version-changes	Framework	Bir varlıkta (host) yazılım versiyonlarında değişiklikleri izler — sürüm değişimi algılama.
@load frameworks/files/detect-MHR	Framework (files)	Malicious / high-risk file (MHR) tespiti için dosya analizi kuralları sağlar (imza/heuristic).
@load packages/ja3/ja3	Paket (TLS fingerprint)	TLS istemci JA3 parmak izlerini oluşturur — istemci TLS el sıkışma fingerprinting.

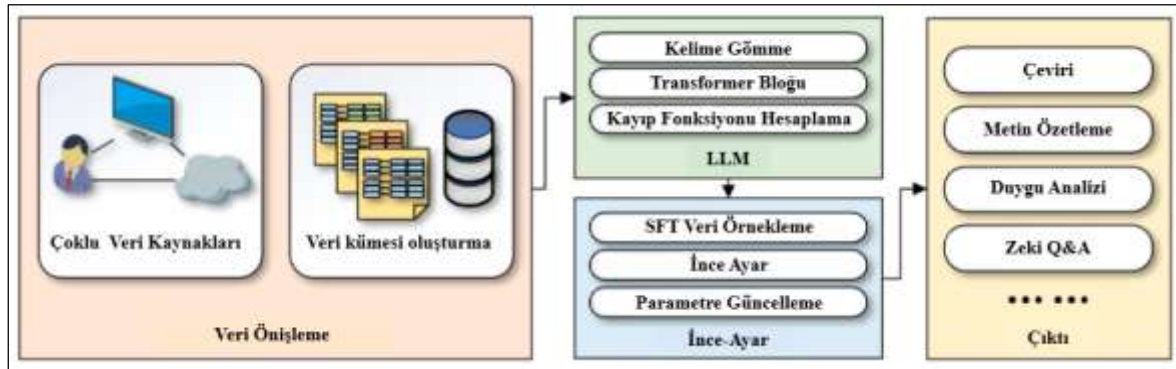
Çizelge 5.7. (devam) Zeek kurulum yapılan politika ve paketler

@load packages/hassh/hassh	Paket (SSH fingerprint)	SSH istemci fingerprint'i (HASSH) üretir — SSH client behavior fingerprinting.
@load packages/zeek-httpattacks/main	Paket (HTTP attack detection)	HTTP'ye yönelik saldırı/pattern tespit kuralları sağlar (ör. scanning, exploit payloads).
@load packages/log-add-http-post-bodies/main	Paket (HTTP logging)	HTTP POST gövdelerini (ör. form-data, JSON body) loglara ekler / ya da ayrı log tutar.
@load packages/bzar/main	Paket (BZAR)	BZAR paketine ait tarama/tehdit analiz paketi.
@load packages/bro-simple-scan/scan	Paket (scan detection)	Basit port/host tarama aktivitelerini tespit eden kurallar sağlar.
@load /opt/zeek/var/lib/zkg/clones/package/icmp-scans.git/scripts/icmp-scans.zeek	Custom / clone script	ICMP tabanlı tarama / ICMP scan patternlerini tespit eden bir script (zkg clone yolundan yüklenmiş).
@load packages/bro-is-darknet/main	Paket (darknet detection)	Darknet / honeypot tarzı adreslere yönelik gelen istekleri tespit eder; darknet mapping.
@load packages/ftp-bruteforce/ftp-bruteforce	Paket (FTP brute-force)	FTP brute-force patternlerini daha kapsamlı tespit eden paket (policy/thresholds).
@load packages/RDP-bruteforce/rdp-bruteforce	Paket (RDP brute-force)	RDP üzerinden brute-force (özellikle Windows RDP) denemelerini tespit eder.
@load packages/zeek-log-all-http-headers/main	Paket (HTTP headers logging)	HTTP istek/cevaplarındaki tüm header alanlarını loglara ekler (standart header'ların dışında kalanlar dahil).
@load packages/ja4/main	Paket (JA4)	JA4 (TLS server-side veya yeni fingerprint standardı) yüklemesi

5.10. BDM Tabanlı Açıklama

Bu tez çalışması kapsamında, önerilen hibrit STS modelimizde elde edilen çıktıların yorumlanması sürecinde güncel teknoloji olan büyük dil modelleri (BDM) üzerinden oluşturulan doğal dil açıklamaları da kullanılmıştır. Son yıllarda BDM'lere olan ilgi giderek artarak büyük dil modellerinin yetenekleri sürekli olarak geliştirilmiştir. Parametre ölçekleri hızlı bir şekilde artarak model girdi ve çıktısının modları daha zengin hale gelmiştir. Bu durum, görüntü büyük modelleri, ses büyük modelleri, kod büyük modelleri ve bilimsel hesaplama büyük modelleri gibi çok modlu büyük modellerin ortaya çıkmasına yol açmıştır [576]. Literatürde genellikle belirli ortaya çıkan yeteneklere sahip, “ön eğitim-ince ayar” paradigmasına dayalı derin sinir ağı modelleri büyük modeller (LM) olarak adlandırmaktadır [576] ve bu tür modellerin oluşturulmasının temel süreci Şekil 5.6'da gösterilmiştir. Büyük

dil modelleri, son yıllarda hızlı teknolojik yinelemeler ve kapsamlı pratik uygulamalarla teknolojik bir yükseliş dönemindedir. BDM üzerinde giderek daha fazla çalışma yapılmaktadır. Bu kapsamda, bu tez çalışması kapsamında güncel teknolojilerin de entegre edilmesi amacıyla OpenAI tarafından geliştirilen ChatGPT ve Google tarafından geliştirilen Gemini BDM'leri yorumlama katmanı olarak önerilen sisteme eklenmiş ve analistlere sistemin kararlarının daha anlaşılır bir şekilde sunulması amaçlanmıştır.



Şekil 5.6. Temel BDM akış diyagramı [576] nolu kaynaktan alıntılanarak Türkçeleştirilmiştir.

OpenAI'nin GPT serisi modelleri, metin ve çoğu zaman görsel ya da çok modelli girdilerden anlam çıkarabilen büyük ölçekli dil modelleridir [572]. ChatGPT, OpenAI tarafından geliştirilen bir generatif büyük dil modeli tabanlı sohbet/yardım arayüzüdür. Başlangıç olarak 2022 yılının Kasım ayında piyasaya sürülmüştür [572]. Temel olarak ön-eğitilmiş (pre-trained) bir “Generative Pre-trained Transformer (GPT)” mimarisine dayanır ve ardından sohbet odaklı görevlere (fine-tuning) ve insan geri bildirimiyle pekiştirilmiş öğrenmeye (Reinforcement Learning from Human Feedback — RLHF) tabi tutulmuştur. ChatGPT'nin kullandığı GPT serisi modelleri sürekli güncellenmektedir. ChatGPT gibi modellerde mimari ayrıntılar (tam parametre sayısı gibi) genellikle kamuya tam açıklanmamaktadır; ancak modelin çok büyük ölçekte veriyle ön-eğitildiği, insan geri bildirimiyle finetune edildiği ve güvenlik/etik kontrollerle desteklendiği belirtilmektedir [573]. ChatGPT AYZ bağlamında kullanıcıya teknik olmayan düzeyde “modelin ne yaptığı” konusunda doğal dil yardımı verebilir; bu noktada bir “yorumlama katmanı” olarak işlev görür.

Gemini ise Google DeepMind tarafından geliştirilen ve multimodal yeteneklere sahip bir BDM ailesidir [574]. Multimodal yetenekleri ön plandadır; yani metin, görüntü, ses, kod

gibi farklı modaliteleri işleyebilir. Gemini ailesi Nano, Pro, Ultra gibi çeşitli boyutlarda sunulmaktadır. Gemini modellerinin “reasoning” (akıl yürütme) kapasitesi, “thinking” modları ve gelişmiş paralel düşünme teknikleri (parallel reasoning) gibi özelliklerle donatıldığı belirtilmektedir [575]. Gemini’nin AYZ bağlamında multimodal ve uzun-bağlam yetenekleri ile açıklanabilirlik katmanı içinde daha zengin girdiler sunma imkânı sağlamaktadır. Bu bağlamda, Gemini bir modelin çıktısını yalnızca teknik düzeyde değil, aynı zamanda farklı grafik çıktısı, kod parçacığı ve metin açıklama gibi modaliteler içinden yorumlama kapasitesine de sahiptir.

Çalışma kapsamında BDM’ler birincil tespit bileşenleri olarak değil, yüksek seviyeli yorumlanabilirlik ve karar destek katmanı olarak önerilen sisteme entegre edilmiştir. Bu çalışmada, saldırı tespit kararlarının kullanılabilirliğini, şeffaflığını ve analist yorumlanabilirliğini artırmak için ChatGPT (GPT-5) ve Gemini 2.5 Pro modelleri kullanılmıştır. Açıklanabilirliğin ham sayısal çıktılar veya statik görselleştirmelerle sınırlı olduğu geleneksel STS mimarilerinin aksine, önerilen sistemde BDM’ler çok boyutlu açıklanabilirlik çıktılarının semantik yorumlayıcıları olarak eklenmiştir. BDM’ler model eğitime veya tahminine katılmamakla birlikte, SHAP ve GNNExplainer yöntemleri tarafından üretilen açıklanabilirlik çıktılarını alarak sonradan çalışmaktadırlar. Böylece, açıklanabilirlik çıktıları insan merkezli açıklamalara dönüştürülmektedir.

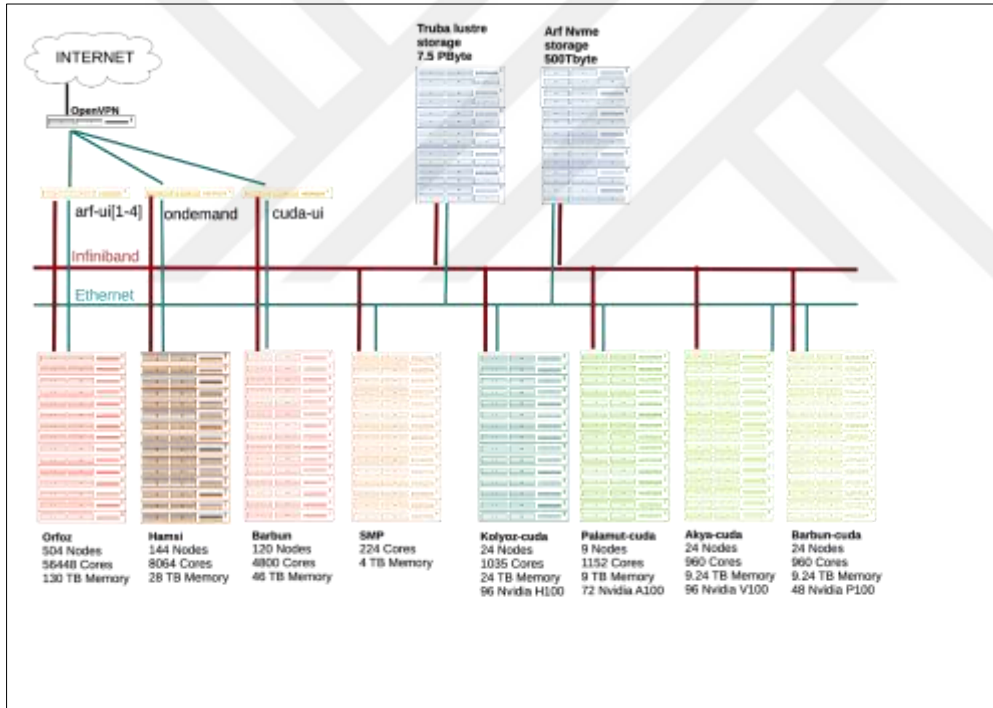
Bu BDM bağlantıları ile GNNExplainer ve SHAP AYZ yöntemlerinden çıkan alt graflar, özellik önemi değerleri, şelale (waterfall) ve force görsellerinin BDM’ye girdi olarak verilmesi sistemin karar mekanizmasının analistlere açık biçimde sunulmasını sağlamıştır. İki farklı BDM’nin (ChatGPT GPT-5 ve Gemini 2.5 Pro) kullanımı, modeller arası tutarlılık analizini mümkün kılarak tek model yanılsamaları riskini azaltmakta ve açıklama sağlamlığını artırmaktadır.

5.11. TRUBA Platformu

Türk Ulusal Bilim e-Altyapısı (TRUBA), 180’den fazla araştırma kurumundan 6000’in üzerinde araştırmacıya yüksek performanslı hesaplama ve depolama kaynakları sunan, ulusal düzeyde stratejik öneme sahip bir araştırma e-altyapısıdır. TRUBA, bilimsel çalışmalara doğrudan katkı sağlamanın yanı sıra, çeşitli kamu kurumları ve araştırma enstitüleriyle ortak projeler yürütmekte ve altyapının faaliyete geçmesinden bu yana 27

Avrupa Birliği Çerçeve Programı projesinde aktif olarak rol üstlenmiştir. Yüksek başarılı hesaplama, veri yoğun hesaplama, bilimsel veri ambarları ve bulut hesaplama gibi servisler TÜBİTAK ULAKBİM Yüksek Başarılı Hesaplama Grubu tarafından işletimi yapılmakta olan ve 2010 yılı sonuna kadar TR-Grid, 2011 itibariyle TÜBİTAK Bilim Kurulu kararı ile TRUBA ismi ile anılmaktadır [541].

Şekil 5.7’de gösterildiği gibi yüksek hesaplama ve depolama gücüne sahip olan TRUBA, 80.000’den fazla işlemci çekirdeği, 312 adet grafik işlemci (NVIDIA P100,V100,A100 ve H100), 7.5 PByte Lustre veri depolama ve 500 TByte yüksek hızlı merkezi NVMe tabanlı bir dosya sistemi alanıyla ülkemizde hesaplamalı bilimler alanında çalışan araştırmacılara ve AR-GE faaliyetleri sürdüren kurumlara hizmet vermektedir.



Şekil 5.7. TRUBA mimarisi [541]

Bu tez çalışmasında uzun süreli deneysel çalışmalar TÜBİTAK ULAKBİM, Yüksek Başarım ve Grid Hesaplama Merkezi’nde (TRUBA kaynaklarında) gerçekleştirilmiştir. Gerekli üyelik aşamalarının tamamlanmasının ardından deneysel çalışmalar sırasında ihtiyaç duyulan CPU ihtiyacı için orfoz kuyrukları, GPU ihtiyacı için ise akya-cuda kuyrukları kullanılmıştır. Kuyruklara gönderilen işler için oluşturulan SLURM betik dosyalarından örnekler Resim 5.1 ve Resim 5.2’de sunulmuştur.

```
#!/bin/bash

#SBATCH --account=samedal
#SBATCH --output=UNSW_LIGHTGBM_GCN.out
#SBATCH --error=UNSW_LIGHTGBM_GCN.err
#SBATCH --time=03-00:00:00
#SBATCH --job-name=UNSW_LIGHTGBM_GCN

#SBATCH --partition=akya-cuda
#SBATCH --ntasks=20
#SBATCH --nodes=1
#SBATCH --cpus-per-task=1
#SBATCH --gres=gpu:2

### Load modules
module purge
module load comp/gcc/12.3.0
module load apps/truba-ai/gpu-2024.0
hostname

which python
python unsw_lightgbm_gcn.py

scontrol show job $SLURM_JOB_ID >> UNSW_LIGHTGBM_GCN.info
exit
```

Resim 5.1. Akya-cuda kuyruğu için örnek SLURM betiği

5.12. Google Colab Platformu

Google Colab, Google Research tarafından sunulan, web tarayıcısı üzerinden erişilebilen ve “Jupyter Notebook” ortamına benzer şekilde Python kodu çalıştırılmasına imkân veren bulut tabanlı bir hizmettir [542]. Kurulum gerektirmemesi, doğrudan tarayıcıdan çalışabilmesi ve hesap ile eşleştirilebilen bir ortam olması sayesinde özellikle veri bilimi, makine öğrenmesi ve derin öğrenme uygulamalarında yoğun şekilde tercih edilmektedir [543].

```
#!/bin/bash

#SBATCH --account=samedal
#SBATCH --output=UNSW_CNN.out
#SBATCH --error=UNSW_CNN.err
#SBATCH --time=03-00:00:00
#SBATCH --job-name=UNSW_CNN

#SBATCH -p orfoz
#SBATCH -c 110
#SBATCH -C weka
#SBATCH -N 1
#SBATCH -n 1

### Load modules
module purge
module load apps/truba-ai/cpu-2024.0
hostname

which python
python unsw_cnn.py

scontrol show job $SLURM_JOB_ID >> UNSW_CNN.info
exit
```

Resim 5.2. Orfoz kuyruğu için örnek SLURM betiği

Platform, ücretsiz katmanda GPU ve TPU hızlandırıcıları sunmakta; ayrıca ücretli yükseltmeler (Pro, Pro+, Enterprise) yoluyla daha yüksek donanım ve kaynak esnekliği sağlamaktadır [544].

Bu tez çalışmasında deneysel çalışmalar için Pro+ üyeliği temin edilerek özellikle veri ön işleme aşamalarında Colab platformu kullanılmıştır. Bununla birlikte, kurulum ihtiyacı olmaması nedeniyle kolay erişilebilirlik sağlayarak çalışmalarda ortam bağımsız olarak süreklilik sağlamıştır. Deneysel çalışmalar sırasında Google tarafından sunulan 50 GB RAM, 235 GB disk kapasitesine sahip T4 GPU donanımları kullanılmıştır.

5.13. Değerlendirme Metrikleri

Karmaşıklık (Confusion) matrisi, bir sınıflandırıcı tarafından yapılan doğru ve yanlış tahminlerin sayısının tablo şeklinde bir özetidir. Bir sınıflandırma modelinin performansını

ölçmek için kullanılmaktadır. Karmaşıklık matrisleri, bir modelin performansı hakkında sınıflandırma doğruluğundan daha iyi fikir verdikleri için yaygın olarak kullanılmaktadır. Tüm sınıflar için doğru ve yanlış sınıflandırılmış örnekleri gösterdiği için sınıflandırıcıların performansı hakkında daha iyi bilgi vermektedir. Şekil 5.8’de karmaşıklık matrisi ve parametreleri gösterilmektedir.

Gerçek değer	Tahmin edilen değer	
	Atak	Normal
Atak	TP	FN
Normal	FP	TN

Şekil 5.8. Karmaşıklık matris sonuçları

Şekilde belirtilen parametreleri saldırı tespit sistemleri (STS) özelinde belirtecek olursak özetle şu şekildedir;

True Positive (TP – Doğru Pozitif) : STS tarafından başarılı bir şekilde tespit edilen saldırıları ifade etmektedir.

False Positive (FP – Yanlış Pozitif) : STS tarafından yanlışlıkla saldırı olarak sınıflandırılan normal (saldırı olmayan) davranışları ifade etmektedir.

False Negative (FN – Yanlış Negatif) : STS tarafından yanlışlıkla normal trafik olarak tespit edilen saldırıları ifade etmektedir.

True Negative (TN – Doğru Negatif) : STS tarafından normal trafiklerin başarılı bir şekilde normal trafik olarak tespit edilmesini ifade etmektedir.

Verilen bilgiler doğrultusunda tüm olayların sayısının yani, $TP+FP+FN+TN = 100$ olması gerekmektedir.

Sonuçların değerlendirilmesinde doğruluk, hassasiyet, F-skoru, recall, ROC-AUC ve specificity eğrileri gibi literatürde en yaygın kullanılan parametreler kullanılmıştır. Bu parametreler karmaşıklık matrisi verilerinden elde edilmektedir.

Doğruluk parametresi, doğru sınıflandırılmış tüm örneklerin (TP, TN) tüm örneklere (TP, TN, FP ve FN) oranı olarak tanımlanmaktadır ve Eşitlik 5.20 ile gösterilmektedir. Hassasiyet, doğru sınıflandırılmış tüm saldırıların (TP) doğru sınıflandırılmış tüm saldırı (TP) ve yanlış sınıflandırılmış normal örneklerin (FP) sayısına oranıdır. Precision, Eşitlik 5.22 ile temsil edilir. Recall doğru sınıflandırılan pozitif örneklerin sayısının doğru olarak sınıflandırılmış tüm örneklerin sayısına oranını ifade etmektedir ve Eşitlik 5.23 ile gösterilmektedir. Hassasiyet ve Recall parametrelerinin harmonik ortalaması F1-skoru olarak bilinir ve Eşitlik 5.21 ile gösterilmektedir.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (5.20)$$

$$F1 - Score = \frac{2TP}{2TP+FP+FN} \quad (5.21)$$

$$Precision = \frac{TP}{TP+FP} \quad (5.22)$$

$$Recall = \frac{TP}{FN+TP} \quad (5.23)$$

$$Specificity = \frac{TN}{TN+FP} \quad (5.24)$$



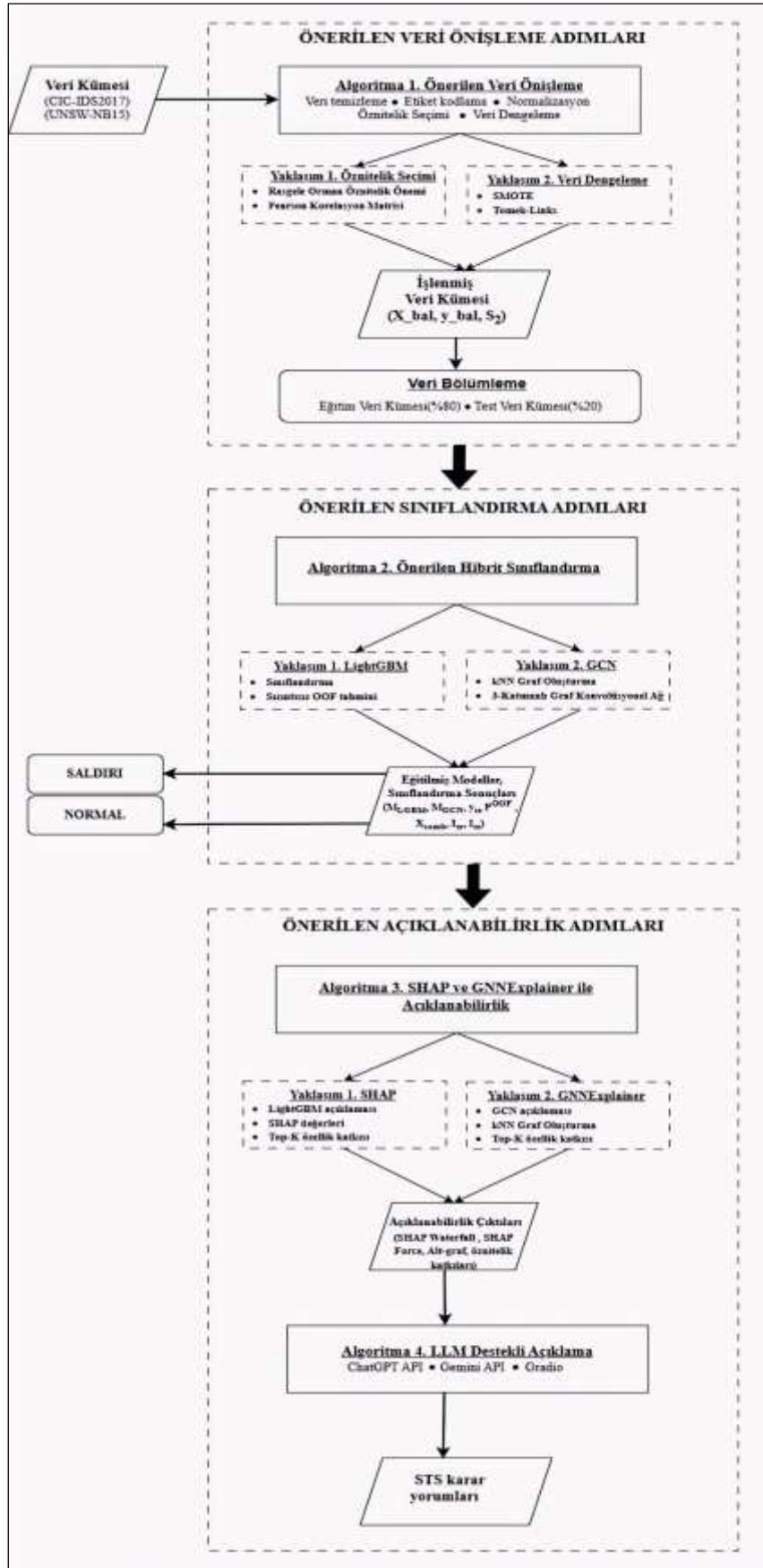
6. ÖNERİLEN MODEL

Bu tez çalışması kapsamında özgün bir açıklanabilir saldırı tespit sistemi önerilmiştir. Önerilen sistemin temel amacı, yüksek performanslı saldırı tespitinin yanı sıra sistem sonuçlarının açıklanarak güvenilir ve anlaşılır bir saldırı tespit sistemi ortaya koymaktır. Bu amaç doğrultusunda önerilen sistemde literatürde bulunan birden fazla teknik bir araya getirilerek hibrit bir yöntem önerilmiştir. Hibrit sınıflandırma, graf tabanlı öğrenme, çok boyutlu AYZ ve BDM entegrasyonunun tek bir STS mimarisi altında bütünleştirilebilirliği, Araştırma Sorusu-4 kapsamında ele alınmıştır. Önerilen sistem Şekil 6.1’de gösterildiği gibi veri önileme, sınıflandırma ve açıklanabilirlik aşamalarından oluşmaktadır. Bu tez çalışması kapsamında dört algoritma ve altı yaklaşım geliştirilerek özgün bir saldırı tespit sistemi önerilmiştir.

Önerilen sistemdeki tüm çalışma hattı, (i) veri önileme adımı, (ii) hibrit LightGBM+GCN temelli sınıflandırma adımı, (iii) SHAP ve GNNExplainer temelli açıklanabilirlik ve BDM-destekli yorumlama adımı olmak üzere üç temel adımdan oluşmaktadır. Her adımın, algoritmik olarak formüle edilip matematiksel temelleri ilerleyen alt bölümlerde ayrıntılı olarak açıklanmıştır.

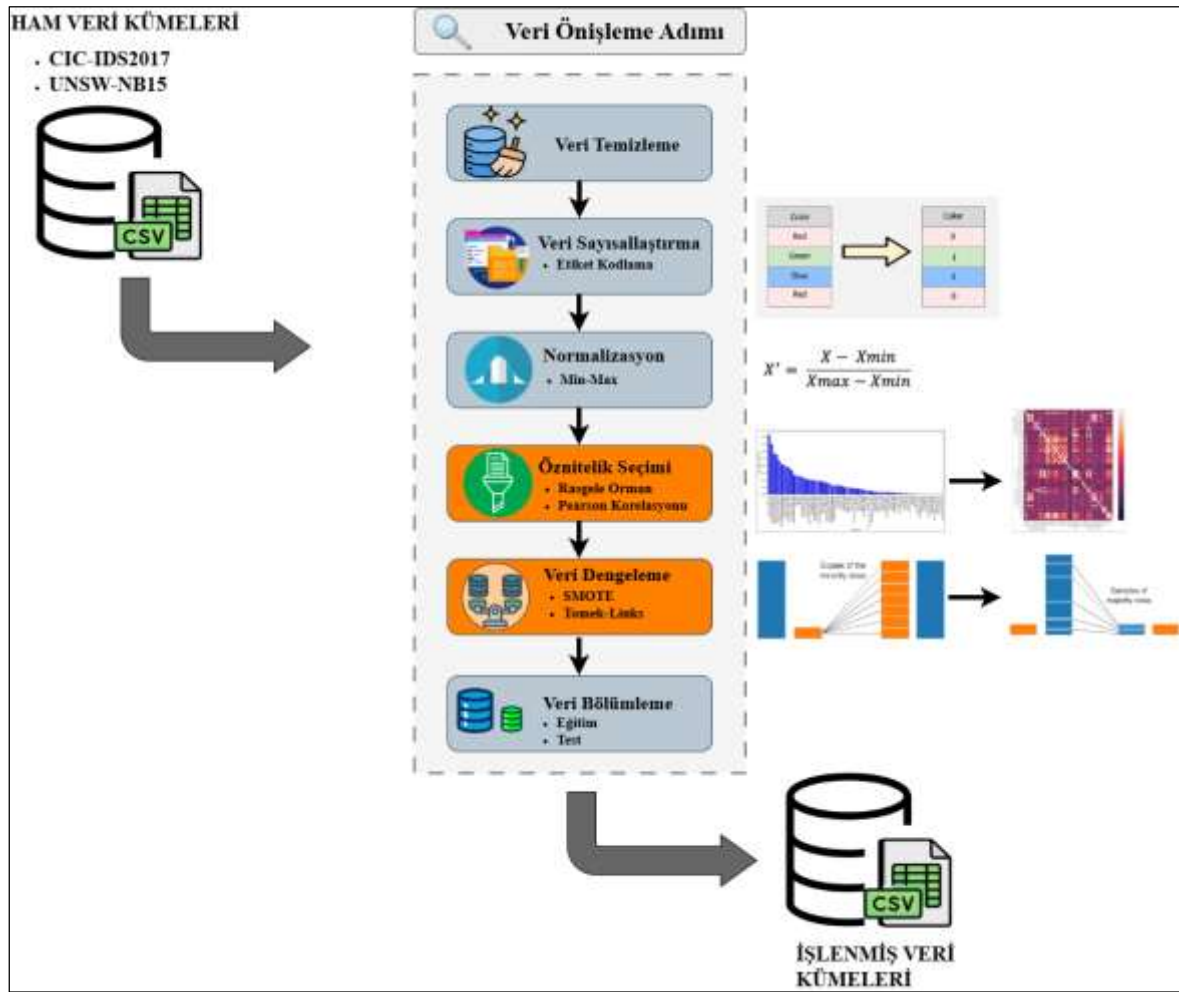
6.1. Önerilen Veri Önileme Adımı

Veri önileme, ham verileri kullanışlı ve verimli bir biçime dönüştürmek için kullanılan bir veri madenciliği tekniğidir. Veri madenciliği temel olarak kullanılan verinin kalitesine bağlıdır. Bu sebepten veri önileme, yapay zekâ sürecinde önemli bir adımdır. Veri önileme ham verileri daha sonraki işlemler için hazırlayan bir aşama olarak değerlendirilebilir. Gerçek dünya verileri genellikle eksik, düzensiz ve birçok hata içermektedir. Veri önileme, bu tür sorunları çözmek için kullanılan bir yöntemdir. Bu yöntem sayesinde verilerin daha düzenli hale getirilerek kullanılacak yapay zekâ algoritmasının performansı artırılmaktadır. Veri ön işleminin, veri hazırlama ve veri kümesinin dönüşümü ile ilgilenen ve aynı zamanda bilgi keşfini daha verimli hale getirmeye çalışan en önemli YZ adımlarından biri olduğu unutulmamalıdır. Veri kümesi önileme aşaması çalışılan verinin araştırmacılar tarafından tanınması açısından büyük önem arz etmektedir. Bu yüzden, YZ çalışmalarının en uzun süre çalışılması gereken adımı bu aşamadır.



Şekil 6.1. Önerilen sistemin çalışma süreçleri

Şekil 6.2’de gösterildiği gibi önerilen veri ön işleme adımı öznitelik seçimi ve veri dengeleme işlemlerinde özgün yaklaşımlar içermektedir. Öznitelik seçimi yaklaşımında ilk olarak rasgele orman algoritması ile hedef sınıfla öznitelikler arasındaki öneme bakılırken Pearson korelasyon katsayısı yaklaşımı ile özniteliklerin kendi aralarındaki etkileşime bakılmaktadır. Veri dengeleme yaklaşımında ise SMOTE ile üretilen sentetik veriler Tomek-Links ile alt örnekleme yapılarak gürültüsüz bir veri dengeleme işlemi yapılmıştır.



Şekil 6.2. Önerilen veri ön işleme adımı

Algoritma 1 önerilen sistemdeki veri ön işleme adımını detaylı olarak açıklamaktadır. Veri kümesi üzerinde yapay zekâ teknikleri uygulanmadan önce veri kümesi incelenerek, veri kümesinde sayısal olmayan değerler sayısal değerlere dönüştürülmüş, eksik veya çift kayıtlar işlenmiş, sonuca katkı sağlamayacak kayıtlar varsa silinmiş ve normalizasyon işlemi gerçekleştirilmiştir.

Ham veri kümesi;

$$\mathcal{D} = \{(x_i, y_i)\}_{i=1}^N, \quad x_i \in \mathbb{R}^F, y_i \in \mathcal{Y} \quad (6.1)$$

N : örnek sayısı,

F : öznitelik sayısı,

$\mathbf{x}_i = [x_{i1}, \dots, x_{iF}]^T$: i. örneğin özellik vektörü,

y_i : sınıf etiketini temsil etmektedir.

Amaç, $\mathcal{D}$ ham veri kümesini sınıflandırma aşamasına (LightGBM+GCN) uygun hale getirmek için temizleme, ölçekleme, etiket dönüştürme, öznitelik seçimi ve sınıf dengeleme işlemlerini sıralı şekilde uygulamaktır. Son çıktı olarak (X_{bal}, y_{bal}, S_2) elde edilir. Burada $S_2 \subseteq \{1, \dots, F\}$ nihai seçilmiş özellik indeks kümesini belirtmektedir. 1. ve 2. adımlar sırasıyla algoritmanın giriş ve çıkış parametrelerini açıklamaktadır. 1. adımda algoritmanın hangi parametrelerle çalıştığı ve hangi filtreleme kurallarını kullandığı açıkça ortaya konulmaktadır. 2. adım, algoritmanın çıktısını tanımlar: temizlenmiş, normalize edilmiş, etiketlenmiş ve dengelenmiş veri seti (X_{bal}, y_{bal}) ile birlikte, nihai seçilmiş öznitelik kümesi S . Böylece 2. Adımdaki sınıflandırma bileşeni, doğrudan bu çıktıyı giriş olarak kullanmaktadır. 3-5. adımlar verilerin temizlenmesinde kullanılan işlemleri göstermektedir. 6. adımda sınıf etiketleri sayısal forma dönüştürülür: $y \leftarrow \text{LabelEncode}(y)$. Bu dönüşüm, çok sınıflı IDS etiketlerinin LightGBM ve GCN kayıp fonksiyonları tarafından işlenebilmesi için gereklidir. 7. adımda özellikler Min–Max ölçekleme ile $[0,1]$ aralığına indirgenir. Bu adım özellikle kNN grafi kurulurken kritik öneme sahiptir; çünkü kNN, örnekler arası benzerliği mesafeye göre hesaplar ve farklı ölçeklerdeki öznitelikler mesafe hesabını hatalı yönlendirebilir. 8–12. adımlar, ilk öznitelik seçimi katmanı olan Random Forest feature importance işlemlerini gösterir. Bu bölümün amacı, sınıflandırmaya düşük katkı yapan öznitelikleri erken aşamada eleyerek boyutluluğu azaltmak, genelleme performansını iyileştirmek ve sonraki korelasyon analizinin daha anlamlı bir özellik uzayında yapılmasını sağlamaktır. 13–23. adımlar, ikinci öznitelik seçimi katmanıdır ve hedef yüksek korelasyonlu (redundant) öznitelikleri elemektir. Bu aşama, çok benzer bilgiyi tekrar eden özniteliklerin modelde gereksiz karmaşıklık oluşturmalarını engeller. Ayrıca özellikle GCN’e giden graf yapısında mesafe/benzerlik hesaplarının daha sağlıklı olmasına katkı sağlar. 24–26. adımlar, veri setindeki sınıf dengesizliğini gidermek için iki aşamalı bir yeniden örnekleme mekanizmasını ifade etmektedir. Bu kombinasyon, sadece örnek sayısını artırmakla kalmaz; aynı zamanda sınıf sınırlarını daha belirgin hale getirerek hem

LightGBM hem de GCN'in daha kararlı öğrenmesini desteklemektedir. 27. adımda algoritma, nihai dengeli veri setini ve seçilen özellikleri döndürür: $\text{return } (X_{bal}, y_{bal}, S_2)$.

Algoritma 1. önerilen veri önileme algoritması

```

1: Giriş  $\leftarrow$  Raw dataset  $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^N$ ; missing policy  $\mathcal{M}$ ; MinMaxScaler;
LabelEncoder; RF – FI threshold  $\tau_{FI} = 0.01$ ; Pearson threshold  $\tau_{\rho} = 0.95$ 
2: Çıkış  $\leftarrow$  Clean, normalized, label – encoded, balanced dataset  $(X_{bal}, y_{bal})$  and
selected feature set  $S$ 
3:  $\mathcal{D} \leftarrow \text{DropDuplicates}(\mathcal{D})$ 
4:  $\mathcal{D} \leftarrow \text{HandleMissing}(\mathcal{D}, \text{policy} = \mathcal{M})$ 
5:  $\mathcal{D} \leftarrow \text{RemoveAllZeroRows}(\mathcal{D})$  //  $\forall i$ : if  $\|x_i\|_0 = 0$  then delete
6:  $y \leftarrow \text{LabelEncode}(y)$ 
7:  $X \leftarrow \text{MinMaxScale}(X)$  //  $X_{ij}' = (X_{ij} - \min_j) / (\max_j - \min_j)$ 
8: // === Feature Selection Step – 1: Random Forest Feature Importance ===
9:  $RF \leftarrow \text{TrainRandomForest}(X, y)$ 
10:  $FI \leftarrow \text{FeatureImportance}(RF)$  //  $FI(j)$  for each feature  $j$ 
11:  $S_1 \leftarrow \{j \mid FI(j) \geq \tau_{FI}\}$ 
12:  $X \leftarrow X[:, S_1]$ 
13: // === Feature Selection Step – 2: Pearson Correlation Pruning ===
14:  $\rho \leftarrow \text{PearsonCorrelationMatrix}(X)$ 
15:  $S_2 \leftarrow S_1$ 
16: for each pair  $(a, b)$  with  $a < b$  do
17: if  $|\rho(a, b)| > \tau_{\rho}$  then
18: // remove the less – informative one (prefer dropping lower FI)
19:  $j_{drop} \leftarrow \text{ArgMin}(FI(a), FI(b))$ 
20:  $S_2 \leftarrow S_2 \setminus \{j_{drop}\}$ 
21: end if
22: end for
23:  $X \leftarrow X[:, S_2]$ 
24: // === Balancing: SMOTE then Tomek – Links ===
25:  $(X_{sm}, y_{sm}) \leftarrow \text{SMOTE}(X, y)$ 
26:  $(X_{bal}, y_{bal}) \leftarrow \text{TomekLinks}(X_{sm}, y_{sm})$ 
27: return  $(X_{bal}, y_{bal}, S_2)$ 

```

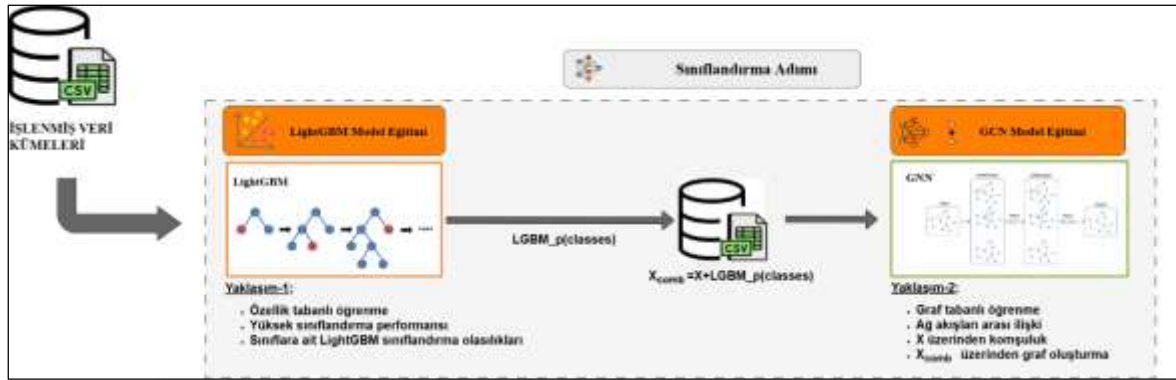
Yaklaşım 1

Yaklaşım 2

6.2. Önerilen Sınıflandırma Adımı

Bu tez çalışmasında sınıflandırma adımı özgün bir sınıflandırma yaklaşımı geliştirilmiştir. LightGBM ve GCN algoritmalarından oluşan hibrit bir sınıflandırma yaklaşımı önerilmektedir. Literatürde gözlemlenen model kullanım eğilimleri ve yöntemsel sınırlılıklar dikkate alındığında, tablosal STS verileri için boosting tabanlı yaklaşımların

daha kararlı ve genellenebilir bir temel sunduğu graf temelli yöntemlerin ise ilişkisel öğrenmeyi desteklediği görülmüş; bu değerlendirme Araştırma Sorusu-1'e doğrudan dayanak oluşturmuştur. Geliştirilen sınıflandırma yaklaşımı LightGBM'nin tablosal ayrıştırma gücü ile GCN'in ilişkisel (graph-based) öğrenme kapasitesini birleştirerek daha kararlı ve bağlamsal bir sınıflandırma üretmeyi amaçlamaktadır. Önerilen sınıflandırma adımı ile ağ trafiğinin hem istatistiksel özelliklerini hem de ilişkisel/topolojik yapısını eş zamanlı öğrenebilme yeteneğine sahip Şekil 6.3'de gösterildiği gibi LightGBM (özellik temelli) ve GCN (ilişkisel temelli) modellerin entegre edildiği yeni bir Zeki STS mimarisi sunulmuştur.



Şekil 6.3. Önerilen sınıflandırma adımı

Algoritma 2, önerilen sistemin özgün olarak oluşturulan LightGBM+GCN tabanlı sınıflandırma adımını uçtan uca açıklamaktadır. Bu algoritma iki temel fikri birleştirmektedir: (i) tablosal veride güçlü bir sınıflandırıcı olan LightGBM ile yüksek ayırt edicilik sağlamak, (ii) örnekler arası benzerlik ilişkilerini kodlayan kNN grafi üzerinde GCN ile bağlamsal (context-aware) düzeltme yapmak. Böylece model, yalnızca örnek içi (feature-level) bilgiye değil, örnekler arası yapısal ilişkilere de dayalı karar vermektedir. Algoritma 2'deki 1. ve 2. adımlar sırasıyla algoritmanın giriş ve çıkış parametrelerini açıklamaktadır. 3-6. adımlar sınıf oranlarını koruyacak şekilde stratified split yaparak (I_{tr} , I_{te}) indekslerini üretir ve X ile y 'yi eğitim/test alt kümelerine ayırır. İndeks bazlı yaklaşım, daha sonra PyG mask'lerinin doğru kurulmasını sağlar ve tüm pipeline boyunca tutarlılık oluşturur. 7-10. adımlarda LightGBM modeli eğitilir test üzerinde tahmin alınarak temel metrikler raporlanır. Bu rapor, hibrit mimariye geçmeden önce LightGBM'nin tek başına başarımını göstermek için kullanılır. 11-19. adımları hibrit mimarinin en kritik noktalarından birini oluşturmaktadır. Adım 12'de P^{OOF} matrisi başlatılır. Adım 13-18'de her fold için model

yeniden eğitilir ve sadece o fold'un validation örnekleri için olasılık tahmini üretilir. Böylece her örneğin olasılık vektörü, o örneği görmeyen bir modelden elde edilir. Adım 19'da $X_{comb} = [X \parallel P^{OOF}]$ elde edilir. Bu birleşim, GCN'e hem ham öznitelikleri hem de LightGBM'nin öğrendiği karar sınırlarını yansıtan "yüksek seviyeli" sınıf olasılıklarını birlikte sunar. Adım 21'de kNN grafi X üzerinden kurulmaktadır. Bunun nedeni, P^{OOF} 'un zaten sınıf sinyali taşımasıdır; grafin sınıf sinyaline göre kurulması, komşulukların "etiket benzerliğine" kaymasına neden olabilir. Bu nedenle graf, normalize edilmiş orijinal özellik uzayında örnek benzerliğine göre oluşturulur. Adım 22'de simetrikleştirme ile graf yönsüz hale getirilir ve Adım 23–24'te PyG'nin kullandığı edge_index formatına dönüştürülür. 25–29. adımlarda PyG veri yapısı ve mask'ler kurulmaktadır. Bu adımlarda düğüm öznitelikleri olarak X_{comb} atanır (Adım 26). Bu, GCN'in "LightGBM'den gelen olasılıkları" da kullanarak öğrenmesini sağlar. Train/test mask'leri, daha önce elde edilen I_{tr}, I_{te} indeksleriyle oluşturulur. Böylece kayıp fonksiyonu yalnızca eğitim düğümlerinde hesaplanır. 30–42. adımlar üç katmanlı GCN eğitim sürecini ifade etmektedir. Bu yapı, hem temsil gücünü artırır (3 katman) hem de aşırı öğrenmeyi azaltır (BN + dropout + weight decay). 43–47. adımlar değerlendirmelerin ve çıktıların alındığı aşamaları göstermektedir.

Algoritma 2. önerilen hibrit sınıflandırma algoritması

```

1: Giriş  $\leftarrow$  Balanced dataset  $(X, y)$  ; LGBM hyperparams  $\theta_L$ ; CV folds  $K = 5$ ;
seed = 42; kNN:  $k = 8$  (connectivity, include_self = False); GCN: hidden_dim =
256; dropout  $p = 0,5$ ; lr = 0,001; weight_decay =  $5e - 4$ ; epochs = 200
2: Çıkış  $\leftarrow$  Trained models  $(M_{LGBM}, M_{GCN})$ ; test predictions  $\hat{y}_{te}$ ; OOF probabilities
 $P^{OOF}$ ; combined features  $X_{comb}$ ; indices  $(I_{tr}, I_{te})$ 
3: indices  $\leftarrow \{1, 2, \dots N\}$ 
4:  $(I_{tr}, I_{te}) \leftarrow$  StratifiedSplit(indices, y, test_size = 0.20, seed = 42)
5:  $X_{tr} \leftarrow X[I_{tr}]$ ,  $y_{tr} \leftarrow y[I_{tr}]$ ,
6:  $X_{te} \leftarrow X[I_{te}]$ ,  $y_{te} \leftarrow y[I_{te}]$ ,
7: // === (A) LightGBM baseline training ===
8:  $M_{LGBM} \leftarrow$  TrainLGBM( $X_{tr}, y_{tr}; \theta_L$ )
9:  $\hat{y}_{te}^L \leftarrow$  Predict( $M_{LGBM}, X_{te}$ )
10: ReportMetrics( $y_{te}, \hat{y}_{te}^L$ )

```

```

11: // === (B) Leak – free OOF predict (stacking features) ===
12:  $P^{OOF} \leftarrow \text{zeros} = (N, C)$ 
13:  $CV \leftarrow \text{StratifiedKfold}(K, \text{shuffle} = \text{True}, \text{seed} = 42)$ 
14: for each fold  $f$  in  $CV$  do
15:    $(I_{\text{train}}(f), I_{\text{val}}(f)) \leftarrow \text{SplitFold}(f)$ 
16:    $M_f \leftarrow \text{TrainLGBM}(X[I_{\text{train}}(f)], y[I_{\text{train}}(f)]; \theta_L)$ 
17:    $P^{OOF}[I_{\text{val}}(f), :] \leftarrow \text{Predict}(M_f, X[I_{\text{val}}(f)])$ 
18: end for
19:  $X_{\text{comb}} \leftarrow \text{Concat}(X, P^{OOF})$ 
20: // === (C) kNN graph construction from  $X$  ===
21:  $A \leftarrow \text{kNNGraph}(X, k = 8, \text{mode} = \text{connectivity}, \text{include\_self} = \text{False})$ 
22:  $A \leftarrow \max(A, A^T)$  // symmetrize
23:  $G \leftarrow \text{GraphFromAdjacency}(A)$ 
24:  $\text{edge\_index} \leftarrow \text{ToEdgeIndex}(G)$ 
25: // === (D) PyG data preparation (node features =  $X_{\text{comb}}$ ) ===
26:  $\text{data}.x \leftarrow \text{Tensor}(X_{\text{comb}})$ 
27:  $\text{data}.y \leftarrow \text{Tensor}(y)$ 
28:  $\text{train\_mask} \leftarrow \text{BoolMask}(N); \text{train\_mask}[I_{\text{tr}}] \leftarrow \text{True}$ 
29:  $\text{test\_mask} \leftarrow \text{BoolMask}(N); \text{test\_mask}[I_{\text{te}}] \leftarrow \text{True}$ 
30: // === (E) 3 – Layer GCN (GCNConv + BN + ReLU + Dropout) ===
31: Initialize  $W_1, W_2, W_3$  and BatchNorm params
32: for epoch  $\leftarrow 1$  to epochs do
33:   // forward
34:    $H_1 \leftarrow \text{GCNConv}(\text{data}.x, \text{edge\_index}; W_1)$ 
35:    $H_1 \leftarrow \text{BN}(H_1); H_1 \leftarrow \text{ReLU}(H_1); H_1 \leftarrow \text{Dropout}(H_1, p)$ 
36:    $H_2 \leftarrow \text{GCNConv}(H_1, \text{edge\_index}; W_2)$ 
37:    $H_2 \leftarrow \text{BN}(H_2); H_2 \leftarrow \text{ReLU}(H_1); H_2 \leftarrow \text{Dropout}(H_2, p)$ 
38:    $Z \leftarrow \text{GCNConv}(H_2, \text{edge\_index}; W_3)$  // logits
39:    $\mathcal{L} \leftarrow \text{CrossEntropy}(Z[\text{train\_mask}], \text{data}.y[\text{train\_mask}])$ 
40:    $(W_1, W_2, W_3, \text{BN}) \leftarrow \text{AdamStep}(\nabla \mathcal{L}, lr = 0,001, \text{weight\_decay} = 5e - 4)$ 
41: end for
42: // === (F) Evaluate on test nodes ===
43:  $Z \leftarrow \text{ForwardPass}(\text{data}.x, \text{edge\_index})$ 

```

Yaklaşım 1**Yaklaşım 2**

```

44:  $\hat{y} \leftarrow \text{ArgMax}(Z, \text{axis} = 1)$ 
45:  $\hat{y}_{te} \leftarrow \hat{y}[\text{test\_mask}]$ 
46:  $\text{ReportMetrics}(\text{data.y}[\text{test\_mask}], \hat{y}_{te})$ 
47:  $\text{return } (M_{LGBM}, M_{GCN}, \hat{y}_{te}, P^{OOF}, X_{comb}, I_{tr}, I_{te})$ 

```

Önerilen hibrit sınıflandırma modelinin detaylı matematiksel altyapısı şu şekildedir:

Önişleme sonrası elde edilen dengeli veri:

$$X \in \mathbb{R}^{N \times F}, y \in \{0, 1, \dots, C - 1\}^N \quad (6.2)$$

N : örnek sayısı (düğüm sayısı)

F : öznitelik sayısı

C : sınıf sayısı

$x_i \in \mathbb{R}^F$: i . örneğin vektörü (satır)

y_i : sınıf etiketini temsil etmektedir.

Hibrit çıktı fonksiyonel olarak şu şekilde ifade edilebilir:

$$\hat{y}_i = \arg \max_c GCN([x_i || p_i^{OOF}], A)_c \quad (6.3)$$

Burada;

$p_i^{OOF} \in \mathbb{R}^C$: i . örnek için leak-free OOF olasılık vektörü

$[x_i || p_i^{OOF}] \in \mathbb{R}^{F+C}$: birleştirilmiş düğüm özniteliği

A : kNN grafindan gelen komşuluk (adjacency) matrisi ifade etmektedir.

LightGBM (GBDT) additive bir modeldir ve burada her f_t bir karar ağacıdır:

$$F(x) = \sum_{t=1}^T f_t(x) \quad (6.4)$$

Çok sınıflı durumda her sınıf için skor üretilir:

$$F_c(x) = \sum_{t=1}^T f_{t,c}(x) \quad (6.5)$$

ve softmax ile olasılığa çevrilir:

$$p^L(y = c | x) = \frac{\exp(F_c(x))}{\sum_{k=1}^C \exp(F_k(x))} \quad (6.6)$$

Test tahmini aşağıdaki şekilde hesaplanır ve raporlanan metrikler bu tahmin üzerinden hesaplanır.

$$\hat{y}_{te}^L = \operatorname{argmax}_c p^L(y = c | x) \forall x \in X_{te} \quad (6.7)$$

Daha sonra stacking ile her örnek için LightGBM'nin “öğrenilmiş” sınıf olasılıklarının ek özellik olarak üretilmesi amaçlanmıştır. Burada bunu yaparken leakage oluşmaması önemlidir.

Stratified K-fold:

$$\{F_1, F_2, \dots, F_K\}, \bigcup_{k=1}^K F_k = I, F_k \cap F_j = \emptyset \ (k \neq j) \quad (6.8)$$

Her fold için eğitim kümesi:

$$I^{(-k)} = I \setminus F_k \quad (6.9)$$

Her $i \in F_k$ için OOF olasılık:

$$p_i^{OOF} = p^L(y | x_i; M^{(-k)}) \quad (6.10)$$

Burada $M^{(-k)}$ yalnızca $I^{(-k)}$ üzerinde eğitilmiş LightGBM modelidir.

OOF matrisi:

$$P^{OOF} = \begin{bmatrix} (p_1^{OOF})^\top \\ \vdots \\ (p_N^{OOF})^\top \end{bmatrix} \in \mathbb{R}^{N \times C} \quad (6.11)$$

Leak-free garantisi: Her p_i^{OOF} x_i 'yi eğitimde görmeyen bir modelden gelir. Bu yüzden stacking özellikleri “sızıntısız” olur:

$$p_i^{OOF} \perp (\text{training on } x_i)$$

Birleştirilmiş düğüm özniteliği:

$$X_{comb} = [X \parallel P^{OOF}] \in \mathbb{R}^{N \times (F+C)} \quad (6.12)$$

Her düğüm için:

$$x_i^{comb} = [x_i \parallel p_i^{OOF}] \in \mathbb{R}^{F+C} \quad (6.13)$$

Grafın düğümleri örneklerdir. k-en yakın komşu ilişkisi normalize edilmiş özellik uzayında kurulmaktadır. Bir metrik $d(.,.)$ (genellikle Euclidean) ile:

$$\mathcal{N}_k(i) = \text{set of } k \text{ nearest neighbors of } x_i \quad (6.14)$$

Adjacency matrisi:

$$A_{ij} = \begin{cases} 1, & j \in \mathcal{N}_k(i) \\ 0, & otherwise \end{cases} \quad (6.15)$$

Yönsüz graf için şu şekilde simetrikleştirme uygulanır:

$$A \leftarrow \max(A, A^T) \quad (6.16)$$

P^{OOF} sınıf sinyali taşıdığı için X_{comb} üzerinden graf kurmak, komşulukları “etiket benzerliğine” kaydırabilir ve yapay bir yapı oluşturabilir. Bu nedenle graf $A = A(X)$ olarak olarak yalnızca orijinal normalize edilmiş özelliklerden inşa edilmektedir.

GCN eğitiminde yalnızca bazı düğümler etiketli eğitim düğümü olarak kullanılır.

Maskeler:

$$m_i^{tr} = \begin{cases} 1, & i \in I_{tr} \\ 0, & otherwise \end{cases}, \quad m_i^{te} = \begin{cases} 1, & i \in I_{te} \\ 0, & otherwise \end{cases} \quad (6.17)$$

Node features:

$$H^{(0)} = X_{comb} \quad (6.18)$$

Kipf & Welling GCN formülasyonuna [556] dayanarak;

Self-loop eklenmiş adjacency:

$$\tilde{A} = A + I \quad (6.19)$$

Degree matrisi:

$$\tilde{D}_{ii} = \sum_j \tilde{A}_{ij} \quad (6.20)$$

Normalize propagation operatörü:

$$\tilde{A} = \tilde{D}^{-1/2} \tilde{A} \tilde{D}^{-1/2} \quad (6.21)$$

GCN katmanı:

$$H^{(l+1)} = \sigma(\hat{A}H^{(l)}W^{(l)}) \quad (6.22)$$

Burada, $W^{(l)}$ öğrenilebilir ağırlık matrisi, σ ise aktivasyon (ReLU) ifade etmektedir.

Bu çalışmada 3 katman kullanılmıştır ve Z logit matrisini ifade etmektedir:

$$H^{(1)} = Dropout(ReLU(BN(\hat{A}H^{(0)}W^{(0)})), p) \quad (6.23)$$

$$H^{(2)} = Dropout(ReLU(BN(\hat{A}H^{(1)}W^{(1)})), p) \quad (6.24)$$

$$Z = \hat{A}H^{(1)}W^{(1)} \in \mathbb{R}^{N \times C} \quad (6.25)$$

i . düğüm için:

$$p^G(y = c | i) = \frac{\exp(Z_{ic})}{\sum_{k=1}^C \exp(Z_{ik})} \quad (6.26)$$

$$\hat{y}_i = \operatorname{argmax}_c p^G(y = c | i) \quad (6.27)$$

Yalnız eğitim düğümleri üzerinde cross-entropy:

$$\mathcal{L} = - \sum_{i \in I_{tr}} \log p^G(y = y_i | i) \quad (6.28)$$

Weight decay ile düzenlenmiş hali:

$$\mathcal{L}_{total} = \mathcal{L} + \lambda \sum_l \|W^{(l)}\|_2^2 \quad (6.29)$$

Burada λ weight_decay katsayısıdır($5e - 4$).

Adam optimizasyonu parametreleri $\theta = \{W^{(l)}, BN\}$ için:

$$\theta \leftarrow \theta - \alpha \frac{\hat{m}_t}{\sqrt{\hat{v}_t + \epsilon}} \quad (6.30)$$

α öğrenme oranını, $\hat{m}_t$ ve $\hat{v}_t$ Adam'ın bias-düzeltemeli moment tahminlerini belirtmektedir.

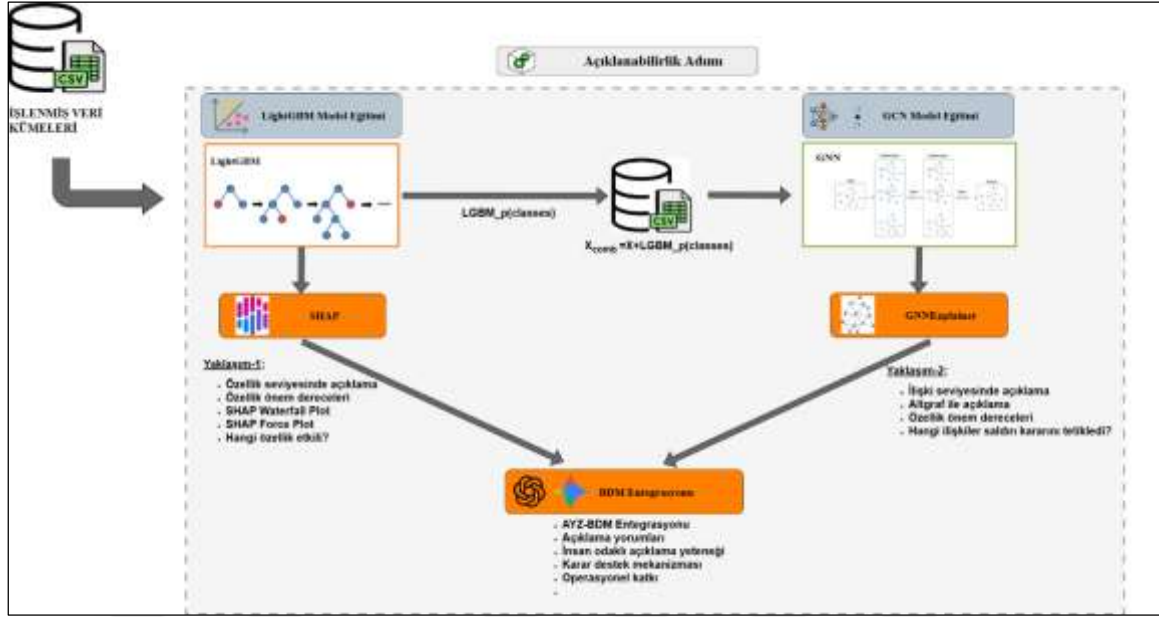
Eğitim sonrası tüm düğümler için ileri besleme ile Z hesaplanır ve test düğümleri için:

$$\hat{y}_{te} = \{\hat{y}_i | i \in I_{te}\} \quad (6.31)$$

Performans metrikleri y_{te} ile karşılaştırılarak raporlanır.

