

T.C.
FIRAT ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ



**SALDIRI TESPİT VE ENGELLEME SİSTEMLERİ İÇİN YAPAY
ZEKA TABANLI YENİ BİR GÜVENLİK MODELİNİN
OLUŞTURULMASI**

İlhan Fırat KILINÇER

Doktora Tezi

ELEKTRİK ELEKTRONİK MÜHENDİSLİĞİ ANABİLİM DALI

Elektrik Elektronik Mühendisliği Teknolojileri Programı

Telekomünikasyon Bilim Dalı

HAZİRAN 2022

T.C.
FIRAT ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

Elektrik Elektronik Mühendisliği Anabilim Dalı
Elektrik Elektronik Mühendisliği Teknolojileri Programı

Doktora Tezi

SALDIRI TESPİT VE ENGELLEME SİSTEMLERİ İÇİN YAPAY ZEKA TABANLI YENİ BİR GÜVENLİK MODELİNİN OLUŞTURULMASI

Tez Yazarı

İlhan Fırat KILINÇER

Danışman

Prof. Dr. Abdulkadir ŞENGÜR

İkinci Danışman

Doç. Dr. Fatih ERTAM

HAZİRAN 2022

ELAZIĞ

T.C.
FIRAT ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

Elektrik Elektronik Mühendisliği Anabilim Dalı
Elektrik Elektronik Mühendisliği Teknolojileri Programı

Doktora Tezi

Başlığı:	Saldırı Tespit ve Engelleme Sistemleri için Yapay Zeka Tabanlı Yeni Bir Güvenlik Modelinin Oluşturulması
Yazarı:	İlhan Fırat KILINÇER
İlk Teslim Tarihi:	29.04.2022
Savunma Tarihi:	08.06.2022

TEZ ONAYI

Fırat Üniversitesi Fen Bilimleri Enstitüsü tez yazım kurallarına göre hazırlanan bu tez aşağıda imzaları bulunan jüri üyeleri tarafından değerlendirilmiş ve akademik dinleyicilere açık yapılan savunma sonucunda OYBİRLİĞİ ile kabul edilmiştir.

Danışman:	Prof. Dr. Abdulkadir ŞENGÜR Fırat Üniversitesi, Teknoloji Fakültesi	<i>İmza</i> Onayladım
İkinci Danışman:	Doç. Dr. Fatih ERTAM Fırat Üniversitesi, Teknoloji Fakültesi	Onayladım
Başkan:	Prof. Dr. Melih Cevdet İNCE Malatya Turgut Özal Üniversitesi, Mühendislik ve Doğa Bilimleri Fakültesi	Onayladım
Üye:	Doç. Dr. Sedat AKLEYLEK On Dokuz Mayıs Üniversitesi, Mühendislik Fakültesi	Onayladım
Üye:	Doç. Dr. Ömer Faruk ALÇİN Malatya Turgut Özal Üniversitesi, Mühendislik ve Doğa Bilimleri Fakültesi	Onayladım

Bu tez, Enstitü Yönetim Kurulunun/...../20..... tarihli toplantısında tescillenmiştir.

İmza
Prof. Dr. Kürşat Esat ALYAMAÇ
Enstitü Müdürü

BEYAN

Fırat Üniversitesi Fen Bilimleri Enstitüsü tez yazım kurallarına uygun olarak hazırladığım “Saldırı Tespit ve Engelleme Sistemleri için Yapay Zeka Tabanlı Yeni Bir Güvenlik Modelinin Oluşturulması ” Başlıklı Doktora Tezimin içindeki bütün bilgilerin doğru olduğunu, bilgilerin üretilmesi ve sunulmasında bilimsel etik kurallarına uygun davrandığımı, kullandığım bütün kaynakları atıf yaparak belirttiğimi, maddi ve manevi desteği olan tüm kurum/kuruluş ve kişileri belirttiğimi, burada sunduğum veri ve bilgileri unvan almak amacıyla daha önce hiçbir şekilde kullanmadığımı beyan ederim.

08.06.2022

İlhan Fırat KILINÇER



ÖNSÖZ

Bu tez çalışması Fırat Üniversitesi Fen Bilimleri Enstitüsü Elektrik-Elektronik Mühendisliği Teknolojileri doktora programında hazırlanmıştır. Saldırı tespit sistemleri ve internet davranış analizi konularında literatürde birçok çalışma bulunmaktadır. Bu alanlara network ve güvenlik uzmanı açısından bakmak, yorumlamak, yeni veri setleri ve yöntemler eklemek, mevcut olanları geliştirmek için bu çalışmaya başlanmıştır.

Tez sürecinde tecrübeleriyle bana yol gösteren, ufkumu açan ve benden desteklerini esirgemeyen danışman hocalarım Prof. Dr. Abdulkadir ŞENGÜR'e ve Doç.Dr. Fatih ERTAM'a sonsuz teşekkürlerimi sunarım.

Verdikleri destek ve ileriye dönük fikirleri için Doç. Dr. Türker TUNCER'e ve Dr. Öğr. Üyesi Orhan YAMAN'a teşekkür ederim. Ayrıca bütün zorluklara rağmen beni sürekli destekleyen aileme teşekkürlerimi sunarım.

Bu doktora tezinin saldırı tespit sistemleri ve internet davranış analizi alanlarında araştırmacılara katkı sunmasını dilerim.

İlhan Fırat KILINÇER
ELAĞ, 2022

İÇİNDEKİLER

	Sayfa
ÖNSÖZ.....	iv
İÇİNDEKİLER	v
ÖZET	vii
ABSTRACT	viii
ŞEKİLLER LİSTESİ	ix
TABLOLAR LİSTESİ	xi
SİMGELER VE KISALTMALAR	xiii
1. GİRİŞ	1
1.1. Gelişmiş Bir Siber Saldırının Uygulama Adımları	3
1.1.1. Keşif (Reconnaissance)	4
1.1.2. Silahlanma (Weaponization)	4
1.1.3. İletme (Delivery)	4
1.1.4. Sömürme (Exploitation)	4
1.1.5. Yükleme (Installation)	4
1.1.6. Komuta ve Kontrol (Command & Control, C2)	4
1.1.7. Eyleme Geçme (Actions on Objectives)	4
1.2. Siber Saldırı Çeşitleri	5
1.2.1. Zafiyet Taraması	5
1.2.2. MAC Adres Taşması	6
1.2.3. DHCP Starvation Saldırısı	7
1.2.4. ARP Zehirlleme Saldırısı	7
1.2.5. Kaba Kuvvet Saldırısı	7
1.2.6. Hizmet Reddi Saldırısı	7
1.2.7. İstismar Saldırıları	8
1.2.8. Fuzzing Saldırısı	8
1.2.9. HeartBleed Saldırısı	9
1.2.10. SQL Enjeksiyonu Saldırısı	9
1.3. Saldırı Tespit Sistemleri	9
1.3.1. STS Veri Setleri	11
1.3.2. Saldırı Tespit Sistemleri Literatür Çalışmaları	16
1.4. Makine Öğrenmesi ve Yöntemleri	24
1.4.1. Destek Vektör Makineleri (Support Vector Machine - SVM)	24
1.4.2. K- En Yakın Komşu Sınıflandırıcısı	25
1.4.3. Karar Ağaçları Sınıflandırıcısı	26
1.4.4. Rastgele Orman Sınıflandırıcısı	26
1.4.5. Ekstra Ağaçlar Sınıflandırıcısı	27
1.4.6. Bagged Tree Sınıflandırıcısı	28
1.4.7. AdaBoost Sınıflandırıcısı	28
1.4.8. Gradient Boosting Machine (GBM) Sınıflandırıcısı	30
1.4.9. EXtreme Gradient Boosting (XGBoost) Sınıflandırıcısı	30
1.4.10. Light GBM (LGBM) Sınıflandırıcısı	32
1.4.11. Category Boosting (CatBoost) Sınıflandırıcısı	33
1.4.12. Özyinelemeli Özellik Eleminasyonu	34

1.4.13. Lojistik Regresyon	34
1.4.14. Çok Katmanlı Algılayıcı	35
2. SİBER GÜVENLİK SALDIRI TESPİTİ İÇİN MAKİNE ÖĞRENMESİ YÖNTEMLERİ: VERİ SETLERİ VE KARŞILAŞTIRMALI ANALİZ	36
2.1. Materyal ve Metod	36
2.2. Bulgular ve Tartışma	39
2.3. Sonuçlar	46
3. BOOSTING METODLARINI KULLANAN YENİ BİR SALDIRI TESPİT ÇERÇEVESİ	47
3.1. Materyal ve Metod	47
3.1.1. Genel Veri Setleri	47
3.1.2. CCiDD Veri Seti	48
3.1.3. Önerilen Metod	55
3.2. Bulgular	62
3.3. Tartışma	65
3.4. Sonuçlar	66
4. SPA-IDS: VMD VE YİNELEMELİ ÖZELLİK SEÇİMİNE DAYALI AKILLI BİR SALDIRI TESPİT SİSTEMİ	68
4.1. Materyal ve Metod	68
4.1.1. SPA-IDS Veri Seti	68
4.1.2. Önerilen Metod	72
4.2. Bulgular	77
4.3. Sonuçlar	82
5. İNTERNET KULLANICILARININ DAVRANIŞ ANALİZİ	83
5.1. Materyal ve Metod	83
5.1.1. IoBe Veri Seti	84
5.1.2. CICDarknet 2020 Veri Seti	84
5.1.3. Önerilen Metod	85
5.2. Bulgular	90
5.3. Sonuçlar	92
6. ÖZYİNELEMELİ ÖZELLİK ELEME VE ÇOK KATMANLI ALGILAYICI OPTİMİZASYONU İLE SAĞLIK SİSTEMLERİNDE OTOMATİK SALDIRI TESPİTİ	94
6.1. Materyal ve Metod	94
6.2. Bulgular	98
6.3. Tartışma	100
6.4. Sonuçlar	101
7. SONUÇLAR	102
ÖNERİLER	104
KAYNAKLAR	105
ÖZGEÇMİŞ	

ÖZET

Saldırı Tespit ve Engelleme Sistemleri için Yapay Zeka Tabanlı Yeni Bir Güvenlik Modelinin Oluşturulması

İlhan Fırat KILINÇER

Doktora Tezi

FIRAT ÜNİVERSİTESİ

Fen Bilimleri Enstitüsü

Elektrik Elektronik Mühendisliği Anabilim Dalı

Elektrik Elektronik Mühendisliği Teknolojileri Programı

Haziran 2022, Sayfa: xvii + 115

Günümüzdeki teknolojik gelişmeler ile beraber siber saldırıların hem çeşitliliği hem de sayıları sürekli artmaktadır. Siber saldırıları önlemede en yaygın kullanılan teknolojilerden biri de Saldırı Tespit Sistemleri (STS)'dir. Bu tez çalışmasında STS sistemlerinin performansını iyileştirmeye yönelik çalışmalar yapılmıştır. Çalışmanın ilk bölümünde literatürdeki ilgili çalışmalar incelendikten sonra, ikinci bölümünde literatürde sıklıkla kullanılan STS veri setleri ele alınmıştır. Ele alınan veri setleri önerilen makine öğrenmesi metodları ile sınıflandırılmıştır. Çalışmanın üçüncü bölümünde, literatürdeki STS veri setlerinin eksiklerini gidermek amacıyla Comprehensive Cyber Security Intrusion Detection Dataset (CCiDD) veri seti oluşturulmuştur. Bu bölümde kullanılan tüm veri setleri için özellik seçimi ve hiper parametre optimizasyon işlemleri uygulanarak kullanılan sınıflandırıcıların performansları artırılmıştır. Çalışmanın dördüncü bölümünde, anahtarlar ve cihazlardaki anomali artışları ele alınarak yeni bir Switch port Anomaly based Intrusion Detection System (SPA-IDS) adı verilen veri seti oluşturulmuştur. Oluşturulan veri seti önerilen Dikey Mod Dönüşümü (vertical mode decomposition - VMD) ve Yinelemeli Komşuluk Bileşen Analiz (iterative neighborhood component analysis - INCA) adımlarından geçirildikten sonra önerilen makine öğrenmesi metodları ile sınıflandırılmıştır. Çalışmanın beşinci bölümünde, kullanıcıların internet davranışlarını analiz etmek amacıyla Internet of Behavior (IoBe) olarak adlandırılan veri seti oluşturulmuştur. Oluşturulan IoBe veri setinin en iyi özellikleri, dört adet havuzlama işleminden sonra ReliefF özellik seçicisi ile seçilmiş ve Bagged Tree algoritması ile sınıflandırılmıştır. Çalışmanın son ve altıncı bölümünde, tıbbi nesnelerin interneti (Internet of Medical Things - IoMT) kavramı ele alınmış ve bu nesnelere yönelik saldırıları tespit etmek için özellik seçimi ve hiper parametre optimizasyonu adımlarından oluşan yeni bir model önerilmiştir. Bu tez çalışmasında, saldırı tespit sistemlerine daha geniş bir perspektif ile bakılmış ve STS sistemlerinin geliştirilmesine yönelik yeni yaklaşımlar ve veri setleri önerilmiştir. Ayrıca, kullanıcıların internet davranışlarını analiz edebilmek için de yeni bir veri seti oluşturulmuştur. Böylece, oluşturulan veri setleri ve önerilen yaklaşımlar ile literatüre katkı sağlanmıştır.

Anahtar Kelimeler: Saldırı tespit sistemleri, Makine öğrenmesi, Hizmet reddi saldırıları, Kaba kuvvet saldırıları, Zafiyet taraması

ABSTRACT

Building a New Artificial Intelligence Based Security Model for Intrusion Detection and Prevention Systems

İlhan Fırat KILINÇER

Ph.D. Thesis

FIRAT UNIVERSITY

Graduate School of Natural and Applied Sciences

Department of Electrical and Electronics Engineering

Electrical and Electronics Engineering Technologies

June 2022, Pages: xvii + 115

With today's technological developments, both the variety and the number of cyber-attacks are constantly increasing. One of the most widely used technologies to prevent cyber-attacks is Intrusion Detection Systems (IDS). In this thesis, studies have been carried out to improve the performance of IDS systems. After examining the related studies in the literature in the first section, STS datasets that are frequently used in the literature are discussed in the second section. Considered datasets are classified by proposed machine learning methods. In the third section, the Comprehensive Cyber Security Intrusion Detection Dataset (CCiDD) dataset was created in order to eliminate the deficiencies of the STS datasets in the literature. The performance of the classifiers has been increased by applying feature selection and hyperparameter optimization processes for all datasets used in this section. In the fourth section, a new dataset called Switch port Anomaly based Intrusion Detection System (SPA-IDS) was created by considering the anomaly increases in switching devices. The created dataset was classified with the recommended machine learning methods after going through the proposed vertical mode decomposition (VMD) and iterative neighborhood component analysis (INCA) steps. In the fifth section, a dataset called Internet of Behavior (IoBe) was created in order to analyze the internet behavior of users. The best features of the created IoBe dataset were selected with the ReliefF feature selector after four pooling operations and then classified with the Bagged Tree algorithm. In the last and sixth section, the concept of Internet of Medical Things (IoMT) is discussed and a new model consisting of feature selection and hyperparameter optimization steps is proposed to detect attacks against these objects. In this thesis, intrusion detection systems are looked at from a broader perspective and new approaches and datasets for the development of STS systems are proposed. In addition, a new dataset was created to analyze the internet behavior of users. Thus, a contribution to the literature has been made with the created datasets and the suggested approaches.

Keywords: Intrusion detection systems, Machine learning, Denial of service attacks, Brute force attacks, Vulnerability scanning

ŞEKİLLER LİSTESİ

	Sayfa
Şekil 1.1. Datareportal 2022 internet kullanım istatistikleri	1
Şekil 1.2. Yıllara göre 3. çeyrekte bildirilen ihlallerin sayısı	2
Şekil 1.3. Yıllara göre 3. çeyrekte bildirilen kayıp kayıtların sayısı.....	2
Şekil 1.4.Yıllara göre siber güvenlik pazarı	3
Şekil 1.5. Lockheed Martin siber ölüm zinciri modeli	3
Şekil 1.6.STS sistemlerinin genel yapısı	10
Şekil 1.7.Veri setleri ve sınıfları.....	11
Şekil 1.8.Destek vektör makineleri.....	25
Şekil 1.9. KNN algoritması	26
Şekil 1.10.Karar ağaçları algoritması	26
Şekil 1.11. Rastgele orman algoritması	27
Şekil 1.12. Ekstra Ağaçlar algoritması	28
Şekil 1.13. Adaboost algoritması.....	29
Şekil 1.14. XGBoost algoritması.....	31
Şekil 1.15. XGBoost level-wise büyüme yöntemi.....	33
Şekil 1.16. LGBM leaf-wise büyüme yöntemi.....	33
Şekil 1.17. RFE algoritması.....	34
Şekil 2.1. Önerilen yöntemin akış diyagramı	37
Şekil 3.1. CCiDD veri seti için oluşturulan ağ topolojisi	49
Şekil 3.2. CCiDD_A veri seti veri dağılımı.....	52
Şekil 3.3. CCiDD_B veri seti veri dağılımı	52
Şekil 3.4. CICFlowMeter arayüzü.....	53
Şekil 3.5. Önerilen yöntemin akış şeması.....	56
Şekil 3.6. Extra Tree algoritmasından sonra seçile en iyi özellikler ve önem dereceleri.....	58
Şekil 3.7. Her veri seti için alınan başarımların değerlerinin görsel karşılaştırması.....	61
Şekil 3.8. Tüm veri setleri için alınan karmaşıklık matrisleri.....	64
Şekil 4.1. SPA-IDS ağ topolojisi.....	69
Şekil 4.2. SPA-IDS saldırı karakteristikleri.....	72
Şekil 4.3.Önerilen VMD ve INCA tabanlı STS sistemi	73
Şekil 4.4.Toplanan paket sinyali ve VMD katsayılarının bir örneği	74
Şekil 4.5. INCA sonrası seçilen optimum özellik sayısı.....	77

Şekil 4.6. 10 sınıflı SPA-IDS veri seti için sınıflandırıcıların karmaşıklık matrisleri	79
Şekil 4.7. 10 sınıflı SPA-IDS veri seti için BT sınıflandırıcısının çapraz doğrulama grafiği	80
Şekil 4.8. Ana saldırı kategorileri için özellik seçimi	80
Şekil 4.9. 6 sınıflı SPA-IDS veri seti için sınıflandırıcıların karmaşıklık matrisleri	81
Şekil 4.10. 6 sınıflı SPA-IDS veri seti için BT sınıflandırıcısının çapraz doğrulama grafiği	82
Şekil 5.1. Önerilen yöntemin akış diyagramı	86
Şekil 5.2. Önerilen sınıflandırma modelinin akış diyagramı.	87
Şekil 5.3. Sınıf bazlı elde edilen sonuçlar	91
Şekil 5.4. Önerilen modelin çapraz doğrulama sonuçları	91
Şekil 5.5. Yinelemeli BT kullanılarak hesaplanmış karmaşıklık matrisleri	92
Şekil 6.1. Önerilen yöntemin akış şeması	95
Şekil 6.2. MLP sınıflandırıcısı sonucu karmaşıklık matrisleri	99

TABLÖLAR LİSTESİ

	Sayfa
Tablo 1.1. Saldırı tespit veri setleri	15
Tablo 1.2. Saldırı tespit sistemleri literatür özeti	18
Tablo 1.3. Adaboost algoritması sözde kodu	30
Tablo 2.1. Sınıf bazlı veri miktarları	38
Tablo 2.2. Gerçek değerler ve tahmin değerleri arasındaki ilişki.....	39
Tablo 2.3. Sınıflandırıcı sonuçları.....	40
Tablo 2.4. CSE-CIC-IDS2018 veri seti karşılaştırmalı analiz	41
Tablo 2.5. UNSW-NB15 veri seti karşılaştırmalı analiz	42
Tablo 2.6. NSL-KDD veri seti karşılaştırmalı analiz.....	43
Tablo 2.7. CIDD-001 veri seti karşılaştırmalı analiz	44
Tablo 2.8. ISCX2012veri seti karşılaştırmalı analiz	45
Tablo 3.1. Kullanılan genel veri setlerinin özellikleri.....	48
Tablo 3.2. CCiDD_A ve CCiDD_B veri seti özellikleri	50
Tablo 3.3. Önerilen CCiDD_A ve CCiDD_B veri setleri.....	51
Tablo 3.4. CICFlowMeter çıktısının özellikleri	53
Tablo 3.5. Sınıflandırma sonuçları	58
Tablo 3.6. Hiper parametre optimizasyonu sonucu elde edilen parametreler	61
Tablo 3.7. Hiper parametre optimizasyonundan sonra veri setleri performansları	62
Tablo 3.8. NSL-KDD veri seti için karşılaştırmalı analiz.....	65
Tablo 3.9. UNSW-NB15 veri seti için karşılaştırmalı analiz.....	65
Tablo 3.10. CSE-CIC-IDS2018 veri seti için karşılaştırmalı analiz	66
Tablo 4.1. SPA-IDS veri seti için kullanılan saldırı senaryoları ve özellikleri	69
Tablo 4.2. Özellik çıkarımı için kullanılan istatistiksel momentler	74
Tablo 4.3. INCA algoritması.....	76
Tablo 4.4. Kullanılan KNN, SVM, DT ve BT sınıflandırıcılarının özellikleri.	78
Tablo 4.5. Önerilen yöntemin 10 sınıflı SPA-IDS veri seti ile performansı	78
Tablo 4.6. Önerilen yöntemin 6 sınıflı SPA-IDS veri seti ile performansı	81
Tablo 5.1. Önerilen IoBe veri seti özellikleri.....	84
Tablo 5.2. CICDarknet 2020 veri seti trafik sınıfları	85
Tablo 5.3. Mod işlevine dayalı çoğunluk oylaması	90
Tablo 5.4. Önerilen IoBe sınıflandırma modelinin hesaplanan sonuçları	90

Tablo 5.5. BT'nin sonuçları	92
Tablo 6.1. IoMT veri setleri en uygun özellikler	96
Tablo 6.2. KNN sınıflandırıcısı performans sonuçları	97
Tablo 6.3. Hiper parametre optimizasyonu sonucu seçilen MLP parametreleri	98
Tablo 6.4. MLP sınıflandırıcısı sonuçları	98
Tablo 6.5. IoMT veri setleri için önerilen yöntem ve literatür karşılaştırması	100



SİMGELER VE KISALTMALAR

Simgeler

w	: Normal vektörü
b	: Ofset değeri
M	: Ağaç sayısı
n_{min}	: Minimum örnek boyutu
$J^{(t)}$	: Amaç fonksiyonu
n	: Tahmin sayısı
Ω	: Düzenleştirme fonksiyonu
γ	: Karmaşıklık katsayısı
T	: Yaprak sayısı
α	: Ceza parametresi
v	: Puanlar vektörü
μ	: Ağırlık
x	: Giriş özellikleri
V_k	: k 'inci VMD katsayısı
p	: Tek boyutlu sinyal
$prob(.)$	: Olasılık hesaplama fonksiyonu
$aver$	: Ortalama değer
Ln	: Sinyal uzunluğu
Dn	: Ayırıştırılmış veriler
in_1	: input
VP^j	: Doğrulama tahmini

Kısaltmalar

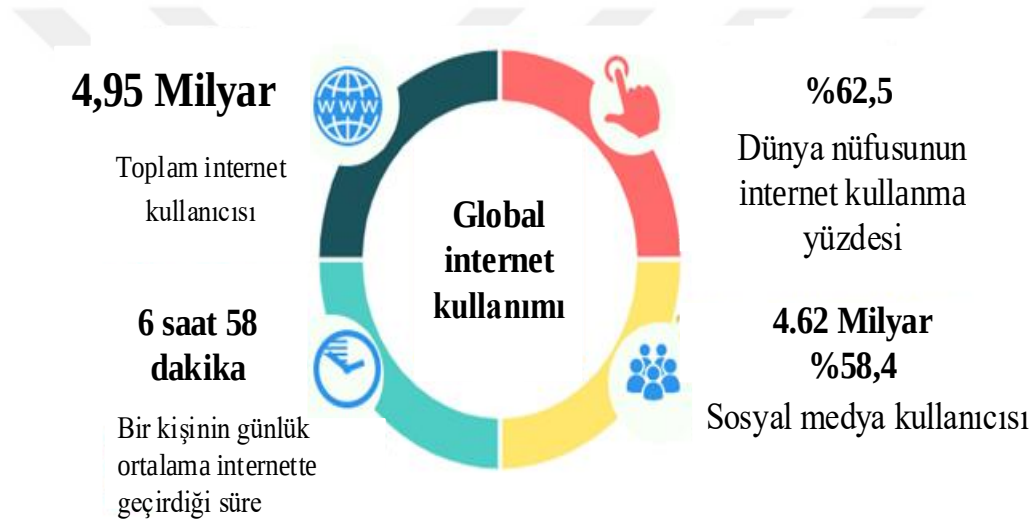
CMR	: Crystal Market Research
IoT	: Internet of Things
APT	: An Advanced Persistent Threat
TCP	: Transmission Control Protocol
UDP	: User Datagram Protocol
SYN	: Synchronize
ICMP	: Internet Control Message Protocol
MiTM	: Man in The Middle
MAC	: Media Access Control
DHCP	: Dynamic Host Configuration Protocol
DNS	: Domain Name Server
ARP	: Address Resolution Protocol
IP	: Internet Protocol
SSH	: Secure Shell
FTP	: File Transfer Protocol

RDP	: Remote Desktop
DoS	: Denial of Service
HTTP	: Hyper-Text Transfer Protocol
DDoS	: Distributed Denial of Service
SSL	: Secure Sockets Layer
OWASP	: Open Web Application Security Project
R2L	: Remote to Local
U2R	: User to Root
IDS	: Intrusion Detection Systems
STS	: Saldırı Tespit Sistemleri
CVE	: Common Vulnerabilities and Exposures
IT	: Information Technologies
OSI	: Open Systems Interconnection
XSS	: Cross Site Scripting
TO	: Tespit Oranı (Detection Rate - DR)
DO	: Doğruluk (Accuracy)
YAO	: Yanlış Alarm Oranı (False Alarm Rate - FAR)
FÖ	: F1-Ölçüsü
H	: Hassasiyet (Precision)
D	: Duyarlılık (D)
GO	: Geometrik Ortalama
TP	: Doğru Pozitif
TN	: Doğru Negatif
YP	: Yanlış Pozitif
YN	: Yanlış Negatif
Cmd	: Combined Metric
DNN	: Deep Neural Network
CNN	: Convolutional Neural Networks
RNN	: Recurrent Neural Networks
RBM	: Restricted Boltzmann Machine
DBN	: Deep Belief Network
DBM	: Deep Boltzmann Machine
DA	: Deep Auto-Encoder
MLP	: Multilayer Perceptron
NN	: Neural Network
SVM	: Support Vector Machine
DVM	: Destek Vektör Makineleri
DT	: Decision Tree
BT	: Bagged Tree
RF	: Random Forest
KNN	: K-Nearest Neighbors
NB	: Naive Bayes
ANN	: Artificial Neural Network
GBM	: Gradient Boosting Machine
LGBM	: Light Gradient Boosting Machine
XGBoost	: EXtreme Gradient Boosting
P2P	: Peer to Peer

LAN	: Local Area Network
VMD	: Vertical Mode Decomposition
INCA	: Iterative Neighborhood Component Analysis
CP	: Conditional Pooling
AP	: Average Pooling
MP	: Maximum Pooling
MAP	: Maximum Absolute Pooling
IoMT	: Internet of Medical Things
RFE	: Recursive Feature Elimination
MLP	: Multilayer Perceptron
LR	: Logistic Regression
NaN	: Boş Değer
MCLT	: Matlab Classification Learner
OA	: Overall Accuracy
OAP	: Overall Average Precision
UAR	: Unweighted Average Recall
CK	: Cohen's kappa
OF1	: overall F1
MCC	: Mathew Correlation Coefficient

1. GİRİŞ

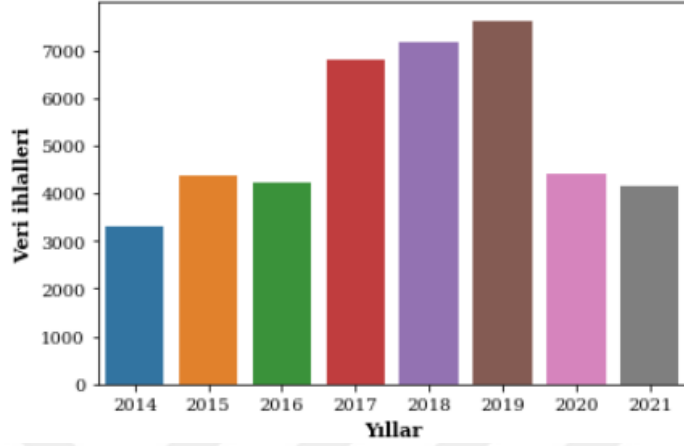
İnternet, bilgiye erişimi kolaylaştıran önemli bir teknolojidir. İnternet teknolojileri iletişim, reklam, seyahat, tatil, e-ticaret, bankacılık, sosyal medya, sağlık, eğitim gibi birçok alanda hayatımıza olumlu yenilikler getirmiştir. Son yıllarda özellikle nesnelerin interneti (Internet of Things - IoT) teknolojilerindeki gelişmeler ile birlikte interneti kullanan kişi ve uygulamaların sayısı artmaktadır. Dünyada internet kullanımı hakkında bilgi veren Datareportal verilerine göre internet kullanımı her geçen gün artmaktadır. Datareportal 2022 istatistiklerine göre dünya nüfusunun %62,5'i internet kullanıcısı, %58,4'ü sosyal medya kullanıcısıdır [1]. Şekil 1.1'de Datareportal istatistikleri verilmiştir.



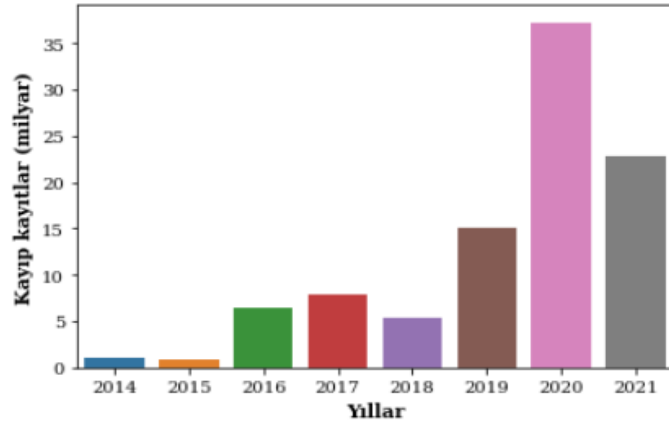
Şekil 1.1. Datareportal 2022 internet kullanım istatistikleri

Günümüzde internetin yaygın kullanımı ile birlikte siber güvenlik kavramı da önem kazanmıştır. Bilgi teknolojilerinin güvenliği olarak da bilinen siber güvenlik; bilgisayarları, ağları, verileri saldırılardan korumayı amaçlar. Kişiler açısından siber güvenlik; kişisel verilerinin gizliliğini sağlamaktır. Kurumlar için siber güvenlik kurum veri güvenliğini sağlamak, operasyonel süreçlerin devamlılığını sağlamak iken, hükümetler için siber güvenlik ise, vatandaşlarının ve kritik kurumlarının verilerini korumak demektir. İnternet kullanımının sürekli artması beraberinde birçok veri ihlallini de getirmektedir. RiskBased Security tarafından yayınlanan raporda, 2019 yılında çok büyük veri ihlallerinin gerçekleştiği görülmektedir. RiskBased Security'nin 2022'de yayınladığı veri ihlallerine yönelik raporu Şekil 1.2'de, sisteme girilmemiş kayıtların sayısı Şekil 1.3'te verilmiştir. Yayımlanan rapora göre 2020 ve 2021 yıllarındaki veri ihlallerinin sayısı 2019 yılındaki veri ihlallerinin sayısının yaklaşık %50 altındadır. Fakat yine RiskBased Security tarafından yayınlanan kayıp kayıt sayısı grafiğine göre 2020 ve 2021'de aşırı derecede kayıp kayıt

bulunmaktadır. Bunun nedeninin Covid-19 pandemisinin dikkatleri üzerine çekmesi olarak düşünülmektedir. Kayıp kayıtların sistemlere girilmesi durumunda 2019 yılı ile 2020 ve 2021 yılları arasındaki veri ihlalleri farkının kapanacağı düşünülmektedir [2].

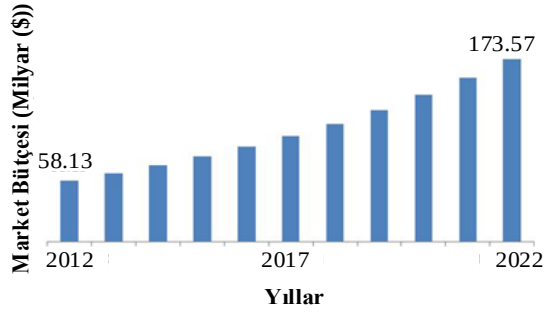


Şekil 1.2. Yıllara göre 3. çeyrekte bildirilen ihlallerin sayısı



Şekil 1.3. Yıllara göre 3. çeyrekte bildirilen kayıp kayıtların sayısı

Kurum ve kuruluşlar, kullanıcılarına güvenli ve sürekli bir hizmet sunabilmek için siber güvenlik teknolojilerine yönelik harcamalarını artırmaktadır. Crystal Market Research (CMR) tarafından yayınlanan rapora göre 2012 yılında yaklaşık 58,13 milyar ABD doları değerinde olan Siber Güvenlik Pazarının 2022 yılında 173,57 milyar ABD dolarına ulaşması beklenmektedir. CMR tarafından yayınlanan siber güvenlik pazarının yıllara göre büyüklüğü Şekil 1.4'teki grafikte verilmiştir [3].

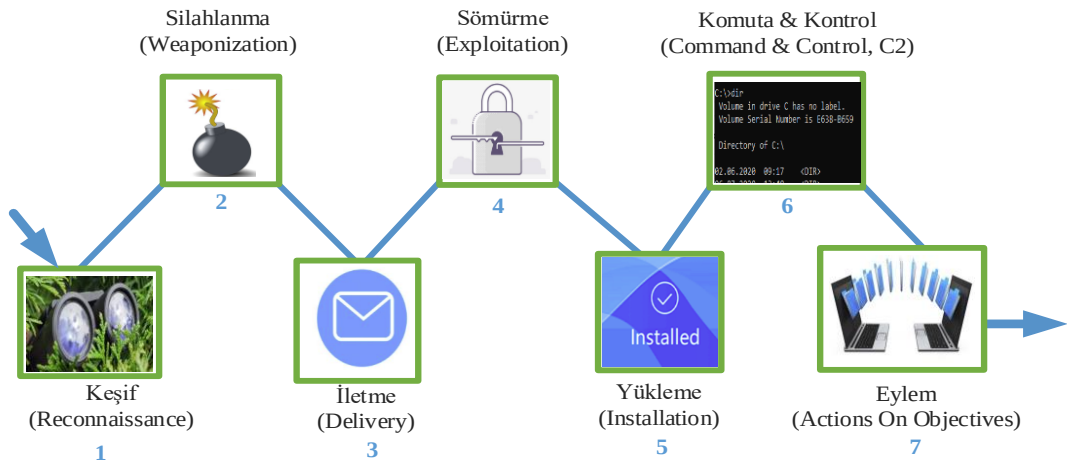


Şekil 1.4.Yıllara göre siber güvenlik pazarı

Artan internet kullanımı birçok güvenlik açığını da beraberinde getirmektedir. Siber tehditlere karşı koyabilmek için, öncelikle gelişmiş bir siber saldırı aşamalarının bilinmesi gerekmektedir.

1.1. Gelişmiş Bir Siber Saldırının Uygulama Adımları

Siber saldırganlar, hedef sisteme saldırmadan önce detaylı bir ön çalışma yaparlar. Hedef ile ilgili bilgiler toplandıktan sonra, sistemli ve organize bir şekilde çalışarak başarımlarını artırmaya çalışırlar. Siber saldırıları önlemenin en temel prensibi, saldırının nedenini ve saldırganın peşinde olduğu verileri saptayabilmektir. Siber güvenlik uzmanları, siber saldırıların analizi için bazı modeller üzerinde çalışmaktadır. Saldırı faaliyetlerini açıklamak için en yaygın kullanılan operasyonel tehdit istihbarat modellerinden birisi Lockheed Martin firmasının geliştirdiği Siber Ölüm Zinciri (Cyber Kill Chain) modelidir. Bu model, gelişmiş sürekli tehdit (An advanced persistent threat - APT) saldırılarını analiz etmek amacıyla oluşturulmuştur. Model, her biri diğerine bağlı 7 adımdan oluşmaktadır ve tüm adımlar sıra ile gerçekleşmeden diğer adıma geçilememektedir. Siber Ölüm Zinciri modeli Şekil 1.5’de verilmiştir [4–7].



Şekil 1.5. Lockheed Martin siber ölüm zinciri modeli

1.1.1. Keşif (Reconnaissance)

Siber ölüm zincirinin ilk halkasıdır. Amaç, saldırıdan önce hedef hakkında bilgi toplamaktır. Aktif ve pasif bilgi toplama yöntemleri ile hedef sistemin IP adresleri, güvenlik mekanizmaları, işletim sistemleri ve sürümleri, ifşa olan basit parolaları gibi bilgileri elde edilir [4–7].

1.1.2. Silahlanma (Weaponization)

Saldırgan, keşfettiği zafiyetleri sömürmek için hazırlıklarını yapar ve sızma yöntemini belirler. Saldırgan bu aşamada, “Keşif” aşamasında elde ettiği işletim sistemi, işletim sistemi sürümleri, açık portlar, yapılmamış güvenlik yamaları gibi bilgileri kullanır ve bu bilgiler doğrultusunda keşfettiği açıklara sızmak için özel yazılımlar geliştirir [4–7].

1.1.3. İletme (Delivery)

Saldırganın, hazırladığı kötü amaçlı yazılımı hedefe iletme adımıdır. Saldırgan; sosyal mühendislik, ortalama (phishing) gibi yöntemler ile hazırladığı kötü amaçlı yazılım hedefe iletir [4–7].

1.1.4. Sömürme (Exploitation)

Saldırganın, tespit ettiği zafiyetleri sömürerek zararlı kodu çalıştırdığı aşamadır. Bu aşamada, hazırlanan zararlı yazılım hedef sistemde çalıştırılır ve hedefe erişim sağlamak için güvenlik açıkları bulunur [4–7].

1.1.5. Yükleme (Installation)

Bu aşamada saldırı, hedef sisteme uzaktan bağlanabilmek, hedef sistemde kalıcı olabilmek için daha önceden hazırladığı yazılım ve uygulamaları yükler. Ayrıca bu aşamada saldırı, hedef ağda daha fazla sistemi ele geçirmeyi ve hazırladığı zararlı yazılımı ağda yaymayı amaçlar [4–7].

1.1.6. Komuta ve Kontrol (Command & Control, C2)

Saldırgan, hedef sistemi kısmen ya da tamamen ele geçirmiştir. Saldırgan, hazırladığı zararlı yazılım ile hedef sistemden kendisine güvenli bir haberleşme kanalı açarak hedef sistemi kontrol altında tutar [4–7].

1.1.7. Eyleme Geçme (Actions on Objectives)

Hedefe yerleşen saldırgan artık veri çalma, şifreleme, silme gibi işlemlerin yanı sıra sistemi çökertme gibi eylemleri de gerçekleştirebilir [4–7].

Siber saldırılara hedef olan kurum veya şirketler siber ölüm zinciri gibi modeller ile saldırı öncesinde sistemlerindeki açıkların tespiti, saldırı esnasında gelen saldırının aşamasına göre müdahale yöntemini belirleme, saldırı sonrası zarar tespiti gibi süreçleri yönetebilirler. Siber saldırıları önlemenin en önemli unsuru saldırı çeşitlerini ve metodolojilerini iyi bilmektir.

1.2. Siber Saldırı Çeşitleri

Teknoloji, gündelik hayatımızın hemen her noktasında kullanılmaktadır. Bu durum, siber tehditlerin büyük etkiler bırakmasının önünü açmıştır. Sayıları sürekli artan siber saldırılar; kişilerin, devlet kurumlarının, şirketlerin önemli verilerini hedef almakta ve büyük zararlara sebep olmaktadır. Örneğin son zamanlarda meydana gelen şirketlere ait verilerin çalınması, devletlerin gizli e-mail veya belgelerinin tüm dünyaya yayınlanması, bazı ünlülere ait fotoğraf ve video benzeri özel bilgilerinin çalınması gibi olaylar siber saldırılar yapılarak elde edilmiştir. Akademik dünya, şirketler ve kurumlar siber saldırılara karşı koyabilmek için sürekli yeni önlemler geliştirmektedirler. Fakat önlemler geliştikçe buna paralel olarak saldırganlar da saldırı yöntemlerini geliştirmektedirler. Bir siber saldırı önlemenin en önemli adımı, yapılan siber saldırı hakkında bilgi sahibi olmaktır. Bu amaçla, bu tez çalışmasında en sık görülen siber saldırı yöntemleri ele alınmıştır.

1.2.1. Zafiyet Taraması

Bir sistemde bulunan tüm açıkların tespit edilmesi için sistemin taranması işlemidir. Hedef sistemin ağ topolojisi, açık portlar, çalışan servisler, çalışan işletim sistemleri ve sürümleri gibi bilgileri sunar. Dolayısıyla saldırının sonraki adımları için kritik öneme sahiptir. Bunun yanında zafiyet taraması, hedef sistemdeki güvenlik açıkları için değerlendirme ve analiz sağlar. Böylece sistem yöneticileri, bazı tehlikeli arka kapıları ve bazı zayıflıkları tespit etmek ve sistem zarar görmeden önce bu açıkları revize etmek için kullanabilir. Zafiyet tespiti aşamasında, aktif ve pasif keşif türleri kullanılır [8,9].

Pasif keşifte hedef sistem ile doğrudan iletişime geçilmez [4–7]. Pasif keşif için yaygın olarak iki yöntem kullanılır. İlk yöntemde wireshark, tcpdump gibi araçlar ile hedef ağ dinlenir. Hedef ağ dinlenirken switchport mirroring, rogue Dynamic Host Configuration Protocol (DHCP), Man in The Middle (MiTM) gibi yöntemler kullanılır. İkinci yöntemde, shodan, foca gibi yazılımlar ile hedef sistem hakkında bilgi toplanır.

Aktif keşif, hedefe doğrudan paket gönderilerek yapılır. Gönderilen paketlerden alınan cevaplar yorumlanır ve hedef hakkında bilgi elde edilir. IP tarama, port ve servis tarama, web

uygulama tarama gibi tarama çeşitleri ile hedef sistemden bilgi toplanır. Nmap, Nessus, Nexpose, OpenVAS aktif tarama için sık kullanılan araçlardır. Bu tez çalışmasında hedef sistemin keşfi için açık kaynak kodlu, ücretsiz ve güçlü bir tarama aracı olan Nmap kullanılmıştır. Adını Network Mapper'in kısaltmasından alan Nmap; açık sistemleri, bu sistemlerin portlarının durumlarını, çalışan servisleri, çalışan işletim sistemleri gibi birçok bilgiyi sunmaktadır. Bu tez çalışmasında Nmap, hedef sistemdeki açık portların keşfi için kullanılmıştır. Hedef sistemdeki tüm açık portların keşfi için Transmission Control Protocol (TCP), User Datagram Protocol (UDP) ve Synchronize (SYN) tarama teknikleri kullanılmıştır [4–7].

TCP taramasında hedef port ile bağlantı kurabilmek için hedefe SYN paketi gönderilir. Hedeften SYN / ACK yanıtı gelirse Acknowledgement (ACK) paketi gönderilerek porta bağlanır ve portun açık olduğu belirlenir. Eğer, gönderilen SYN paketine karşın Reset (RST) yanıtı alınırsa portun kapalı olduğu raporlanır. Dolayısıyla TCP taramasında açılan oturumlar hedef sistemde loglanmaktadır. SYN taraması, hedef sistemde oturumu tamamen açmaz. Gönderilen SYN paketi karşılığında SYN/ACK cevabı alındığında hedef portun açık olduğu belirlenir ve hedefe RST paketi gönderilerek oturum kapatılır. Eğer hedef port kapalı ise, hedef bilgisayar RST gönderir ve oturumu kapatır. UDP taraması hedef ağda açık UDP portlarının belirlenmesi için kullanılır. Hedefe gönderilen UDP paketine gelen cevap “Internet Control Message Protocol (ICMP) Port Unreachable” ise port kapalıdır. Bu tez çalışmasında TCP taraması sonucu alınan veriler “TCP_scan”, SYN taraması sonucu alınan veriler “SYN_scan” ve UDP taraması sonucu alınan veriler ise “UDP_scan” olarak etiketlenmiştir. Test aşamasında her bir tarama türü için Kali Linux işletim sistemi üzerinde aşağıdaki Nmap komutları çalıştırılmıştır. Komutlardaki “-sT” ifadesi TCP taraması, “-sS” ifadesi SYN taraması, “-sU” ifadesi UDP taraması için kullanılmıştır. Tüm komutlarda ortak kullanılan “-sV” parametresi işletim sistemi, port üzerinde çalışan uygulamanın versiyonu ve port üzerinde çalışan servisleri bulmak için kullanılmıştır. “-p-” ifadesi 0-65536 arasındaki tüm portların taranması için kullanılmıştır. 192.168.20.20-37 ifadesi hedef ağdaki bilgisayarların IP aralığıdır [10].

TCP_scan Nmap komutu: *nmap -sT -sV -p- 192.168.20.20-37*

SYN_scan Nmap komutu: *nmap -sS -sV -p- 192.168.20.20-37*

UDP_scan Nmap komutu: *nmap -sU -sV -p- 192.168.20.20-37*

1.2.2. MAC Adres Taşması

Her anahtarlama cihazının belirli bir Media Access Control (MAC) adres barındırma kapasitesi bulunmaktadır. MAC adres taşması (MAC Flooding) saldırısı ile saldırgan, anahtarlama cihazına hızlı bir şekilde sahte MAC adresleri gönderir ve anahtarlama cihazının MAC tablosunu doldurur. MAC tablosu dolan anahtarlama cihazı çoklayıcı (hub) mantığı ile çalışır. Böylelikle,

saldırgan anahtarlama cihazına bağı tüm bilgisayarların ağ trafiğini dinleyebilme yetkisine sahip olur [11].

1.2.3. DHCP Starvation Saldırısı

DHCP, ağ cihazlarına; IP adresi, ağ maskesi ve ağ geçidi adreslerini otomatik olarak atayan bir protokoldür. DHCP Starvation saldırısında, sürekli sahte MAC adresleri üretilir. Üretilen sahte MAC adresleri ile ortama sürekli DHCP discover mesajları gönderilir. Saldırgan, oluşturduğu birçok sahte kullanıcı için DHCP sunucusuna IP isteğinde bulunur ve DHCP havuzundaki tüm IP adreslerini kiralar. DHCP starvation saldırısı hizmet reddi saldırılarının bir çeşididir. Bu tür saldırılarda yersinia ve gobbler araçları sık kullanılır [12,13].

1.2.4. ARP Zehirleme Saldırısı

Adres Çözümleme Protokolü (Address Resolution Protocol-ARP), Open Systems Interconnection (OSI) referans modelindeki 3.katman adreslerinin (IP adreslerinin), 2. katman adreslerine (MAC adreslerine) çözümlenmesini sağlayan bir protokolüdür. Arp zehirleme (Arp Poisoning) saldırıları, MAC adresleri ile IP adresleri arasındaki ilişkiyi manipüle eder [14].

1.2.5. Kaba Kuvvet Saldırısı

Hedef sistemdeki tüm yetkilere sahip olmak için yapılır. Kaba kuvvet (Brute Force) saldırısı, eski bir saldırı tipi olmasına rağmen, etkili ve düşük maliyetli olduğu için popülerliğini korumaktadır. Amaç, sisteme erişim yetkisi bulunan kullanıcının sisteme giriş bilgilerini elde etmektir. Bu amaçla, hedef sisteme veri yüklemesi yapılır ve sistem meşgul edilir. Bu yöntemde elde herhangi bir bilgi yok ise, farklı kullanıcı adı ve şifre kombinasyonları kullanılır ve şifre tespit edilmeye çalışılır. Bu saldırı çeşidi sıklıkla Telnet, Secure Shell (SSH), Remote Desktop (RDP), File Transfer Protocol (FTP), Hyper Text Transfer Protocol (HTTP) gibi protokoller kullanan sistemlere uygulanmaktadır. Bu tez çalışmasında Telnet, SSH, RDP, FTP ve XSS kaba kuvvet saldırıları uygulanmıştır [15–17]. Alınan veriler Brute Force Telnet, Brute Force SSH, Brute Force RDP, Brute Force FTP, Brute Force XSS olarak etiketlenmiştir. Saldırı esnasında rastgele şifre kombinasyonları oluşturmak için Crunch şifre oluşturma aracı kullanılmıştır. Saldırıları düzenlemek için xHydra aracı kullanılmıştır. xHydra, güçlü bir Kaba Kuvvet saldırı aracı olan Hydra'nın web tabanlı sürümüdür.

1.2.6. Hizmet Reddi Saldırısı

Hizmet reddi (Denial of Service – DoS) saldırıları, hedef sistemin kullanıcılarına hizmet vermesini engelleyen siber saldırı çeşididir. Her sistemin sahip olduğu kaynaklar sınırlıdır.

Sistemlerin sahip olduđu kaynaklar sömürüldüğünde verilen hizmetler yavaşlamakta hatta sistem hizmet veremez duruma gelebilmektedir. DoS saldırılarında amaç, hedef sistemi gereksiz istekler ile aşırı yüklemek ve bunun sonucunda hedef sistemi kullanıcılarına cevap veremez hale getirmektir. DoS saldırılarının birden fazla kaynaktan aynı anda gerçekleştirilmesi sonucu Dağıtılmış Hizmet Reddi (Distributed Denial of Service - DDoS) saldırıları oluşur. DDoS saldırılarında amaç hedefi daha kısa sürede etkisizleştirmektir [18–21]. Bu tez çalışmasında DoS saldırı türlerinden ICMP Flood, SYN Flood, HTTP Flood saldırıları düzenlenmiştir.

ICMP Flood, hedef ağı sonsuz sayıda ICMP paketinin gönderilmesi ile hedef ağı kaynaklarının tüketilmesi sonucu oluşan bir DoS saldırısıdır. SYN Flood saldırısı, ağ haberleşmesinin temelini hedef alır. Normal bir ağ bağlantısında istemci, sunucuya TCP/SYN paketi gönderir ve haberleşmeyi başlatır. SYN Flood saldırısı, sunucuya sonsuz TCP/SYN paketi göndererek normal haberleşme mekanizmasını bozar. Sunucuda yarı-açık bağlantı sayısı arttığı için sunucu kaynakları tükenir ve sunucu, yeni isteklere yanıt veremez hale gelir. 7. katman saldırılarından biri olan HTTP GET Flood saldırıları, bir ya da birden fazla ip adresinden hedef web sitesine yapılan GET istekleri ile yapılır. GET saldırıları, özellikle veri tabanı sorgusu yapan web sayfalarında iş yükü yapar. Buda hedef sistemin yapılan legal isteklere geç dönmesine sebep olur [22,23]. Bu tez çalışmasında ICMP Flood ve SYN Flood saldırıları için Kali Linux işletim sisteminde Hping3 aracı, HTTP GET Flood saldırısı için Perl aracı kullanılmıştır.

1.2.7. İstismar Saldırıları

İstismar (exploit), hedef sistemdeki yazılım veya donanımlarda bulunan açıkları sömürmek için yazılmış kod parçalarıdır. Exploitler; hedef sisteme izinsiz giriş yapabilmek, yetkili kullanıcı oluşturmak, var olan kullanıcının erişim haklarını yükselterek sistemi tamamen ele geçirmek gibi amaçlar ile oluşturulurlar. Exploitlerin birçok çeşidi bulunmaktadır. Sıfırınca gün saldırıları (zero day attack) en tehlikeli istismar saldırılarıdır. Çünkü bu exploitler sistemde açık olduğu tespit edildiği anda yazılırlar ve sistem yöneticileri bu açıktan haberdar değildirler [24,25].

1.2.8. Fuzzing Saldırısı

Hedef uygulamalardaki zafiyetleri tespit edebilmek için kullanılan bir yöntemdir. Bu yöntemde hedef uygulamaya “fuzzing” adı verilen rastgele veriler gönderilir ve hedefin bu verilere verdiği cevaplar analiz edilir. Fuzzing testleri daha çok hedefin kapalı kutu (black-box) olarak kabul edildiği test sınıfına girer. Çünkü hedef uygulama ile ilgili herhangi bir bilgi yoktur. Fuzzing, web uygulamalarındaki zafiyet tespiti, yazılım geliştirme ve test süreçlerinde sıklıkla kullanılan bir yöntemdir [26–28].

1.2.9. HeartBleed Saldırısı

Heartbleed, Open Secure Socket Layer (OpenSSL) kullanan sunucularda meydana gelen bir zafiyettir. OpenSSL, SSL ve TLS protokollerinin açık kaynak kodlu bir sürümüdür. Open SSL'deki bu açık web sitelerinden, e-posta sunucularından ve başka çevrimiçi sitelerden kimlik bilgilerinin saldırganlar tarafından çalınabileceği anlamına gelmektedir. OpenSSL'in 1.0.1 ve 1.0.1 f sürümleri heartbleed saldırılarına açıktır. OpenSSL 1.0.1g ve daha önceki sürümleri olan 1.0.0 ve 0.9.8 bu açıktan etkilenmemiştir [29,30].

1.2.10. SQL Enjeksiyonu Saldırısı

SQL enjeksiyonu (SQL injection), web uygulama güvenliği ile ilgilenen Open Web Application Security Project (OWASP)'ın ilk 10 listesinde yer alan bir saldırı çeşididir. Web sitelerinin veri tabanlarındaki açıklardan faydalanarak veri tabanlarının ele geçirilmesidir. Bu saldırı ile web sitelerindeki kullanıcı bilgilerine, gizlenmiş bilgilere erişilebilir. Ayrıca mevcut veriler değiştirilebilir veya veri tabanı komple silinebilir. SQL injection saldırılarını gerçekleştirmek için birçok araç geliştirilmiştir. Bu saldırı için en yaygın kullanılan araç SQLMap aracıdır [31,32].

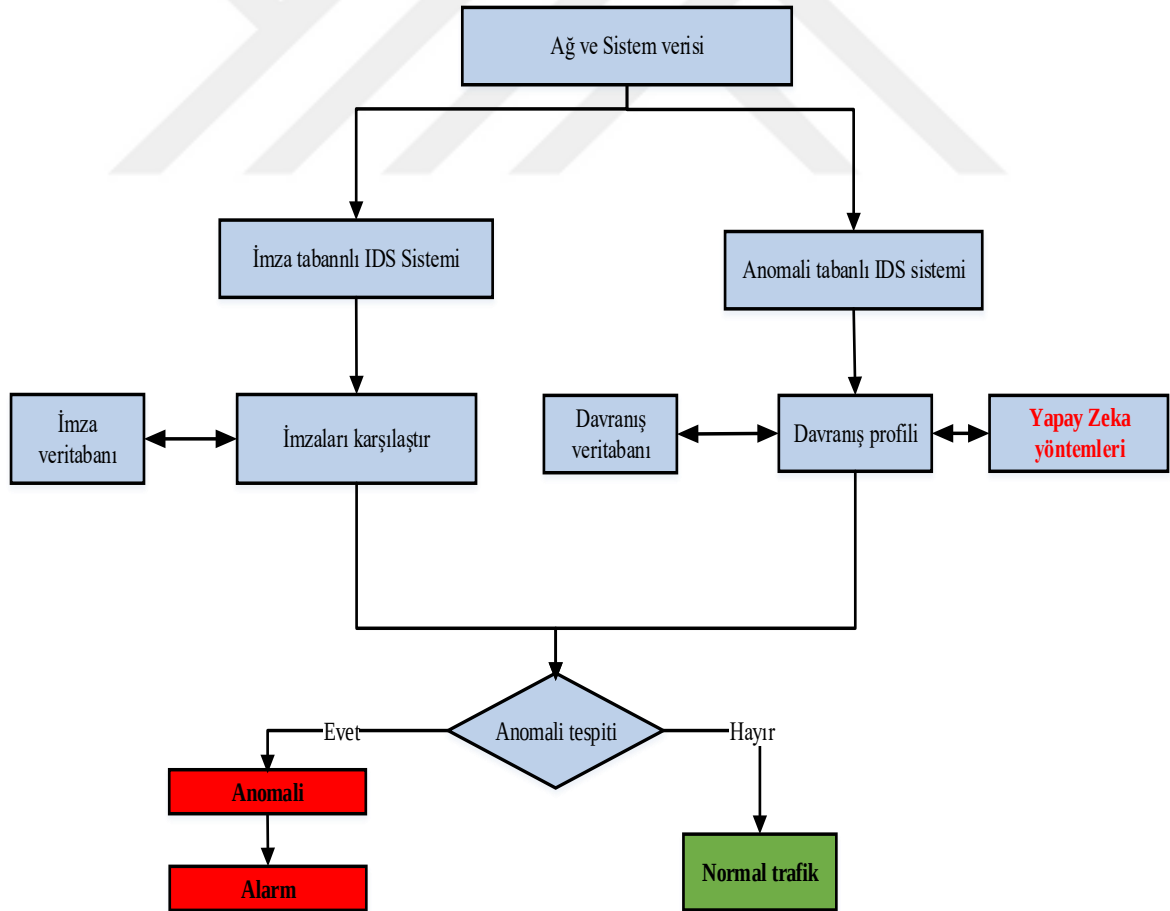
Sürekli artan internet kullanımı, beraberinde birçok veri ihlalini de getirmektedir. Birçok kurum ve kuruluş siber saldırılara maruz kalmaktadırlar. Bu nedenle siber güvenlik alanındaki çalışmalar, hem endüstride hem de akademide yoğun ilgi görmektedir. Günümüzde, siber saldırıları engellemek için çeşitli güvenlik mekanizmaları kullanılmaktadır. Buna rağmen istenilen başarımlar elde edilememektedir. Siber saldırıları önlemek için geliştirilen yönetimlerin başında Saldırı Tespit Sistemleri (STS) gelmektedir.

1.3. Saldırı Tespit Sistemleri

Günümüzde güvenlik açıklarını önlemek için güvenlik duvarı, veri şifreleme, kullanıcı kimlik doğrulama gibi birçok yöntem kullanılmaktadır. Bu güvenlik teknolojileri birçok saldırı türünü engelleyebilir, fakat bu teknolojiler derinlemesine paket analizi yapamazlar. Bu nedenle istenilen saldırı tespit düzeyine ulaşamazlar. Bu güvenlik teknolojilerinin eksikliklerini tamamlamak için saldırı önleme sistemleri (Intrusion Prevention System-IPS) ve STS sistemleri geliştirilmiştir. Bu sistemler; makine öğrenmesi, derin öğrenme, gibi yapay zeka yöntemleri sayesinde diğer güvenlik sistemlerine göre daha derin ve akıllı veri analizi yapabilmektedir. IPS sistemleri hem saldırı tespit hem de önleme mekanizması olarak çalışırken, STS sistemleri saldırı tespiti ve analizi için kullanılır [33]. Bu tez çalışmasında STS sistemleri ele alınmıştır.

Veri ihlallerini önlemenin en önemli unsurlarından biri veri paketlerini derinlemesine analiz edebilmektir. Bu bağlamda, STS sistemleri; ağı izlemek, olabilecek tehditleri tanımlamak ve bunlar

ile ilgili olay kayıtlarını tutmak, güvenlik açıklarını tespit etmek ve veri ihlallerinin önüne geçebilmek için sıklıkla kullanılan teknolojilerdendir. STS sistemleri, saldırıları tespit ve analiz ederken imza tabanlı veya anomali tabanlı olarak çalışırlar. İmza tabanlı STS sistemlerinde veri paketleri incelenir ve incelenen veri paketleri STS sisteminin veri tabanında bulunan saldırı kalıpları ile eşleştirir. Eşleşme bulunursa alarm üretilir. Bu tip imza tabanlı STS sistemlerinin en büyük dezavantajı, sıfıncı gün saldırısı gibi yeni çıkmış saldırı senaryolarının imzalarını veri tabanlarında bulundurmamaları için alarm üretemezler. Anomali tabanlı sistemlerde, normal davranış durumları STS veri tabanında depolanmıştır. Bu sistemler, ağdaki tüm olayları izleyerek, belirlenen kuralların dışına sapma olduğunda alarm üretirler. Anomali tabanlı saldırı tespit sistemlerinin en büyük avantajı, bilinmeyen saldırıları tespit edebilmeleridir. Bu tip STS sistemleri, hem veri ihlallerinin önüne geçebilmek hem de ağ saldırılarını önlemek amacıyla makine öğrenmesi, yapay zeka, derin öğrenme gibi aktif öğrenme yetenekleri kullanırlar. Böylelikle, ağlardaki anormal davranışlar izlenir ve bu anormalliklere karşı alarm üretilir. Şekil 1.6’da STS sistemlerinin genel yapısı özetlenmiştir [33–39].

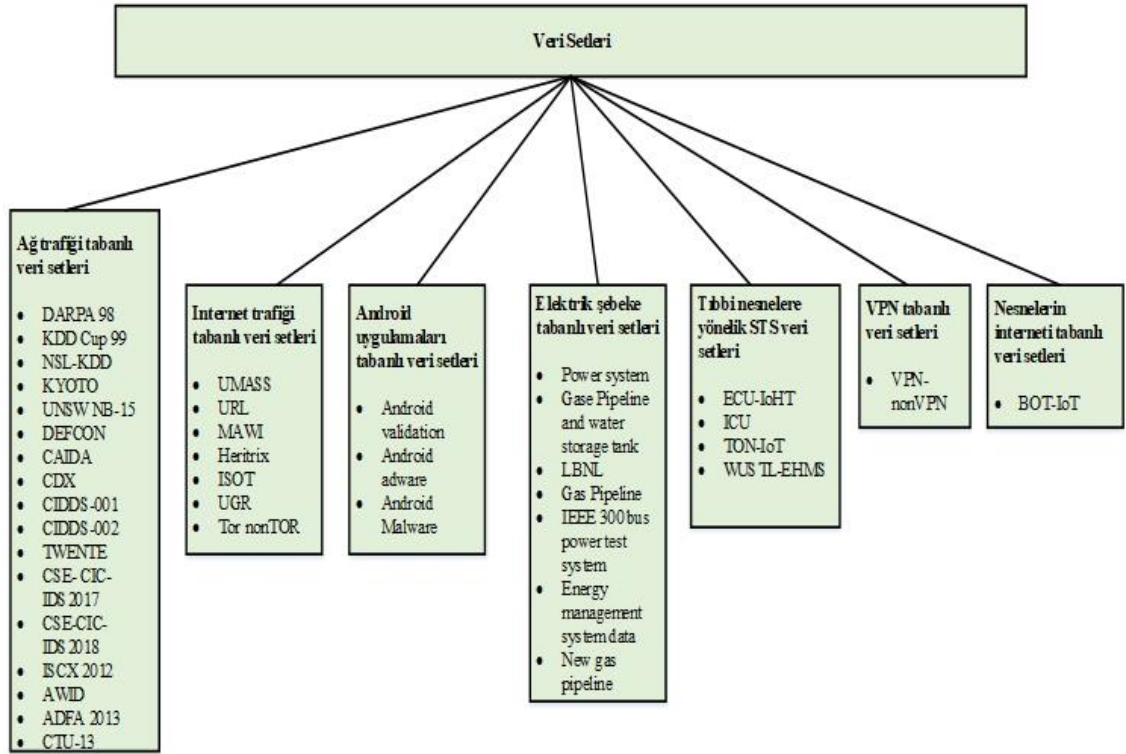


Şekil 1.6.STS sistemlerinin genel yapısı

Saldırganlar saldırı senaryolarını geliştirdikçe, bu saldırılara karşı koyabilmek için de sürekli yeni sistemler geliştirilmekte veya var olan sistemler güncellenmektedir. Saldırı tespit sistemlerinin gelişimine yönelik çalışmalar hem akademide hem de endüstride çok popülerdir. Literatürde saldırı tespit sistemlerini geliştirmeye yönelik birçok çalışma mevcuttur. Bu tez çalışmasında, literatürde sıklıkla kullanılan saldırı tespit veri setleri ve bu konuda yapılan çalışmalardan bir kısmı ele alınmıştır.

1.3.1. STS Veri Setleri

STS sistemlerinin iyileştirilmesine yönelik literatür çalışmalarının çoğunda hazır veri setleri kullanılmaktadır. Siber saldırıların tespitinde kullanılabilecek birçok veri seti bulunmaktadır. Ferrag ve arkadaşları [40], genel veri setlerini içeriklerine göre Şekil 1.7’de gösterildiği gibi 7 sınıfa ayırmıştır.



Şekil 1.7. Veri setleri ve sınıfları

Bu tez çalışmasında ağ saldırılarının tespiti için üretilmiş ve literatürde sıklıkla kullanılan ağ tabanlı ve tıbbi nesnelere yönelik STS veri setleri incelenmiştir. Literatür çalışmalarında yaygın kullanılan veri setlerine aşağıda değinilmiştir:

- **DARPA veri seti:** DARPA veri seti, 1998 yılında MT Lincoln Laboratuvarı'nda üretilen ağ tabanlı bir veri setidir. Eğitim verileri yedi haftalık, test verileri ise iki haftalık saldırı verisi içermektedir [40–42].
- **KDD Cup 99 veri seti:** KDD Cup 99 veri seti, DARPA'98 veri seti programına dayanmaktadır. DoS, Remote to Local (R2L), User to Root (U2R), Probing saldırıları simüle edilmiştir. Veri seti 7 haftalık ağ trafiğini içermekte ve yaklaşık 5 milyon satırdan oluşmaktadır. Bu veri seti, saldırı tespit modellerinin değerlendirilmesi için en yaygın kullanılan veri setlerinden biridir [40,41,43–45].
- **NSL-KDD veri seti:** NSL-KDD veri seti, KDD 99 veri setindeki problemleri çözmek için geliştirilmiştir. Orijinal KDD 99 veri setine göre gereksiz ve tekrarlanan kayıtları içermez. Mükerrer ve gereksiz kayıtların kaldırılması sonucunda veri seti yaklaşık 5 milyon kayıttan 150.000 kayıda düşürülmüştür. Ayrıca izinsiz giriş tespit yöntemleri için önceden tanımlanmış eğitim ve test alt gruplarına ayrılmıştır. NSL-KDD, KDD CUP 99 ile aynı özellikleri ve sınıfları kullanır. DoS, Remote to Local (R2L), User to Root (U2R), Probing saldırıları bu veri seti için de kullanılmıştır [40,41,43,44].
- **Kyoto veri seti:** Bu veri setinde yer alan 24 özelliğin 14'ü KDD Cup 99 veri setinden oluşturulmuştur. Bu 14 özellik, Kyoto Üniversitesi'nde kurulu balküpu (honeypot) sistemleri ile KDD Cup 99 veri setinin 41 özelliği arasından seçilmiştir. Bu 14 özelliğe ek olarak daha etkin bir saldırı tespit sistemi için 10 adet yeni özellik belirlenmiştir [40,46].
- **UNSW-NB15 veri seti:** UNSW-NB15, Australian Center for Cyber Security (ACCS) tarafından IXIA Perfect-Storm, Tcpdump, Argus ve Bro-IDS kullanılarak oluşturulan bir veri setidir. Normal ve anormal trafik oluşturucu olarak kullanılan IXIA aracı, üç sanal sunucu üzerine kurulmuştur. IXIA aracı kullanılarak Fuzzers, Analysis, Backdoors, DoS, Exploits, Generic, Reconnaissance, Shellcode, Worms olmak üzere 9 saldırı senaryosu kullanılmıştır. IXIA aracı, modern bir tehdit ortamı oluşturmak için CVE sitesini kullanır. Test ortamında kullanılan yönlendiriciler güvenlik duvarına bağlanmıştır. Güvenlik duvarı, tüm trafiği normal veya anormal şekilde geçirecek şekilde yapılandırılmıştır. Yönlendiricilerden birinde tcpdump çalıştırılmış ve simülasyon sırasında alınan veriler kaydedilmiştir. UNSW-NB15 veri seti, 49 özellikten oluşmaktadır. Ayrıca Moustafa ve diğerleri bu veri setinden, 175.341 adet vektör içeren eğitim seti ve 82.332 vektör içeren test seti yayınlamışlardır [40,47,48].
- **CAIDA veri seti:** Bu veri seti, Center of Applied Internet Data Analysis (CAIDA) tarafından önerilmiştir. CAIDA DDoS, CAIDA Internet traces 2016 ve RSDoS Attack Metadata (2018-09) dâhil olmak üzere farklı veri setlerini içermektedir [39].
- **CIDDS-001 veri seti:** Coburg Network Intrusion Detection Dataset veri seti, anomali tabanlı bir ağ saldırı tespit sisteminden oluşturulmuştur. Veri seti oluşturulurken Nmap ve

python dilinde yazılmış scriptler kullanılmıştır. Portscan, Pingscan, DoS ve Brute force saldırıları yapılmıştır. Veri seti 14 özellik içerir ve Normal, Attacker, Victim, Suspicious, Unknown sınıflarından oluşur. Veri seti, Openstack ve harici bir sunucudan gelen trafik verilerinden oluşmaktadır [49–52].

- **ISCX-2012 veri seti:** Bu veri seti yedi günlük ağ verilerinden oluşturulmuştur. Veri seti, normal ve kötü niyetli ağ trafiğini içerir. Kötü amaçlı ağ trafiği, ağa içeriden sızmayı, HTTP DoS, DDoS ve Brute Force SSH saldırılarını içerir. Veri setinde “Normal” ve “Attack” sınıfları bulunmaktadır [40,53].
- **AWID veri seti:** AWID, IEEE 802.11 standardına göre oluşturulan bir veri setidir. Veri seti 156 özellik içermektedir. Veri seti oluşturulurken 802.11 ağlarında sıklıkla meydana gelen honeypot, rogue access point, evil twin, deauthentication attack, dictionary, fragmentation gibi farklı saldırı yöntemleri kullanılmıştır [41,54,55].
- **CSE-CIC-IDS 2017 veri seti:** Bu veri seti, Communications Security Establishments (CSE) & the Canadian Institute for Cybersecurity (CIC) tarafından 2017 yılında oluşturulmuştur. Veri setinin oluşturulması için hazırlanan test ortamında saldırgan ve kurban ağları oluşturulmuştur. Saldırıların yapıldığı ağda bir switch, Kali Linux işletim sistemli bir bilgisayar ve Windows 8 işletim sistemli 3 bilgisayar bulunmaktadır. Hedef ağ üzerinde 1 adet Windows Server 2016, 1 adet Ubuntu 16 işletim sistemli WEB sunucu, 1 adet Ubuntu 12 işletim sistemli sunucu, 1 adet router ve 1 adet firewall bulunmaktadır. Windows Server 2016’da Active Directory özelliği açık olup kurban ağındaki tüm cihazlar bu domaine alınmaktadır. Ayrıca, kurbanın ağındaki tüm trafiğin dinlenmesi için yönlendiricinin uplink portu izlenmiştir. CSE-CIC-IDS 2017 veri setinin içeriğinde, normal trafik oluşturmak için Java-B-profil sistemine dayalı bir ajan yazılmıştır. Bu ajan ile HTTP, HTTPS, FTP, SSH ve e-posta gibi protokoller makine öğrenmesi ve istatistiksel yöntemler kullanılarak yeniden üretilmiştir. Saldırı trafiği verilerini oluşturmak için Patator, slowloris, Slowhttps, Metasploit, Ares gibi bazı araçlar kullanılmıştır. Brute force, heartbleed saldırısı, botnet, DoS, DDoS, Web attack, infiltration gibi saldırılar düzenlenmiştir. Bu temel saldırı türlerinde meydana gelen toplam 14 farklı saldırı türü veri setinde etiketlenmiştir. Etiketli saldırı türleri şunlardır: DoS Golden Eye, Heartbleed, DoS hulk, DoS Slow http, DoS Slowloris, DDoS, SSH-Patator, FP, Patator, Brute force, XSS, Botnet, infiltration, PortScan, SQL injection. Ayrıca veri seti oluşturulurken CICFlowMeter kullanılmış ve yakalanan trafikten 80 adet ağ akış özelliği çıkarılmıştır [40,56,57].
- **CSE-CIC-IDS 2018 veri seti:** 2017 yılında oluşturulan veri seti ile aynı özellikleri göstermektedir. Ancak saldırıların daha iyi modellenebilmesi için test ortamında daha fazla cihaz kullanılmıştır. Bu veri setini oluşturmak için saldırgan ağında 50 makine, kurban

ağında 420 makine ve 30 sunucu kullanılmıştır. Ayrıca kurban ağı: Ar-Ge bölümü, yönetim bölümü, teknisyen bölümü, sekreter ve operasyon bölümü, bilgi işlem bölümü ve sunucu bölümü olmak üzere 6 bölüme ayrılmıştır. Bilgi işlem bölümü dışındaki tüm bölümlere farklı Windows işletim sistemleri (Windows 8.1 ve Windows 10) kurulmuştur. Bilgi işlem bölümündeki tüm bilgisayarlarda Ubuntu işletim sistemi kurulmuştur. Sunucu odası için Windows Server 2012 ve Windows Server 2016 gibi farklı sunucu işletim sistemleri kurulmuştur. Böylece gerçek dünya ağlarına benzer makine çeşitliliğine sahip bir topoloji oluşturulmuştur. Bu veri setini oluşturmak için Brute force, heartbleed, botnet, DoS, DDoS, Web saldırısı, infiltration saldırısı gibi 7 farklı saldırı senaryosu uygulanmıştır [56,58].