6.3. Önerilen Açıklanabilirlik Adımı

Literatürde AYZ uygulamalarının büyük ölçüde öznitelik-seviyesi açıklamalarla sınırlı kaldığı; graf-seviyesi ve ilişkisel açıklamaların ise oldukça sınırlı olduğu gözlemi, Araştırma Sorusu-2 kapsamında ele alınan temel boşluğu ortaya koymaktadır. Bu çalışmada SHAP ve GNNExplainer’ın birlikte kullanılmasıyla elde edilen çok boyutlu açıklamalar, literatürde sıklıkla ihmal edilen graf-seviyesi gerekçeleri de ortaya koymuş ve Araştırma Sorusu-2 kapsamında belirlenen boşluğa somut bir yanıt sunmuştur. Şekil 6.4 önerilen açıklanabilirlik adımını göstermektedir. Önerilen adım SHAP, GNNExplainer AYZ tekniklerini ve BDM entegrasyonunu içermektedir. Algoritma 3 önerilen sistemin açıklanabilirlik adımının AYZ bileşenlerini açıklamaktadır. Bu algoritmanın amacı, hibrit sınıflandırma (LightGBM+GCN) sonucunda elde edilen tahminleri hem AYZ hem de BDM düzeyinde açıklanabilir hale getirerek kullanıcıya bir web arayüzü üzerinden sunmaktır. Böylece güvenlik analistleri, yalnızca “hangi sınıf?” sonucunu değil, aynı zamanda “neden bu sınıf?” ve “hangi ilişkisel bağlamda?” sorularının cevaplarını da görebilir. 1. ve 2. adımlar sırasıyla algoritmanın giriş ve çıkış parametrelerini açıklamaktadır. 3-6. adımlarda LightGBM modelinin sonucu özellik katkıları üzerinden açıklanmaktadır. 4. adımda LightGBM’nin sınıf olasılıkları p_L hesaplanır. 5. adımda SHAP değerleri ϕ_i üretilerek her özneliğin tahmine olan katkısı nicel hale getirilir. 6. adımda en yüksek mutlak katkıya sahip top-K öznitelik seçilerek açıklama raporlanabilir hale getirilir. 7-10. adımlarda GCN’in hedef düğüm için ürettiği karar elde edilmektedir. 8. adımda düğüm logitleri z_i çıkarılır. 9. adımda softmax ile olasılıklar p_G hesaplanır. 10. adımda nihai tahmin sınıfı $\hat{y}$ belirlenir. Bu çıktı, GNNExplainer’ın hedef sınıfını ve BDM paketinin temel “decision” bilgisini oluşturur. 11-21. adımlarda GCN’in kararının hangi alt-graf ve hangi düğüm öznitelikleri üzerinden oluştuğunu açıklanmaktadır. 12. adımda kenar maskesi M_E ve özellik maskesi M_F başlatılır. 13–19. adımlar boyunca maskeler optimize edilir: amaç, maskelenmiş graf/özelliklerle modelin $\hat{y}$ sınıfını yüksek olasılıkla üretmesini sağlarken aynı zamanda açıklamayı seyrek (sparse) tutmaktır. 20. adımda en önemli kenarlardan oluşan açıklayıcı alt-graf G_i^* çıkarılır. 21. adımda M_F maskesine göre en önemli top-K öznitelik listesi elde edilir. Bu kısım ile, “GCN bu kararı komşuluk bağlamında nasıl verdi?” sorusu cevaplanmaktadır.



Şekil 6.4. Önerilen açıklanabilirlik adımı

Algoritma 3. SHAP + GNNExplainer ile açıklanabilirlik

1: Giriş $\leftarrow$ Trained models (M_{LGBM}, M_{GCN}); graph data ($data.x, data.edge\_index$);
 test sample (node) i with features x_i ; SHAP explainer;
 GNNExplainer params (E_e, λ_E); LLM APIs (ChatGPT, Gemini);
 UI framework Gradio; top - K features K_f

2: Çıkış $\leftarrow$ Human - readable explanation (feature - level + graph - level)

3: // === (A) Feature - level explanation (SHAP on LightGBM) ===

4: $p_L \leftarrow \text{Predict}(M_{LGBM}, x_i)$

5: $\phi_i \leftarrow \text{SHAPValues}(M_{LGBM}, x_i)$

6: $\text{TopK}^{SHAP} \leftarrow \text{SelectTopK}(|\phi_i|, K_f)$ Yaklaşım 1

7: // === (B) Contextual prediction (GCN) ===

8: $z_i \leftarrow \text{ForwardNodeLogits}(M_{GCN}, data.x, data.edge\_index, i)$

9: $p_G \leftarrow \text{Softmax}(z_i)$

10: $\hat{y} \leftarrow \text{ArgMax}(p_G)$

11: // === (C) Graph - level explanation (GNNExplainer on GCN) ===

12: $(M_E, M_F) \leftarrow \text{InitMasks}()$

13: for epoch $\leftarrow 1$ to E_e do

14: $\widetilde{edge\_index} \leftarrow \text{ApplyEdgeMask}(data.edge\_index, M_E)$

```

15:  $\tilde{x} \leftarrow \text{ApplyFeatureMask}(\text{data}.x, M_F)$ 
16:  $\tilde{z}_i \leftarrow \text{ForwardNodeLogits}(M_{GCN}, \tilde{x}, \widetilde{\text{edge\_index}}, i)$ 
17:  $\mathcal{L}_E \leftarrow -\log P(\hat{y} | \tilde{z}_i) + \lambda_E \|M_E\|_1 + \lambda_E \|M_F\|_1$ 
18:  $(M_E, M_F) \leftarrow \text{GradientStep}(\nabla \mathcal{L}_E)$ 
19: end for
20:  $G_i^* \leftarrow \text{ExtractSubgraph}(\text{data}.edge\_index, M_E, \text{threshold})$ 
21:  $\text{TopK}^{GNN} \leftarrow \text{SelectTopK}(M_F, K_f)$ 
22: return  $(p_L, \emptyset_i, \text{TopK}^{SHAP}, p_G, \hat{y}, G_i^*, \text{TopK}^{GNN})$ 

```

Yaklaşım 2

6.4. BDM Yaklaşımı ve Entegrasyonu

Bu tez çalışmasında, önerilen hibrit IDS modelimizde elde edilen çıktıların yorumlanması sürecinde yalnızca SHAP ve GNNExplainer AYZ yöntemleri değil, aynı zamanda BDM üzerinden oluşturulan doğal dil açıklamaları da kullanılmıştır. Büyük dil modellerinin STS süreçlerine entegrasyon biçimleri ve bu entegrasyonların kullanıcı/analist perspektifinden nasıl ele alındığı, Araştırma Sorusu-3 doğrultusunda literatür bağlamında değerlendirilmiştir. Bu çalışmada BDM'ler, literatürde yaygın olarak görülen doğrudan alarm yorumlama yaklaşımından farklı olarak, açıklanabilirlik çıktılarının yorumlayıcısı olarak konumlandırılmış; bu tercih Araştırma Sorusu-3'te vurgulanan kullanıcı değerlendirmesi boşluğunu hedeflemektedir. Bu bağlamda, OpenAI tarafından geliştirilen ChatGPT ve Google tarafından geliştirilen Gemini BDM'leri yorumlama katmanı olarak önerilen modele eklenmiş ve analistlere sistemin kararlarının daha anlaşılır bir şekilde sunulması amaçlanmıştır. BDM entegrasyonu ile siber operasyon merkezlerinde(SOC) çalışan siber güvenlik analistlerine yönelik bir karar destek mekanizması oluşturulması amaçlanmıştır.

Algoritma 4 önerilen sisteme BDM entegrasyonunu açıklamaktadır. 1. ve 2. adımlar sırasıyla algoritmanın giriş ve çıkış parametrelerini açıklamaktadır. 3 ve 4. adımlarda SHAP ve GNNExplainer çıktıları, LightGBM ve GCN olasılıklarıyla birlikte tek bir payload içinde birleştirilir. Payload; tahmin sınıfını, güven skorlarını, top-K öznitelikleri ve açıklayıcı alt-grafları içerir. Böylece BDM'ler, sadece sayısal çıktıyı değil, kararın hem özellik hem de graf temelli gerekçesini okuyabilir. 5-7. adımlarda payload ChatGPT ve Gemini olmak üzere iki farklı BDM'e gönderilir. Bu iki çıktı, kullanıcıya farklı anlatım biçimlerinde açıklama

sunarak kullanılabilirliği artırır. 8-10. adımlarda ise açıklamalar ve görseller (SHAP plot + alt-graf görselleştirmesi) Gradio arayüzüne bağlanır ve kullanıcıya web üzerinden sunulur. Böylece modelin çıktıları operasyonel bir IDS arayüzünde kullanılabilir hale gelir.

Algoritma 4. LLM-destekli açıklama(Algoritma 3 genişletilmiştir)

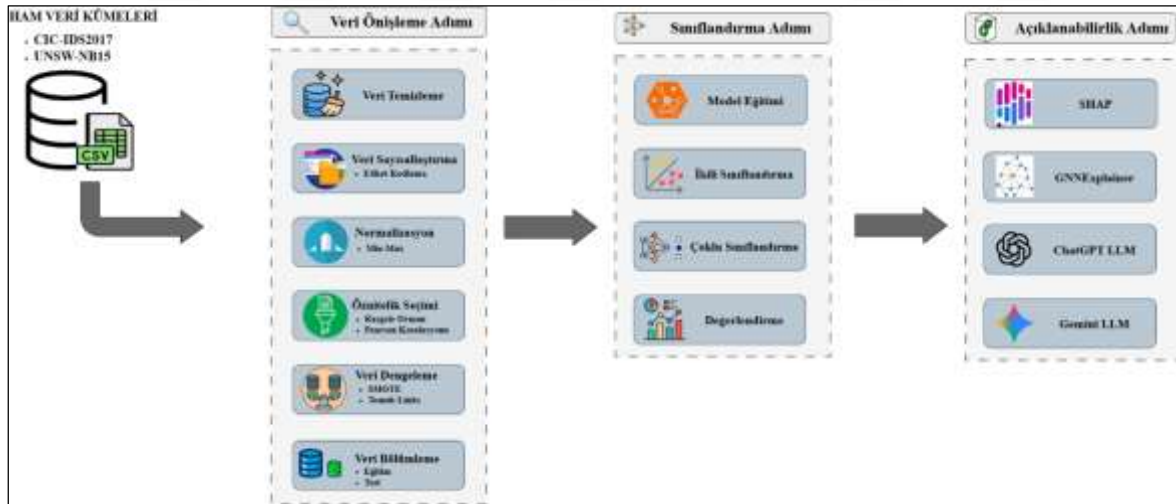
```

1: Giriş  $\leftarrow$  Outputs from Algorithm 3:  $(p_L, \emptyset_i, TopK^{SHAP}, p_G, \hat{y}, G_i^*, TopK^{GNN})$ ;
   LLM APIs (ChatGPT, Gemini); UI framework Gradio
2: Çıkış  $\leftarrow$  Human – readable explanation texts + interactive web UI response
3: // === (A) Build LLM prompt package (payload) ===
4: payload  $\leftarrow$  {
    "pred_label":  $\hat{y}$ ,
    "prob_gcn":  $p_G$ ,
    "prob_lgbm":  $p_L$ ,
    "top_features_shap":  $TopK^{SHAP}$ ,
    "shap_values":  $\emptyset_i$ ,
    "subgraph":  $G_i^*$ ,
    "top_features_gnn":  $TopK^{GNN}$ 
}
5: // === (B) Query LLMs ===
6: text_gpt  $\leftarrow$  CallChatGPT_API(payload)
7: text_gemini  $\leftarrow$  CallGemini_API(payload)
8: // === (F) Serve via Gradio ===
9: UI  $\leftarrow$  GradioInterface(
    inputs =  $x_i$ ,
    outputs = { text_gpt, text_gemini, SHAPPlots( $\emptyset_i$ ), SubgraphPlot( $G_i^*$ ) }
)
10: return UI

```

7. DENEYSEL ÇALIŞMALAR

Bu bölümde tez çalışması kapsamında geliştirilen sistemin deneysel çalışma akışı Şekil 7.1’de verilmiştir. Modellerin eğitim ve test çalışmaları TÜBİTAK TRUBA ve Google Colab ortamlarında python programlama dili kullanılarak yapılmıştır. Makine öğrenmesi ve derin öğrenme algoritmaları için scikit-learn, keras, torch ve tensorflow kütüphaneleri kullanılmıştır. Deneysel çalışmalarda; RF (Rastgele Orman), DT (Karar Ağacı), NB (Naive Bayes), LR (Lojistik Regresyon), MLP (Çok Katmanlı Algılayıcı), LSTM (Uzun Kısa Süreli Bellek), CNN (Evrişimsel Sinir Ağı), Gated Recurrent Unit (GRU), CNN-LSTM, CNN-GRU, eXtreme Gradient Boosting (XGBoost), LightGBM, Graph Convolutional Network (GCN), Graph Attention Networks (GAT), Graph Isomorphism Network (GIN), GraphSAGE, LightGBM-GIN, XGBoost-GCN, XGBoost-GIN ve önerilen model (LightGBM-GCN) dâhil olmak üzere 20 farklı teknik kullanılmıştır. Ayrıca, sanal sunucu ihtiyaçları için Intel Core i7-6500U işlemci, 12 GB RAM, 1 TB disk alanına ve 2 GB NVIDIA GeForce ekran kartına sahip laptop üzerinde VMware Workstation Pro kullanılmıştır.



Şekil 7.1. Deneysel akış şeması

7.1. Veri Önleme Adımı Sonuçları

Bu bölümde önerilen sistemin veri önleme adımı sonucunda elde edilen sonuçlar sunulmuştur.

MinMaxScaler kullanılarak yapılan normalizasyon işlemi sayesinde yüksek sayısal değerlere sahip bazı özniteliklerin algoritma sonucunu etkileyerek performansı olumsuz etkilemesinin önüne geçilmiştir. Veri kümeleri üzerinde yapılan label encoding işlemi sonucundaki sınıfların sayısal etiketleri Çizelge 7.1 ve Çizelge 7.2’de sunulmaktadır.

Çizelge 7.1. CIC-IDS2017 veri kümesi sınıf-etiket eşleştirmeleri

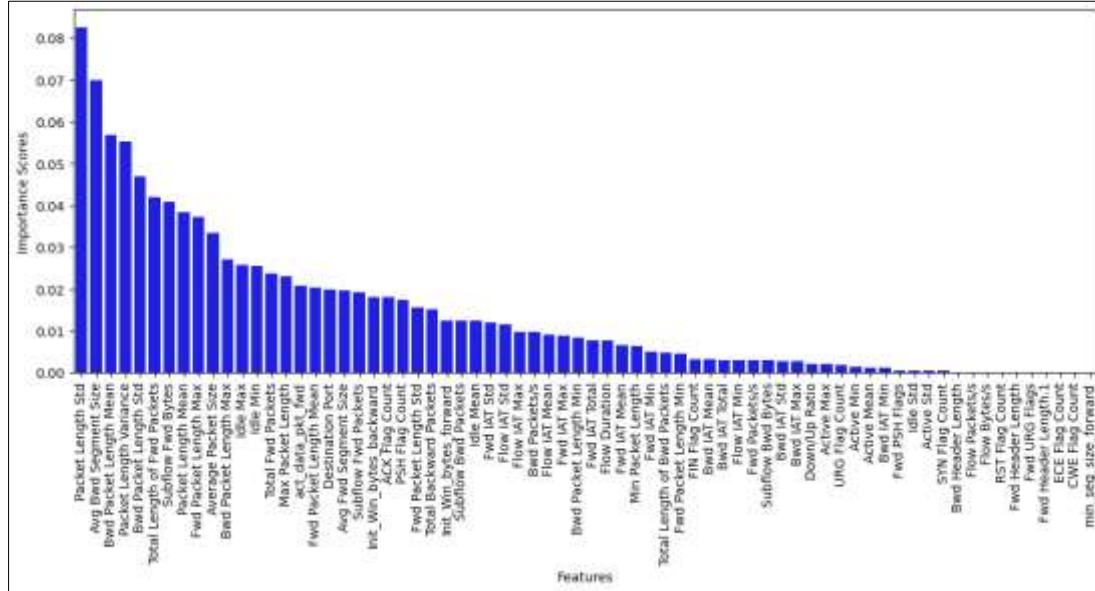
Sınıf	Etiket
Benign	0
DoS Hulk	4
DDoS	2
PortScan	10
DoS GoldenEye	3
FTP-Patator	7
DoS Slowloris	6
DoS Slowhttptest	5
SSH-Patator	11
Bot	1
Web Attack-Brute Force	12
Web Attack-XSS	14
Infiltration	9
Web Attack-SQL Injection	13
Heartbleed	8

Çizelge 7.2. UNSW-NB15 veri kümesi sınıf-etiket eşleştirmeleri

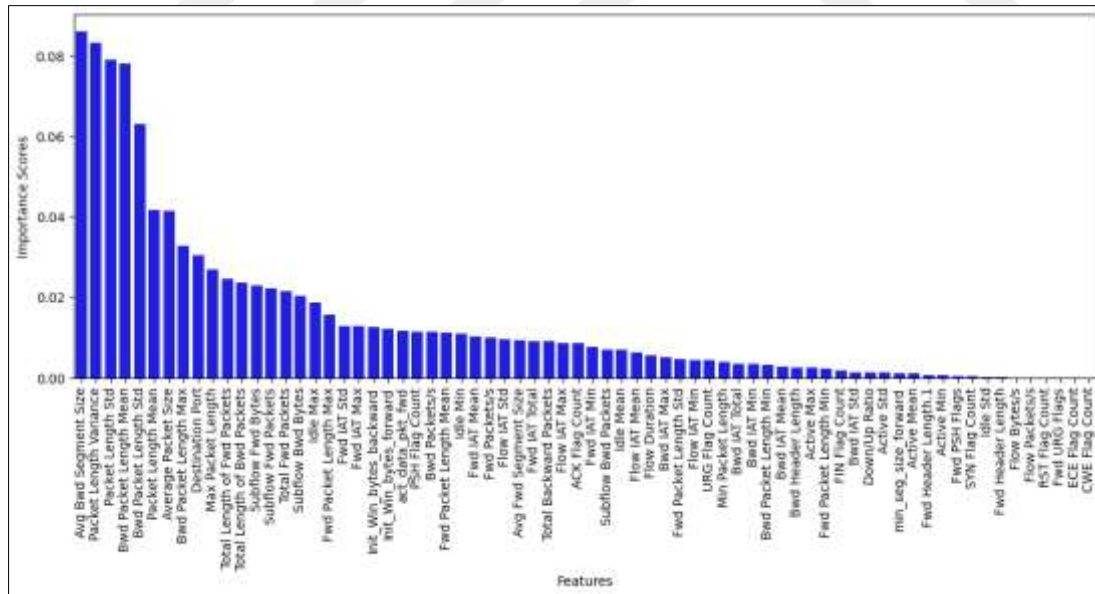
Sınıf	Etiket
Normal	6
Generic	5
Exploits	3
Fuzzers	4
DoS	2
Reconnaissance	7
Analysis	0
Backdoor	1
Shellcode	8
Worms	9

RF özellik seçimi sürecinin bir sonucu olarak, ikili ve çok sınıflı için ayrı ayrı olmak üzere CIC-IDS2017 ve UNSW-NB15 veri kümelerindeki özelliklerin önem dereceleri Şekil 7.2, Şekil 7.3, Şekil 7.4 ve Şekil 7.5’de gösterilmiştir. RF özellik seçimi yöntemiyle elde edilen önem düzeylerinin eşik değeri hem literatürdeki yapılan çalışmaların incelenmesi hem de

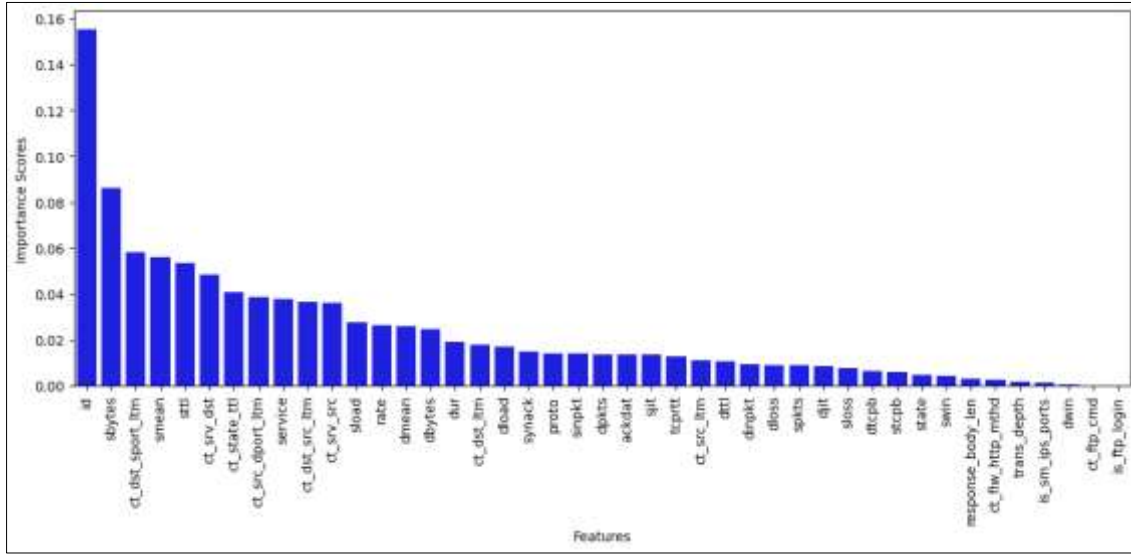
yapılan deneysel çalışmaların sonucunda 0,01 olarak belirlenmiştir. Buna göre, eşik değerinin üzerindeki özellikler sınıflandırma çalışmaları için seçilmiştir.



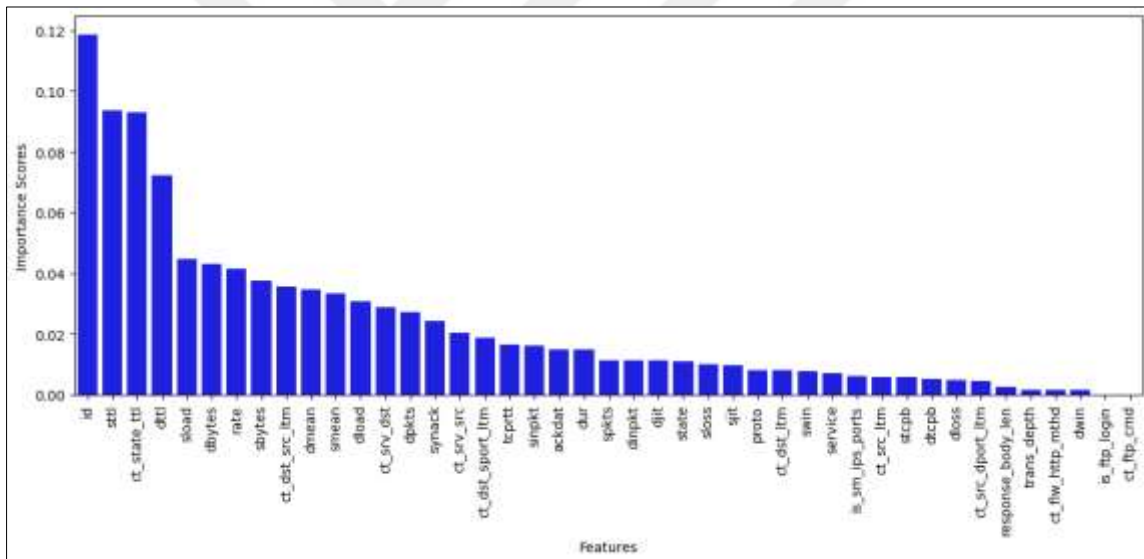
Şekil 7.2. CIC-IDS2017 çok sınıflı veri kümesi RF özellik önemlerinin gösterimi



Şekil 7.3. CIC-IDS2017 ikili veri kümesi RF özellik önemlerinin gösterimi

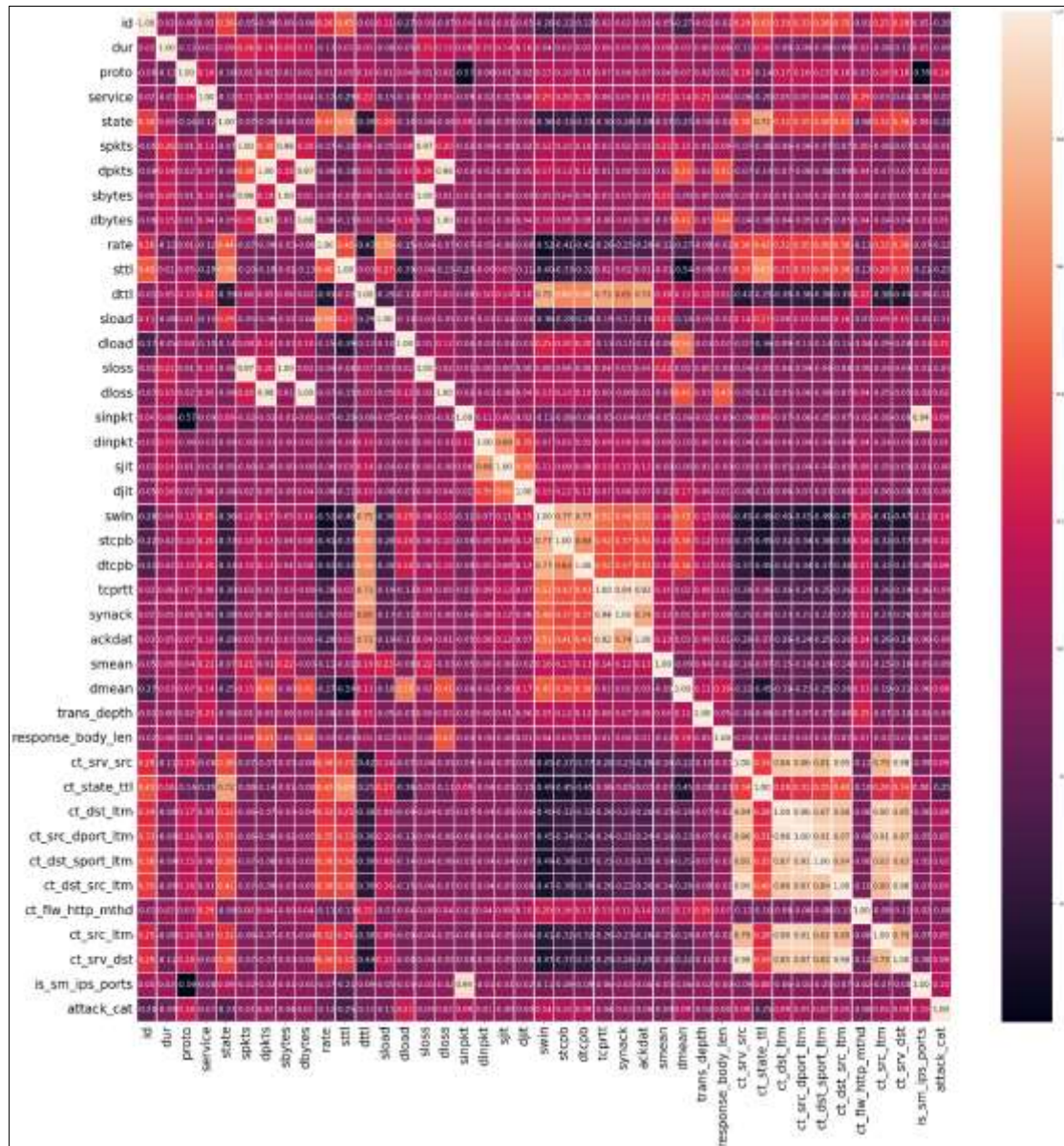


Şekil 7.4. UNSW-NB15 çok sınıflı veri kümesi RF özellik önemlerinin gösterimi

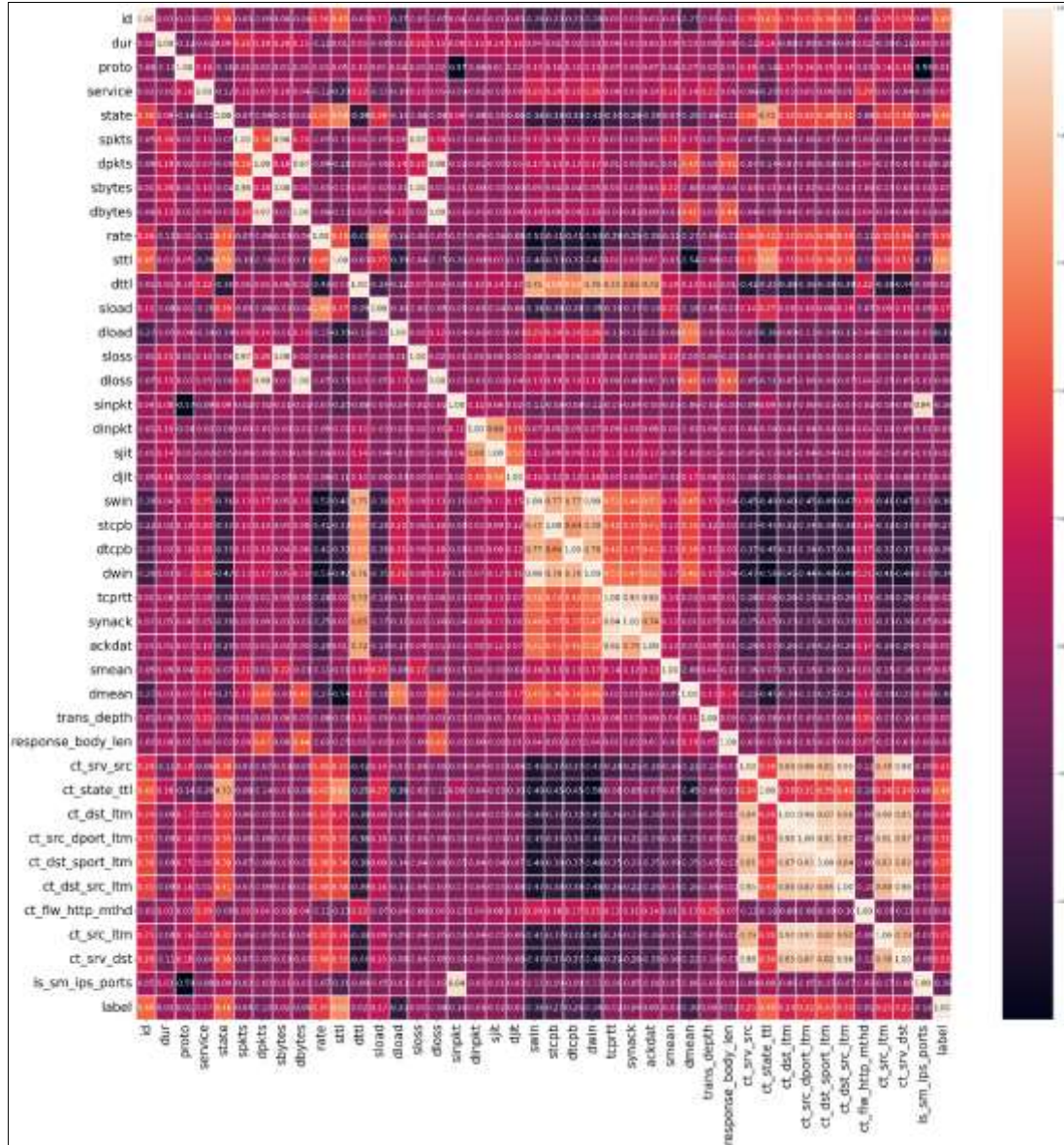


Şekil 7.5. UNSW-NB15 ikili veri kümesi RF özellik önemlerinin gösterimi

RF öznitelik seçme işleminin ardından elde edilen veri kümelerinin korelasyon matrisleri Şekil 7.6, Şekil 7.7, Şekil 7.8 ve Şekil 7.9’da gösterilmektedir. Matrislerden görüleceği üzere veri kümelerindeki birçok özellik yüksek korelasyona sahip öznitelikler bulunmaktadır. Önerilen sistemin hem işlem yükünün azaltılması hem de yüksek korelasyonlu özniteliklerden ikisinin birden bulunmasının sonuca katkı sağlamaması amacıyla yüksek korelasyonlu özniteliklerden birisi veri kümesinden çıkarılmıştır. Pearson korelasyon öznitelik seçimi yöntemiyle elde edilen önem düzeylerinin eşik değeri hem literatürdeki

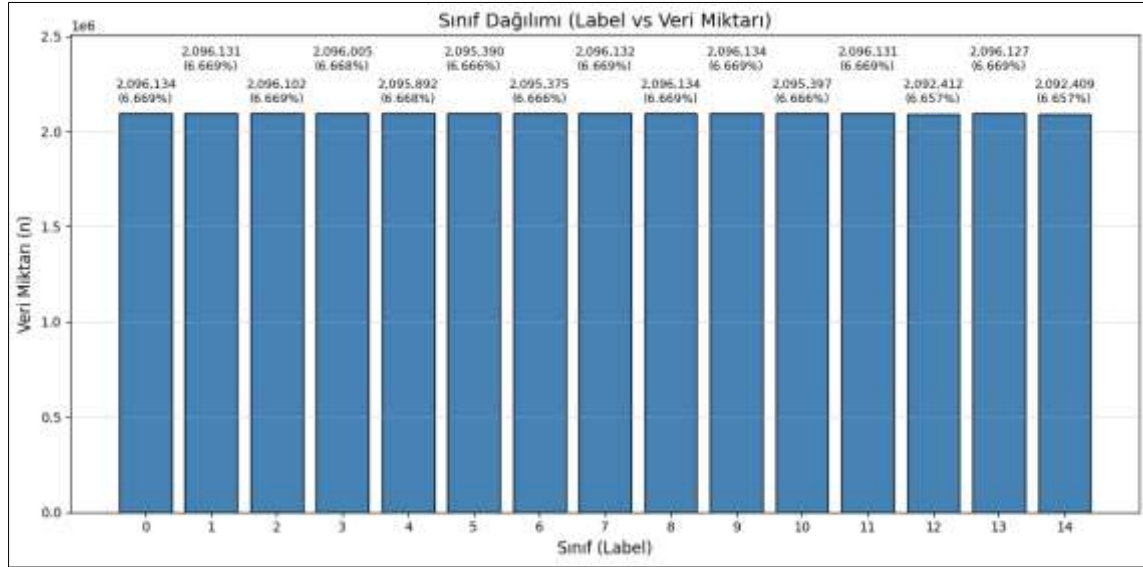


Şekil 7.8. UNSW-NB15 çok sınıflı veri kümesi Pearson korelasyonlarının gösterimi

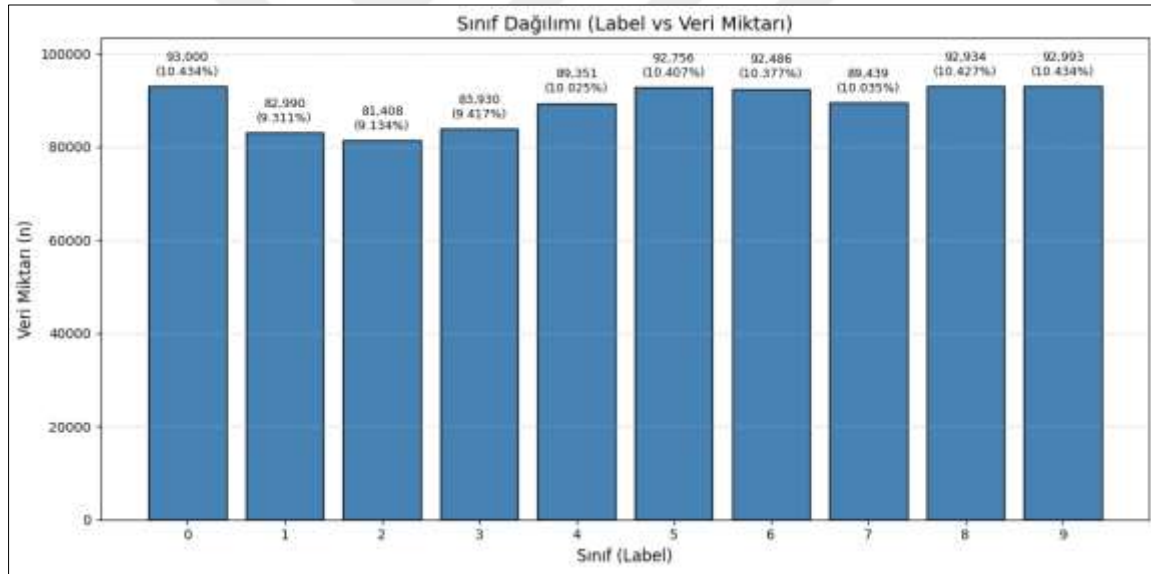


Şekil 7.9. UNSW-NB15 ikili veri kümesi Pearson korelasyonlarının gösterimi

Şekil 7.10 ve Şekil 7.11’de STL veri dengeleme işlemi sonucunda oluşan veri kümelerinin sınıf dağılımları gösterilmektedir.



Şekil 7.10. CIC-IDS2017 veri kümesi için veri dengeleme işlemi sonrası veri dağılımı



Şekil 7.11. UNSW-NB15 veri kümesi için veri dengeleme işlemi sonrası veri dağılımı

7.2. Model Parametreleri

Tez çalışmasında veri setlerinin işlem maliyetlerinin azaltılması ve kullanılan platformlarda işlenebilmesi amacıyla sınıf dağılımları korunacak şekilde her iki veri kümesinden örnekler alınarak tüm deneysel çalışmalar bu veri kümeleri üzerinden yürütülmüştür. Veri kümeleri %80 eğitim seti ve %20 test seti olarak ikiye ayrılmıştır. Modellerde eğitim ve test setlerinin aynı örnekleri kullanmasını sağlamak için seed değeri “42” olarak belirlenmiştir. Kurulan modellerde varsayılan parametreler haricinde girilen parametre değerleri bulunmaktadır.

Önerilen hibrit LightGBM-GCN algoritması için yapılan tuning işlemi sonucuna göre en iyi sonuçları veren hiperparametreler Çizelge 7.3 ve Çizelge 7.4’de sunulmuştur. LightGBM ve GCN algoritmalarına ait diğer hiper parametreler varsayılan değerlerde bırakılmıştır.

Random Forest algoritması ile kurulan modelde maxDept= 20 ve maxBins=50 olarak belirlenmiştir. Maksimum derinlik ağacın sıklığını ifade eder ve derinlik artıkça veriler hakkında daha fazla bilgi yakalama şansı artar. Decision Tree algoritması ile kurulan modelde de maxDept= 20 ve maxBins=50 olarak belirtilmiştir. Naive Bayes algoritması ile kurulan modelde smooting ve Naive Bayes model tipi parametreleri seçilmiştir. Smooting değeri 1.0, model tipi ise GaussianNB’dır. Logistic Regression modelinde kullanılan tüm parametreler varsayılan olarak kullanılmıştır. MLP ile kurulan modelde dört katman vardır. Giriş katmanında veri setlerinde kullanılan öznetelik sayısına eşit sayıda sinir hücresi kullanılmıştır. Gizli katmanlarda 5 adet sinir hücresi yer alır. Çalışmada çıkış katmanı UNSW-NB15 verisetinde 10, CIC-IDS2017 veri setinde multiclass sınıflandırma için 15 ve binary sınıflandırma için 2 değerlerini almıştır. Ayrıca MLP modeli için maksimum iterasyon sayısı 100 ve blok boyutu 128 olarak belirlenmiştir.

Long Short-Term Memory (LSTM) ile kurulan modelde üç katman vardır. Giriş katmanında ve gizli katmanda 128 sinir hücresi kullanılmıştır. Gated Recurrent Units (GRU) ile kurulan modelde üç katman vardır. Giriş katmanında 128 ve gizli katmanda 64 sinir hücresi kullanılmıştır. Convolutional Neural Network (CNN-1D) ile kurulan modelin birinci ve ikinci evrişim katmanlarında filter 64 ve kernel size 2 olarak alınmıştır. Pooling katmanlarında stride parametresi 2’dir. Son evrişim katmanında filter 128 ve kernel size 2’dir. CNN-LSTM modelinde bir evrişim katmanı ve bir LSTM katmanı bulunmaktadır. Birinci evrişim katmanında filter 128 ve kernel size 2 kabul edilmiştir. Pooling katmanlarında stride parametresi 2’dir. LSTM katmanında 64 nöron yer almaktadır. Daha sonra 128 nörondan oluşan bir Dense katmanı kullanılmıştır. CNN-GRU modelinde bir evrişim katmanı ve bir GRU katmanı bulunmaktadır. Birinci evrişim katmanında filter 128 ve kernel size 2 kabul edilmiştir. Pooling katmanlarında stride parametresi 2’dir. GRU katmanında 64 nöron yer almaktadır. Daha sonra 128 nörondan oluşan bir Dense katmanı kullanılmıştır. GAT, GraphSAGE ve GIN algoritmaları için 2 evrişim katmanı kullanılmıştır. Evrişim katmanlarında filter 128, aktivasyon fonksiyonu olarak ReLu, optimizasyon algoritması olarak ise Adam kullanılmıştır.

Keras kütüphanesi kullanılarak kurulan LSTM, GRU, CNN-1D, CNN-LSTM ve CNN-GRU modellerinde, Giriş ve gizli katmanlarda ReLu aktivasyon fonksiyonu kullanılmıştır. Batch_size parametresi 32 olarak ayarlanmıştır. Çıkış katmanı UNSW-NB15 verisetinde 10, CIC-IDS2017 veri setinde multiclass sınıflandırma için 15 ve binary sınıflandırma için 1 değerlerini almıştır. Çıkış katmanında multiclass sınıflandırmada aktivasyon fonksiyonu Softmax ve binary sınıflandırmada Sigmoid olarak belirlenmiştir. Modelde kullanılan kayıp fonksiyonu multiclass sınıflandırmada “categorical_crossentropy”, binary sınıflandırmada “binary_crossentropy” dir. Optimizasyon algoritması “Adam” dır. Modeller 50 epoch için çalıştırılmıştır.

Çizelge 7.3. LightGBM hiperparametreleri

Hiperparametreler	Değerler
Boosting type	gbdt
Learning rate	0,05
Number of leaves (Num_leaves)	31
Number of estimators (N_estimators)	500
Random state	42
Min_child samples	20
objective	Multiclass/binary

Çizelge 7.4. GCN hiperparametreleri

Hiperparametreler	Değerler
Hidden layer	3
Hidden_dim	256
Dropout	0,5
weight_decay	5e-4
Learning rate	0,001

7.3. Sınıflandırma Adımı Sonuçları

Tez çalışmasının bu bölümünde önerilen açıklanabilir saldırı tespit sisteminin CIC-IDS2017 ve UNSW-NB15 veri kümeleri üzerinde gerçekleştirilen sınıflandırma sonuçları paylaşılmıştır. Deneysel çalışmaların sonucunda, her iki veri kümesi üzerinde gerçekleştirilen çok sınıflı (multiclass) ve ikili (binary) sınıflandırma deneylerinin sonuçları ayrıntılı biçimde analiz edilmiştir. Deneysel çalışmalar, klasik makine öğrenmesi yöntemlerini, derin öğrenme modellerini, graf sinir ağı tabanlı yaklaşımları ve bu yöntemlerin hibrit olarak birleştirildiği modelleri kapsamaktadır.

Tez çalışması kapsamında, önerilen LightGBM-GCN hibrit yaklaşımının veri kümeleri üzerinde çok sınıflı (multiclass) ve ikili (binary) olmak üzere sınıflandırma performansı ölçülmüştür. Elde edilen sonuçlar çeşitli makine öğrenmesi ve derin öğrenme algoritmaları ile karşılaştırılmıştır. Aynı zamanda, önerilen LightGBM algoritmasının hibrit derin öğrenme yöntemlerine karşı başarısı da değerlendirilmiştir. Önerilen hibrit yaklaşımın sınıflandırma performansı literatürde var olan 19 farklı model ile karşılaştırılmıştır.

7.3.1. CIC-IDS2017 veri kümesi sınıflandırma sonuçları

Bu bölümde CIC-IDS2017 veri kümesi üzerinde yürütülen çok sınıflı ve ikili sınıflandırma sonuçları paylaşılmış ve değerlendirilmiştir.

Çizelge 7.5, önerilen yaklaşımın çeşitli makine öğrenmesi ve derin öğrenme algoritmalarının CIC-IDS2017 veri seti üzerinde gerçekleştirilen çok sınıflı sınıflandırmasını, Çizelge 7.6 ise ikili sınıflandırma sonucunda elde edilen Accuracy, Precision, Recall ve F-Score parametrelerinin karşılaştırmalarını sunmaktadır. Bununla birlikte, her bir yöntem için elde edilen karmaşıklık matrisleri ve ROC eğrilerinin çıktıları Ek-C ve Ek-D’de sunulmuştur.

CIC-IDS2017 veri kümesi çok sınıflı sınıflandırma sonuçları değerlendirildiğinde klasik makine öğrenmesi modelleri içinde en yüksek performans LightGBM (Accuracy=0.9794) ve XGBoost (Accuracy=0.9774) ile elde edilmiştir. Bu durum, boosting algoritmalarının tabular veri yapısındaki güçlü genelleme yetenekleri ile uyumludur. Decision Tree ve Random Forest modelleri de yüksek performans göstermelerine rağmen boosting yöntemlerinin hassas ayırıştırma kapasitesine ulaşamamıştır. Logistic Regression ve Naive Bayes ise doğrusal model yapısı ve bağımsızlık varsayımı nedeniyle çok sınıflı karmaşık saldırı tiplerini ayırt etmekte zorlanmış, özellikle Naive Bayes modelinin Recall ve Specificity değerleri düşük kalmıştır.

Çizelge 7.5. CIC-IDS2017 çok sınıflı sınıflandırma sonuçları

Modeller	Precision	Recall	F1-Score	Specificity	Accuracy
Decision Tree	0,9737	0,9706	0,9701	0,9979	0,9706
Random Forest	0,9763	0,9749	0,9747	0,9982	0,9749
Naive Bayes	0,8131	0,8174	0,7996	0,9870	0,8174
Logistic Regression	0,8165	0,8449	0,8210	0,9889	0,8449
MLP	0,9433	0,9441	0,9433	0,9960	0,9441
CNN	0,9512	0,9248	0,9107	0,9947	0,9248
LSTM	0,9333	0,9249	0,9154	0,9946	0,9249
GRU	0,9422	0,9337	0,9258	0,9953	0,9337
CNN-GRU	0,9528	0,9269	0,9127	0,9948	0,9269
CNN-LSTM	0,9557	0,9300	0,9158	0,9951	0,9300
XGBoost	0,9780	0,9774	0,9774	0,9984	0,9774
LightGBM	0,9796	0,9794	0,9794	0,9985	0,9794
GCN	0,8164	0,8394	0,8157	0,9885	0,8395
GAT	0,8827	0,8794	0,8608	0,9914	0,8795
GIN	0,8895	0,8611	0,8393	0,9901	0,8611
GraphSAGE	0,7939	0,8118	0,7887	0,9866	0,8119
LightGBM+GIN	0,9122	0,9101	0,9090	0,9936	0,9101
XGBoost+GCN	0,9788	0,9788	0,9788	0,9985	0,9788
XGBoost+GIN	0,9326	0,9062	0,8912	0,9933	0,9062
Önerilen yöntem	0,9797	0,9797	0,9797	0,9986	0,9797

Çizelge 7.6. CIC-IDS2017 ikili sınıflandırma sonuçları

Modeller	Precision	Recall	F1-Score	Specificity	Accuracy
Decision Tree	0,9764	0,9772	0,9768	0,9775	0,9773
Random Forest	0,9810	0,9889	0,9849	0,9809	0,9849
Naive Bayes	0,6783	0,8752	0,7642	0,6048	0,7367
Logistic Regression	0,9424	0,8409	0,8888	0,9511	0,8974
MLP	0,9891	0,9657	0,9773	0,9899	0,9781
CNN	0,9898	0,9519	0,9705	0,9907	0,9718
LSTM	0,9212	0,9204	0,9205	0,9022	0,9204
GRU	0,9854	0,9372	0,9607	0,9868	0,9626
CNN-GRU	0,9797	0,9038	0,9402	0,9821	0,9439
CNN-LSTM	0,9817	0,9641	0,9728	0,9829	0,9737
XGBoost	0,9992	0,9967	0,9980	0,9992	0,9980
LightGBM	0,9980	0,9980	0,9980	0,9992	0,9980
GCN	0,9520	0,8436	0,8945	0,9571	0,9001
GAT	0,8415	0,8465	0,8440	0,8393	0,8429
GIN	0,9641	0,8515	0,9043	0,9681	0,9095
GraphSAGE	0,9642	0,8535	0,9055	0,9680	0,9105
LightGBM+GIN	0,9960	0,9960	0,9960	0,9960	0,9960
XGBoost+GCN	0,9968	0,9968	0,9968	0,9968	0,9968
XGBoost+GIN	0,9944	0,9960	0,9952	0,9944	0,9952
Önerilen yöntem	0,9999	0,9999	0,9999	0,9999	0,9999

Derin öğrenme modelleri incelendiğinde, MLP, CNN, LSTM, GRU ve hibrit CNN-LSTM ve CNN-GRU yapılarının boosting algoritmalarına göre daha düşük performans sergilediği görülmektedir. Bunun temel nedeni, CIC-IDS2017 veri kümesinin görüntü veya zaman serisi gibi doğal bir sinyal içermemesi, daha çok istatistiksel akış özelliklerinden oluşan yüksek boyutlu bir tablo yapısına sahip olmasıdır. CNN ve RNN tabanlı modeller bu tür veri üzerinde ek mühendislik yapılmadan optimum düzeyde performans gösteremez. Bu nedenle, derin öğrenme modelleri çoğunlukla 0.92–0.95 aralığında seyreden doğruluk değerlerine ulaşmıştır.

Grafik sinir ağı (GNN) tabanlı yöntemler olan GCN, GAT, GIN ve GraphSAGE çok sınıflı senaryoda klasik yöntemlere göre daha düşük performans göstermiştir. Bunun nedeni, CICIDS2017'nin doğal bir grafik veri kümesi olmaması ve graf yapılarının sonradan KNN veya benzerlik matrisleri kullanılarak oluşturulmasıdır.

Hibrit modeller incelendiğinde, boosting ile GNN birleşiminin belirgin performans artışı sağladığı görülmektedir. Örneğin XGBoost+GCN hibrit modeli çok sınıflı senaryoda 0.9788 Accuracy elde ederek bağımsız XGBoost modelini dahi geride bırakmıştır. Benzer şekilde LightGBM+GIN modeli bağımsız GIN'e göre ciddi bir iyileşme göstermiştir. Ancak tüm hibrit modeller arasında en yüksek performans, önerilen LightGBM+GNN tabanlı hibrit yaklaşım ile elde edilmiştir. Önerilen yöntem, çok sınıflı sınıflandırmada Precision, Recall, F1-Score ve Accuracy metriklerinde 0.9797 gibi son derece başarılı bir sonuç üretmiştir. Tüm metriklerin eşit olması, modelin hem saldırı hem normal trafik üzerinde dengeli ve kararlı bir performans gösterdiğini kanıtlamaktadır.

İkili sınıflandırma senaryosunda tüm saldırı tipleri tek bir 'Attack' sınıfı altında birleştirilmektedir. Bu nedenle sınıf ayrımı çok daha belirgin hale gelmekte, modelin karar sınırları keskin bir pozitif–negatif ayrımına dönüşmektedir. Bu durum, özellikle boosting ve hibrit modellerde belirgin performans artışına yol açmıştır.

Klasik makine öğrenmesi yöntemleri değerlendirildiğinde, Random Forest (Accuracy=0.9849), Decision Tree (Accuracy=0.9773) gibi yöntemler oldukça başarılı sonuçlar elde etmiştir. Ancak en yüksek performans yine boosting tabanlı yöntemlerde görülmektedir. LightGBM ve XGBoost ikili sınıflandırmada 0.9980 Accuracy değerine

ulaşarak neredeyse hatasız sınıflandırma yapmıştır. Precision ve Recall değerlerinin birbirine çok yakın veya eşit olması, modele yüksek güvenilirlik kazandırmaktadır.

Derin öğrenme yöntemleri ikili sınıflandırmada da genel olarak başarılı olmakla birlikte, boosting algoritmalarının gerisinde kalmıştır. MLP (0.9781), CNN (0.9718), LSTM (0.9204) ve GRU (0.9626) gibi modeller yüksek doğruluk değerlerine ulaşsa da, tablo verisinin doğasına uygun olmayan öğrenme yapıları nedeniyle boosting kadar hassas genelleme yapamamaktadır. CNN-LSTM ve CNN-GRU gibi hibrit derin öğrenme mimarileri belirli avantajlar sağlasa da boosting modellerinin doğruluğuna ulaşamamıştır.

GNN modellerinin ikili sınıflandırma performansları çok sınıflı senaryoya göre kısmen artmış olsa da bağımsız boosting teknolojilerinin oldukça gerisinde kalmıştır. GCN (0.9001), GAT (0.8429), GIN (0.9095) ve GraphSAGE (0.9105) gibi değerler, graf tabanlı yöntemlerin bu veri kümesinde doğal bir temsil avantajına sahip olmadığını ve performanslarının büyük ölçüde graf yapısının kalitesine bağlı olduğunu göstermektedir.

Hibrit yöntemler ikili sınıflandırmada çok daha çarpıcı sonuçlar üretmiştir. LightGBM+GIN 0.9960, XGBoost+GIN 0.9952 ve özellikle XGBoost+GCN 0.9968 gibi çok yüksek doğruluk değerlerine ulaşılmıştır. Bu yöntemler, boosting'in öznitelik bazlı ayrıştırma gücünü, GNN modellerinin topolojik zenginleştirme kapasitesi ile birleştirerek saldırı ve normal trafiğin daha doğru ayrışmasını sağlamıştır.

Önerilen hibrit yöntem çok sınıflı sınıflandırmada olduğu gibi ikili sınıflandırmada da tüm modellerin üzerinde bir performans sergileyerek Precision, Recall, F1-Score, Specificity ve Accuracy metriklerinin tamamında 0.9999 değerine ulaşmıştır. Bu sonuç, modelin hem normal hem saldırı trafiğini neredeyse hatasız ayırdığını göstermektedir. Bu düzeydeki bir başarı, boosting tabanlı öznitelik temsili GNN tabanlı grafik temsil ile harmanlanmasının doğrudan sonucudur.

7.3.2. UNSW-NB15 veri kümesi sınıflandırma sonuçları

Bu bölümde CIC-IDS2017 veri kümesi üzerinde yürütülen çok sınıflı ve ikili sınıflandırma sonuçları paylaşılmış ve değerlendirilmiştir.

Çizelge 7.7, önerilen yaklaşımın çeşitli makine öğrenmesi ve derin öğrenme algoritmalarının UNSW-NB15 veri seti üzerinde gerçekleştirilen çok sınıflı sınıflandırmasını, Çizelge 7.8 ise ikili sınıflandırma sonucunda elde edilen Accuracy, Precision, Recall ve F-Score parametrelerinin karşılaştırmalarını sunmaktadır. Her bir yöntem için elde edilen karmaşıklık matrisleri ve ROC eğrilerinin çıktıları Ek-A ve Ek-B’de sunulmuştur.

Çizelge 7.7. UNSW-NB15 çok sınıflı sınıflandırma sonuçları

Modeller	Precision	Recall	F1-Score	Specificity	Accuracy
Decision Tree	0,8119	0,8030	0,8045	0,9789	0,8030
Random Forest	0,8343	0,8258	0,8269	0,9812	0,8258
Naive Bayes	0,4987	0,4646	0,4123	0,9409	0,4646
Logistic Regression	0,5581	0,5629	0,5461	0,9513	0,5629
MLP	0,7388	0,7259	0,7260	0,9701	0,7259
CNN	0,7047	0,6827	0,6722	0,9645	0,6827
LSTM	0,7633	0,7443	0,7478	0,9728	0,7443
GRU	0,7618	0,7468	0,7447	0,9722	0,7468
CNN-GRU	0,7326	0,7213	0,7206	0,9698	0,7213
CNN-LSTM	0,7587	0,7445	0,7451	0,9724	0,7445
XGBoost	0,8338	0,8260	0,8267	0,9814	0,8260
LightGBM	0,8512	0,8440	0,8449	0,9832	0,8440
GCN	0,6920	0,6645	0,6622	0,9630	0,6645
GAT	0,6749	0,6518	0,6406	0,9611	0,6518
GIN	0,7050	0,6849	0,6842	0,9656	0,6849
GraphSAGE	0,6770	0,6641	0,6566	0,9627	0,6641
LightGBM+GIN	0,8486	0,8436	0,8449	0,9832	0,8436
XGBoost+GCN	0,8378	0,8314	0,8326	0,9819	0,8314
XGBoost+GIN	0,8128	0,8031	0,8049	0,9789	0,8031
Önerilen yöntem	0,8499	0,8444	0,8457	0,9833	0,8444

UNSW-NB15 veri kümesi çok sınıflı sınıflandırma sonuçları değerlendirildiğinde, Decision Tree (Accuracy = 0,8030) ve Random Forest (Accuracy = 0,8258) yöntemleri tabular veri setlerine doğal olarak uyum sağladıkları için oldukça başarılı sonuçlar üretmiştir. XGBoost (Accuracy = 0,8260) ve LightGBM (Accuracy = 0,8440) gibi boosting tabanlı modeller, klasik yöntemler arasında en yüksek performansa ulaşmıştır. Boosting algoritmaları, ardışık hatayı azaltan öğrenme yaklaşımı sayesinde non-lineer ve karmaşık karar sınırlarını oldukça başarılı biçimde modelleyerek yüksek performanslı sonuçlar elde etmişlerdir. Naive Bayes (Accuracy = 0,4646) ve Logistic Regression (Accuracy = 0,5629) ise belirgin biçimde geride kalmıştır. Bu düşük performansın nedeni modelin kapasitesi ve veri setinin doğasıyla ilişkilidir. Naive Bayes’in bağımsızlık varsayımı UNSW-NB15 gibi korelasyonlu bir veri setinde geçerli olmamıştır. Buna karşın Logistic Regression, doğrusal karar sınırlarıyla çok modlu saldırı tiplerini doğru şekilde temsil edememiştir.