- **ECU-IoHT veri seti:** Sağlık alanında kullanılan sensörlerde ve cihazlarda meydana gelebilecek istismarların önlenmesine yönelik oluşturulmuştur. Veri seti oluşturma aşamasında, sızma testleri için kullanılan standart yöntemler kullanılmıştır. Edith Cowan University- Internet of Health Things (ECU-IoHT), test ortamı için, e-sağlık uygulamaları ve tıbbi cihazlar için bir geliştirme platformu olan Libelium MySignals sağlık seti kullanılmıştır. Veri setinde Arp Spoofing, DoS, Smurf, nmap port scan atakları kullanılmıştır. Veri seti oluşturulurken Argus aracı kullanılmıştır. Veri seti 7 adet özellik ve 111207 adet kayıttan oluşmaktadır [59].
- **ICU Veri seti:** Intensive Care Unit (ICU) veri seti, hastanın hayatında önemli rol oynayan yoğun bakım ünitesindeki cihazların güvenliğinin sağlanmasına yönelik hazırlanmıştır. Hussain vd. [60], tıbbi sensörler ve aktüatörler arasında güvenli ve sağlam veri iletişimini sağlamak için, siber saldırılara dayanıklı IoT tabanlı bir sağlık sistemi önermişlerdir. Bu amaçla, IoT tabanlı ve gerçek zamanlı bir ICU veri seti geliştirilmiştir. Veri seti oluşturma esnasında, hem normal hem de IoT trafiği üretebilen açık kaynak kodlu IoT-Flock yazılımı kullanılmıştır. Veri seti 46 özellik içermekte ve “Attack”, “environmentMonitoring”, “patientMonitoring” sınıflarından oluşmaktadır. Veri setinde toplamda 188610 adet kayıt bulunmaktadır [60].
- **TON-IoT Veri Seti:** Telemetry data, Operating systems’ data, and Network data from the testbed IoT/IIoT network (ToN_IoT) veri seti, UNSW @ ADFA, Cyber IoT Lab, da geliştirilmiştir. Veri seti cloud, edge, fog olmak üzere üç katman kullanılarak elde edilmiştir. Böylelikle veri setinin gerçek bir IoT ağını yansıtması hedeflenmiştir. Veri seti, IoT sistemlerinin güvenlik açıklarını tespit etmek amacıyla birçok saldırı senaryosu kullanılarak oluşturulmuştur. Veri seti oluşturulurken Password, Scanning, Ransomware, Backdoor, DoS, DDoS, MITM, Injection, and XSS saldırı türleri kullanılmıştır. Çalışmada kullanılan ToN_IoT_Train_Test_Network veri seti 43 adet özellik, 10 adet sınıf ve 460925 adet kayıt içermektedir [61,62].

- **WUSTL-EHMS Veri seti:** Washington University in St. Louis enhanced healthcare monitoring system (WUSTL-EHMS), veri seti sağlık hizmetleri sensörleri kullanılarak oluşturulmuştur. Oluşturulan sistem, çeşitli sensörlerden bilgi toplayan bir sağlık izleme sensör kartından oluşmaktadır. Veri seti oluşturulurken Ortadaki adam saldırısı (Man in The Middle - MiTM) saldırısı kullanılmıştır. Veri seti “normal” ve “Attacker” sınıflarından oluşmaktadır. Ayrıca veri seti 43 özellik ve 16318 adet kayıt içermektedir [63]. Bu çalışmada 43 adet özellikten 5 tanesi gereksiz özellik olarak tespit edilmiş ve veri setinden silinmiştir.

Saldırı tespit sistemlerinde sıklıkla kullanılan veri setleri Tablo 1.1’de özetlenmiştir. Tabloda veri setlerinin oluşturulduğu yıl, veri setleri ile ilgili birtakım literatür çalışmaları, veri setlerindeki özellik sayıları ve saldırı sınıfları bulunmaktadır.

Tablo 1.1. Saldırı tespit veri setleri

Veri Seti	Referans	Özellik sayısı	Saldırı sınıfları
CSE-CIC-IDS2017	[56],[64],[65],[41],[66]	80	DoS Golden Eye, Heartbleed, DoS hulk, DoS Slow http, DoS Slowloris, DDoS, SSH-Patator, FP, Patator, Brute force, XSS, Botnet, infiltration, PortScan, SQL injection.
CSE-CIC IDS2018	[40],[67],[65],[68]	80	DoS Golden Eye, Bening, DoS hulk, DoS Slow http, DoS Slowloris, DDoS-LOIC HTTP, DDoS-LOIC-UDP, DDoS-HOIC, SSH-Patator, FTP Patator, Brute force, XSS, Botnet, infiltration, SQL injection.
UNSW-NB15	[64],[66],[41],[47],[48],[69],[70],[71],[72],[73],[42],[74],[75]	49	Fuzzers, Analysis, Backdoors, DoS, Exploits, Generic, Reconnaissance, Shellcode, Worms
ISCX-2012	[41],[75],[76],[77],[78],[79]	19	Normal, Attacker
NSL-KDD	[66],[41],[69],[74],[43],[80],[81],[44]	43	Normal, DoS, Remote to Local (R2L), User to Root(U2R), Probing
DARPA	[41],[42],[43],[82],[83]	43	DoS, Remote to Local (R2L), User to Root(U2R), Probing
KYOTO	[41],[46]	23	Oth, rej, rstos0, rstoh, s0, s1, s2, s3, sf, sh, shr
KDD99	[41],[48],[72],[77],[43],[44],[42],[84]	43	DoS, Remote to Local (R2L), User to Root(U2R), Probing
CIDDS-001	[41],[81],[51],[52],[49],[50]	14	Normal, Attacker, Victim, Suspicious, Unknown
ECU-IoHT	[59]	7	Arp Spoofing, DoS, Smurf, port scan
ICU	[60]	46	Attack, environmentMonitoring, patientMonitoring
TON-IoT	[61],[62]	43	normal, password, scanning, ransomware, backdoor, DoS, DDoS, MITM, Injection, and XSS
WUSTL - EHMS	[63]	43	Normal, attacker

1.3.2. Saldırı Tespit Sistemleri Literatür Çalışmaları

Saldırganlar, kendilerini ve kullandıkları yazılımları her geçen gün güncellemekte ve yeni saldırı senaryoları oluşturmaktadırlar. Bu durum, veri güvenliğine yönelik saldırıların hem çeşitlerinin hem de sayılarının artması anlamına gelmektedir. Veri güvenliğine yönelik saldırıların artması ile birlikte, güvenlik teknolojilerine olan ihtiyaç sürekli artmakta ve buna paralel olarak saldırı tespit sistemlerinde hızlı bir gelişim yaşanmaktadır. Güncellenen saldırıları önlemenin en iyi yolu, saldırı tespit sistemlerini güncellemek ve geliştirmektir. Saldırı tespit sistemlerinin geliştirilmesi amacıyla literatürde sürekli yeni çalışmalar yapılmakta ve yeni yöntemler önerilmektedir.

Ferrag ve arkadaşları [40], yaptıkları çalışmada Recurrent Neural Networks (RNN), Deep Neural Networks (DNN), Restricted Boltzmann Machines (RBM), Deep Belief Networks (DBN), Convolutional Neural Networks (CNN), Deep Boltzmann Machines (DBM), ve Deep Autoencoders (DA) derin öğrenme yöntemlerini CSE-CIC-IDS2018 ve Bot-IoT veri setlerine uygulamışlardır. Ayrıca yaptıkları çalışmada özellikle derin öğrenme yöntemlerine dayalı saldırı tespit sistemleri incelemiş ve bu anlamda literatürde kullanılan 35 saldırı tespit veri seti kategorize etmişlerdir. Sharafaldin ve arkadaşları [56], mevcut veri setleri günümüzün saldırı tespit ihtiyaçlarını karşılamadığından CSE-CIC-IDS-2017 veri setini oluşturmuşlardır. Oluşturdukları yeni veri setinin performansını ölçmek için, K-Nearest Neighbors (KNN), Random Forest (RF), Iterative Dichotomiser3(ID3), Adaboost, Multilayer perceptron (MLP), Naive-Bayes (NB) ve Quadratic Discriminant Analysis (QDA) makine öğrenmesi metodlarını kullanmışlardır.

Patil ve arkadaşları [64], çalışmalarında bulut bilişimin hiper yönetici düzeyinde dağıtılmış ağ güvenliği (hypervisor level distributed network security - HLDNS) çerçevesini önermişlerdir. Saldırı tespiti için kullandıkları bu yöntemde, sunuculara yapılan yetkisiz erişimler sanal makineler ile izlenmiştir. Çalışmada, Feature similarity-based Fitness Function (FSFF) ve Classifier Accuracy based Fitness Function (CAFF) uygunluk işlevleri, özellik çıkarımı için Binary Bat Algorithm (BBA) algoritması ile birlikte kullanılmıştır. Performans analizi için önerilen güvenlik çerçevesi, UNSW-NB15 ve CSE-CIC-IDS-2017 veri setleri kullanılarak test edilmiştir. Khammassi ve arkadaşları [66], Non-Dominated Sorting Genetic Algorithm II (NSGA-II) ve lojistik regresyona dayalı bir öznitelik seçim yöntemi önermişlerdir. Önerilen yaklaşım, Non-Dominated Sorting Genetic Algorithm Binomial Logistic Regression (NSGA2-BLR) ve Non-Dominated Sorting Genetic Algorithm Multinomial Logistic Regression (NSGA2-MLR) yöntemlerine göre test edilmiştir. Elde edilen en iyi alt setler C4.5, RF, NB yöntemleri ile sınıflandırılmış ve önerilen yöntemin performansı NSL-KDD, UNSW-NB15 ve CIC-IDS2017 veri setleri ile test edilmiştir.

Moustafa ve arkadaşları [48], modern ağ trafiğini daha iyi simüle etmek için kurdukları test ortamından UNSW-NB15 veri setini üretmişlerdir. Veri seti üretilirken IXIA Perfect-Storm, Tcpdump, Argus ve Bro-IDS araçları kullanılmıştır. IXIA aracı kullanılarak test ortamında Fuzzers,

Analysis, Backdoors, DoS, Exploits, Generic, Reconnaissance, Shellcode, Worms olmak üzere 9 adet güncel saldırı senaryosu oluşturulmuştur. Oluşturulan UNSW-NB15 ve KDD Cup99 veri setleri NB, Decision Tree (DT), Artificial Neural Network (ANN), Logistic Regression (LR) ve Expectation Maximization (EM) Clustering yöntemleri ile sınıflandırılmıştır. Moustafa ve arkadaşları [69], yaptıkları başka bir çalışmada özellikler ve gözlemler arasındaki mesafeler için Beta Mixture Model (BMM) parametrelerinden hesaplanan ve Trapezoidal Area Estimation (TAE) tahminine dayalı Geometric Area Analysis (GAA) tekniği sunmuşlardır. NSL-KDD ve UNSW-NB15 veri setleri üzerinde bu yöntem test edilmiştir. Zhang ve arkadaşları [70], Multiscale Convolutional Neural Network (MSCNN)'i Long Short-Term Memory (LSTM) ile birleştiren birleşik bir yöntem önermişlerdir. Önerilen yöntemin ilk aşamasında veri setinin uzamsal özelliklerini analiz etmek için MSCNN kullanılmıştır. Yöntemin ikinci aşamasında geçici özellikleri işlemek için LSTM ağı kullanılmıştır. Modelin eğitimi ve testi için UNSW-NB15 veri seti kullanılmıştır. Gottwalt ve arkadaşları [71], çok değişkenli korelasyon tabanlı ağ anomali tespiti için multivariate correlation (MC)'ye dayanan CorrCorr özellik seçme yöntemi önermişlerdir. Önerilen yöntem UNSW-NB15 ve NSL-KDD veri setleri üzerinde test edilmiştir. Kasongo ve arkadaşları [73], Wrapper Based Feature Extraction Unit (WFEU) özellik çıkarma yöntemi ile birlikte Feed-Forward Deep Neural Network (FFDNN) kullanarak AWID ve UNSW-NB15 veri setleri üzerinde testler yapmışlardır.

Awad ve arkadaşları [76], Katmanlı örnekleme yöntemini ve farklı maliyet fonksiyon şemalarını Weighted Extreme Machine Learning (WELM) ile birleştirmiş ve ISCX2012 veri setine uygulamışlardır. Tan ve arkadaşları [77], hizmet reddi saldırılarını tespit etmek için Earth Mover's Distance (EMD) yöntemini ISCX 2012 ve KDDCup99 veri setlerine uygulamışlardır. Bouteraa ve arkadaşları [78], ISCX 2012 veri setini kullanarak izinsiz giriş tespiti için veri madenciliği tekniklerinin karşılaştırmalı bir çalışmasını yürütmüşlerdir. Önerdikleri yöntemde RF, Destek Vektör Makineleri (DVM), Extreme Learning Machine (ELM), Recursive Partitioning And Regression Trees (RPART) ve One-Class Support Vector Machine (OCSVM) algoritmalarını kullanmışlardır.

Injadat ve arkadaşları [85], anomali tespiti için önerdikleri yöntemde SVM, RF ve KNN parametrelerini ayarlamak için Bayesian Optimization (BO) tekniğini kullanmışlardır. Yasin ve arkadaşları [79], K-Means Clustering (KMC) ve Naive Bayes Classifier (NBC) sınıflandırıcılarını beraber kullanarak KMC + NBC algoritmasını önermişlerdir. Çalışmalarında ISCX2012 veri setini kullanmışlardır. Hasan ve arkadaşları [75], ağ saldırılarını yüksek doğruluk değerleriyle tespit etmek için bir hibrit derin öğrenme modeli önermişlerdir. Bu amaçla CNN ve weight-dropped, long short-term memory (WDLSTM) derin öğrenme yöntemlerini birlikte kullanmışlardır. Önerilen yöntemin performansını ölçmek için UNSW-NB15 ve ISCX2012 veri setlerini kullanmışlardır. Hajisalem ve arkadaşları [74], yaptıkları çalışmada Artificial Bee Colony (ABC) ve Artificial Fish

Swarm (AFS) kullanarak hibrit bir sınıflandırma yöntemi geliştirmişlerdir. Fuzzy C-Means Clustering (FCM) ve Correlation-based Feature Selection (CFS) teknikleri ile özellik seçimi yapmışlardır. Son adımda, normal ve anormal kayıtları ayırt etmek için seçilen özelliklere göre CART tekniği ile if-then kuralları oluşturmuşlardır. Geliştirdikleri yöntem NSL-KDD ve UNSW-NB15 veri setlerine uygulanmış ve %99 doğruluk oranı elde edilmiştir.

Rashid ve arkadaşları [81], NSL-KDD ve CIDD5-0001 veri setlerini SVM, NB, KNN, Neural networks (NN), DNN ve DAE algoritmaları ile sınıflandırmışlardır. Sınıflandırıcı başarısını artırmak için sınıflandırma algoritmalarının öncesinde, hibrit özellik seçimi ve sıralama yöntemleri kullanılmış ve yüksek doğruluk oranları elde edilmiştir. David ve arkadaşları [82], dağıtık hizmet reddi saldırılarını tespit etmek için dinamik eşik algoritmasına dayalı istatistiksel bir yaklaşım kullanmışlardır. Yöntemi test etmek için DARPA veri seti kullanılmış ve deneysel sonuçlara göre, önerilen yöntem mevcut yöntemlere kıyasla daha yüksek tespit oranı ve daha az işlem süresi gerektirmiştir. Ertam ve arkadaşları [84], KDD Cup 99 veri seti üzerinde NB, bayes Net (bN), RF, MLP, Sequential Minimal Optimization (SMO) algoritmalarını kullanarak sınıflandırma yapmışlardır. Kolias ve arkadaşları [54], 802.11 ağlarında meydana gelen saldırıları kapsamlı olarak değerlendirmiş ve bu bağlamda AWID veri setini oluşturmuşlardır. Daha sonra oluşturdukları AWID veri seti üzerinde Adaboost, Hyperpipes, J48, NB, OneR, RF, ZeroR algoritmaları ile sınıflandırma yapmışlardır. Verma ve arkadaşları [50], yaptığı başka bir çalışmada yine CIDD5-001 veri setini k-Nearest Neighbour (kNN), SVM, Decision Tree (DT), RF, NB, DL, Artificial Neural Networks (ANN), Self Organising Maps (SOMs), Expectation Maximisation (EM), k-means algoritmaları ile sınıflandırmış ve bu sınıflandırıcı sonuçlarını karşılaştırmıştır. Dash ve arkadaşları [86], NSL-KDD veri setini sınıflandırmak için hibrit bir makine öğrenmesi yöntemi önermişlerdir. Veri setini ön işleme adımlarından geçirdikten sonra, Improved Harmony Search Extreme Learning Machine based IDS (IHSELMIDS) yöntemiyle sınıflandırmışlardır. STS sistemleri ile ilgili literatür çalışmaları, Tablo 1.2’de özetlenmiştir. Tabloda önerilen yöntemler, kullanılan veri setleri, ölçülen başarımlar metrikleri verilmiştir.

Tablo 1.2. Saldırı tespit sistemleri literatür özeti

Referans	Yıl	Özellik seçimi	Metod	Kullanılan veri setleri	Performans metrikleri
[40]	2020	-	DNN, RNN, CNN	CSE-CIC-IDS 2018	Doğruluk(DO), DO Süresi(saniye)
			RBM, DBN, DBM, DA	BoT-IoT	
[56]	2018	-	KNN, RF, ID3, Adaboost, MLP, NB, QDA	CSE-CIC-IDS 2017	Hassasiyet (H), Duyarlılık (D), F1-Değeri (FÖ)
[64]	2019	BBA+CAFF BBA+FSFF	RF	CSE-CIC-IDS 2017	DO, Yanlış Pozitif (YP)
				UNSW-NB15	

Tablo 1.2. (Devamı)

[67]	2019	-	CNN	CSE-CIC-IDS 2018	DO
[65]	2019	-	ANN, RF, KNN, SVM, Adaboost, NB	CSE-CIC-IDS 2018	DO, H, D, FÖ
[66]	2020	NSGA2-BLR	C4.5, RF, NB Tree	CSE-CIC-IDS 2017	DO
		NSGA2-MLR		UNSW-NB15	
				NSL-KDD	
[68]	2019	MLP	ANN	CSE-CIC-IDS 2018	DO, H, D,
[48]	2016	-	DT, LR, NB, ANN, EM Clustering	UNSW-NB15	DO, Yanlış Alarm Oranı (YAO)
				KDD99	
[69]	2019	GAA-ADS	k-Means	UNSW-NB15	DO, Tespit Oranı (TO), YP
				NSL-KDD	
[70]	2020	-	Lenet, MSCNN, MSCNN-LSTM	UNSW-NB15	DO, YAO, Yanlış Negatif (YN)
[71]	2019	CorrCorr	CorrCorr	UNSW-NB15	DO, TO, YP
[72]	2017	GA (Genetic Algorithm) -LR	C4.5, RF, NB Tree	UNSW-NB15	DO, TO, YAO
				KDD99	
[73]	2020	WFEU	FFDNN	UNSW-NB15	DO
				AWID	
[76]	2019	-	WELM	ISCX2012	DO, H, D, FÖ
[77]	2015	EMD	-	ISCX2012	DO, TO, YP
				KDDCUP99	
[78]	2020	-	RF, SVM, ELM, RPART, OCSVM	ISCX2012	DO, TO, YAO
[85]	2018	BO	SVM, KNN, RF	ISCX2012	DO
[79]	2013	-	KMC + NBC	ISCX2012	DO, TO, YAO
[75]	2020	-	CNN + WDLSTM	ISCX 2012	DO
				UNSW-NB15	
[74]	2018	FCM, CFS	ABC, AFS	UNSW-NB15	DO, TO, YP
				NSL-KDD	
[81]	2020	Hibrit özellik seçimi	KNN, SVM, NB, DNN, DAE	CIDDS-001	DO
				NSL-KDD	
[82]	2019	-	Dynamic Threshold Detection Algoritim	DARPA 98	DO
				DARPA 2000	
				Üretilen veri seti	
[84]	2017	-	NB, bN, RF, SMO, MLP	KDD Cup 99	DO
[54]	2016	-	Adaboost, Hyperpipes, J48, NB, OneR, RF, ZeroR	AWID	DO
[49]	2018	-	KNN	CIDDS-001	DO
[50]	2018	-	KNN, SVM, DT, RF, NB, DL, ANN, SOMs, EM, k-means	CIDDS-001	DO
[87]	2020	Spider Monkey Optimizasyonu	Fuzzy-C-Means	NSL-KDD	DO, H, D, FÖ Doğru Pozitif (DP), Yanlış Negatif (YN)
[88]	2021	DA	DNN	NSL-KDD CSE-CIC-IDS 2018	H, D, FÖ

Tablo 1.2. (Devamı)

[86]	2021	-	IHSELMIDS	NSL-KDD	DO, DP, Doğru Negatif (DN), Geometrik Ortalama (GO), FÖ, YAO
[89]	2020	-	DNN	KDDCup99 NSL-KDD UNSW-NB15	DO, H, D, FÖ
[90]	2018	K-mean, K-medoids, parallel K-Medoids	KNN	NSL-KDD UNSW-NB15	DO, H, D, DP, YP, DN, Yanlış Negatif (YN)
[44]	2009	-	J48, NB, NB Tree, RF, RF Tree, MLP, SVM	NSL-KDD	DO
[91]	2021	-	Autoencoder	CIC-IDS2017	D
[92]	2017	-	Gaussian Mixture Model (GMM)	UNSW-NB15	DO, YP
[93]	2016	-	DT, LR, NB, ANN, EM	UNSW-NB15 KDD99	DO, YAO
[94]	2016	RBM	RBM	DARPA KDDCUP99	DO, DP, DN, YP, YN
[95]	2012	Entropi tabanlı özellik seçimi	k-Means, KNN, NB	KDD99	DO, TO, YP
[38]	2020	-	Tfidfvectorizer, singular value decomposition	ADFA-LD ADFA-WD	H, D, YP, FÖ
[96]	2018	-	CNN, autoencoders, LSTM	NSL-KDD	DO
[97]	2016	-	DNN	NSL-KDD	DO, H, D, YP, FÖ
[98]	2018	-	CNN, LSTM	DARPA ISCX2012	DO, TO, YAO
[99]	2017	-	DT, NB	DARPA	DO, YP, YN
[100]	2019	Dengesiz Sınıfları İşleme	DNN, RF, Variational Autoencoder (VAE), VAE, Voting, Stacking	CIDDS-001	DO, TO, YAO
[101]	2020	-	DNN, Gated Recurrent Neural Network (GRU-RNN)	NSL-KDD	DO, H, D, FÖ
[102]	2018	Sparse Autoencoder	SVM	NSL-KDD	DO, H, D, FÖ
[103]	2018	-	ELM, RF, SVM	NSL-KDD	DO, H, D
[104]	2015	-	Online Sequential Extreme Learning Machine (OS-ELM)	NSL-KDD	DO, DP, DN, YP, YN, FÖ, H
[105]	2017	Hypergraph based Genetic Algorithm (HG - GA)	SVM	NSL-KDD	DO, TO, YAO
[106]	2018	Feature selection, weight, and parameter optimization (FWP)	SVM, GA	KDDCup99	TO, YN, YP

Tablo 1.2. (Devamı)

[107]	2020	Feed-Forward Neural Network, IGAN	Deep Learning (DL)	CSE-CIC-IDS 2017 UNSW-NB15 NSL-KDD	DO, H, D, FÖ
[108]	2018	GA	SVM	KDDCup99	DP, YP
[109]	2020	Ensemble of Feature Selection (EFS) and Adaptive Grasshopper Optimization Algorithm (AGOA)	SVM	ISCX2012	DO, H, D, FÖ, YA
[110]	2019	Firefly algorithm (FA)	SVM	NSL-KDD	DP, DN, YP, YN
[111]	2020	Particle swarm optimization (PSO), grey wolf optimizer (GWO), firefly optimization (FFA) and GA	SVM, J48 ML	UNSW-NB15	DO, DP, DN, YN, YP, FÖ
[112]	2019	Tabu-Artificial Bee Colony (TABCO)	SVM	KDDCup99	DO, YN, YP
[113]	2018	VOTE algorithm and Information Gain (IG)	J48, Meta Paggging (MP), RF, REP Tree, AdaBoost M1, Decision Stump (DS), NB	NSL-KDD	DO, YN, YP

İnternet; sosyal medya, eğitim, haber, multimedya, bankacılık, e-ticaret gibi birçok alanda vazgeçilmez bir unsur haline gelmiştir. Özellikle sosyal medya araçları toplum, yaş, kültür ve cinsiyet fark etmeksizin birçok insan için önemli bir alışkanlık haline gelmiştir. İnternet kullanımındaki bu denli artış hem araştırmacıları hem de internet üzerinde işlem yapan tüm endüstriyi internet trafiğinin incelenmesine yönlendirmiştir. Bu tez çalışmasının beşinci bölümünde de bu konu ele alınmış ve internet kullanıcılarının davranışlarını sınıflandırmak için bir yöntem önerilmiştir.

Günümüzde kullanıcıların internet davranışları şirketler için önemli bir gelir kaynağı haline gelmiştir. Artık birçok firma ürünlerini internet üzerinden satmakta ve ürünlerinin reklamını yapmak için interneti kullanmaktadır. Kullanıcı görüşleri, birçok sosyal medya platformunda veya web sitesinde yayınlanan reklamlardan etkilenmektedir. Arasu ve arkadaşları [114], sosyal medyanın bireylerin satın alma süreçlerine etkisi incelenmişlerdir. Çalışmalarında, müşteri satın alma tercihlerini tahmin etmek için Machine Learning integrated Social Media Marketing (ML-SMM) yöntemini önermişlerdir. Xu ve arkadaşları [115], internet kullanım davranışları ile akademik performans arasındaki ilişkiyi ortaya çıkarmak için DT, SVM ve NN makine öğrenme yöntemleri kullanmışlardır. Devaraj ve arkadaşları [116], doğal afetler sırasında sosyal medyada yayınlanan acil yardım tweet'lerini tespit etmek için bir yöntem önermişlerdir. Önerdikleri yöntemde, 2017 yılında Harvey kasırgasında yayınlanan 5 milyondan fazla tweet'i CNN, SVM ve

MLP yöntemlerini kullanarak sınıflandırmışlardır. Ngejane ve arkadaşları [117], sosyal medya platformları üzerinden çocukların istismar edilmemesi için bir yöntem önermişlerdir. Yöntemlerinde, sosyal medya sohbet günlüklerindeki zararlı konuşmaları tespit etmek için makine öğrenmesi yöntemlerini kullanan bir dijital adli süreç modeli kullanmışlardır.

İnternetin, kullanıcılara getirdiği bir diğer yenilik de e-ticarettir. Çevrimiçi alışveriş kullanıcılar için büyük kolaylıklar sunmaktadır. Ancak, aynı zamanda büyük sahtekârlıklara da neden olabilmektedirler. Chin ve arkadaşları [118], internette sahte satışların önüne geçmek için bir çalışma yapmışlardır. Çalışmalarında, e-ticarette sahte ürünleri tespit etmek için makine öğrenmesi ve IoT yöntemlerini birlikte kullanmışlardır. Koehn ve arkadaşları [119], tıklama verilerini kullanarak çevrimiçi alışveriş davranışını tahmin etmeye çalışmışlardır. Böylece pazarlama müdahalelerinin gerçek zamanlı olarak yapılmasını sağlamışlardır. Tıklama akışının sınıflandırılması için LSTM ve Gated Recurrent Unit (GRU) yöntemlerini kullanmışlardır. Mubarak ve arkadaşları [120], Massive Open Online Courses (MOOC) platformlarında öğrenim gören öğrencilerin performansını belirlemek için bir yöntem önermişlerdir. Yöntemlerinde, video tıklama akışlarından çıkarılan özellikler LSTM ile sınıflandırılmıştır. Mokhtar ve arkadaşları [121], internet verilerini verimli bir şekilde depolamak için insan belleğinden ilham alan NetMem ağ belleğini önermişlerdir. Li ve arkadaşları [122], sahte haberleri tespit etmek için kendi kendine öğrenen yarı denetimli bir derin öğrenme ağı tasarlamışlardır. Kaliyar ve arkadaşları [123], sahte haberleri tespit etmek için Bidirectional Encoder Representations from Transformers (BERT) tabanlı derin öğrenme yaklaşımını önermişlerdir.

Kullanıcıların internette sıklıkla takip ettikleri kanallardan biri de haber siteleridir. Hakak ve arkadaşları [124], internetteki sahte haberleri tespit etmek için bir yöntem önermişlerdir. Önerdikleri yöntemin ilk adımında, ensemble sınıflandırıcısı kullanılarak öznelilikler seçilmiştir. Daha sonra elde ettikleri veri setini DT, RF ve Extra Tree Classifier(ET) sınıflandırıcıları ile sınıflandırmışlardır. Çalışmalarını Liar ve ISOT veri setleri üzerinde gerçekleştirmişlerdir. Agarwal ve arkadaşları [125], metinsel bilgilerden öznelilikler çıkararak ve güvenilirlik puanları üreterek sahte haberleri tespit etmek için bir yöntem önermişlerdir. Yüksek doğruluk elde etmek için SVM, CNN, LSTM, KNN algoritmalarını kullanmışlardır. Mulahuwaish ve arkadaşları [126], büyük veriyi web uygulamalarında hızlı ve doğru bir şekilde işleyebilmek için haberleri iş, teknoloji, bilim, sağlık ve eğlence olarak kategorize etmişlerdir. Elde edilen haberler SVM, KNN, DT ve LSTM yöntemleri ile sınıflandırılmıştır. Hosni ve arkadaşları [127], sosyal ağlarda kötü niyetli söylentileri ve haberleri engellemek için dinamik engelleme dönemi yaklaşımını kullanmışlardır. Wu ve arkadaşları [128], Çin'in en büyük sosyal medya sitelerinden biri olan Sina Weibo'daki sosyal botları tespit etmek için Deep Neural Networks and Active Learning (DABot) çerçevesi önermişlerdir. Shang ve arkadaşları [129], youtube gibi çevrimiçi video paylaşım platformlarında tıklama tuzağı videolarını algılamak için Online Video Clickbait Protector (OVCP) yöntemini

geliştirmişlerdir. Kumar ve arkadaşları [130], metin ortamında siber zorbalığın tespiti için bir hibrit model Bi-GRU-Attention-CapsNet (Bi-GAC) yöntemi önermişlerdir.

STS sistemlerinin kullanıldığı bir diğer kritik alan ise sağlık hizmetleridir. Sağlık sektörleri, hasta bakımını iyileştirecek, benzeri görülmemiş hızlı bir dijital dönüşüm geçirmektedir [131]. Sağlık hizmetlerinin her yerde dijitalleştirilmesinin ürünü olan tıbbi nesnelerin interneti (Internet of Medical Things - IoMT), tıbbi cihazlar, giyilebilir sensörler, sağlık uygulamaları ve bunları kurumsal sunucu ve bulut tabanlı hasta düzeyinde verilerle bağlayan kablosuz iletişim ağından oluşur. Birbirine bağlı sağlık cihazlarının beraberindeki yazılımların, işletim sistemlerinin ve ağların yaygınlaşması, güvenlikten taviz verme riskini artırmaktadır. Wi-Fi özellikli kontrollü infüzyon pompalarından akıllı manyetik rezonans görüntüleme makinelerine kadar internete bağlı tıbbi ekipmanın büyük kısmı, internet saldırılarına dayanacak şekilde tasarlanmamıştır. Bu da onları izinsiz girişlere maruz bırakmaktadır. Özellikle, düşük güçlü ve düşük maliyetli sensörlerdeki gelişmeler, kimlik doğrulama, şifreleme, yetkilendirme ve diğer ağ güvenliği önlemlerinden yoksundurlar [132]. Ayrıca, kablosuz bağlantılar aracılığıyla son derece hassas tıbbi verileri bulut sunucularına aktaran IoMT sağlık cihazlarının gelişen dağıtımı, siber saldırı yüzeyini genişletmekte ve sağlık sistemlerini siber saldırılara maruz bırakmaktadır. Bu gelişmeler, IoMT'ye yönelik siber güvenlik tehditlerini şiddetlendirmektedir.

Rani ve arkadaşları [133], bir kimlik doğrulama algoritması kullanarak güvenli veri işleme ve hastalık tahmini için SVM tabanlı bir makine öğrenme algoritmasını içeren, ancak saldırı tespiti içermeyen, uzaktan hasta izleme için akıllı bir bulut tabanlı kablosuz vücut sensörü ağını tanımlamışlardır. Alabdulatif ve arkadaşları [134], bulut depolama sistemi ve akıllı tahmin modelinden oluşan hastaların yaşamsal belirtilerini kullanan bulut tabanlı bir gerçek zamanlı anormallik tahmin çerçevesi önermişlerdir. Gizliliğin korunması için geleneksel veri güvenliği önlemleri vurgulanmış, ancak güvenlik ihlallerini tahmin etmek için makine öğrenmesi gibi yenilikçi teknolojiler benimsenmemiştir. Chakraborty ve arkadaşları [135], güvenliği artırması beklenen blok zincir teknolojisine dayalı bir sağlık sistemi önermişlerdir. Yukarıdaki örnekler, IoMT'de kaybedilen siber koruma fırsatlarını temsil etmektedir, çünkü makine öğrenmesi, kötüye kullanım ve anormallik tespitini kapsayan hibrit ağ analizi için siber güvenlik uygulamalarında uzun yıllardır kullanılmaktadır [136]. Anormallik tespiti, ağdaki herhangi bir garip etkinliği tanımlarken, kötüye kullanım tespiti, imzalarını kullanarak bilinen saldırıları arar. Daha önce keşfedilmemiş saldırılar için en umut verici strateji, güvenlik açıklarını yönetmek için makine öğrenmesini kullanmaktır. Hady ve arkadaşları [63], izinsiz giriş tespit, saldırılar ve sunucudan oluşan sızdırma tehditlerini ve veri değişikliklerini tespit etmek için bir sağlık hizmeti güvenlik sistemi geliştirmişlerdir. 14.000 normal veri ve 2.000 saldırı verisinden oluşan bir veri seti, RF, KNN, DVM ve ANN kullanarak eğitilmiş ve %7 ila %25 arasında iyileştirmeler sağlanmıştır.

Saldırı tespiti ve kullanıcı internet davranış analizi hem endüstride hem de akademide yoğun ilgi görmektedir. Günümüzde hem saldırı tespit başarımının artmasında hem de kullanıcı internet davranış analizinde makine öğrenmesi yöntemlerinin rolü büyüktür. Günümüzde hemen her alanda karşımıza çıkan makine öğrenmesi yöntemleri bu tez çalışmasında önerilen saldırı tespit sistemlerinde ve kullanıcı internet davranış analizinde kullanılmıştır.

1.4. Makine Öğrenmesi ve Yöntemleri

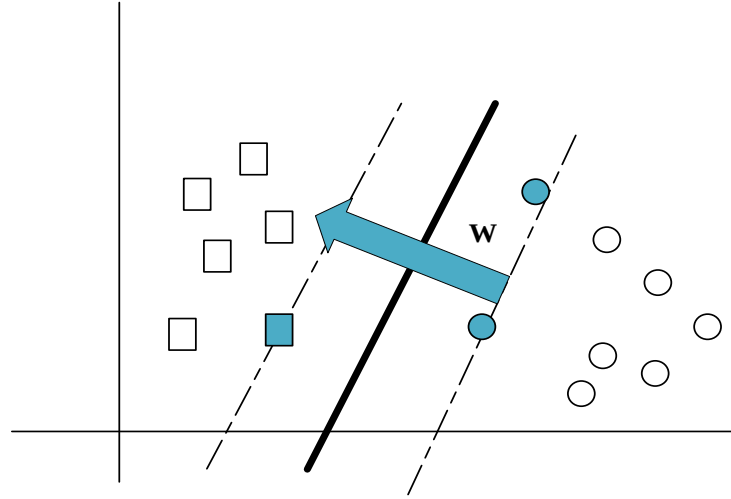
Datareportal 2022 istatistiklerine göre dünya nüfusunun %62,5'i internet, %58,4'sı sosyal medya kullanmaktadır [1]. Teknoloji kullanımının bu seviyeye çıkması, sistemlerin işleyişinde hızlanmayı gerektirmektedir. Verilerin elde edilmesi ve bu verilerin işlenerek sonuca varılması insan eliyle yapıldığında uzun zaman almaktadır. Bu noktada yapay zekânın alt dallarından biri olan makine öğrenmesi yöntemleri verilerin detaylı bir şekilde ve kısa bir sürede analiz edilmesinde önemli rol üstlenmektedir. Makine öğrenmesi, insanlara ait öğrenme şeklini temel alarak, istatistiksel ve matematiksel işlemler kullanarak veriler üzerinde tahminler yaparak sonuca ulaşmaya çalışan bir yapay zeka dalıdır. Makine öğrenmesi, her geçen gün büyüyen ve karmaşıklığı artan verilerin analizinde çok önemli bir bileşendir. Bu tez çalışmasında, başarımlarını artırmak amacıyla birden fazla makine öğrenmesi yöntemi kullanılmıştır.

1.4.1. Destek Vektör Makineleri (Support Vector Machine - SVM)

İstatistiksel yöntemlere ve yapısal risk minimizasyonuna dayalı güçlü bir sınıflandırıcıdır. Uygulaması kolay ve iyi performans sağladığı için sınıflandırma işlemlerinde sıklıkla kullanılmaktadır. DVM, örnekleri sınıflandırmak için çok boyutlu bir hiper düzlem kullanır ve bu düzlemdeki noktalar arasında bir sınır çizer. Sınır, veri kümelerine en uzak mesafede olmalıdır. Yeni eklenen veriler bu sınır çizgisine bakılarak sınıflandırılır [43,137,138]. DVM'de öğrenme verileri $x_i \in R^d$, $i = 1 \dots n$, $y_i \in \{-1, +1\}$ olduğunda, $\{x_i, y_i\}$ ile etiketlenirse, Şekil 1.8'de verilen iki boyutlu uzayda iki sınıflı veri seti örneği oluşturulur. Şekil 1.8'de kalın çizgi ayırıcı hiper düzlemi gösterirken w normal vektörü gösterir. Hiper düzlemin pozitif ve negatif örnekleri ayırdığını varsayarsak, iki sınıflı lineer sınıflandırıcı probleminde hiper düzlemin normal vektörü w ve ofset değeri b olur. Buna göre karar limiti $w^T x + b = 0$ doğrusudur. Bu durumda Denklem 1.1 ve Denklem 1.2'deki koşullar sağlanmalıdır [39,139,140].

$$w^T x_i + b \geq 1 \quad (1.1)$$

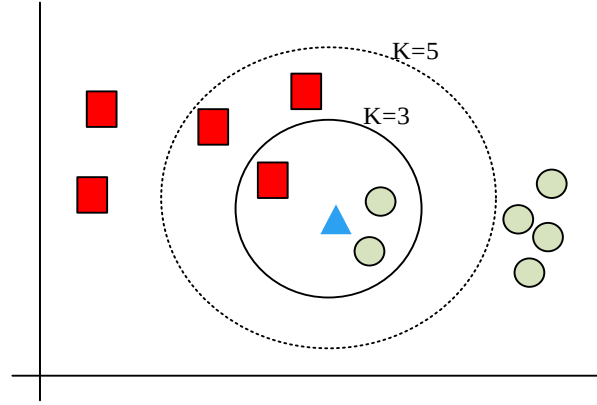
$$w^T x_i + b \leq -1 \quad (1.2)$$



Şekil 1.8. Destek vektör makineleri

1.4.2. K- En Yakın Komşu Sınıflandırıcısı

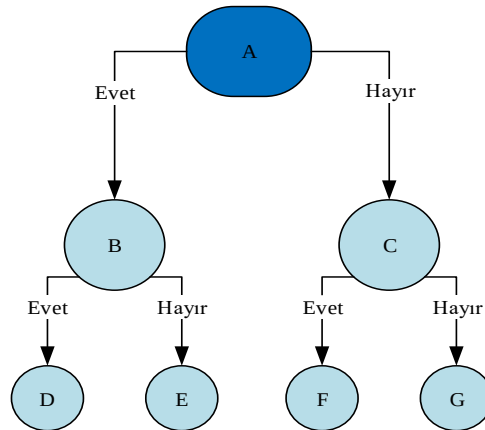
K-En Yakın Komşu (K-Nearest Neighbors – KNN), sınıflandırma süreçlerinde sıklıkla kullanılan denetimli bir makine öğrenmesi yöntemidir. Bu algoritma mevcut verileri depolar ve yeni gelen verileri mevcut verilere olan uzaklıklarına göre sınıflandırır. Yeni gelen verinin mevcut veriye uzaklığı hesaplanır ve k 'nın yakın komşuluğu kontrol edilir. Veriler, kendisine en yakın olan komşusunun sınıfına dahil edilir. Verinin komşularından uzaklığı hesaplanırken Euclid, Manhattan gibi fonksiyonlar kullanılır. Şekil 1.9'da örnek bir KNN çizimi verilmiştir. Buna göre kare ve dairelerden oluşan kümelerden ek bir veri geldiğinde bu verinin komşularına olan uzaklığı ölçülmektedir. KNN makine öğrenmesi yönteminde, K değeri sonucu doğrudan etkiler. K değerinin 1 olması aşırı öğrenme olasılığını artırır. Eğer K değeri çok büyük seçilirse algoritmanın aşırı öğrenme (overfit) olasılığı çok yüksek olacaktır. Bu nedenle en uygun K değerinin seçilmesi bu algoritma için kritik öneme sahiptir. K değerinin önemi Şekil 1.9'da gösterilmektedir. Eğer $K=3$ seçilirse algoritma yeni gelen mavi renkli üçgen verisini yeşil renkli daire olarak sınıflandıracaktır. Eğer $K=5$ seçilirse algoritma yeni gelen mavi renkli üçgen verisini kırmızı renkli kareler sınıfına alacaktır [137,141–143].



Şekil 1.9. KNN algoritması

1.4.3. Karar Ağaçları Sınıflandırıcısı

Karar ağaçları (Decision Tree - DT), eğitimleri ve testleri hızlı ve etkili olduğu için sınıflandırmada sıklıkla kullanılan makine öğrenme yöntemlerinden biridir. Karar ağaçları ile sınıflandırma iki adımda gerçekleştirilir. İlk adımda ağaç oluşturulur. İkinci adımda, bu ağaç yapısından sınıflandırma kuralları elde edilir. Karar ağaçları temel olarak kökler, dallar ve yapraklardan oluşur. Karar verme süreci kökten yapağa doğrudur. Kök hücrelerden sonra düğümler bulunmaktadır. Her bir gözlem vektörü düğümlere göre sınıflandırılır. Modelin karmaşıklığı düğüm sayısı ile doğru orantılıdır [43], [144]. Basit bir karar ağacı yapısı Şekil 1.10'da verilmiştir. Buna göre A kök düğümü, B ve C dalları, D, E, F ve G daireleri de yaprakları göstermektedir.

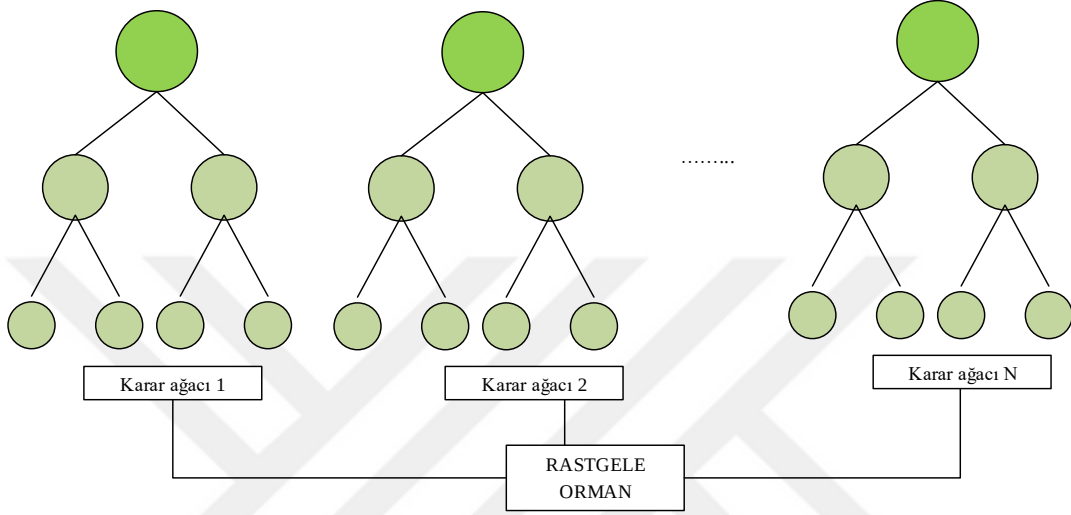


Şekil 1.10. Karar ağaçları algoritması

1.4.4. Rastgele Orman Sınıflandırıcısı

Aşırı öğrenme, karar ağaçlarının en büyük problemlerinden birisidir. Rastgele Orman (Random Forest - RF) algoritması bir topluluk öğrenme metodu olup, sınıflandırma başarımını

artırmak için birden fazla karar ağacı üretir. Karar ağaçları algoritmasındaki aşırı öğrenme problemini çözmek için, veri setinden ve öznitelik setinden farklı alt setleri rastgele bir şekilde seçer ve bunları eğitir. Böylelikle birden fazla belki 10 belki de 100'lerce karar ağacı oluşur. Meydana gelen her karar ağacı bireysel tahminde bulunur. Tahminler arasında en çok oy alan tahmin sınıflandırma sonucunu verir [145]. Rastgele orman algoritması Şekil 1.11'de verilmiştir.



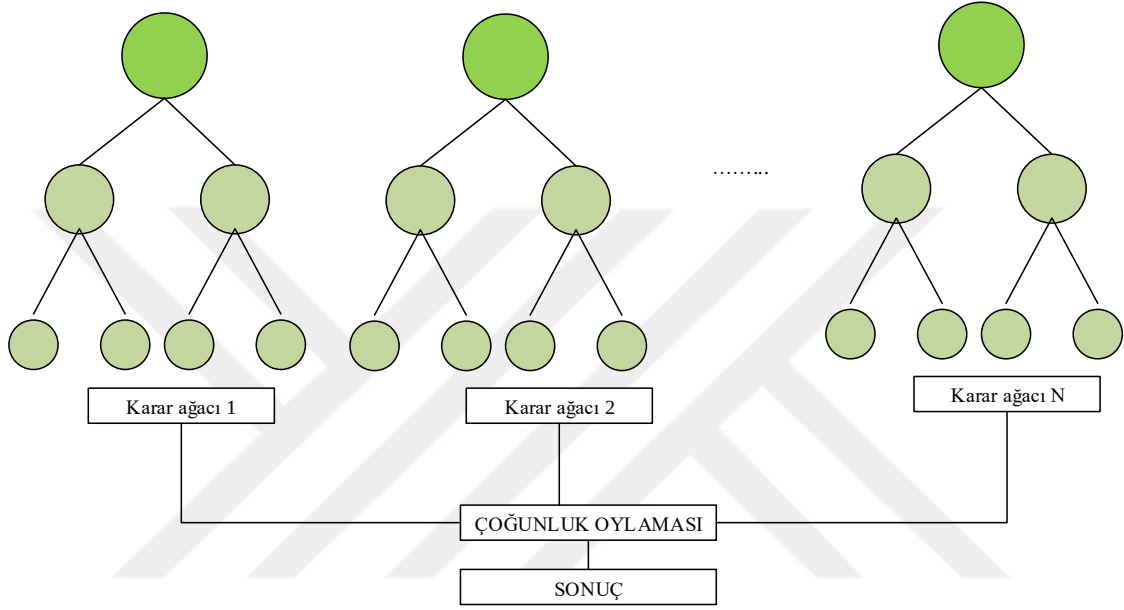
Şekil 1.11. Rastgele orman algoritması

1.4.5. Ekstra Ağaçlar Sınıflandırıcısı

Rastgele orman algoritması gibi, Ekstra Ağaç (Extra Tree) sınıflandırıcısı da belirli kararları ve veri alt kümelerini rastgele seçer. Böylece aşırı öğrenme sorununun önüne geçer. Ekstra Ağaçlar, rastgele orman gibi birden çok ağaç oluşturur ve özellikleri rastgele alt kümeler kullanarak düğümlere ayırır. Ancak, rastgele orman algoritması gibi gözlemleri önceden yüklemes ve düğümler en iyi bölünmeler ile değil, rastgele bölünmeler ile bölünür. Ekstra Ağaç algoritması, klasik bir şekilde yukarıdan aşağıya regresyon ağacı oluşturur. Bu algoritma ile diğer ağaç tabanlı kümeleme yöntemleri arasında iki temel fark vardır. Farklardan ilki, düğümleri ayırarak rastgele kesme noktalarını seçer ve tüm öğrenme modelini kullanarak ağacı büyütür. İkinci fark, ağaç tahminlerinin sınıflandırma problemlerinde çoğunluk oyu ve regresyon problemlerinde aritmetik ortalama ile birleştirir [146].

Ekstra Ağaç algoritması M , K ve n_{min} parametrelerinden oluşmaktadır. M ağaç sayısını, K her düğümde rastgele seçilen özelliklerin sayısını ve n_{min} düğümü bölmek için gereken minimum örnek boyutunu temsil eder. Ekstra Ağaç algoritması, tüm özelliklerden bölünmüş düğümlere rastgele K özellikleri seçer ve her özellik için rastgele bir bölme noktası belirler. Bir sonraki adımda, her özelliğe bir puan verilir. Seçilen özellik, düğümü bölmek için maksimum varyans azaltma özelliklerine sahiptir. Ekstra Ağaç algoritması, bir düğümdeki örnek sayısını kullanarak düğüm

bölünmesinin ne zaman durdurulacağını belirler. n_{min} , örnek sayısından büyük olduğunda, düğüm bölünmeyi durdurur ve yaprak oluşur. Son adımda, M ağaçlarının tahminleri ve çoğunluk oyu toplanır. Böylece nihai tahmin sonucu elde edilir. Bu tez çalışmasının üçüncü bölümünde en iyi özelliklerin seçimi için Ekstra Ağaçlar algoritmasını kullanılmıştır. Bu algoritmanın yapısında, kesme noktalarının rastgele seçilmesi ve kümülatif ortalama ile varyansın azaltılması gibi çeşitli avantajlar vardır [147]. Şekil 1.12’de Ekstra Ağaç algoritması verilmiştir.



Şekil 1.12. Ekstra Ağaçlar algoritması

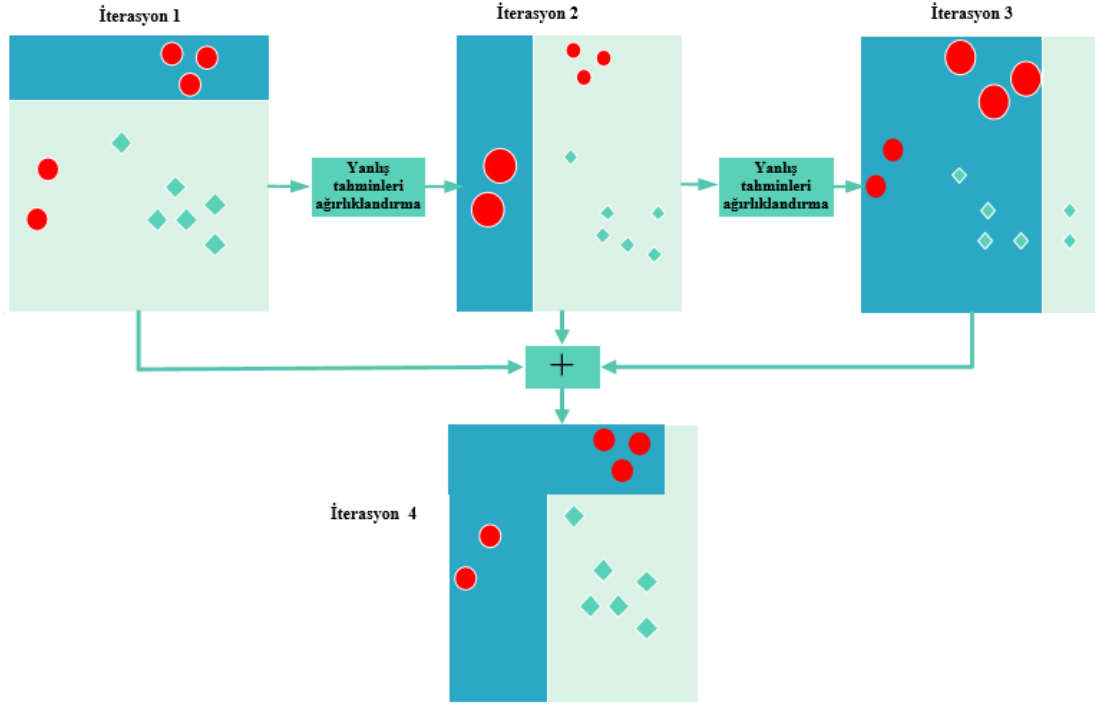
1.4.6. Bagged Tree Sınıflandırıcısı

Aşırı öğrenmeyi önlemek ve başarımlarını arttırmaya yönelik geliştirilen ağaca dayalı yöntemdir. Önyükleme toplaması (Bootstrap aggregating) ifadesinin kısaltması olan Bagging yönteminde örnek veri setinden yerine koymak şartıyla tekrarlı bir şekilde örnekler çekilir ve yeni ağaçlar oluşturulur. Oluşturulan bu yeni ağaçlardan bir ağaç topluluğu ortaya çıkar. Oluşturulan her ağaç modelleme için kullanılır ve tüm ağaçların tahmini alınır. Bütün tahminler değerlendirilir. Problem bir regresyon problemi ise tüm ağaç tahminlerinin ortalaması, problem bir sınıflandırma problemi ise ağaçlar arasında oylama yapılır ve sonuç bulunur [148].

1.4.7. AdaBoost Sınıflandırıcısı

Boosting algoritmaları, makine öğrenmesi modellerinde tahmin performansını artırmak için sıklıkla kullanılan algoritmalar. Güçlendirme yöntemleri, zayıf öğrenenleri bir araya getirerek güçlü bir öğrenen yaratma fikrine dayanır. Boosting algoritmalarında bir sonraki ağaç, önceki ağaçtan gelen hatayı en aza indirmeye çalışır. Bu, boosting algoritmalarını güçlü kılan en büyük

faktörlerden biridir. Zayıf öğrencilerin eksikliklerinin hangi yöntemler ile güçlendirildiği, boosting algoritmaları arasındaki farkı belirler. 1996 yılında Robert Schapire ve Yoav Freund tarafından geliştirilen Adaboost algoritması ilk başarılı boosting algoritmasıdır [149]. Boosting algoritmalarının temeli olan Adaboost algoritmasının akış şeması Şekil 1.13'te verilmiştir.



Şekil 1.13. Adaboost algoritması

İlk boosting algoritması sayılan Adaboost algoritmasında eğitim seti önce zayıf bir öğrenci ile eğitilir. Bir sonraki iterasyonda ilk iterasyonda yanlış öğrenilen eğitim verileri daha fazla ağırlıklandırılır. Bu işlem belirlenen iterasyona kadar devam eder. Son adımda, nihai karar sınırlarını oluşturmak için sonuçlar birleştirilir. Adaboost, Boosting algoritmalarının temelidir. Adaboost sınıflandırıcısının sözde kodu Tablo 1.3'te verilmiştir [150,151]. Şekil 1.13'e göre ilk yinelemede sınıflandırıcı daireleri sınıflandırır. Ancak yanlış sınıflandırılan bazı kırmızı daireler bulunmaktadır. İkinci iterasyonda, yanlış sınıflandırılan dairelere daha fazla ağırlık verilir ve doğru sınıflandırılan dairelerin ağırlıkları azaltılır. İşlemden sonra sınıflandırıcı yeniden eğitilir. İkinci yinelemede, yanlış sınıflandırılan dairelere daha fazla ağırlık verilir ve daha büyük daireler kullanılırken, doğru sınıflandırılan veriler daha az ağırlık ve küçültülmüş boyutlarla gösterilmiştir. İkinci yinelemenin bir sonucu olarak, sınıflandırıcı üç kırmızı noktayı yanlış sınıflandırmıştır. Üçüncü yinelemede, güncellenen her veri noktası yeniden eğitilir. Üçüncü iterasyonda iki yeşil karenin yanlış sınıflandırıldığı görülmektedir. Son olarak, 3 iterasyon üzerinden eğitilen sınıflandırıcıların birleştirilmesiyle nihai sonuç alınmıştır.

Tablo 1.3. Adaboost algoritması sözde kodu

1: Ağırlık Katsayısı verileri $w_i = \frac{1}{N}, i = 1, 2, 3 \dots N$
2: Temel Sınıflandırıcı Seçimi (G)
3: $m = 1, \dots, M$ $gm(x)$ Sınıflandırıcısını eğitim verilerini w_i ağırlıklarını kullanarak düzenleyin. e_m değerini hesapla $e_m = \sum_{n=1}^N W_n^m I(G_m(X_n) \neq t_n) / \sum_{n=1}^N W_n^m$ a_m değerini hesapla $a_m = \ln(1 - e_m / e_m)$ Ağırlık katsayısı güncellemesi $W_n^{m+1} = W_n^m \exp\{a_m I(G_m(X_n) \neq t_n)\}$
4: Nihai Çıktı $G_m(x) = \text{sign}(\sum_{m=1}^M a_m G_m(x))$

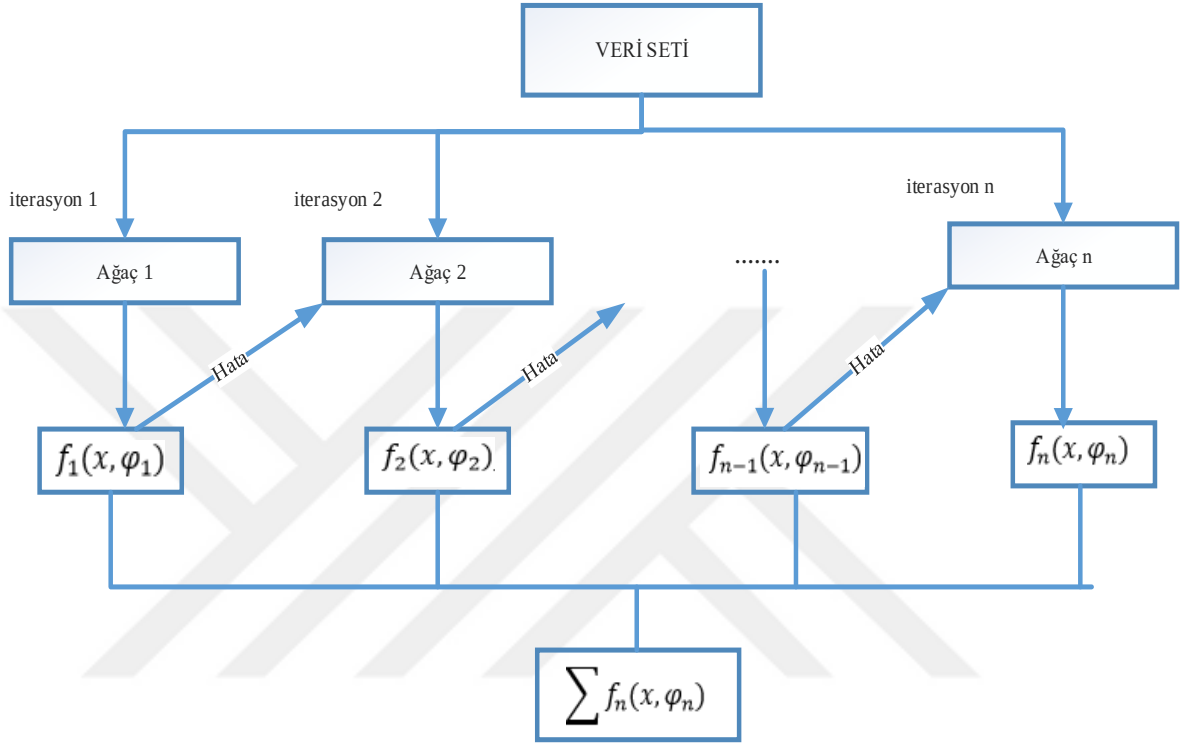
1.4.8. Gradient Boosting Machine (GBM) Sınıflandırıcısı

Bir topluluk makine öğrenmesi algoritmasıdır. 2001 yılında Friedman tarafından geliştirilmiştir. AdaBoost algoritmasının hem regresyon hem de sınıflandırma işlemlerine uyarlanabilen geliştirilmiş halidir. GBM, Adaboost'un yaptığı yanlış sınıflandırılmış örnek ağırlıklarını sırayla artırmak yerine, tek bir tahminsel model formunda olan modeller serisi oluşturur. Seri içindeki bir model bir önceki modelin tahmin hatalarının (artıklarının) üzerine kurulur. GBM genellikle öğrenici olarak karar ağaçlarını kullanır. Bir önceki karar ağacının eğitim hatalarından yeni bir karar ağacı oluşturulur. Hata minimuma indirildiğinde işlem sonlandırılır. Zayıf öğrenicileri güçlü bir öğrenici meydana getirmek için birleştiren GBM algoritması, yükseltme tekniği kullanır. GBM, her yeni ağacı optimize ederek, onu güçlü bir öğrenci haline getirir. Böylece sınıflandırıcının daha az yanlış alarm ve daha yüksek performansa sahip olmasını sağlar [152–154].

1.4.9. EXtreme Gradient Boosting (XGBoost) Sınıflandırıcısı

Hız ve tahmin performansını artırmak için GBM'nin optimize edilmiş sürümüdür. 2014 yılında Tianqi Chen tarafından geliştirilmiştir. R, Python, Hadoop, Scala, Julia gibi platformlarla kullanılabilir. Windows ve Linux üzerinde otomatik olarak paralel işleme yapabilir ve bu yüzden geleneksel GBM'e göre çok daha hızlı çalışır. XGBoost, Google, Tencent, Alibaba gibi Terabaytlarca veri içeren platformlarda yaygın olarak kullanılmaktadır. Şekil 1.14'te XGBoost algoritmasının genel yapısı verilmiştir. XGBoost, aşırı uyumluluğu azaltan ve genel başarımlarını artıran düzenleme ayarları içerir. Böylece doğruluk, fizibilite ve verimlilik sağlar. GBM, sınıflandırıcı performansını artırmak için, zayıf sınıflandırıcıları boosting yöntemiyle

birleştirir. GBM ve XGBoost algoritmaları temelde aynı prensiplere sahip olmalarına rağmen XGBoost farklı düzenleme teknikleri ile ağaç karmaşıklığını kontrol eder. XGBoost seviye bazlı büyüme (Level-wise tree growth) stratejisini kullanır. Böylece daha iyi performans değerlerine ulaşır [155–157].



Şekil 1.14. XGBoost algoritması

Şekil 1.14'te gösterildiği gibi, Gradient Boosting'in her iterasyonunda hatalar, önceki tahmin ediciyi düzeltmek için kullanılır. XGBoost, Denklem 1.3'te verilen amaç fonksiyonunu oluşturmak için kayıp fonksiyonuna düzenleme fonksiyonu ekler.

$$j^{(t)} = \sum_{i=1}^n L(y_i, \hat{y}_i) + \sum_{k=1}^K \Omega(f_k) \quad (1.3)$$

Denklem 1.3'te t 'inci iterasyondaki amaç fonksiyonu verilmiştir. Denklemdeki $j^{(t)}$ amaç fonksiyonunu, n tahmin sayısını, $L(y_i, \hat{y}_i)$ i 'inci örneğin eğitim hatasını, Ω ise düzenleme fonksiyonunu göstermektedir. Boosting algoritmaları öğrenici olarak genelde karar ağaçlarını kullanılır. Modelin çıktısı olan ve Denklem 1.4'te verilen $\hat{y}_i$, n ağaçtan oluşan bir F fonksiyonu ile oylanır veya ortalaması alınır.

$$\hat{y}_i = \sum_{n=1}^n f_n(x_i), f_n \in F \quad (1.4)$$

Bir karar ağacı için $\Omega(f_k)$ düzenleme terimi Denklem 1.5'te tanımlanmıştır.

$$\Omega(f_k) = \gamma T + \frac{1}{2} \alpha \sum_{j=1}^T w_j^2 \quad (1.5)$$

Burada γ her yaprak için karmaşıklığı ifade ederken, T karar ağacındaki yaprak sayısını, α ceza parametresini ifade etmektedir. Ayrıca, v yapraklar üzerindeki puanların vektörüdür. XGBoost'taki kayıp fonksiyonunun, GBM'deki birinci dereceden yerine ikinci dereceden Taylor açılımı alınır. Kayıp fonksiyonunun ortalama kare hatası (mean square error - MSE) olduğu varsayıldığında, amaç fonksiyonu Denklem 1.6'daki gibi türetilir [158].

$$j^{(t)} \approx \sum_{i=1}^n \left[g_i v_{q(x_i)} + \frac{1}{2} (h_i v_{q(x_i)}^2) \right] + \gamma T + \frac{1}{2} \alpha \sum_{j=1}^T v_j^2 \quad (1.6)$$

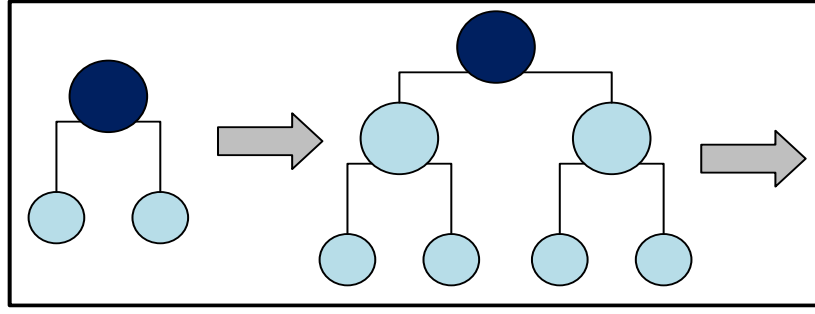
Denklem 1.6'daki sabitler kaldırıldığında $q(\cdot)$, ilgili yaprağa veri noktası atayan bir fonksiyondur. g_i ve h_i MSE kayıp fonksiyonunun birinci ve ikinci türevidir. Kayıp fonksiyonu, her bir veri örneği için kayıp değerinin toplamı ile belirlenir. Her bir veri örneği yalnızca bir yaprak düğüme karşılık geldiğinden, kayıp işlevi her bir yaprak düğüm için kayıp değerinin toplamı ile de ifade edilebilir. Böylece Denklem 1.7 elde edilir [158].

$$j^{(t)} \approx \sum_{j=1}^T \left[(\sum_{i \in I_j} g_i) v_j + \frac{1}{2} (\sum_{i \in I_j} g_i + \alpha) v_j^2 \right] + \gamma T \quad (1.7)$$

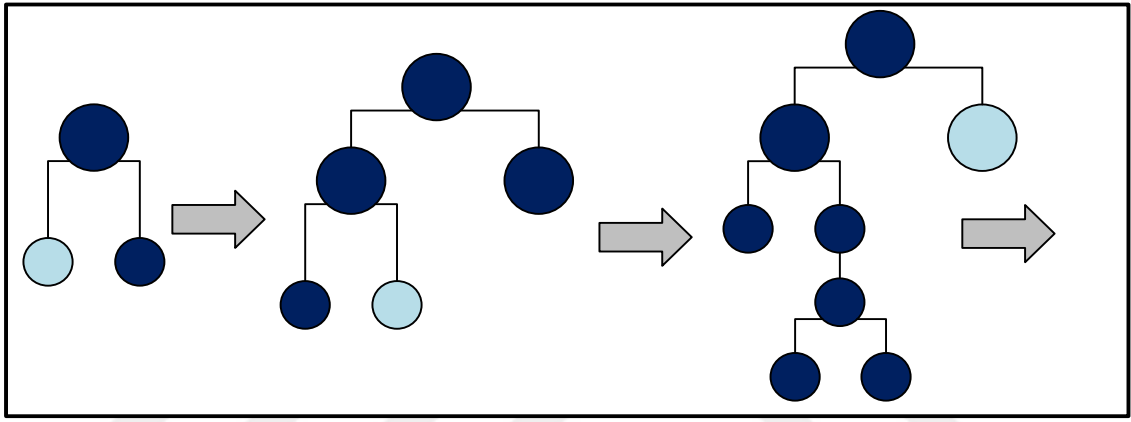
Denklemdaki I_j , j 'inci yaprak düğümündeki tüm veri örneklerini temsil eder. Bu nedenle, amaç fonksiyonunun optimizasyonu, ikinci dereceden bir fonksiyonun minimumunu bulma problemine dönüştürülebilir. Başka bir deyişle, karar ağacında belirli bir düğüm bölünmesinden sonra, model performansının değişimi amaç fonksiyonuna göre değerlendirilebilir. Bu düğüm bölünmesinden sonra karar ağacı modelinin performansı iyileşirse, bu değişiklik benimsenecek, aksi takdirde bölme durdurulacaktır. Ayrıca, amaç fonksiyonunu optimize ederken düzenlileştirme nedeniyle, bir tahmine dayalı sınıflandırıcı fazla uyumluluğa karşı eğitilebilir.

1.4.10. Light GBM (LGBM) Sınıflandırıcısı

XGBoost algoritmasının eğitim süresi performansını artırmak için 2017 yılında Microsoft tarafından geliştirilmiştir. Veri setindeki gözlem ve değişken sayısı çok fazla olduğunda XGBoost sınıflandırıcısının eğitim süresi uzar. Light GBM, XGBoost'taki bu tür eksikliklerin üstesinden gelmek için geliştirilmiştir. LGBM, seviye bazlı büyüme (Level-wise tree growth) stratejisi yerine yaprak bazlı büyüme (leaf-wise tree growth) stratejisi kullanır. XGBoost algoritmasının kullandığı level-wise büyüme yöntemi ve LGBM'nin kullandığı leaf-wise büyüme yöntemi Şekil 1.15 ve Şekil 1.16'da özetlenmiştir [159,160].



Şekil 1.15. XGBoost level-wise büyüme yöntemi



Şekil 1.16. LGBM leaf-wise büyüme yöntemi

Şekil 1.15 ve Şekil 1.16’da XGBoost ve LGBM algoritmalarının kullandıkları karar ağaçlarının büyüme Level-wise büyüme stratejisi yatayda gelişir ve bir alt seviyeye inmek için bulunulan seviyenin açılması gerekir. Leaf-wise büyüme yöntemi ise gidebildiği kadar diklemesine ağacı açar, maksimum derinlik elde edildiğinde yukarıdan itibaren diğer dalı diklemesine açmaya başlar.

1.4.11. Category Boosting (CatBoost) Sınıflandırıcısı

Kategorik değişkenleri otomatik olarak sınıflandırabilen başarılı bir GBM türüdür. Yüksek bir öğrenme hızına sahiptir. Sayısal, kategorik ve metin verileriyle başarılı bir şekilde çalışır. GPU desteği ve görselleştirme seçenekleri sunar. Bu özellikleri ile diğer algoritmalarından farklıdır [161]. CatBoost algoritması, hedef tabanlı istatistikler, one_hot_max_size (OHMS) ve permütasyon teknikleri ile kategorik sınıflara odaklanır. CatBoost, her özellik için aşağıdaki adımları kullanır:

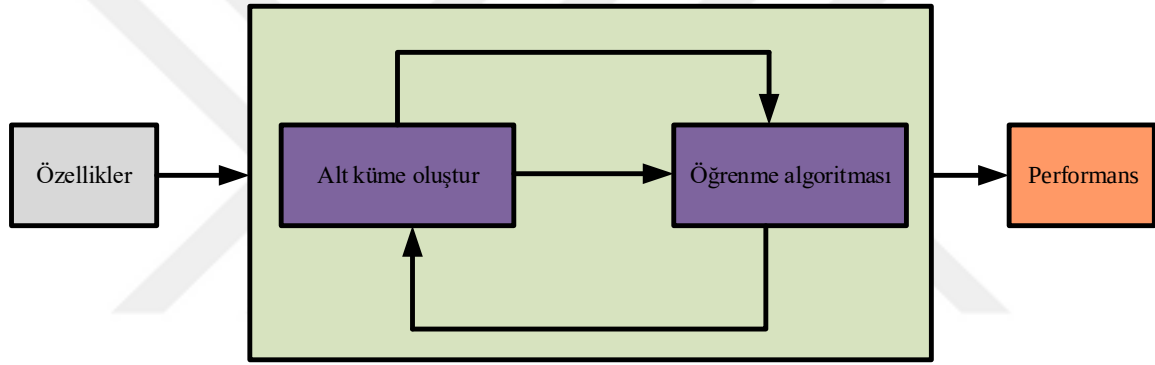
- Kayıtları rastgele alt kümelerine ayırır
- Etiketleri tam sayılara dönüştürür
- Kategorik özellikleri aşağıdaki gibi sayısal değere dönüştürür.

$$avgTarget = \frac{countinclass + prior}{totalcount + 1} \quad (1.8)$$

Denklem 1.8’de *countinclass*, belirli bir kategorik özellik için hedef sayısıdır, *totalcount* nesnelerin toplam sayısıdır ve *prior*, başlangıç parametreleridir [160].

1.4.12. Özyinelemeli Özellik Eleminasyonu

Özyinelemeli özellik eleminasyonu (Recursive Feature Elimination - RFE) bir Wrapper özellik seçim yöntemidir. Wrapper yöntemleri, makine öğrenmesi algoritmaları ile seçtiği özelliklerin performansını kontrol eder. Böylelikle makine öğrenmesi algoritması için en uygun özellikler bulunur. Wrapper yöntemleri genellikle filtre yöntemlerinden daha iyi tahmin doğruluğu sağlar [162]. RFE algoritması Şekil 1.17’de verilmiştir.