Derin öğrenme yöntemleri de çok sınıflı veride orta düzey performans sergilemiştir. UNSW-NB15 veri kümesinde uzamsal/zamansal bağıllık düşük seviyede olduğu için CNN veya RNN mimarileri yüksek performans sergileyememiştir. Bazı ağ bağlantı davranışlarının zaman temelli olmasından dolayı LSTM ve GRU, CNN'e göre daha iyi performans göstermiştir, ancak bu avantaj boosting modellerini geçecek seviyeye ulaşamamıştır. Dolayısıyla, derin öğrenme temelli modeller boosting modellerinin gerisinde kalmıştır.

GNN modelleri (GCN, GAT, GIN, GraphSAGE), çok sınıflı senaryoda boosting ve bazı derin öğrenme modellerinin gerisinde kalmıştır. Bu sonuçların temel nedeni UNSW-NB15'in doğal bir grafik veri seti olmamasından dolayıdır. Öznelik benzerliğine dayalı KNN ile oluşturulan yapay graf yapılarında farklı sınıflara ait örneklerin birbirine yakın konumlanması modelin karar sınırını belirsizleştirebilir ve gürültülü komşular bilgi karışımına yol açabilmektedir. Dolayısıyla GNN'lerin çok sınıflı UNSW-NB15 üzerinde düşük performans göstermesi veri setinin yapısal özelliklerinden kaynaklanmaktadır.

Hibrit modellerin performansı değerlendirildiğinde, boosting tabanlı öğrenmeye GNN tabanlı ilişkisel bilgiyi eklemenin sınırlı ama anlamlı bir katkı sağladığı görülmektedir. Bununla birlikte en iyi sonuç önerilen yöntem tarafından elde edilmiştir.

Çizelge 7.8. UNSW-NB15 ikili sınıflandırma sonuçları

Modeller	Precision	Recall	F1-Score	Specificity	Accuracy
Decision Tree	0,9592	0,9148	0,9365	0,9607	0,9376
Random Forest	0,9293	0,9483	0,9387	0,9271	0,9378
Naive Bayes	0,7404	0,9438	0,8298	0,6659	0,8055
Logistic Regression	0,8839	0,9082	0,8959	0,8795	0,8939
MLP	0,9608	0,9539	0,9573	0,9607	0,9573
CNN	0,9866	0,9506	0,9683	0,9870	0,9687
LSTM	0,9604	0,9604	0,9604	0,9588	0,9604
GRU	0,9759	0,9683	0,9721	0,9758	0,9721
CNN-GRU	0,9643	0,9300	0,9468	0,9653	0,9475
CNN-LSTM	0,9779	0,9659	0,9719	0,9780	0,9719
XGBoost	0,9896	0,9823	0,9859	0,9896	0,9859
LightGBM	0,9865	0,9864	0,9864	0,9884	0,9864
GCN	0,8761	0,8807	0,8784	0,8762	0,8784
GAT	0,7853	0,8893	0,8341	0,7580	0,8235
GIN	0,9342	0,8900	0,9116	0,9385	0,9145
GraphSAGE	0,8873	0,8875	0,8874	0,8888	0,8881
LightGBM+GIN	0,9833	0,9821	0,9827	0,9970	0,9828
XGBoost+GCN	0,9610	0,9517	0,9563	0,9616	0,9567
XGBoost+GIN	0,9555	0,9549	0,9552	0,9558	0,9554
Önerilen yöntem	0,9974	0,9912	0,9943	0,9974	0,9943

UNSW-NB15 veri kümesinde ikili sınıflandırma senaryosunun sonuçları değerlendirildiğinde, veri kümesindeki yapısal fark nedeniyle tüm modellerin performansı çok sınıflı senaryoya göre belirgin biçimde yükselmiştir. Klasik makine öğrenmesi algoritmalarının sonuçlarında Naive Bayes'in Recall değerinin çok yüksek (0,9438) ancak Specificity'nin çok düşük (0,6659) olması, modelin çok sayıda yanlış alarm ürettiğini göstermektedir. Bu durum STS senaryolarında özellikle istenmeyen bir durumdur.

Boosting yöntemleri ikili senaryoda da yüksek doğruluk değerleri üretmişlerdir. Bu modeller hem Precision hem Recall tarafında dengeli performans sergileyerek normal trafikleri ve saldırı sınıflarını yüksek doğrulukla tespit etmiştir.

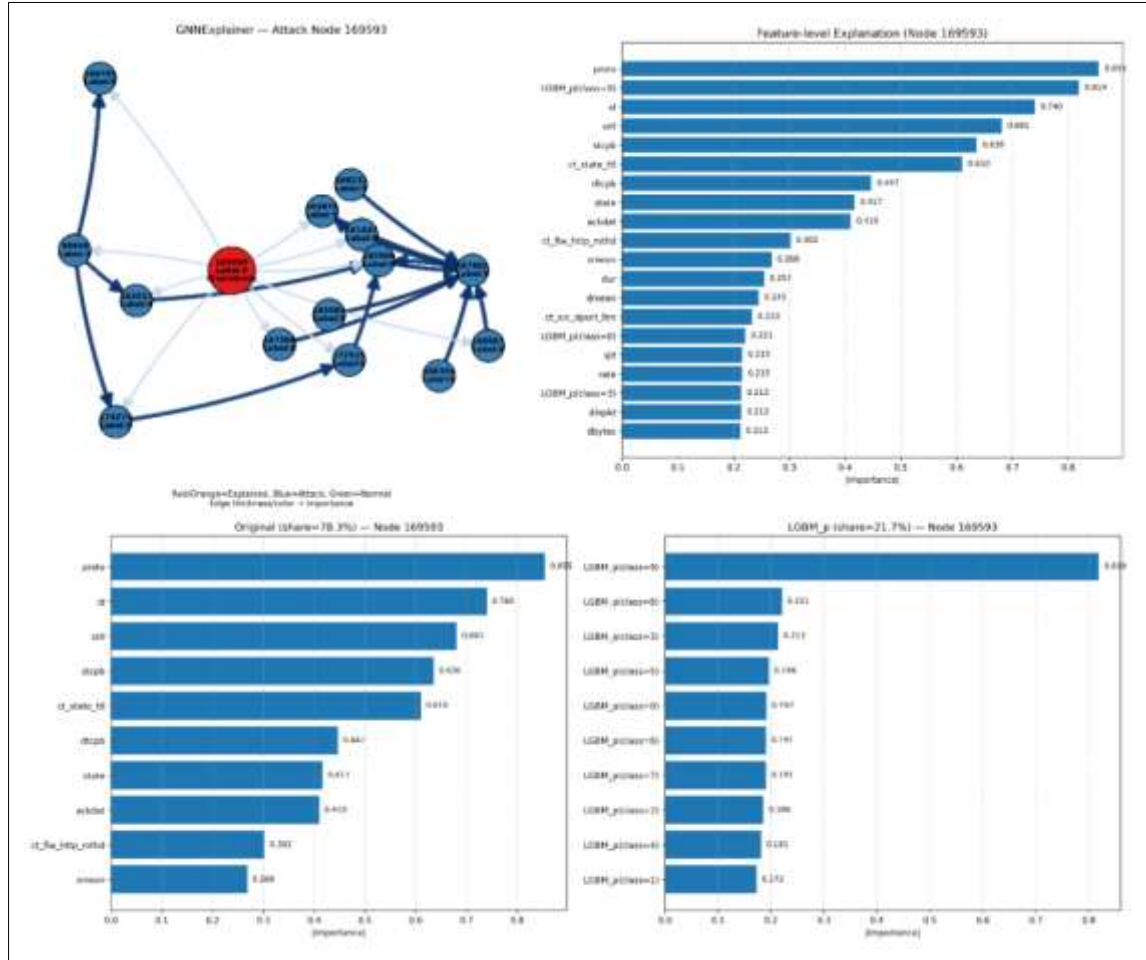
İkili sınıflandırmada derin öğrenme modelleri yüksek doğruluklara ulaşmıştır. Bu başarı artışı, ikili sınıflandırmada karar sınırının daha net olmasıyla ilişkilidir. GNN modelleri ikili sınıflamada sınırlı ama çok sınıflıya göre daha iyi performans göstermiştir. Bununla birlikte, GNN modelleri boosting modellerinin gerisinde kalmıştır. Veri seti graf tabanlı olmadığı sürece bu sonuç doğaldır. Binary senaryoda da hibrit modeller güçlü performans sergileyerek tüm modeller içinde en yüksek başarıya karşılık gelmekte ve modelin saldırı/normal ayrımını neredeyse hatasız gerçekleştirdiğini göstermektedir.

7.4. Açıklanabilirlik Adımı Sonuçları

Tez çalışmasının bu bölümünde önerilen sisteminin açıklama adımı sonucunda elde edilen deneyel sonuçlar paylaşılmıştır.

7.4.1. UNSW-NB15 veri kümesi açıklama katmanı sonuçları

Bu bölümde önerilen sistemin UNSW-NB15 veri kümesi kullanılarak sınıflandırılan bir saldırı örneği üzerinden açıklanabilirlik çıktıları verilmiş ve çıktılar detaylı olarak değerlendirilmiştir. Bununla birlikte, önerilen sistemin normal olarak sınıflandırılmış bir örnek üzerindeki açıklamaları da gösterilmiştir.



Şekil 7.12. UNSW-NB15 veri kümesi örnek bir saldırı trafiği için GNNExplainer açıklama çıktısı

Şekil 7.12, önerilen açıklanabilir saldırı tespit sisteminde saldırı olarak sınıflandırılmış bir örneğe ait açıklanabilirlik çıktısını göstermektedir. Sol tarafta, açıklanan saldırı düğümünü ve onunla ilişkili komşuluk altgrafını içeren bağlantı haritası, sağ tarafta ise bu düğümün ait özellik önem dereceleri yer almaktadır. Ağ topolojisinde kırmızı düğüm açıklanmakta olan saldırı örneğini, mavi düğümler aynı veya benzer saldırı etiketine sahip komşuları göstermektedir. Kenar kalınlığı ve renk yoğunluğu ise, kenarların GCN modelinin kararına katkı oranını ifade etmektedir. Kalın ve koyu mavi kenarlar, modelin bu bağlantılardan yüksek oranda bilgi aktardığını ifade etmektedir.

GNNExplainer, her düğümün tahmininde en kritik rolü oynayan altgrafı ve öznelilikleri belirlemektedir. Bu durumda, Worms sınıfına ait belirtilen örnek için modelin kararında baskın olan komşular ve özellikler şu şekilde öne çıkmaktadır:

- Komşuluk yapısı olarak incelenen örneğin, benzer saldırı tipine (Label: 9) sahip diğer düğümlerle yüksek yoğunluklu bağlantılar kurmadığı görülmektedir. Bu durum, GCN'in saldırı kararını komşulardan gelen bilgilerden ziyade kendi sınıflandırması sonucunda verdiğini göstermektedir. Bununla birlikte, oluşturulan alt graf içerisindeki tüm akışların aynı sınıfa atanması ve bu akışlarla aynı gratta yer alması, bu örneğin davranışsal olarak diğer saldırı örnekleriyle benzerlikler taşıdığını işaret etmektedir.
- Modelin öğrenme stratejisi olarak ise GCN modeli, sadece öznitelik değerlerine değil, aynı zamanda topolojik bağlama da duyarlıdır. Bu grafikte, atak kümelenmesi olarak adlandırılabilen kümelenmelerin olduğu gözlenmektedir. İncelenen örnek, bu kümelenme içerisinde merkezi konumda yer alarak modelin yüksek performansla saldırı tahmini yapmasını sağlamıştır.

Açıklamanın sağ tarafındaki yer alan çubuk grafikte ise, GNNExplainer tarafından belirlenen özniteliklerin göreceli katkı oranları verilmiştir. Bu bulgular, modelin özellikle ağ protokol özelliklerine ve paket tabanlı zamanlama değişkenlerine yüksek duyarlılık gösterdiğini ortaya koymaktadır. “proto” özneliğinin baskınlığı, modelin saldırı trafiğini protokol düzeyinde ayırt ettiğini göstermektedir.

Bununla birlikte, açıklamanın alt kısmında yer alan çubuk grafiklerde ise belirtilen örneğin sınıflandırılmasındaki orijinal öznelilik katkısı (Original share = %78,3) ile LightGBM tahmin çıktısının katkısı (LGBM_p share = %21,7) olarak ikiye ayrıldığı gösterilmektedir. Sol grafik (Original share), GNN'in doğrudan özneliliklerden öğrendiği katkıları göstermektedir. Burada proto, sttl, ct_state_ttl, ackdat gibi temel ağ değişkenlerinin baskın olduğu görülmektedir. Sağ grafik (LGBM_p share) ise hibrit modelde LightGBM'den gelen sınıf olasılıklarının GCN kararına nasıl etki ettiğini göstermektedir. En yüksek etki LGBM_p (class=9) olasılığına aittir (0,819), bu da LightGBM'in Worms sınıfına yüksek güvenle tahmin yaptığı ve bu bilginin GCN tarafından güçlendirilerek nihai karara dahil edildiğini göstermektedir. Diğer sınıflara ait olasılıklar (ör. class=8, class=3) düşük etkiye sahiptir; bu da GCN'in hibrit yapıda LightGBM'in sınıf olasılıklarını destekleyici yardımcı sinyaller olarak değerlendirdiğini göstermektedir.

Bu açıklanabilirlik çıktıları, siber güvenlik analistleri açısından “proto”, “ct_state_ttl”, “stcpb” gibi özniteliklerin baskınlığı, bu saldırı tipinin (Worms) ağ seviyesinde belirli protokol davranışları sergilediğini göstererek saldırı tipine özgü imzaların yazılabilmesine

olarak sağlamaktadır. Bununla birlikte, benzer saldırı etiketlerine sahip düğümlerin birbirine yakın konumlanması, bir saldırı kümesini ortaya koyarak analiz aşamasında bütüncül bir bakış açısının yakalanmasına olanak sağlamaktadır.

Şekil 7.13 ve Şekil 7.14’de açıklamalar ile belirtilen trafik akış örneği için SHAP analizinden elde edilen iki tamamlayıcı açıklamayı gösterilmektedir. SHAP Waterfall ve Force plot açıklamaları ile birlikte, açıklanabilirlik çıktısının örnek temelli sonuçları güçlendirilmiştir. Waterfall çıktısı ile her bir özneliğin model çıktısını ne kadar ve hangi yönde (arttırıcı/azaltıcı) etkilediği kademeli biçimde gösterilmektedir. Force çıktısı ile ise aynı açıklama yatay ekseninde güç vektörleri şeklinde sunulmaktadır; kırmızı alanlar sınıf tahminini yükselten, mavi alanlar baskılayan değişkenleri temsil etmektedir. Her iki görsel, modelin neden bu örneği saldırı (class 9) olarak işaretlediğini açıklamaktadır.

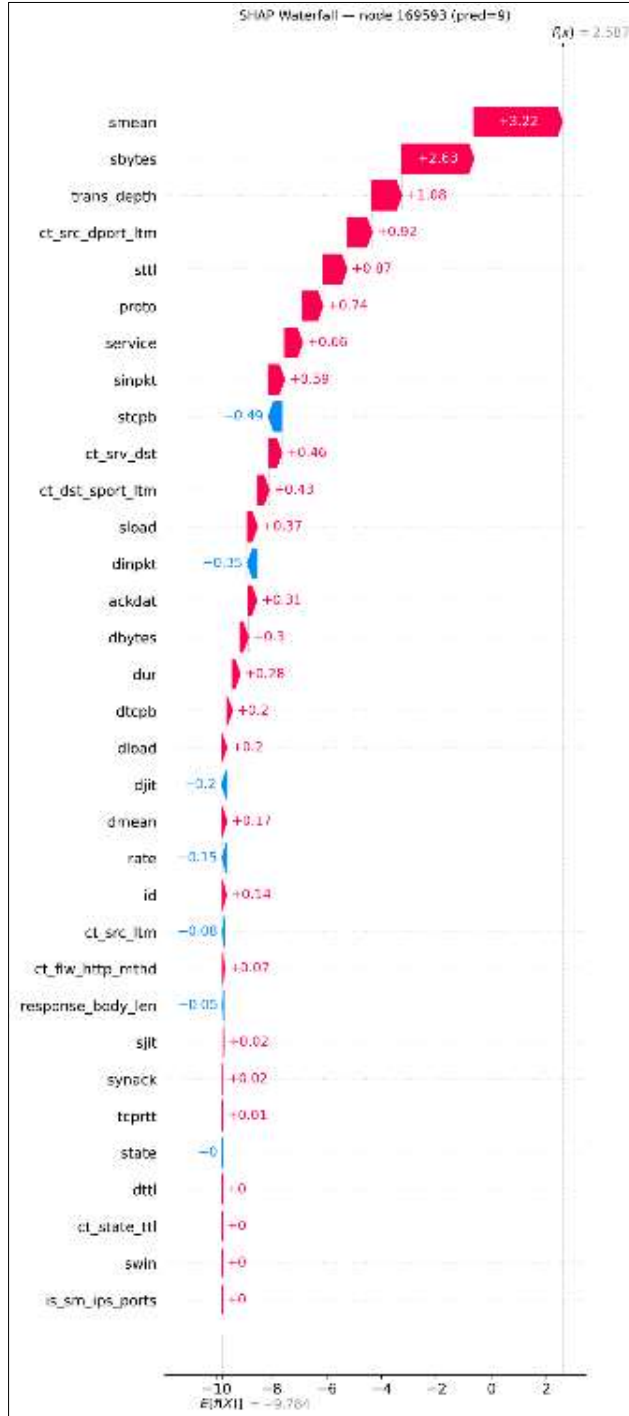
SHAP Waterfall çıktısı, taban değerden ($E[f(x)] = -9,784$) başlayarak tahmin değerine ($f(x) = 2,587$) ulaşmaya kadar her özneliğin katkısını göstermektedir. Açıklama üzerinden smean (+3,22), sbytes (+2,63), trans_depth (+1,08), ct_src_dport_ltm (+0,92), sttl (+0,87), proto (+0,74) özniteliklerinin en baskın pozitif katkı yapan öznitelikler olduğu görülebilmektedir. Bu öznitelikler, saldırı trafiğinin yüksek hacimli, tekrarlayan ve protokol düzeyinde anormal yapılar içerdiğini belirtmektedir. stcpb (-0,49), dinpkt (-0,35), djit (-0,20) gibi öznitelikler ise model tahminine negatif yönde etki etmiştir. Bunlar genellikle hedef tarafındaki pasif akış veya düşük jitter değerlerinden kaynaklanmaktadır ve model tarafından normal davranış göstergesi olarak algılanmıştır.

SHAP Force çıktısı, aynı katkı ilişkilerini doğrusal ekseninde itici-çekici kuvvet biçiminde görselleştirir. Kırmızı olarak belirtilen öznitelikler pozitif katkı sağlamaktadır. Waterfall çıktısı ile uyumlu olarak sinpkt, service, proto, sttl, ct_src_dport_ltm, trans_depth, sbytes, smean gibi öznitelikler model tahminini pozitif yönde iterek ve saldırı olasılığını yükseltmektedir. stcpb, dinpkt, djit gibi negatif katkıları gösteren mavi öznitelikler ise tahminine negatif etki yapmaktadır. Görselin en sağında $f(x) = 2,59$ noktasında, kırmızı itici kuvvetlerin baskınlığı açıkça görülmekte olup bu durum modelin bu örneği Worms saldırısı olarak sınıflandırma kararının güçlü bir şekilde desteklendiğini göstermektedir. Yani, $f(x)$ değerinin pozitif bölgede sonlanması ($\approx 2,6$), sistemin bu örneği yüksek güvenle saldırı olarak işaretlediğini ve SHAP açıklamasının bu kararı nicel olarak doğruladığını göstererek modele olan güveni artırmaktadır.

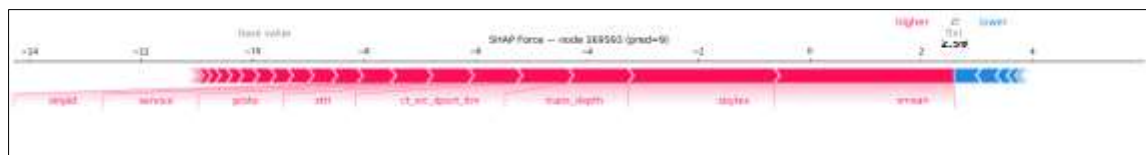
SHAP Waterfall ve Force çıktıları modelin saldırı tespiti mantığını analiste saydam biçimde sunmaktadır. Her iki çıktı birlikte değerlendirildiğinde yüksek sbytes, smean ve trans_depth değerleri gibi kısa sürede büyük miktarda veri transferi ve tekrarlayan bağlantıları ifade eden öznitelikler tipik olarak bulaşıcı zararlı yazılım trafiğinin (ör. Worms) belirterek analistlere davranışsal açıklama bilgisi sunmaktadır. Bununla birlikte, sttl, ct_src_dport_ltm ve proto değişkenleri, ağ protokol davranışının zaman içindeki sapmalarını yakalayarak saldırı modeline uygun anormallikleri tespit edilmesinde analistlere önemli ipuçları sağlamaktadır.

Bu tez çalışması kapsamında önerilen sistemin normal bir trafik örneğine ilişkin elde edilen açıklama çıktıları da Şekil 7.16, Şekil 7.17 ve Şekil 7.18’de sunulmuştur. Benzer yorum ve açıklamalar sunulan normal trafik çıktıları için de yapılabilir.

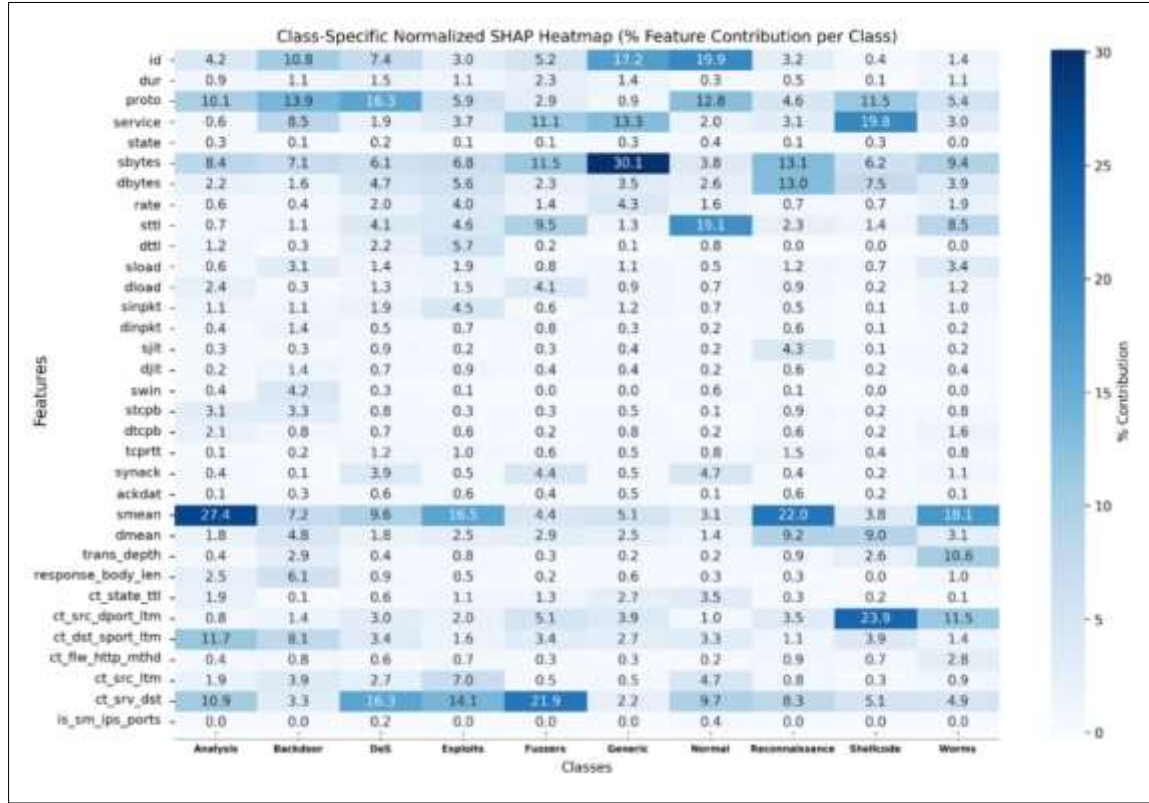
Önerilen model tarafından siber güvenlik analistlerine sunulan bir diğer açıklama Şekil 7.15’de gösterilen ısı haritasıdır. Şekil 7.15 önerilen modele ait sınıf-bazlı normalize edilmiş SHAP ısı haritasını göstermektedir. Isı haritası farklı saldırı türleri için özniteliklerin göreceli katkı oranlarını detaylı olarak belirterek modelin karar mekanizmasının hangi özniteliklere dayanarak belirli saldırı türlerini tespit ettiğini açıklanabilir biçimde görselleştirmektedir. Her hücre, belirli bir öznitelik ile bir sınıf arasındaki yüzdesel katkı oranını göstermektedir. Renk yoğunluğu, o özneliğin ilgili sınıf için ne ölçüde etkili olduğunu belirtmektedir. Örneğin, Analysis sınıfı üzerinde en etkili olan özneliğin smean, Generic sınıfı üzerinde en etkili olan özelliğin ise sbytes olduğu görülmektedir. Bununla birlikte, ısı haritası daha detaylı yorumlandığında örnek olarak Smean özneliği için, birden fazla sınıfta (özellikle Analysis, Exploits, Reconnaissance ve Worms) yüksek katkı oranları göstermektedir. Bu durum, bu özneliğin ağ genelinde belirleyici bir parametre olduğunu göstermektedir. STS perspektifinde değerlendirildiğinde bu tür bir öznitelik genellikle “flow-temelli korelasyon” veya “zaman bazlı yoğunluk (rate)” benzeri parametrelerle ilişkilidir ve DoS, Reconnaissance gibi yüksek hacimli saldırıların tespitinde belirleyici olabilir. Belirtilen ısı haritası sayesinde her bir saldırı türü için en belirleyici trafik özellikleri görsel olarak tespit edilmekte ve modelin hangi özelliklere duyarlı olduğu ortaya çıkarılmaktadır.



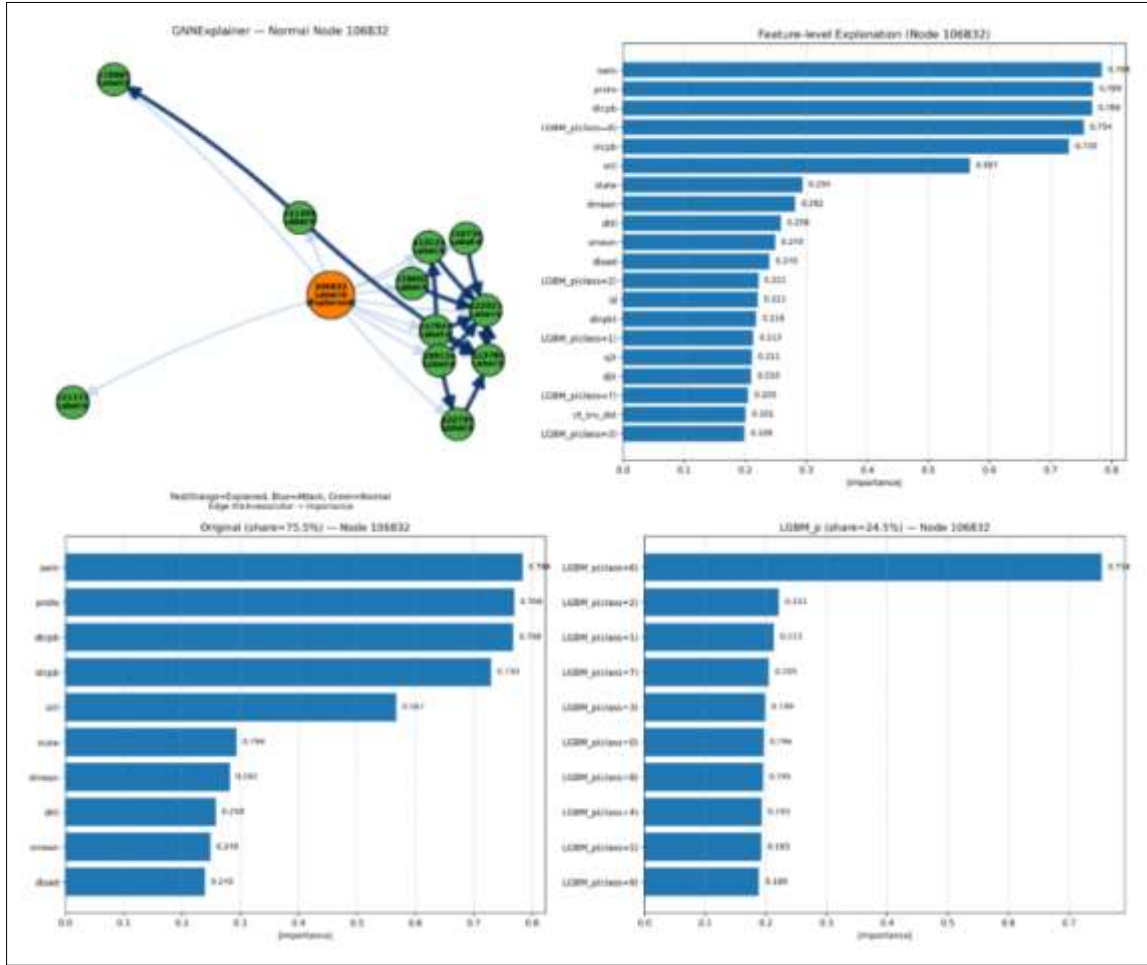
Şekil 7.13. UNSW-NB veri kümesi örnek bir saldırı trafiği için SHAP waterfall açıklama çıktısı



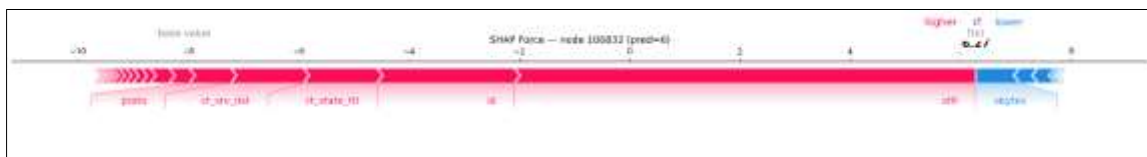
Şekil 7.14. UNSW-NB veri kümesi örnek bir saldırı trafiği için SHAP force açıklama çıktısı



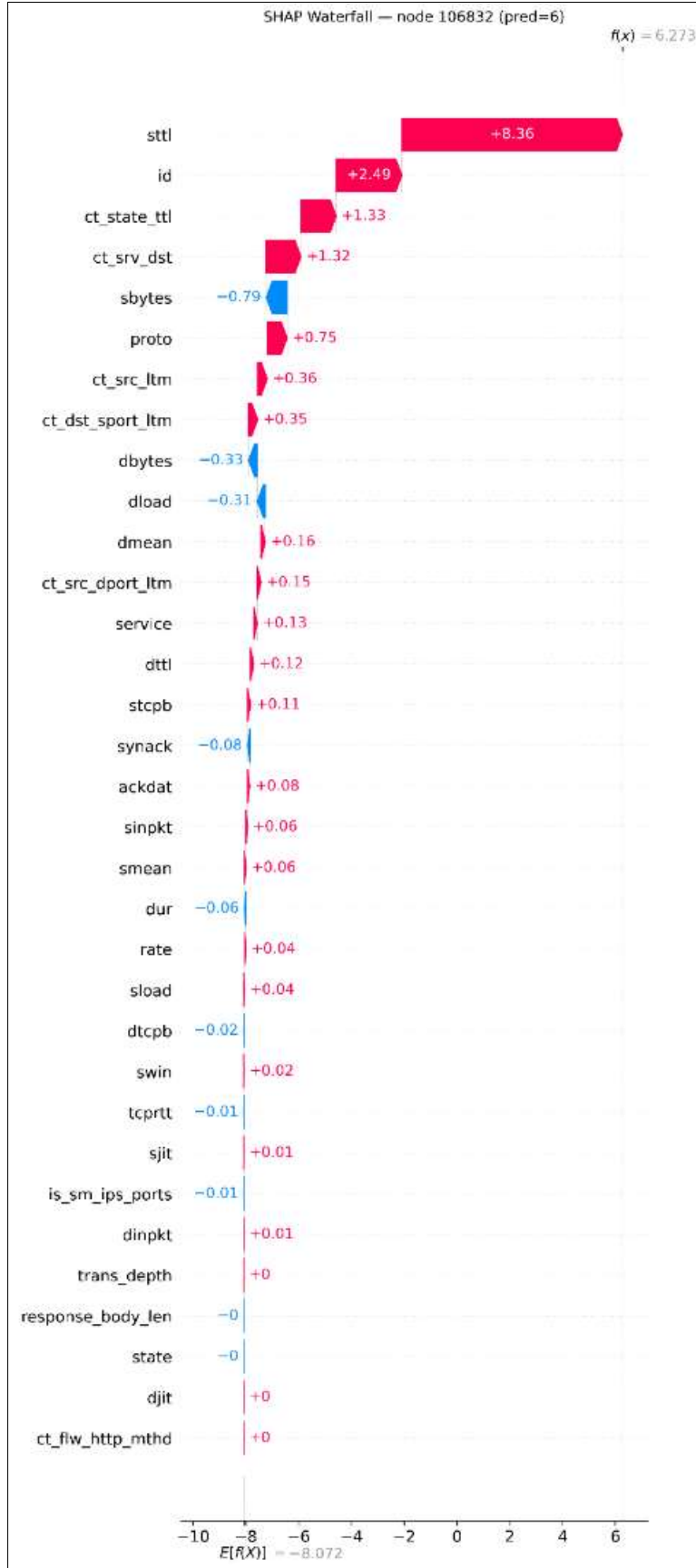
Şekil 7.15. UNSW-NB15 çok sınıflı sınıflandırmada özneliklerin sınıflara etkisinin gösterimi



Şekil 7.16. UNSW-NB veri kümesi örnek bir normal trafik için GNNExplainer açıklama çıktısı



Şekil 7.17. UNSW-NB veri kümesi örnek bir normal trafik için SHAP force açıklama çıktısı



Şekil 7.18. UNSW-NB veri kümesi örnek bir normal trafik için SHAP waterfall açıklama çıktısı

katkı oranını ifade etmektedir. Kalın ve koyu mavi kenarlar, modelin bu bağlantılardan yüksek oranda bilgi aktardığını ifade etmektedir.

İncelenen örnek ağ trafiği DoS GoldenEye saldırı türü olarak tespit edilmiş bir saldırı örneğini temsil etmektedir. Şekil 6.9'da gösterilen GNNExplainer açıklama çıktısı incelendiğinde, belirtilen örneğin diğer DoS GoldenEye olarak sınıflandırılmış düğümlerle (örneğin Node 46071, 44694, 47772, 49439) yüksek ağırlıklı kenarlarla bağlandığı görülmektedir.

Bu durum, Graph Neural Network'ün saldırı trafiğini benzer yapısal davranışlara göre kümelendirdiğini göstermektedir. DoS GoldenEye saldırıları, HTTP tabanlı flood trafiği üretmek için çok sayıda istek göndermektedir. GNNExplainer çıktısında kalın kenarlarla birbirine bağlanan düğümler, aynı anda yüksek istek sıklığı (request rate), yüksek paket uzunluğu varyansı (Packet Length Variance) ve benzer zaman aralıkları (Flow IAT) sergileyen trafiği temsil etmektedir. Bu altgraf, ağ topolojisinde DoS davranış kümelenmesini açıkça yansıtmaktadır.

Sağ üst taraftaki çubuk grafik, bu trafik örneği için özniteliklerin katkı değerlerini göstermektedir. Idle Max, Packet Length Std, LGBM_p (class=3), PSH Flag Count, Packet Length Variance, Flow IAT Std, Avg Bwd Segment Size, Bwd Packet Length Std, Init_Win_bytes_forward ve Bwd IAT Total öznitelikleri modelin kararını en çok etkileyen öznitelikler olmuştur. Bu sonuçlara göre önerilen model, DoS GoldenEye saldırılarını tanımlarken ağırlıklı olarak zamanlama (IAT), paket varyansı, ve TCP bayrak davranışlarını dikkate almaktadır. Bu sonuçlar GoldenEye saldırısının doğasıyla tam uyumludur. Çünkü HTTP flood trafiği genellikle düzensiz, kısa aralıklı ve yüksek varyanslı paket akışları üretmektedir.

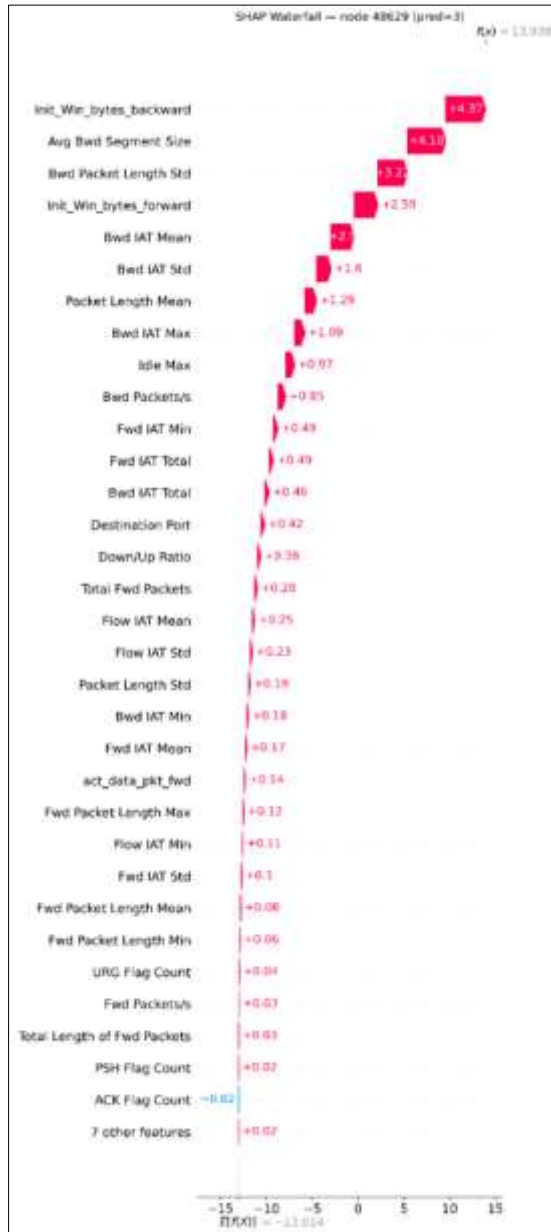
Açıklama çıktısının alt kısmında yer alan çubuk grafiklerde ise belirtilen örneğin sınıflandırılmasındaki orijinal öznitelik katkısı %72,8 ile LightGBM tahmin çıktısının katkısı ise %27,2 olarak belirtilmektedir. Orijinal öznitelik katkıları modelin fiziksel trafik özellikleri üzerinden yaptığı çıkarımı temsil etmektedir. LightGBM tahmin çıktılarının katkısı değerlendirildiğinde LGBM_p (class=3) (DoS GoldenEye olasılığı) en baskın değere (0.80) sahiptir. Bu sonuç, LightGBM'in örneği yüksek güvenle DoS GoldenEye olarak işaretlediğini ve GCN'in bu kararı topolojik bağlamda doğruladığını göstermektedir.

DoS GoldenEye saldırısı, yüksek HTTP istek oranı ve özellikle web sunucuları olmak üzere ağ servislerini aşırı yük altına sokmaya çalışmaktadır. Bu açıklanabilirlik çıktıları, siber güvenlik analistlerine bu saldırının model tarafından doğru şekilde yüksek varyans, kısa IAT ve PSH flag artışı kombinasyonu üzerinden tanındığını göstermektedir. Flow IAT ve Idle Max gibi zamansal özniteliklerin model kararında etkili olması nedeniyle Flood tipi DoS ataklarda bağlantı aralıklarının düzensizleşmesi doğrudan fark edilebilmektedir. Ayrıca, Yüksek PSH oranı, TCP üzerinden ani veri gönderimi anlamına gelmekte olup DoS GoldenEye saldırılarının tipik bir göstergesidir. Benzer şekilde HTTP GET/POST isteklerinin yoğunluğu Packet Length Std ve Packet Length Variance özniteliklerinin değerlerini artırması ve bu özniteliklerin modelin sınıflandırılmasında etkili olması modele olan güveni artırmaktadır. GNNExplainer çıktıları değerlendirildiğinde, önerilen model saldırıyı hem istatistiksel hem de graf-topolojik düzeyde doğru şekilde tespit etmiştir.

Şekil 7.20’de gösterilen SHAP Waterfall çıktısı, önerilen modelin bu örneği DoS GoldenEye olarak sınıflandırmasına hangi özniteliklerin ne ölçüde katkı yaptığını görsel olarak ortaya koymaktadır. Çıktı incelendiğinde $E[f(x)] = -13,014$ değeri modelin tüm veri üzerindeki ortalama tahmin değerini ifade etmektedir. $f(x) = 13,939$ değeri ise bu örnek için modelin çıktısını belirtmekle birlikte güçlü biçimde saldırı sınıfının desteklendiği açıkça görülmektedir. Sonuçta pozitif SHAP katkılarının baskın olması, modelin yüksek güvenle DoS GoldenEye saldırısını tespit ettiğini göstermektedir. Init_Win_bytes_backward (+4,37), Avg Bwd Segment Size (+4,18), Bwd Packet Length Std (+3,22), Init_Win_bytes_forward (+2,59), Bwd IAT Mean (+2,50), ve Bwd IAT Std (+1,60) gibi özniteliklerin sınıflandırmada baskın olması DoS GoldenEye saldırısının karakteristik davranışlarını da doğrudan yansıtmaktadır. Açıklama çıktısı modelin bu saldırıyı DoS GoldenEye’in davranışsal doğasıyla uyumlu olacak şekilde yüksek frekanslı ama dengesiz HTTP trafiği üzerinden algıladığını göstermektedir. Bununla birlikte, açıklamadan görüldüğü üzere az sayıda mavi blok, modelin saldırı kararını kısmen sınırlayan özellikleri göstermekle birlikte bu negatif katkılar küçük olduğundan dolayı modelin kararında önemsiz düzeyde etki oluşturmuştur.

Şekil 7.21’de gösterilen SHAP Force çıktısı, $f(x)=13,94$ değeri nihai model çıktısını ifade etmekle birlikte; bu değer yüksek pozitif olması belirtilen örneğin güçlü biçimde DoS GoldenEye olarak sınıflandırıldığını göstermektedir. Base value ≈ -13 değeri modelin tüm veri kümesi üzerindeki ortalama tahminini ifade etmektedir. Bu iki değer arasındaki fark,

pozitif SHAP katkılarını gösteren kırmızı kuvvetlerin mavi kuvvetlere baskın geldiğini, yani modelin saldırı kararını kuvvetli biçimde destekleyen bir örnekle karşı karşıya olduğumuzu göstermektedir. SHAP Force açıklaması ile IAT Std ve Segment Size gibi Flood saldırılarına özgü bilinen metriklerin pozitif yönde baskın olmasından dolayı modelin ağ güvenliği bilgisini öğrenmiş olduğu gösterilmiştir. Analistler bu çıktıyı kullanarak DoS saldırılarının tespitine ilişkin STS kurallarını şekillendirebilir.



Şekil 7.20. CIC-IDS2017 veri kümesi örnek bir saldırı trafiği için SHAP waterfall açıklama çıktısı



Şekil 7.21. CIC-IDS2017 veri kümesi örnek bir saldırı trafiği için SHAP force açıklama çıktısı

Farklı saldırı türleri için özniteliklerin katkı oranlarını gösteren Şekil 7.22'deki ısı haritası incelendiğinde, Patator saldırılarında destination port özniteliğinin önemli ölçüde etkili olduğu görülmektedir. Patator, çok amaçlı bir brute-force ve fuzzing aracıdır. Python ile yazılmıştır ve ftp, ssh, http_fuzz, smtp gibi farklı protokoller/servisler için modüller içermektedir. FTP-Patator saldırı hedef sistemdeki ftp servisini hedef alırken SSH-Patator ise SSH servisini hedef almaktadır. Bu bilgiler doğrultusunda önerilen sistemin patator saldırılarını tespit ederken hedef sistemin servis portlarına ağırlık vermesi sistemin güvenilirliğini ortaya koymaktadır. Benzer şekilde Heartbleed saldırı tespitinde Avg Bwd Segment Size özniteliğinin Bot saldırı tespitinde ise Destination Port özniteliğinin etkin olması sistemin güvenilirliği ve analistler tarafından daha anlaşılır bir STS altyapısı sunması açısından önemlidir.

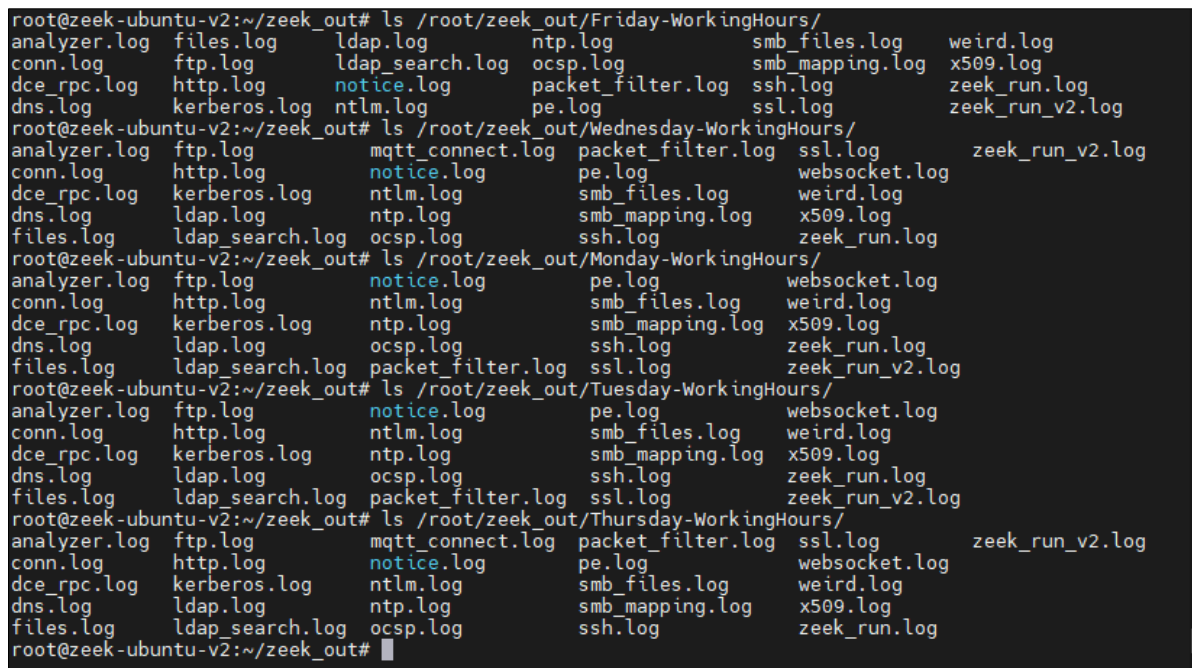
7.5. Zeek Performans Sonuçları

Bu tez çalışması kapsamında önerilen sistemin performansı Zeek açık kaynak saldırı tespit sistemi ile karşılaştırılmıştır. Önerilen bütünleşik sistemin, kural ve politika tabanlı STS'lerle (örneğin Zeek) karşılaştırmalı başarımı, Araştırma Sorusu-5 kapsamında değerlendirilmiştir. Karşılaştırma işlemleri CIC-IDS2017 ve UNSW-NB15 veri kümelerine ait pcap dosyaları kullanılarak offline paket işleme neticesinde gerçekleştirilmiştir. Farklı özellik dağılımlarına ve saldırı taksonomilerine sahip CIC-IDS2017 ve UNSW-NB15 veri kümeleri üzerindeki sonuçlar, Araştırma Sorusu-5 çerçevesinde sistemin genellenebilirliğini analiz etmek amacıyla incelenmiştir.

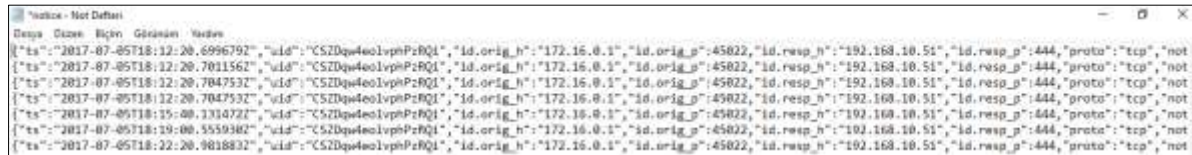
Zeek kurulumları, VMware Workstation Pro platformunda 2 GB RAM, 2 CPU ve 100 GB Disk kapasitesine sahip Ubuntu sanal sunucu üzerinde yapılmıştır. Kurulumlarda Ubuntu 24.04 ve zeek 8.0.4 versiyonları kullanılmıştır. CIC-IDS2017 veri kümesi üzerinde Zeek çalıştırıldıktan sonra oluşan log dosyaları Resim 7.2'de gösterilmiştir. Zeek pcap dosyalarından elde ettiği trafik verilerini conn.log dosyasına, saldırı olarak işaretlediklerini



Resim 7.1. CIC-IDS2017 veri kümesi örnek conn.log dosyası



Resim 7.2. CIC-IDS2017 veri kümesi için Zeek tarafından üretilen log dosyaları



Resim 7.3. CIC-IDS2017 veri kümesi örnek notice.log dosyası

Literatürde gerçek ortam STS testlerinde karşılaştırma parametresi olarak kullanılan tespit oranı (detection rate) şu şekilde hesaplanmaktadır:

$$DR = \frac{\text{Tespit edilen saldırı örneklerinin sayısı}}{\text{Toplam saldırı örneklerinin sayısı}} = \frac{TP}{TP + FN} \quad (7.1)$$

Çizelge 7.9. CIC-IDS2017 veri kümesi için Zeek tarafından üretilen günlere göre trafik ve saldırı sayıları

Günler	Trafik miktarı	Saldırı miktarı
Pazartesi	375432	63
Salı	323342	71
Çarşamba	509362	81
Perşembe	363788	70
Cuma	547353	65

Çizelge 7.10. UNSW-NB15 veri kümesi için Zeek tarafından üretilen trafik ve saldırı sayıları

Günler	Trafik miktarı	Saldırı miktarı
17.02.2015, 22.01.2015	2119795	9235

Çizelge 7.11. CIC-IDS2017 ve UNSW-NB15 veri kümeleri için Zeek performans sonuçları

Veri Kümesi	Doğruluk (%)	Tespit Oranı (%)
CIC-IDS2017	73,81	4,76e-4
UNSW-NB15	17,30	0,0287

CIC-IDS2017 için gün bazında üretilen saldırı miktarları oldukça düşük düzeydedir (ör. Pazartesi 63, Salı 71, Çarşamba 81 vb.). UNSW-NB15 tarafında ise tüm pcap'lar üzerinde Zeek'in notice üretimi 9235 olarak raporlanmıştır. Bununla birlikte, performans göstergeleri (doğruluk ve tespit oranı) incelendiğinde Zeek'in veri kümelerine göre oldukça dengesiz sonuçlar verdiği görülmektedir: CIC-IDS2017 için doğruluk %73,81 olmasına rağmen tespit oranı $4,76e-4$ gibi çok düşük bir seviyededir; UNSW-NB15 için doğruluk %17,30, tespit oranı ise %0,0287 olarak raporlanmıştır. Sonuçlar değerlendirildiğinde Zeek açık kaynak STS sistemi saldırıların tespitine yönelik gerekli paket ve politikalar yüklenmiş olsa bile Zeek'in birçok saldırıyı tespit edemeyip yüksek performans gösterememiştir. Bu durum Zeek'in esasen imza tabanlı olmasından kaynaklanmaktadır. Bununla birlikte, Zeek üzerinde ağ altyapısı ve güvenlik gereksinimlerine göre özel politikalar da oluşturulabilmektedir. Her iki veri kümesinin Zeek performanslarının yakın olmaması altyapıya özgü konfigürasyon gereksinimini açıkça göstermektedir. UNSW-NB15 veri kümesi üzerindeki kötü performans temel seviyede yapılmış bir Zeek konfigürasyonunun UNSW-NB15'in etiketlediği çok geniş

saldırı yelpazesinin sadece çok küçük bir kısmını kapsadığını göstermektedir. Elde edilen sonuçlardaki bir diğer önemli nokta ise UNSW-NB15 veri kümesi ile elde edilen tespit oranının CIC-IDS2017 veri kümesine göre oldukça yüksek çıkmasıdır. Bu durum, Zeek'in akış paketleri içerisinde notice.log dosyasına saldırı olarak veri üretebildiğini ancak üretilen saldırı trafiklerinin etiketli veri kümesi içindeki akış bilgileri ile eşlemediğini göstermektedir. Zeek çıktılarındaki akış anahtarı genellikle 5-tuple + zaman mantığına dayanır (srcIP/srcPort/dstIP/dstPort/proto ve ts/uid). Veri kümesi etiketleri ise çoğu zaman başka bir akış tanımı, farklı zaman penceresi, NAT/anonimleştirme veya farklı "flow timeout" varsayımlarıyla üretilmiştir. Bu nedenle Zeek'in notice ürettiği olaylar "doğru saldırı" olsa bile, veri kümesindeki etiketli akışla birebir eşleşmeyebilir. Sonuç olarak, önerilen sistem mimarisi ile Zeek karşılaştırıldığında önerilen sistem özellikle öğrenme yeteneği sayesinde daha yüksek performans elde etmiştir. Bunun yanı sıra, iki veri kümesindeki farklı davranışlarından dolayı Zeek'in genelleştirme yeteneğinin oldukça zayıf olduğu görülmektedir.

Zeek'in daha adil ve daha güçlü bir performans sergileyebilmesi için daha kapsamlı Zeek script'leri etkinleştirilmeli, datasetin saldırı türleriyle eşleştirilecek özel politikalar(policy) yazılmalıdır. Bununla birlikte, Zeek'in pcap üzerinde çalıştırılmasında; akış birleştirme, timeout ve yeniden iletim gibi parametreler datasetin üretim mantığına yaklaştırılmalıdır. Aksi halde aynı saldırı aktivitesi farklı akışlara bölünür veya farklı zamanlarda farklı uid'ler üretilir. Ayrıca, kullanılan kaynaklar pcap'ların yoğunluğuna göre yetersiz kalabilir ve analiz gecikmesi/atlama yaratabilir. Daha güçlü CPU/RAM veya Zeek cluster yaklaşımı offline analizde log bütünlüğünü artırabilir.

Önerilen sistemin güçlü yanı öğrenme yeteneği ve AYZ+BDM ile açıklanabilirlik katmanıdır. Bu avantajı gerçek ortama taşıyabilmek için bir kurum ağına benzer topoloji oluşturularak üretilen trafikler üzerinde kontrollü bir gerçek ortam (testbed) doğrulaması yapılabilir. Bununla birlikte, gerekli altyapı gereksinimleri karşılanarak SPAN/TAP ile trafik kopyalanarak veya Zeek sensörleri ağına kritik noktasına konumlandırılarak trafikler Zeek üzerinde gerçek zamanlı olarak işlenebilir ve önerilen model altyapı üzerinde çalıştırılabilir. Bu tür gerçek ortam senaryolarının uygulanabilmesi için ağ trafik miktarı doğrultusunda altyapı gereksinimlerinin doğru analiz edilerek kullanılacak donanım ve yazılımların bunlara göre belirlenmesi önem arz etmektedir.

7.6. BDM Tabanlı Açıklamalar

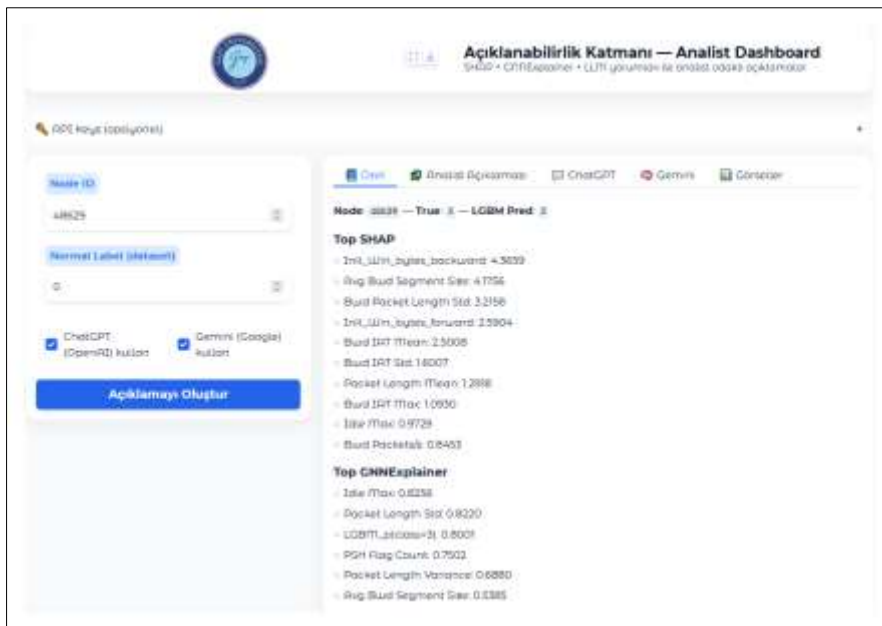
Bu tez çalışması kapsamında, önerilen sistemde elde edilen açıklama sonuçları ChatGPT ve Gemini sohbet ara yüzlerine API ile bağlanarak yorumlatılmıştır. Bu aşamada elde edilen sonuçların analistlere daha anlaşılır ve görsel olarak sunulması amaçlanmıştır. Elde edilen yorumlar belirtilen sohbet robotlarına *"Sen bir SOC analistisin. SHAP ve GNNExplainer çıktılarını kullanarak bu düğümün neden bir saldırı olarak sınıflandırıldığını açıkla"* promptu verilerek elde edilmiştir. Verilen prompt içerisinde açıklanabilirlik çıktılarından elde edilen veriler bulunmaktadır. Verilen prompt bir örnek olup SOC yapısında ilk seviyede yer alan analistlere alarmların nedenlerini sağlamak amacıyla oluşturulmuştur. Verilen prompt örneklerinin derinliği ve kapsamı genişletilerek daha üst seviyedeki analist ve yöneticilere yönelik de yorumlamalar üretilebilmektedir.

Bu çalışmada BDM, STS çıktılarının kendisini değil, AYZ yöntemleri aracılığıyla elde edilen açıklama çıktılarını yorumlamaktadır. Bu kapsamda, SHAP ile elde edilen özellik düzeyindeki katkılar ve GNNExplainer ile elde edilen graf ve alt-graf düzeyindeki ilişkisel açıklamalar, BDM'ye girdi olarak sunulmaktadır. Bu yaklaşımın en önemli avantajı, üretilen açıklamaların doğrudan model karar mekanizmasına dayalı olmasıdır. SHAP değerleri, hangi özelliklerin kararı hangi yönde ve ne ölçüde etkilediğini nicel olarak ortaya koyarken; GNNExplainer, grafik tabanlı modellerde kararın hangi düğümler ve kenarlar üzerinden şekillendiğini açıkça göstermektedir. BDM bu noktada bir karar verici değil, kanıta dayalı açıklamaları insan tarafından anlaşılabilir dile dönüştüren bir arayüz rolü üstlenmektedir. Bu yapı, açıklamaların da denetlenebilirliğini artırmaktadır. Üretilen her doğal dil açıklaması, SHAP değerleri veya alt-graf yapıları gibi somut AYZ çıktılarıyla ilişkilendirilebilmekte ve gerektiğinde geriye dönük olarak doğrulanabilmektedir. Böylece BDM kaynaklı olası tutarsızlıklar veya hatalı genellemeler, AYZ çıktıları üzerinden kontrol edilebilir hâle gelmektedir. Öte yandan, bu yaklaşımın başarısı doğrudan kullanılan AYZ yöntemlerinin kalitesine bağlıdır. AYZ çıktılarının yanıltıcı veya kararsız olduğu durumlarda, AYZ tarafından üretilen açıklamalar da hatalı olabilmektedir. Ancak bu durum, açıklamanın hangi bileşenlerden üretildiğinin açıkça izlenebilir olması sayesinde yönetilebilir bir risk oluşturmaktadır.

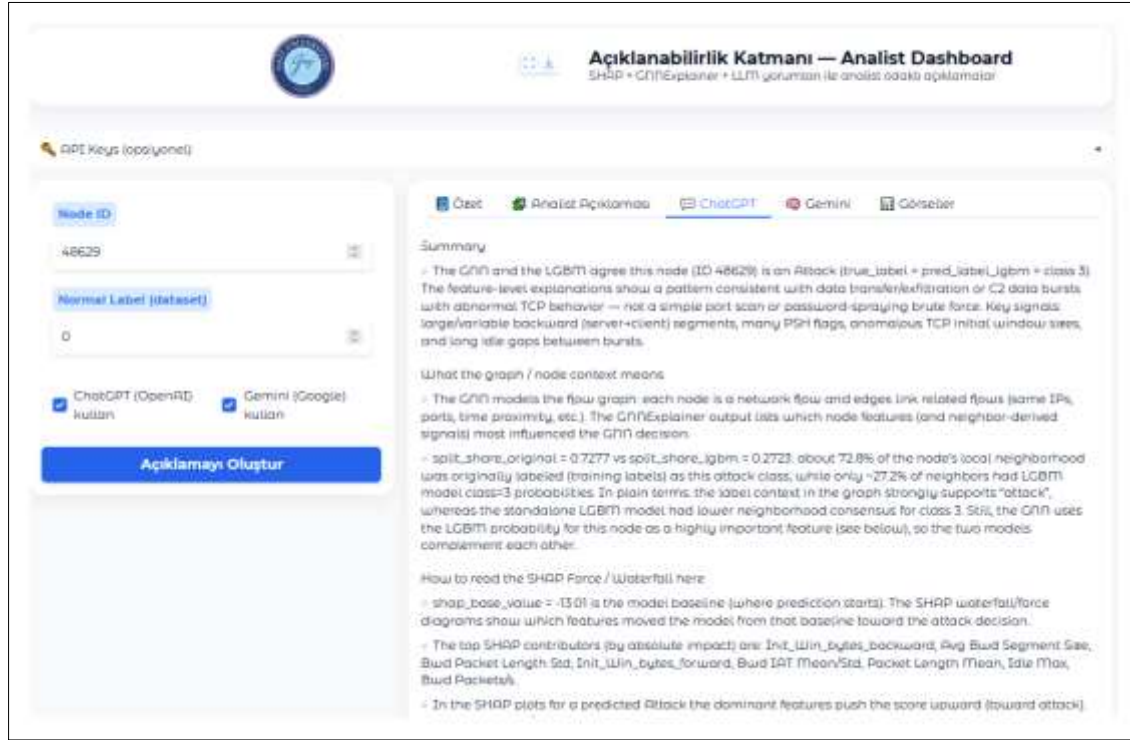
Bazı çalışmalarda[577,578] BDM'ler, kural tabanlı veya öğrenme tabanlı STS'lerden elde edilen ham alarm çıktıları, log kayıtları veya uyarı mesajlarını doğrudan girdi olarak almakta

ve bu çıktıları doğal dilde özetlemekte veya yorumlamaktadır. Bu yaklaşımda BDM, genellikle alarmın türünü, olası nedenlerini ve güvenlik analistinin alabileceği aksiyonları açıklamaya yönelik bir rol üstlenmektedir. Özellikle Seviye-1 analistlerin iş yükünü azaltmaya yönelik bu yaklaşım, pratik açıdan faydalar sunabilmekle birlikte alarmın gerçek nedenlerini değil, alarm metninden çıkarılabilecek olası senaryoları anlatmakta bu da özellikle yanlış veya aşırı genelleştirilmiş açıklamalar üretme riskini beraberinde getirmektedir. Halüsilasyon gibi ciddi problemler içeren BD modelleri bu tür kullanımda hatalı sonuçlar üreterek ciddi sonuçlara sebep olabilecektir. AYZ çıktılarının LLM ile yorumlanması bilimsel açıklanabilirlik, güven ve denetlenebilirlik açısından daha güçlü bir çerçeve sunmaktadır. STS sonuçlarının doğrudan BDM'lere iletilmesi özellikle günlük SOC operasyonlarında alarm özetleme ve hızlı aksiyon alma açısından avantaj sağlarken; bu çalışma kapsamında önerilen yaklaşım, STS kararlarının nedenlerini derinlemesine analiz etmeyi, model davranışlarını anlamayı ve güvenlik analistlerine gerekçeli karar desteği sunmayı mümkün kılmaktadır.

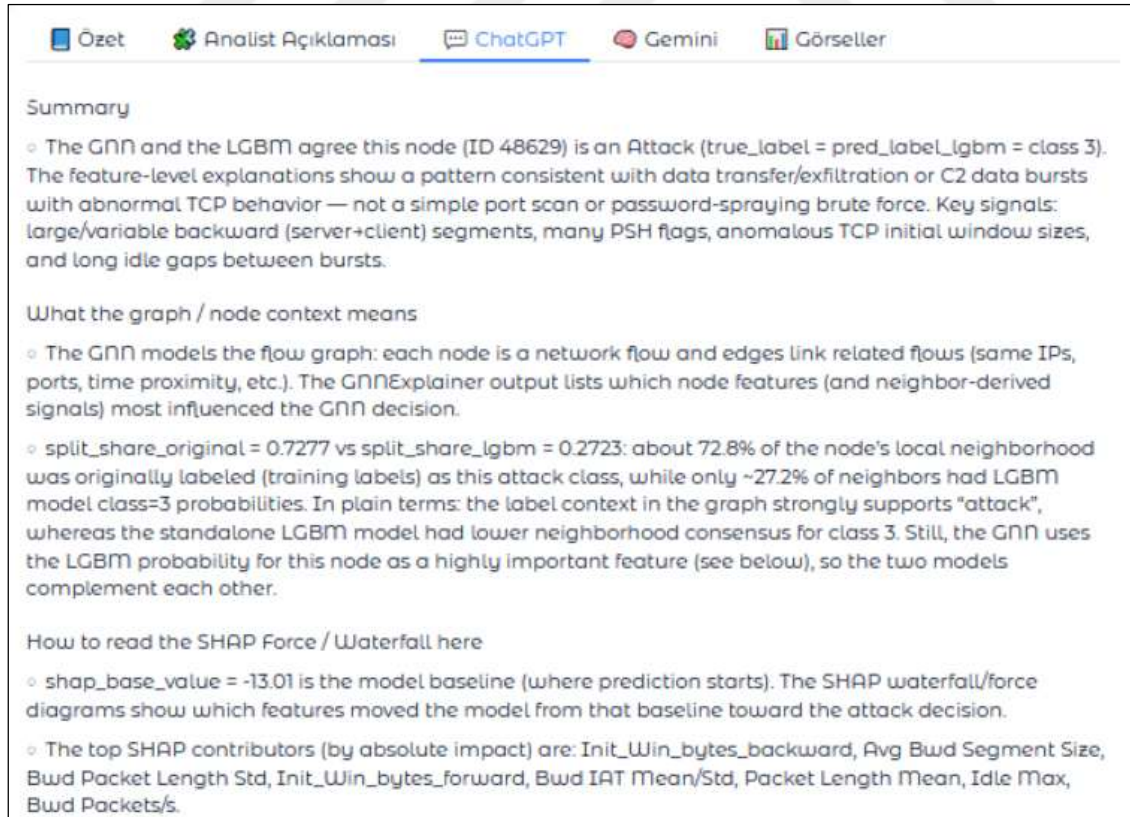
Çalışma kapsamında, açıklamalar her iki BDM bağlantısı sonucunda hem rapor hem de bir web arayüzü üzerinden analistlere sunulurken önerilen sistemin daha anlaşılır hale gelmesi sağlanmıştır. Bu çalışmalar kapsamında oluşturulan web arayüzüne ilişkin örnek görseller Resim 7.4, Resim 7.5 ve Resim 7.6'da gösterilmektedir. Bununla birlikte, elde edilen açıklama çıktıları rapor formatında da alınabilmektedir.



Resim 7.4. Açıklanabilirlik katmanı web arayüzü özet bölümü



Resim 7.5. Açıklanabilirlik katmanı web arayüzü ChatGPT bölümü



Resim 7.6. Resim 7.5'in genişletilmiş versiyonu

8. SONUÇ VE DEĞERLENDİRMELER

Bu tez çalışmasında; yapay zekâ destekli STS'lerin karar mekanizmalarının anlaşılmasının ve güvenilirliğinin önünde önemli bir engel olan kara kutu probleminin çözümüne yönelik AYZ yaklaşımları ele alınmış, alandaki boşluklar belirlenerek yeni yaklaşım ve algoritmalar önerilmiş, belirlenen araştırma soruları kapsamında analizler yapılmış, yapay zekâ temelli STS'lerin modellenmesi ve kararlarının açıklanmasına yönelik özgün bir STS geliştirilmiş, modelin açıklanabilirlik katkıları ve doğruluğu test edilmiş ve elde edilen sonuçlar değerlendirilmiştir.

Çalışma kapsamında belirlenen araştırma soruları, gerçekleştirilen kapsamlı literatür araştırmaları ve deneysel çalışmalarla sistematik biçimde değerlendirilmiştir;

- Bu çalışma kapsamında gerçekleştirilen kapsamlı literatür taraması, STS alanında yapay zekâ kullanımının son yıllarda belirgin biçimde arttığını ve çalışmaların ağırlıklı olarak (i) ağaç/boosting tabanlı yöntemler, (ii) derin öğrenme tabanlı yaklaşımlar ve daha sınırlı biçimde (iii) graf tabanlı öğrenme etrafında yoğunlaştığını göstermektedir. Literatürde özellikle tablosal ağ trafiği verilerinde boosting tabanlı yöntemlerin yaygınlaştığı, derin öğrenme yöntemlerinin ise çoğunlukla yüksek hesaplama maliyetine rağmen tercih edildiği; graf tabanlı yaklaşımların ise veri temsili ve graf kurulumuna yönelik zorluklar nedeniyle daha az sayıda çalışmada kullanıldığı gözlemlenmiştir. Bununla birlikte literatürde tekrar eden yöntemsel zayıflıkların; tek veri kümesine bağımlılık, sınıf dengesizliği probleminin ihmal edilmesi, tek/az sayıda metrikle değerlendirme yapılması ve tekrarlanabilirliği sınırlı deneysel kurulumlar olduğu tespit edilmiştir. Hibrit mimariler belirli bir artış eğilimi gösterse de, bu hibrit yaklaşımların önemli bir kısmı sistematik bir tasarım çerçevesi sunmamakta; bileşenlerin nasıl bütünleştirildiği ve hangi koşullarda katkı sağladığı yeterince şeffaf biçimde ortaya konmamaktadır. Bu tezde gerçekleştirilen deneysel karşılaştırmalar, tablosal STS verileri üzerinde boosting tabanlı yaklaşımın (LightGBM) genel olarak daha kararlı ve daha genellenebilir performans ürettiğini; GCN'in tek başına kullanılmasından ziyade ilişkisel bağlamı modele dâhil ederek hibrit mimariye katkı verdiğini göstermiştir. Sonuç olarak, tablosal veri yapısı baskın olan STS problemlerinde temel sınıflandırıcı olarak boosting tabanlı yöntemlerin daha uygun olduğu; buna ek olarak graf tabanlı öğrenmenin, özellikle hibrit tasarım içinde, bazı

saldırı örüntülerinin yakalanmasına katkı sağlayabildiği değerlendirilmiştir (Çizelge 2.2, Çizelge 2.3, Çizelge 7.5, Çizelge 7.6, Çizelge 7.7 ve Çizelge 7.8).

- Literatür bulguları, STS alanında AYZ kullanımının çoğunlukla öznitelik-seviyesi açıklamalarla sınırlı kaldığını göstermektedir. Bu bağlamda SHAP ve LIME gibi yöntemler yaygın biçimde(325 adet) kullanılsa da, açıklamalar çoğu çalışmada yalnızca “hangi özellik önemli?” sorusuna yanıt vermekte; modelin karar mekanizmasını yapısal/ilişkisel düzeyde ele alan açıklamalar ise sınırlı kalmaktadır. Graf tabanlı açıklanabilirlik (örneğin GNNExplainer) ve kararların ilişkisel doğasını ortaya koyan yaklaşımların literatürde az sayıda(8 adet) çalışmada yer alması, çok boyutlu AYZ (öznitelik + graf seviyesi) yaklaşımının açık bir araştırma boşluğu olduğunu ortaya koymaktadır (Çizelge 2.2 ve Çizelge 2.3). Bu tezde SHAP ve GNNExplainer birlikte kullanılarak, STS kararlarının hem öznitelik-seviyesinde nicel gerekçeleri hem de graf-seviyesinde ilişkisel gerekçeleri ortaya konmuş; böylece literatürde sınırlı biçimde ele alınan çok boyutlu AYZ ihtiyacına doğrudan yanıt verilmiştir.
- Literatürde BDM entegrasyonunun STS süreçlerine genellikle alarm/log çıktılarının doğrudan BDM ile yorumlanması ve özetlenmesi ve SOC iş akışlarında alarm filtreleme/false positive ayıklama gibi ajan tabanlı yaklaşımlar biçiminde yansıdığı görülmektedir. Bu tez kapsamında BDM’ler, doğrudan STS kararını “üreten” bir bileşen olarak değil, SHAP ve GNNExplainer çıktıları gibi kanıta dayalı AYZ sonuçlarını insan-odaklı biçime dönüştüren yorumlayıcı bir katman olarak konumlandırılmıştır (Resim 7.4, Resim 7.5 ve Resim 7.6). Bu sayede teknik açıklamalar, güvenlik analistleri açısından daha anlaşılır, bağlamsal ve operasyonel hale getirilmiştir. Ayrıca literatürde sıkça görülen “açıklama üretildi ama kullanıcı açısından değerlendirilemedi” boşluğu, en azından açıklamaların yapılandırılması, standardize edilmesi ve analist-dostu hale getirilmesi yönünde somut bir katkı ile ele alınmıştır. Bu yaklaşım, BDM’lerin açıklanabilirliği artırmak için daha güvenilir bir kullanım biçimini temsil etmektedir.
- Bu tez, hibrit sınıflandırma, graf tabanlı öğrenme, çok boyutlu AYZ ve BDM entegrasyonunun tek bir STS çerçevesinde bütünleştirilebileceğini ve bunun yeni/özgün bir çözüm ortaya koyabileceğini göstermiştir (Şekil 6.1). Önerilen mimaride, veri ön işleme ve sınıf dengeleme adımları sınıflandırma performansını ve özellikle az temsil edilen sınıfların tespit başarımını iyileştirirken; LightGBM tablosal özelliklerden güçlü bir temel karar mekanizması üretmiş, GCN ise örnekler arası ilişkisel yapıyı modele dahil ederek hibrit karar mekanizmasını zenginleştirmiştir. Açıklanabilirlik boyutunda SHAP ve GNNExplainer birlikte kullanılarak, tek boyutlu açıklamaların sınırlılıkları aşılmış ve

kararların hem öznitelik-seviyesi hem graf-seviyesi gerekçeleri ortaya konmuştur. BDM entegrasyonu ise bu teknik çıktıları analist-dostu bir dile çevirerek açıklanabilirliği “kullanılabilir” hale getirmiştir. Bu bütünleşik yapı, performans ve açıklanabilirliği birlikte optimize edebilen güvenilir bir STS tasarımının mümkün olduğunu desteklemektedir (Şekil 7.12-7.21, Çizelge 7.5, Çizelge 7.6, Çizelge 7.7 ve Çizelge 7.8).

- Önerilen bütünleşik sistem tasarımı, CIC-IDS2017 ve UNSW-NB15 üzerinde yüksek performans sunarak farklı özellik dağılımlarına ve saldırı taksonomilerine rağmen genellenebilir bir yapı sergilemiştir. Kural/politika tabanlı STS’ler (örneğin Zeek) ile karşılaştırmalı değerlendirmeler, Zeek’in başarısının büyük ölçüde aktif kural setine, ortama özgü politikalara ve offline pcap değerlendirmesinde etiket-eşleştirme zorluklarına bağlı olduğunu; buna karşın öğrenme tabanlı yaklaşımın daha geniş saldırı çeşitliliğinde daha güçlü bir genelleme kabiliyeti sunduğunu göstermiştir (Çizelge 7.5, Çizelge 7.6, Çizelge 7.7 ve Çizelge 7.8 ve Çizelge 7.11). Gerçek ortam simülasyonları bağlamında, önerilen sistemin başarımını doğrulamak için Zeek’in sensör olarak kullanıldığı; Zeek loglarının özellik çıkarımıyla modele aktarıldığı ve hibrit modelin gerçek zamanlı/yarı gerçek zamanlı değerlendirmeye tabi tutulduğu bir testbed yaklaşımı en uygun doğrulama yolu olarak değerlendirilmiştir. Böylece modelin yalnızca etiketli veri kümelerinde değil, gerçekçi akışlarda da ölçülebilir biçimde sınanması mümkün olacaktır.

Gerçekleştirilen çalışmada;

- Öncelikle AYZ ve STS çalışma alanları kapsamlı olarak ele alınmış, AYZ ve STS’ler hakkında ayrıntılı tanımlar ve terminoloji sunulmuş, fırsatlar ve zorluklar incelenmiş ve AYZ modelleri ve uygulamaları geliştirmek için yaklaşımlar, araçlar, veri kümeleri ve teknolojileri özetlenmiş, kavram karmaşaları ortadan kaldırılmış ve AYZ STS ilişkisi derinlemesine incelenmiş ve elde edilen kapsamlı bulgular yayınlanmıştır.
- Kara kutu probleminin çözümüne yönelik literatürde AYZ destekli STS çalışmaları kapsamlı olarak ele alınmış, 2018-2025 yılları arasında yapılan 386 adet çalışma ML/DL modeli, veri kümesi, AYZ modeli, lokal/global açıklama yeteneği, modele özgü/bağımsız özelliği ve post-hoc/içsel aşamalarına göre farklı perspektiflerden değerlendirilmiş ve sonuçlar Çizelge 2.2 ve Çizelge 2.3’de sunulmuştur. Böylece, alandaki ilk kapsamlı çalışma yapılmış, literatüre yeni bir bakış açısı kazandırılmış ve AYZ–STS alanındaki kritik boşluklar belirlenmiş ve çalışmalar yayınlanmıştır.

- Literatür çalışmalarından elde edilen bilgi, birikim ve deneyimler doğrultusunda, STS model kararlarının güvenle açıklanabilmesine yönelik özgün, çok boyutlu ve 3 adımdan oluşan bir açıklanabilir saldırı tespit sistemi çözümü (Şekil 6.1) geliştirilmiştir.
 - Veri ön işleme adımında; ham CIC-IDS2017 ve UNSW-NB15 veri kümeleri üzerinde veri temizleme, etiket kodlama, normalizasyon, öznitelik seçimi ve veri dengeleme işlemleri yapılmış olup öznitelik yaklaşımı ve veri dengeleme yaklaşımı olmak üzere iki yaklaşımdan oluşan yeni bir veri ön işleme algoritması geliştirilmiş ve literatürde ilk kez bu çalışmada kullanılmıştır.
 - Önerilen sistemde geliştirilen öznitelik seçim adımı, rastgele orman ve Pearson korelasyon matrisini içeren iki farklı teknikten oluşmaktadır. Veri kümeleri üzerinde geliştirilen yaklaşım kullanılarak öznitelik seçimi yapılmış ve sonuçlar paylaşılmıştır. Öznitelik seçiminin ardından veri dengesizliğinin oluşturduğu problemlerin üstesinden gelmek amacıyla STL olarak adlandırdığımız SMOTE aşırı örnekleme ve Tomek-Links alt örnekleme tekniklerinden oluşan bir veri dengeleme yaklaşımı geliştirilerek hem aşırı örnekleme hem de yetersiz örnekleme tekniklerinin dezavantajları ortadan kaldırılmıştır. Veri dengeleme işlemi sonucunda veri kümelerinin sınıf dağılımlarına ilişkin sonuçlar Şekil 7.10 ve Şekil 7.11’de gösterilmiştir.
 - Sınıflandırma adımında; LightGBM ve GCN yaklaşımlarından oluşan hibrit bir sınıflandırma algoritması geliştirilmiş olup hem örnek içi (feature-level) bilgiye hem de örnekler arası yapısal ilişkilere dayalı karar mekanizmasına sahip bir sınıflandırma algoritması literatürde ilk kez bu çalışmada kullanılmıştır. Geliştirilen modelin literatürdeki mevcut yapay zekâ modelleriyle performans karşılaştırılması yapılmış ve sonuçlar değerlendirilmiştir. Şekil 7.12, Şekil 7.16 ve Şekil 7.19’da gösterilen açıklanabilirlik çıktıları ile sınıflandırma karar mekanizmasında her iki yaklaşımdan gelen sonuçların da kullanıldığı gösterilmiş ve hibrit mimari doğrulanmıştır. Böylece, AYZ’nin modelleri şeffaflaştırarak geliştirme süreçlerine katkısı bu çalışma üzerinden doğrulanmıştır.
- AYZ modellerinin açıklanabilirliği bir YZ modeli ve onun performansına bağlı olarak değiştiğinden; geliştirilen sistemin en yüksek sistem performansını elde etmesi ve kapsamlı karşılaştırılması amacıyla literatürde sık kullanılan RF (Rastgele Orman), DT (Karar Ağacı), NB (Naive Bayes), LR (Lojistik Regresyon), MLP (Çok Katmanlı Algılayıcı), LSTM (Uzun Kısa Süreli Bellek), CNN (Evrişimsel Sinir Ağı), Gated Recurrent Unit (GRU), CNN-LSTM, CNN-GRU, eXtreme Gradient Boosting

(XGBoost), LightGBM, Graph Convolutional Network (GCN), Graph Attention Networks (GAT), Graph Isomorphism Network (GIN), GraphSAGE, LightGBM-GIN, XGBoost-GCN, XGBoost-GIN olmak üzere 19 farklı yapay zekâ modeli her iki veri kümesi üzerinde ikili ve çok sınıflı sınıflandırma testlerine tabii tutulmuş ve sonuçlar değerlendirilmiştir. Çizelge 7.5, Çizelge 7.6, Çizelge 7.7 ve Çizelge 7.8’de sunulan deneysel sonuçlardan, önerilen sınıflandırma algoritmasının literatürdeki mevcut yöntemlerden daha iyi performans sergilediği görülmüştür.

- Açıklanabilirlik adımı; SHAP ve GNNExplainer AYZ yaklaşımlarından oluşan yeni bir açıklanabilir AYZ algoritması önerilmiş, literatürde ilk kez çok boyutlu bir açıklanabilirlik yeteneğine sahip bir sistem geliştirilmiştir. Mevcut sistemin daha açıklanabilir olması için büyük dil modelleri (BDM) entegrasyonu sağlanarak elde edilen açıklanabilirlik çıktıları literatürde ilk kez ChatGPT ve Gemini sohbet robotları üzerinden yorumlatılmış ve kullanıcılara daha etkin açıklamalar üretilmiştir. Yapılan deneysel çalışmalar, SHAP ile elde edilen özellik önemi, waterfall ve plot çıktıları Şekil 7.17, Şekil 7.18, Şekil 7.20 ve Şekil 7.21’de sunularak öznetelik seviyesinde üretilen açıklamalar değerlendirilmiştir. Bununla birlikte, GNNExplainer ile elde edilen alt-graf ve özellik önemi çıktıları Şekil 7.12, Şekil 7.16 ve Şekil 7.19’da sunularak saldırıların ilişkisel yapısı değerlendirilmiş ve deneysel çalışmalar ile doğrulanmıştır.
- Önerilen sistemde kullanılan CIC-IDS2017 ve UNSW-NB15 veri kümeleri gerçek dünya uygulamalarında kullanılan Zeek açık kaynak STS üzerinde çalıştırılarak gerçek bir STS ortamında geliştirilen sistemlerin performansları değerlendirilmiştir. Yapılan deneysel çalışmalarda elde edilen sonuçlar Çizelge 7.11’de sunulmuştur.

Bu çalışma, STS literatüründeki yöntemsel boşlukları kapsamlı biçimde ortaya koymuş; hibrit sınıflandırma, çok boyutlu AYZ ve BDM entegrasyonunu bir araya getiren özgün bir mimari önermiştir. Elde edilen sonuçlar, yalnızca yüksek performanslı değil, aynı zamanda güvenilir, genellenebilir ve insan-odaklı STS çözümlerinin mümkün olduğunu göstermektedir.

Sonuç olarak tez çalışması kapsamında;

- STS ve AYZ ilişkisi derinlemesine incelenerek literatürdeki AYZ destekli STS çalışmalarını konu alan mevcut çalışmalarla kıyaslandığında, alandaki ilk kapsamlı çalışma yapılmıştır. Her iki alan için verilen detaylı bilgiler doğrultusunda bu çalışmanın

siber güvenlik alanında çalışan araştırmacılar için önemli bir kılavuz olacağı değerlendirilmektedir.

- Literatürdeki yapay zekâ destekli STS çalışmaları ile karşılaştırıldığında veri ön işleme adımlarının sistematik ve kapsamlı olarak ele alındığı ilk çalışma olduğu değerlendirilmektedir.
- Literatürde yapılan araştırmalar temel alındığında; önerilen hibrit sınıflandırma algoritmasını içeren herhangi bir çalışma bulunmamaktadır. Bu doğrultuda, LightGBM ve GCN algoritmalarının her ikisinin de karar mekanizmasında yer aldığı ilk sistematik YZ modeli bu çalışma kapsamında geliştirilmiştir.
- Geliştirilen sınıflandırma algoritmasının performansı literatürde sık kullanılan ML/DL teknikleriyle karşılaştırılmış ve performans açısından en yüksek performansa sahip model olduğu doğrulanmıştır.
- CIC-IDS2017 ve UNSW-NB15 veri kümeleri üzerinde yürütülen deneysel çalışmalarda önerilen hibrit sınıflandırma algoritması çok sınıflı sınıflandırmada olduğu gibi ikili sınıflandırmada da tüm modellerin üzerinde bir performans sergileyerek Precision, Recall, F1-Score, Specificity ve Accuracy metriklerinin tamamında yüksek performans değerlerine ulaşmıştır. Yüksek performans sonuçlarının, boosting tabanlı öz nitelik temsilinin GNN tabanlı grafik temsil ile harmanlanmasının doğrudan sonucu olduğu değerlendirilmektedir.
- Her iki veri kümesi üzerinde yapılan deneysel çalışmalarda GNN temelli modellerin performansı boosting temelli modellerin gerisinde kalmıştır. Veri setleri graf tabanlı olmadığı sürece bu sonucun doğal olduğu değerlendirilmektedir.
- Sınıflandırma performans sonuçlarından boosting tabanlı öğrenme modellerine GNN tabanlı modeller aracılığıyla ilişkisel bilgi eklemenin sınırlı ama anlamlı bir katkı sağladığı değerlendirilmektedir.
- Literatürdeki yapılan araştırmalar temel alındığında, çalışmaların birçoğu SHAP/LIME analizine odaklanmakla birlikte sınırlı sayıda GNNExplainer ile açıklanabilirlik sağlanmıştır. Bu çalışma, ağaç-temelli SHAP açıklamalarını ve GNN temelli alt-graf açıklamalarını bütüncül bir şekilde yorumlayan ilk sistematik yaklaşımı sunmaktadır. Bu yaklaşım ile açıklanabilirlik çıktılarına çok yönlü bir bakış açısının kazandırıldığı değerlendirilmektedir.
- Zeek açık kaynak STS ile yapılan deneysel çalışmalarda saldırıların tespitine yönelik gerekli paket ve politikalar yüklenmiş olsa bile Zeek'in birçok saldırıyı tespit edemeyip

yüksek performans sergileyemediği gösterilmiştir. Her iki veri kümesinin Zeek performanslarının yakın olmaması altyapıya özgü konfigürasyon gereksinimini açıkça gösterdiği değerlendirilmektedir.

Bununla birlikte;

- Bu çalışmada özetlenen katkılar ile modern siber güvenlik ortamlarında STS'lerin güvenliğini, güvenilirliğini ve yorumlanabilirliğini artırmak için güçlü ML/DL modellerini AYZ teknikleriyle birleştirmenin önemi vurgulanmıştır.
- Literatürde önerilen STS modellerinin performansını değerlendirmek için çeşitli kıyaslama veri kümelerinin (CICIDS2017, UNSW-NB15, NSL-KDD, vb.) kullanıldığı gösterilmiştir.
- STS veya diğer siber tehditlerin tespiti ve sınıflandırılmasını iyileştirmek için AE, CNN, RNN ve topluluk modelleri gibi gelişmiş modeller mevcuttur. Derin öğrenme mimarileri (CNN'ler, LSTM'ler, AE'ler gibi) saldırı tespiti için kullanılmış ve ağ trafiğini sınıflandırmada etkili olduğu gösterilmiştir. Özellikle, LIME ve SHAP modellerinin, Random Forest, XGBoost, LSTM, CNN ve Gradient Boosting ML/DL modelleri kullanan açıklanabilir modellerde etkili bir şekilde kullanıldığı gösterilmiştir.
- YZ model seçimi, tespit etmesi gereken saldırı türü ve mevcut hesaplama kaynakları gibi STS'nin özel gereksinimlerine bağlıdır. Açıklanabilir STS'ler, saldırıların nasıl çalıştığı ve bunlara karşı nasıl daha iyi savunma yapılacağı konusunda bilgi sağlayarak güvenliğini iyileştirmeye yardımcı olabilmektedir. Güvenlik analistleri, bu açıklamaları ağ güvenlik açıklarını belirlemek ve daha etkili güvenlik politikaları geliştirmek için kullanabilirler. AYZ, DDoS, DoS, sıfırıncı gün web saldırıları ve kötü amaçlı yazılımlar dâhil olmak üzere çeşitli saldırıların tespitinde yardımcı olmaktadır.
- Model tahminlerine katkıda bulunan ve modellerin yorumlanabilirliğine, açıklanabilirliğine ve verimliliğine yardımcı olan en alakalı özellikleri belirlemek için özellik seçimi tekniklerinin uygulanması, dikkate alınması gereken önemli bir sorundur. Bu çalışmada saldırı tespiti için en önemli özellikleri belirleyerek model yorumlanabilirliğini ve verimliliğini artırmak için özellik seçimi teknikleri araştırılmıştır. AYZ yöntemleri, analistlerin belirli bir saldırının neden tespit edildiğini anlamalarına yardımcı olarak özellik önemine dair bilgiler sağlamaktadır.
- Gelecek için gerçek zamanlı, ölçeklenebilir ve sağlam STS çözümlerinin geliştirilmesi gereklidir. STS sistemine açıklanabilirlik özelliği eklemek, büyük ölçekli ağ trafiğini ve

karmaşık siber saldırıları etkili bir şekilde ele almak için gerçek bir zorluktur. Dahası, bu gelişmiş YZ modellerinin ve AYZ yöntemlerinin IoT, IIoT, siber istihbarat, araç ağları vb. dâhil olmak üzere çeşitli alanlarda uygulanması da çok yönlülükleri ve etkinliklerini gösteren diğer göstergelerdir.

Ayrıca;

- Düşman saldırıları altında GenAI modellerinin ve veri kümelerinin gizlilik ve güvenlik sorunlarının yeni zorluklar olacağı değerlendirilmektedir.
- Açıklanabilirliği artırmak için yeni yapay zekâ modellerinin oluşturulması, ayarlanması, dikkat mekanizması ve verilerinin geliştirilmesinin yeni araştırma alanları olacağı düşünülmektedir.
- Daha iyi yapay zekâ ve AYZ modelleri geliştirmek için veri kümesi kalitesi ve boyutu daha önemli hale gelmiştir. Veri işleme süreçlerinin ve altyapılarının ilerde kritik bileşenler olacağı düşünülmektedir. Bununla birlikte, Şeffaflığı ve güveni artırmak için AYZ tekniklerinin giderek daha fazla STS'ye uygulanacağı değerlendirilmektedir.
- STS modellerinin düşman saldırılarına karşı dayanıklılığını artırmak için AYZ'yi düşman eğitimle(adversarial training) birleştirmenin umut vadeden bir araştırma yönü olacağı düşünülmektedir.
- Doğruluk ve açıklanabilirlik arasındaki dengeyi göz önünde bulundurarak açıklanabilir STS geliştirme ile ilgili zorluklar hala mevcuttur. STS modellerinin açıklanabilirliği ve doğruluğunu dengelemek devam eden bir zorluk olmaya devam etmektedir. Bu alandaki araştırmaların devam edeceği ve gelecek için umut verici olacağı değerlendirilmektedir.
- Gelecekte siber güvenlik analistleri ve tasarımcılarının, STS kararlarının anlaşılması konusunda iyileştirilmiş sonuçlardan ve kararlardan daha fazla faydalanabileceği değerlendirilmektedir.
- Gelişmiş tespit tekniklerinin geliştirilmesi için her zaman yeni ve kompakt yapay zekâ ve AYZ modellerine, veri kümelerine, platformlara ve uygulamalara ihtiyaç duyulacağı değerlendirilmektedir.
- AYZ yöntemlerinin sayısı gün geçtikçe önemli ölçüde artmaktadır. Özellikle GenAI'deki son gelişmelerin, yeni AYZ tekniklerinin geliştirilmesinde yeni bakış açıları getirebileceği değerlendirilmektedir. Bununla birlikte, yapay zeka, makine öğrenmesi, derin öğrenme, AYZ ve özellikle GenAI'nın, yalnızca STS'lere değil, diğer siber güvenlik sorunlarına da yeni bakış açıları getireceği değerlendirilmektedir.

Bu tez çalışmasında özetlendiği üzere, AYZ modelleri belirtilen STS sorunlarına çözümler sunmaktadır. AYZ, YZ kararlarını anlamaya yardımcı olmak için çeşitli yöntemler kullanır ve bu kararları hangi verilerin veya veri bölgelerinin etkilediğini gösterir. AYZ'nin birçok alanda ilgi görmesi ve uygulanmasına rağmen, STS'lerde kullanımı sınırlı kalmaktadır. Veri ihlallerini ve saldırılarını tespit etmede kritik öneme sahip olan STS'ler, yüksek performansa ve kararlarını gerekçelendirme yeteneğine ihtiyaç duymaktadır. AYZ yöntemlerini STS'lerde uygulamak, kararları açıklamayı, sonuçları analiz etmeyi ve bunları anlaşılır formatlara dönüştürmeyi gerektirmektedir. AYZ destekli STS'ler için literatürdeki tekniklerin neredeyse tamamı post-hoc ve modelden bağımsız çalışmaktadır. Bunun nedeni, mevcut AYZ tekniklerinin STS'lere ek bir modül olarak eklenmesidir. STS'lere özgü açıklanabilirlik yaklaşımlarının geliştirilmesi siber güvenlik ekosistemine katkıda bulunacak olsa da, bu yaklaşımlar yeni güvenlik açıklarını da beraberinde getirebilmektedir.

Bu tez çalışmasının sonucunda, AYZ modelleri için dikkate alınması, yeni çözümler geliştirilmesi ve tartışılması gereken birkaç önemli konu bulunmaktadır:

- Açıklanabilirlik bugün isteğe bağlı olarak değerlendirilebilir, ancak YZ modelleri ve karar alma sistemlerinin büyümesiyle birlikte, gelecekte açıklama, tasarım, güvenlik, anonimleştirme, uygulama, doğrulama ve belirli kriterlere göre test etme gibi bir dizi prosedürün ele alınması şüphesiz gerekecektir.
- Açıklanabilirlik yoluyla YZ sistemlerine güven oluşturmak kritik öneme sahiptir, ancak her zaman mümkün değildir; özellikle de analistlerin AYZ yaklaşımlarının sunduğu açıklamaları anlamaları ve kabul etmeleri gerekmektedir.
- DL veya ML modelleri gibi kara kutu modellerini şeffaf hale getirmek, karmaşık ve doğrusal olmayan yapıları nedeniyle özellikle zordur. SHAP, LIME, ELI5 vb. yaklaşımlar bunun için yardımcı olur, ancak bu modellerin karar alma sürecini tam olarak desteklemek için her zaman yeterli değildir.
- Şeffaf ve anında karar almaya olanak tanıyan AYZ'yi gerçek zamanlı uygulamalara, özellikle de STS'lere entegre etmek önemli son zamanlardaki önemli zorluklardan biridir. Literatürde çeşitli AYZ teknikleri incelendiğinde LIME ve SHAP'ın son zamanlarda STS'lerde kullanılan ana teknikler olsa da STS'lerin dinamik bir yapı içerisinde hizmet verdiği değerlendirildiğinde bu sistemlerin daha güvenilir yapılara ihtiyaç vardır.
- Genel olarak, STS'lerdeki açıklamalar, analistlerin kalıpları anlamalarına, anomalileri tespit etmelerine, saldırıları sınıflandırmalarına, özellikle AYZ tabanlı STS'ler olmak

üzere daha iyi ve akıllı sistemler tasarımlarına yardımcı olabilir. Literatürdeki IDS çalışmaları incelendiğinde, hem yerel hem de global açıklama tekniklerinin birlikte kullanılması, kullanıcılara daha ayrıntılı açıklamalar sunmakta ve daha iyi ve daha güvenilir STS'ler geliştirmeye yardımcı olacaktır.

- Birçok STS modeli, sonuçları siber güvenlik uzmanlarının kararlarını gözden geçirmelerine olanak tanıyan açıklamalar sunmadığı için pratik uygulamalarda kullanılmamıştır. AYZ, güvenlik uzmanlarına ihlal riskini azaltmak ve STS'lerden başlayarak genel güvenliği iyileştirmek için temel akıl yürütme ve tehdit tespit yetenekleri sağlayabilir.
- BDM temelli sohbet programları uzman olmayanlara STS uyarı ve kararlarını açıklamak için birçok fayda sağlasa da, olağanüstü yetenekleri ve karmaşık yapıları nedeniyle açıklanabilirlik açısından bazı zorlukları da beraberinde getirmektedir. BDM'nin karar alma süreçlerinde şeffaflığı sağlaması ve üretilen çıktıları açık bir şekilde açıklaması güven oluşturmada önemlidir.

Gelecek çalışmalarda STS'lere ve saldırılara özgü açıklanabilirlik tekniklerinin geliştirilerek siber güvenlik ihtiyaçları doğrultusunda özelleştirilmiş açıklama çıktılarının elde edilebileceği değerlendirilmektedir. Bu sayede elde edilen çıktıların siber saldırılara karşı savunma mekanizmasını daha da güçlendireceği düşünülmektedir. Bununla birlikte, alana özgü açıklamaların üretilerek STS çıktılarına olan güvenin ve kullanılabilirliğin artırılacağı değerlendirilmektedir.

KAYNAKLAR

1. İnternet: Artificial Intelligence Market Summary, URL-1: <https://www.grandviewresearch.com/industry-analysis/artificial-intelligence-ai-market>, Son Erişim Tarihi: 24.11.2025.
2. İnternet: Market Overview, URL-2: <https://www.nextmsc.com/report/explainable-ai-market>, Son Erişim Tarihi: 24.11.2025.
3. İnternet: Google Scholar, URL-3: <https://scholar.google.com>, Son Erişim Tarihi: 24.11.2025.
4. İnternet: IEEE Xplore Digital Library, URL-4: <https://ieeexplore.ieee.org>, Son Erişim Tarihi: 24.11.2025.
5. İnternet: SpringerLink, URL-5: <https://link.springer.com>, Son Erişim Tarihi: 24.11.2025.
6. İnternet: ScienceDirect, URL-6: <https://sciencedirect.com>, Son Erişim Tarihi: 24.11.2025.
7. İnternet: Applications of artificial intelligence to network security. URL-7: <https://arxiv.org/abs/1803.09992> Son Erişim Tarihi: 24.11.2025.
8. Wickramasinghe, C.S., Marino, D.L., Amarasinghe, K. (27-29 Kasım 2018). *Generalization of deep learning for cyber-physical system security: A survey*. In Proceedings of the 44th IEEE Industrial Electronics Society Annual Conference (IECON), Varşova, 745–751.
9. Shaukat, K., Luo, S., Varadharajan, V., Hameed, I. A., Xu, M. (2020). A survey on machine learning techniques for cyber security in the last decade. *IEEE Access*, 8, 222310–222354.
10. Mishra, P., Varadharajan, V., Tupakula, U., Pilli, E. S. (2019). A detailed investigation and analysis of using machine learning techniques for intrusion detection. *IEEE Communications Surveys & Tutorials*, 21(1), 686–728.
11. Gümüşbaş, D., Yıldırım, T., Genovese, A., Scotti, F. (2021). A comprehensive survey of databases and deep learning methods for cybersecurity and intrusion detection systems. *IEEE Systems Journal*, 15(2), 1717–1731.
12. Muneer, S., Farooq, U., Athar, A. (2024). A critical review of artificial intelligence-based approaches in intrusion detection: A comprehensive analysis. *Journal of Engineering*, 2(24), 1–16.
13. Kuppa, A., Le-Khac, N.A. (2021). Adversarial XAI methods in cybersecurity. *IEEE Transactions on Information Forensics and Security*, 16, 4924–4938.

14. İnternet: XAI for cybersecurity: State of the art, challenges, open issues and future directions. URL-8: <https://arxiv.org/abs/2206.03585> Son Erişim Tarihi: 25.11.2025.
15. Capuano, N., Fenza, G., Loia, V., Stanzione, C. (2022). Explainable artificial intelligence in cybersecurity: A survey. *IEEE Access*, 10, 93575–93600.
16. Fabien, C., Tanuwidjaja, H.C., Ayoubi, S. (2022). Explainable artificial intelligence for cybersecurity: A literature survey. *Annals of Telecommunications*, 77, 789–812.
17. Zhang, Z., Hamadi, H.A., Damiani, E., Yeun, C.Y. (2022). Explainable artificial intelligence applications in cyber security: State-of-the-art research. *IEEE Access*, 10, 93104–93139.
18. Rjoub, G., Bentahar, J., Abdel Wahab, O. (2023). A survey on explainable artificial intelligence for cybersecurity. *IEEE Transactions on Network and Service Management*, 20(4), 5115–5140.
19. Rudin, C. (2019). Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature Machine Intelligence*, 1, 206–215.
20. Sahakyan, M., Aung, Z., Rahwan, T. (2021). Explainable artificial intelligence for tabular data: A survey. *IEEE Access*, 9, 135392–135422.
21. Alicioglu, G., Sun, B. (2021). A survey of visual analytics for explainable artificial intelligence methods. *Computers & Graphics*, 102, 502–520.
22. Ahmed, I., Jeon, G., Piccialli, F. (2022). From artificial intelligence to explainable artificial intelligence in Industry 4.0: A survey on what, how, and where. *IEEE Transactions on Industrial Informatics*, 18(8), 5031–5042.
23. İnternet: Explainable Artificial Intelligence (XAI) for Internet of Things: A survey. URL-9: <https://arxiv.org/abs/2206.04800> Son Erişim Tarihi: 25.10.2025.
24. Chamola, V., Hassija, V., Sulthana, A.R. (2023). A review of trustworthy and explainable Artificial Intelligence (XAI). *IEEE Access*, 11, 78994–79015.
25. Letrache, K., Ramdani, M. (13-14 Kasım 2023). *Explainable artificial intelligence: A review and case study on model-agnostic methods*. In Proceedings of the 14th International Conference on Intelligent Systems: Theories and Applications (SITA), Bükreş, 1–8.
26. Mohale, V.Z., Obagbuwa, I.C. (2025). A systematic review on the integration of explainable artificial intelligence in intrusion detection systems to enhancing transparency and interpretability in cybersecurity. *Frontiers in Artificial Intelligence*, 8, 1526221.
27. Rawat, M., Singal, G. (2025). Surveying technology fusion in 10t networks for IDS: Exploring datasets, tools, challenges, and research prospects. *ACM Transactions on Intelligent Systems and Technology*, 16(5),107.

28. Stanek, D., Klaban, I., Coufalíková, A. (4-6 Mayıs 2025). *Explainable artificial intelligence: State of the art and beyond*. In 2025 International Conference on Military Technologies (ICMT), Mumbai, 1–7.
29. Riba, M., Askari, S.M.S., Prasad, R.K. (21-24 Eylül 2025). *Applications of machine learning and deep learning in intrusion detection system: A comprehensive survey*. In 2025 3rd International Conference on Intelligent Systems, Advanced Computing and Communication (ISACC), Berlin, 1164–1172.
30. Alketbi, K. S., Mehmood, A. (2025). A comprehensive survey of explainable Artificial Intelligence techniques for malicious insider threat detection. *IEEE Access*, 13, 121772–121798.
31. Mahmoud, M.M., Youssef, Y.O., Abdel-Hamid, A.A. (2025). XI2S-IDS: An explainable intelligent 2-stage intrusion detection system. *Future Internet*, 17(1), 25.
32. Shoukat, S., Gao, T., Javeed, D., Saeed, M.S., Adil, M. (2025). Trust my IDS: An explainable AI integrated deep learning-based transparent threat detection system for industrial networks. *Computers & Security*, 149, 104191.
33. Fatema, K., Dey, S.K., Anannya, M., Khan, R.T., Rashid, M.M., Su, C., Mazumder, R. (2025). Federated XAI IDS: An explainable and safeguarding privacy approach to detect intrusion combining federated learning and SHAP. *Future Internet*, 17(6), 234.
34. Bin Hulayyil, S., Li, S., Saxena, N. (2025). Explainable AI-based intrusion detection in IoT systems. *Internet of Things*, 31, 101589.
35. Hosain, Y., Çakmak, M. (2025). XAI-XGBoost: An innovative explainable intrusion detection approach for securing internet of medical things systems. *Scientific Reports*, 15, 22278.
36. Muhammad, A.E., Yow, K.C., Baćanin-Džakula, N. (2025). L-xaids: A LIME-based explainable AI framework for intrusion detection systems. *Cluster Computing*, 28, 654.
37. Wali, S., Farrukh, Y.A., Khan, I. (2025). Explainable AI and random forest-based reliable intrusion detection system. *Computers & Security*, 157, 104542.
38. Wang, Y., Azad, M.A., Zafar, M., Gul, A. (2025). Enhancing AI transparency in IoT intrusion detection using explainable AI techniques. *Internet of Things*, 33, 101714.
39. Salloum, S., Norozpour, S. (2025). XAI-IDS: A transparent and interpretable framework for robust cybersecurity using explainable artificial intelligence. *SHIFRA*, 614(2), 9–80.
40. İnternet: A versatile XAI-based framework for efficient and explainable intrusion detection systems. URL-10: <https://link.springer.com/article/10.1007/s12243-025-01118-9> Son Erişim Tarihi: 25.11.2025.

41. Alamro, H., Alahmari, S., Nemri, N. (2025). Enhanced intrusion detection in cybersecurity through dimensionality reduction and explainable artificial intelligence. *Scientific Reports*, 15, 33848.
42. Boubertakh, O., Harous, S., Zitouni, A., Sahnoun, A. (2-5 Mart 2025). *XD-IDS: Explainable deep learning-based intrusion detection system for IoT*. In 2025 7th International Conference on Pattern Analysis and Intelligent Systems (PAIS), Ürdün, 1–6.
43. Rajagopalan, N. (2025). Federated learning and explainable AI-driven intrusion detection with hyperband optimization. *Journal of Computer Virology and Hacking Techniques*, 21(2), 28.
44. Al-Hchaimi, A.A.J., (11-13 November 2025). *Enhancing cybersecurity in cyber-physical systems: An explainable AI approach for intrusion detection*. In 2025 5th International Conference on Emerging Smart Technologies and Applications (eSmarTA), Varşova, 1–8.
45. Tserenkhuu, M., Hossain, M.D., Taenaka, Y., Kadobayashi, Y. (2025). Intrusion detection system framework for SDN-based IoT networks using deep learning approaches with XAI-based feature selection techniques and domain-constrained features. *IEEE Access*, 13, 136864–136880.
46. Alatawi, M.N. (2025). Enhancing intrusion detection systems with advanced machine learning techniques: An ensemble and explainable artificial intelligence (AI) approach. *Security and Privacy*, 8, e496.
47. Shaker, B.N., Al-Musawi, B., Hassan, M. F. (2025). Explainable AI for enhancing IDS against advanced persistent kill chain. *Cluster Computing*, 28, 459.
48. Alabbadi, A., Bajaber, F. (2025). An intrusion detection system over IoT data streams using explainable artificial intelligence (XAI). *Sensors*, 25 (3), 847.
49. Mohale, V. Z., Obagbuwa, I. C. (2025). Evaluating machine learning-based intrusion detection systems with explainable AI: Enhancing transparency and interpretability. *Frontiers in Computer Science*, 7, 28-41.
50. Dasari, A., Bisawas, S.K., Purkayastha, B. (2025). Enhanced network intrusion detection systems with explainable artificial intelligence for network security. *International Journal of Communication Systems*, 38(14), e70209.
51. Uccello, F., Pawlicki, M., D'Antonio, S., Kozik, R., Choraś, M. (2025). A new cybersecurity approach enhanced by xAI-derived rules to improve network intrusion detection and SIEM. *Computers, Materials & Continua*, 83(2), 1607–1621.
52. Ebrahimi, F., Javidan, R., Akbari, R., (2025). Intrusion detection in the Internet of Things using convolutional neural networks: An explainable AI approach. *Cybersecurity*, 8, 66.

53. Rahman, S. (2025). DAE-IDS: A domain-aware ensemble intrusion detection system with explainable AI for industrial IoT networks. *International Journal of Advanced Computer Science and Applications*, 16(9), 29-37.
54. Alabdulatif, A. (2025). A novel ensemble deep learning approach for cybersecurity intrusion detection with explainable artificial intelligence. *Applied Sciences*, 15(14), 7984.
55. Mohammed, S.J., Nema, B.M. (2025). Threat detection based on explainable AI (XAI) and hybrid learning. *Mesopotamian Journal of CyberSecurity*, 5(2), 477–490.
56. Fernández-Morales, E., Tobarra, L., Robles-Gómez, A., Pastor-Vargas, R., Hernández, R., Sarraipa, J. (2025). eXplicitability AI (XAI) for attack detection toward smart rural applications. *Internet of Things*, 34, 101804.
57. Boubertakh, O., Harous, S., Zitouni, A., Sahnoun, A. (9-11 March 2025). *XD-IDS: Explainable deep learning-based intrusion detection system for IoT*. In 2025 7th International Conference on Pattern Analysis and Intelligent Systems (PAIS), Bükreş, 1–6.
58. Garikapati, H., Challapalli, K., Ramineni, S.V., Adusumilli, V.M.S., Kothamasu, R.S.C., Anamalamudi, S. (2025, 23-26 Mart). *An explainable AI-driven hybrid model for enhanced intrusion detection in network security*. In 2025 Fifth International Conference on Advances in Electrical, Computing, Communication and Sustainable Technologies (ICAECT), Sofya, 1–7.
59. Kalakoti, R., Vaarandi, R., Bahşi, H., Nömm, S. (19-22 November 2025). *Evaluating explainable AI for deep learning-based network intrusion detection system alert classification*. In Proceedings of the 11th International Conference on Information Systems Security and Privacy, Dabriç, 47–58.
60. Tewari, T., Singal, G. (2025, 20-22 January). *Towards white-box IDS: Integrating explainability in IoT ecosystems*. In 2025 National Conference on Communications (NCC), 1–6.
61. Hermosilla, P., Berríos, S., Allende-Cid, H. (2025). Explainable AI for forensic analysis: A comparative study of SHAP and LIME in intrusion detection models. *Applied Sciences*, 15(13), 7329.
62. Sadhwani, S., Navare, A., Mohan, A., Muthalagu, R., Pawar, P.M. (2025). IoT-based intrusion detection system using explainable multi-class deep learning approaches. *Computers & Electrical Engineering*, 123(2), 110256.
63. Dhruv, G., Parshad, U., Rajkumar, S., M.K., Mohammed, F., Rutvij, H. (2025). Enhancing anomaly detection in industrial control systems through supervised learning and explainable artificial intelligence. *Journal of Cybersecurity and Information Management*, 6(1), 314–331.
64. Almadhor, A., Alsubai, S., Kryvinska, N., (2025). Transfer learning with XAI for robust malware and IoT network security. *Scientific Reports*, 15, 26971.

65. Saleh, R.A.A., Saleh, B.A.Q., Al-bashiri, A.F.A., Al-wajih, E., Ertunç, H.M., Al-antari, M. A. (06-09 December 2025). *Enhancing IoT security through the integration of explainable AI and ensemble XGBoost for improved intrusion detection*. In 2025 IEEE 4th International Conference on Computing and Machine Intelligence (ICMI), Sofya, 1–7.
66. Xue, Q., Zhang, Z., Fan, K., Wang, M. (2025). xIIRS: Industrial internet intrusion response based on explainable deep learning. *Electronics*, 14(5), 987.
67. Purshottam, J.,N.V., Mohanambal, K., Chitra, R. (2025). Explainable artificial intelligence-driven intrusion detection system for enhancing reliability and interpretability in IoT-based network security solutions. *Journal of Intelligent Systems and Internet of Things*, 1(1),219–238.
68. İnternet: Interpretable anomaly-based DDoS detection in AI-RAN with XAI and LLMs URL-11: <https://arxiv.org/abs/2507.21193> Son Erişim Tarihi: 20.10.2025.
69. Kilincer, İ.F. (2025). Explainable AI-supported hybrid deep learning method for layer 2 intrusion detection. *Egyptian Informatics Journal*, 30, 100669.
70. Oyedotun, S.A., Oise, G. P., Ozobialu, C.E. (2025). Towards intelligent cybersecurity in SCADA and DCS environments: Anomaly detection using multimodal deep learning and explainable AI. *Journal of Science Research and Reviews*, 2(3), 20–31.
71. Alzaylaee, M.K. (2025). Enhancing cybersecurity through artificial intelligence: A novel approach to intrusion detection. *International Journal of Advanced Computer Science and Applications*, 16(4),130-147.
72. Fatima, Z., Hussain, R., Dilshad, A., Shakir, M., Laghari, A.A. (2025). Explainable AI-driven firewall evaluation: Empowering cybersecurity decision-making for optimal network defense. *Security and Privacy*, 8, e70026.
73. İnternet: Improving IIoT anomaly detection precision via XAI-guided feature engineering and LSTM tuning on imbalanced data under resource constraints. URL-12: https://link.springer.com/chapter/10.1007/978-3-032-07373-0_49 Son Erişim Tarihi: 20.10.2025.
74. İnternet: A lightweight IDS for early APT detection using a novel feature selection method, URL-13: <https://arxiv.org/abs/2506.12108> Son Erişim Tarihi: 20.10.2025.
75. Dhiya Eddine, A., Abdelkader, G., Mourad, B. (2025). Enhancing IIoT security: A hybrid AI framework for intrusion detection powered by Gen-X AI. *International Journal of Computers and Applications*, 47(9), 749–765.
76. İnternet: Trustworthy adaptive AI for real-time intrusion detection in Industrial IoT security. URL-14: <https://www.mdpi.com/2624-831X/6/3/53> Son Erişim Tarihi: 20.10.2025.
77. Bilal, M., Zeng, R., Muhammad, O., (2025). A cyber-resilient and explainable intrusion detection system for Internet of Things networks. *Cluster Computing*, 28, 308.

78. Alabbadi, A., Bajaber, F. (2025). X-FuseRLSTM: A cross-domain explainable intrusion detection framework in IoT using attention-guided dual-path feature fusion and residual LSTM. *Sensors*, 25(12), 3693.
79. Toluwaleke, O., (2025, 11-14 Mart). *Temporal-XAI: An attention-based explainable framework for DoS attack detection with temporal pattern analysis*. In 2025 10th International Conference on Machine Learning Technologies (ICMLT), Pekin, 439–450.
80. Ravi Kumar, M., Madhu, G.C., Narra, V.B., Chidipothu, V.K., Chakradhar, K. S., Sawan, V. (2025, 13-15 Nisan). *Improved intrusion detection in cyber-physical systems with explainable AI and hybrid optimization*. In 2025 IEEE International Conference on Emerging Technologies and Applications (MPSec ICETA), Brest. 1–6.
81. Khandan, S., Beyazgul, D., Jogunola, O., Tsado, Y., Dargahi, T. (2025, 19-22 Kasım). *Explainable AI-driven threat detection and response for Industrial IoT*. In 2025 IEEE Conference on Communications and Network Security (CNS), 1–6.
82. Zhang, J., Dai, Q., Zhou, X., Chen, L. (2025). NonsaliencyCrossover: A novel adversarial example generation strategy for IDS using explainable AI. *IEEE Internet of Things Journal*, 12(18), 38680–38696.
83. Metwally, M., Zamzam, M. (2025, 15-18 Eylül). *FEX-IDS: A federated learning-based explainable intrusion detection system for IoT security*. In 2025 5th International Conference on Innovative Research in Applied Science, Engineering and Technology (IRASET), Bologna. 1–7.
84. Ahmad Bilal, M., Ul Islam, I., Iltaf, N., Junaid Khan, M., Jaleed Khan, M. (2025). Federated learning with explainable AI for malicious traffic detection in IoT networks. *IEEE Access*, 13, 173368–173383.
85. Hemalatha, A., Suresh, R. (2025, 06-09 Mayıs). *A hybrid intrusion detection system using explainable AI for enhanced accuracy and transparency*. In 2025 International Conference on Electronics and Renewable Systems (ICEARS), Minsk. 923–929.
86. Wan, X., Xue, G., Zhong, Y., (2025). Separating prediction and explanation: An approach based on explainable artificial intelligence for analyzing network intrusion. *Journal of Network and Systems Management*, 33, 16-21.
87. Bibers, I., Abdallah, M. (2025, 27-29 Mart). *Ensemble feature selection for network intrusion detection systems using explainable AI: A frequency-based approach*. In 2025 IEEE 4th International Conference on Computing and Machine Intelligence (ICMI), Roma. 1–7.
88. Krishnaveni, S., Zeadally, S., Sivamohan, S. (2025). SmartHive-IDS: An advanced intrusion detection framework for SDN-HoneyNet-enabled ICPS using hybrid explainable ensemble feature selection with optimized Bi-LSTM. *Cluster Computing*, 28, 913.