Şekil 1.17. RFE algoritması

RFE yöntemi, belirlenen en uygun öznitelik alt kümesini oluşturana kadar, özyinelemeli olarak tüm öznitelik kümesinden adım adım en kötü performansa sahip öznitelikleri eleyen bir özellik seçim algoritmasıdır. Algoritma her yinelemede özelliklerin önemlerini ölçer ve ölçüm sonucunda en düşük değerli öznitelikleri kaldırır.

1.4.13. Lojistik Regresyon

Doğrusal sınıflandırma problemlerinde sıklıkla kullanılan yöntemlerden biridir. Lojistik regresyon (Logistic regression – LR) istatistiksel bir yöntem olup, bir ya da birden fazla değişkeni bulunan veri setlerinin analizinde kullanılır. Kategorik ya da sayısal verilerin sınıflandırılmasında kullanılan Logistic Regression algoritması Denklem 1.9’da tanımlanmıştır.

$$P(x) = \frac{\exp(\mu x + b)}{1 + \exp(\mu x + b)} \quad (1.9)$$

Denklemdaki $x \in R^n$ giriş özelliklerini, μ ağırlık, b ofset değeri, $\mu \cdot x$ ise matrislerin nokta çarpımlarını ifade etmektedir. Logistic regression, tahminlerini, örnek değerlerin gözlem olasılığını yükseğe çıkaran parametrelerini seçer ve x' e atar [163].

1.4.14. Çok Katmanlı Algılayıcı

XOR problemini çözmek için geliştirilen çok katmanlı algılayıcı (Multilayer Perceptron - MLP) modeli hatayı ağı yaydığı için geri yayımlı model (Back Propagation Model) olarak da bilinir. Öğrenme metodu olarak delta öğrenme metodunu kullanan MLP algoritması, sınıflandırma problemlerinde oldukça etkilidir. MLP, giriş, ara (gizli) ve çıkış olmak üzere üç katmandan oluşur. Giriş katmanı gelen verileri alarak gizli katmana gönderir. Problemin türüne göre belirlenen gizli katman sayısı en az 1 olmalıdır. MLP'deki her işlem elemanı nöron olarak adlandırılır ve her nöron bir sonraki katmanda bulunan tüm nöronlara bağlıdır. Dolayısıyla her nöronun girdileri ve çıktıları vardır. Nöronların girdileri ağırlıklandırıldıktan sonra aktivasyon fonksiyonu aracılığıyla bir çıktıya dönüştürülür. Her katmanın çıkışı bir sonraki katmanın girişi olmaktadır. Bu durum MLP'nin çıkışında son bulur. Çıkış katmanı önceki katmanlardan gelen verileri işleyerek, çıkış katmanında bulunan eleman sayısına göre ağı çıkışını belirler [164,165].

2. SİBER GÜVENLİK SALDIRI TESPİTİ İÇİN MAKİNE ÖĞRENMESİ YÖNTEMLERİ: VERİ SETLERİ VE KARŞILAŞTIRMALI ANALİZ

Her geçen gün güncellenen saldırı yöntemlerine karşı STS sistemlerinin de güncellenmesi ve geliştirilmesi zorunlu hale gelmiştir. Bu kapsamda anomali tabanlı STS sistemleri, saldırı tespiti için sıklıkla kullanılan teknolojilerden biridir. Saldırı tespit sistemleri ile ilgili literatür çalışmaları incelendiğinde, çalışmaların genellikle az sayıda veri seti üzerinde yapıldığı görülmüştür. Bu durum, veri setlerinin performansı hakkında net bir fikir verememektedir. Bu çalışmada literatürdeki bu boşluğu gidermek için literatürde en sık kullanılan 5 veri seti ele alınmıştır. Ayrıca literatür çalışmaları sınırlı sayıda veri seti kullandığından sınırlı sayıda saldırı türünün performansı incelenebilmektedir. Çalışmada ele aldığımız 5 veri setinde saldırı türlerinin çoğu hakkında bilgi bulunmaktadır. Bu nedenle saldırı türlerinin performansı daha iyi ölçülebilir. Bu çalışmada, mevcut ağ saldırı tiplerini simüle edebilen CSE-CIC-IDS2018, UNSW-NB15, NSL-KDD, CIDDS-001 ve ISCX2012 veri setleri kullanılmıştır. Ağ tabanlı saldırı tespit sistemleri üzerinde geliştirilen veri setleri kullanılarak gerçekleştirilen bu çalışmanın katkıları aşağıda özetlenmiştir.

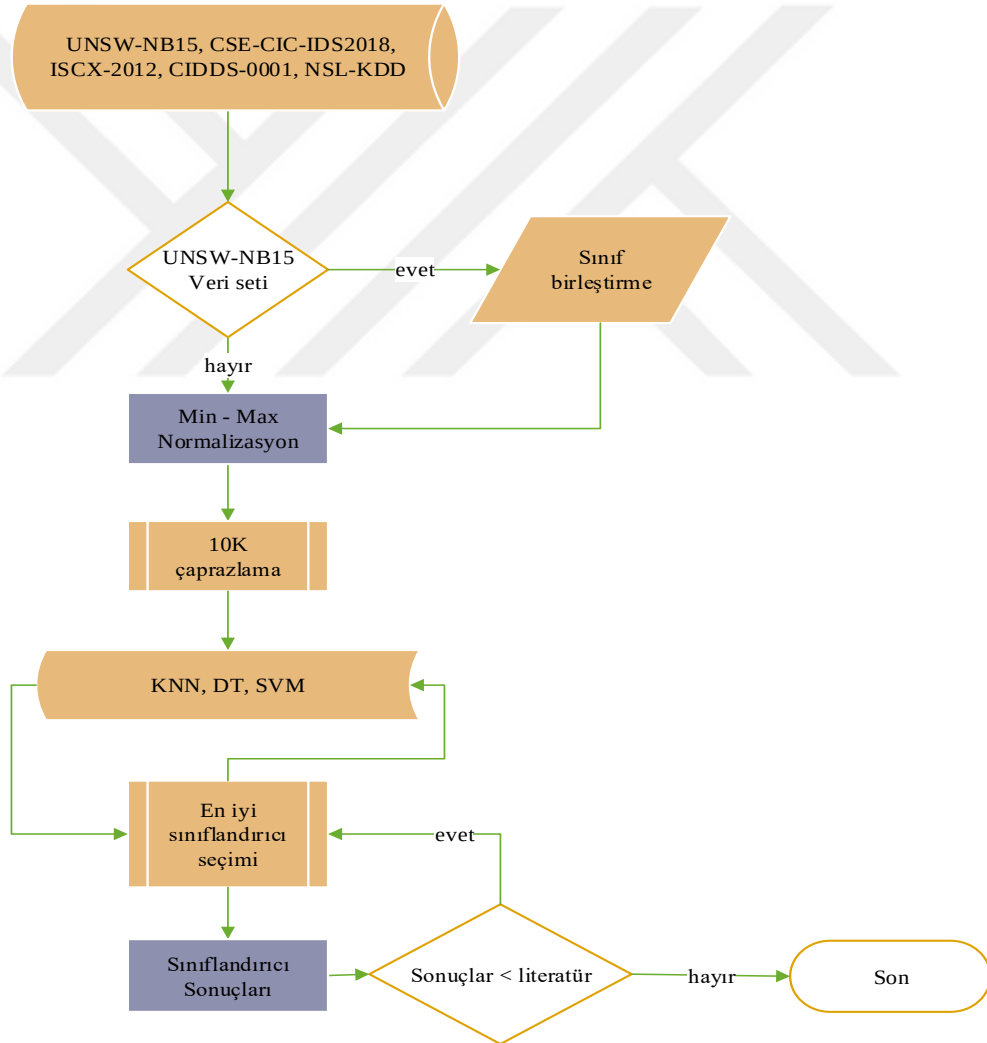
- Literatürde, makine öğrenmesi tabanlı STS yöntemleri genellikle sınırlı sayıda veri seti kullanarak sonuçlar elde etmiştir. Bu çalışmada, literatürdeki veri setlerinin saldırı algılama performanslarını daha iyi ölçmek için makine öğrenmesi tabanlı bir saldırı tespit sistemi sunulmuştur. Yöntemin genel başarısını ölçmek için, literatürde yaygın olarak tercih edilen CSE-CIC-IDS2018, UNSW-NB15, NSL-KDD, CIDDS-001 ve ISCX2012 STS veri setleri kullanılmış ve sonuçlar literatürdeki en son yöntemler ile karşılaştırılmıştır. Sonuçlar için SVM, KNN ve DT sınıflandırıcıları ve bu sınıflandırıcıların türevleri kullanılmıştır.
- UNSW-NB15 veri setindeki bazı sınıflar aralarındaki ilişkiden dolayı birleştirilmiştir. Bu veri seti için sınıf birleştirme işlemi önerilen yöntemde detaylı olarak açıklanmıştır. Ayrıca bu veri seti için literatürdeki sınıflandırıcı performans oranları incelenmiş ve bu veri seti için yüksek performans oranları elde edilmiştir.

2.1. Materyal ve Metod

Önerilen yöntemin akış şeması Şekil 2.1’de verilmiştir. Buna göre ilk aşamada, her bir veri setinden belirli bir miktarda örnek veri alınmış ve veri setlerindeki boş satırlar kaldırılmıştır. İkinci aşamada elde edilen veri setleri min-maks normalizasyonu ile normalize edilmiştir. Üçüncü aşamada normalize edilmiş veri setleri SVM, KNN ve DT makine öğrenmesi yöntemleri ile sınıflandırılmıştır. Dördüncü aşamada sınıflandırıcı sonuçları literatür ile karşılaştırılmıştır. Bu

aşamada CSE-CIC-IDS2018, NSL-KDD, CIDDs-001 ve ISCX2012 veri setlerinin sınıflandırıcı performanslarının yüksek ve literatüre benzer olduğu görülmüştür.

UNSW-NB15 veri setinin performans oranları hem literatürde hem de bu çalışmada istenilen düzeyde olmamıştır. UNSW-NB15 veri setinin başarımının artırılması için, son aşamada UNSW-NB15 veri setindeki birbirleri ile ilintili sınıflar birleştirilmiştir. Buna göre, Backdoors, Exploits ve Shellcode saldırıları, Exploits saldırısı olarak tek bir sınıfta birleştirilmiştir [70]. Bir sonraki adımda, tanımları gereği Fuzzers ve DoS saldırıları hedef ağa veya sistem çökmesine yönelik saldırılardır. Dolayısıyla bu iki saldırı tipi DoS saldırısı olarak sınıflandırılmıştır [18]. Sonuç olarak UNSW-NB15 veri seti Analysis, DoS, Exploits, Generic, Reconnaissance, Normal ve Worms adları altında yeniden etiketlenmiştir.



Şekil 2.1. Önerilen yöntemin akış diyagramı

Bu çalışmada kullanılan veri setlerinde çok miktarda veri bulunmaktadır. Bu sebeple, bütün veri setlerinden her sınıf için belirli miktarda veri alınmıştır. Veri setlerinden her bir sınıf için alınan veri miktarları Tablo 2.1’de verilmiştir.

Tablo 2.1. Sınıf bazlı veri miktarları

Veri seti	Sınıflar	Veri Miktarı
CSE-CIC-IDS 2018	Benign	1552
	Bot	1541
	Brute Force-WEB	363
	Brute Force-XSS	152
	DDOS attack-HOIC	1552
	DDOS attack-LOIC-UDP	1561
	DoS attacks-GoldenEye	1557
	DoS attacks-Hulk	1563
	DoS attacks-SlowHTTPTest	1523
	DoS attacks-Slowloris	1584
	FTP-BruteForce	1527
	infiltration	1546
	SQL Injection	54
	SSH-Bruteforce	1452
ISCX 2012	Normal	2517
	Attacker	2515
NSL-KDD	Normal	6817
	DoS	11617
	Probe	988
	R2L	53
	U2R	3086
CIDDs-001	Normal	633
	Attacker	985
	Suspicious	655
	Unknown	662
	Victim	1135
Kategorilendirilmiş UNSW-NB15 veri seti	Analysis	678
	DoS	2593
	Exploits	2431
	Generic	2089
	Normal	2587
	Reconnaissance	2253
	Worms	40

2.2. Bulgular ve Tartışma

Önerilen yöntem, 32 Gb RAM ve 4 CPU’lu bir Windows Server 2012 R2 sanal makine üzerine kurulu Matlab programı ile test edilmiştir. Her bir veri seti Matlab ortamında SVM, KNN ve DT sınıflandırıcıları ile sınıflandırılmıştır. Her sınıflandırıcı için 10 K çapraz doğrulama (cross-validation) kullanılmış ve her veri setinde 100 iterasyon yapılmıştır. Sınıflandırıcı çıktılarında Doğruluk (Accuracy), Hassasiyet (Precision), Duyarlılık (Recall), F-Değeri (F-measure) ve Geometrik Ortalama (Geometric Mean) değerleri alınmıştır. Bu parametreler sırasıyla Denklem 2.1, 2.2, 2.3, 2.4 ve 2.5’te verilmiştir. Denklemlerde verilen gerçek değerler ile tahmin edilen değerler arasındaki ilişki Tablo 2.2’de verilmiştir.

Tablo 2.2. Gerçek değerler ve tahmin değerleri arasındaki ilişki

Tahmin edilen sınıflar	Doğru Sınıflar	
	Doğru	Yanlış
	Doğru	Yanlış
Doğru	Doğru Pozitif (DP)	Yanlış Pozitif (YP)
Yanlış	Yanlış Negatif (YN)	Doğru Negatif (DN)

Tablo 2.2’de yer alan Doğru Pozitif (DP), karmaşıklık matrisinde doğru olarak sınıflandırılan ve gerçekten de doğru olan değerlerin sayısıdır. Yanlış Pozitif (YP), doğru olarak sınıflandırılan fakat gerçekte yanlış olan değerlerin sayısıdır. Yanlış Negatif (YN), yanlış olarak sınıflandırılan ve gerçekte de yanlış olan değerlerin sayısıdır. Doğru Negatif (DN), yanlış olarak sınıflandırılan fakat gerçekte doğru olan değerleri göstermektedir [166].

$$\text{Doğruluk}(DO) = \frac{DP+DN}{DP+DN+YP+YN} \quad (2.1)$$

$$\text{Hassasiyet}(H) = \frac{DP}{DP+YP} \quad (2.2)$$

$$\text{Duyarlılık}(D) = \frac{DP}{DP+YN} \quad (2.3)$$

$$\text{Geometrik Ortalama}(GO) = \sqrt{\frac{DP*DN}{(DP+YN)*(DN+YP)}} \quad (2.4)$$

$$F - \text{Ölçüsü}(FÖ) = 2 \times \frac{H \times D}{H+D} \quad (2.5)$$

Verilerin sınıflandırılmasında SVM, DT ve KNN sınıflandırıcılarının farklı çekirdek fonksiyonları kullanılmıştır. SVM sınıflandırıcısı için SVM Linear, SVM Quadratic, SVM Cubic algoritmaları, KNN sınıflandırıcısı için KNN Fine, KNN Medium ve KNN Cubic algoritmaları, DT sınıflandırıcısı için Tree Fine ve Tree Medium algoritmaları kullanılmıştır. Her veri seti için SVM, DT, KNN sınıflandırıcıları sonucunda elde edilen değerler Tablo 2.3'te verilmiştir.

Tablo 2.3. Sınıflandırıcı sonuçları

Veri setleri	Sınıflandırıcılar	Doğruluk	Hassasiyet	Duyarlılık	GO	F-Ölçüsü
CSE-CIC-IDS2018	SVM Linear	99,64	0.9864	0.9864	0.9713	0.9803
	SVM Quadratic	0.9981	0.9941	0.9941	0.9929	0.9935
	SVM Cubic	0.9913	0.9876	0.9876	0.9836	0.9858
	KNN Fine	0.9918	0.9577	0.9577	0.9463	0.9546
	KNN Medium	0.9865	0.9582	0.9582	0.9108	0.9414
	KNN Cubic	0.9865	0.9547	0.9547	0.9108	0.9414
	TREE Fine	0.9992	0.9994	0.9994	0.9994	0.9994
	TREE Medium	0.9992	0.9994	0.9994	0.9994	0.9994
NSL-KDD	SVM Linear	0.9847	0.9517	0.9517	0.8579	0.9156
	SVM Quadratic	0.9932	0.9635	0.9635	0.9181	0.9447
	SVM Cubic	0.9946	0.971	0.971	0.9146	0.944
	KNN Fine	0.9964	0.9808	0.9808	0.9476	0.9657
	KNN Medium	0.9915	0.9477	0.9477	0.8837	0.9227
	KNN Cubic	0.9909	0.9388	0.9388	0.8836	0.9199
	TREE Fine	0.9992	0.9994	0.9994	0.9994	0.9994
	TREE Medium	0.9992	0.9994	0.9994	0.9994	0.9994
ISCX 2012	SVM Linear	0.9849	0.9819	0.9819	0.9871	0.9845
	SVM Quadratic	0.998	0.9978	0.9978	0.998	0.9979
	SVM Cubic	0.9849	0.9819	0.9819	0.9871	0.9845
	KNN Fine	0.998	0.9978	0.9978	0.998	0.9979
	KNN Medium	0.9964	0.9959	0.9959	0.9967	0.9963
	KNN Cubic	0.996	0.9954	0.9954	0.9964	0.9959
	TREE Fine	1	0.9994	0.9994	1	0.9994
	TREE Medium	1	0.9994	0.9994	1	0.9994
CIDD5-001	SVM Linear	0.9284	0.9172	0.9172	0.9062	0.9147
	SVM Quadratic	0.9678	0.9648	0.9648	0.9593	0.9631
	SVM Cubic	0.9729	0.9682	0.9682	0.9668	0.9679
	KNN Fine	0.9781	0.9732	0.9732	0.9726	0.9731
	KNN Medium	0.9631	0.9553	0.9553	0.9541	0.9554
	KNN Cubic	0.9611	0.9534	0.9534	0.9522	0.9534
	TREE Fine	0.9966	0.9957	0.9957	0.9957	0.9957
	TREE Medium	0.9914	0.9894	0.9894	0.9891	0.9893
UNSW-NB15 kategorize edilmiş versiyon	SVM Linear	0.9937	0.8521	0.8521	0.8563	0.8534
	SVM Quadratic	0.9962	0.9668	0.9668	0.8563	0.9346
	SVM Cubic	0.9955	0.9382	0.9382	0.8631	0.922
	KNN Fine	0.9546	0.8239	0.8239	0.6858	0.8182
	KNN Medium	0.9479	0.8138	0.8138	0.6861	0.7942
	KNN Cubic	0.9448	0.8103	0.8103	0.6758	0.7905

Tablo 2.3. (Devamı)

	TREE Fine	0.9984	0.979	0.979	0.9507	0.968
	TREE Medium	0.9984	0.9942	0.9942	0.9508	0.9754
UNSW-NB15 orijinal veri seti	SVM Quadratic	0.7795	0.65	0.65	0.4302	0.619
	KNN Medium	0.7187	0.5606	0.5606	0.2798	0.5355
	TREE Fine	0.8057	0.7229	0.7229	0.4934	0.6687

Tablo 2.3 incelendiğinde CSE-CIC-IDS 2018 veri seti için en iyi SVM sonucu %99,81 ile SVM Quadratic algoritması ile elde edilirken, en iyi KNN sonucu %99,18 ile KNN Fine algoritması ile sağlanmıştır. Bu veri seti için en iyi sonuç %99,92 ile DT sınıflandırıcısı ile elde edilmiştir. NSL-KDD veri seti için en iyi SVM sonucu %99,46 ile SVM Cubic algoritmasına aitken, en iyi KNN sonucu %99,64 ile KNN Fine algoritması ve en iyi DT sonucu %99,92 ile Fine Tree algoritmasına aittir. NSL-KDD veri seti için en iyi sonuç %99,92 ile DT sınıflandırıcısı ile elde edilmiştir. ISCX 2012 veri seti için en iyi SVM sonucu %99,8 doğruluk oranı ile SVM Quadratic algoritması ile elde edilirken, en iyi KNN sonucu yine %99,8 ile KNN Fine algoritması ve en iyi DT sonucu %100 ile Fine Tree algoritması ile elde edilmiştir. CIDDS-001 veri seti için en iyi SVM sonucu %97,29 ile SVM Cubic algoritması ile, en iyi KNN sonucu %97,81 ile KNN Fine algoritması ile ve en iyi DT sonucu %99,66 ile Fine Tree algoritması ile elde edilmiştir. CIDDS-001 veri seti için en iyi sonuç %99,66 doğruluk oranı sağlayan DT sınıflandırıcısı ile elde edilmiştir. UNSW-NB15 veri setinin kategorize edilmiş versiyonu için en iyi SVM sonucu %99,62 ile SVM Quadratic algoritması, en iyi KNN sonucu %95,46 KNN Fine algoritması ve en iyi DT sonucu %99,84 Fine Tree algoritması ile elde edilmiştir. UNSW-NB15 veri setinin kategorize edilmemiş versiyonu için en iyi sonuç %80,57 ile DT yöntemi kullanılarak elde edilmiştir.

Çalışmada uygulanan makine öğrenmesi yöntemlerinin performansı literatürdeki çalışmalarla karşılaştırılmıştır. Tablo 2.4'te CSE-CIC-IDS2018 veri seti ile ilgili karşılaştırmalı tablo verilmiştir.

Tablo 2.4. CSE-CIC-IDS2018 veri seti karşılaştırmalı analiz

Referans	Sınıflandırıcı	Özellik Seçimi	Doğruluk	Hassasiyet	Duyarlılık	F-Ölçüsü
[40]	DNN	-	0.9728	-	-	-
	RNN		0.9731	-	-	-
	CNN		0.9376	-	-	-
	RBM		0.9728	-	-	-
	DBN		0.9730	-	-	-
	DBM		0.97371	-	-	-
	DA		0.97372	-	-	-
[67]	CNN	-	SD-1 (0.9998)	-	-	-
			SD-2 (0.9005)	-	-	-
			SD-3 (0.9677)	-	-	-
			SD-4 (0.9998)	-	-	-

Tablo 2.4. (Devamı)

			SD-5 (0.9997)	-	-	-
			SD-6 (0.9995)	-	-	-
			SD-7 (0.9994)	-	-	-
			SD-8 (0.8879)	-	-	-
			SD-9 (0.7601)	-	-	-
			SD-10 (0.9997)	-	-	-
[65]	ANN	-	0.9997	0.9996	1.0	0.9998
	RF		0.9983	0.9992	0.9988	0.9992
	KNN		0.9973	0.998	0.9988	0.9984
	SVM		0.998	0.9	0.9988	0.9994
	Adaboost		0.9996	0.9996	0.9988	0.9992
	NB		0.992	0.9929	0.9976	0.9953
[68]	ANN	MLP	0.9998	0.9995	1.0	-
Önerilen Yöntem	Fine Tree	-	0.9992	0.9994	0.9994	0.9992
	Medium Tree		0.9992	0.9994	0.9994	0.9994
	Cubic Tree		0.9865	0.9547	0.9547	0.9414
	Fine KNN		0.9918	0.9577	0.9577	0.9546
	Medium KNN		0.9865	0.9582	0.9582	0.9414
	Cubic KNN		0.9865	0.9547	0.9547	0.9414
	Linear SVM		0.9964	0.9864	0.9864	0.9803
	Quadratic SVM		0.9981	0.9941	0.9941	0.9935
	Cubic SVM		0.9913	0.9876	0.9876	0.9858

Tablo 2.4'te CSE-CIC-IDS2018 veri seti üzerinde yapılan çalışmalar ve önerilen yöntemin performansı karşılaştırılmıştır. Tabloya göre CSE-CIC-IDS 2018 veri seti için %99,81, %99,18 ve %99,92 başarı oranları elde edilmiş olup, elde edilen doğruluk oranları literatür çalışmaları ile benzerdir.

UNSW-NB15 veri seti ile ilgili karşılaştırmalı analiz Tablo 2.5'te verilmiştir.

Tablo 2.5. UNSW-NB15 veri seti karşılaştırmalı analiz

Referans	Sınıflandırma Yöntemi	Özellik Seçimi	Doğruluk	TO	YAO	YP	YN
[48]	DT	-	0.8556	-	0.1578	-	-
	LR		0.8315	-	0.1848	-	-
	NB		0.8207	-	0.1856	-	-
	ANN		0.8137	-	0.2113	-	-
	EM Clustering		0.7847	-	0.2379	-	-
[64]	RF	BBA+FSFF	0.9909	-	-	0.63	-
		BBA+CAFF	0.9927	-	-	0.51	-
		BBA+FU	0.9943	-	-	0.39	-
[66]	C4.5, NB, Tree RF	NSGA2-BLR	0.9490	-	-	-	-
		NSGA2- MLR	0.6600	-	-	-	-
[69]	k-Means	GAA-ADS	0.776	0.754	-	0.082	-
			0.86	0.852	-	0.063	-
			0.882	0.871	-	0.061	-
			0.927	0.912	-	0.059	-
			0.928	0.913	-	0.051	-

Tablo 2.5. (Devamı)

[70]	MSCNN-LSTM	-	0.898	-	0.474	-	0.086
[71]	-	CorrCorr	0.9865	0.9974	-	0.012	-
[72]	C4.5	GA-LR	0.8149	-	-	-	-
	RF		0.1408	-	-	-	-
	NBTree		0.821252	-	-	-	-
[73]	WFEU	Binary Sınıflandırma FFDNN	0.8710	-	-	-	-
		Çok Sınıflı FFDNN	0.7716	-	-	-	-
[74]	FCM, CFS	ABC, AFS	0.95	0.88	-	0.021	-
			0.968	0.891	-	0.0089	-
			0.974	0.928	-	0.0063	-
			0.983	0.973	-	0.0032	-
			0.989	0.980	-	0.0013	-
Önerilen Yöntem	Fine Tree	-	0.9984	-	-	-	-
	Medium Tree		0.9984	-	-	-	-
	Cubic Tree		0.9448	-	-	-	-
	Fine KNN		0.9546	-	-	-	-
	Medium KNN		0.9479	-	-	-	-
	Cubic KNN		0.9448	-	-	-	-
	Linear SVM		0.9937	-	-	-	-
	Quadratic SVM		0.9962	-	-	-	-
	Cubic SVM		0.9955	-	-	-	-

Tablo 2.5'te UNSW-NB15 veri seti üzerinde yapılan literatür çalışmaları ve önerilen yöntem karşılaştırılmıştır. Bu çalışmada UNSW-NB15 veri setinde birbiriyle ilişkili olan sınıflar aynı kategoride toplanmıştır. Tablo 2.3'te bu veri setinin kategorize edilmiş versiyonu ve orijinal versiyonu aynı sınıflandırıcılarda sınıflandırılmış ve sonuçlar verilmiştir. Tablo 2.3'te görüldüğü gibi sınıfların kategorilere ayrılması UNSW-NB15 veri setinin başarımını artırmıştır. Ayrıca Tablo 2.5'te görüldüğü gibi, sınıfların kategorilendirilmesi işlemi, bu veri seti için literatürdeki en yüksek başarımı sağlamıştır. Bu da, UNSW-NB15 veri seti için sınıflandırma işleminin ne kadar doğru yapıldığını göstermektedir.

NSL-KDD veri seti ile ilgili karşılaştırmalı analiz Tablo 2.6'da verilmiştir.

Tablo 2.6. NSL-KDD veri seti karşılaştırmalı analiz

Referans	Sınıflandırma Yöntemi	Özellik Seçimi	Doğruluk	TO	YAO	YP
[66]	C4.5, NB, Tree RF	NSGA2-BLR	0.9965	-	-	-
		NSGA2-MLR	0.9899	-	-	-
[69]	k-Means	GAA-ADS	0.95	0.942	-	0.011
			0.953	0.951	-	0.0007
			0.977	0.964	-	0.0002

Tablo 2.6. (Devamı)

			0.988	0.987	-	0.0002
			0.997	0.996	-	0.0002
[74]	FCM, CFS	ABC, AFS	0.967	0.902	-	0.0082
			0.97	0.923	-	0.006
			0.973	0.951	-	0.0034
			0.976	0.986	-	0.0015
			0.99	0.99		0.0011
[81]	SVM	Hybrid (CFS, IGR, Gini Index)	1.00	-	-	-
	KNN		0.9980	-	-	-
	NB		0.9860	-	-	-
	NN		0.9990	-	-	-
	DNN		0.9990	-	-	-
	Auto Encoder		0.9860	-	-	-
Önerilen Yöntem	Fine Tree	-	0.9992	-	-	-
	Medium Tree		0.9991	-	-	-
	Cubic Tree		0.9909	-	-	-
	Fine KNN		0.9964	-	-	-
	Medium KNN		0.9915	-	-	-
	Cubic KNN		0.9909	-	-	-
	Linear SVM		0.9847	-	-	-
	Quadratic SVM		0.9932	-	-	-
	Cubic SVM		0.9946	-	-	-

Tablo 2.6’da NSL-KDD veri seti ile yapılan literatür çalışmaları ve önerilen yöntem karşılaştırılmıştır. NSL-KDD veri seti için %99.46, %99.64 ve %99.92 doğruluk oranları elde edilmiş olup literatür ile benzer oranlarda başarı elde edilmiştir.

CIDDS-001 veri seti ile ilgili karşılaştırmalı analiz Tablo 2.7’de verilmiştir.

Tablo 2.7. CIDDS-001 veri seti karşılaştırmalı analiz

Referans	Sınıflandırma Yöntemi	Özellik Seçimi	Doğruluk	Hassasiyet	Duyarlılık	F-Ölçüsü
[49]	KNN	-	0.995	-	-	-
			0.996	-	-	-
			0.994	-	-	-
			0.995	-	-	-
			0.993	-	-	-
			0.995	-	-	-
[50]	kNN	-	0.995	-	-	-
	SVM		0.953	-	-	-
	DT		0.999	-	-	-
	RF		0.999	-	-	-
	NB		0.871	-	-	-
	DL		0.941	-	-	-
	ANN		0.638	-	-	-
	EM		0.459	-	-	-
	SOMs		0.384	-	-	-
	k-means		0.381	-	-	-
Önerilen Yöntem	Fine Tree	-	0.9966	0.9957	0.9957	0.9957
	Medium Tree		0.9914	0.9894	0.9894	0.9893

Tablo 2.7. (Devamı)

	Cubic Tree		0.9611	0.9534	0.9534	0.9534
	Fine KNN		0.9781	0.9732	0.9732	0.9731
	Medium KNN		0.9631	0.9553	0.9553	0.9554
	Cubic KNN		0.9611	0.9534	0.9534	0.9534
	Linear SVM		0.9284	0.9172	0.9172	0.9147
	Quadratic SVM		0.9678	0.9648	0.9648	0.9631
	Cubic SVM		0.9729	0.9682	0.9682	0.9679

CIDDS-001 veri seti ile ilgili literatür çalışmaları ve önerilen yöntem Tablo 2.7’de karşılaştırılmıştır. Bu veri seti için yüksek performans oranları elde edilmiştir ve elde edilen %97,29, %97,81 ve %99,66 doğruluk değerleri literatür ile benzerlik göstermektedir.

ISCX2012 veri seti ile ilgili karşılaştırmalı analiz Tablo 2.8’de verilmiştir.

Tablo 2.8. ISCX2012veri seti karşılaştırmalı analiz

Referans	Sınıflandırma Yöntemi	Özellik Seçimi	Doğruluk	Hassasiyet	Recall	F-Ölçüsü	TO	YP
[76]	WELM	-	0.9910	-	-	-	-	-
[77]	EMD	-	0.9012	-	-	-	0.9004	0.0792
[78]	OCSVM	-	0.7630	-	-	-	0.106	0.0991
	ELM		0.9287	-	-	-	0.8284	0.0504
	RPART		0.9807	-	-	-	0.9582	0.0146
	SVM		0.9953	-	-	-	0.9936	0.0043
	RF		0.9985	-	-	-	0.9967	0.0012
[85]	SVM	BO optimizasyonu	0.9984	0.9998	1.00	-	-	0.003
	KNN		0.9993	0.9999	1.00	-	-	0.001
	RF		0.9992	0.9999	1.00	-	-	0.001
[79]	KMC + NBC	-	0.99	-	-	-	0.988	0.022
Önerilen Yöntem	Fine Tree	-	1.00	0.9994	0.9994	0.9994	-	-
	Fine KNN		0.998	0.9978	0.9978	0.9979	-	-
	Medium KNN		0.9964	0.9959	0.9959	0.9963	-	-
	Cubic KNN		0.996	0.9954	0.9954	0.9959	-	-
	Linear SVM		0.9849	0.9819	0.9819	0.9845	-	-
	Quadratic SVM		0.998	0.9978	0.9978	0.9979	-	-
	Cubic SVM		0.9849	0.9819	0.9819	0.9845	-	-

Tablo 2.8’de ISCX2012 veri seti için literatür çalışmalarının başarı oranları önerilen yöntemle karşılaştırılmıştır. Tabloya göre önerilen yöntemde en iyi doğruluk değeri %100 ile DT algoritmasına aittir. Ayrıca bu veri seti üzerinden elde edilen sonuçlar literatürle benzerlik göstermektedir.

2.3. Sonular

Bu alıřmada saldırı tespiti iin kullanılan CSE-CIC-IDS2018, UNSW-NB15, ISCX-2012, NSL-KDD ve CIDDS-001 veri setleri kullanılmıřtır. Veri setlerini sınıflandırmak iin SVM, KNN ve DT algoritmaları kullanılmıřtır. alıřma sonucunda tm veri setleri iin kullanılan sınıflandırıcıların literatr ile benzer veya daha yksek performans elde ettiėi grlmřtr. Ayrıca DT sınıflandırıcının sınıflandırıcı performansı aısından diėer sınıflandırıcılara gre daha bařarılı olduėu tespit edilmiřtir. DT'nin CSE-CIC-IDS2018, ISCX-2012, NSL-KDD ve CIDDS-001 veri setleri iin %99 - %100 aralıėındaki bařarı oranları literatr ile benzerlik gstermektedir. alıřmada UNSW-NB15 veri seti iin bir kategorizasyon iřlemi yapılmıřtır. Sonu olarak UNSW-NB15 veri seti iin tm sınıflandırıcılarda elde edilen performans oranları literatrdeki alıřmaların ilerisindedir. Bu anlamda UNSW-NB15 veri seti iin yapılan kategorizasyon iřleminin hedefine ulařtıėı grlmřtr.

3. BOOSTING METODLARINI KULLANAN YENİ BİR SALDIRI TESPİT ÇERÇEVESİ

STS sistemlerinin performansını artırmak için makine öğrenmesi ve derin öğrenme gibi birçok yöntem kullanılmasına rağmen istenilen performans henüz yakalanamamıştır. Literatürde, STS sistemlerinin performansını artırmak için çeşitli veri setleri oluşturulmuş ve bu veri setleri üzerinde testler yapılmıştır. Literatür çalışmalarının çoğu hazır ve eski STS veri setleri kullanılarak yapılmaktadır. Tez çalışmasının bu bölümünde, günümüzde yaygın olarak kullanılan saldırı yöntemleri ve sık kullanılan internet uygulamaları ile yeni bir veri seti oluşturulması gerekli görülmüştür. Bunun sonucunda, yeni ve kapsamlı bir Comprehensive Cyber Security intrusion Detection Dataset (CCiDD) oluşturulmuştur. Güncel bir STS veri setinin yanı sıra, ağ üzerinden alınan verilerin hızlı ve doğru bir şekilde sınıflandırılması da önem arz eden bir konudur. Verilerden en iyi özelliklerin seçimi ve verilerin sınıflandırılması için çeşitli algoritmalar test edilerek hızlı ve başarılı bir yöntem önerilmiştir. Çalışmanın katkıları aşağıda özetlenmiştir.

- Bu çalışmada, yaygın olarak kullanılan saldırı yöntemleri ve internet uygulamaları ile yeni bir CCiDD veri seti oluşturulmuştur. Oluşturulan veri setinin performansını arttırmak için normal kullanıcı trafiği kendi içinde değerlendirilerek CCiDD_A ve CCiDD_B olarak adlandırılan iki veri seti türetilmiştir. Literatürdeki birçok çalışmada “Normal” trafik sınıfına ait veriler haber, facebook, instagram gibi çeşitli uygulamalardan oluşmaktadır. Bu çalışmada haber, Youtube, Facebook, Twitter, Instagram, LinkedIn, Science Direct, Elsevier, IeeeExplore, Alibaba, Amazon, Ebay, uTorrent, Anydesk, Teamviewer, Google Meet, Zoom gibi yaygın olarak kullanılan uygulamalar ile “Normal” trafik sınıfı oluşturulmuştur. Oluşturulan veri setleri, yapay zeka çalışmalarına katkı sağlamak amacıyla paylaşılmıştır [167].
- Ağ üzerinden alınan verilerin hızlı ve başarılı bir şekilde işlenmesi için Extra Tree algoritması ile veri setlerinin en uygun özellikleri seçilmiştir. Daha sonra veri setleri yüksek performanslı GBM, LGBM, XGBoost, Catboost algoritmaları kullanılarak sınıflandırılmıştır.