89. Nalinipriya, G., Rama Sree, S., Radhika, K. (2025). Leveraging explainable artificial intelligence for early detection and mitigation of cyber threats in large-scale network environments. *Scientific Reports*, 15, 24662.
90. Ahmed, U., Jiangbin, Z., Khan, S. (2025). HCIVAD: Explainable hybrid voting classifier for network intrusion detection systems. *Cluster Computing*, 28, 343.
91. Arreche, O., Abdallah, M. (2025). A comparative analysis of DNN-based white-box explainable AI methods in network security. *EURASIP Journal on Information Security*, 16(2), 23-37.
92. Arun Kumar, S., Sasikala, S., Dhanusha, T. (2025, 11-14 Şubat). *Explainable AI based improved intrusion detection system in wireless sensor networks*. In 2025 3rd International Conference on Advancements in Electrical, Electronics, Communication, Computing and Automation (ICAECA), Islamabad.. 1–6.
93. Bacevicius, M., Paulauskaite-Taraseviciene, A., Zokaityte, G., Kersys, L., Moleikaityte, A. (2025). Comparative analysis of perturbation techniques in LIME for intrusion detection enhancement. *Machine Learning and Knowledge Extraction*, 7(1), 21.
94. Ogunseyi, T.B., Thiyagarajan, G. (2025). An explainable LSTM-based intrusion detection system optimized by firefly algorithm for IoT networks. *Sensors*, 25(7), 2288.
95. Kumar, A. B. S., Achary, R. (2025, 24-27 Mayıs). *Explainable graph-based reinforcement learning for intrusion detection in cybersecurity*. In 2025 8th International Conference on Computing Methodologies and Communication (ICCMC), Verona. 1098–1104.
96. İnternet: *LENS-XAI: Redefining lightweight and explainable network security through knowledge distillation and variational autoencoders for scalable intrusion detection in cybersecurity* URL-15: <https://arxiv.org/abs/2501.00790> Son Erişim Tarihi: 20.10.2025.
97. Alohalı, M. A., Aljebreen, M., Ahmad, N. (2025). Privacy preserving blockchain integrated explainable artificial intelligence with two tier optimization for cyber threat detection and mitigation in the Internet of Things. *Scientific Reports*, 15, 36520.
98. Hoang, T.M., Nguyen, V.N., Le Thi, T.L. (2025). A hybrid intrusion detection system model integrating explainable AI and multi expert systems to adapt edge computing. *Cluster Computing*, 28, 649.
99. Holubenko, V., Gaspar, D., Leal, R. (2025). Autonomous intrusion detection for IoT: A decentralized and privacy-preserving approach. *International Journal of Information Security*, 24(2), 7-19.
100. Ahmad, J., Latif, S., Khan, I. U., Alshehri, M. S., Khan, M. S., Alasbali, N., Jiang, W. (2025). An interpretable deep learning framework for intrusion detection in industrial Internet of Things. *Internet of Things*, 33, 101681.

101. Asal, B., Cakin, A., Dilek, S. (2025, 14-16 Mart). *Enhancing industrial IoT cybersecurity with explainable AI: A SHAP and LIME-based intrusion detection methodology*. In 2025 7th International Congress on Human-Computer Interaction, Optimization and Robotic Applications (ICHORA), Zürich. 1–8.
102. İnternet: Enhancing botnet detection with machine learning and explainable AI: A step towards trustworthy AI security. URL-16: <https://doi.org/10.2139/ssrn.5193008> Son Erişim Tarihi: 10.08.2025.
103. İnternet: Enhancing IoT network security through adaptive curriculum learning and XAI URL-18: <https://arxiv.org/abs/2501.11618> Son Erişim Tarihi: 10.08.2025.
104. Hewa, T., Siriwardhana, Y., Ylianttila, M. (2025, 18-23 Ekim). *Leveraging explainable AI for adaptive adversarial DoS attack detection in 6G IoT networks*. In 2025 IEEE Wireless Communications and Networking Conference (WCNC), Pekin. 1–7.
105. Vishwakarma, M., Kesswani, N. (2025). StaEn-IDS: An explainable stacking ensemble deep neural network-based intrusion detection system for IoT. *IEEE Access*, 13, 109713–109728.
106. Ericson, A., Thar, K., Forsström, S. (10-13 Kasım 2025). *Enhancing intrusion detection in CPS and IIoT with lightweight explainable AI models*. In 2025 IEEE 21st International Conference on Factory Communication Systems (WFCS), Helsinki. 1–8.
107. Kalutharage, C.S., Liu, X., Chrysoulas, C. (2025). Neurosymbolic learning and domain knowledge-driven explainable AI for enhanced IoT network attack detection and response. *Computers & Security*, 151, 104318.
108. Hasan, T., Tasnim, S. (2025). Real-time explainable IoT security with machine learning and CTGAN-enhanced detection for resource-constrained devices. *Ad Hoc Networks*, 178, 103937.
109. Obidiagha, C.C., Rahouti, M., Drid, H. (2025). Iforestexplain: A dual-stage isolation forest framework with explainable AI for DoS/DDoS anomaly detection in IoT networks. *Cluster Computing*, 28(2), 668.
110. Farooq, M.S., Saleem, M., Khan, M.A., Khan, M.F., Siddiqui, S.Y., Aslam, M.S., Adnan, K.M. (2025). Interpretable federated learning model for cyber intrusion detection in smart cities with privacy-preserving feature selection. *Computers, Materials & Continua*, 85(3), 5183–5206.
111. Singh, S., Rathee, T. (2025). AI-powered cyber vigilance: Explainable threat detection for next-gen security. *International Journal of Environmental Science*, 11(4), 811–833.
112. Farhat, S., Patel, A., Barros, A. L. B. (2025). MF2S-CID: A dynamic multi-model framework for scalable and interpretable intrusion detection. *International Journal of Information Security*, 24(3), 165.

113. Saied, M., Guirguis, S. (2025). Explainable artificial intelligence for botnet detection in Internet of Things. *Scientific Reports*, 15(2), 7632.
114. Soni, V., (2025, 20-23 Eylül). *Explainable machine learning-based false data injection classification framework for AVs*. In 2025 IEEE International Conference on Communications Workshops (ICC Workshops), Luzern. 1924–1929.
115. Hariharan, S., Jerusha, Y.A., Suganeshwari, G., Ibrahim, S.P.S., Tupakula, U., Varadharajan, V. (2025). A hybrid deep learning model for network intrusion detection system using Seq2Seq and ConvLSTM-subnets. *IEEE Access*, 13, 30705–30721.
116. Saravanan, P., Balaji, A., Murugan, H., Muruganantham, M., Varadharajan, I. (2026). A novel approach to anomaly detection in critical CPSs: The ERXAI-MS algorithm. *Expert Systems with Applications*, 298(A), 129437.
117. Idima, S., Nwaga, P., Evah, P. (2025). Comprehensive analysis of SCADA system data for intrusion detection using machine learning. *Global Journal of Engineering and Technology Advances*, 22(2), 64–89.
118. Rajkumar, K., Shalinie, S.M. (2025). SHAP-based intrusion detection in IoT networks using quantum neural networks on IonQ hardware. *Journal of Parallel and Distributed Computing*, 204(4), 105133.
119. Izuazu, U.U., Nwakanma, C.I., Kim, D.S. (2025). Explainable and perturbation-resilient model for cyber-threat detection in industrial control systems networks. *Discover Internet of Things*, 5(2), 9.
120. Krishnan, D., Singh, S., Sugumaran, V. (2025). Explainable AI for zero-day attack detection in IoT networks using attention fusion model. *Discover Internet of Things*, 5(2), 83.
121. Sharma, N., Shambharkar, P.G. (2025). Multi-attention DeepCRNN: An efficient and explainable intrusion detection framework for Internet of Medical Things environments. *Knowledge and Information Systems*, 67, 5783–5849.
122. Mittal, S., Wazid, M., Singh, D.P., Das, A.K., Hossain, M. S. (2025). A deep learning ensemble approach for malware detection in Internet of Things utilizing explainable artificial intelligence. *Engineering Applications of Artificial Intelligence*, 139(A), 109560.
123. Senevirathna, T., Paltun, B., Fouladi, R., Wang, S., Liyanage, M. (24-28 Kasım 2025). *From insight to action: XAI-enhanced detection of DDoS attacks in software-defined networks*. 2025 IEEE 36th Annual International Symposium on Personal, Indoor and Mobile Radio. İstanbul. 121-129.
124. Nazreen, B.A., Sangeetha, S. (2025). Intrumer: A multi-module distributed explainable IDS/IPS for securing cloud environment. *Computers, Materials & Continua*, 82(1), 579–607.

125. Ahsan, S.I., Legg, P., Alam, S.M.I. (2025). An explainable ensemble-based intrusion detection system for software-defined vehicle ad-hoc networks. *Cyber Security and Applications*, 3(4), 100090.
126. Ambritta, P.N., Mahalle, P.N., Bhapkar, H.R. (2025). Improving explainable AI interpretability with mathematical models for evaluating explanation methods. *International Journal of Information Technology*. 3(5),207-226.
127. Sunkara, G. (2025). Explainable AI for cyber threat intelligence: Enhancing analyst trust. *Open Access Research Journal of Science and Technology*, 14, 29–40.
128. Wakili, A., Bakkali, S. (2025). A resilient IoT intrusion detection system using hybrid feature selection and explainable ensemble learning. *Results in Engineering*, 28, 107392.
129. Patel, T., Jhaveri, R., Thakker, D., Verma, S., Ingle, P. (11-14 Aralık 2025). *Enhancing cybersecurity in Internet of Vehicles: A machine learning approach with explainable AI for real-time threat detection*. In Proceedings of the 40th ACM/SIGAPP Symposium on Applied Computing (SAC), Napoli. 2024–2031.
130. Imtiaz, N., Wahid, A., Ul Abideen, S.Z., Kamal, M.M., Sehito, N., Khan, S., Virdee, B.S., Kouhalvandi, L. (2025). A deep learning-based approach for detecting Internet of Things intrusion attacks through optical networks. *Photonics*, 12(1), 35.
131. Khan, A., Li, Y., Shoukat, S. (2025). Towards secure IoT-enabled transportation: An explainable AI and deep learning-based approach for efficient threat detection. *Cluster Computing*, 28(2), 699.
132. Melendrez, A., Goncalves, H., Alomari, Z., Shao, Y. (12-15 Nisan 2025). *Boosted ensemble voting for intrusion detection: A SHAP-driven analysis of XGBoost and CatBoost*. IEEE International Conference on Service Operations and Logistics, and Informatics (SOLI)At, Singapur. 67-72.
133. Garroppo, R.G., Giardina, P.G., Landi, G., Ruta, M. (2025). Trustworthy AI and federated learning for intrusion detection in 6G-connected smart buildings. *Future Internet*, 17(5), 191.
134. Saxena, A., Mittal, S. (2025). CluSHAPify: Synergizing clustering and SHAP value interpretations for improved reconnaissance attack detection in IIoT networks. *International Journal of Performability Engineering*, 21(1), 36–47.
135. Lau, J. L. T., Pang, Y. H., Zarakovitis, C., Lim, H. S., Skordoulis, D., Ooi, S. Y., Chan, K. Y., Pang, W. L. (2025). BiTAD: An interpretable temporal anomaly detector for 5G networks with TwinLens explainability. *Future Internet*, 17(11), 482.
136. Patel, V., Shukla, H., Raval, A. (2025). Enhancing botnet detection with machine learning and explainable AI: A step towards trustworthy AI security. *International Journal for Multidisciplinary Research*, 7(3),121-138.

137. Dey, A. K., Gupta, G. P., Sahu, S. P. (2025). EMTD: Explainable malicious traffic detection model using hybrid deep learning techniques for Industrial IoT networks. *Arabian Journal for Science and Engineering*, 50, 12219–12238.
138. Randika, H., Sukarno, P., Wardana, A. A. (27-30 Kasım 2025). *Attack detection on edge IIoT using ensemble learning and XAI model transparency*. In 2025 International Conference on Advancement in Data Science, E-learning and Information System (ICADEIS), Tokyo. 1–6.
139. Turgut, Z., Başarslan, M.S. (2025). XBiDeep: A novel explainable artificial intelligence-based intrusion detection system for Internet of Medical Things environment. *Internet of Things*, 33, 101675.
140. Habibi, M.F., Sukarno, P. (2025, 7-10 Mart). *Analysis of attack detection on IoT using XGBoost algorithm and SHAP*. In 2025 International Conference on Information and Communication Technology (ICoICT), Moskova. 1–5.
141. Kayode Saheed, Y., Chukwuere, J.E. (2025). CPS-IIoT-P2Attention: Explainable privacy-preserving with scaled dot-product attention in cyber-physical system–industrial IoT network. *IEEE Access*, 13, 81118–81142.
142. Wazid, M., (2025). Explainable deep learning-enabled malware attack detection for IoT-enabled intelligent transportation systems. *IEEE Transactions on Intelligent Transportation Systems*, 26(5), 7231–7244.
143. Chettier, T.M., Boyina, V.A.K., Jorepalli, S., Singh, C., Gupta, N. (2025). *Scalable explainable AI with a cloud-native approach for cybersecurity threat detection*. In 2025 3rd International Conference on Advancement in Computation & Computer Technologies (InCACCT), Roma. 686–691.
144. Shiddiqi, A.M., Purba, R.C., Pratomo, B.A. (22-26 Mayıs 2025). *Detecting IoT botnet attacks using explainable artificial intelligence (XAI) with decision tree and random forest*. In 2025 International Conference on Smart Computing, IoT and Machine Learning (SIML), Paris. 1–6.
145. Gogoi, B., Suklabaidya, M., Das, D.K. (14-16 Haziran 2025). *Uncovering malicious activity: Interpretable web log analysis with XAI, SHAP, and LLM*. In 2025 IEEE Guwahati Subsection Conference (GCON), St.Petersburg 1–7.
146. Somayajula, R., Raghavan, P., Chippagiri, S., Ravula, P. (16-17 Ocak 2025). *Adaptive fuzzy-neural architectures for explainable intrusion detection in big data environments*. In 2025 Global Conference in Emerging Technology (GINOTECH), Zenit. 1–7.
147. Mathew, J.P., Babu, N.V. (21-23 Mayıs 2025). *Explainable hybrid framework for network intrusion detection using graph attention networks and advanced feature engineering*. In 2025 International Conference on Advances in Modern Age Technologies for Health and Engineering Science (AMATHE), Helsinki. 1–8.

148. Ullah, Z., (2025). Interpretable and adaptive GAN-BiLSTM approach for cyber threat detection in IoMT-based Healthcare 5.0. *IEEE Journal of Biomedical and Health Informatics*, 16(2),53-69.
149. Senthamizh, G., (2025). *Explainable AI-powered networking intrusion detection systems*. In 2025 3rd International Conference on Intelligent Data Communication Technologies and Internet of Things (IDCIoT), Batum. 399–405.
150. Bani Ahmad, A. Y. A., Sarkar, P., Goswami, B., Patil, P. R., Al-Said, K., Al Said, N. (19-22 Şubat 2025). *A framework for evaluating the effectiveness of explainability methods in deep learning*. In 2025 International Conference on Pervasive Computational Technologies (ICPCT), Madrid. 426–430.
151. Fan, M., Zuo, J., Zhu, J., Lu, Y. (2025). Explainable anomaly-based intrusion detection for specialized IoT environments enabled by rule extraction from autoencoder. *IEEE Internet of Things Journal*, 12(12), 19504–19521.
152. Alzakari, S. A., Aljebreen, M., Ahmad, N., (2025). Explainable artificial intelligence-based cyber resilience in Internet of Things networks using hybrid deep learning with improved chimp optimization algorithm. *Scientific Reports*, 15, 33160.
153. Kumar, A., Sharma, B., Nooniah, A. (2025). Secure blockchain-based intrusion detection for IoT networks. *Discover Computing*, 28, 226.
154. Kaur, N., Gupta, L. (2025). Securing the 6G–IoT environment: A framework for enhancing transparency in artificial intelligence decision-making through explainable artificial intelligence. *Sensors*, 25(3), 854.
155. Alsharaiah, M.A., Almaiah, M.A., Shehab, R., (2025). An explainable AI-driven transformer model for spoofing attack detection in Internet of Medical Things (IoMT) networks. *Discover Applied Sciences*, 7(3), 488.
156. Ahmed, U., Jiangbin, Z., Almogren, A., vd. (2024). Explainable AI-based innovative hybrid ensemble model for intrusion detection. *Journal of Cloud Computing*, 13(2), 150.
157. Hooshmand, M. K., Huchaiah, M. D., Alzighaibi, A. R., Hashim, H., Atlam, E.-S., Gad, I. (2024). Robust network anomaly detection using ensemble learning approach and explainable artificial intelligence (XAI). *Alexandria Engineering Journal*, 94,120–130.
158. İnternet: Lecture Notes in Computer Science, URL-17: <https://www.springer.com/gp/computer-science/lncs?srsId=AfmBOooSYejXInOcnc2gGI8PycoY9sm8vCKFBssQ2RDRIrwZWDhuKwYE> Son Erişim Tarihi: 06.11.2025
159. Arreche, O., Guntur, T., Abdallah, M. (2024). XAI-IDS: Toward proposing an explainable artificial intelligence framework for enhancing network intrusion detection systems. *Applied Sciences*, 14(10), 4170.

160. Attique, D., Hao, W., Ping, W., Javeed, D., Adil, M. (2024). *EX-DFL: An explainable deep federated-based intrusion detection system for industrial IoT*. In 2024 21st International Joint Conference on Computer Science and Software Engineering (JCSSE), Malaga. 358–364.
161. Kaya, K., Ak, E., Bas, S., Canberk, B., Oguducu, S.G. (15-18 Aralık 2024). *X-CBA: Explainability aided CatBoosted Anomal-E for intrusion detection system*. In ICC 2024 – IEEE International Conference on Communications, (ITNAC) Tiflis. 2288–2293.
162. Kumar, A., Fok, K.W., Thing, V.L.L. (23-27 Aralık 2024). *ExpIDS: A drift-adaptable network intrusion detection system with improved explainability*. In 2024 34th International Telecommunication Networks and Applications Conference (ITNAC), Tiflis, 1–7.
163. Gyawali, S., Huang, J., Jiang, Y. (11-14 Ekim 2024). *Leveraging explainable AI for actionable insights in IoT intrusion detection*. In 2024 19th Annual System of Systems Engineering Conference (SoSE), Tokyo. 92–97.
164. Gummadi, A.N., Napier, J.C., Abdallah, M. (2024). XAI-IoT: An explainable AI framework for enhancing anomaly detection in IoT systems. *IEEE Access*, 12, 71024–71054.
165. Uysal, I., Kose, U. (19-21 Ekim 2024). *Analysis of network intrusion detection via explainable artificial intelligence: Applications with SHAP and LIME*. In 2024 Cyber Awareness and Research Symposium (CARS), Monaco. 1–6.
166. Ahakonye, L.A.C., Nwakanma, C.I., Lee, J.M., Kim, D.-S. (2024). Machine learning explainability for intrusion detection in the Industrial Internet of Things. *IEEE Internet of Things Magazine*, 7(3), 68–74.
167. Mahmoud, M., Youssef, Y.O., Abdel-Hamid, A. (19-22 Ekim 2024). *An explainable intelligent intrusion detection system*. In 2024 International Symposium on Networks, Computers and Communications (ISNCC), Luzern. 1–6.
168. Xu, Y., Li, C., Zhang, K., Xia, H., ve Tu, B. (20-23 Kasım 2024). *EI-XIDS: An explainable intrusion detection system based on integration framework*. In 2024 27th International Conference on Computer Supported Cooperative Work in Design (CSCWD), Brüksel. 2680–2685.
169. Nugraha, B., Jnanashree, A.V., Bauschert, T. (21-23 Kasım 2024). *An efficient explainable artificial intelligence (XAI)-based framework for a robust and explainable IDS*. In 2024 8th Cyber Security in Networking Conference (CSNet), Marsilya. 173–181.
170. Tai, N. N., (11-14 Eylül 2024). *A robust and trustworthy intrusion detection system using adversarial machine learning and XAI*. In 2024 International Conference on Advanced Technologies for Communications (ATC), Tokyo. 407–412.

171. Haripriya, C., Jagadeesh, M.P. (2024). An explainable and optimized network intrusion detection model using deep learning. *International Journal of Advanced Computer Science and Applications*, 15(1),279-296.
172. Ananya, G., Siri, N., Kodipalli, A., Rao, T. (15-19 Eylül 2024). *Explainable AI for enhanced intrusion detection systems*. In 2024 3rd International Conference for Advancement in Technology (ICONAT), Sofya. 1–6.
173. Atagün, E., Temür, G., Biroğul, S. (2024). Application of explainable artificial intelligence in intrusion detection system classification using boosting algorithms. *Mugla Journal of Science and Technology*, 10(1),1–7.
174. Komarchesqui, M., Lent, D.M.B., Da Silva Ruffo, V.G., Carvalho, L.F., Lloret, J., Proenca, M. L. (2024). *Explainable AI feature selection in generative adversarial networks system aiming to detect DDoS attacks*. In 2024 11th International Conference on Software Defined Systems (SDS), Atina. 27–34.
175. Khediri, A., Slimi, H., Yahiaoui, A., Derdour, M., Bendjenna, H., Ghenai, C. E. (8-11. Mayıs 2024). *Enhancing machine learning model interpretability in intrusion detection systems through SHAP explanations and LLM-generated descriptions*. In 2024 6th International Conference on Pattern Analysis and Intelligent Systems (PAIS), Limasol. 1–6.
176. Mia, M., Pritom, M.M.A., Islam, T., Hasan, K. (20-23 Kasım 2024). *Visually analyze SHAP plots to diagnose misclassifications in ML-based intrusion detection*. In 2024 IEEE International Conference on Data Mining Workshops (ICDMW),Lyon. 632–641.
177. İnternet: GNNexPIDS: An interpretation method for provenance-based intrusion detection based on GNNExplainer. URL-19: https://link.springer.com/chapter/10.1007/978-981-96-2417-1_13 Son Erişim Tarihi: 11.10.2025.
178. Sun, Z., Teixeira, A.M.H., Toor, S. (19-22 Kasım 2024). *GNN-IDS: Graph neural network-based intrusion detection system*. In Proceedings of the 19th International Conference on Availability, Reliability and Security (ARES '24), Rio. 1–12.
179. Kaya, K., Ak, E., Bas, S., Canberk, B., Oguducu, S.G. (7-9 Mayıs 2024). *X-CBA: Explainability aided CatBoosted Anomal-E for intrusion detection system*. In ICC 2024 – IEEE International Conference on Communication, (ICDMW), Budapeşte 2288–2293.
180. Son, N. D. H., Thi, H. T., Duy, P. T., e Pham, V. H. (2023). XFedGraph-Hunter: An interpretable federated learning framework for hunting advanced persistent threat in provenance graph. *Lecture Notes in Computer Science*, 1(3),546-561.
181. Himmelhuber, A., Dold, D., Grimm, S., Zillner, S., Runkler, T. (2022, 17-20 Kasım). *Detection, explanation and filtering of cyber attacks combining symbolic and sub-symbolic methods*. In 2022 IEEE Symposium Series on Computational Intelligence (SSCI), Kazan. 381–388.

182. Wang, X., (2025). *Enterprise threat detection with explainable AI for cloud–network convergence*. In 2025 IEEE 33rd International Conference on Network Protocols (ICNP), Lyon. 1–6.
183. Lee, S., Jung, H. R., Lee, T. J. (2025). Adaptive hybrid NIDS: Graph-based detection and domain-aware reasoning for encrypted traffic. *SSRN preprint*. 2(2), 24.
184. Osswald, M., Schönenberger, T., Cantali, G., Soussi, W., Gür, G. (25-28 Kasım 2025). *Anomaly detection in microservices architecture using graph neural networks*. In 2025 33rd Euromicro International Conference on Parallel, Distributed, and Network-Based Processing (PDP), Brüksel. 560–567.
185. İnternet: ProvX: Generating counterfactual-driven attack explanations for provenance-based detection. URL-18: <https://arxiv.org/abs/2508.06073>, Son Erişim Tarihi: 11.09.2025.
186. Qian, K., Yang, H., Li, R., Chen, W., Luo, X., Yin, L. (2024). Distributed detection of large-scale Internet of Things botnets based on graph partitioning. *Applied Sciences*, 14(4), 1615.
187. Wang, M., Zheng, K., Yang, Y., (2020). An explainable machine learning framework for intrusion detection systems. *IEEE Access*, 8, 73127–73141.
188. Bacevicius, M., Paulauskaite-Taraseviciene, A. (2023). Machine learning algorithms for raw and unbalanced intrusion detection data in a multi-class classification problem. *Applied Sciences*, 13, 7328.
189. İnternet: Zero-shot learning approach to adaptive cybersecurity using explainable AI. URL-19: <https://doi.org/10.48550/arXiv.2106.14647> Son Erişim Tarihi: 25.09.2025.
190. Sarica, A. K., Angin, P. (2020). Explainable security in SDN-based IoT networks. *Sensors*, 20(3), 7326.
191. Gaspar, D., Silva, P., Silva, C. (2024). Explainable AI for intrusion detection systems: LIME and SHAP applicability on multi-layer perceptron. *IEEE Access*, 12, 1–1.
192. Hooshmand, M.K., Huchaiah, M.D., Alzighaibi, A.R., Hashim, H., Atlam, E., Gad, I. (2024). Robust network anomaly detection using ensemble learning approach and explainable artificial intelligence (XAI). *Alexandria Engineering Journal*, 94, 120–130.
193. Ahamed Maricar, S.B., Anoop, A., Samuel, B.E., Appukuttan, A., Alsinjlawi, K. H. (2024). An improved explainable artificial intelligence for intrusion detection system. *International Journal of Intelligent Systems and Applications in Engineering*, 12(14s), 108–115.
194. Mallampati, S., Hari, S. (2024). Enhancing intrusion detection with explainable AI: A transparent approach to network security. *Cybernetics and Information Technologies*, 24, 98–117.

195. Sivamohan, S., Sridhar, S.S. (2023). An optimized model for network intrusion detection systems in Industry 4.0 using XAI-based Bi-LSTM framework. *Neural Computing and Applications*, 35, 11459–11475.
196. Marwa, K., Nickolaos, K., Nam, P., (2023). An explainable deep learning-enabled intrusion detection framework in IoT networks. *Information Sciences*, 639, 119000.
197. Lanfer, E., Sylvester, S., Aschenbruck, N., Atzmueller, M. (25.10. Aralık 2023). *Leveraging explainable AI methods towards identifying classification issues on IDS datasets*. In Proceedings of the IEEE 48th Conference on Local Computer Networks (LCN), Luzern. 1–4.
198. Siganos, M., Radoglou-Grammatikis, P., Kotsiuba, I., Markakis, E., Moscholios, I., Goudos, S., Sarigiannidis, P. (2023). *Explainable AI-based intrusion detection in the Internet of Things*. In Proceedings of the 18th International Conference on Availability, Reliability and Security (ARES '23), Dallas. 1–10.
199. Manh-Dung, N., Anis, B., Valeria, V., (23-27 Ekim 2023). *A deep learning anomaly detection framework with explainability and robustness*. In Proceedings of the International Conference on Availability, Reliability and Security (ARES). New York. 257-263.
200. Sherif, K. H., Mohamed, A.E. (2023). Anomaly detection in cyber-physical systems using explainable artificial intelligence and machine learning. *Journal of Theoretical and Applied Information Technology*, 101(8), 238-251.
201. Almuqren, L., Maashi, M. S., Alamgeer, M., (2023). Explainable artificial intelligence enabled intrusion detection technique for secure cyber-physical systems. *Applied Sciences*, 13(3081),159-183.
202. Sharma, B., Sharma, L., Lal, C. (14-16 Ekim 2023). *Anomaly-based DNN model for intrusion detection in IoT and model explanation: Explainable artificial intelligence*. In International Conference on Computing, Electronics & Wireless Communications, (ARES). New York. 156-171.
203. Rehman, A., Farrakh, A., Khan, S. (2023). Explainable AI in intrusion detection systems: Enhancing transparency and interpretability. *International Journal of Advanced Sciences and Computing*, 2(1), 7–20.
204. Youcef, D., Asma, B., Gautam, S., Jerry, C. L., Anis, Y. (2023). Interpretable intrusion detection for next generation of Internet of Things. *Computer Communications*, 203, 192–198.
205. Pande, S., Khamparia, A. (2023). Explainable deep neural network–based analysis on intrusion detection systems. *Computer Science*, 24(1), 45-83.
206. Sharma, B., Sharma, L., Lal, C., Rajasekaran, S. (2024). Explainable artificial intelligence for intrusion detection in IoT networks: A deep learning-based approach. *Expert Systems with Applications*, 238(A), 121751.

207. Sevri, M., Karacan, H. (27 Mayıs 2023). *Explainable artificial intelligence (XAI) for deep learning-based intrusion detection systems*. In 4th International Conference on Artificial Intelligence and Applied Mathematics in Engineering. (IFSS). Londra. 39-55.
208. Larriva-Novo, X., Sánchez-Zas, C., Villagrà, V. A., Marín-Lopez, A., Berrocal, J. (2023). Leveraging explainable artificial intelligence in real-time cyberattack identification: Intrusion detection system approach. *Applied Sciences*, 13, 8587.
209. Gürbüz, E., Turgut, Ö., Kök, İ. (27-29 Haziran 2023). Explainable AI-based malicious traffic detection and monitoring system in next-gen IoT healthcare. In 2023 International Conference on Smart Applications, Communications and Networking (SmartNets), 1–6.
210. İnternet: IsoEx: An explainable unsupervised approach to process event logs cyber investigation. URL-20: <https://doi.org/10.48550/arXiv.2306.09260>. Son Erişim Tarihi: 07.10.2025.
211. Chinaechetam, N., Njoku, J., Nwakanma, C., Lee, J.M., Kim, D.-S. (13-16 Ekim 2023). *SHAP-based intrusion detection framework for zero-trust IoT maritime security*. The 2nd International Conference on Maritime IT Convergence, 2, (ARES). New York. 122–125.
212. İnternet: Case-based explanation of classification models for the detection of SQL injection attacks. URL-21: https://ceur-ws.org/Vol-3438/paper_15.pdf Son Erişim Tarihi: 11.09.2025.
213. Kalutharage, C.S., Liu, X., Chrysoulas, C., Pitropakis, N., Papadopoulos, P. (2023). Explainable AI-based DDoS attack identification method for IoT networks. *Computers*, 12(3), 32.
214. Javeed, D., Gao, T., Kumar, P., Jolfaei, A. (2024). An explainable and resilient intrusion detection system for Industry 5.0. *IEEE Transactions on Consumer Electronics*, 70(1), 1342–1350.
215. Patil, A.P., Devarakonda, J., Singuru, M., Tilak, S., Jadon, S. (2023, 24-25 Eylül). *XAI for securing cyber-physical systems*. In 2023 3rd International Conference on Secure Cyber Computing and Communications (ICSCCC), Nice. 671–677.
216. Muna, R.K., Hossain, M.I., Alam, M.G.R., Hassan, M.M., Ianni, M., Fortino, G. (2023). Demystifying machine learning models of massive IoT attack detection with explainable AI for sustainable and secure future smart cities. *Internet of Things*, 24, 100919.
217. Hassan, F., Yu, J., Syed, Z.S., Ahmed, N., Reshan, M.S.A., Shaikh, A. (2023). Achieving model explainability for intrusion detection in VANETs with LIME. *PeerJ Computer Science*, 9, e1440.

218. Lanvin, M., Gimenez, P.-F., Han, Y., Majorczyk, F., Mé, L., (24-26 Kasım 2023). *Towards understanding alerts raised by unsupervised network intrusion detection systems*. In Proceedings of the 26th International Symposium on Research in Attacks, Intrusions and Defenses (RAID), Varşovs. 135–150.
219. İnternet: Analysis and detection of DDoS backscatter using NetFlow data, Hyperband-optimised deep learning and explainability techniques. URL-22: https://link.springer.com/chapter/10.1007/978-3-031-23492-7_8 Son Erişim Tarihi: 23.10.2025.
220. İnternet: RIN: Towards a semantic rigorous interpretable artificial immune system for intrusion detection. URL-23. https://link.springer.com/chapter/10.1007/978-3-031-27609-5_17 Son Erişim Tarihi 9.11.2025.
221. İnternet: XAI-IDS: Towards proposing an explainable artificial intelligence framework for enhancing network intrusion detection systems. URL-24. <https://www.mdpi.com/2076-3417/14/10/4170> Son Erişim Tarihi: 12.09.2025.
222. Sivamohan, S., Sridhar, S.S., Krishnaveni, S. (2023). TEA-EKHO-IDS: An intrusion detection system for industrial CPS with trustworthy explainable AI and enhanced krill herd optimization. *Peer-to-Peer Networking and Applications*, 16, 1993–2021.
223. El Houda, Z.A., Moudoud, H., Brik, B., Khoukhi, L. (2023, 15-19 Ekim). *Securing federated learning through blockchain and explainable AI for robust intrusion detection in IoT networks*. In IEEE INFOCOM Workshops, 1–6.
224. Lopes, R.S., Duarte, J.C., Goldschmidt, R.R. (2023). False positive identification in intrusion detection using XAI. *IEEE Latin America Transactions*, 21(6), 745–751.
225. İnternet: Bridging the gap: Advancing the transparency and trustworthiness of network intrusion detection with explainable AI. URL-25: <https://doi.org/10.21203/rs.3.rs-3263546/v1> Son Erişim Tarihi: 11.10.2025.
226. Hasan, M.M., Islam, M.U., Uddin, J. (2023). Advanced persistent threat identification with boosting and explainable AI. *SN Computer Science*, 4(1), 271.
227. Kostopoulos, N., Kalogeras, D., Pantazatos, D., Grammatikou, M., Maglaris, V. (2023). SHAP interpretations of tree and neural network DNS classifiers for analyzing DGA family characteristics. *IEEE Access*, 11, 61144–61160.
228. İnternet: XFedHunter: An explainable federated learning framework for advanced persistent threat detection in SDN. URL-25: <https://doi.org/10.48550/arXiv.2309.08485> Son Erişim Tarihi: 24.10.2025.
229. İnternet: *Classification and explanation of distributed denial-of-service attack detection using SHAP*. URL_26: <https://arxiv.org/abs/2306.17190> Son Erişim Tarihi:
230. M., I., Wang, Y., Hei, X., Song, X., Turiho, J. C., Nyesheja, E. M. (2023). XTS: A hybrid framework to detect DNS-over-HTTPS tunnels based on XGBoost and cooperative game theory. *Mathematics*, 11, 2372.

231. Rumes, Y., Senevirathna, T., Porambage, P., Liyanage, M., Ylianttila, M. (2023). *Comprehensive analysis over centralized and federated learning-based anomaly detection in networks with explainable AI (XAI)*. IEEE International Conference on Communications (ICC), Rome, Italy. 278.
232. İnternet: Image representation of domain names and CNNs for detecting data exfiltration. URL-27: <https://doi.org/10.2139/ssrn.4535811> Son Erişim Tarihi: 17.10.2025.
233. Alani, M.M., Damiani, E. (2023). XRecon: An explainable IoT reconnaissance attack detection system based on ensemble learning. *Sensors*, 23(2), 5298.
234. Bashaiwth, A., Binsalleeh, H., AsSadhan, B. (2023). An explanation of the LSTM model used for DDoS attacks classification. *Applied Sciences*, 13(4), 8820.
235. Feng, Y., Li, J., Sisodia, D., Reiher, P. (2024). On explainable and adaptable detection of distributed denial-of-service traffic. *IEEE Transactions on Dependable and Secure Computing*, 21(4), 2211–2226.
236. Miao, C., Wang, P., Wang, Z.X., Li, Z.Y. (19-21 Ekim 2022). *A trusted classification method of encrypted traffic based on attention mechanism*. Intl Conf on Cyber Science and Technology Congress (DASC/PiCom/CBDCom/CyberSciTech), Berlin. 1–8.
237. Zhu, X., Zhang, Y., Zhang, Z., Guo, D., Li, Q., Li, Z. (18-20 Haziran 2022). *Interpretability evaluation of botnet detection model based on graph neural network*. IEEE INFOCOM 2022 - IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), Hamburg. 1-6.
238. Mowla, N.I., Rosell, J., Vahidi, A. (10-13 Mart 2022). *Dynamic voting-based explainable intrusion detection system for in-vehicle network*. In 2022 International Conference on Advanced Communication Technology (ICACT), İstanbul. 406–411.
239. Alani, M.M., Miri, A. (2022). Towards an explainable universal feature set for IoT intrusion detection. *Sensors*, 22(5), 5690.
240. Szczepański, M., Pawlicki, M., Kozik, R., Choraś, M. (24-26 Ocak 2022). *Fast hybrid oracle-explainer approach to explainability using optimized search of comprehensible decision trees*. In 2022 IEEE 9th International Conference on Data Science and Advanced Analytics (DSAA), Lyon (1–10).
241. İnternet: Explainable boosting machines for network intrusion detection with features reduction. URL-28: https://link.springer.com/chapter/10.1007/978-3-031-21441-7_20#citeas Son Erişim Tarihi: 08.10.2025.
242. Huong, T. T., Bac, T. P., Ha, K. N., Hoang, N. V., Hoang, N. X., Hung, N. T. (2022). Federated learning-based explainable anomaly detection for industrial control systems. *IEEE Access*, 10, 53854–53872.

243. İnternet: XAI to explore robustness of features in adversarial training for cybersecurity. URL-29: https://doi.org/10.1007/978-3-031-16564-1_12 Son Erişim Tarihi: 14.10.2025.
244. Chawla, A., Jacob, P., Farrell, P., Aumayr, E., Fallon, S. (24-26 Ekim 2022). *Towards interpretable anomaly detection: Unsupervised deep neural network approach using feedback loop*. In NOMS 2022 – IEEE/IFIP Network Operations and Management Symposium, 1–9.
245. Carrera, F., Dentamaro, V., Galantucci, S., Iannaccone, A., Impedovo, D., Pirlo, G. (2022). Combining unsupervised approaches for near real-time network traffic anomaly detection. *Applied Sciences*, 12(3), 1759.
246. Sauka, K., Shin, G.Y., Kim, D.W., Han, M.M. (2022). Adversarial robust and explainable network intrusion detection systems based on deep learning. *Applied Sciences*, 12, 6451.
247. Eriksson, H. S., Grov, G. (2022). *Towards XAI in the SOC: A user-centric study of explainable alerts with SHAP and LIME*. In 2022 IEEE International Conference on Big Data, (ARES). New York. 2595–2600.
248. Roy, S., Li, J., Pandey, V., Bai, Y. (2022). *An explainable deep neural framework for trustworthy network intrusion detection*. In 2022 10th IEEE International Conference on Mobile Cloud Computing, Services, and Engineering (MobileCloud). Üsküp. 25–30.
249. Roshan, K., Zafar, A. (21-24 Ekim 2022). *Using kernel SHAP XAI method to optimize the network anomaly detection model*. In 2022 9th International Conference on Computing for Sustainable Global Development (INDIACom), Batum. 74–80.
250. Tabassum, S., Parvin, N., Hossain, N., Tasnim, A., Rahman, R., Hossain, M. I. (19-21 Eylül 2022). *IoT network attack detection using XAI and reliability analysis*. In 25th International Conference on Computer and Information Technology (ICCIT), Roma. 176–181.
251. Ha, D. T., Hoang, N. X., Hoang, N. V., Du, N. H., Huong, T. T., Tran, K. P. (2022). Explainable anomaly detection for industrial control system cybersecurity. *IFAC-PapersOnLine*, 55(10), 1183–1188.
252. Aslam, N., Khan, I. U., Mirza, S., AlOwayed, A., Anis, F. M., Aljuaid, R. M., ve Baageel, R. (2022). Interpretable machine learning models for malicious domain detection using explainable artificial intelligence (XAI). *Sustainability*, 14, 7375.
253. İnternet: Towards explainable meta-learning for DDoS detection. URL: <https://doi.org/10.48550/arXiv.2204.02255> Son Erişim Tarihi: 21.09.2025.
254. Ramyavarshini, P., Sriram, G. K., Rajasekaran, U., ve Malini, A. (07-09 Kasım 2022). *Explainable AI for intrusion detection systems*. In 2022 5th International Conference on Contemporary Computing and Informatics (IC3I). Helsinki. 1563–1567.

255. Patil, S., Varadarajan, V., Mazhar, S. M., Sahibzada, A., Ahmed, N., Sinha, O., Kumar, S., Shaw, K., Kotecha, K. (2022). Explainable artificial intelligence for intrusion detection system. *Electronics*, 11, 3079.
256. Younis, R., Ahmad, A., ve Abu Al-Haija, Q. (2022). Explaining intrusion detection-based convolutional neural networks using Shapley additive explanations (SHAP). *Big Data and Cognitive Computing*, 6, 126.
257. Kalutharage, C. S., Liu, X., Chrysoulas, C. (2022). Explainable AI and deep autoencoders-based security framework for IoT network attack certainty, *Lecture Notes in Computer Science*, 2(6),41-50.
258. Hutchison, J., Pham, D. S., Soh, S. T., Ling, H. C. (2022). Explainable network intrusion detection using external memory models. *Lecture Notes in Computer Science*, 1(2),220-233.
259. Yilmaz, M. N., Bardak, B. (14-16 Aralık 2022). *An explainable anomaly detection benchmark of gradient boosting algorithms for network intrusion detection systems*. In 2022 Innovations in Intelligent Systems and Applications Conference (ASYU). Londra. 1–6.
260. Barnard, P., Marchetti, N., Da Silva, L.A.D. (2022). Robust network intrusion detection through explainable artificial intelligence (XAI). *IEEE Network Letters*. 4(3),167-171.
261. Andresini, G., Appice, A., Caforio, F.P., (2022). ROULETTE: A neural attention multi-output model for explainable network intrusion detection. *Expert Systems with Applications*, 201, 117144.
262. El Houda, Z. A., Brik, B., Khoukhi, L. (2022). Why should I trust your IDS? An explainable deep learning framework for intrusion detection systems in Internet of Things networks. *IEEE Open Journal of the Communications Society*. 1(3),1164-1176.
263. Yiwen, W., Lei, X., Wanli, L. (2022). Network intrusion detection based on explainable artificial intelligence. *Wireless Personal Communications*, 15(2),842-869.
264. Zebin, T., Rezvy, S., Luo, Y. (2022). An explainable AI-based intrusion detection system for DNS over HTTPS (DoH) attacks. *IEEE Transactions on Information Forensics and Security*, 17, 2339–2349.
265. Hsupeng, B., Lee, K.-W., Wei, T.-E. (06-10 Eylül 2022). *Explainable malware detection using predefined network flow*. In Proceedings of the 24th International Conference on Advanced Communication Technology (ICACT). Tokyo. 27–33.
266. Dias, T., Oliveira, N., Sousa, N. (27-28 Mart 2022). A hybrid approach for an interpretable and explainable intrusion detection system, International Conference on Intelligent Systems Design and Applications, 1035-1045.

267. Oseni, A., Moustafa, N., Creech, G. (2022). An explainable deep learning framework for resilient intrusion detection in IoT-enabled transportation networks. *IEEE Transactions on Intelligent Transportation Systems*. 9(2),357-395.
268. Lundberg, H., Mowla, N.I., Abedin, S.F. (2022). Experimental analysis of trustworthy in-vehicle intrusion detection system using explainable artificial intelligence (XAI). *IEEE Access*. 5(3),97-121.
269. El Houda, Z. A., Brik, B., Senouci, S. M. (2022). A novel IoT-based explainable deep learning framework for intrusion detection systems. *IEEE Internet of Things Magazine*, 5, 20–23.
270. Ho, T.-N.-P., Nguyen, T.-L., Nguyen, X.-H., Le, K.-H. (20-23 Mayıs 2022). *Enhancing explainability of machine learning-based intrusion detection systems*. In Proceedings of the International Conference on Computing and Communication Technologies (RIVF), Pekin. 606–611.
271. Khan, I.A., Moustafa, N., Pi, D., (2022). A new explainable deep learning framework for cyber threat discovery in industrial IoT networks. *IEEE Internet of Things Journal*, 9(13), 11604–11613.
272. Ables, J., Kirby, T., Anderson, W., Mittal, S., Rahimi, S., Banicescu, I., Seale, M. (3-6 Aralık 2022). Creating an explainable intrusion detection system using self-organizing maps. In *2022 IEEE Symposium Series on Computational Intelligence (SSCI)*, 404–412.
273. Batchu, R. K., Seetha, H. (2022). An integrated approach explaining the detection of distributed denial of service attacks. *Computer Networks*, 8(3),109269.
274. AL-Essa, M., Andresini, G., Appice, A., Malerba, D. (14-16 Şubat 2022). *An XAI-based adversarial training approach for cyber-threat detection*. In 2022 IEEE International Conference on Dependable, Autonomic and Secure Computing, (RIVF), Pekin. 1–8.
275. Liu, H., Zhong, C., Alnusair, A., (2021). FAIXID: A framework for enhancing AI explainability of intrusion detection results using data cleaning techniques. *Journal of Network and Systems Management*, 29(4), 40.
276. İnternet: Explaining network intrusion detection system using explainable AI framework. URL- 31 <https://doi.org/10.48550/arXiv.2103.07110> Son Erişim Tarihi: 05.11.2025.
277. Mahbooba, B., Timilsina, M., Sahal, R., (2021). Explainable artificial intelligence (XAI) to enhance trust management in intrusion detection systems using decision tree model. *Complexity*, 4(1),29-43.
278. Wali, S., Khan, I. (2021). Explainable AI and random forest based reliable intrusion detection system. *TechRxiv*. 5(2),128-141.

279. Prasad, R.R., Robinson, R.R., Thomas, C., (19-21 Kasım 2021). *Evaluation of strategic decision taken by autonomous agent using explainable AI*. In Proceedings of the 4th International Conference on Security and Privacy (ISEA-ISAP), Glaskow. 1–8.
280. İnternet: Evaluating standard feature sets towards increased generalisability and explainability of ML-based network intrusion detection. URL-32: <https://arxiv.org/abs/2104.07183> Son Erişim Tarihi: 02.10.2025.
281. Tcydenova, E., Kim, T. W., Lee, C., (2021). Detection of adversarial attacks in AI-based intrusion detection systems using explainable AI. *Human-Centric Computing and Information Sciences*, 11, 1–14.
282. Wang, Y., Wang, P., Wang, Z. (26-28 Temmuz 2021). *An explainable intrusion detection system*. In Proceedings of the IEEE 23rd International Conference on High Performance Computing and Communications (HPCC/DSS/SmartCity/DependSys), Pekin. 1657–1662.
283. Kim, H., Lee, Y., Lee, E., Lee, T. (2021). Cost-effective valuable data detection based on the reliability of artificial intelligence. *IEEE Access*, 9, 108959–108974.
284. Gu, M., Zhou, B., Du, F. (09-10 Haziran 2021). *Grasp the key: Towards fast and accurate host-based intrusion detection in data centers*, International Conference on Computational Science. (HPCC/DSS/SmartCity/DependSys), Pekin. 81–194.
285. Islam, S. R., Eberle, W. (2021). Implications of combining domain knowledge in explainable artificial intelligence. *CEUR Workshop Proceedings*, 4(3), 2846.
286. Andresini, G., Pendlebury, F., Pierazzi, F. (09-12 Ağustos 2021). *Insomnia: Towards concept-drift robustness in network intrusion detection*. In Proceedings of the ACM Conference on Availability, Reliability and Security, 111–122.
287. Zolanvari, M., Yang, Z., Khan, K. (2021). Trust XAI: Model-agnostic explanations for AI with a case study on IIoT security. *IEEE Internet of Things Journal*, 5(8), 421–437.
288. Roshan, K., Zafar, A. (2021). Utilizing XAI technique to improve autoencoder-based model for computer network anomaly detection with Shapley additive explanation (SHAP). *International Journal of Computer Networks and Communications*, 2(13), 109–128.
289. Das, S., Agarwal, N., Shiva, S. (15-17 Mart 2021). *DDoS Explainer using interpretable machine learning*. In Proceedings of the IEEE 12th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON), Luzern. 1–7.
290. Wierzbicki, Ł., Majewski, M., Banaszczyk, A., (2021). Detecting anomalies and attacks in network traffic monitoring with classification methods and XAI-based explainability. *Procedia Computer Science*, 192, 2259–2268.

291. İnternet: An explainable machine learning-based network intrusion detection system for enabling generalisability in securing IoT networks. URL-33: <https://doi.org/10.48550/arXiv.2104.07183> Son Erişim Tarihi: 09.11.2025.
292. Dang, Q.-V. (2021). Improving the performance of intrusion detection systems by machine learning explainability. *International Journal of Web Information Systems*, 17(5), 537–555.
293. Szczepanski, M., Komisarek, M., Pawlicki, M., Kozik, R., Choraś, M. (27-28 Eylül 2021). *The proposition of balanced and explainable surrogate method for network intrusion detection in streamed real difficult data*. International Conference on Multi-disciplinary Trends in Artificial Intelligence, Varşova. 279–289.
294. Al-Imran, M., Rahaman, K.J., Rasel, M., Ripon, S.H. (27-29 Eylül 2021). *An analytical evaluation of a deep learning model to detect network intrusion*. International Conference on Multi-disciplinary Trends in Artificial Intelligence, Varşova. 148–160.
295. Caforio, F. P., Andresini, G., Vessio, G., Appice, A., Malerba, D. (03-05 Şubat 2021). *Leveraging Grad-CAM to improve the accuracy of network intrusion detection systems*. International Conference on Discovery Science, 410–426.
296. Wickramasinghe, C.S., Amarasinghe, K., Marino, D.L., Rieger, C., Manic, M. (2021). Explainable unsupervised machine learning for cyber-physical systems. *IEEE Access*, 9, 131824–131843.
297. Alenezi, R., Ludwig, S.A. (25-26 Kasım 2021). *Explainability of cybersecurity threats data using SHAP*. In 2021 IEEE Symposium Series on Computational Intelligence (SSCI), Londra. 1–10.
298. AL-Essa, M., Appice, A. (18-20 Mayıs 2021). *Dealing with imbalanced data in multi-class network intrusion detection systems using XGBoost*. Joint European Conference on Machine Learning and Knowledge Discovery in Databases, Zürih. 1525.
299. Botana, I.L., Eiras-Franco, C., Alonso-Betanzos, A. (2020). Regression tree-based explanation for anomaly detection algorithm. *Proceedings*, (2)54, 7.
300. Rathod, J., Joshi, C., Khochare, J., Kazi, F. (21-23 Ekim 2020). *Interpreting a black-box model used for SCADA attack detection in gas pipelines control system*. In 2020 IEEE 17th India Council International Conference (INDICON), Londra. 1–7.
301. Sejr, J. H., Zimek, A., Schneider-Kamp, P. (13-14 Haziran 2020). *Explainable detection of zero day web attacks*. 2020 3rd International Conference on Data Intelligence and Security (ICDIS), Nice. 71–78.
302. Karn, R.R., Kudva, P., Huang, H., Suneja, S., Elfadel, I. M. (2020). Cryptomining detection in container clouds using system calls and explainable machine learning. *IEEE Transactions on Parallel and Distributed Systems*, 32(3), 674–691.

303. Reyes, A.A., Vaca, F.D., Castro Aguayo, G.A. (2020). A machine learning based two-stage Wi-Fi network intrusion detection system. *Electronics*, 9(10), 1689.
304. Szczepanski, M., Choraś, M., Pawlicki, M. (2020). *Achieving explainability of intrusion detection system by hybrid oracle-explainer approach*. Proceedings of the International Joint Conference on Neural Networks (IJCNN). Lyon. 18-25.
305. İnternet: Domain knowledge aided explainable artificial intelligence for intrusion detection and response. URL-34: <https://ceur-ws.org/Vol-2600/paper1.pdf> Son Erişim Tarihi: 29.10.2025.
306. İnternet: Explaining anomalies detected by autoencoders using SHAP. URL-35: <https://doi.org/10.48550/arXiv.1903.02407> Son Erişim Tarihi: 11.09.2025.
307. Ahn, S., Kim, J., Young, P. S., Cho, S. (2020). Explaining deep learning-based traffic classification using a genetic algorithm. *IEEE Access*, 9(3),108959–108974.
308. Wu, C., Qian, A., Dong, X., Zhang, Y. (10-12 Ocak 2020). *Feature-oriented design of visual analytics system for interpretable deep learning based intrusion detection*. In International Conference on Autonomous Decision-Making Systems, Pekin. 73–80.
309. Dang, Q.V. (19-21 Haziran 2020). *Understanding the decision of machine learning based intrusion detection systems*. International Conference on Future Data and Security Engineering, 379-396.
310. Guerra-Manzanares, A., Nömm, S., Bahsi, H. (11-13 Kasım 2019). *Towards the integration of a post-hoc interpretation step into the machine learning workflow for IoT botnet detection*. 2019 18th IEEE International Conference on Machine Learning and Applications (ICMLA), Londra. 1162–1169.
311. Morichetta, A., Casas, P., Mellia, M. (18-20. Ekim 2019). *Dataset: lora underground farm sensor network*. DATA'19: Proceedings of the 2nd Workshop on Data Acquisition To Analysis, 26-28.
312. Li, H., Wei, F., Hu, H. (20-22 Kasım 2019). *Source-free machine-checked validation of native code in coq*. In Proceedings of the ACM International Workshop on Security in Software Defined Networks & Network Function Virtualization, 13–16.
313. Amarasinghe, K., Manic, M. (08-10 Aralık 2018). *Improving user trust on deep neural networks based intrusion detection systems*. In 2018 44th Annual Conference of the IEEE Industrial Electronics Society (IECON), 3262–3268.
314. Marino, D. L., Wickramasinghe, C. S., Manic, M. (02-05 Mart 2018). *An adversarial approach for explainable AI in intrusion detection systems*. In 2018 44th Annual Conference of the IEEE Industrial Electronics Society (IECON), Helsinki. 3237–3243.
315. Amarasinghe, K., Kenney, K., Manic, M. (10-11 Nisan 2018). *Toward explainable deep neural network based anomaly detection*. In Proceedings of the 11th International Conference on Human System Interaction (HSI), Londra. 311–317.

316. Le, T.T.H., Kim, H., Kang, H., Kim, H. (2022). Classification and explanation for intrusion detection system based on ensemble trees and SHAP method. *Sensors*, 22(4), 1154.
317. İnternet: Gelişmişliğin vazgeçilmez unsuru: Ulusal bilgi politikası. URL-36: <https://genckutuphaneciler.wordpress.com/wp-content/uploads/2014/12/bilgi-politikasc4b1-21.pdf> Son Erişim Tarihi: 07.10.2025.
318. Naibitt, J. (1982). The role of think tanks in megatrends analysis and future research, *Open Journal of Political Science*, 13(4), 219.
319. Baykara, M. (2016). *Bilişim Sistemleri İçin Saldırı Tespit ve Engelleme Yaklaşımlarının Tasarımı ve Gerçekleştirilmesi*. Doktora Tezi, Fırat Üniversitesi Fen Bilimleri Enstitüsü. Elazığ. 109.
320. Baykara, M., Daş, R., Karadoğan, İ. (24-26 Mayıs 2013). *Bilgi güvenliği sistemlerinde kullanılan araçların incelenmesi*. In 1st International Symposium on Digital Forensics and Security, Elazığ, 231–239.
321. Güven, E.N. (2007). *Zeki Saldırı Tespit Sistemlerinin İncelenmesi, Tasarımı ve Gerçekleştirilmesi*. Yüksek Lisans Tezi, Gazi Üniversitesi Fen Bilimleri Enstitüsü. Ankara. 163.
322. Sağiroğlu, Ş., Yolaçan, E.N., Yavanoğlu, U. (2011). Zeki saldırı tespit sistemi tasarımı ve gerçekleştirilmesi. *Gazi Üniversitesi Mühendislik Mimarlık Fakültesi Dergisi*, 26 (2), 325–340.
323. Anderson, J.P. (1980). Computer security threat monitoring and surveillance (Technical report). *Fort Washington*, Washington, 56.
324. Denning, D.E. (1987). An intrusion-detection model. *IEEE Transactions on Software Engineering*, 13(2), 222–232.
325. Buczak, A. L., Guven, E. (2016). A survey of data mining and machine learning methods for cyber security intrusion detection. *IEEE Communications Surveys & Tutorials*, 18(2), 1153–1176.
326. Sundaram, A. (1996). An introduction to intrusion detection. *Crossroads: The ACM Student Magazine*, 2(4), 3–7.
327. İnternet: Understanding intrusion detection systems. URL-37: <https://www.sans.org/white-papers/337> Son Erişim Tarihi: 13.10.2025.
328. İnternet: Saldırı tespit sistemlerinde derin analiz. URL-38: <https://blog.crypttech.com/2018/11/saldr-tespit-sistemlerinde-derin.html> Son Erişim Tarihi: 13.10.2025.
329. Anwar, S., Mohamad Zain, J., Zolkipli, M. F., Inayat, Z., Khan, S., Anthony, B., Chang, V. (2017). From intrusion detection to an intrusion response system: Fundamentals, requirements, and future directions. *Algorithms*, 10(39), 371–389.

330. Eshakagdy, M., Matter, A., Hussin, S., Hassan, D., Elsaid, S. (2022). A comparative study of intrusion detection systems applied to NSL-KDD dataset. *Egyptian International Journal of Engineering Sciences and Technology*. 19(1), 348-351.
331. Vasilomanolakis, E., Karuppayah, S., Mühlhäuser, M., Fischer, M. (2015). Taxonomy and survey of collaborative intrusion detection. *ACM Computing Surveys*, 47(2),19-30.
332. Spirakis, P., Katsikas, S., Gritzalis, D., Allegre, F., Darzentas, J., Gigante, C., Karagiannis, D., Kess, P., Putkonen, H., Spyrou, T. (1994). SECURENET: A network-oriented intelligent intrusion prevention and detection system. *Network Security Journal*, 1(1),153-167.
333. İnternet: An İntroduction to İntrusion-Detection Systems. URL-39: https://www.researchgate.net/publication/228589845_An_Introduction_to_Intrusion-Detection_Systems Son Erişim Tarihi: 13.10.2025.
334. Axelsson, S. (2000). *Intrusion detection systems: A survey and taxonomy*, (Technical Report 99-15). Chalmers University of Technology.
335. Tenekeci, M. E. (2008). *Java programlama çevrelerinde süreç gözetleme ve bir saldırı tespit sisteminin gerçekleştirilmesi*. Yüksek Lisans Tezi, Karadeniz Teknik Üniversitesi Fen Bilimleri Enstitüsü. Trabzon. 89.
336. Denning, D. E. (1986). An intrusion detection model. *IEEE Transactions on Software Engineering*, 13(2), 222-232.
337. Jones, A.K. (2000). *Computer system intrusion detection: A survey* (Technical report). University of Virginia, Computer Science Department.
338. Sancak, S. (2008). *Saldırı Tespit Sistemi Tekniklerinin Karşılaştırılması*. Yüksek Lisans Tezi, Gebze Üniversitesi Yüksek Teknoloji Enstitüsü. Kocaeli. 121.
339. Stallings, W. (1998). *Cryptography and network security: (4. Edition)*. New York: Prentice Hall. 423.
340. Wu, N., Zhang, J. (18-20 Haziran 2003). *Factor analysis based anomaly detection*. IEEE Systems, Man and Cybernetics Society Information Assurance Workshop, 375–389.
341. Jyothsna, V., Prasad, K.M. (2019). Anomaly-based intrusion detection system. *International Journal of Computer Applications*, 28(7), 26–35.
342. İnternet: A survey of intrusion detection analysis methods. URL-40: <https://cseweb.ucsd.edu/classes/sp99/cse221/projects/IDS.pdf> Son Erişim Tarihi: 16.11.2025.
343. Yıldız, E. (2010). *Veri madenciliği teknikleriyle saldırı tespiti ve bir uygulama*. Yüksek lisans tezi, Gazi Üniversitesi Bilişim Enstitüsü. Ankara. 140.