3.1. Materyal ve Metod

3.1.1. Genel Veri Setleri

Çalışmada oluşturulan veri setlerinin performansını literatürde sıklıkla kullanılan CSE-CIC-IDS2018, UNSW-NB15 ve NSL-KDD veri setleri ile karşılaştırılmıştır. Bu veri setlerinin her bir sınıfından alınan veri miktarları Tablo 3.1’de verilmiştir.

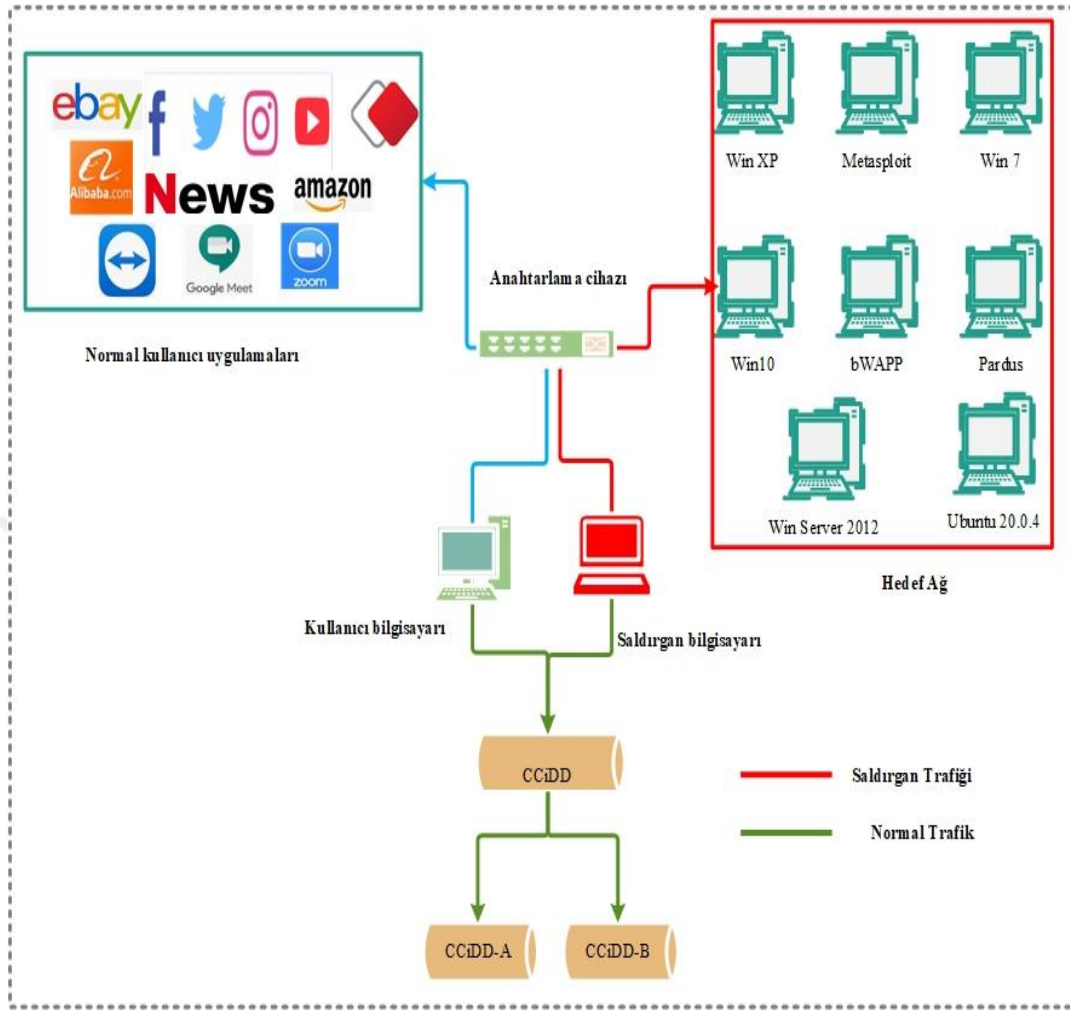
Tablo 3.1. Kullanılan genel veri setlerinin özellikleri

Veri Setleri	Karmaşıklık Matrisi Numarası (Şekil 3.8 bkz.)	Ana Trafik Sınıfları	Saldırı Senaryoları	Veri Miktarı
CSE-CIC-IDS2018 [69,93] [47]	1	Benign	Benign	1552
	2	Bot	Bot	1541
	3	WEB saldırıları	Brute Force-WEB	363
			Brute Force-XSS	152
	4	Hizmet Reddi Saldırıları	DDOS attack-HOIC	1552
			DDOS attack-LOIC-UDP	1561
			DoS attacks-GoldenEye	1557
			DoS attacks-Hulk	1563
			DoS attacks-SlowHTTPTest	1523
			DoS attacks-Slowloris	1584
	5	Kaba Kuvvet saldırıları	FTP-BruteForce	1527
			SSH-Bruteforce	1546
	6	SQL Enjeksiyonu	SQL Injection	54
NSL-KDD [44]	7	Sızma	Infiltration	1451
	1	Normal	Normal	6870
	2	DoS	DoS	11657
	3	Probe	Probe	976
	4	R2L	R2L	53
UNSW-NB15 [56]	5	U2R	U2R	3067
	0	Normal	Normal	37000
	1	Attacker	Fuzzers	45332
			Analysis	
			DoS	
			Backdoors	
			Exploits	
			Generic	
			Shellcode	
			Reconnaissance	
			Worms	

Her bir veri setinden alınan veri miktarı, veri setlerindeki trafik sınıfları ve bu trafik sınıflarının ana kategorileri Tablo 3.1’de verilmiştir. Önerilen yöntem veri setlerinin ana kategorilerine uygulanmıştır. Tablo 3.1’de verilen ve literatürde sıklıkla kullanılan bu veri setlerindeki eksikliklerin giderilmesi amacıyla bu çalışmada CCiDD veri seti oluşturulmuştur.

3.1.2. CCiDD Veri Seti

Bu çalışmada, saldırı tespit sistemlerinin performansını artırmak için kapsamlı bir veri seti oluşturulmuştur. Veri setini oluşturmak için Şekil 3.1’de verilen topoloji oluşturulmuştur. Veri seti, normal ağ trafiği ve saldırı trafiğinden oluşmaktadır. Normal trafik oluştururken, yalnızca web tarama trafiği değil, aynı zamanda normal bir kullanıcının kullanabileceği uygulamalardan gelen ağ trafikleri de alınmıştır. Böylece gerçek dünyaya benzer bir ağ trafiği veri seti oluşturulmuştur.



Şekil 3.1. CCiDD veri seti için oluşturulan ağ topolojisi

Çalışmada günlük hayatta en sık görülen saldırı senaryoları ve kullanıcıların sıklıkla kullandıkları uygulamaların trafikleri alınmıştır. Hedef sistemin gerçekçi bir sistemi yansıtması istenmiştir. Normal kullanıcı trafiği için Tablo 3.2’de verilen e-ticaret, sosyal medya, web, uzak masaüstü, toplantı uygulamaları, peer to peer (P2P) trafikleri kullanılmıştır. Saldırı trafiği için hedef ağa i7 işlemcili ve 16 GB RAM’li üç ayrı fiziksel bilgisayar kurulmuştur. Daha sonra Windows XP, Windows 7, Windows 10, Windows Server 2012, Ubuntu 20.04, Pardus, bWAPP ve Metasploit makineleri her bilgisayara sanal olarak kurulmuştur. Toplamda 3 fiziksel makineye, 24 sanal bilgisayar kurulmuştur. Yönlendirme yeteneğine sahip 3. katman kenar anahtarı, ağların birbirleriyle iletişim kurması için kullanılmıştır. Tüm bilgisayarların DHCP sunucusundan IP alması sağlanmıştır. Şekil 3.1’de saldırgan makineden hedef ağa giden kırmızı çizgi saldırı trafiğini, kullanıcı bilgisayarından gelen mavi çizgi ise normal kullanıcı trafiğini göstermektedir.

Tablo 3.2. CCiDD_A ve CCiDD_B veri seti özellikleri

Numara	Ana Trafik Sınıfları	Trafik	Kullanılan Araçlar	Hedef	Veri sayısı	CCiDD_A Veri Miktarı	CCiDD_B Veri Miktarı
1	Zafiyet Taraması	TCP_scan	Nmap	Tüm hedef ağ bilgisayarları	1674	4750	4750
		UDP_scan	Nmap	Tüm hedef ağ bilgisayarları	1533		
		SYN_scan	Nmap	Tüm hedef ağ bilgisayarları	1543		
2	Kaba Kuvvet Saldırıları	FTP	Hydra	Windows 10	1520	7576	7576
		RDP	Hydra	Ubuntu 20.04	1508		
		SSH	Hydra	Ubuntu 20.04	1514		
		TELNET	Hydra	Windows 7	1511		
		XSS	Xsstrike.py	bWAPP	1523		
3	Hizmet Reddi Saldırıları	HTTP Flood	Perl	hackthebox.eu	1422	4621	4621
		ICMP Flood	Hping3	Tüm hedef ağ bilgisayarları	1597		
		SYN Flood	Hping3	Tüm hedef ağ bilgisayarları	1602		
4	İstismar Saldırıları	Exploit	Beef, Bettercap, Msfconsole, Veil	Windows XP, Windows 7	1014	1014	1014
5	Fuzzing	Fuzzing	Dotddotpwn	Metasploit	1518	1518	1518
6	Heartbleed	Heartbleed	Xsstrike.py	bWAPP	1518	1518	1518
7	SQL Enjeksiyonu	SQL Injection	SQLmap	bWAPP	1555	1555	1555
8	WEB Uygulama	Haberler	Google Chrome	CNN-International	611	2778	8893
		Akademik araştırma	Science Direct	Science Direct	470		
		Akademik araştırma	IEEE Explore	IEEE Explore	584		
		Akademik araştırma	Elseiver	Elseiver	504		
		Ekonomi haberleri	Investing	tr.investing.com	609		
9	e-ticaret	Alibaba	Google Chrome	Alibaba	540	1697	
		Amazon	Google Chrome	Amazon	621		
		Ebay	Google Chrome	Ebay	536		
10	Uzak masaüstü bağlantıları	Anydesk	Anydesk	Anydesk	327	696	
		Teamviewer	Teamviewer	Teamviewer	369		
11	Sosyal Medya	Facebook	Facebook	Facebook	412	2526	
		Instagram	Instagram	Instagram	348		
		Linkedin	Linkedin	Linkedin	390		
		Twitter	Twitter	Twitter	454		
		Youtube	Youtube	Youtube	569		
12	Toplantılar	Zoom	Zoom	Zoom	353	805	
		Google Meet	Google Meet	Google Meet	452		

Tablo 3.2. (Devamı)

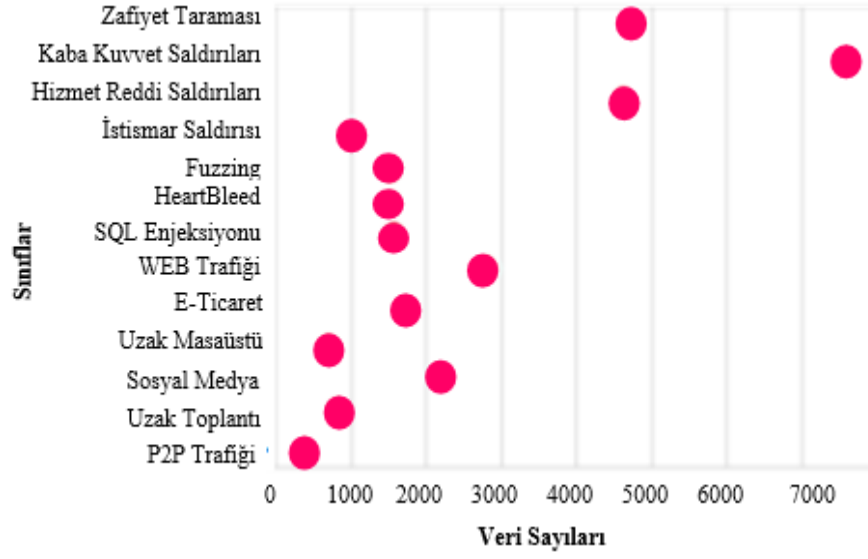
13	P2P Trafik	Torrent	uTorrent	uTorrent	391	391	
	Toplam				31092	31092	31092

Önerilen CCiDD veri seti, CCiDD_A ve CCiDD_B olarak ayrıştırılmıştır. CCiDD_B veri setinde e-ticaret, sosyal medya, web, uzak masaüstü, toplantı uygulamaları, P2P uygulamaları “normal” trafik olarak kategorize edilmektedir. CCiDD_A ve CCiDD_B veri setlerinin sınıfları Tablo 3.3’te özetlenmiştir.

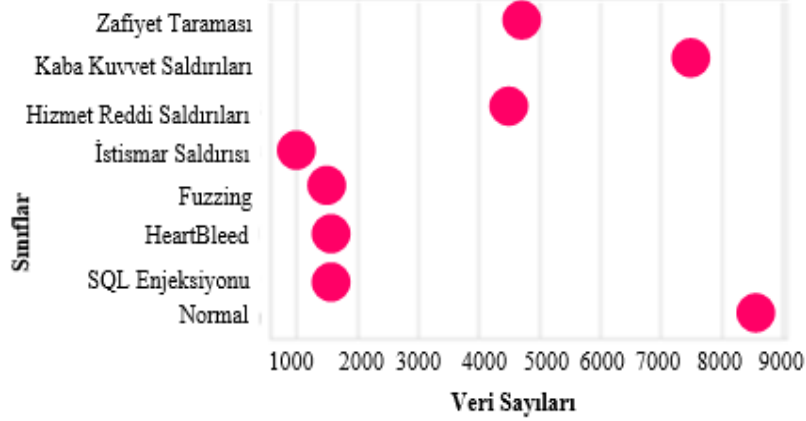
Tablo 3.3. Önerilen CCiDD_A ve CCiDD_B veri setleri

Veri seti	Yıl	Özellik Sayısı	Veri seti saldırı ve uygulama sınıfları
CCiDD_A	2021	80	TCP, UDP ve SYN Zafiyet taramaları, FTP, SSH, Telnet, RDP ve XSS kaba kuvvet saldırıları, HTTP Flood, ICMP Flood ve SYN Flood hizmet reddi saldırıları, istismar saldırıları, Fuzzing, Heartbleed, SQL enjeksiyonu, web uygulamaları, e-ticaret, sosyal medya, uzak masaüstü bağlantıları, toplantı uygulamaları, P2P trafiği
CCiDD_B	2021	80	TCP Taraması, UDP Taraması, SYN Taraması, FTP, SSH, Telnet, RDP, XSS Kaba Kuvvet saldırıları, HTTP Flood, ICMP Flood, SYN Flood Hizmet Reddi saldırıları, istismar saldırıları, Fuzzing, Heartbleed, SQL Enjeksiyonu, Normal

Çalışmada elde edilen verilerden CCiDD_A ve CCiDD_B veri setleri üretilmiştir. Bu veri setlerindeki sınıfların dağılımı Şekil 3.2 ve Şekil 3.3’te görselleştirilmiştir. Şekil 3.2 ve Şekil 3.3’te görüldüğü gibi bazı sınıflar diğer sınıflara göre daha fazla veriye sahiptir. Bunun nedeni, her sınıf için farklı saldırı türlerinin veya normal ağ trafiğinin üretilmesidir. Örneğin kaba kuvvet saldırısı için FTP, RDP, SSH, Telnet olmak üzere dört farklı saldırı tipi düzenlenmiş ve toplam 7576 veri elde edilmiştir. Öte yandan İstismar, Fuzzing, HeartBleed, SQL Enjeksiyonu gibi saldırı türleri tek saldırı senaryosu olarak düşünülmüş ve daha az veri toplanmıştır. Normal trafik sınıflarında her trafik sınıfı için farklı sayıda uygulama denenmiştir. Örneğin sosyal medya sınıfı için Facebook, Twitter, Instagram, Youtube, LinkedIn gibi sıklıkla kullanılan sosyal medya araçları kullanılmış ve toplam 2173 veri elde edilmiştir. Öte yandan P2P sınıfı için sadece Torrent, toplantı uygulamaları sınıfı için Zoom ve Google Meets uygulamaları kullanılmıştır. Her sınıf için alınan toplam veri miktarı Tablo 3.2’de paylaşılmıştır. CCiDD_B veri seti için e-ticaret, sosyal medya, web uygulamaları, uzak masaüstü bağlantıları, toplantı uygulamaları, P2P trafikleri birleştirilmiş ve 8540 adet “normal” trafik verisi elde edilmiştir.

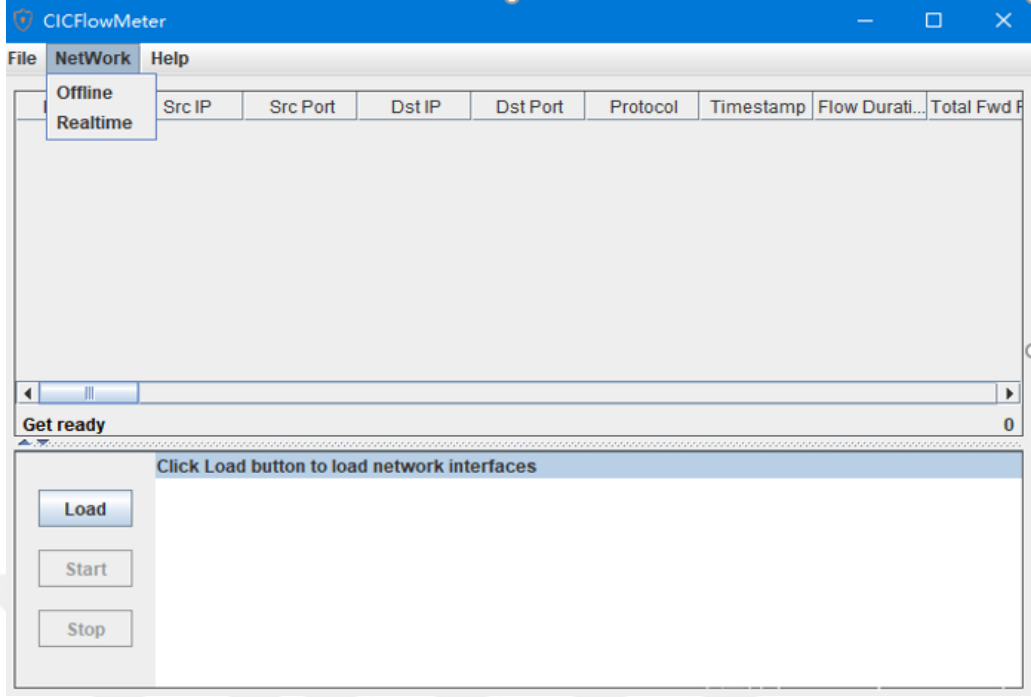


Şekil 3.2. CCiDD_A veri seti veri dağılımı



Şekil 3.3. CCiDD_B veri seti veri dağılımı

Çalışmada veri setini oluşturmak için CICFlowMeter yazılımı kullanılmıştır. Saldırgan makineden düzenlenen her saldırı senaryosu ve kullanıcı bilgisayarından ziyaret edilen her uygulama için wireshark ile veri paketleri toplanmıştır. Elde edilen wireshark dosyaları, CICFlowMeter yazılımı ile özelliklerine ayrıştırılmıştır. CICFlowMeter, ağ trafiğini 80 özelliğe ayıran açık kaynaklı bir yazılımdır. Çevrimiçi ağ trafiğinden veya hazır “pcap” uzantılı dosyalardan Biflow’lar oluşturur ve özellik çıkarır [56,168]. CICFlowMeter yazılımının arayüzü Şekil 3.4’te verilmiştir.



Şekil 3.4. CICFlowMeter arayüzü

CICFlowMeter çıkışında alınan özellikler ve bu özelliklerin açıklamaları Tablo 3.4'te verilmiştir [58].

Tablo 3.4. CICFlowMeter çıktısının özellikleri

Özellik	Açıklama	Özellik	Açıklama
src.port	Kaynak bağlantı noktası	pkt.len.min	Bir paketin minimum uzunluğu
dst.port	hedef bağlantı noktası	pkt.len.max	Bir paketin minimum uzunluğu
Protocol	Protokol	pkt.len.mean	Bir paketin ortalama uzunluğu
fl.dur	Akış (flow) süresi	pkt.len.std	Bir paketin standart sapma uzunluğu
tot.fwd.pk	İleri yön (FWD) paketleri toplamı	pkt.len.va	Bir paketin varyans uzunluğu
tot.bwd.pk	Geri yön (BWD) paketleri toplamı	fin.cnt	FIN paket sayısı
tot.len.fwd.pkt	İleri yön paketlerinin boyutu	syn.cnt	SYN paket sayısı
tot.len.BWD.pkt	Geri yön paketlerinin boyutu	rst.cnt	RST paket sayısı
fwd.pkt.l.max	Maksimum FWD paket boyutu	psh.cnt	PUSH ile paket sayısı
fwd.pkt.l.min	Minimum FWD paket boyutu	ack.cnt	ACK içeren paket sayısı

Tablo 3.4. (Devamı)

fwd.pkt.l.avg	FWD paketlerinin ortalama boyutu	urg.cnt	URG içeren paket sayısı
fwd.pkt.l.std	FWD paketinin standart sapma boyutu	cwr.cnt	CWR içeren paket sayısı
bwd.pkt.l.max	Maksimum BWD paket boyutu	ece.cnt	ECE içeren paket sayısı
bwd.pkt.l.min	Minimum BWD paket boyutu	down.up.ratio	İndirme ve yükleme oranı
bwd.pkt.l.avg	BWD paketlerinin ortalama boyutu	pkt.size.avg	Ortalama paket boyutu
bwd.pkt.l.std	BWD paketinin standart sapma boyutu	fwd.seg.avg	İleri yönde gözlemlenen ortalama boyut
fl.byt.s	bayt akış hızı	bwd.seg.avg	Geri yönde gözlemlenen ortalama boyut
fl.pkt.s	Paket akış hızı	fwd.byt.blk.avg	İleri yöndeki ortalama bayt sayısı toplu oranı
fl.iat.mean	Akışlar arasındaki ortalama süre	fwd.pkt.blk.avg	İleri yöndeki ortalama paket sayısı toplu oranı
fl.iat.std	İki akış arasındaki standart sapma süresi	fwd.blk.rate.avg	İleri yöndeki toplu oranın ortalama sayısı
fl.iat.max	Akışta gönderilen iki paket arasındaki maksimum süre	bwd.byt.blk.avg	Geri yöndeki ortalama bayt sayısı toplu oranı
fl.iat.min	Akışta gönderilen iki paket arasındaki minimum süre	bwd.pkt.blk.avg	Geri yöndeki ortalama paket sayısı toplu oranı
fwd.iat.tot	İki FWD paketi arasındaki süre	bwd.blk.rate.avg	Geri yöndeki toplu oranın ortalama sayısı
fwd.iat.mean	İleri yönde gönderilen iki paket arasındaki ortalama süre	subfl.fwd.pk	İleri yönde bir alt akıştaki ortalama paket sayısı
fwd.iat.std	İleri yönde gönderilen iki paket arasındaki standart sapma süresi	subfl.fwd.byt	İleri yönde bir alt akıştaki ortalama bayt sayısı
fwd.iat.max	İleri yönde gönderilen iki paket arasındaki maksimum süre	subfl.bwd.pk	Geri yönde bir alt akıştaki ortalama paket sayısı
fwd.iat.min	İleri yönde gönderilen iki paket arasındaki minimum süre	subfl.bwd.byt	Geri yönde bir alt akıştaki ortalama bayt sayısı
bwd.iat.tot	Geri yönde gönderilen iki paket arasındaki toplam süre	fwd.win.byt	İlk pencerede ileri yönde gönderilen toplam bayt sayısı
bwd.iat.mean	Ortalama Geri yönde gönderilen iki paket arasındaki ortalama süre	bwd.win.byt	İlk pencerede geriye doğru gönderilen toplam bayt sayısı
bwd.iat.std	Geri yönde gönderilen iki paket arasındaki standart sapma süresi	fwd.act.pkt	İleri yönde en az 1 bayt TCP veri yüküne sahip paketlerin sayısı
bwd.iat.max	Geri yönde gönderilen iki paket arasındaki maksimum süre	fwd.seg.min	İleri yönde gözlemlenen minimum segment boyutu

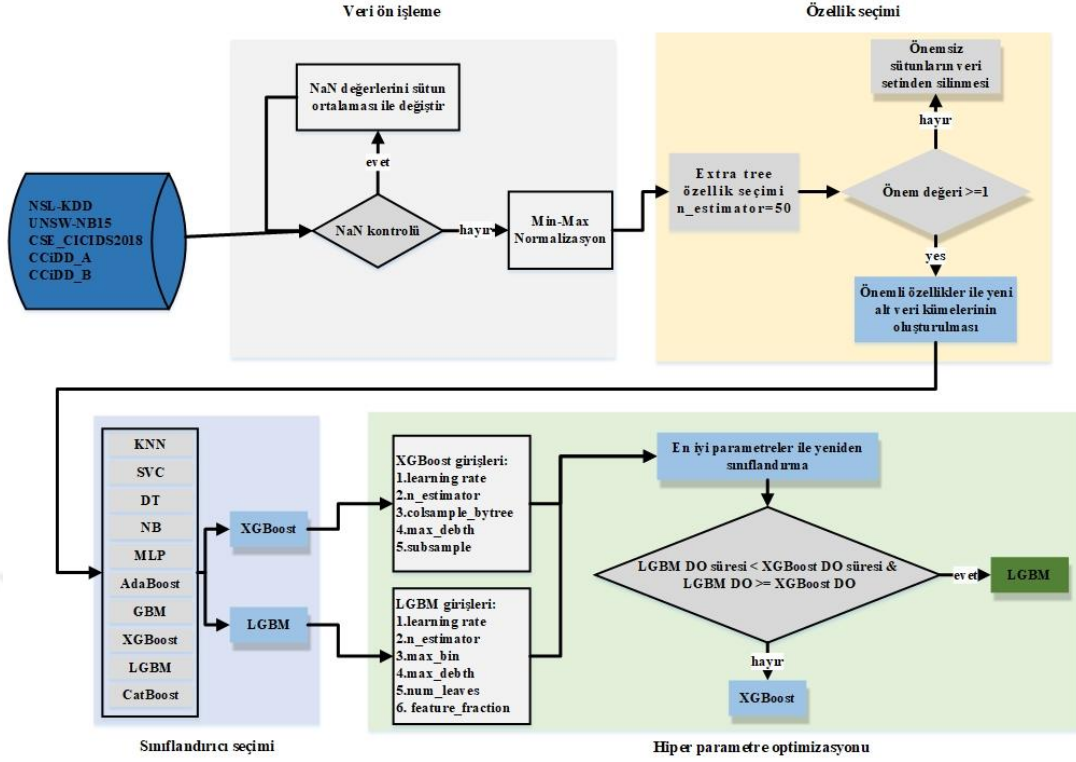
Tablo 3.4. (Devamı)

bwd.iat.min	Geri yönde gönderilen iki paket arasındaki minimum süre	atv.avg	Bir akışın boştaki olmada önce etkin olduđu ortalama süre
fwd.psh.flag	İleri yönde hareket eden paketlerde PSH bayrağının ayarlanma sayısı (UDP için 0)	atv.std	Bir akışın boştaki olmada önce etkin olduđu standart sapma süresi
bwd.psh.flag	Geri yönde hareket eden paketlerde PSH bayrağının ayarlanma sayısı (UDP için 0)	atv.max	Bir akışın boştaki olmada önce etkin olduđu maksimum süre
fwd.urg.flag	URG bayrağının ileri yönde hareket eden paketlerde ayarlanma sayısı (UDP için 0)	atv.min	Bir akışın boştaki olmada önce etkin olduđu minimum süre
bwd.urg.flag	Geri yönde hareket eden paketlerde URG bayrağının ayarlanma sayısı (UDP için 0)	idl.avg	Bir akışın aktif hale gelmeden önce boştaki kaldığı ortalama süre
fwd.hdr.len	İleri yönde başlıklar için kullanılan toplam bayt	idl.std	Bir akışın aktif hale gelmeden önce boştaki kaldığı standart sapma süresi
bwd.hdr.len	Geri yönde başlıklar için kullanılan toplam bayt	idl.max	Bir akışın aktif hale gelmeden önce boştaki kaldığı maksimum süre
Fwd. Packets/s	Saniyedeki ileri paket sayısı	idl.min	Bir akışın aktif hale gelmeden önce boştaki kaldığı minimum süre
Bwd. Packets/s	Saniyede geriye dönük paket sayısı	classes	sınıflar

3.1.3. Önerilen Metod

Bu çalışma, yeni nesil bir STS sınıflandırma modelini önermektedir. Önerilen yöntem veri ön işleme, özellik seçimi, en iyi sınıflandırıcı seçimi ve hiper parametre optimizasyonu olmak üzere dört adımdan oluşmaktadır. Önerilen yöntemin akış diyagramı Şekil 3.5’te verilmiştir.

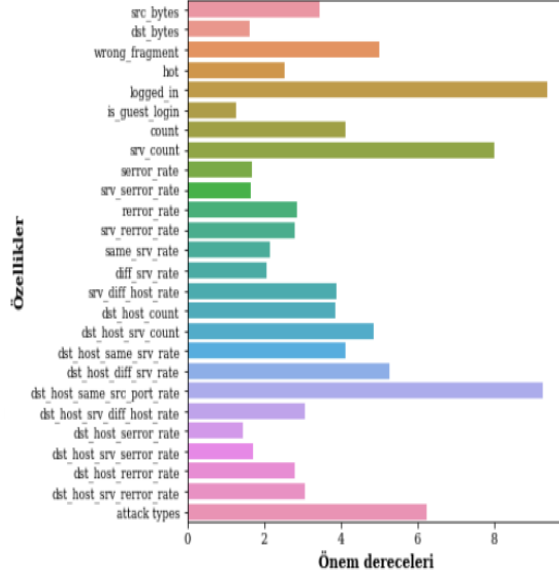
Veri ön işleme adımı: Bu aşamada, veri setlerinde boş değerlerin (NaN) olduğu sütunların ortalaması alınmış ve boş değerlerin yeri, alınan sütun ortalamaları ile değiştirilmiştir. Böylece veri setleri boş değerlerden arındırılmıştır. Daha sonra veri setleri min-maks normalizasyon işlemi ile normalize edilmiştir. Bu aşamanın son adımında, veri setleri %70 eğitim ve %30 test verisi olarak ayrıştırılmıştır. Eğitim ve test ayrımını yaparken “random_state” parametresi 42 olarak seçilmiştir. Bu değer, birçok veri işleme kodunda sıklıkla kullanılan bir değer olduğu için seçilmiştir.



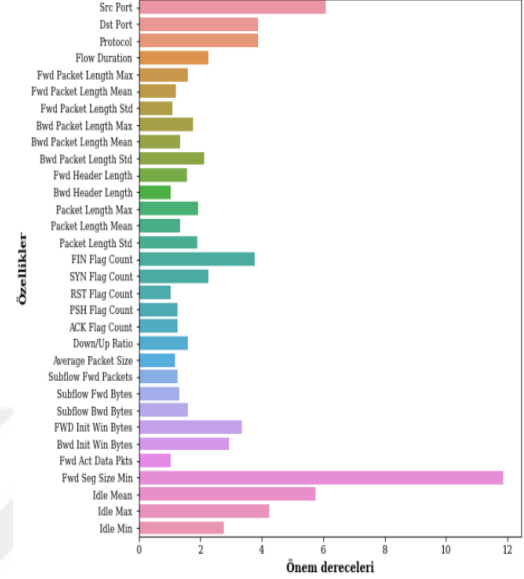
Şekil 3.5. Önerilen yöntemin akış şeması

Özellik seçimi adımı: Önerilen yöntemin ikinci adımında Extra Tree algoritması ile tüm veri setlerini temsil eden en iyi öz nitelikler seçilmiştir. Ekstra Ağaç algoritmasında kullanılacak ağaç sayısını belirleyen “n_estimator” parametresi 50 olarak seçilmiştir. Böylece çok fazla ağaç oluşturulması engellenmiş ve işlem yükü azaltılmıştır. Ekstra Ağaç algoritması, veri setlerindeki tüm özellikleri önemlerine göre ağırlıklandırmıştır. Ağırlık değeri 1 ve 1’den büyük olan özellikler en önemli özellikler olarak seçilmiştir. Tüm veri setleri için seçilen özelliklerden yeni alt veri setleri oluşturulmuştur. Ekstra Ağaç algoritması sonucunda NSL-KDD veri setinden 26 öz nitelik, UNSW-NB15 veri setinden 29 öz nitelik ve CSE-CIC-IDS2018 veri setinden 37 öz nitelik en önemli öz nitelikler olarak seçilmiştir. Önerilen veri seti iki şekilde işlenmiştir. İlk işlemde önerilen CCI_DD veri setinde e-ticaret, sosyal medya, web uygulamaları, uzak masaüstü, toplantılar ve P2P sınıfları ayrı ayrı bölünmüştür. Sonuç olarak 13 ana sınıftan oluşan CCI_DD_A veri seti elde edilmiştir. İkinci işlemde CCI_DD veri setindeki e-ticaret, sosyal medya, web uygulamaları, uzak masaüstü, toplantılar ve P2P sınıfları literatürde olduğu gibi “Normal Trafik” sınıfı altında birleştirilmiş ve 8 ana sınıftan oluşan CCI_DD_B veri seti oluşturulmuştur. Extra Tree algoritması sonucunda; CCI_DD_A veri setinden 41, CCI_DD_B veri setinden 32 öz nitelik en önemli öz nitelikler olarak seçilmiştir. Her bir veri setinde seçilen en önemli özellikler ve özelliklerin önem derecesi Şekil 3.6’daki grafiklerde gösterilmiştir. Her bir veri setinde seçilen en önemli özellikler ve özelliklerin

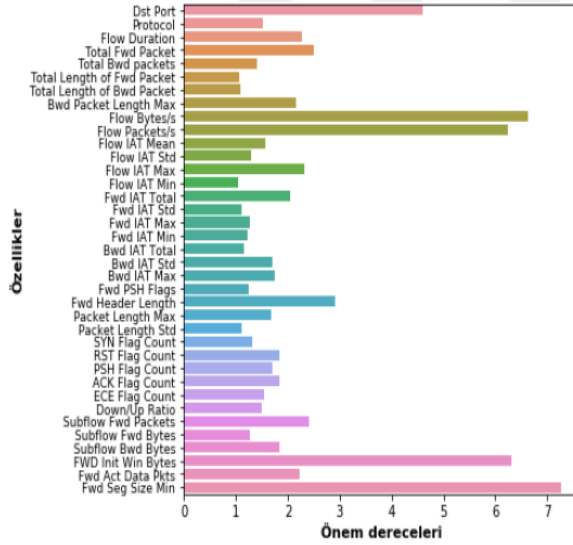
önem derecesi Şekil 3.6a, Şekil 3.6b, Şekil 3.6c, Şekil 3.6d, Şekil 3.6e’ deki grafiklerde gösterilmiştir.



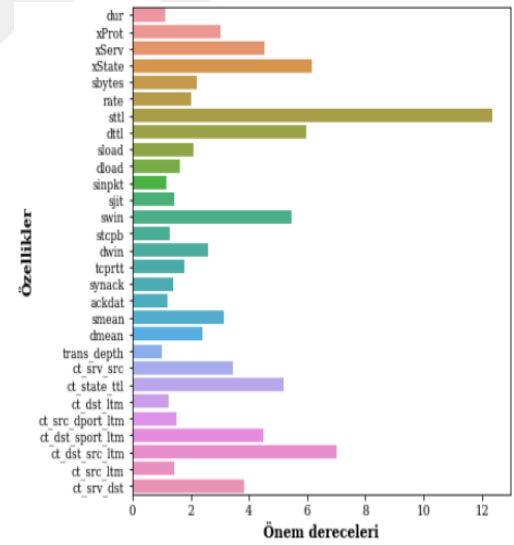
a) NSL-KDD veri seti



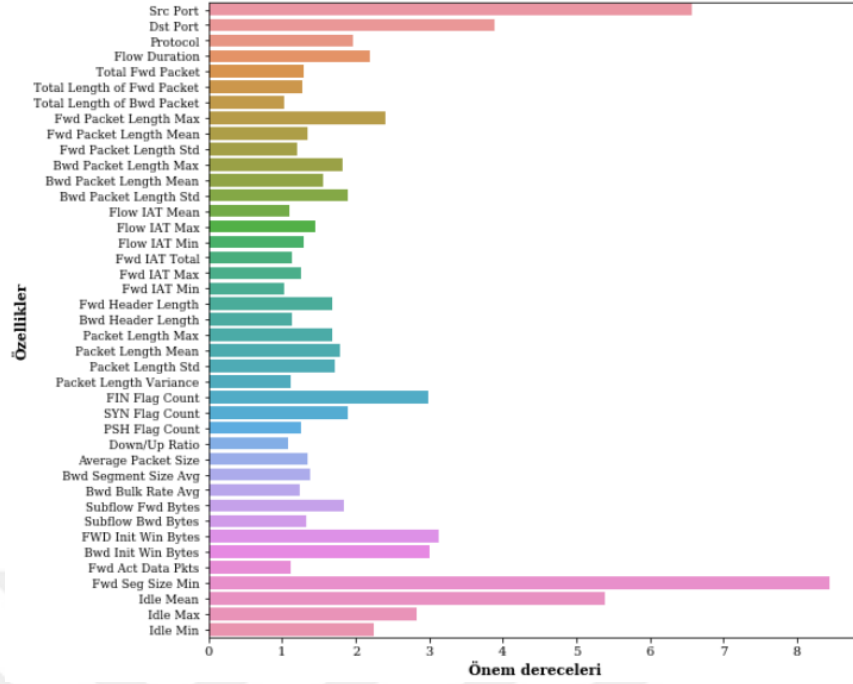
b) CCiDD_B veri seti



c) CSE-CIC-IDS2018 veri seti



d) UNSW-NB15 veri seti



Şekil 3.6. Extra Tree algoritmasından sonra seçile en iyi özellikler ve önem dereceleri

Sınıflandırıcı seçim adımı: Çalışmadaki veri setlerini sınıflandırmak için KNN, SVM, NB, DT, MLP, AdaBoost, GBM, LGBM, XGBoost ve CatBoost olmak üzere on sınıflandırma algoritması kullanılmıştır. Önerilen yöntem, Intel Core i7-4510U işlemci, 16 GB RAM ve Windows 10 işletim sistemine sahip bir dizüstü bilgisayarda gerçekleştirilmiştir. Kodlar Python 3.7 ortamında Scikit-learn, Matplotlib ve Seaborn kütüphaneleri kullanılarak yazılmıştır. Her sınıflandırıcı için Doğruluk, Hassasiyet, Duyarlılık ve F-Ölçüsü değerleri hesaplanmıştır. Alınan sonuçlar Tablo 3.5’te verilmiştir.