344. Kaynar, O., Arslan, H., Görmez, Y., Işık, Y.E. (2018). Makine öğrenmesi ve öznelilik seçim yöntemleriyle saldırı tespiti. *Bilişim Teknolojileri Dergisi*, 11(2), 15-29.
345. İnternet: Statistical-Based Intrusion Detection, URL-41: <https://community.broadcom.com/symantecenterprise/communities/community-home/librarydocuments/viewdocument?DocumentKey=326c2d89-b6f7-4f6e-bf8f-f5fc1d435f8a&CommunityKey=1ecf5f55-9545-44d6-b0f4-4e4a7f5f5e68&tab=librarydocuments>, Son Erişim Tarihi: 24.11.2025.
346. Bhattacharyya, D. K., Bhuyan, M. H., ve Kalita, J. K. (2017). *Network traffic anomaly detection and prevention: Concepts, techniques, and tools*. (1. Edition). New York: Springer. 263.
347. Prayote, A. (2007). Knowledge-based anomaly detection, *Engineering Applications of Artificial Intelligence*, 136(2), 108996.
348. İnternet: Detecting unusual program behaviour using the statistical component of the next-generation intrusion detection expert system (NIDES), URL-42: <https://www.semanticscholar.org/paper/Detecting-Unusual-Program-Behavior-Using-the-of-the-Anderson-Lunt/dbd212bd2bd56d11955b7aa3b947e59d32c1164d> Son Erişim Tarihi: 17.11.2025.
349. Barbará, D., Jajodia, S. (2012). *Applications of data mining in computer security*, (1. Edition). New York: Springer. 252.
350. İnternet: Intelligent systems: Concepts, methodologies, tools, and applications. URL-43: <https://dl.acm.org/doi/10.5555/3279014> Son Erişim Tarihi: 19.10.2025.
351. İnternet: A'dan Z'ye Anomaly Detection, URL-44: <https://medium.com/@oguzkircicek/adan-z-ye-anomaly-detection-62b54f6bdd63>, Son Erişim Tarihi: 24.11.2025.
352. Caulkins, L. T. C. B. D., Lee, J., Wang, M. (2007). *A dynamic data mining technique for intrusion detection systems*. In Proceedings of the 43rd Annual Southeast Regional Conference (ACMSE 43), 148–153.
353. Tsai, C.-F., (2009). Intrusion detection by machine learning: A review. *Expert Systems with Applications*, 36(10), 11994–12000.
354. Kaya, Ç., Yıldız, O. (2014). Makine öğrenmesi teknikleriyle saldırı tespiti: Karşılaştırmalı analiz. *Marmara Fen Bilimleri Dergisi*, 3, 89–104.
355. Dua, S., Du, X. (2011). *Data mining and machine learning in cybersecurity*. (2. Edition). New York: CRC Press. 193.
356. Çavuşoğlu, Ü., Kaçar, S. (2019). Anormal trafik tespiti için veri madenciliği algoritmalarının performans analizi. *Academic Platform Journal of Engineering and Science*, 7(2), 205–216.

357. Chandola, V., Banerjee, A., Kumar, V. (2009). Anomaly detection: A survey. *ACM Computing Surveys*, 18(3),1–72.
358. Elmas, Ç. (2007). *Yapay zeka uygulamaları*. (2. Baskı). Ankara: Seçkin Yayınları. 258.
359. İnternet: Yapay sinir ağları, URL-45: https://tr.wikipedia.org/wiki/Yapay_sinir_a%C4%9Flar%C4%B1, Son Erişim Tarihi: 24.11.2025.
360. Bhuyan, M.H., Bhattacharyya, D.K., Kalita, J.K. (2014). Network anomaly detection: Methods, systems and tools. *IEEE Communications Surveys & Tutorials*, 16(1), 303–336.
361. Wagh, S. K., Pachghare, V. K., Kolhe, S. R. (2013). Survey on intrusion detection system using machine learning techniques. *International Journal of Computer Applications*, 78(16), 30–37.
362. Gilmore, C., Haydaman, J. (27-28 Kasım 2016). *Anomaly detection and machine learning methods for network intrusion detection: An industrially focused literature review*. In International Conference on Security and Management. Londra. 45-49.
363. Pedro, G.-T. (2009). Anomaly-based network intrusion detection: Techniques, systems and challenges. *Computers & Security*, 28(1), 18–28.
364. Patcha, A., Park, J.-M. (2007). An overview of anomaly detection techniques: Existing solutions and latest technological trends. *Computer Networks*, 51(12), 3448–3470.
365. İnternet : Saldırı tespit sistemleri, URL-46: https://tr.wikipedia.org/wiki/Sald%C4%B1r%C4%B1_tespit_sistemleri, Son Erişim Tarihi: 24.11.2025.
366. Yavanoglu, O., Aydos, M. (03-04 Eylül 2017). *A review on cyber security datasets for machine learning algorithms*. In 2017 IEEE International Conference on Big Data. Londra. 2186–2193.
367. Xin, Y., Kong, L., Liu, Z., (2018). Machine learning and deep learning methods for cybersecurity. *IEEE Access*, 6, 35365–35381.
368. İnternet: NSL-KDD Data Set. URL-47: <https://www.unb.ca/cic/datasets/nsl.html> Son Erişim Tarihi: 24.11.2025.
369. Moustafa, N., Slay, J. (13-14 Aralık 2015). *UNSW-NB15: A comprehensive data set for network intrusion detection systems*. In Military Communications and Information Systems Conference (MilCIS). Brüksel. 214-223.
370. İnternet: CICIDS2017 Dataset. URL-48: <https://www.unb.ca/cic/datasets/ids-2017.html>. Son Erişim Tarihi: 24.11.2025.

371. Creech, G., Hu, J. (23-25 Haziran 2013). *Generation of a new IDS test dataset: Time to retire the KDD collection*. In 2013 IEEE Wireless Communications and Networking Conference, Varşova. 4487–4492.
372. Alsaedi, A., Moustafa, N., Tari, Z., Mahmood, A., Anwar, A. (2020). TON_IoT telemetry dataset: A new generation dataset of IoT and IIoT for data-driven intrusion detection systems. *IEEE Access*, 8(1), 165130–165150.
373. İnternet: IoT Dataset for Intrusion Detection 2022. URL-49: <https://www.unb.ca/cic/datasets/iotdataset-2022.html> Son Erişim Tarihi: 24.11.2025.
374. İnternet: CSE-CIC-IDS2018 Dataset. URL-50: <https://www.unb.ca/cic/datasets/ids-2018.html> Son Erişim Tarihi: 20.11.2025.
375. Faramondi, L., Flammini, F., Guarino, S., Setola, R. (2021). A hardware-in-the-loop water distribution testbed dataset for cyber-physical security testing. *IEEE Access*, 9(3), 122385–122396.
376. Sharafaldin, I., Lashkari, A. H., Hakak, S., Ghorbani, A. A. (2019). *Developing realistic distributed denial of service (DDoS) attack dataset and taxonomy*. In IEEE 53rd International Carnahan Conference on Security Technology (ICCST), Pekin. 187-190.
377. İnternet: IoT-23: A labeled dataset with malicious and benign IoT network traffic. URL-51: <https://doi.org/10.5281/zenodo.4743746> Son Erişim Tarihi: 20.11.2025.
378. İnternet: Malicious Android Apps 2020 (MalDroid). URL-52: <https://www.unb.ca/cic/datasets/maldroid-2020.html> Son Erişim Tarihi: 20.11.2025.
379. İnternet: KDD Cup 1999 Data. URL-53: <https://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html> Son Erişim Tarihi: 20.11.2025.
380. Koroniotis, N., Moustafa, N., Sitnikova, E., Turnbull, B. (2019). Towards the development of realistic botnet dataset in the Internet of Things for network forensic analytics: Bot-IoT dataset. *Future Generation Computer Systems*, 100(2), 779–796.
381. Gabriel, M. F., (2018). UGR'16: A new dataset for the evaluation of cyclostationarity-based network IDSs. *Computers & Security*, 73(1), 411–424.
382. Song, J., Takakura, H., Okabe, Y., Eto, M., Inoue, D., Nakao, K. (17-19 Mayıs 2011). *Statistical analysis of honeypot data and building of Kyoto 2006+ dataset for NIDS evaluation*. In Proceedings of the Workshop on Building Analysis Datasets and Gathering Experience Returns for Security, 29–36.
383. Mamun, M.S.I., Rathore, M.A., Lashkari, A.H., Stakhanova, N., Ghorbani, A.A. (08-09 Kasım 2016). *Detecting malicious URLs using lexical analysis*. International Conference on Network and System Security, 467–482.

384. Shiravi, A., Shiravi, H., Tavallaee, M., Ghorbani, A. A. (2012). Toward developing a systematic approach to generate benchmark datasets for intrusion detection. *Computers & Security*, 31(3), 357–374.
385. Gil, G.D., Lashkari, A.H., Mamun, M., Ghorbani, A.A. (25-26 Eylül 2016). *Characterization of encrypted and VPN traffic using time-related features*. In Proceedings of the 2nd International Conference on Information Systems Security and Privacy (ICISSP 2016), Londra. 407–414.
386. Kolas, C., Kambourakis, G., Stavrou, A., Gritzalis, S. (2016). Intrusion detection in 802.11 networks: Empirical evaluation of threats and a public dataset. *IEEE Communications Surveys & Tutorials*, 18(1), 184–208.
387. Ring, M., Wunderlich, S., Grödl, D., Landes, D., Hotho, A. (2017). Creation of flow-based data sets for intrusion detection. *Journal of Information Warfare*, 16(3), 40–53.
388. Beigi, E.B., Jazi, H.H., Stakhanova, N., Ghorbani, A.A. (03-05 Aralık 2014). *Towards effective feature selection in machine learning-based botnet detection approaches*. In IEEE Conference on Communications and Network Security, Moskova. 247–255.
389. Sangster, B., O'Connor, T., Cook, T., Fanelli, R., Dean, E., Morrell, C., Conti, G. J. (24-26 Kasım 2009). *Toward instrumenting network warfare competitions to generate labeled datasets*. In Proceedings of the Workshop on Cyber Security Experimentation and Test (CSET), Glaskow. 258-261
390. Deng, L. (2014). A tutorial survey of architectures, algorithms, and applications for deep learning. *APSIPA Transactions on Signal and Information Processing*, 3(19, 1–29.
391. Yilin, Y., Min, C., Saad, S., (2017). Efficient imbalanced multimedia concept retrieval by deep learning on spark clusters. *International Journal of Multimedia Data Engineering and Management*, 8(1), 1–20.
392. Yilin, Y., Min, C., Mei-Ling, S., Shu-Ching, C. (2015). *Deep learning for imbalanced multimedia data classification*. In IEEE International Symposium on Multimedia, 483–488.
393. Turing, A.M. (1950). Computing machinery and intelligence. *Mind*, 59,(236), 433–460.
394. Russell, S.J., Norvig, P. (2010). *Artificial intelligence: A modern approach*. (4. edition). California: Pearson. 341.
395. Kleinberg, J., Lakkaraju, H., Leskovec, J. (2018). Human decisions and machine predictions. *The Quarterly Journal of Economics*, 133(1), 237–293.
396. Lakkaraju, H., Kamar, E., Caruana, R., (2019). *Faithful and customizable explanations of black box models*. In Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society, Birmingham. 131–138.

397. Rudin, C. (2019). Please stop explaining black box models for high stakes decisions. *Nature Machine Intelligence*, 1(4), 206–215.
398. Lipton, C. (2018). The mythos of model interpretability: In machine learning, the concept of interpretability is both important and slippery. *Communications of the ACM*, 61(10), 36–43.
399. İnternet: Towards a rigorous science of interpretable machine learning. URL-54: <https://doi.org/10.48550/arXiv.1702.08608> Son Erişim Tarihi: 20.11.2025.
400. Gilpin, L.H., Bau, D., Yuan, B. (12-14 Kasım 2018). *Explaining explanations: An overview of interpretability of machine learning*. In IEEE International Conference on Data Science and Advanced Analytics (DSAA), Oxford. 80–89.
401. Adadi, A., Berrada, M. (2018). Peeking inside the black-box: A survey on Explainable Artificial Intelligence (XAI). *IEEE Access*, 6(2), 52138–52160.
402. Arrieta, A.B., Díaz-Rodríguez, N., Ser, J. D., (2020). Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI. *Information Fusion*, 58(2), 82–115.
403. Tjoa, E., Guan, C. (2021). A survey on Explainable Artificial Intelligence (XAI): Toward medical XAI. *IEEE Transactions on Neural Networks and Learning Systems*, 32(11), 4793–4813.
404. Linardatos, P., Papastefanopoulos, V., Kotsiantis, S. (2021). Explainable AI: A review of machine learning interpretability methods. *Entropy*, 23(1), 18.
405. Miller, T. (2019). Explanation in artificial intelligence: Insights from the social sciences. *Artificial Intelligence*, 267, 1–38.
406. İnternet: Four principles of explainable artificial intelligence URL-55: <https://nvlpubs.nist.gov/nistpubs/ir/2021/NIST.IR.8312.pdf> Son Erişim Tarihi: 20.11.2025.
407. İnternet: Explainability for fair machine learning. URL-56: <https://doi.org/10.48550/arXiv.2010.07389> Son Erişim Tarihi: 20.11.2025.
408. İnternet: Bias in data-driven AI systems: An introductory survey. URL-57: <https://arxiv.org/abs/2001.09762> Son Erişim Tarihi: 20.11.2025.
409. İnternet: Assessing and addressing algorithmic bias—but before we get there. URL-58: <https://arxiv.org/abs/1809.03332> Son Erişim Tarihi: 20.11.2025.
410. Kamath, U., Liu, J. (2021). *Explainable artificial intelligence: an introduction to interpretable machine learning*. (1. Baskı). New York: Springer. 330.
411. Rai, A. (2019). Explainable AI: From black box to glass box. *Journal of the Academy of Marketing Science*, 48, 137–141.

412. Wolf, C., ve Ringland, K. (08-09 Aralık 2020). *Designing accessible, explainable AI (XAI) experiences*. ACM SIGACCESS Accessibility and Computing, 1–6.
413. Vigano, L., Magazzeni, D. (13-15 Nisan 2020). *Explainable security*. In Proceedings of the IEEE European Symposium on Security and Privacy Workshops (EuroS&PW), Napoli. 293–300.
414. Carvalho, D.V., Pereira, E.M., Cardoso, J.S. (2019). Machine learning interpretability: A survey on methods and metrics. *Electronics*, 8(8), 832.
415. Ivanovs, M., Kadikis, R., Ozols, K. (2021). Perturbation-based methods for explaining deep neural networks: A survey. *Pattern Recognition Letters*, 150, 228–234.
416. İnternet: Opportunities and challenges in explainable artificial intelligence (XAI): A survey. URL-59: <https://arxiv.org/abs/2006.11371> Son Erişim Tarihi: 20.11.2025.
417. Friedman, J.H. (2001). Greedy function approximation: A gradient boosting machine. *Annals of Statistics*, 29(5), 1189–1232.
418. Hariharan, S., Velicheti, A., Anagha, A. S., Thomas, C., Balakrishnan, N. (20-21 Haziran 2021). *Explainable artificial intelligence in cybersecurity: A brief review*. In Proceedings of the International Conference on Security and Privacy (ISEA-ISAP), Lizbon. 1–12.
419. İnternet: Explainable AI concepts techniques and challenges, URL-60: <https://www2.deloitte.com/uk/en/pages/deloitte-analytics/articles/a-review-of-explainable-ai-concepts-techniques-and-challenges.html> Son Erişim Tarihi: 24.11.2025.
420. İnternet: Ceteris-paribus profiles, URL-61: <https://ema.drwhy.ai/ceterisParibus.html> Son Erişim Tarihi: 24.11.2025.
421. Staniak, M., Biecek, P. (2018). Explanations of model predictions with live and breakDown packages. *The R Journal*, 10, 395.
422. Collaris, D., van Wijk, J.J. (23-25 Kasım 2020). *Machine learning interpretability through contribution-value plots*. In Proceedings of the 13th International Symposium on Visual Information Communication and Interaction (VINCI '20), Amsterdam. 139-143.
423. Yuan, H., Tang, J., Hu, X., Ji, S. (19-22 Ekim 2020). *XGNN: Towards model-level explanations of graph neural networks*. In Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (KDD '20), 430–438.
424. Lundberg, S. M., Lee, S.-I. (2017). A unified approach to interpreting model predictions. *Advances in Neural Information Processing Systems*, 30(3), 142-157.
425. İnternet: Consistent individualized feature attribution for tree ensembles. URL-62: <https://arxiv.org/abs/1802.03888> Son Erişim Tarihi: 24.11.2025.

426. Ribeiro, M.T., Singh, S., Guestrin, C. (28-29 Mayıs 2016). *Why should I trust you? ” Explaining the predictions of any classifier*. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 1135–1144.
427. Ribeiro, M.T., Singh, S., Guestrin, C. (07-08 Temmuz 2018). *Anchors: High-precision model-agnostic explanations*. In Proceedings of the AAAI Conference on Artificial Intelligence, 493-516.
428. İnternet: Interpretable machine learning: URL-63: <https://christophm.github.io/interpretable-ml-book/> Son Erişim Tarihi: 24.11.2025.
429. Breiman, L. (2001). Random forests. *Machine Learning*, 45(1), 5–32.
430. İnternet: Ideas on interpreting machine learning. URL-64: <https://www.oreilly.com/radar/ideas-on-interpreting-machine-learning/> Son Erişim Tarihi: 24.11.2025.
431. Confalonieri, R., (10-11 Mayıs 2019). *TREPAN Reloaded: A knowledge-driven approach to explaining black-box models*. In Proceedings of the European Conference on Artificial Intelligence, 347-351.
432. Zilke, J. R., Mencía, E. L., Janssen, F. (13-15 Kasım 2016). *DeepRED: Rule extraction from deep neural networks*. In Lecture Notes in Artificial Intelligence, 457–473.
433. İnternet: Local rule-based explanations of black box decision systems. URL-65: <https://arxiv.org/abs/1805.10820> Son Erişim Tarihi: 24.11.2025.
434. Choi, E., Bahadori, M.T., Sun, J., Kulas, J., Schuetz, A., Stewart, W. (2016). RETAIN: An interpretable predictive model for healthcare using reverse time attention mechanism. *Advances in Neural Information Processing Systems*, 29, 3504–3512.
435. İnternet: GraphLIME: Local interpretable model explanations for graph neural networks. URL-66: <https://arxiv.org/abs/2001.06216> Son Erişim Tarihi: 24.11.2025.
436. İnternet: InterpretML: A unified framework for machine learning interpretability. URL-67: <https://arxiv.org/abs/1909.09223> Son Erişim Tarihi: 24.11.2025.
437. Sharma, S., Henderson, J., Ghosh, J. (18-20 Haziran 2020). *CERTIFAI: A common framework to provide explanations and analyse the fairness and robustness of black-box models*. In Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society (AIES '20), Bournemouth. 166–172.
438. Van Eck, D., McAdams, D. A., Vermaas, P. E. (03-05 Nisan 2007). *Functional decomposition in engineering: A survey*. In International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, Rotherdam. 227–236.

439. Guo, W., Mu, D., Xu, J., Su, P., Wang, G., Xing, X. (09-11 Nisan 2018). *LEMNA: Explaining deep learning-based security applications*. In Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security, 364–379.
440. Zolanvari, M., Yang, Z., Khan, K., Jain, R., Meskin, N. (2021). TRUST XAI: Model-agnostic explanations for AI with a case study on IIoT security. *IEEE Internet of Things Journal*, 14(2), 187-201.
441. Främling, K. (13-14 Mart 2022). *Contextual importance and utility: A theoretical foundation*, Australasian Joint Conference on Artificial Intelligence, 13151.
442. Chen, J., Song, L., Wainwright, M. J., Jordan, M.I. (18-19 Eylül 2018). *Learning to explain: An information-theoretic perspective on model interpretation*. In Proceedings of the 35th International Conference on Machine Learning (ICML 2018), Pekin. 882–89).
443. İnternet: Streaming weak submodularity: Interpreting neural networks on the fly. URL-68: <https://arxiv.org/abs/1703.02647> Son Erişim Tarihi: 24.11.2025.
444. Lei, J., G'Sell, M., Rinaldo, A., Tibshirani, R. J., Wasserman, L. (2018). Distribution-free predictive inference for regression. *Journal of the American Statistical Association*, 113(523), 1094–1111.
445. Dhurandhar, A., Chen, P.-Y., Luss, R., Tu, C.-C., Ting, P., Shanmugam, K., Das, P. (11-12 Kasım 2018). *Explanations based on the missing: Towards contrastive explanations with pertinent negatives*. In Advances in Neural Information Processing Systems, 592–603.
446. İnternet: Explainable artificial intelligence (XAI) for increasing user trust in deep reinforcement learning driven autonomous systems. URL-69: <https://arxiv.org/abs/2106.03775> Son Erişim Tarihi: 24.11.2025.
447. İnternet: Model-agnostic supervised local explanations. URL-70: <https://arxiv.org/abs/1807.02910> Son Erişim Tarihi: 24.11.2025.
448. İnternet: Examples are not enough, learn to criticize! Criticism for interpretability, URL-71: https://papers.nips.cc/paper_files/paper/2016/hash/5680522b8e2bb01943234bce7bf84534-Abstract.html Son Erişim Tarihi: 24.11.2025.
449. Mothilal, R. K., Sharma, A., Tan, C. (29-30 Eylül 2020). *Explaining machine learning classifiers through diverse counterfactual explanations*. In Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency, Berlin. 607–617.
450. Gurumoorthy, K., Dhurandhar, A., Cecchi, G., Aggarwal, C. (26-27 Kasım 2019). *Efficient data representation by selecting prototypes with importance weights*. In 2019 IEEE International Conference on Data Mining (ICDM), Glaskow. 260–269.
451. Zeiler, M. D., Fergus, R. (25-27 Ekim 2014). Visualizing and understanding convolutional networks. In *European Conference on Computer Vision*. 147-150.

452. Petsiuk, V., Das, A., Saenko, K. (23-26 Ağustos 2018). *RISE: Randomized input sampling for explanation of black-box models*. In Proceedings of the British Machine Vision Conference (BMVC 2018), Hamburg. 151.
453. İnternet: Visualizing deep neural network decisions: Prediction difference analysis. URL-72: <https://arxiv.org/abs/1702.04595> Son Erişim Tarihi: 24.11.2025.
454. Fong, R.C., Vedaldi, A. (11-13 Ocak 2017). *Interpretable explanations of black boxes by meaningful perturbation*. In Proceedings of the IEEE International Conference on Computer Vision, Roma. 3429–3437.
455. İnternet: Axiomatic attribution for deep networks. URL-73: <https://arxiv.org/abs/1703.01365> Son Erişim Tarihi: 24.11.2025.
456. İnternet: Deep inside convolutional networks: Visualising image classification models and saliency maps. URL-74: <https://arxiv.org/abs/1312.6034> Son Erişim Tarihi: 24.11.2025.
457. Shrikumar, A., Greenside, P., Kundaje, A. (20-22 Kasım 2017). *Learning important features through propagating activation differences*. In Proceedings of the 34th International Conference on Machine Learning, Lyon. 3145–3153.
458. Zeiler, M. D., Fergus, R. (2014). *Computer vision-ECCV 2014*, European Conference on Computer Vision, 818-833.
459. İnternet: Striving for simplicity: The all convolutional net. URL-75: <https://arxiv.org/abs/1412.6806> Son Erişim Tarihi: 24.11.2025.
460. Zhou, B., Khosla, A., Lapedriza, A., Oliva, A., Torralba, A. (18-20 Mayıs 2016). *Learning deep features for discriminative localization*. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Tokyo. 2921–2929.
461. Selvaraju, R. R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., Batra, D. (09-11 Haziran 2017). *Grad-CAM: Visual explanations from deep networks via gradient-based localization*. In Proceedings of the IEEE International Conference on Computer Vision, Londra. 618–626.
462. Chattopadhyay, A., Sarkar, A., Howlader, P., Balasubramanian, V.N. (12-13 Haziran 2018). *Grad-CAM++: Generalized gradient-based visual explanations for deep convolutional networks*. In Proceedings of the IEEE Winter Conference on Applications of Computer Vision (WACV), Liverpool. 839–847.
463. Morbidelli, P., Carrera, D., Rossi, B., Fragneto, P., Boracchi, G. (2020). *Augmented Grad-CAM: Heat-maps super resolution through augmentation*. In ICASSP 2020 IEEE International Conference on Acoustics, Speech and Signal Processing, Londra. 4067–4071.
464. Bany Muhammad, M., Yeasin, M. (2021). Eigen-CAM: Visual explanations for deep convolutional neural networks. *SN Computer Science*, 2(1), 47.

465. Patro, B., Lunayach, M., Patel, S., Namboodiri, V. (08-10 Mart 2019). *U-CAM: Visual explanation using uncertainty based class activation maps*. In 2019 IEEE/CVF International Conference on Computer Vision (ICCV), Varşova. 7443–7452.
466. İnternet: SmoothGrad: Removing noise by adding noise. URL-76: <https://arxiv.org/abs/1706.03825> Son Erişim Tarihi: 24.11.2025.
467. İnternet: Smooth Grad-CAM++: An enhanced inference-level visualization technique for deep convolutional neural network models. URL-77: <https://arxiv.org/abs/1908.01224> Son Erişim Tarihi: 24.11.2025.
468. Erhan, D., Bengio, Y., Courville, A., Vincent, P. (2009). *Visualizing higher-layer features of a deep network*. Université de Montréal, Technical Report. 1341.
469. İnternet: On pixel-wise explanations for nonlinear classifier decisions by layer-wise relevance propagation. URL-78: <https://pmc.ncbi.nlm.nih.gov/articles/PMC4498753/> Son Erişim Tarihi: 24.11.2025.
470. Montavon, G., Lapuschkin, S., Binder, A., Samek, W., Müller, K.-R. (2017). Explaining nonlinear classification decisions with deep Taylor decomposition. *Pattern Recognition*, 65, 211–222.
471. Kim, B., (2018). *Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (TCAV)*. In International Conference on Machine Learning, 2668–2677.
472. Kumar, D., Wong, A., Taylor, G.W. (21-23 Ekim 2017). *Explaining the unexplained: A class-enhanced attentive response (CLEAR) approach to understanding deep neural networks*. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, Glaskow. 36–44.
473. Hastie, T., Tibshirani, R., Friedman, J. (2009). *The elements of statistical learning*, (2. Edition). New York: Springer. 745.
474. Kleinbaum, D. G. (2002). *Logistic regression*, (3. Edition). New York: Springer. 43-53.
475. Faraway, J.J. (2016). *Extending the linear model with R*. (2. Edition). Londra: Chapman and Hall/CRC. 413.
476. Wood, S.N. (2017). *Generalized additive models: An introduction with R* (2. Edition). Londra: Chapman and Hall/CRC. 388.
477. Kingsford, C., Salzberg, S.L. (2008). What are decision trees?. *Nature Biotechnology*, 26(9), 1011–1013.
478. Breiman, L. (2001). Random forests. *Machine Learning*, 45(1), 5–32.

479. Ahmad, M. W., Reynolds, J., Rezgui, Y. (2018). Predictive modelling for solar thermal energy systems: A comparison of support vector regression, random forest, extra trees and regression trees. *Journal of Cleaner Production*, 203, 810–821.
480. Rish, I. (4 Ağustos 2001). *An empirical study of the naive Bayes classifier*. In IJCAI 2001 Workshop on Empirical Methods in Artificial Intelligence, 41–46.
481. Friedman, J.H. (2001). Greedy function approximation: A gradient boosting machine. *Annals of Statistics*, 29(5), 1189–1232.
482. Peterson, L. (2009). K-nearest neighbor. *Scholarpedia*, 4(1), 1883.
483. Chen, T., Guestrin, C. (17-19 Kasım 2016). *XGBoost*. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. sBakü. 144-149.
484. İnternet: *NGBoost: Natural gradient boosting for probabilistic prediction*. URL-79: <https://arxiv.org/abs/1910.03225> Son Erişim Tarihi: 24.11.2025.
485. İnternet: Boolean decision rules via column generation. URL-80: <https://arxiv.org/abs/1805.09901> Son Erişim Tarihi: 24.11.2025.
486. İnternet: Hrm, we can't find that page, URL-81: <https://www.oreilly.com/content/interpreting-predictive-models-with-skater-unboxing-modelopacity/> Son Erişim Tarihi: 24.11.2025.
487. İnternet: InterpretML: A unified framework for machine learning interpretability. URL-82: <https://doi.org/10.48550/arXiv.1909.09223> Son Erişim Tarihi: 24.11.2025.
488. Alber, M. (2019). Software and application patterns for explanation methods. (1. Edition). New York: Springer. 399.
489. İnternet: One explanation does not fit all: A toolkit and taxonomy of AI explainability techniques. URL-83: <https://doi.org/10.48550/arXiv.1909.03012> Son Erişim Tarihi: 24.11.2025.
490. Spinner, T., Schlegel, U., Schäfer, H., (2020). explAIner: A visual analytics framework for interactive and explainable machine learning. *IEEE Transactions on Visualization and Computer Graphics*, 26(1), 1064–1074.
491. Sokol, K., Hepburn, A., Poyiadzi, R. (2020). FAT Forensics: A Python toolbox for implementing and deploying fairness, accountability, and transparency algorithms in predictive systems. *Journal of Open Source Software*, 5(49), 1904.
492. Klaise, J., Van Looveren, A., Vacanti, G., (2021). Alibi Explain: Algorithms for explaining machine learning models. *Journal of Machine Learning Research*, 22 (181), 1–7.
493. İnternet: Partial Dependence Plots Documentation, URL-84: <https://pdpbox.readthedocs.io/>, Son Erişim Tarihi: 24.11.2025.

494. Goldstein, A., Kapelner, A., Bleich, J., Pitkin, E. (2015). Peeking inside the black box: Visualizing statistical learning with plots of individual conditional expectation. *Journal of Computational and Graphical Statistics*, 24(1), 44–65.
495. İnternet: PyCEbox Documentation, URL-85: <http://austinrochford.github.io/PyCEbox/docs/>, Son Erişim Tarihi: 24.11.2025.
496. İnternet: Captum: A unified and generic model interpretability library for PyTorch. URL-86: <https://doi.org/10.48550/arXiv.2009.07896> Son Erişim Tarihi: 24.11.2025.
497. İnternet: H2O Documentation, URL-87: <https://docs.h2o.ai/h2o/latest-stable/h2o-docs/welcome.html>, Son Erişim Tarihi: 24.11.2025.
498. İnternet: Ethical ML XAI Documentation, URL-88: <https://ethicalml.github.io/xai/index.html>, Son Erişim Tarihi: 24.11.2025.
499. İnternet: tf-explain Documentation, URL-89: <https://tf-explain.readthedocs.io/en/latest/>, Son Erişim Tarihi: 24.11.2025.
500. Wexler, J., Pushkarna, M., Bolukbasi, T. (2019). The What-If Tool: Interactive probing of machine learning models. *IEEE Transactions on Visualization and Computer Graphics*, 26(1), 56–65.
501. İnternet: ELI5 Documentation, URL-90: <https://eli5.readthedocs.io/en/latest/overview.html>, Son Erişim Tarihi: 24.11.2025.
502. Biecek, P. (2018). DALEX: Explainers for complex predictive models in R. *Journal of Machine Learning Research*, 19(1), 3245–3249.
503. İnternet: Metrics for explainable AI: Challenges and prospects. URL-91: <https://doi.org/10.48550/arXiv.1812.04608> Son Erişim Tarihi: 24.11.2025.
504. Mohseni, S., Zarei, N., Ragan, E.D. (2021). A multidisciplinary survey and framework for design and evaluation of explainable AI systems. *ACM Transactions on Interactive Intelligent Systems*, 8(4), 287–289.
505. İnternet: Shedding light on black box machine learning algorithms: Development of an axiomatic framework to assess the quality of methods that explain individual predictions. URL-92: <https://arxiv.org/abs/1808.05054> Son Erişim Tarihi: 24.11.2025.
506. Gunning, D., Aha, D. (2019). DARPA’s explainable artificial intelligence (XAI) program. *AI Magazine*, 40(2), 44–58.
507. İnternet: Explanation in Human-AI systems: A literature meta-review. URL-93: <https://doi.org/10.48550/arXiv.1902.01876> Son Erişim Tarihi: 24.11.2025.
508. Alvarez-Melis, D., Jaakkola, T.S. (14-16 Kasım 2018). Towards robust interpretability with self-explaining neural networks. In *Advances in Neural Information Processing Systems*. 180-183.

509. Hanif, A., Zhang, X., Wood, S. (14-16 2021). *A survey on explainable artificial intelligence techniques and challenges*. In Proceedings of the IEEE 25th International Enterprise Distributed Object Computing Workshop (EDOCW), Varşova. 81–89.
510. Gedikli, F., Jannach, D., ve Ge, M. (2014). How should I explain? A comparison of different explanation types for recommender systems. *International Journal of Human-Computer Studies*, 72(4), 367–382.
511. Coppers, S., Van Bergh, J., Luyten, K., Coninx, K. (17-19 Kasım 2018). *Intellino: An intelligible translation environment*. In Proceedings of the CHI Conference on Human Factors in Computing Systems. Glaskow. 213-219.
512. Lage, I., Chen, E., He, J. (2019). Human evaluation of models built for interpretability. In *Proceedings of the AAAI Conference on Human Computation and Crowdsourcing*, 7(1), Bakü. 59–67.
513. Eldrandaly, K.A., Abdel-Basset, M., Ibrahim, M., Abdel-Aziz, N.M. (2023). Explainable and secure artificial intelligence: Taxonomy, cases of study, learned lessons, challenges, and future directions. *Enterprise Information Systems*, 17(9), 255-267.
514. Lim, B.Y., Dey, A.K. (19-22 Kasım 2009). *Assessing demand for intelligibility in context-aware applications*. In Proceedings of the 11th International Conference on Ubiquitous Computing, Roma. 195–204.
515. Lim, B.Y., Dey, A.K., Avrahami, D. (02-03 Mayıs 2009). *Why and why not explanations improve the intelligibility of context-aware intelligent systems*. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, Varşova. 2119–2128.
516. Lakkaraju, H., Bach, S. H., Leskovec, J. (18-20 Kasım 2016). Interpretable decision sets: A joint framework for description and prediction. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Luzern. 1675–1684.
517. Yin, M., Wortman Vaughan, J., Wallach, H. (06 -08 Haziran 2019). *Understanding the effect of accuracy on trust in machine learning models*. In Proceedings of the CHI Conference on Human Factors in Computing Systems, Bologna. 1–12.
518. Nourani, M., Kabir, S., Mohseni, S., Ragan, E.D. (02-05 Ekim 2019). *The effects of meaningful and meaningless explanations on trust and perceived system accuracy in intelligent systems*. In Proceedings of the AAAI Conference on Human Computation and Crowdsourcing, Londra. 97–105.
519. Haiyan, Z., Hanjie, C., Fan, Y., Ninghao, L., Huiqi, D., Hengyi, C., Shuaiqiang, W., Dawei, Y., Mengnan, D. (2024). Explainability for large language models: A survey. *ACM Transactions on Intelligent Systems and Technology*, 15(2), 20-31.
520. İnternet: Usable XAI: 10 strategies towards exploiting explainability in the LLM era. URL-94: <https://doi.org/10.48550/arXiv.2403.08946> Son Erişim Tarihi: 24.11.2025.

521. İnternet: The Language Interpretability Tool: Extensible, interactive visualizations for NLP models URL-95: <https://doi.org/10.48550/arXiv.2008.05122> Son Erişim Tarihi: 24.11.2025.
522. Lin, R., Crichton, G., Verga, P., McCallum, A., Boyd-Graber, J. (25-27 Kasım 2022). *Phoenix: A machine learning debugging tool*. In Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP), Glaskow. 412-419.
523. Murari, K., Malkin, P., Hwang, H., ve Branson, S. (16-18 Ekim 2022). *Comgra: Interpreting neural networks through compositional graphical explanations*. In Proceedings of the International Conference on Learning Representations (ICLR), Luzern. 251-259.
524. İnternet: LLMs for XAI: Future directions for explaining explanations URL-96: <https://doi.org/10.48550/arXiv.2405.06064> Son Erişim Tarihi: 24.11.2025.
525. İnternet: XAI meets LLMs: A survey of the relation between explainable AI and large language models URL-97: <https://doi.org/10.48550/arXiv.2407.15248> Son Erişim Tarihi: 24.11.2025.
526. İnternet: Interactive analysis of LLMs using meaningful counterfactuals URL-98: <https://doi.org/10.48550/arXiv.2405.00708> Son Erişim Tarihi: 24.11.2025.
527. Lubos, S., Tran, T. N. T., Felfernig, A., Erdeniz, S. P., Le, V.-M. (17-19 Ağustos 2024). *LLM-generated explanations for recommender systems*. In Adjunct Proceedings of the 32nd ACM Conference on User Modeling, Adaptation and Personalization (UMAP Adjunct '24), 276–285.
528. Castelnovo, A., Depalmas, R., Mercorio, F., Mombelli, N., Poterti, D., Serino, A., vd. (17-19 Temmuz 2024). *Augmenting XAI with LLMs: A case study in banking marketing recommendation*. In Explainable Artificial Intelligence: Second World Conference, xAI 2024, Valletta, Malta, July 17–19, 2024, Proceedings, Part I, 211–229).
529. İnternet: Properties and challenges of LLM-generated explanations. URL-99: <https://doi.org/10.48550/arXiv.2402.10532> Son Erişim Tarihi: 24.11.2025.
530. Ajwani, R., Javaji, S. R., Rudzicz, F., Zhu, Z. (2024). LLM-generated black-box explanations can be adversarially helpful URL-100: <https://doi.org/10.48550/arXiv.2405.06800> Son Erişim Tarihi: 24.11.2025.
531. Perković, G., Drobnjak, A., Botički, I. (19-21 Haziran 2024). *Hallucinations in LLMs: Understanding and addressing challenges*. In 2024 47th MIPRO ICT and Electronics Convention (MIPRO), Belgrad. 2084–2088.
532. Barman, K. G., Wood, N., Pawlowski, P. (2024). Beyond transparency and explainability: On the need for adequate and contextualized user guidelines for LLM use. *Ethics and Information Technology*, 26(47), 197-211.

533. İnternet: Navigating LLM ethics: Advancements, challenges, and future directions URL-101: <https://doi.org/10.48550/arXiv.2406.18841> Son Erişim Tarihi: 24.11.2025.
534. İnternet: Current state of LLM risks and AI guardrails URL-102: <https://doi.org/10.48550/arXiv.2406.12934> Son Erişim Tarihi: 24.11.2025.
535. Šarčević, A., Pintar, D., Vranić, D., (2022). Cybersecurity knowledge extraction using XAI. *Applied Sciences*, 12(3), 8669.
536. Warnecke, A., Arp, D., Wressnegger, C., Rieck, K. (12-14 Kasım 2020). *Evaluating explanation methods for deep learning in security*. In Proceedings of the IEEE European Symposium on Security and Privacy (EuroS&P), Helsinki. 158–174.
537. İnternet: *Interpretable machine learning: Definitions, methods, and applications* URL-103: <https://doi.org/10.48550/arXiv.1901.04592> Son Erişim Tarihi: 24.11.2025.
538. Yan, F., Wen, S., Nepal, S., Paris, C., ve Xiang, Y. (2022). Explainable machine learning in cybersecurity: A survey. *International Journal of Intelligent Systems*, 19(4), 512-519.
539. Taddeo, M., McCutcheon, T., Floridi, L. (2019). Trusting artificial intelligence in cybersecurity is a double-edged sword. *Nature Machine Intelligence*, 1(12), 557–560.
540. Elkhawaga, G., Elzeki, O. M., Abu-Elkheir, M., Reichert, M. (2024). Why should I trust your explanation? An evaluation approach for XAI methods applied to predictive process monitoring results. *IEEE Transactions on Artificial Intelligence*, 5(4), 1458–1472.
541. İnternet: TRUBA Hakkında, URL-104: https://docs.truba.gov.tr/0-truba/TRUBA_genel.html Son Erişim Tarihi: 24.11.2025.
542. İnternet: Google Research, Frequency Asked Questions, URL-105: <https://research.google.com/colaboratory/faq.html> Son Erişim Tarihi: 24.11.2025.
543. İnternet: Sharma A., A Comprehensive Guide to Google Colab: Features, Usage, and Best Practices, URL-106: <https://www.analyticsvidhya.com/blog/2020/03/google-colab-machine-learning-deep-learning/> Son Erişim Tarihi: 24.11.2025.
544. İnternet: Das T., Google Colab: What, How, and Features, URL-107: <https://geekflare.com/dev/google-colab/> Son Erişim Tarihi: 24.11.2025.
545. İnternet: CICFlowMeter, URL-108: <https://www.unb.ca/cic/research/applications.html#CICFlowMeter>, Son Erişim Tarihi: 24.11.2025.
546. Sharafaldin, I., Gharib, A., Habibi Lashkari, A., Ghorbani, A. (2017). Towards a reliable intrusion detection benchmark dataset. *Journal of Software Networking*, 20(7), 177–200.

547. Uddin, M.T., Uddiny, M.A. (22-23 Aralık 2015). *A guided random forest-based feature selection approach for activity recognition*. In 2015 International Conference on Electrical Engineering and Information Communication Technology (ICEEICT), 1–6.
548. Schober, P., Boer, C., Schwarte, L.A. (2018). Correlation coefficients: Appropriate use and interpretation. *Anesthesia & Analgesia*, 126(5), 1763–1768.
549. Al, S., Dener, M. (2021). STL-HDL: A new hybrid network intrusion detection system for imbalanced dataset on big data environment. *Computers & Security*, 110(5), 102435.
550. Chawla, N.V., Bowyer, K.W., Hall, L.O., Kegelmeyer, W.P. (2002). SMOTE: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, 16(1), 321–357.
551. Ivan, T. (1976). Two modifications of CNN. *IEEE Transactions on Systems, Man, and Cybernetics*, 6, 769–772.
552. Tyree, S., Weinberger, K.Q., Agrawal, K., Paykin, J. (03-05 Haziran 2011). *Parallel boosted regression trees for Web search ranking*. In Proceedings of the 20th International Conference on World Wide Web, 387–396.
553. İnternet: SHGNN: Structure-aware heterogeneous graph neural network. URL-109: <https://arxiv.org/abs/2112.06244> Son Erişim Tarihi: 24.11.2025.
554. İnternet: Geometric deep learning: Grids, groups, graphs, geodesics, and gauges. URL-110: <https://doi.org/10.48550/arXiv.2104.13478> Son Erişim Tarihi: 24.11.2025.
555. İnternet: Local structure-aware graph contrastive representation learning. URL-111: <https://doi.org/10.48550/arXiv.2308.03271> Son Erişim Tarihi: 24.11.2025.
556. Kipf, T.N., Welling, M. (02-05 Mart 2017). *Semi-supervised classification with graph convolutional networks*. In International Conference on Learning Representations (ICLR), Varşova. 288-293.
557. Hamilton, W. L., Ying, R., Leskovec, J. (19-21 Ocak 2017). *Inductive representation learning on large graphs*. In Advances in Neural Information Processing Systems (NeurIPS). 22-37.
558. Veličković, P., (09-11 Kasım 2018). *Graph attention networks*. In International Conference on Learning Representations (ICLR). Belgrad. 521-533.
559. Xu, K. (22-24 Aralık 2019). *How powerful are graph neural networks?*. In International Conference on Learning Representations (ICLR), Pekin. 228-230.
560. Defferrard, M., Bresson, X., Vandergheynst, P. (28-30 Kasım 2016). *Convolutional neural networks on graphs with fast localized spectral filtering*. In Neural Information Processing Systems (NeurIPS), Glaskow. 411-419.

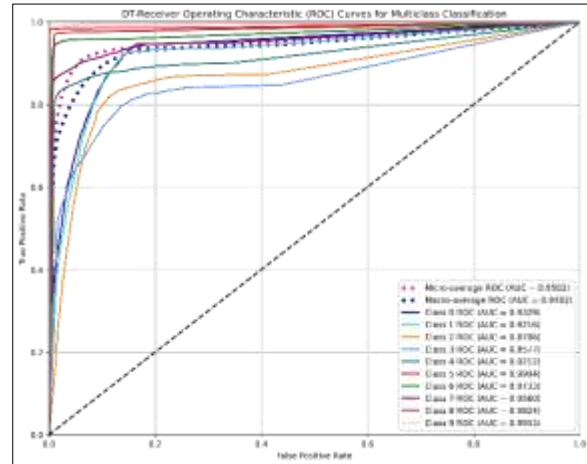
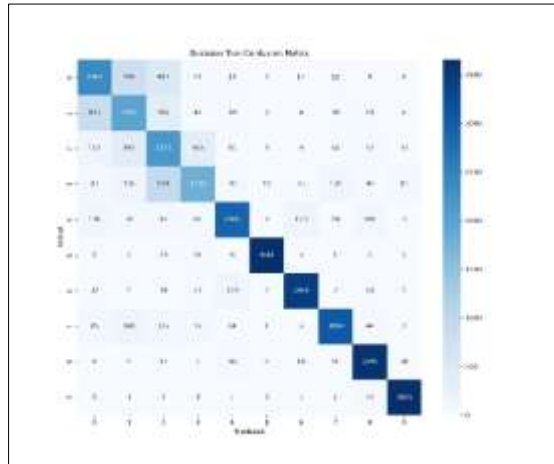
561. Shapley, L.S. (1953). *A value for n-person games*. (2. Baskı). New Jersey: Princeton University Press. 307-317.
562. İnternet: Margot V., A Brief Overview of Methods to Explain AI (XAI), URL-112: <https://towardsdatascience.com/a-brief-overview-of-methods-to-explain-ai-xai-fe0d2a7b05d6/>, Son Erişim Tarihi: 24.11.2025.
563. Dwivedi, R., Dave, D., Naik, H., Singhal, S., Omer, R., Patel, P., Qian, B., Wen, Z., Shah, T., Morgan, G., Ranjan, R. (2023). Explainable AI (XAI): Core ideas, techniques, and solutions. *ACM Computing Surveys*, 55(9),194.
564. İnternet: Molnar C., SHAP, URL-113: <https://christophm.github.io/interpretable-ml-book/shap.html>, Son Erişim Tarihi: 24.11.2025.
565. İnternet: GNNExplainer: Generating explanations for graph neural networks. URL-114: <https://doi.org/10.48550/arXiv.1903.03894> Son Erişim Tarihi: 24.11.2025.
566. İnternet: A survey of explainable graph neural networks. URL-115: <https://arxiv.org/pdf/2306.01958> Son Erişim Tarihi: 24.11.2025.
567. Paxson, V. (1999). Bro: A system for detecting network intruders in real time. *Computer Networks*, 31(23–24), 2435–2463.
568. İnternet: Zeek Documentation, URL-: <https://docs.zeek.org>, Son Erişim Tarihi: 24.11.2025.
569. İnternet: Zeek industrial protocol parsers. URL-116: <https://github.com/cisagov/ICSNPP> Son Erişim Tarihi: 24.11.2025.
570. Sommer, R., Paxson, V. (08 Temmuz 2010). *Outside the Closed World: On Using Machine Learning for Network Intrusion Detection*. 2010 IEEE Symposium on Security and Privacy. Viyana, 305-316.
571. Garcia, S., Zunino, A., Campo, M. (2023). Integrating Zeek and machine learning for network intrusion detection: A data-driven approach. *IEEE Access*, 11, 145820–145836.
572. İnternet: Araştırmalara genel bakış, URL-117: <https://openai.com/tr-TR/research/>, Son Erişim Tarihi: 24.11.2025.
573. İnternet: How ChatGPT and our foundation models are developed, URL-118: <https://help.openai.com/en/articles/7842364-how-chatgpt-and-our-foundation-models-are-developed>, Son Erişim Tarihi: 24.11.2025.
574. İnternet: Gemini, URL-119: <https://tr.wikipedia.org/wiki/Gemini>, Son Erişim Tarihi: 24.11.2025.
575. İnternet: Gemini 3, URL-120: <https://deepmind.google/models/gemini/>, Son Erişim Tarihi: 24.11.2025.

576. Han, S., Wang, M., Zhang, J., Li, D., Duan, J. (2024). A Review of Large Language Models: Fundamental Architectures, Key Technological Evolutions, Interdisciplinary Technologies Integration, Optimization and Compression Techniques, Applications, and Challenges. *Electronics*, 13(24), 5040.
577. İnternet: ChatIDS: Explainable cybersecurity using generative AI URL-121: <https://arxiv.org/abs/2306.14504> Son Erişim Tarihi: 24.11.2025.
578. İnternet: Using LLMs as AI agents to identify false positive alerts in security operation center, URL-122: <https://doi.org/10.21203/rs.3.rs-5420741/v1> Son Erişim Tarihi: 24.11.2025.

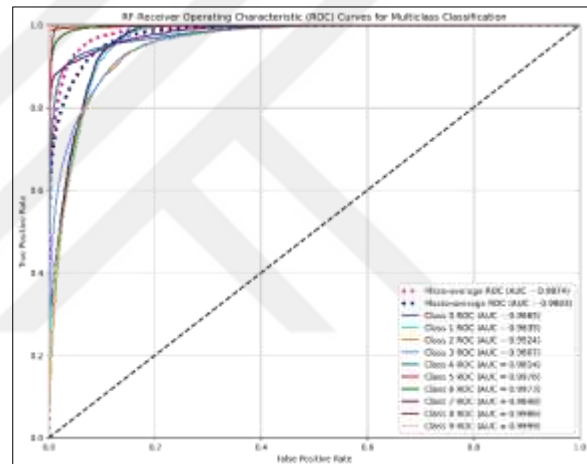
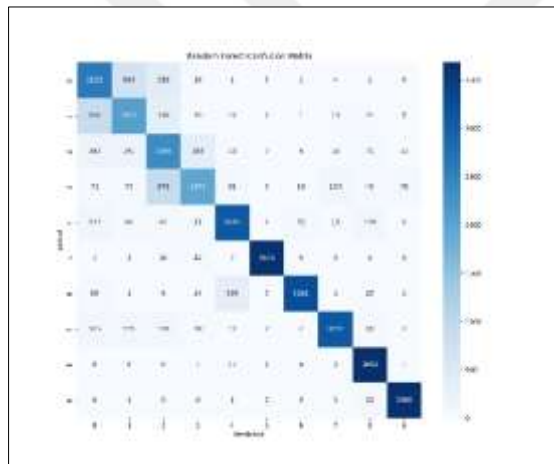




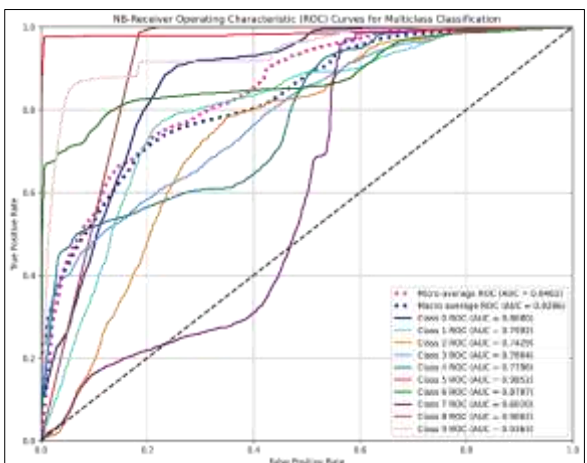
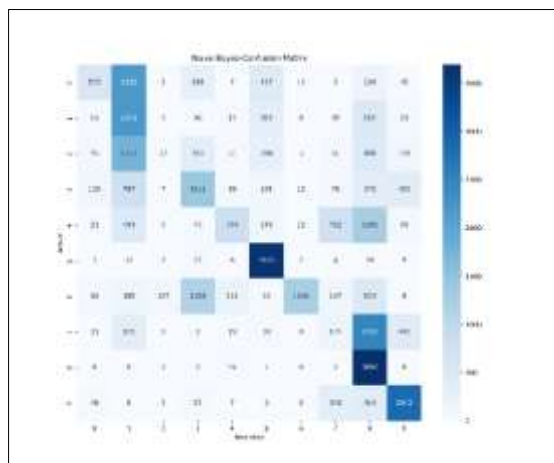
EK-1. UNSW-NB15 veri kümesi çok sınıflı sınıflandırma karmaşıklık matrisleri ve ROC eğrileri



Decision Tree

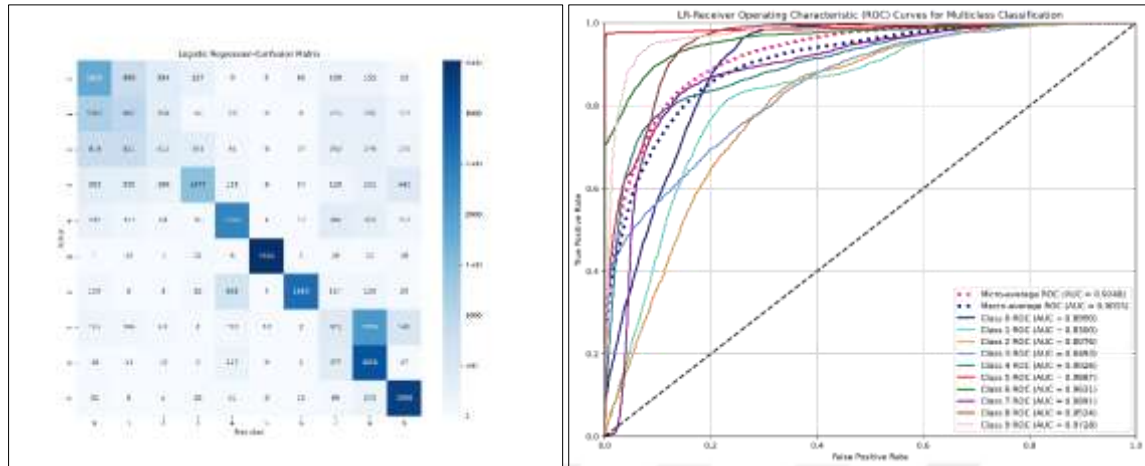


Random Forest

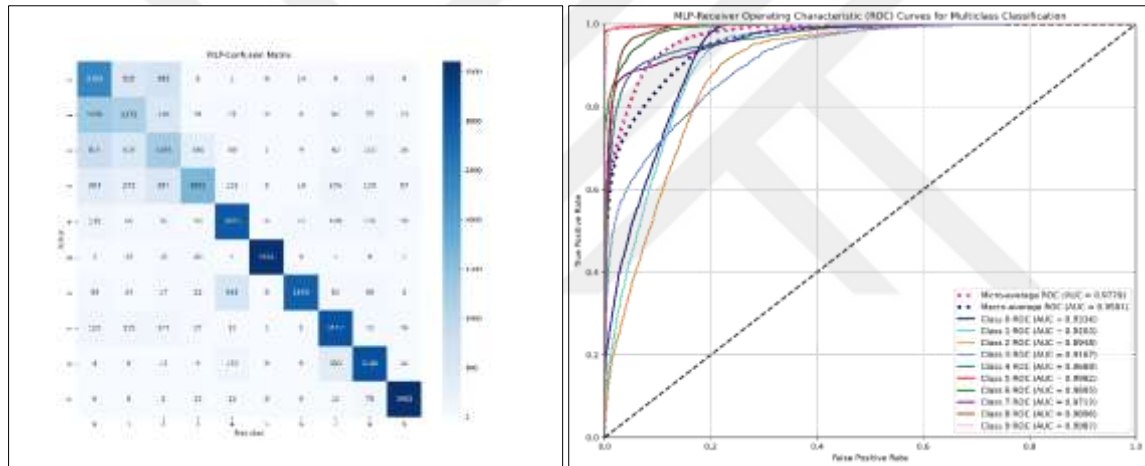


Naive Bayes

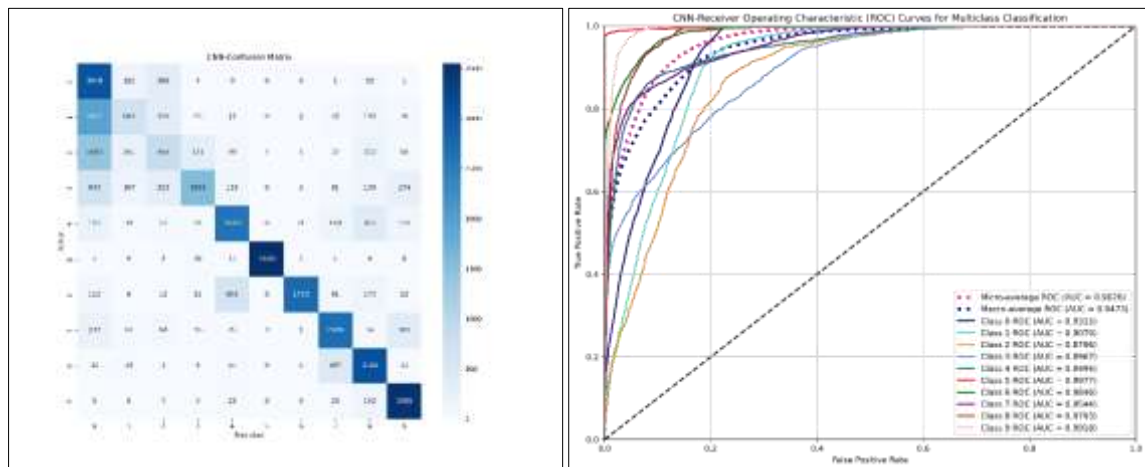
EK-1. (devam) UNSW-NB15 veri kümesi çok sınıflı sınıflandırma karmaşıklık matrisleri ve ROC eğrileri