Tablo 3.5. Sınıflandırma sonuçları

Sınıflandırıcılar	Metrikler	Veri Setleri				
		NSL-KDD	UNSW-NB15	CSE-CIC-IDS2018	CCiDD_A	CCiDD_B
KNN	Doğruluk	0.969	0.8595	0.967	0.7395	0.889
	Hassasiyet	0.97	0.86	0.96	0.74	0.89
	Duyarlılık	0.96	0.86	0.96	0.74	0.89
	F-Ölçüsü	0.97	0.86	0.97	0.74	0.89
DVM	Doğruluk	0.509	0.7708	0.4895	0.372	0.435
	Hassasiyet	0.30	0.77	0.53	0.22	0.40
	Duyarlılık	0.51	0.77	0.49	0.37	0.44
	F-Ölçüsü	0.35	0.77	0.46	0.26	0.36
NB	Doğruluk	0.3524	0.655	0.7025	0.1954	0.258
	Hassasiyet	0.59	0.65	0.78	0.19	0.38
	Duyarlılık	0.35	0.66	0.70	0.20	0.26

Tablo 3.5. (Devamı)

	F-Ölçüsü	0.23	0.66	0.66	0.16	0.25
MLP	Doğruluk	0.9805	0.788	0.7497	0.3982	0.444
	Hassasiyet	0.98	0.85	0.79	0.31	0.48
	Duyarlılık	0.98	0.79	0.75	0.4	0.41
	F-Ölçüsü	0.98	0.78	0.75	0.3	0.32
DT	Doğruluk	0.995	0.961	0.99	0.9295	0.9977
	Hassasiyet	1.00	0.96	1.00	0.92	1.00
	Duyarlılık	1.00	0.96	1.00	0.93	1.00
	F-Ölçüsü	1.00	0.96	1.00	0.93	1.00
AdaBoost	Doğruluk	0.7612	0.955	0.7942	0.5182	0.5929
	Hassasiyet	0.83	0.96	0.79	0.46	0.43
	Duyarlılık	0.76	0.96	0.74	0.52	0.590
	F-Ölçüsü	0.78	0.96	0.74	0.41	0.49
GBM	Doğruluk	0.998	0.963	0.9984	0.9434	0.9997
	Hassasiyet	1.00	0.96	1.00	0.94	1.00
	Duyarlılık	1.00	0.96	1.00	0.94	1.00
	F-Ölçüsü	1.00	0.96	1.00	0.94	1.00
LGBM	Doğruluk	0.9983	0.9777	0.999	0.9422	0.9995
	Hassasiyet	1.00	0.98	1.00	0.94	1.00
	Duyarlılık	1.00	0.98	1.00	0.94	1.00
	F-Ölçüsü	1.00	0.98	1.00	0.94	1.00
XGBoost	Doğruluk	0.9986	0.9779	0.99923	0.9502	0.9998
	Hassasiyet	1.00	0.98	1.00	0.95	1.00
	Duyarlılık	1.00	0.98	1.00	0.95	1.00
	F-Ölçüsü	1.00	0.98	1.00	0.95	1.00
CatBoost	Doğruluk	0.9985	0.9776	0.9984	0.9457	0.9998
	Hassasiyet	1.00	0.98	1.00	0.95	1.00
	Duyarlılık	1.00	0.98	1.00	0.95	1.00
	F-Ölçüsü	1.00	0.98	1.00	0.95	1.00

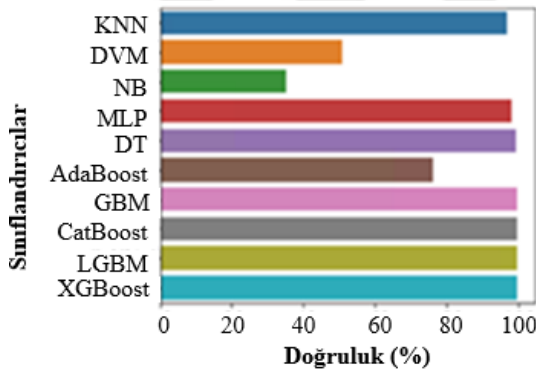
Tablo 3.5’te görüldüğü gibi Boosting algoritmaları tüm veri setlerinin sınıflandırma performansını artırmıştır. Verilen sonuçlar ortalama değerleri temsil etmektedir. Her bir veri seti için Tablo 3.5’e göre yapılan analizler aşağıdaki gibidir.

- NSL-KDD veri seti için DT, GBM, LGBM, XGBoost ve CatBoost algoritmalarının sınıflandırmada oldukça etkili olduğu görülmektedir. Bu veri seti için XGBoost sınıflandırıcısı %99.86 doğruluk değerine ulaşmıştır. DT, GBM, LGBM, XGBoost, CatBoost sınıflandırıcıları için hassasiyet, duyarlılık ve F-Ölçüsü değerleri %100’e ulaşmıştır.
- UNSW-NB15 veri seti için en iyi performans, %97,79 doğruluk oranıyla XGBoost sınıflandırıcısı ile elde edilmiştir. LGBM, XGBoost ve CatBoost algoritmaları da benzer doğruluk değerlerine ulaşmıştır. Üç sınıflandırıcının da hassasiyet, duyarlılık ve F-Ölçüsü değerleri %98 olmuştur.
- CSE-CIC-IDS2018 veri seti için en iyi performans DT, GBM, LGBM, XGBoost ve Catboost sınıflandırıcıları ile sağlanmıştır. Bu sınıflandırıcılar ile başarı oranı %99,9

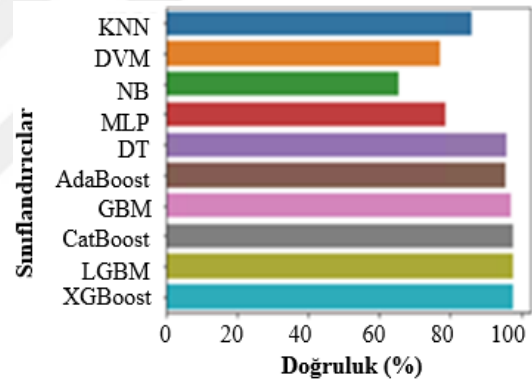
seviyeleri yakalanmıştır. DT, GBM, LGBM, XGBoost, CatBoost sınıflandırıcıları için hassasiyet, duyarlılık ve F-Ölçüsü değerleri %100'e ulaşmıştır.

- Çalışmada önerilen 13 sınıflı CCiDD_A veri seti için en iyi doğruluk oranı %95,02 ile XGBoost sınıflandırıcısı tarafından sağlanmıştır. Ayrıca GBM, LGBM ve CatBoost sınıflandırıcıları XGBoost ile yakın doğruluk oranlarına ulaşmıştır.
- Çalışmada önerilen CCiDD_B veri seti için %99,98 doğruluk değerine ulaşılmıştır. Bu durum CICFlowmeter yazılımının normal ve anormal trafiği özelliklerine ayırmadaki başarısının yanı sıra önerilen yöntemin de başarımını göstermektedir.
- Tablonun tamamında görüldüğü gibi, DT sınıflandırıcısı tüm veri setleri için Boosting yöntemlerine yakın bir şekilde performans göstermiştir. Bunun temel nedeni Boosting yöntemlerinin karar ağaçlarına dayalı olmasıdır.

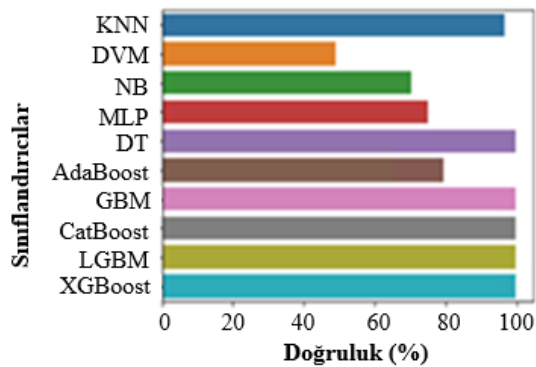
Tablo 3.5'te her bir veri seti için elde edilen performans oranlarının görsel karşılaştırması Şekil 3.7'de verilmiştir.



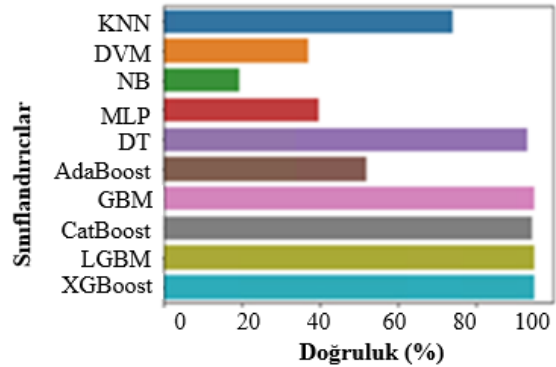
a) NSL_KDD veri seti performansları



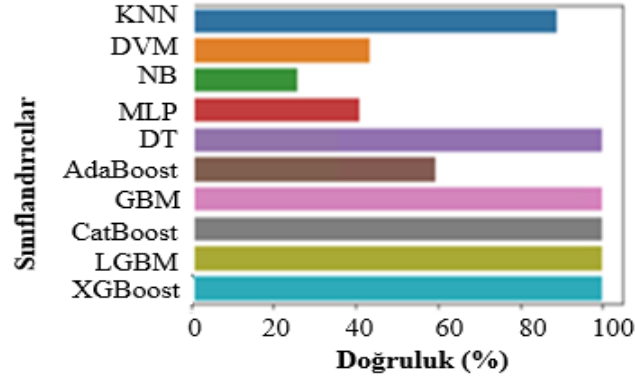
b) UNSW-NB15 veri seti performansları



c) CSE-CIC-IDS2018 veri seti performansları



d) CCiDD_A veri seti performansları



e) CCiDD_B veri seti performansları

Şekil 3.7. Her veri seti için alınan başarımların görsel karşılaştırması

Şekil 3.7’de görüldüğü gibi DT, GBM, LGBM, XGBoost ve CatBoost yöntemlerinin tüm veri setleri için sınıflandırma başarımlarını arttırdığı görülmektedir. XGBoost ve LGBM sınıflandırıcıları, tüm veri setleri için en iyi performansı göstermiştir. Önerilen yöntemin son adımında sınıflandırma performansının artırılması için XGBoost ve LGBM sınıflandırıcılarına hiper parametre optimizasyonu uygulanmıştır.

Hiper parametre optimizasyonu adımı: Veri ön işleme ve öznitelik seçimi adımlarından sonra elde edilen yeni veri setleri, önerilen makine öğrenmesi yöntemleri ile sınıflandırılmış ve performansları karşılaştırılmıştır. Buna göre tüm veri setleri için en başarılı ve en hızlı sonuçlar LGBM ve XGBoost sınıflandırıcıları ile elde edilmiştir. Son adımda ise tüm veri setleri için LGBM ve XGBoost sınıflandırıcılarının hiper parametreleri optimize edilmiştir. Böylece tüm veri setleri için en uygun sınıflandırıcı ve hiper parametreler ayarlanmıştır. Kullanılan hiper parametreler ve değerleri Tablo 3.6’da verilmiştir. Hiper parametre optimizasyonu tüm veri setlerine ayrı ayrı uygulanmış ve optimizasyonlar 5 kat çapraz doğrulama ile yapılmıştır. “learning_rate”, “colsample_bytree”, “subsample”, “n_estimators” ve “max_debth” parametreleri XGBoost algoritması için optimize edilirken, “learning_rate”, “n_estimators”, “max_bin”, “num_leaves”, “max_debth” ve “feature_fraction” parametreleri de LGBM algoritması için optimize edilmiştir. “learning_rate”, “colsample_bytree”, “subsample”, “n_estimators”, “max_debth”, “max_bin”, “num_leaves” ve “feature_fraction” değerleri, sırasıyla [0.02-0.2], [0.1-0.5], [0.8-2], [200-600], [3-7], [400-800], [10-50], [0.1-0.9] aralıklarından seçilmiştir.

Tablo 3.6. Hiper parametre optimizasyonu sonucu elde edilen parametreler

XGBoost hiper parametreleri					
Hiper parametreler	NSL-KDD	UNSW-NB15	CSE-CICIDS-2018	CCiDD_A	CCiDD_B
learning_rate	0.1	0.1	0.1	0.1	0.1

Tablo 3.6. (Devamı)

colsample_bytree	0.5	0.5	0.5	0.4	0.5
subsample	0.8	0.8	0.8	0.8	0.8
n_estimators	500	600	600	800	500
max_depth	8	8	8	4	6
LGBM hiper parametreleri					
Hiper parametreler	NSL-KDD	UNSW-NB15	CSE-CICIDS-2018	CCiDD_A	CCiDD_B
learning_rate	0.1	0.2	0.2	0.2	0.1
n_estimators	500	600	600	500	500
max_bin	500	800	800	490	490
max_depth	5	5	5	3	3
num_leaves	20	20	20	10	20
feature_fraction	0.6	0.9	0.9	0.7	0.7

3.2. Bulgular

Çalışmanın son aşamasında, hiper parametre optimizasyonu işlemi ile LGBM ve XGBoost sınıflandırıcılarının optimum parametreleri belirlenmiştir. Araştırmada kullanılan veri setleri elde edilen hiper parametreler ile sınıflandırılmış ve performansları Tablo 3.7’de sunulmuştur. Ayrıca hiper parametre optimizasyonu sonrasında modellerin sınıflandırma süreleri (Doğruluk süresi) hesaplanmıştır.

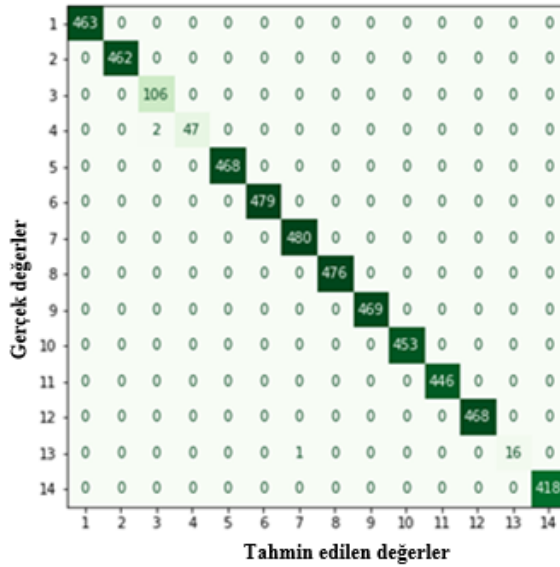
Tablo 3.7. Hiper parametre optimizasyonundan sonra veri setleri performansları

Sınıflandırıcılar / Performans metrikleri		Veri Setleri				
		NSL-KDD	UNSW-NB15	CSE-CIC-IDS2018	CCiDD_A	CCiDD_B
LGBM	Doğruluk	0.9984	0.9802	0.9994	0.9568	0.9998
	Hassasiyet	1.00	0.98	1.00	0.96	1.00
	Duyarlılık	1.00	0.98	1.00	0.96	1.00
	F-Ölçüsü	1.00	0.98	1.00	0.96	1.00
	Sınıflandırma süresi (saniye)	1.71	1.76	2.96	5.41	2.69
XGBoost	Doğruluk	0.9983	0.9791	0.9992	0.9561	0.9998
	Hassasiyet	1.00	0.98	1.00	0.96	1.00
	Duyarlılık	1.00	0.98	1.00	0.96	1.00
	F-Ölçüsü	1.00	0.98	1.00	0.96	1.00
	Sınıflandırma süresi (saniye)	8.88	17.62	16.77	55.47	17.35

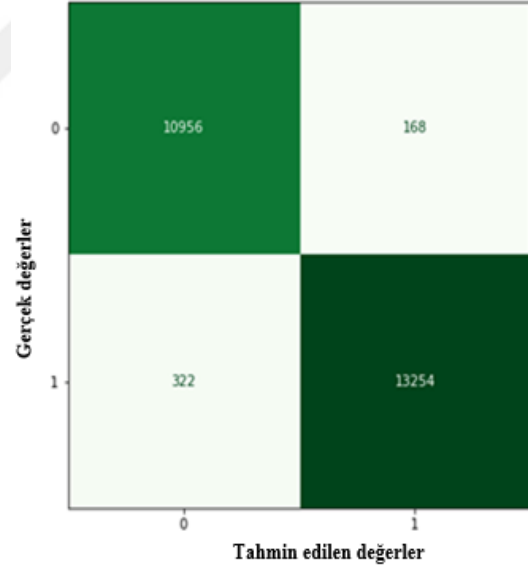
Hiper parametre optimizasyonu sonrası LGBM ve XGBoost sınıflandırıcıları ile elde edilen performanslar Tablo 3.7’de verilmiştir. Tabloya göre hiper parametre optimizasyonunun tüm veri setleri üzerinde olumlu etkisi olmuştur. NSL-KDD, CSE-CIC_IDS2018 ve CCiDD_B veri setleri için LGBM ve XGBoost sınıflandırıcıları neredeyse %100 başarıma ulaşmışlardır. Hiper parametre optimizasyonu ana etkisini UNSW-NB15 ve CCiDD_A veri setleri üzerinde göstermiştir. UNSW-

NB15 veri seti, LGBM ve XGBoost sınıflandırıcılarının varsayılan ayarlarında %97.77 ve %97.79 performans değerlerine sahipken, optimizasyon süreci sonrasında performanslar sırasıyla %98.02 ve %97.91 değerlerine ulaşmıştır. Bu çalışmada önerilen CCiDD_A veri seti, LGBM ve XGBoost sınıflandırıcılarının varsayılan ayarlarında sırasıyla %94.22 ve %95.02 performans gösterirken, optimizasyondan sonra CCiDD_A veri seti için performans değerleri sırasıyla %95.68 ve %95.61 değerlerine ulaşmıştır. LGBM sınıflandırıcısından elde edilen tüm veri setlerinin karmaşıklık matrisleri Şekil 3.8'de verilmiştir.

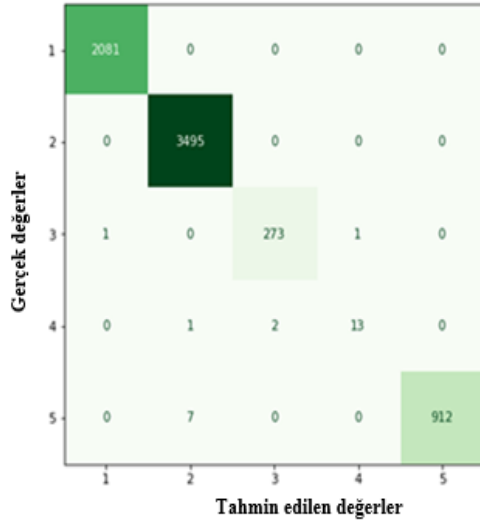
Ayrıca Tablo 3.7'de sınıflandırma süresi, her bir veri seti için saniye cinsinden hesaplanmıştır. Tablodan da görüleceği üzere LGBM sınıflandırıcı sınıflandırma süresini önemli ölçüde azaltmıştır. Doğruluk oranları çok yakın olmasına rağmen LGBM sınıflandırıcı sınıflandırma süresi açısından daha avantajlıdır. LGBM sınıflandırıcısının işlem süresi NSL-KDD, UNSW-NB15, CSE_CIC_IDS2018, CCiDD_A, CCiDD_B veri setleri için sırasıyla 1.71, 1.76, 2.96, 5.41 ve 2.69 saniyedir. XGBoost sınıflandırıcısının işlem süresi sırasıyla 8.88, 17.62, 16.77, 55.47 ve 17.35 saniyedir. Bu değerler, LGBM sınıflandırıcısının yaklaşık 8 kat daha hızlı olduğunu göstermektedir.



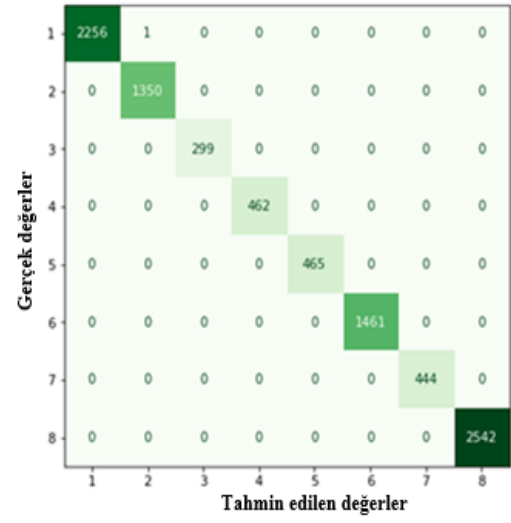
a) CSE-CIC-IDS2018 veri seti



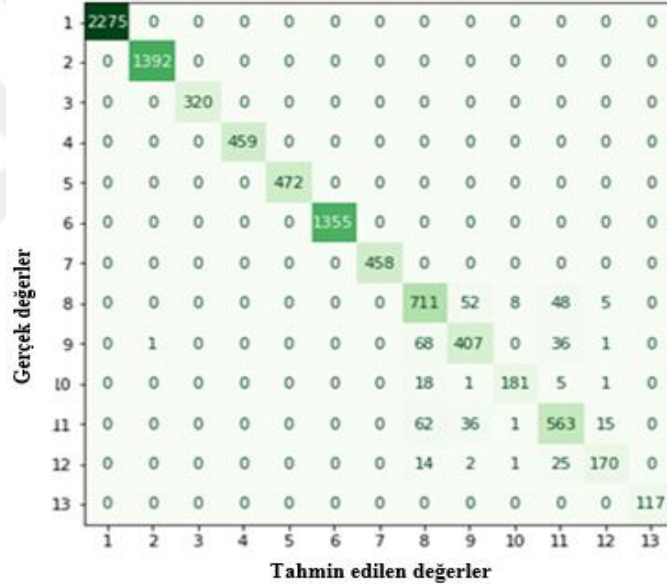
b) UNSW-NB15 veri seti



c) NSL-KDD veri seti



d) CCiDD_B veri seti



e) CCiDD_A veri seti

Şekil 3.8. Tüm veri setleri için alınan karmaşıklık matrisleri

CCiDD_A veri setinde %95.68 başarı oranı elde edilirken, CCiDD_B veri setinde daha yüksek performans elde edilmiş ve değer %99.98 olmuştur. Şekil 3.8'e'deki karmaşıklık matrisinde görüldüğü gibi, Tablo 3.2'de de verilen 8, 9, 10, 11, 12 ve 13 normal trafik sınıfları arasında geçişler vardır. Bu geçişlerin ana nedeni, çoğu uygulamanın http veya https protokollerini kullanmasıdır. Özellikle Şekil 3.8'e'deki 8, 9, 11 sınıfları arasındaki geçiş yüksektir. Bunun temel nedeni, bu sınıflardaki çoğu uygulamanın http veya https protokollerini kullanmasıdır.

3.3. Tartışma

NSL-KDD, UNSW-NB15 ve CSE-CIC-IDS2018 veri setlerini kullanan birçok literatür çalışması bulunmaktadır. Bu bölümde NSL-KDD, UNSW-NB15 ve CSE-CIC-IDS2018 veri setlerinin sonuçları literatür çalışmaları ile karşılaştırılmıştır. NSL-KDD veri seti için yapılan bazı literatür çalışmaları ve önerilen yöntemin karşılaştırılması Tablo 3.8’de verilmiştir.

Tablo 3.8. NSL-KDD veri seti için karşılaştırmalı analiz

Referans	Sınıflandırma Yöntemi	Özellik Seçimi	Doğruluk(%)
[44]	J48	-	93.82
	NB		81.66
	NB Tree		93.51
	RF		92.79
	Random Tree		92.53
	MLP		92.26
	SVM		65.01
[88]	DNN	DAE	83.33
[86]	IHSELMSTS	-	99.84
[87]	Fuzzy C-Means	Spider Monkey	88.66
[89]	DNN	-	91.5
[90]	Parallel K-Medoids	KNN	91.7
[69]	k-Means	GAA-ADS	99.7
Önerilen Yöntem	LGBM	Extra Tree algoritması	99.84
	XGBoost		99.83

Tablo 3.8’de görüldüğü gibi önerilen yöntem NSL-KDD veri seti için %99.84 başarı sağlamıştır. Önerilen yöntemin en büyük avantajı, özellik seçimi adımında en iyi 26 özelliğin seçilmesidir. Bu adım, sınıflandırma performansını olumlu yönde etkilemiştir. Tablo 3.9, UNSW-NB15 veri seti için önerilen yöntemi ve literatür karşılaştırmasını göstermektedir.

Tablo 3.9. UNSW-NB15 veri seti için karşılaştırmalı analiz

Referans	Sınıflandırma Yöntemi	Özellik Seçimi	Doğruluk(%)
[89]	DNN	-	99.2
[90]	Parallel K-Medoids	KNN	94.0
[169]	DT	GA-LR	81.42
[64]	RF	BBA+CAFF	99.27
		BBA+FU	99.43
		BBA+FSFF	99.09
[93]	DT	-	85.56
	LR		83.15
	NB		82.07
	ANN		81.34
	EM Clustering		78.47

Tablo 3.9. (Devamı)

[92]	GMM	-	96.7
[69]	k-Means	GAA-ADS	92.8
Önerilen Yöntem	LGBM	Extra Tree algoritması	98.02
	XGBoost		97.91

UNSW-NB15 veri seti için elde edilen doğruluk değerleri literatüre benzerdir. Bu veri seti için en yüksek performans hiper parametre optimizasyonu sonrasında %98,02 ile LGBM sınıflandırıcısı ile elde edilmiştir. Tablo 3.10, CSE-CIC-IDS2018 veri seti için önerilen yöntemi ve literatür karşılaştırmasını göstermektedir.

Tablo 3.10. CSE-CIC-IDS2018 veri seti için karşılaştırmalı analiz

Referans	Sınıflandırma Yöntemi	Özellik Seçimi	Doğruluk(%)
[88]	DNN	DAE	95.79
[65]	ANN	-	99.97
	RF		99.83
	KNN		99.73
	SVM		99.8
	Adaboost		99.96
	NB		99.2
[40]	DNN	-	97.28
	RNN		97.31
	CNN		93.76
	RBM		97.28
	DBN		97.30
	DBM		97.371
	DA		97.372
[68]	ANN	MLP	99.97
Önerilen Yöntem	LGBM	Extra Tree algoritması	99.94
	XGBoost		99.92

CSE-CIC-IDS2018 veri seti için Tablo 3.10’da görüldüğü gibi literatür ile benzer ve yüksek performans değerleri elde edilmiştir. Önerilen yöntem, CSE-CIC-IDS2018 veri seti için %99,94 performans değerine ulaşmıştır. Önerilen yöntem ile %99.94 doğruluğa ulaşan LGBM sınıflandırıcısının performansı literatürdeki birçok yöntemden daha başarılı sonuçlar elde etmiştir. Tablo 3.8, Tablo 3.9 ve Tablo 3.10’da yapılan karşılaştırmalar sonucunda, önerilen sistemin tüm veri setleri için yüksek sınıflandırma başarımları elde etmiştir.

3.4. Sonuçlar

Saldırı tespit sistemlerinin performansını artırmak için birçok makine öğrenmesi yöntemi önerilmiş olsa da saldırı tespitinde hala istenilen seviyeye ulaşamamıştır. Bu çalışmada gerçek bir sistemde bulunabilecek cihazlara benzer bir laboratuvar ortamı oluşturulmuştur. Daha sonra

sıklıkla kullanılan saldırı türleri ve internet uygulamaları bu laboratuvar ortamında test edilmiştir. Test sürecinde alınan veriler ile CCiDD_A ve CCiDD_B veri setleri oluşturulmuştur. Oluşturulan veri setlerinin performanslarını incelemek için literatürde yaygın olarak kullanılan NSL-KDD, UNSW-NB15 ve CSE-CIC-IDS2018 veri setleri kullanılmıştır. Tüm veri setlerinde en optimum öznelilikleri seçmek için Extra Tree algoritması önerilmiştir. Ekstra Ağaç algoritması ile tüm veri setlerinin optimum özellikleri seçildikten sonra, veri setleri makine öğrenmesi yöntemleriyle sınıflandırılmıştır. Sınıflandırma sonucunda LGBM ve XGBoost sınıflandırıcıları diğer makine öğrenmesi yöntemlerinden daha başarılı olmuştur. Sınıflandırma performansını artırmak için XGBoost ve LGBM sınıflandırıcılarına hiper parametre optimizasyonu uygulanmıştır. Hiper parametre optimizasyonu sonrasında XGBoost ve LGBM benzer performans sonuçları elde etmesine rağmen, LGBM sınıflandırıcı sınıflandırma hızı ve performans açısından daha başarılıdır. Tüm veri setleri için en iyi sonuçlar LGBM algoritması ile elde edilmiştir. LGBM algoritması NSL-KDD, UNSW-NB15, CSE-CIC-IDS2018, CCiDD_A ve CCiDD_B veri setleri için sırasıyla %99.84, %98.02, %99.94, %95.68 ve %99.98 performans değerlerine ulaşmıştır. Ayrıca IDS sistemlerinde saldırıların tespit süresi de oldukça kritik bir konudur. Saldırıları hızlı bir şekilde tespit edilebilirse, saldırıyı engellemek daha hızlı olacaktır. LGBM sınıflandırıcısının sınıflandırma süresini kısalttığı gözlemlenmiştir.

4. SPA-IDS: VMD VE YİNELEMELİ ÖZELLİK SEÇİMİNE DAYALI AKILLI BİR SALDIRI TESPİT SİSTEMİ

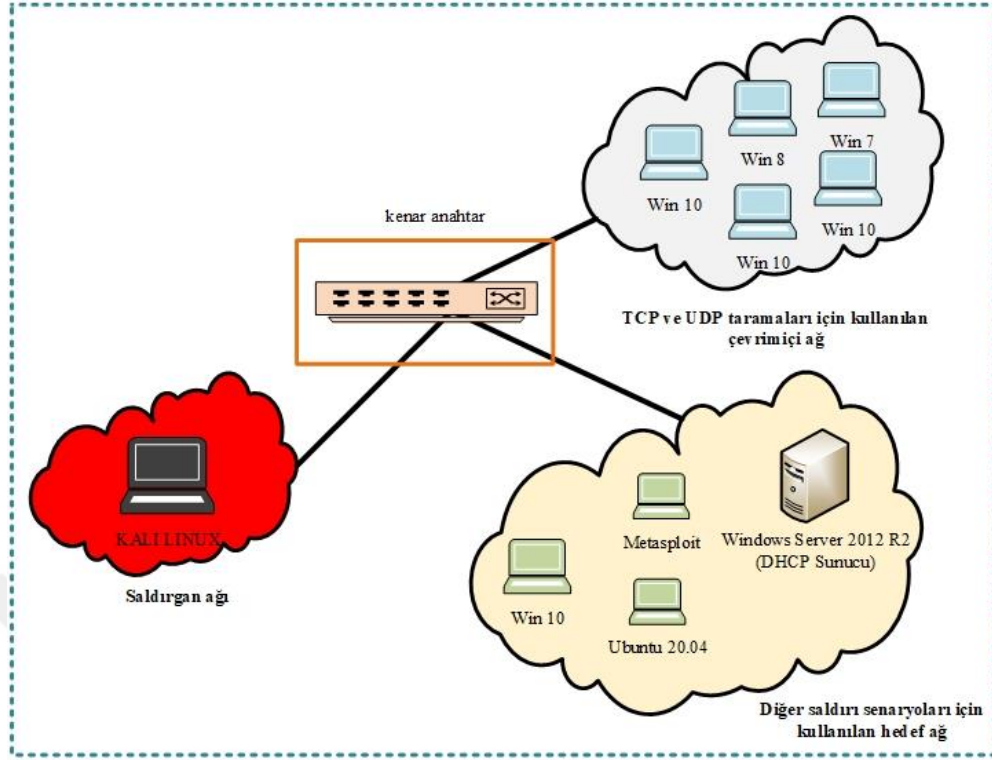
Kritik altyapıların ve bilgi sistemlerinin güvenliğinin sağlanması her zaman zorlu bir sorun olmuştur. STS sistemleri, ağ yapılarının ve bilgi sistemlerinin güvenliğinin sağlanmasında vazgeçilmez bir unsur haline gelmiştir. Bu çalışmada, anahtarlama cihazlarının portlarında oluşan anomali paket sayıları alınarak, STS sistemleri için yeni bir Switchport Anomaly based Intrusion Detection System (SPA-IDS) veri seti önerilmiştir. Daha sonra elde edilen veri setinin sınıflandırma başarımını artırmak için veri setine öznelik seçimi uygulanmıştır. Bu bölüm çalışmasının katkıları şunlardır.

- Bu çalışmada, anahtarlama cihazların portları üzerinden alınan veriler ile özgün bir STS veri seti oluşturulmuştur. Veri seti, yerel ağlarda sıklıkla meydana gelen saldırı senaryoları kullanılarak çevrimiçi bir yerel alan ağı (Local Area Network-LAN) üzerinde oluşturulmuştur.
- Saldırı senaryolarının, anahtarlama cihazlarının portları üzerindeki etkileri kullanılarak OSI referans modelinin ikinci katmanındaki saldırıların genel özellikleri elde edilmiştir. Böylece önerilen yöntem, ağ yöneticileri için geniş bir ağ görünürlüğü sunmakta ve kritik ağ altyapılarında hizmet kesintilerinin önüne geçmektedir. Ayrıca, bu saldırıları otomatik olarak sınıflandırmak için temel bir makine öğrenme modeli sunulmaktadır.

4.1. Materyal ve Metod

4.1.1. SPA-IDS Veri Seti

Literatürde STS sistemlerinin performansını iyileştirmeye yönelik birçok çalışma bulunmaktadır. Çalışmaların çoğunda mevcut STS veri setleri kullanılmaktadır. Literatürdeki bu döngüyü kırmak için bu çalışmada yeni bir SPA-IDS veri seti oluşturulmuştur. Veri seti saldırgan ve hedef ağlardan oluşan bir ağ mimarisi kullanılarak oluşturulmuştur. Oluşturulan ağ mimarisi Şekil 4.1’de verilmiştir.



Şekil 4.1. SPA-IDS ağ topolojisi

Saldırgan ağındaki Kali Linux işletim sistemine sahip bilgisayardan hedef ağıdaki bilgisayarlara çeşitli saldırılar yapılmıştır. Düzenlenen saldırı türleri, ana saldırı kategorileri, kullanılan saldırı araçları ve her saldırı için elde edilen veri miktarı Tablo 4.1’de özetlenmiştir. Gerçekçi sonuç alabilmek için saldırılar canlı bir ağ trafiği üzerinde gerçekleştirilmiştir. Ayrıca, Şekil 4.2’de gösterilen sinyaller/paketler, bir dakikalık uzunlukta örtüşmeyen bloklara bölünmüştür.