Logistic Regression

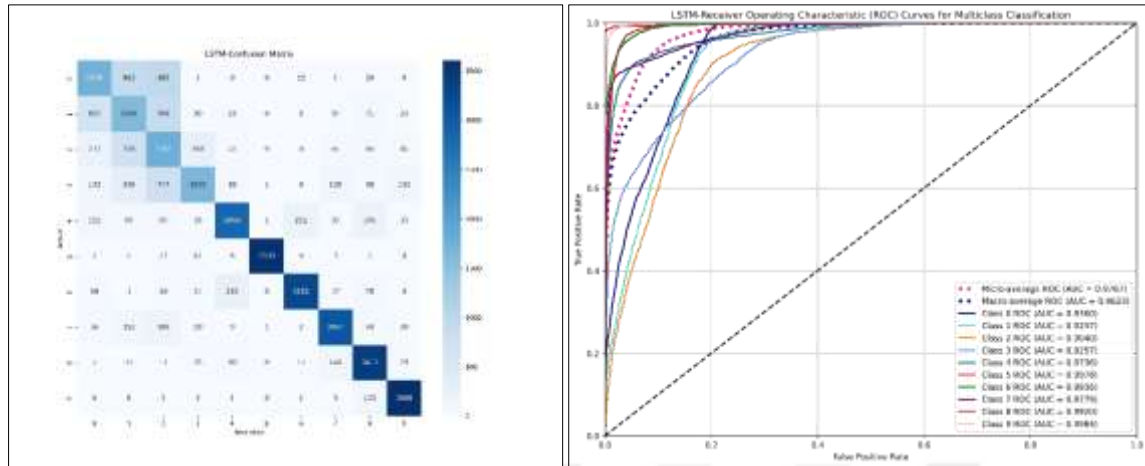


MLP

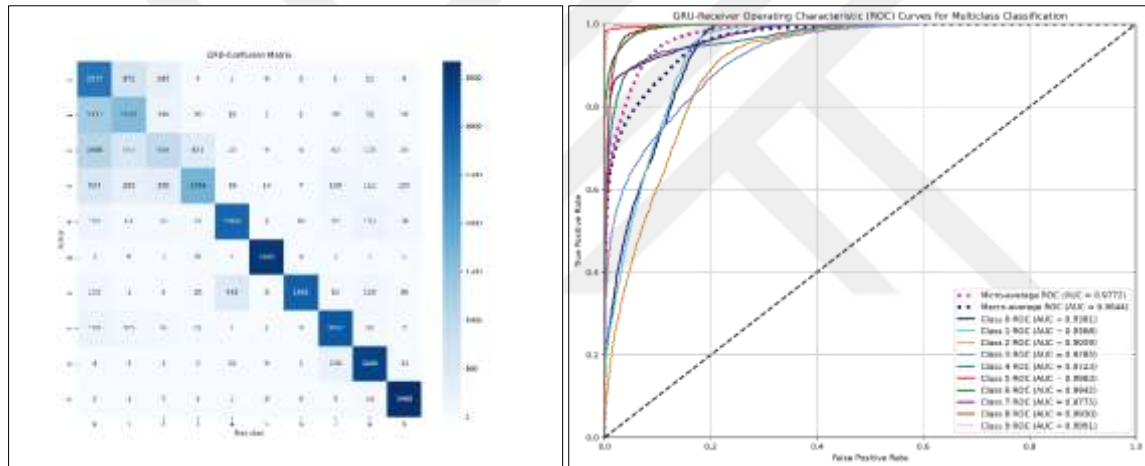


CNN

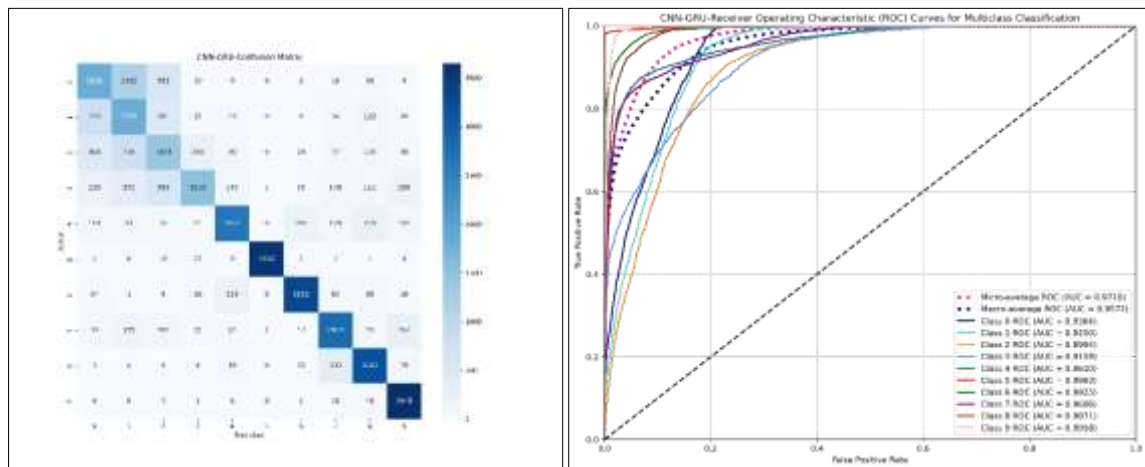
EK-1. (devam) UNSW-NB15 veri kümesi çok sınıflı sınıflandırma karmaşıklık matrisleri ve ROC eğrileri



LSTM

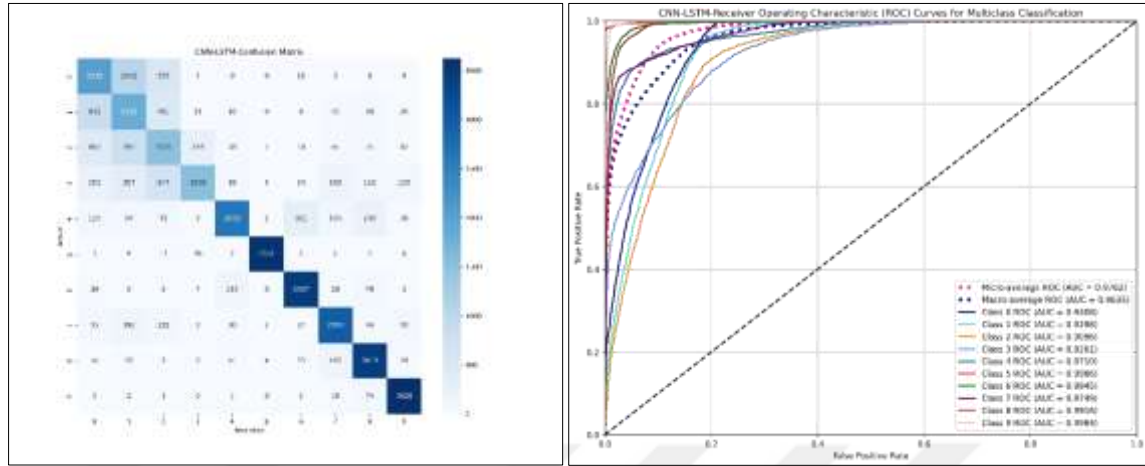


GRU

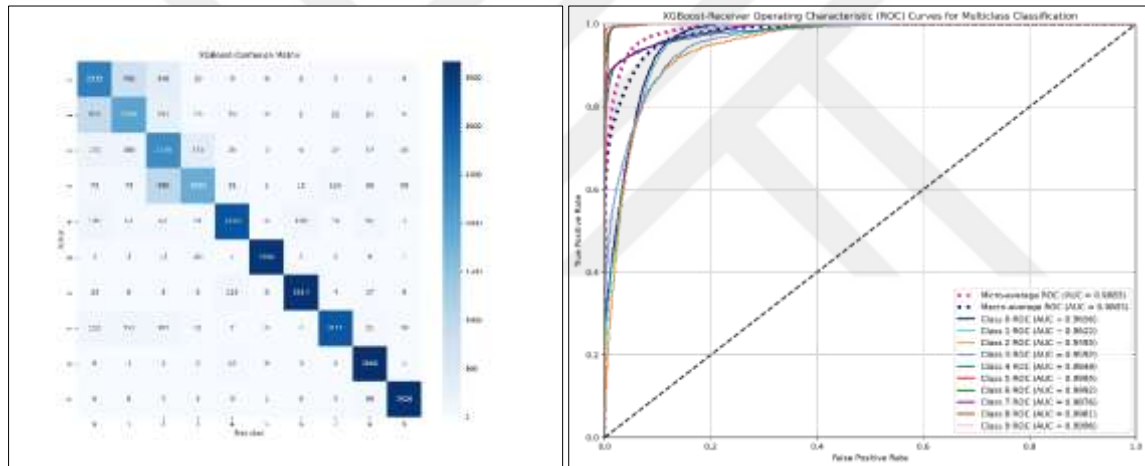


CNN-GRU

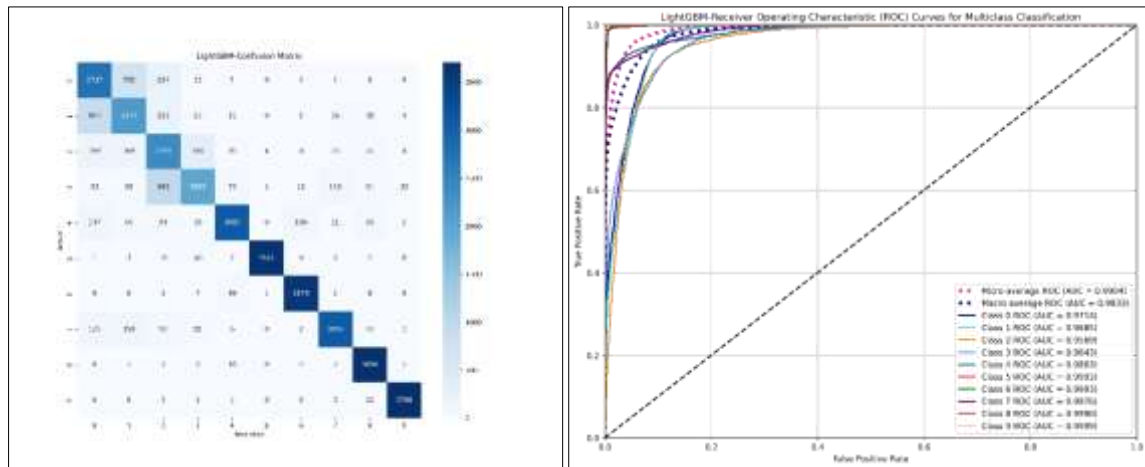
EK-1. (devam) UNSW-NB15 veri kümesi çok sınıflı sınıflandırma karmaşıklık matrisleri ve ROC eğrileri



CNN-LSTM

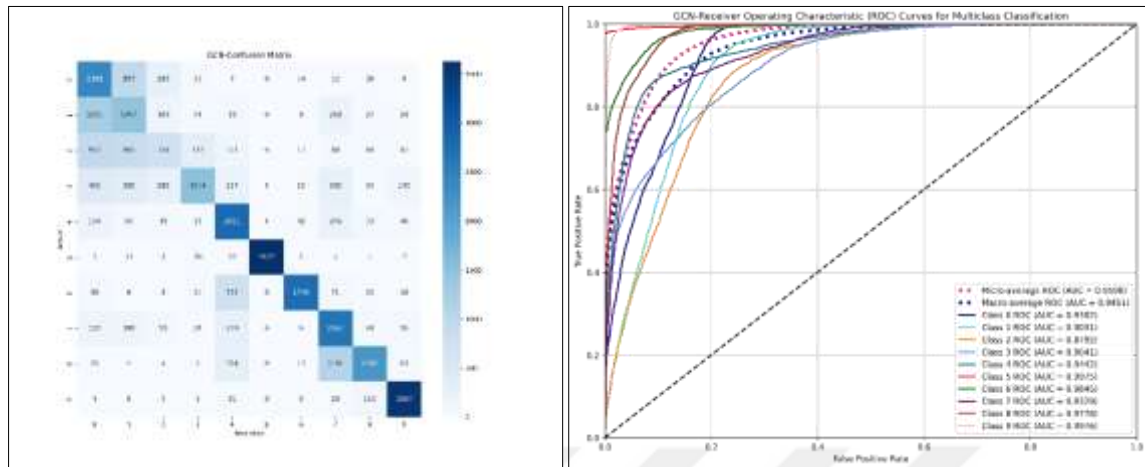


XGBoost

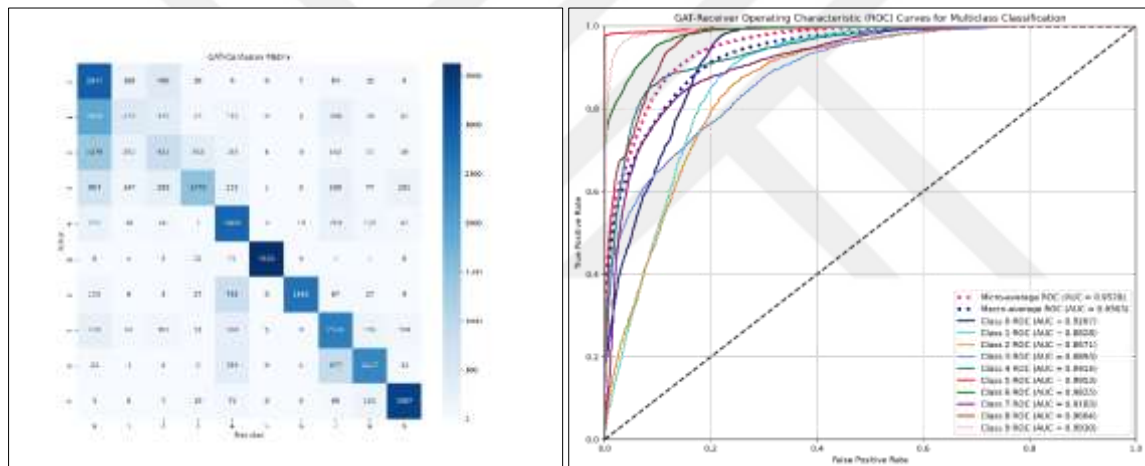


LightGBM

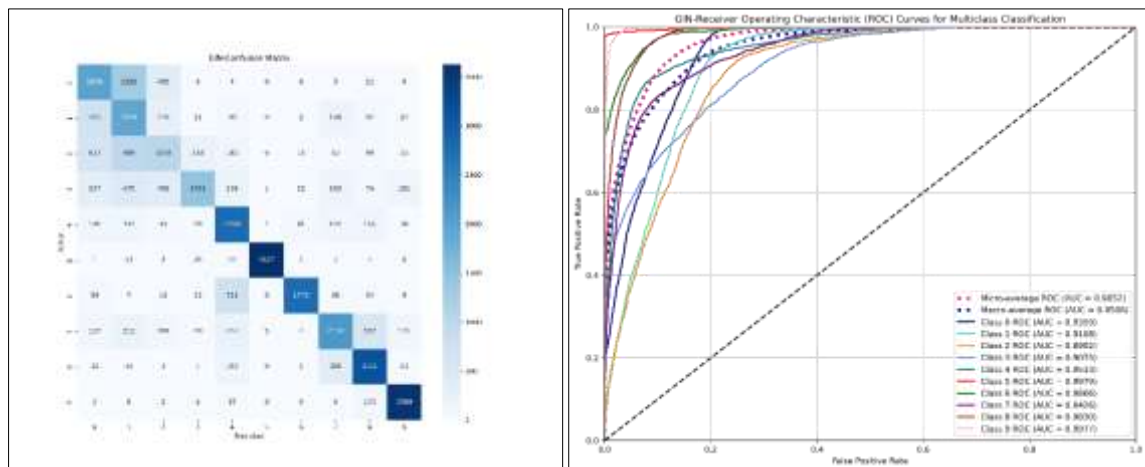
EK-1. (devam) UNSW-NB15 veri kümesi çok sınıflı sınıflandırma hata matrisleri ve ROC eğrileri



GCN

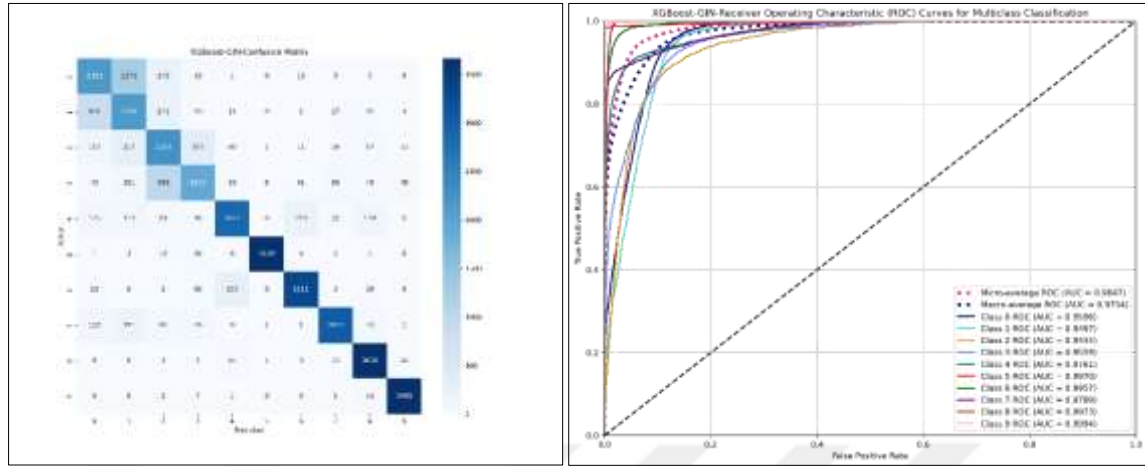


GAT

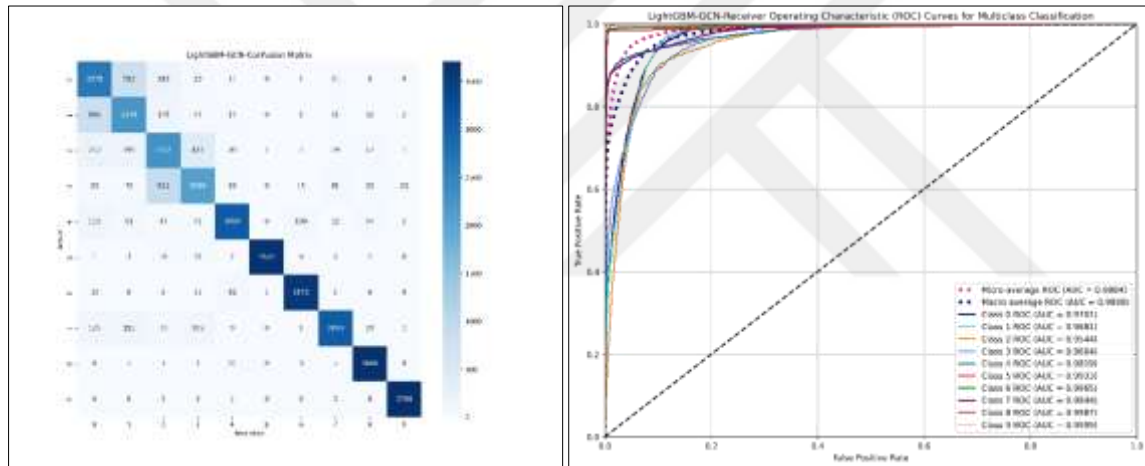


GIN

EK-1. (devam) UNSW-NB15 veri kümesi çok sınıflı sınıflandırma karmaşıklık matrisleri ve ROC eğrileri

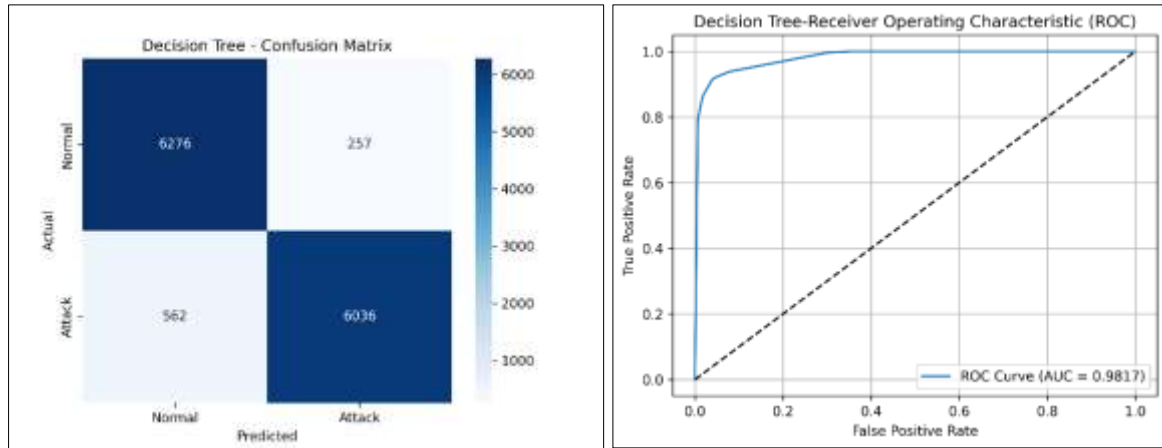


XGBoost-GIN

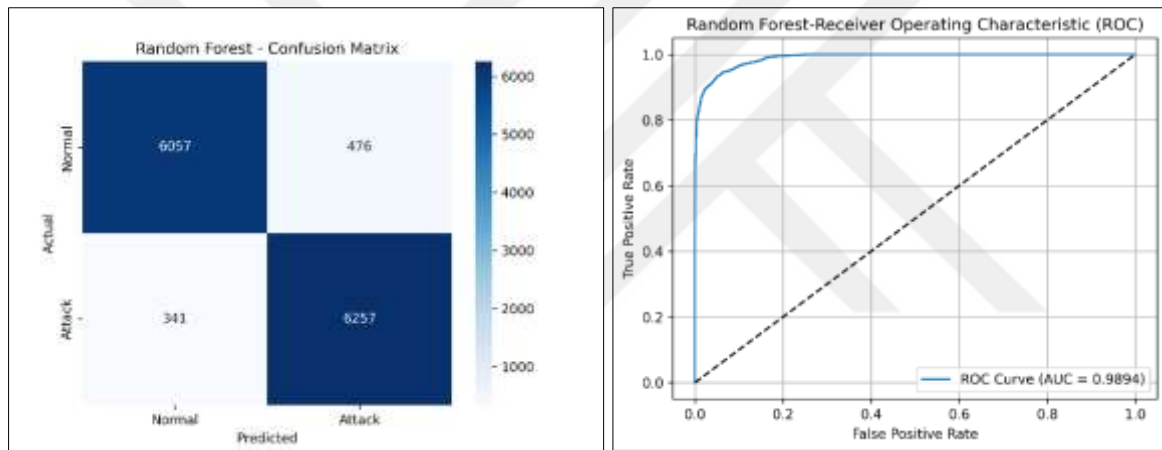


Önerilen Model

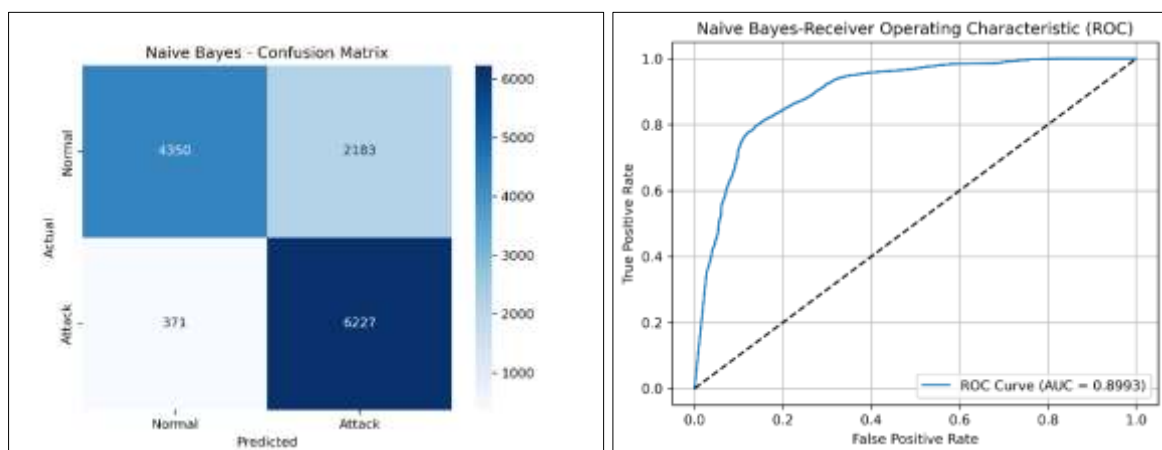
EK-2. UNSW-NB15 veri kümesi ikili sınıflandırma karmaşıklık matrisleri ve ROC eğrileri



Decision Tree

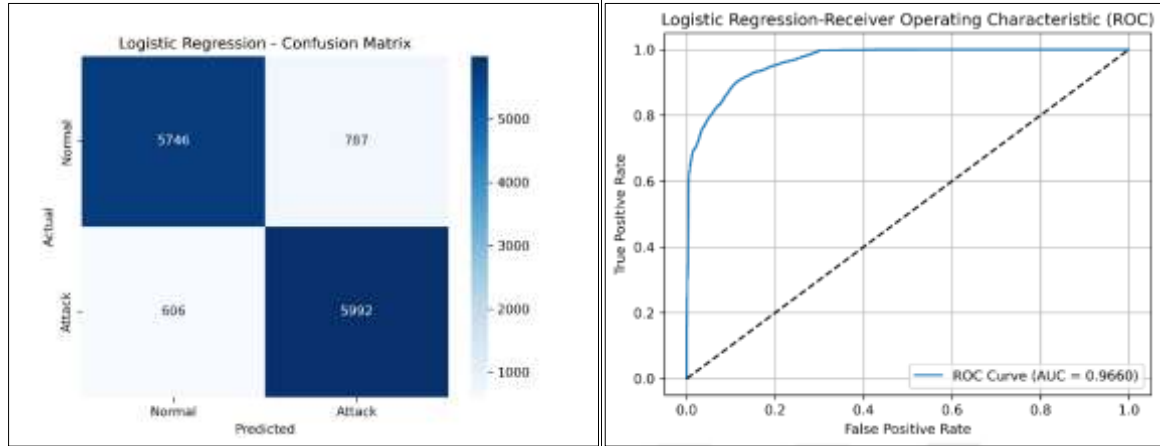


Random Forest

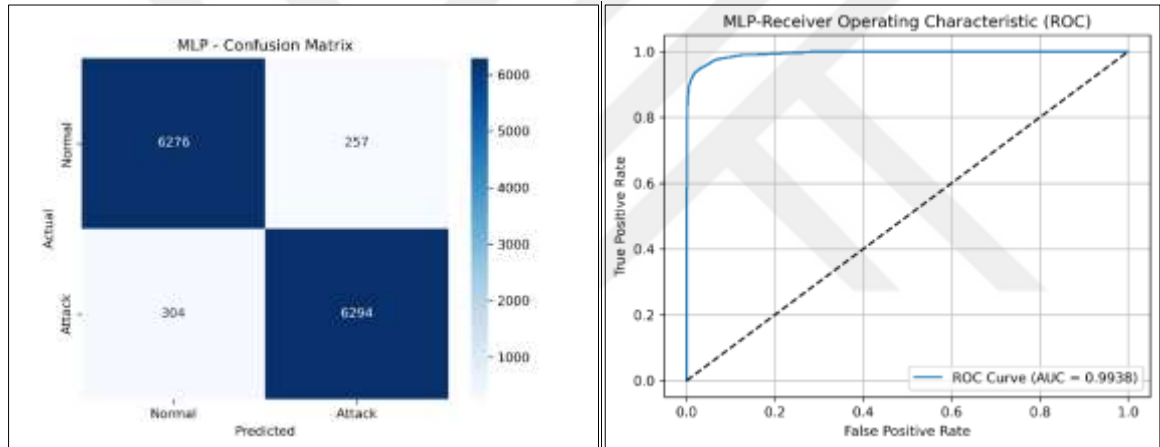


Naive Bayes

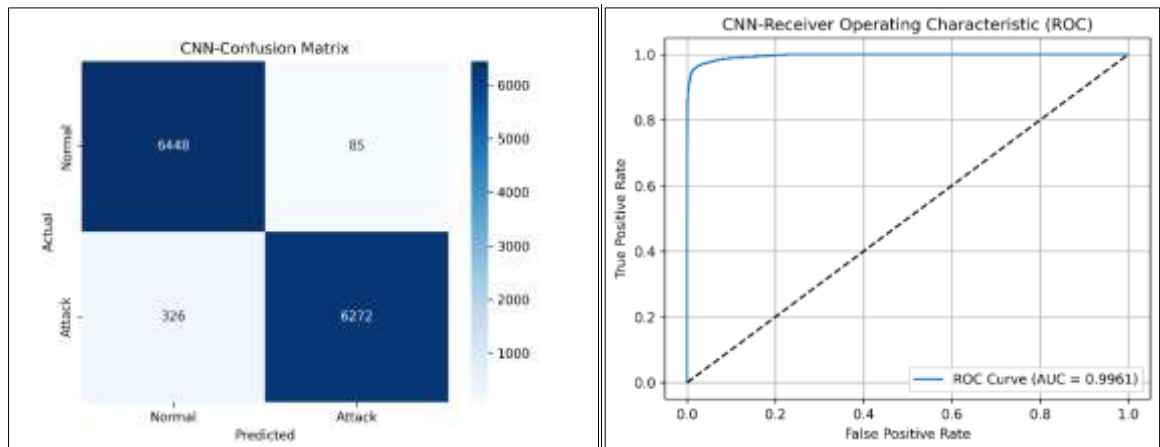
EK-2. (devam) UNSW-NB15 veri kümesi ikili sınıflandırma karmaşıklık matrisleri ve ROC eğrileri



Logistic Regression

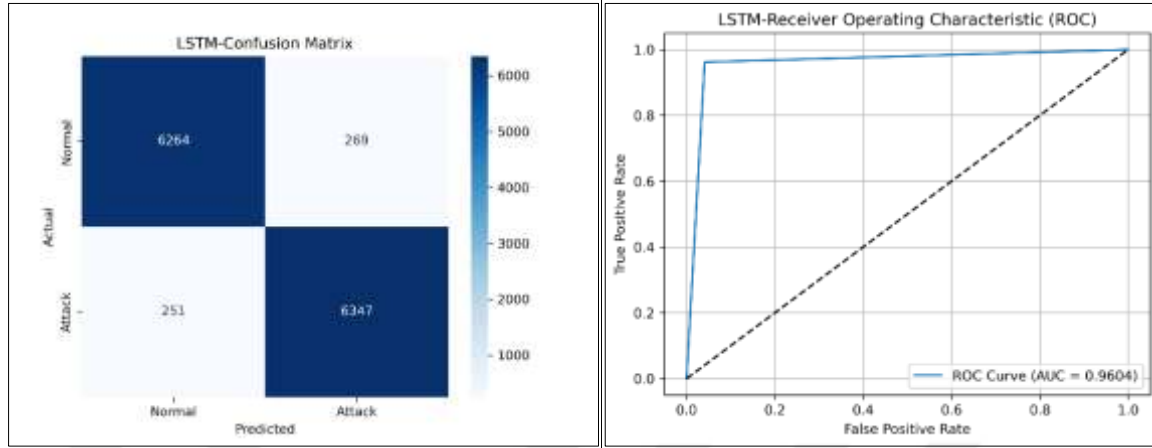


MLP

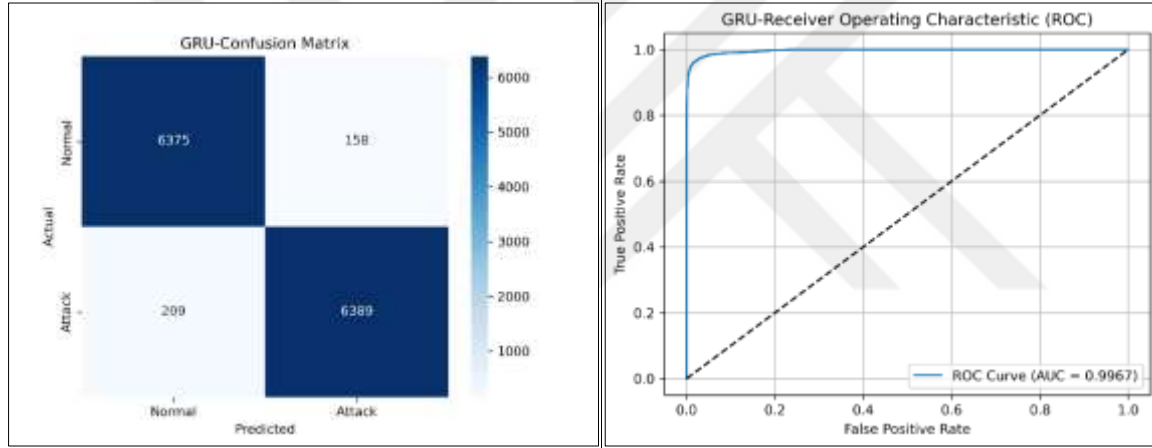


CNN

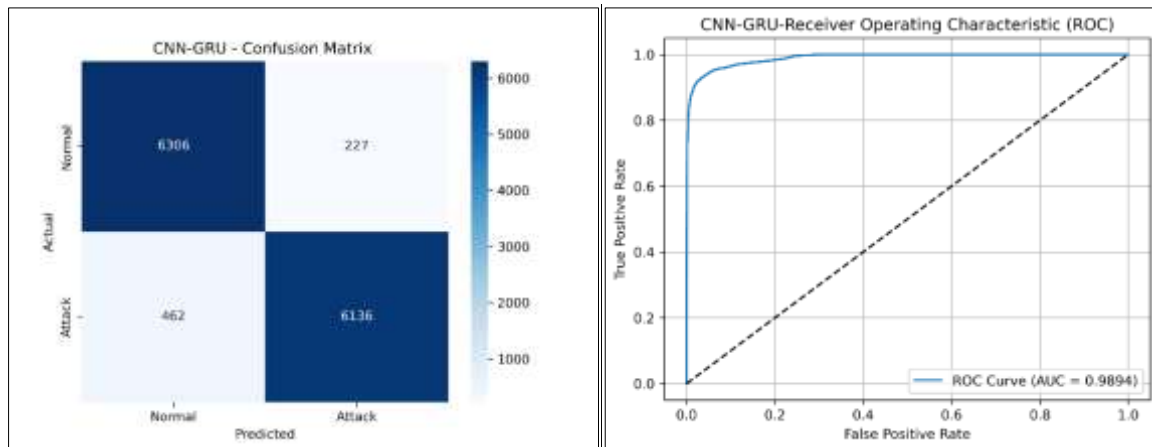
EK-2. (devam) UNSW-NB15 veri kümesi ikili sınıflandırma karmaşıklık matrisleri ve ROC eğrileri



LSTM

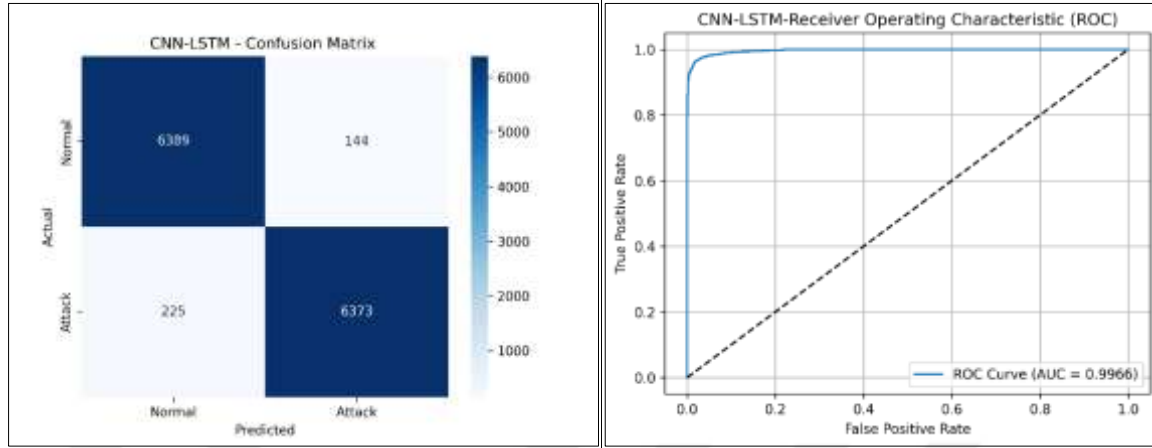


GRU

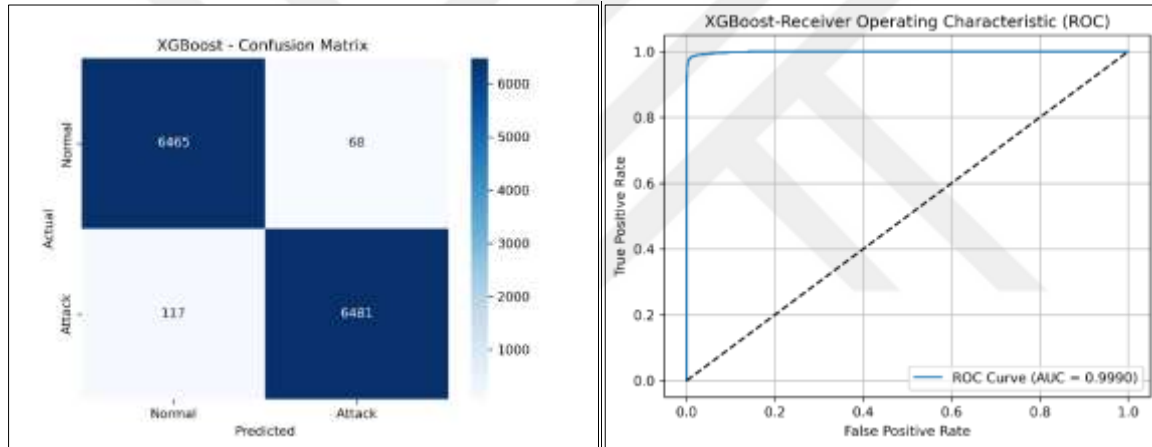


CNN-GRU

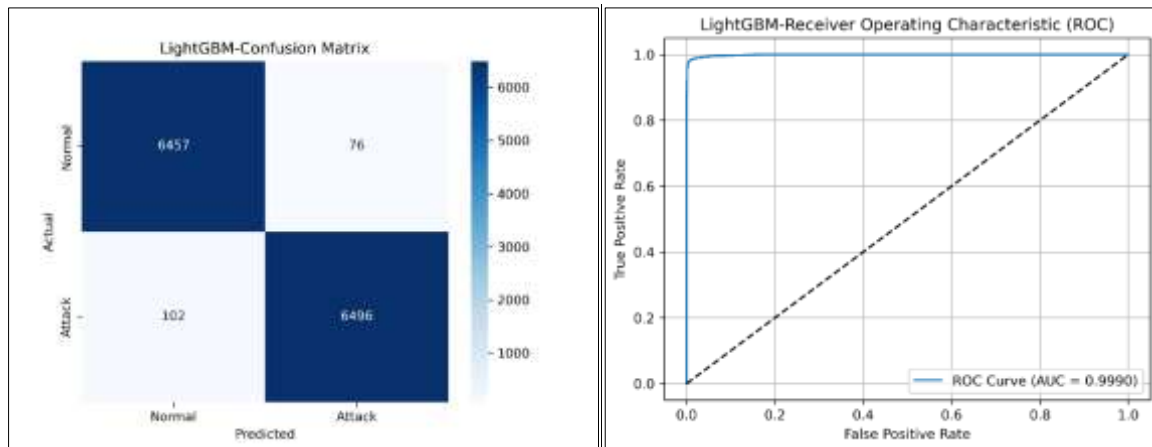
EK-2. (devam) UNSW-NB15 veri kümesi ikili sınıflandırma karmaşıklık matrisleri ve ROC eğrileri



CNN-LSTM

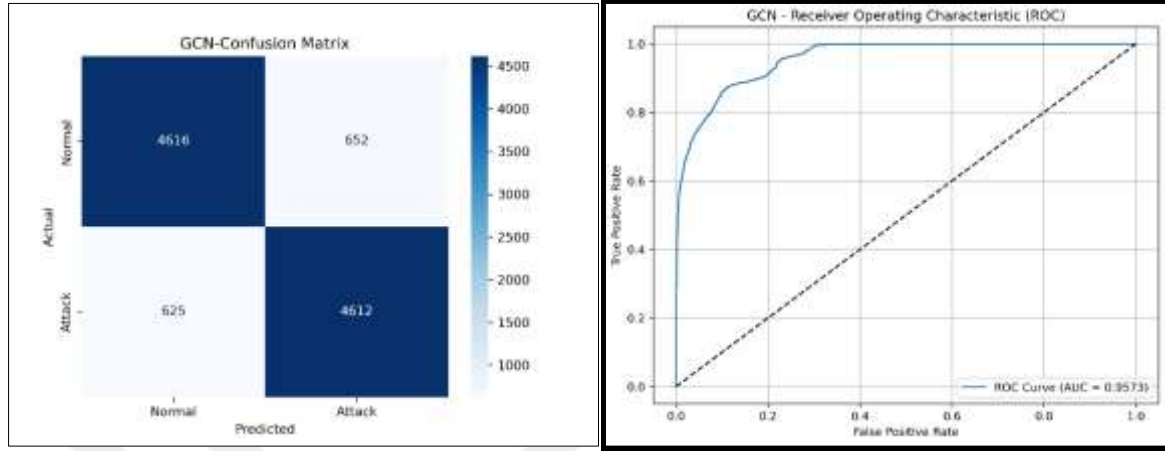


XGBoost

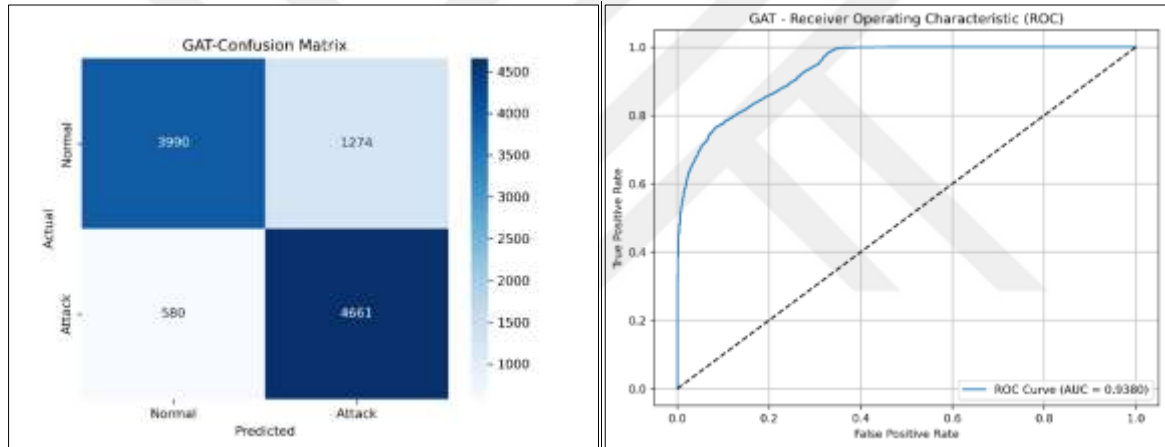


LightGBM

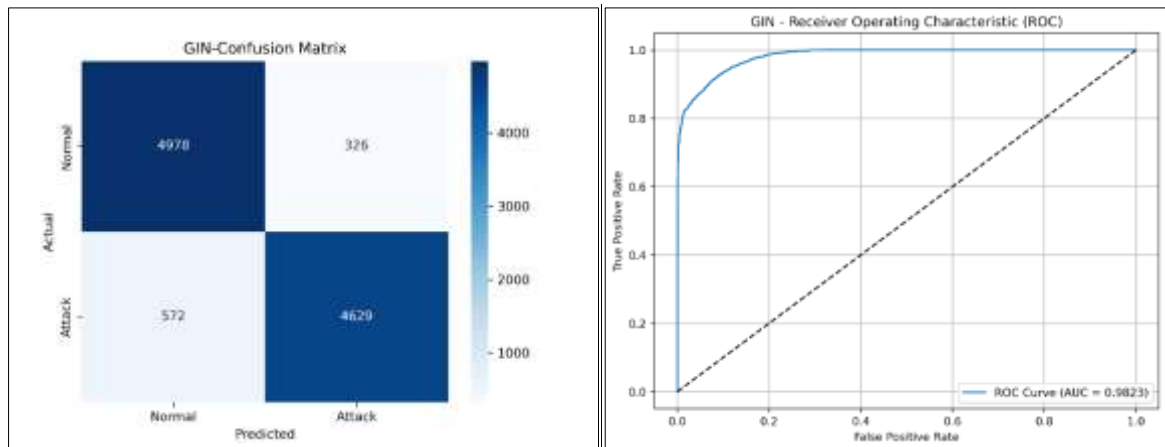
EK-2. (devam) UNSW-NB15 veri kümesi ikili sınıflandırma karmaşıklık matrisleri ve ROC eğrileri



GCN

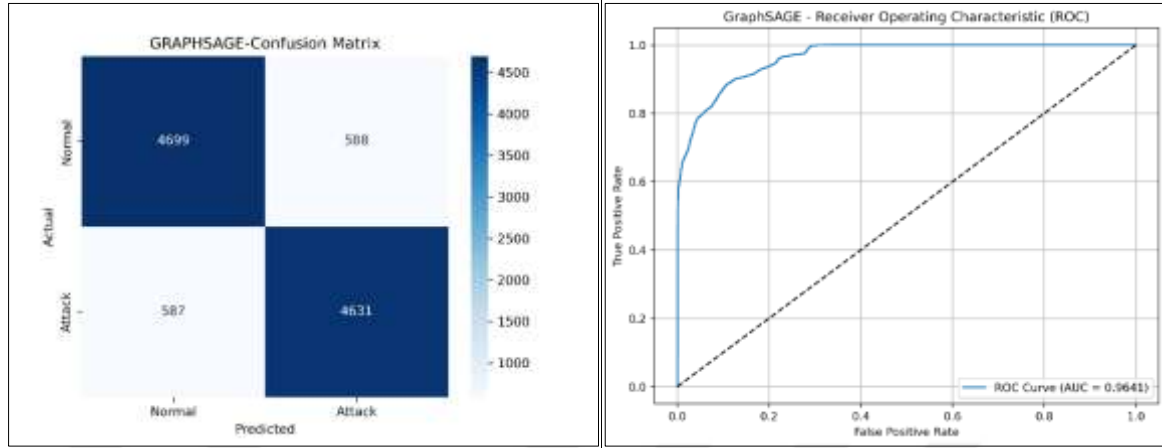


GAT

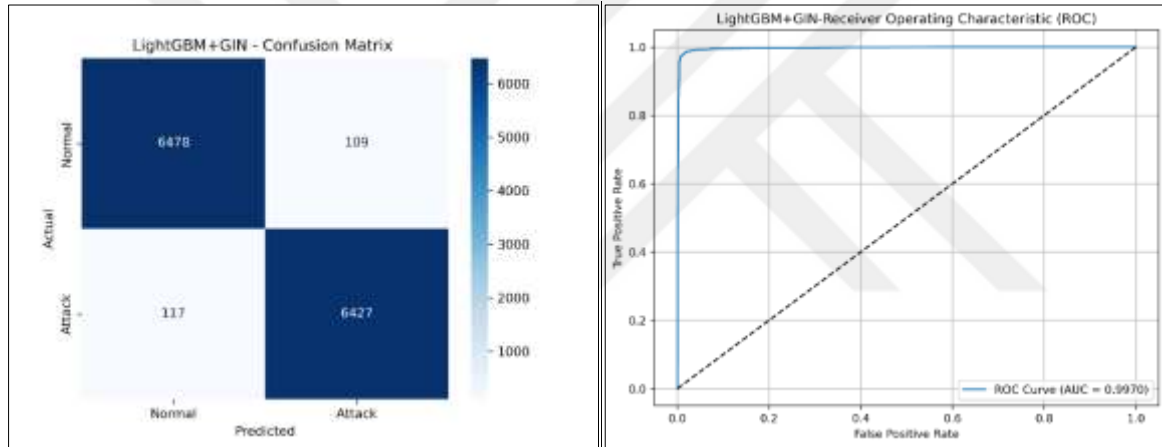


GIN

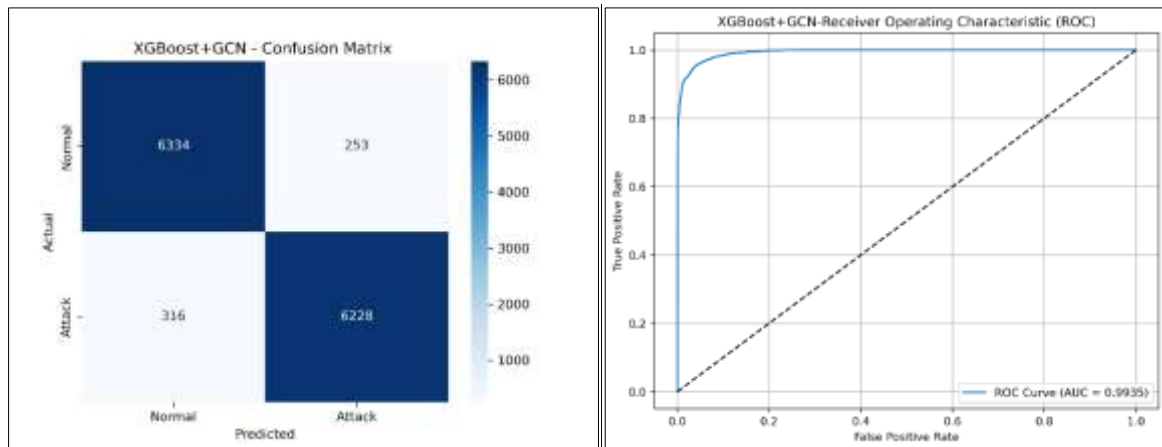
EK-2. (devam) UNSW-NB15 veri kümesi ikili sınıflandırma karmaşıklık matrisleri ve ROC eğrileri



GraphSAGE

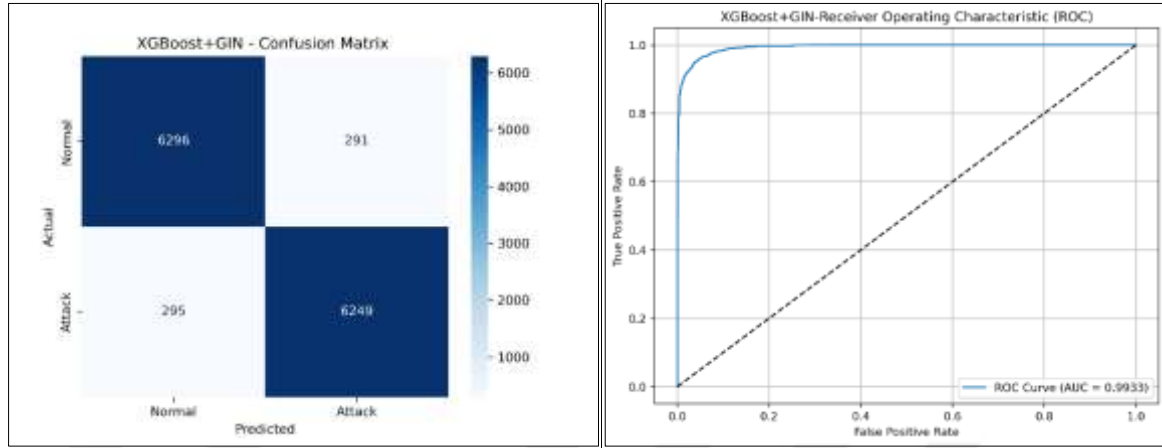


LightGBM-GIN

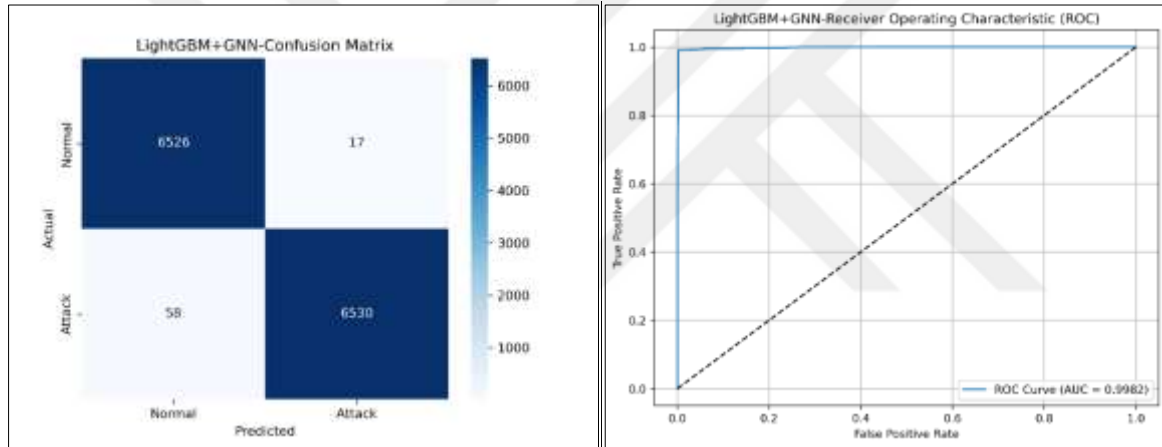


XGBoost-GCN

EK-2. (devam) UNSW-NB15 veri kümesi ikili sınıflandırma karmaşıklık matrisleri ve ROC eğrileri

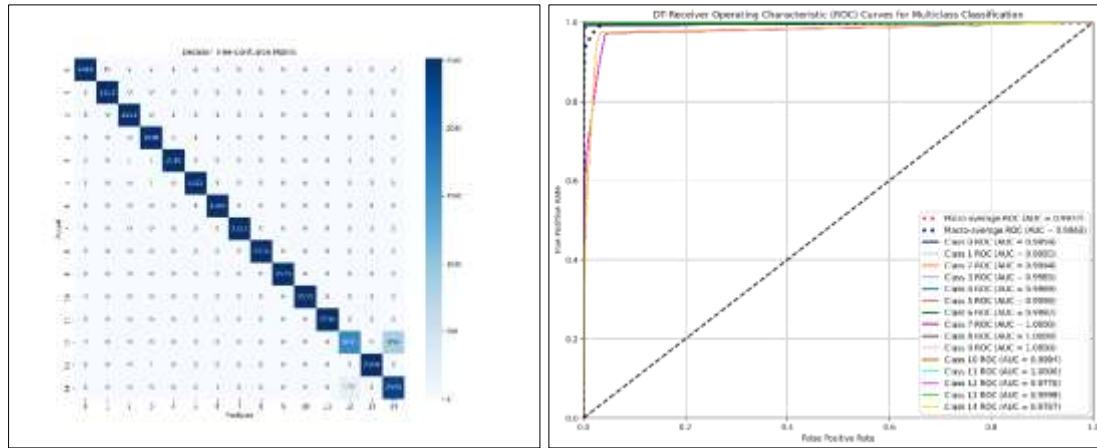


XGBoost-GIN

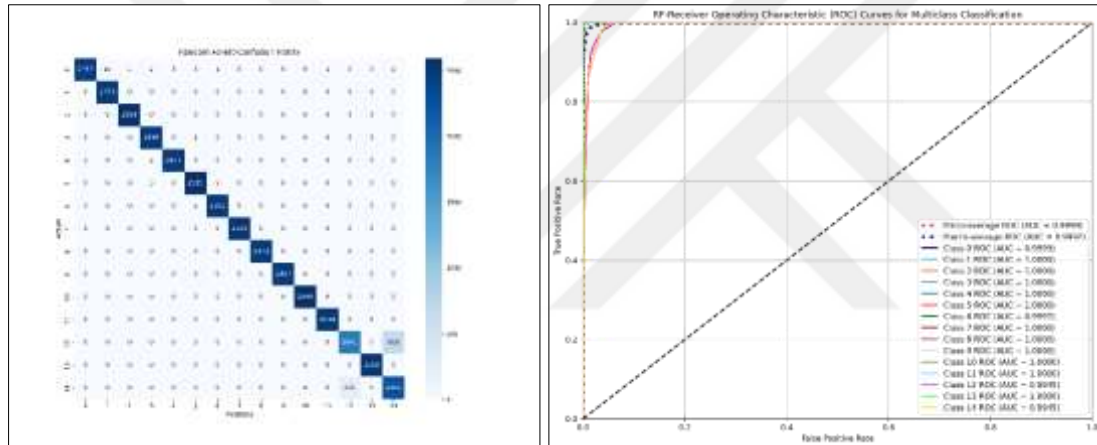


Önerilen Model

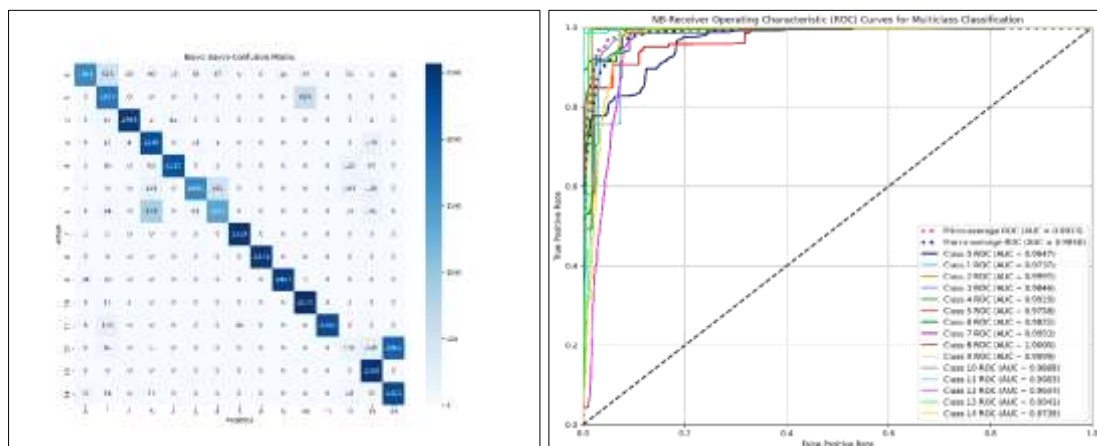
EK-3. CIC-IDS2017 veri kümesi çok sınıflı sınıflandırma karmaşıklık matrisleri ve ROC eğrileri



Decision Tree

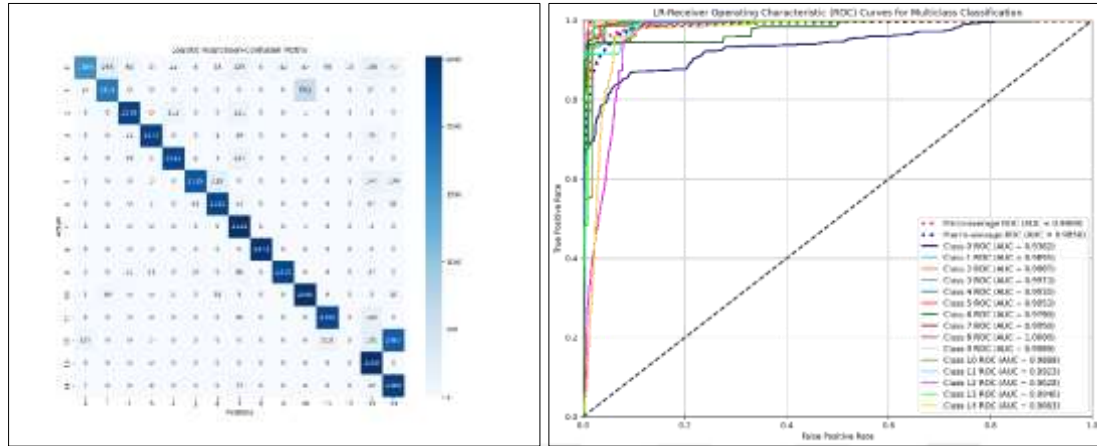


Random Forest

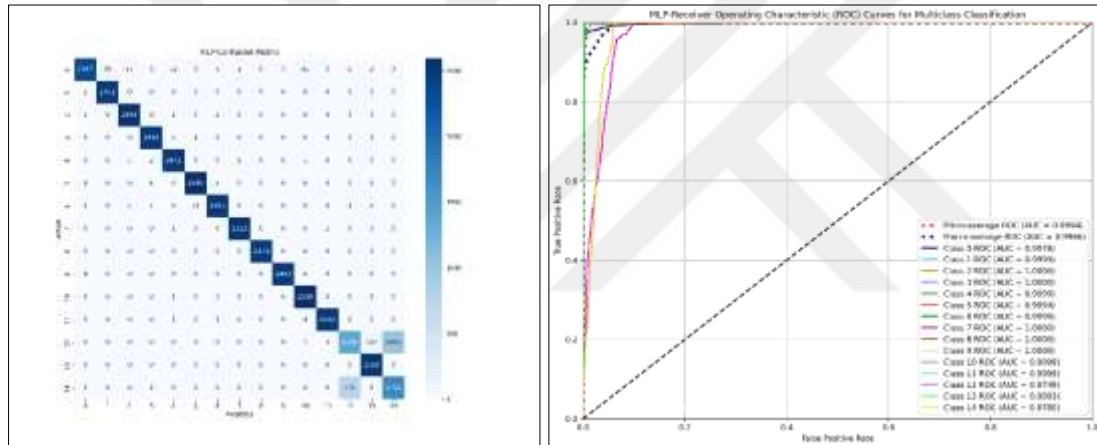


Naive Bayes

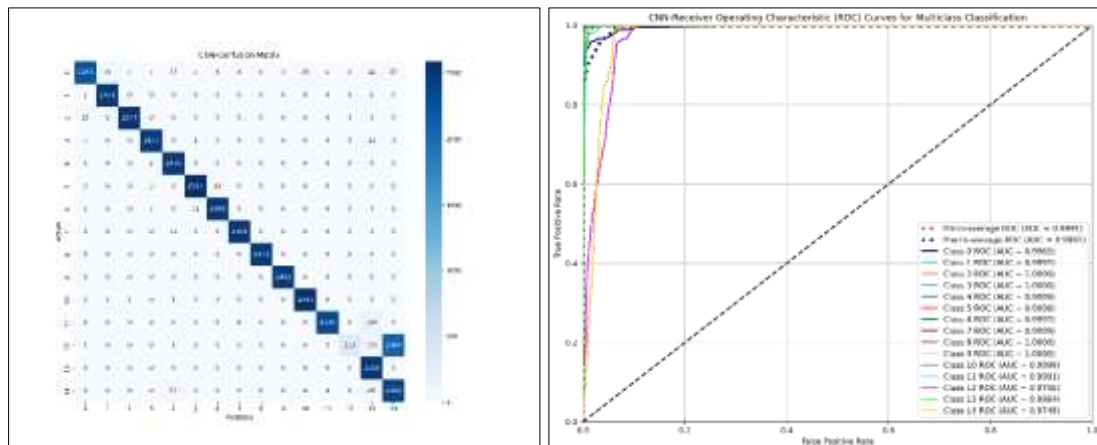
EK-3. (devam) CIC-IDS2017 veri kümesi çok sınıflı sınıflandırma karmaşıklık matrisleri ve ROC eğrileri



LR

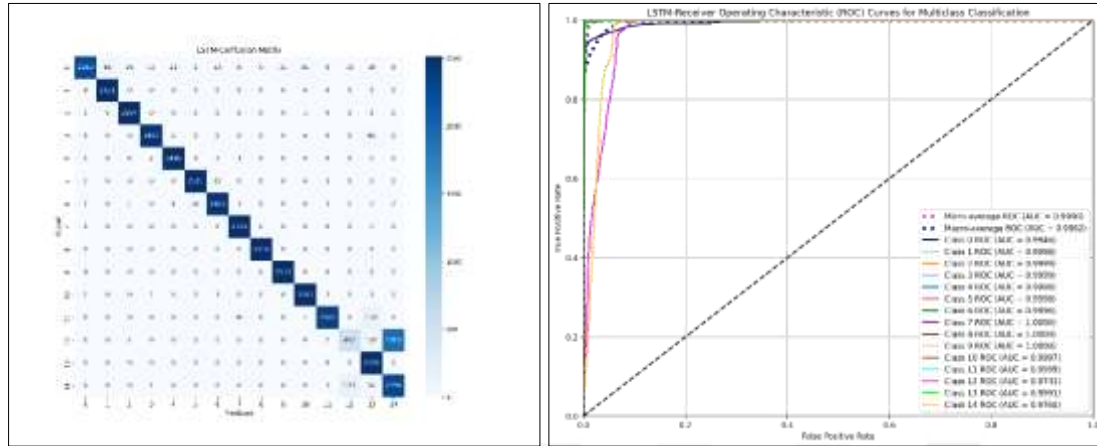


MLP

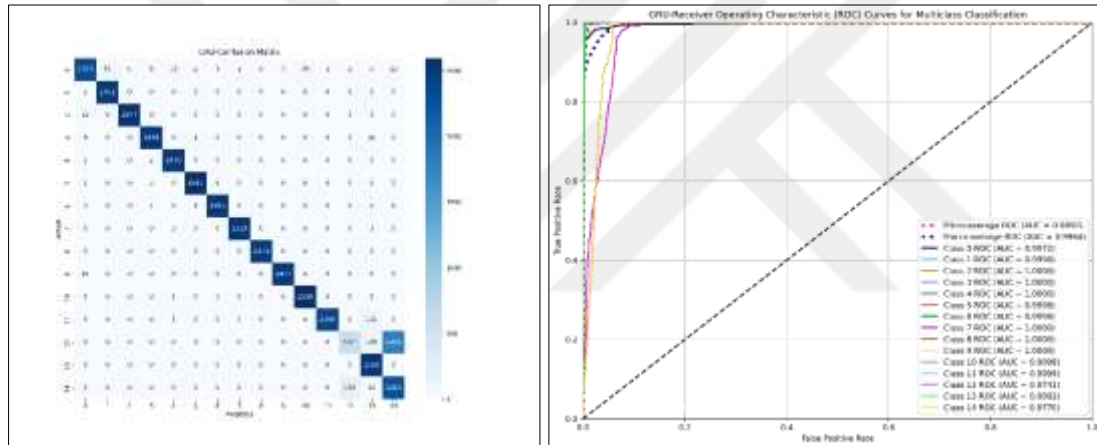


CNN

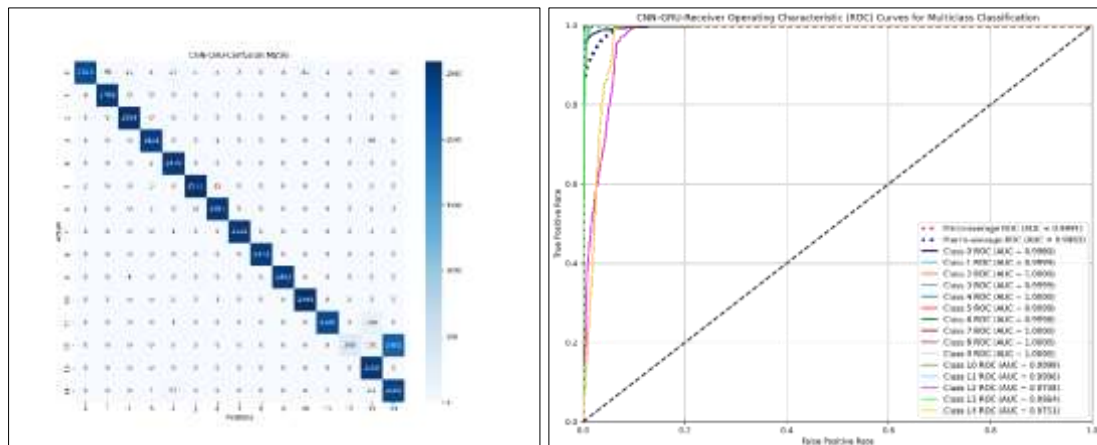
EK-3. (devam) CIC-IDS2017 veri kümesi çok sınıflı sınıflandırma karmaşıklık matrisleri ve ROC eğrileri



LSTM

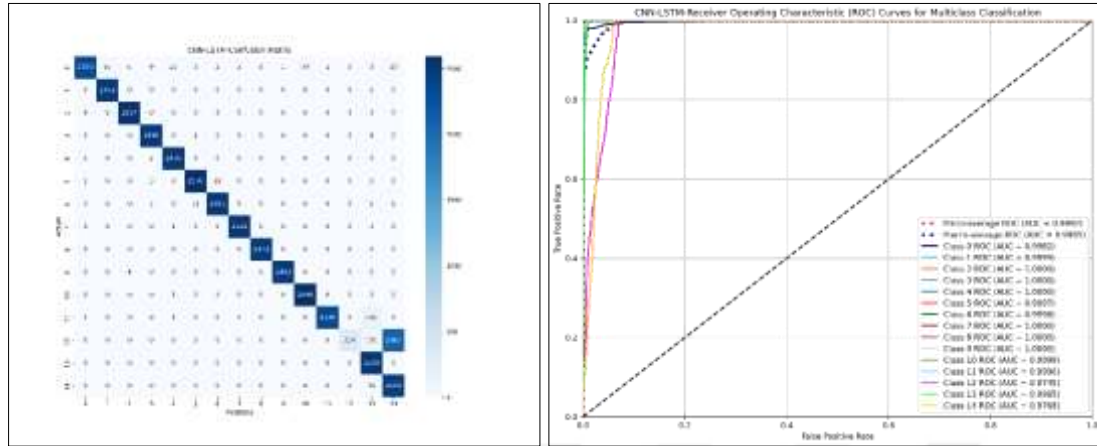


GRU

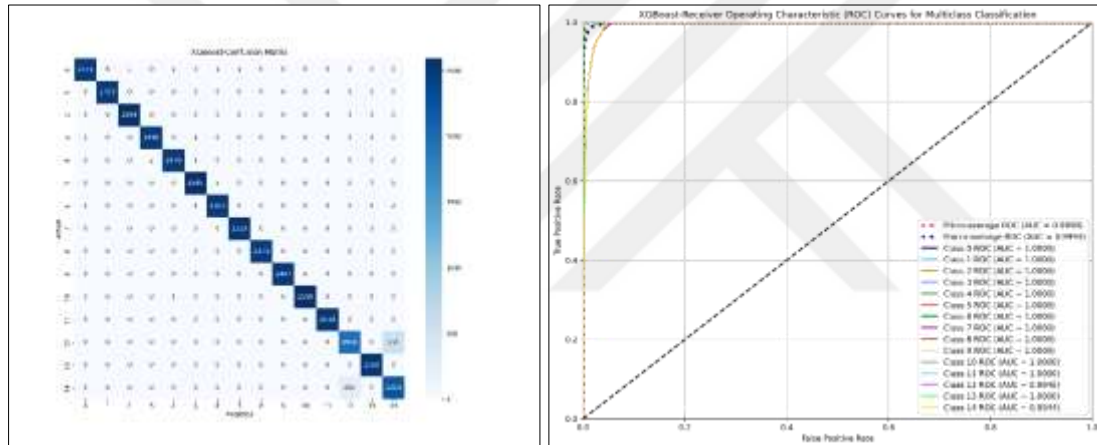


CNN-GRU

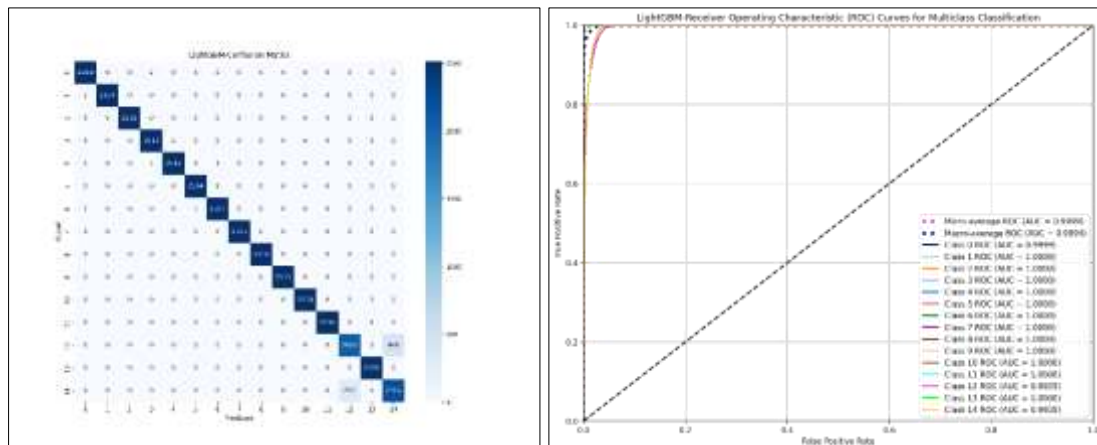
EK-3. (devam) CIC-IDS2017 veri kümesi çok sınıflı sınıflandırma karmaşıklık matrisleri ve ROC eğrileri



CNN-LSTM

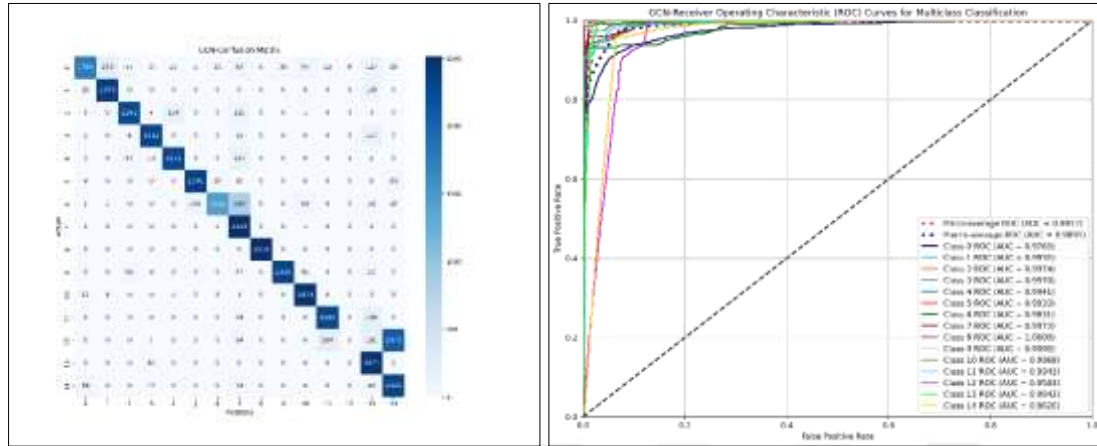


XGBoost

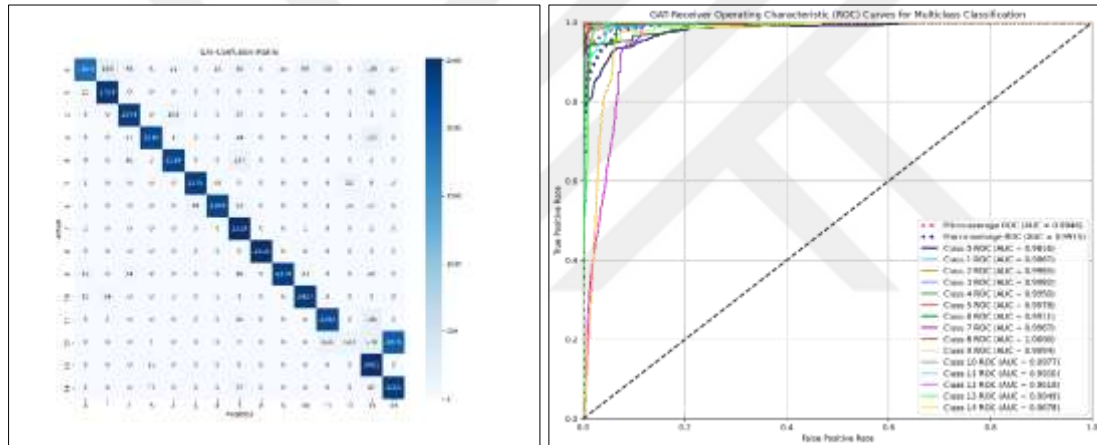


LightGBM

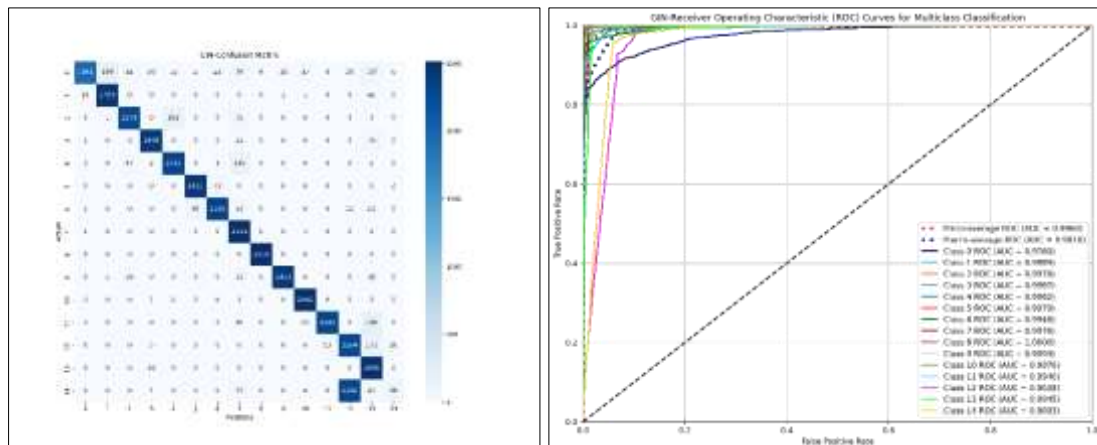
EK-3. (devam) CIC-IDS2017 veri kümesi çok sınıflı sınıflandırma karmaşıklık matrisleri ve ROC eğrileri



GCN

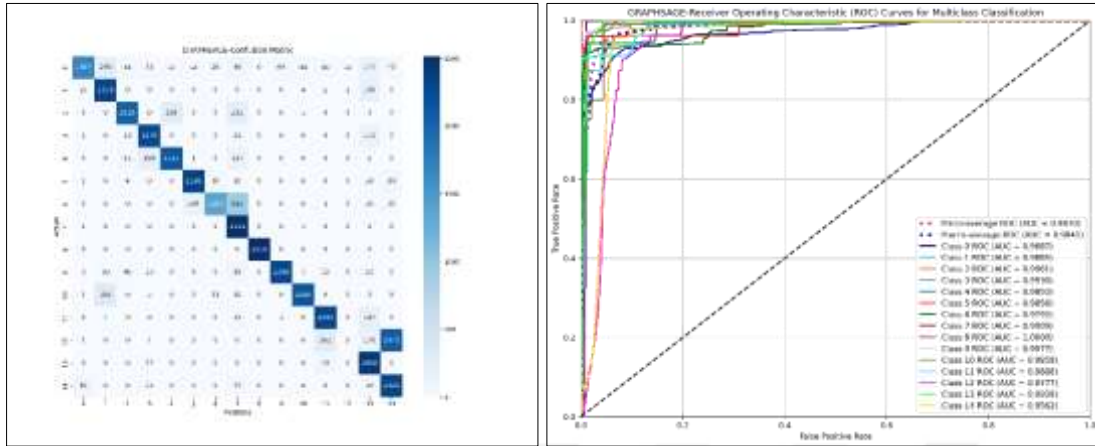


GAT

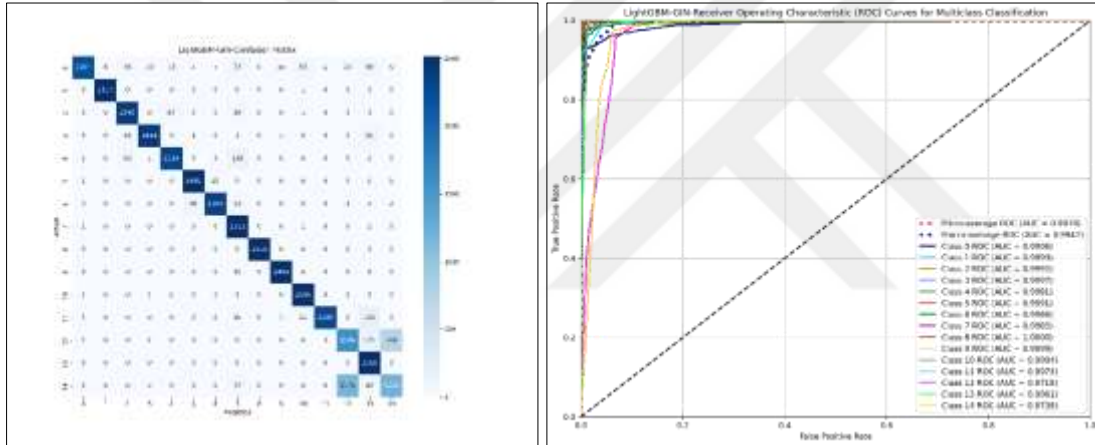


GIN

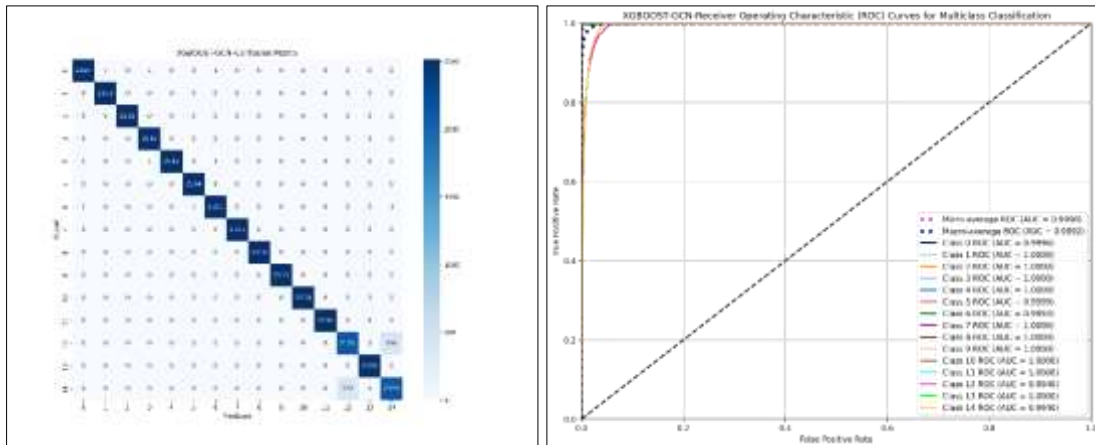
EK-3. (devam) CIC-IDS2017 veri kümesi çok sınıflı sınıflandırma karmaşıklık matrisleri ve ROC eğrileri



GraphSAGE

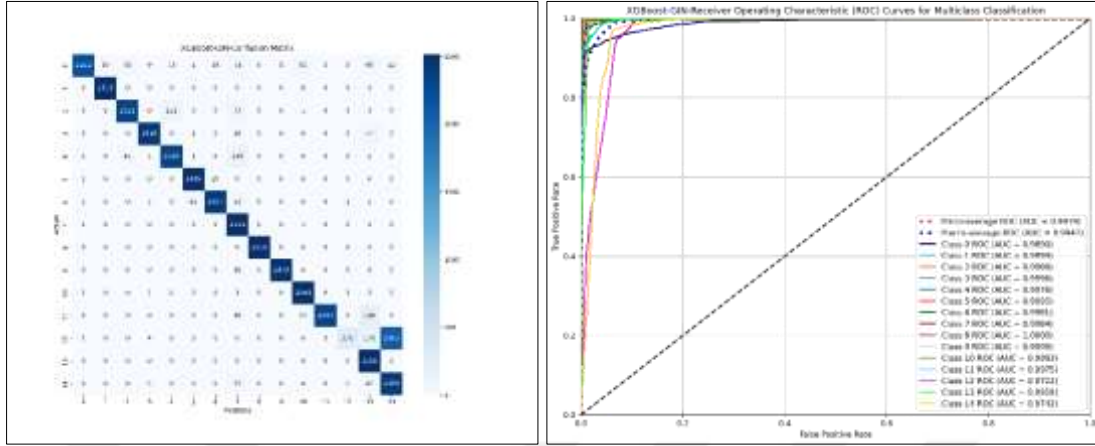


LightGBM-GIN

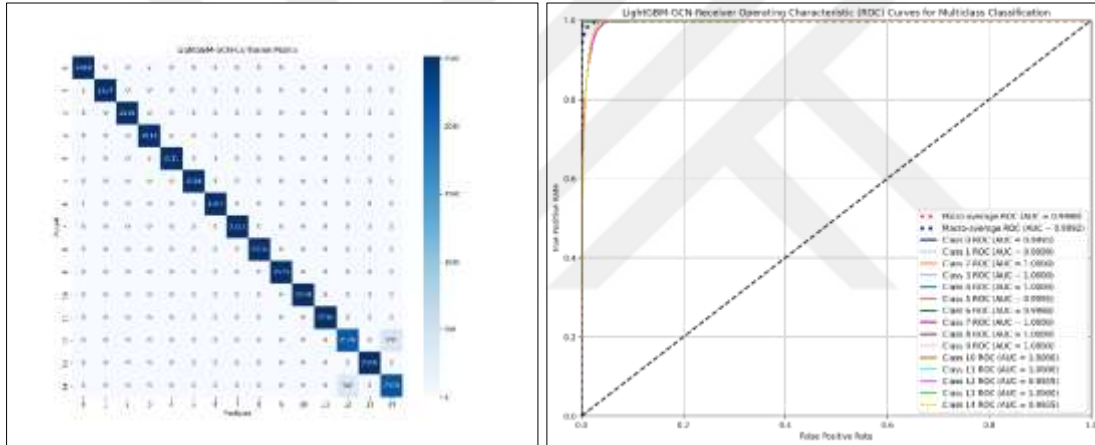


XGBoost-GCN

EK-3. (devam) CIC-IDS2017 veri kümesi çok sınıflı sınıflandırma karmaşıklık matrisleri ve ROC eğrileri

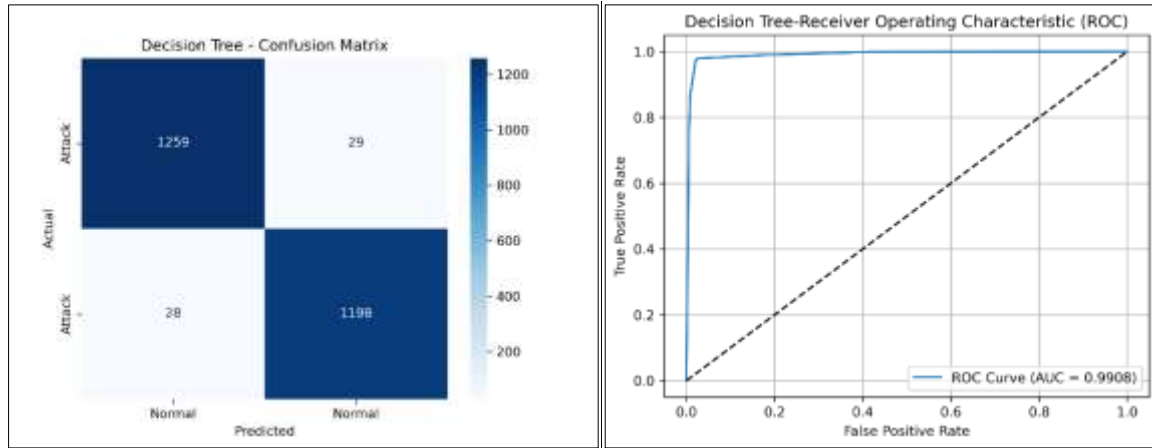


XGBoost-GIN

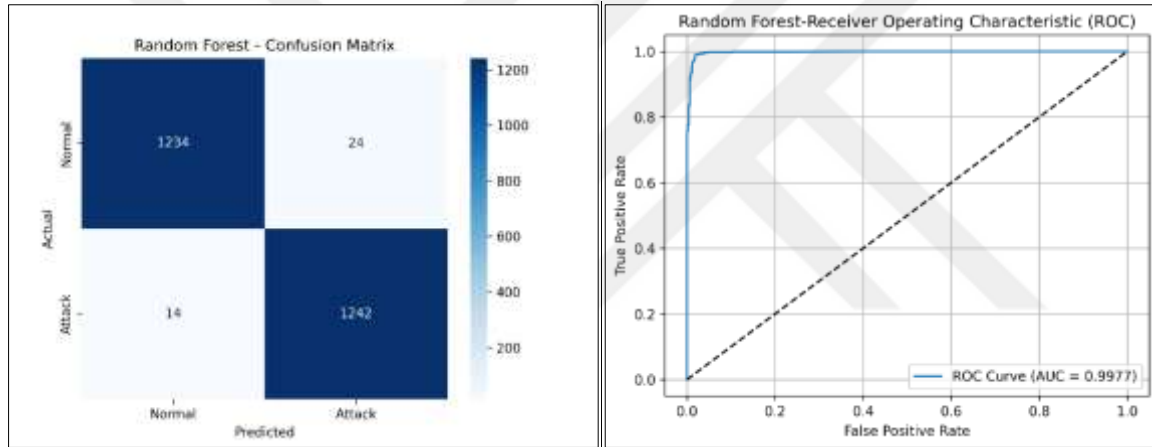


Önerilen Model

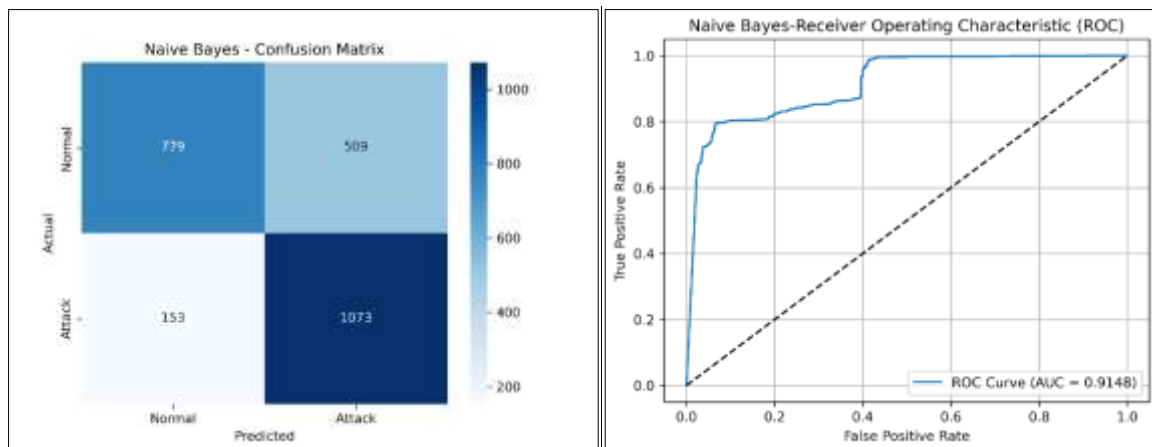
EK-4. CIC-IDS2017 veri kümesi ikili sınıflandırma karmaşıklık matrisleri ve ROC eğrileri



Decision Tree

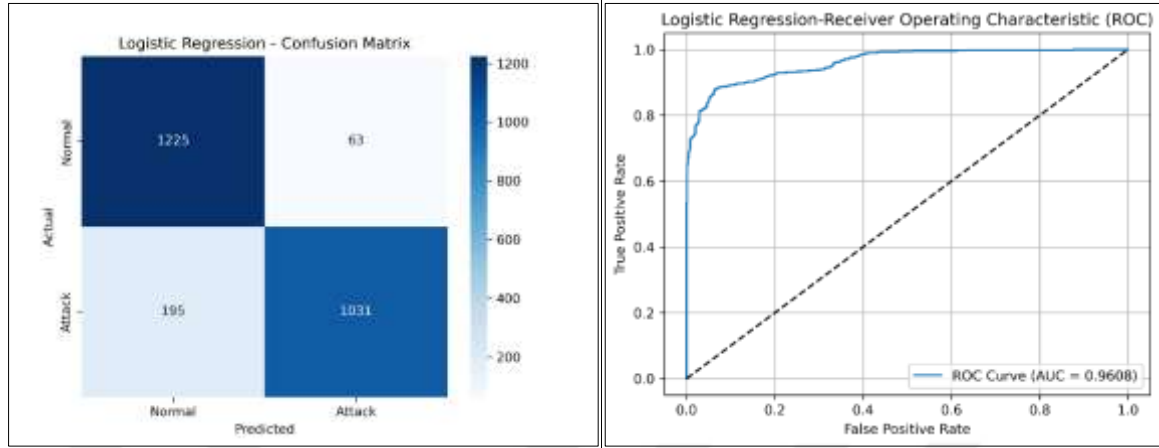


Random Forest

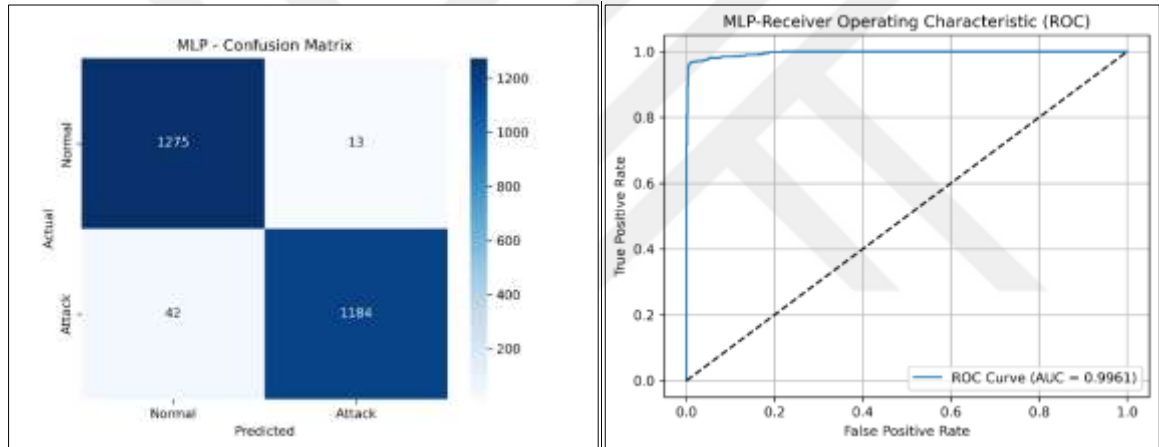


Naive Bayes

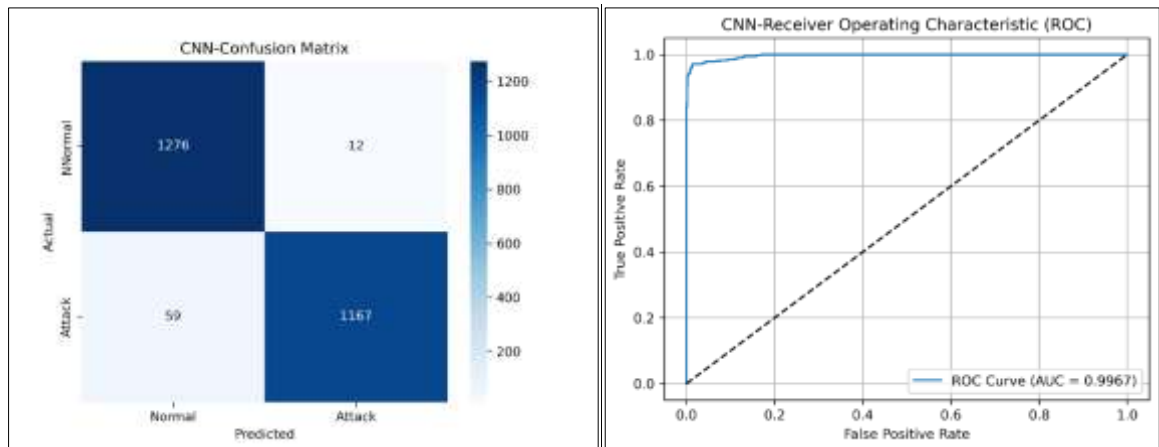
EK-4. (devam) CIC-IDS2017 veri kümesi ikili sınıflandırma karmaşıklık matrisleri ve ROC eğrileri



Logistic Regression

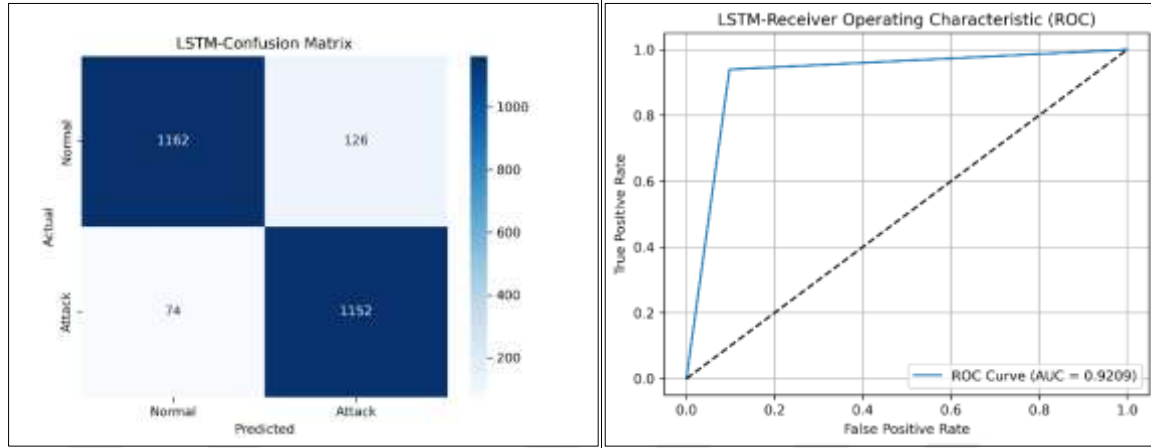


MLP

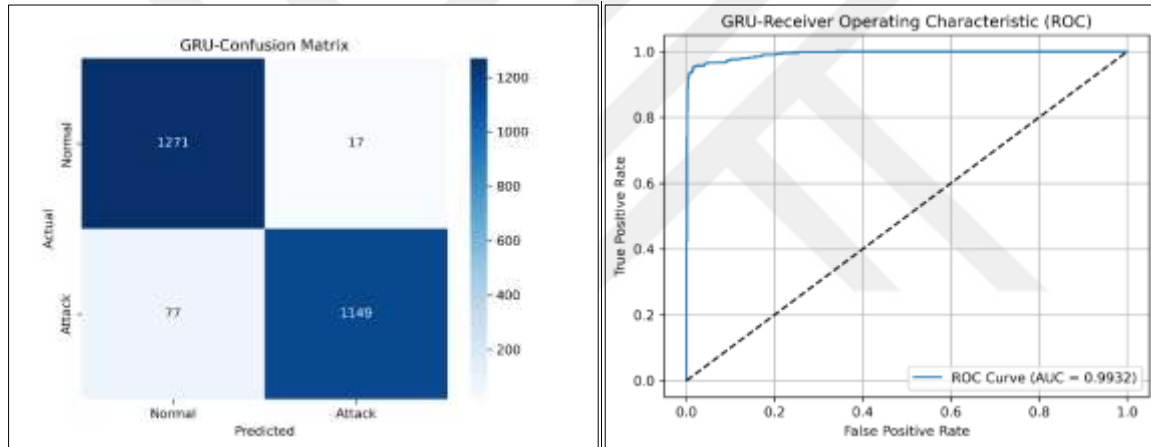


CNN

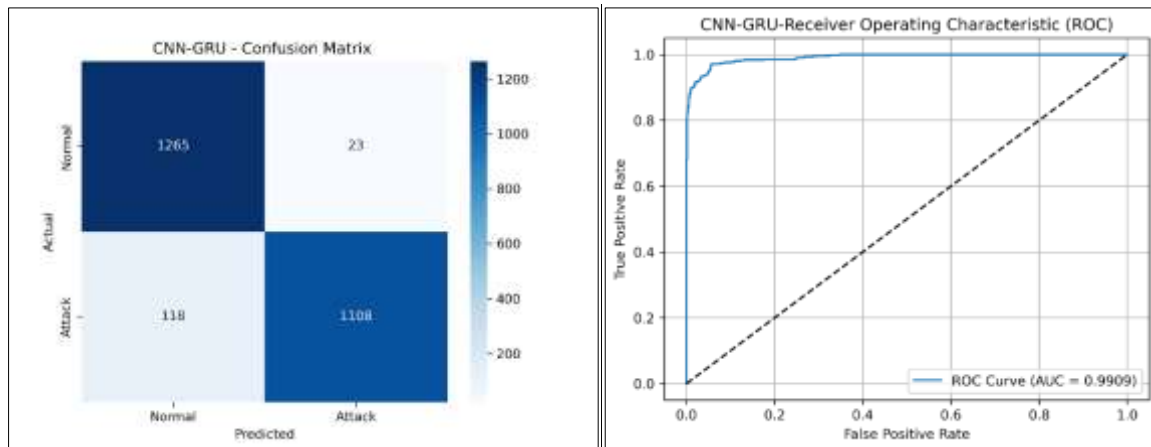
EK-4. (devam) CIC-IDS2017 veri kümesi ikili sınıflandırma karmaşıklık matrisleri ve ROC eğrileri



LSTM

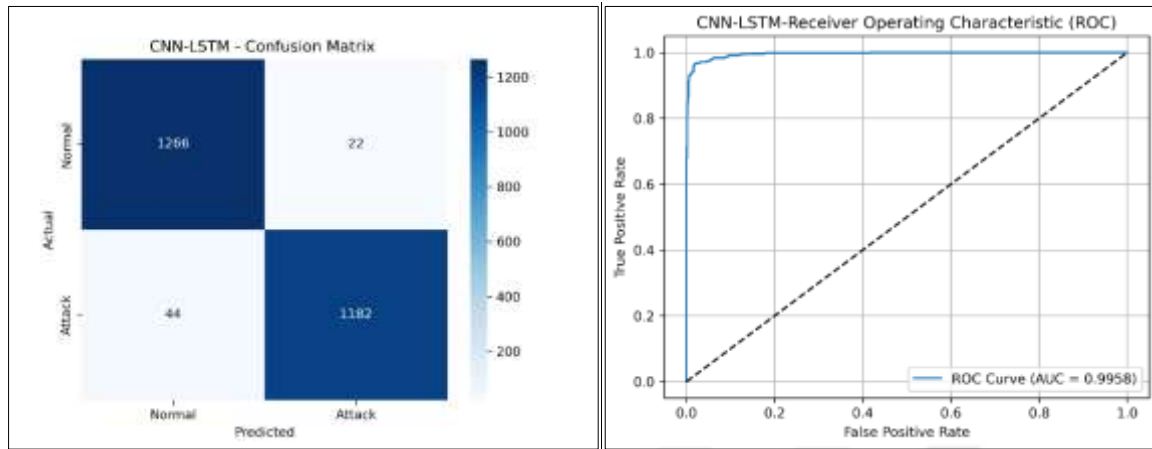


GRU

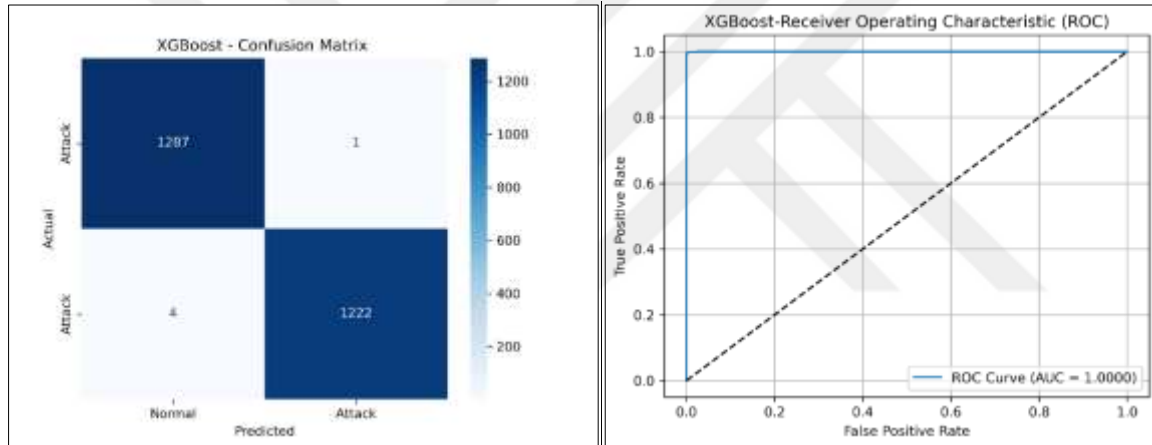


CNN-GRU

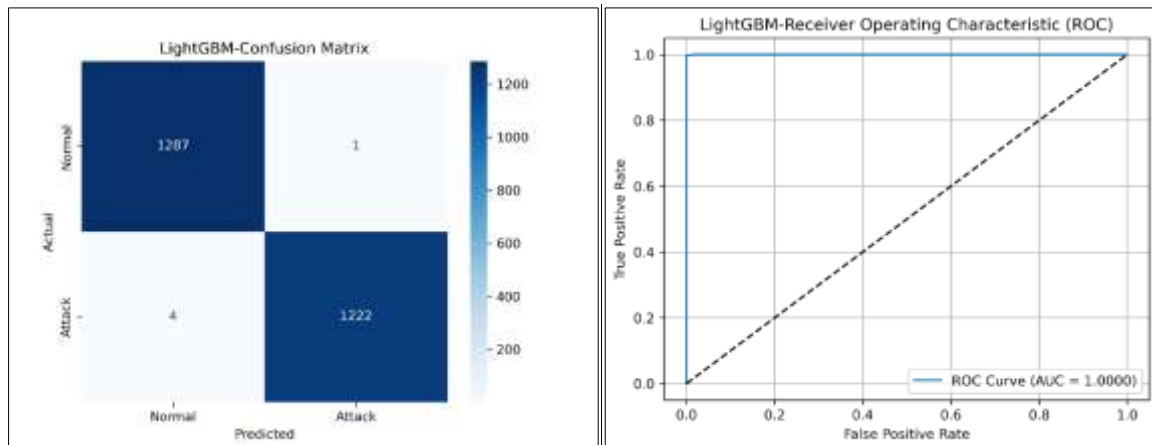
EK-4. (devam) CIC-IDS2017 veri kümesi ikili sınıflandırma karmaşıklık matrisleri ve ROC eğrileri



CNN-LSTM

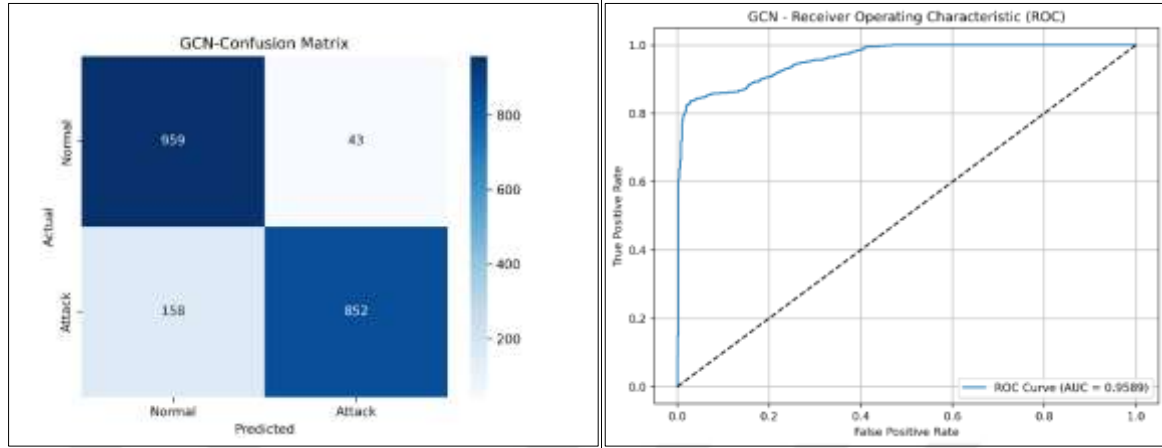


XGBoost

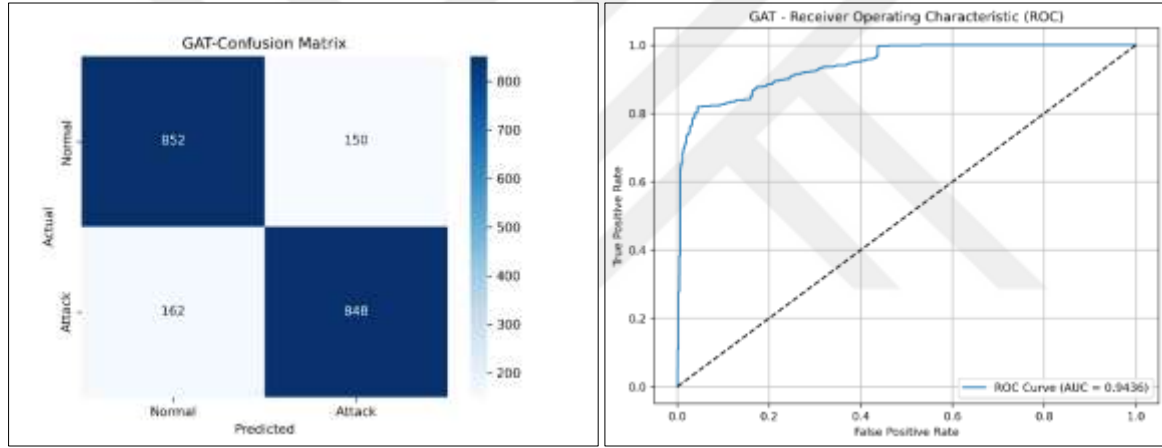


LightGBM

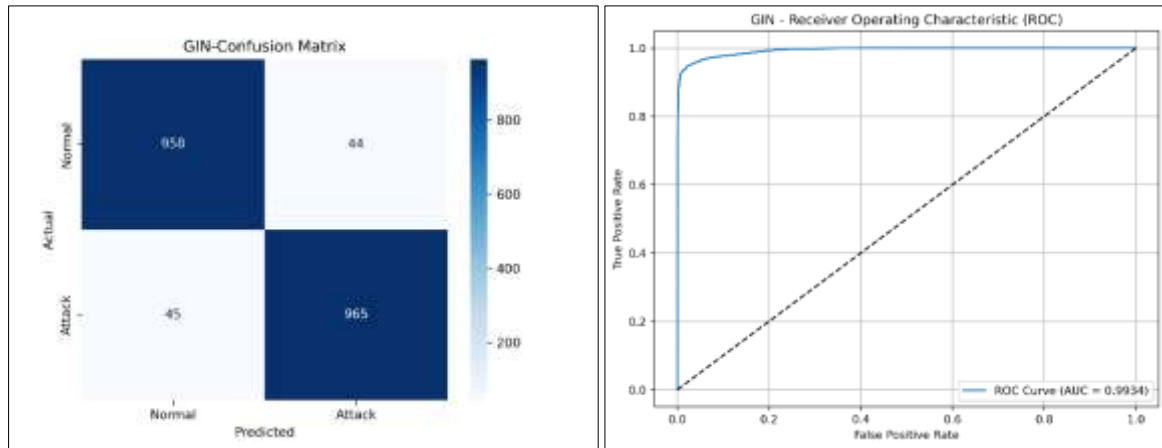
EK-4. (devam) CIC-IDS2017 veri kümesi ikili sınıflandırma karmaşıklık matrisleri ve ROC eğrileri



GCN

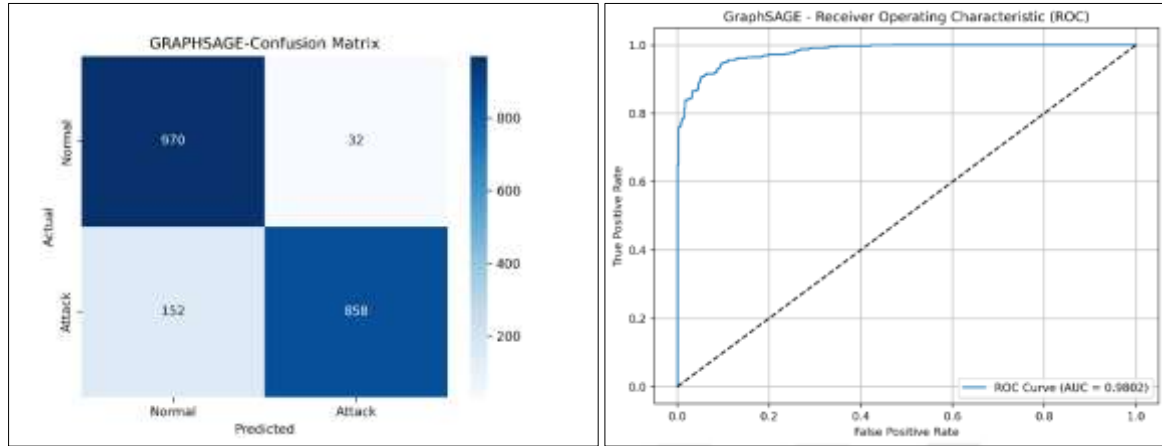


GAT

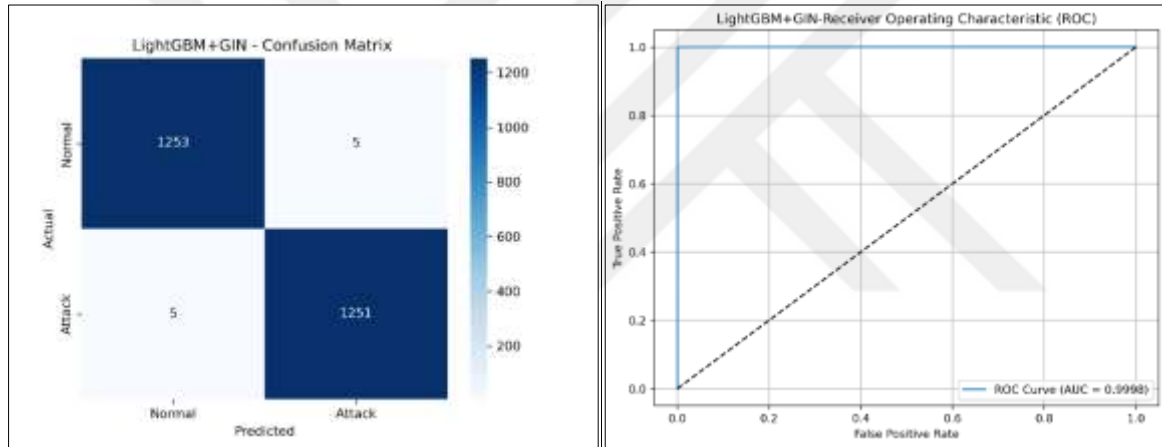


GIN

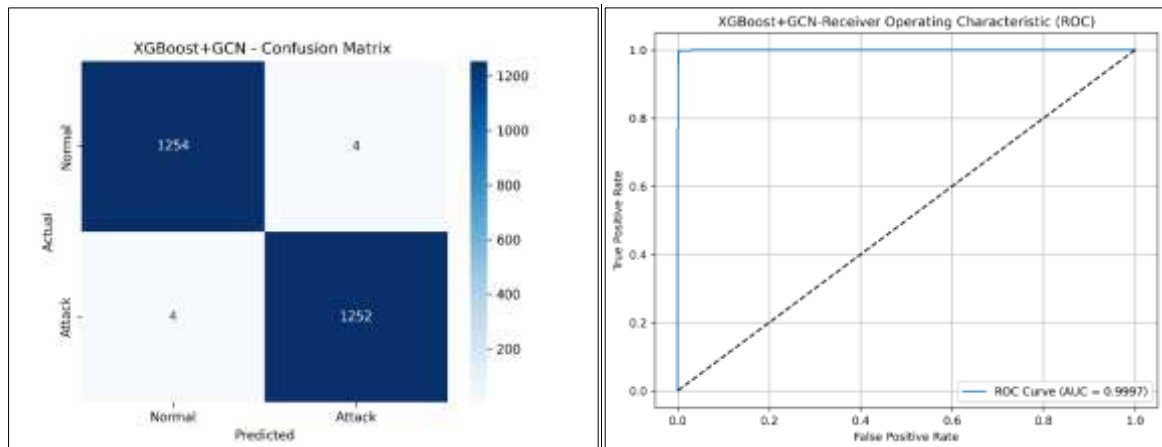
EK-4. (devam) CIC-IDS2017 veri kümesi ikili sınıflandırma karmaşıklık matrisleri ve ROC eğrileri



GraphSAGE

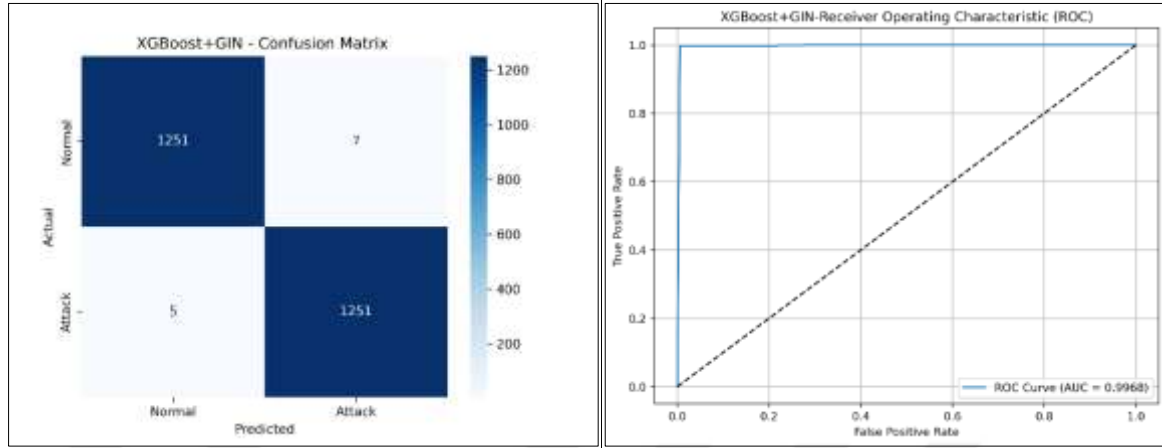


LightGBM-GIN

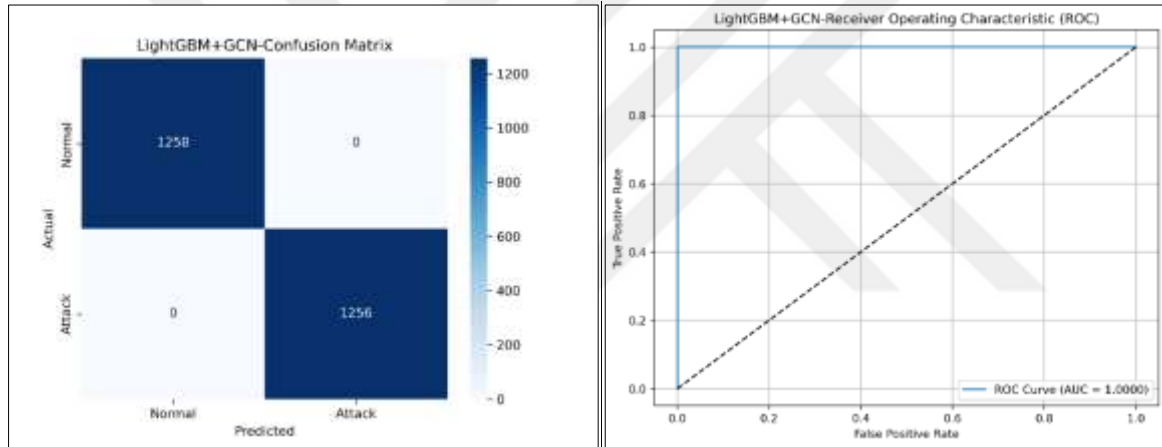


XGBoost-GCN

EK-4. (devam) CIC-IDS2017 veri kümesi ikili sınıflandırma karmaşıklık matrisleri ve ROC eğrileri



XGBoost+GIN



Önerilen Model



Gazili olmak ayrıcalıktır