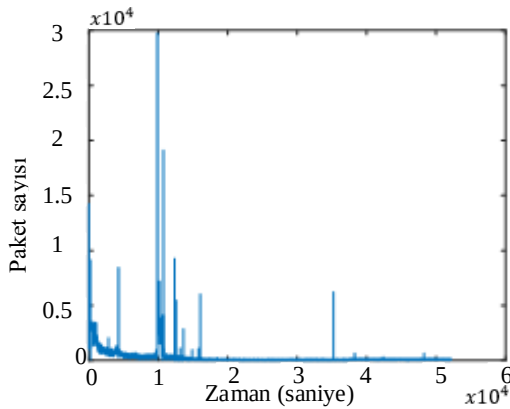
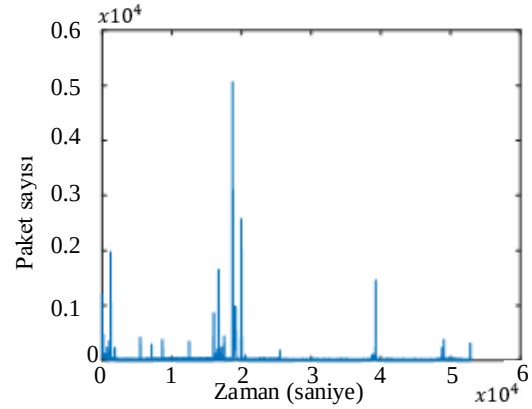
Tablo 4.1. SPA-IDS veri seti için kullanılan saldırı senaryoları ve özellikleri

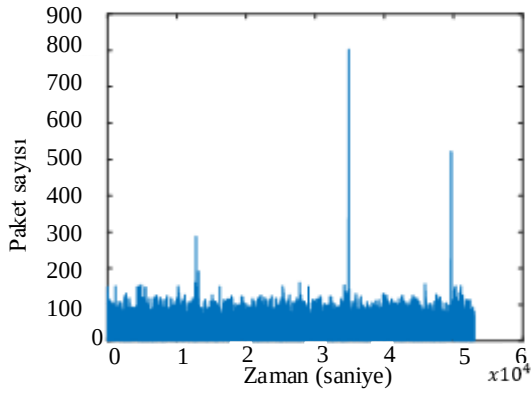
Saldırıların ana kategorisi	Saldırı tipleri	Kullanılan araçlar	Hedef	Alınan veri miktarı	Gözlem sayısı (dakika)
Kaba Kuvvet Saldırıları	FTP	Hydra	Windows Server 2012	53030	883
	RDP	Hydra	Windows Server 2012	56008	933
	SSH	Hydra	Windows 10	52993	883
DHCP Starvation	DHCP Starvation	Yersinia	Metasploit	49112	818
Hizmet Reddi Saldırıları	ICMP	Hping3	Windows 7	52449	874
	SYN	Hping3	Windows 10	50439	840

Tablo 4.1. (Devamı)

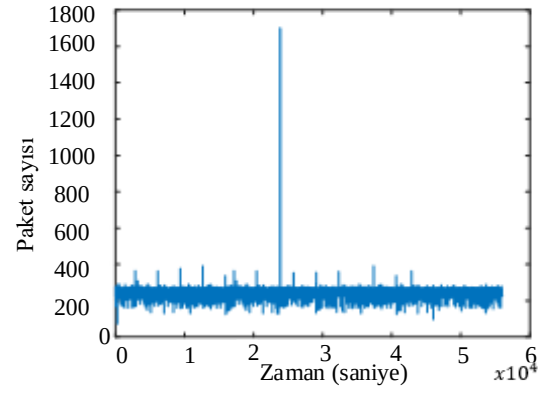
Zafiyet Taramaları	TCP taraması	Nmap	Ağıdaki tüm cihazlar	52130	868
	UDP taraması	Nmap	Ağıdaki tüm cihazlar	52905	881
MAC adres taşıması	MAC adres taşıması	Macof	Anahtarlama cihazı	56212	861
Normal	Normal	-	-	60255	1004
Toplam	-	-	-	535.503	8845

TCP ve UDP taramaları anahtarlama cihazına bağlı tüm bilgisayarlara yapılmıştır. DHCP Starvation saldırısı için Windows Server 2012 R2 sunucusu kurulmuş ve bu sunucu üzerinde alt ağ maskesi 255.0.0.0 olan bir DHCP havuzu oluşturulmuştur. DHCP havuzunun çok geniş tutulmasındaki amaç, IP havuzunun çok hızlı rezerve edilmesini önlemektir. Mac adres taşıma saldırısı, topolojideki anahtarlama cihazına yapılmıştır. Anahtarlama cihazının mac tablosunu doldurmamak için saniyede 800 mac adresi gönderilmiştir. Bu saldırı sırasında anahtarlama cihazına 800 mac adresi gönderildiğinde anahtarlama cihazının portundaki anomali paketi artışı her defasında 800'dür. Bu saldırı senaryosu için Şekil 4.2i'de görüldüğü gibi anahtarlama cihazına saniyede 800 mac adresi gönderildiğinde, anahtarlama cihazı portundaki paket sayısı artışı en az 800'dür. Diğer saldırı senaryolarında olduğu gibi bu saldırı senaryosunda da normal ağ trafiği kullanılmıştır. Bu nedenle Şekil 4.2i'deki grafik; 800 limitli düz bir grafik değil, mac adres taşıma saldırısı üzerine eklenmiş ağ trafiğinin grafiğidir. Normal trafik sınıfı için bilgisayarda youtube web sitesinde 3 video, 2 haber kanalı ve bir e-ticaret sitesi açılmıştır. Google Chrome'a eklenen bir eklenti ile her sayfa 1, 2 ve 3 dakikalık aralıklarla yenilenmiştir. Böylece gerçekçi bir normal ağ trafiği elde edilmeye çalışılmıştır.

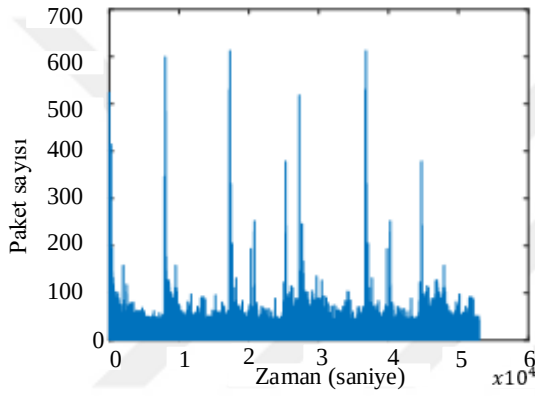
**a)** TCP taraması**b)** UDP taraması



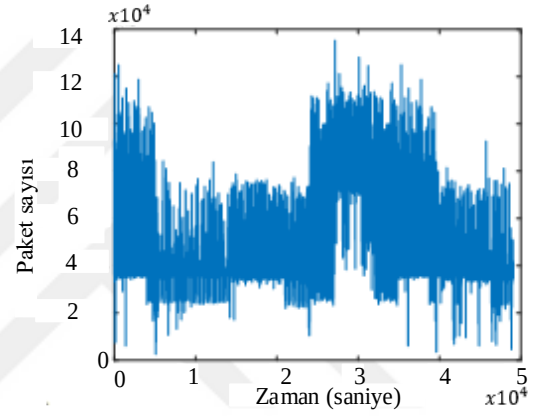
c) FTP kaba kuvvet saldırısı



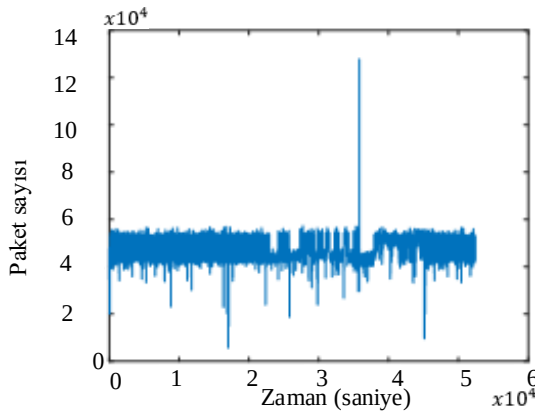
d) RDP kaba kuvvet saldırısı



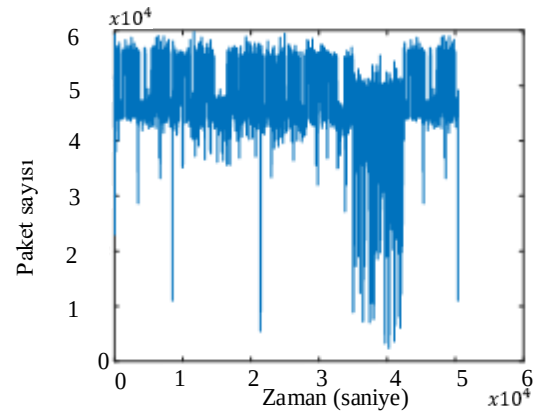
e) SSH kaba kuvvet saldırısı



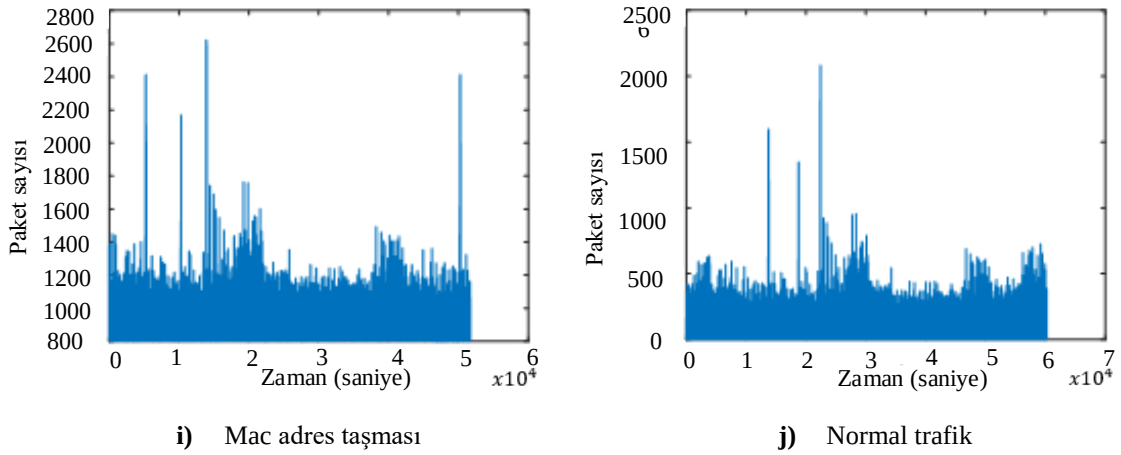
f) DHCP Starvation saldırısı



g) ICMP hizmet reddi saldırısı



h) SYN hizmet reddi saldırısı



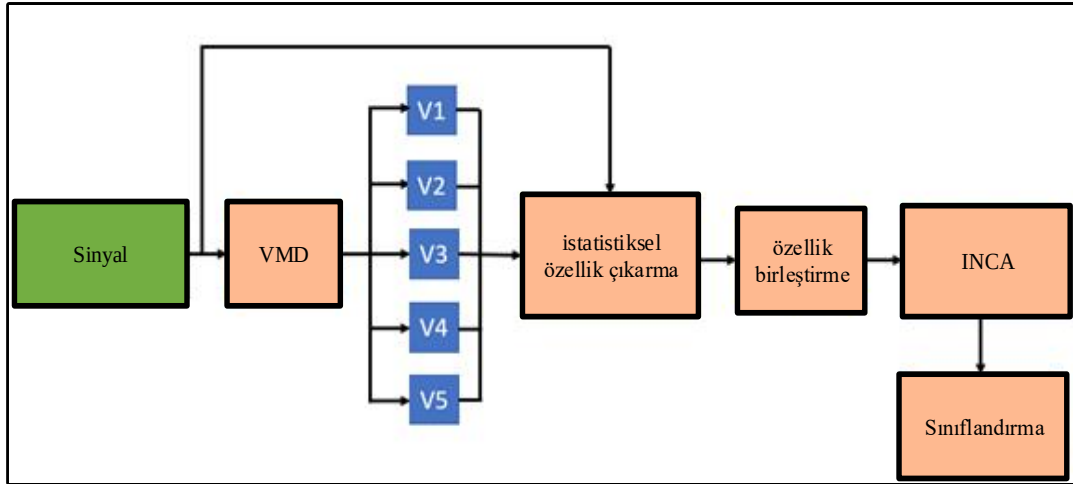
Şekil 4.2. SPA-IDS saldırı karakteristikleri

Veri seti alınırken anahtarlama cihazına Secure Shell (SSH) ile bağlantı yapılmıştır. Ardından anahtarlama cihazı portundaki anormallikleri görüntülemek için gerekli komut çalıştırılmıştır. Her bir saldırı senaryosu için anomali paket sayılarının değişimini veren komutun çıktıları bir gün boyunca alınmış ve bir dosyaya kaydedilmiştir. Daha sonra log dosyasındaki veriler excel ortamına alınmış ve sıralanmıştır. Excel ortamında art arda iki saniye arasındaki paket sayısı farkı her saldırı senaryosu için farklıdır. Örneğin ICMP hizmet reddi saldırısında iki saniye arasındaki paket sayısı 34362 iken RDP kaba kuvvet saldırısındaki fark 229'dur.

Önerilen yöntem, anahtarlama cihazı markasından bağımsızdır. Önerilen yöntem, herhangi bir marka anahtarlama cihazına uygulanabilir. Ancak her markanın otomatik yenileme özelliği olmayabilir. Bu durumu aşmak için basit bir script ile saniyede bir kez istenen komut anahtarlama cihazı üzerinde çalıştırılır ve alınan veriler bir veri tabanına kaydedilebilir.

4.1.2. Önerilen Metod

Bu bölümde, vertical mode decomposition (VMD) ve istatistiksel öznitelik tabanlı saldırı sınıflandırma modeli sunulmuştur [170,171]. Bu model VMD ve istatistik tabanlı öznitelik çıkarma, iterative neighborhood component analysis (INCA) kullanılarak özellik seçimi ve sınıflandırma aşamalarından oluşmaktadır [172]. Sunulan VMD ve INCA tabanlı STS modelinin şematik açıklaması Şekil 4.3'te verilmiştir.



Şekil 4.3.Önerilen VMD ve INCA tabanlı STS sistemi

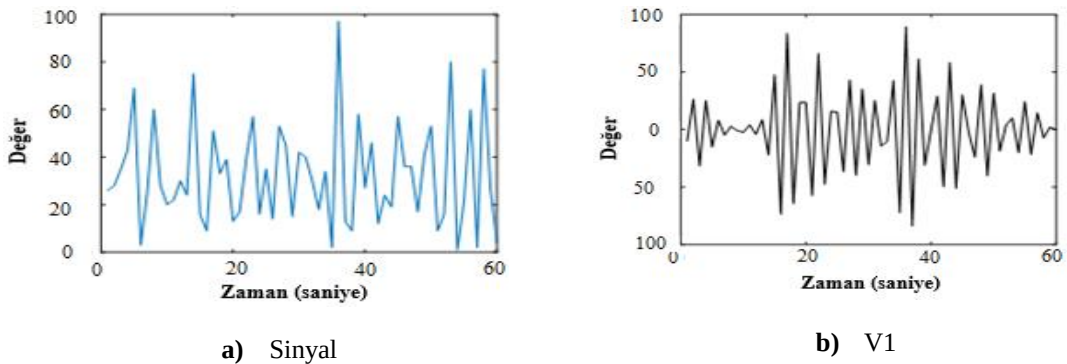
Paket sinyalleri (paketlerin zaman alanındaki farkları) kullanılan anahtarlama cihazından elde edilmiştir. Bu sinyallere VMD uygulanır ve istatistiksel momentler VMD katsayılarına (V1, V2, ..., V5) uygulanır. Özellik birleştirmede $15 \times 6 = 90$ öznitelik elde edilir ve INCA en ayırt edici 48 özniteliği seçer. Seçilen 48 öznitelik DT, BT, KNN ve SVM sınıflandırıcıları kullanılarak sınıflandırılmıştır. Sunulan modelin adımları alt bölümlerde verilmiştir.

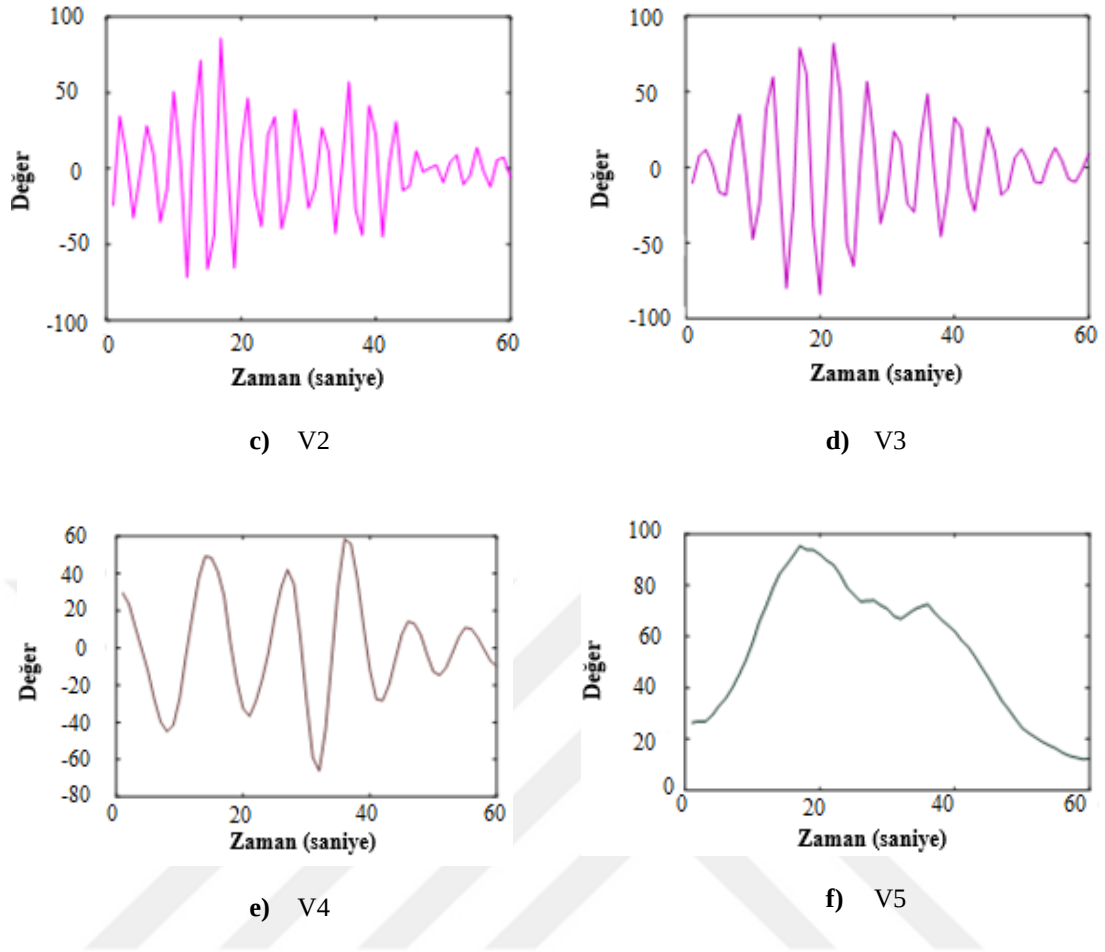
Özellik Çıkarımı: Çalışmanın bu bölümünde elde edilen SPA-IDS verilerinden (sinyallerinden) yeni özellikler çıkartılmıştır. Özellik çıkarımı aşamasında aşağıdaki adımlar uygulanmıştır.

- Sinyaller bir dakikalık uzunluklara bölünmüş ve önerilen sisteme yüklenmiştir.
- Önerilen VMD modeli Denklem 4.1'deki gibi sinyale uygulanmıştır.

$$V_k = VMD(sinyal), k \in \{1, 2, \dots, 5\} \quad (4.1)$$

Denklemden $VMD(.)$ fonksiyonu VMD dönüşümünü tanımlarken, V_k , k 'inci VMD katsayısını tanımlamaktadır. Paket sinyalinin bir örneği ve bu örneğin VMD katsayıları Şekil 4.4'te gösterilmektedir.





Şekil 4.4. Toplanan paket sinyali ve VMD katsayılarının bir örneği

- V_k sinyal paketine ve orijinal sinyale istatistiksel özellik oluşturma adımı uygulanmıştır [173]. Kullanılan istatistiksel momentler Tablo 4.2’de tanımlanmıştır.

Tablo 4.2. Özellik çıkarımı için kullanılan istatistiksel momentler

Momentler	Denklem	Denklem No
Maximum	$\max(P)$	(4.2)
Minimum	$\min(P)$	(4.3)
Orta değer	$\text{median}(P)$	(4.4)
Ortalama	$\frac{\sum_{i=1}^{Ln} P_i}{Ln}$	(4.5)

Tablo 4.2. (Devamı)

Varyans	$\frac{\sum_{i=1}^{Ln} (P_i - aver)^2}{Ln}$	(4.6)
Standart sapma	$\sqrt{\frac{\sum_{i=1}^{Ln} (P_i - aver)^2}{Ln}}$	(4.7)
Aralık	$\max(P) - \min(P)$	(4.8)
Kare kök ortalama	$\sqrt{\frac{\sum_{i=1}^{Ln} P_i^2}{Ln}}$	(4.9)
Enerji (energy)	$\sum_{i=1}^{Ln} P_i^2$	(4.10)
Shannon entropi	$-\sum_{i=1}^{Ln} prob(P_i) * \log(prob(P_i))$	(4.11)
Sure entropi	$-\sum_{i=1}^{Ln} \log(prob(P_i))$	(4.12)
Log enerji entropi	$-\sum_{i=1}^{Ln} \log(prob(P_i))^2$	(4.13)
Çarpıklık	$\frac{\sqrt{Ln(Ln-1)}}{Ln-2} \left(\frac{\frac{1}{Ln} \sum_{i=1}^{Ln} (P_i - aver)^3}{\frac{1}{Ln} \sum_{i=1}^{Ln} (P_i - aver)^2} \right)$	(4.14)
Yığılma	$\frac{Ln-1}{(Ln-2)(Ln-3)} \left[(Ln+1) \left(\left(\frac{\frac{1}{Ln} \sum_{i=1}^{Ln} (P_i - aver)^4}{\frac{1}{Ln} \sum_{i=1}^{Ln} (P_i - aver)^2} \right) - 3 \right) + 6 \right]$	(4.15)
Ortalama mutlak sapma	$\sum_{i=1}^{Ln} \frac{ P_i - aver }{Ln}$	(4.16)

Tablo 4.2’de, P kullanılan tek boyutlu sinyali tanımlar, $prob(.)$ olasılık hesaplama fonksiyonudur, $aver$ ortalama değeri ve Ln ise sinyalin uzunluğudur.

Verilen momentler, kullanılan istatistiksel özniteliklerin oluşturulması ile elde edilmiştir. Burada 4 doğrusal olmayan (enerji, shannon entropi, sure entropi ve log enerji entropi) istatistik ve

11 tane doğrusal istatistik kullanılmıştır. Bu momentler kullanılarak istatistiksel çıkarım ($\vartheta(\cdot)$) fonksiyonu oluşturulmuştur. Bu fonksiyon sinyal paketinden ve VMD katsayılarından öznelilikler çıkarmaktadır.

$$f^1 = \vartheta(packet) \quad (4.17)$$

$$f^t = \vartheta(V_{t-1}), t \in \{2, 3, \dots, 6\} \quad (4.18)$$

Burada f^t uzunluğu 15 olan t 'inci özellik vektörüdür. Oluşturulan özellikler birleştirilmiş ve 90 boyutundaki son vektör elde edilmiştir.

$$X((t-1) * 15 + j) = f^h(j), t \in \{1, 2, \dots, 6\}, j \in \{1, 2, \dots, 15\} \quad (4.19)$$

Burada X birleştirilmiş vektördür ve Denklem 4.19, özellik birleştirme sürecini tanımlar.

Yinelemeli Özellik Seçimi: Önerilen yöntemin son adımında oluşturulan özelliklere INCA özellik seçme algoritması uygulanmış ve en uygun özellikler seçilmiştir. INCA, NCA'nın geliştirilmiş sürümüdür [174]. Özellikleri yinelemeli olarak seçer. En uygun öznelilik vektörünü seçmek için bir hata hesaplayıcı kullanılır. Bu model genellikle hata değeri üretici olarak bir sınıflandırıcı kullanır. Bu araştırmada hata üretici olarak DT kullanılmıştır. Bu hata oluşturucunun doğrulama stratejisi, on kat çapraz doğrulamadır. INCA'yı kullanmanın temel amacı, en uygun özellikleri otomatik olarak seçmektir. INCA özellik seçicisinin sözde kodu Tablo 4.3'te verilmiştir.

Tablo 4.3. INCA algoritması

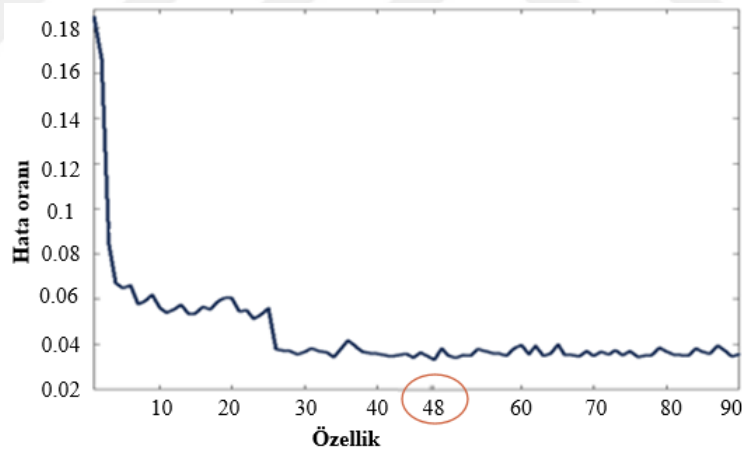
Prosedür: $INCA(X, hedef, C)$
Giriş: dxK boyutunda oluşturulan özellikler (X), hedef boyutu d ve sınıflandırıcı (C)
Çıkış: j boyutlu final özellik vektörü (son)
00: X
01: for $i=1$ to K do
02: $X(:, i) = \frac{X(:, i) - \min(X(:, i))}{\max(X(:, i)) - \min(X(:, i))}$; // Özellik vektörünün her sütununa min-maks normalleştirme uygulayın.
NCA, mesafeye dayalı bir özellik seçicidir. NCA'yı etkin bir şekilde uygulamak için normalizasyon yapılmalıdır.
03: end for i
04: $index = NCA(X, hedef)$; // NCA kullanarak özelliklerin indekslerini hesaplayın. Bu indeksler azalan şekilde sıralanmıştır.
05: for $k=0$ to 89 do // Yinelemeli NCA uygula
06: for $l=1$ to $1+k$ do
07: $f^{NCA}(:, l) = X(:, index(l))$; // Yinelemeli NCA özellik seçim süreci.
08: end for l
09: $hata(k) = C(f^{NCA})$; // Seçilen her bir özellik vektörünün hata oranlarını hesaplayın.

Tablo 4.3. (Devamı)

```
10: end for k
11: [minimum, index] = min(hata); // Minimum hatayı ve indeksini bulun.
12: for j=1 to index do // Optimum özellikleri seçin.
13:     son(:,j) = X(:,index(j));
14: end for j
```

4.2. Bulgular

Sunulan VMD ve INCA tabanlı model, MATLAB (2020a) programlama ortamı kullanılarak gerçekleştirilmiştir. Çalışmada elde edilen SPA-IDS veri seti iki şekilde etiketlenmiştir. SPA-IDS veri setinin ilk halinde 9 saldırı sınıfı ve 1 normal ağ trafiği sınıfı bulunmaktadır. Önerilen yöntem SPA-IDS veri setinin ilk hali olan 10 sınıflı haline uygulanmış ve alınan sonuçlar Tablo 4.5'te listelenmiştir. VMD ve istatistiksel özellikler (15 istatistiksel moment) özellik üretici olarak kullanılmış ve 90 özellik oluşturulmuştur. INCA modeli bu özelliklerden en ayırt edici 48 tanesini seçmiştir. INCA'nın bu model için öznelik seçim süreci Şekil 4.5'te gösterilmektedir.



Şekil 4.5. INCA sonrası seçilen optimum özellik sayısı

Bu çalışmada INCA algoritmasında, zaman karmaşıklığı düşük olduğu için hata değeri üretici olarak DT sınıflandırıcısı seçilmiştir. İlk durum (SPA-IDS 10 sınıflı durumu) için en uygun öznelik sayısı 48 olarak bulunmuştur. Daha sonra belirlenen bu 48 öznelik dört geleneksel sınıflandırıcı ile sınıflandırılmış ve sırasıyla %87.25, %95.98, %96.51 ve %97.74 doğruluk oranları elde edilmiştir. Sınıflandırma esnasında 10 kat çapraz doğrulama kullanılmıştır. Performans metrikleri olarak Doğruluk, Cohen Kappa (CK), Mathew Correlation Coefficient (MCC), F-Ölçüsü, Hassasiyet, Duyarlılık, ve Geometrik Ortalama (GO) kullanılmıştır. MCC ve CK performans metriklerinin denklemleri Denklem 4.20 ve Denklem 4.21'de verilmiştir [175].

Sunulan VMD ve INCA tabanlı otomatik STS yöntemini test etmek için iki durum kullanılmıştır. Kullanılan sınıflandırıcıların özellikleri Tablo 4.4'te verilmiştir.

$$CK = \frac{2(DP.DN-YP.YN)}{(DP+YP).(YP+DN)+(DP+YN).(YN+DN)} \quad (4.20)$$

$$MCC = \frac{DP.DN-YP.YN}{\sqrt{(DP+YP).(DP+YN).(DN+YP).(YN+DN)}} \quad (4.21)$$

Tablo 4.4. Kullanılan KNN, SVM, DT ve BT sınıflandırıcılarının özellikleri

Sınıflandırıcı	Ayarlar	Değerler
KNN	K Değeri	1
	Mesafe Metriği	Spearman
	Mesafe Ağırlığı	Equal
DVM	Çekirdek	Gaussian
	C Değeri	1
	Çekirdek Modu	Manual
	Çekirdek Ölçeği	6.9
	Kodlama	One-Vs-One
DT	Bölünmüş Sayı	100
	Bölme Kriteri	Gini
BT	Topluluk Modu	Bag
	Öğrenici	DT
	Bölünmüş Sayı	1500
	Öğrenici Numarası	30
	Öğrenme Oranı	0.1
	Alt Uzay Boyutu	1

VMD ve INCA adımlarından sonra seçilen özellikler, KNN, DVM, DT ve BT sınıflandırıcıları kullanılarak sınıflandırılmıştır. 10 sınıflı SPA-IDS veri seti için alınan başarımlar Tablo 4.5'te verilmiştir.

Tablo 4.5. Önerilen yöntemin 10 sınıflı SPA-IDS veri seti ile performansı

Performans metrikleri	KNN	DVM	DT	BT
Doğruluk (%)	87,25	95.98	96.51	97.74
CK	85.82	95.53	96.12	97.49
MCC	85.71	95.47	96.10	97.47
F-Ölçüsü	87.12	95.91	96.48	97.71
Hassasiyet	87.23	95.94	96.49	97.73
Duyarlılık	87.04	95.91	96.48	97.71
Geometrik Ortalama	85.91	95.82	96.41	97.68

Hesaplanan sonuçların karmaşıklık matrisleri de Şekil 4.6'da listelenmiştir.

	1	2	3	4	5	6	7	8	9	10
1	780	0	12	0	0	0	6	32	0	53
2	0	922	0	0	0	0	5	4	2	0
3	6	0	833	0	0	0	0	14	0	30
4	0	0	0	758	20	40	0	0	0	0
5	0	0	0	7	553	314	0	0	0	0
6	0	0	0	18	314	508	0	0	0	0
7	13	23	0	0	0	0	787	14	30	1
8	31	2	21	0	0	0	15	803	0	9
9	0	5	0	0	0	0	6	1	849	0
10	36	0	38	0	0	0	2	4	0	924
Tahmin edilen değerler										

a) KNN karmaşıklık matrisi

	1	2	3	4	5	6	7	8	9	10	
Gerçek değerler	1	871	0	0	1	0	0	1	3	0	7
	2	0	928	0	0	0	0	4	1	0	0
	3	0	0	854	4	0	0	0	5	0	20
	4	0	0	0	803	10	5	0	0	0	0
	5	0	0	0	7	788	79	0	0	0	0
	6	0	0	0	8	106	726	0	0	0	0
	7	0	1	0	4	0	0	854	5	4	0
	8	15	0	1	2	0	0	18	842	0	3
	9	0	0	0	0	0	0	17	0	844	0
	10	3	0	12	3	0	0	6	1	0	979
Tahmin edilen değerler											

b) SVM karmaşıklık matrisi

	1	2	3	4	5	6	7	8	9	10
Gerçek değerler	1	861	0	2	0	0	0	11	0	9
2	0	929	0	0	0	0	4	0	0	0
3	1	0	858	0	0	0	0	5	0	19
4	0	0	0	811	4	3	0	0	0	0
5	0	0	0	2	782	90	0	0	0	0
6	0	0	0	4	79	757	0	0	0	0
7	0	2	0	0	0	0	851	5	2	0
8	5	0	7	0	0	0	8	842	0	6
9	0	1	0	0	0	0	1	0	844	0
10	4	0	21	0	0	0	4	2	0	973
	Tahmin edilen değerler									

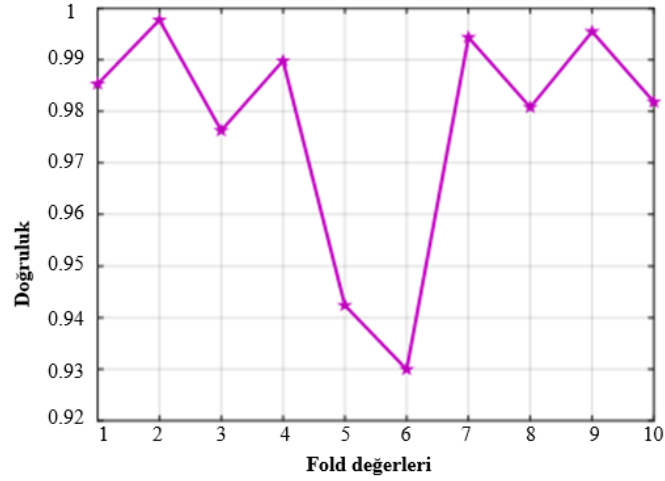
c) DT karmaşıklık matrisi

	1	2	3	4	5	6	7	8	9	10	
Gerçek değerler	1	870	0	2	0	0	0	0	4	0	9
	2	0	930	0	0	0	0	3	0	0	0
	3	1	0	863	0	0	0	0	3	0	17
	4	0	0	0	811	2	4	0	0	0	0
	5	0	0	0	4	822	48	0	0	0	0
	6	0	0	0	8	55	777	0	0	0	0
	7	1	1	0	0	0	0	865	1	0	0
	8	3	0	0	0	0	0	13	862	0	3
	9	0	1	0	0	0	0	1	1	858	0
	10	2	0	11	0	0	0	3	1	0	987
Tahmin edilen değerler											

d) BT karmaşıklık matrisi

Şekil 4.6. 10 sınıflı SPA-IDS veri seti için sınıflandırıcıların karmaşıklık matrisleri

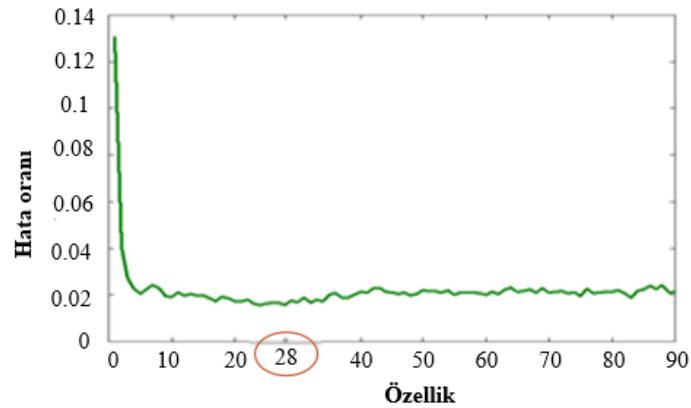
Şekil 4.6’da kullanılan sınıflar sırasıyla FTP, RDP ve SSH kaba kuvvet saldırıları, DHCP starvation, ICMP ve SYN hizmet reddi saldırıları, TCP ve UDP zafiyet taramaları, Mac adres taşma saldırısı ve Normal trafik olarak sıralanmıştır. Şekil 4.6’dan ve Tablo 4.5’ten de görülebileceği gibi, en iyi sonuç veren sınıflandırıcı BT’dir ve on kat çapraz doğrulama ile %97.74 doğruluk elde etmiştir. BT’nin on kat çapraz doğrulama sonuçları Şekil 4.7’de verilmiştir.



Şekil 4.7. 10 sınıflı SPA-IDS veri seti için BT sınıflandırıcısının çapraz doğrulama grafiği

SPA-IDS veri setinin ikinci halinde; FTP, RDP ve SSH kaba kuvvet saldırı senaryoları “Kaba Kuvvet” saldırısı olarak birleştirilirken, ICMP ve SYN hizmet reddi saldırıları “Hizmet Reddi” saldırısı, TCP ve UDP taramaları da “Zafiyet Taraması” olarak ana kategoriler halinde birleştirilmiştir. Saldırıların ana kategorileri Tablo 4.1’de verilmiştir. Daha sonra önerilen VMD ve INCA yöntemleri kullanılarak veri seti yeniden sınıflandırılmıştır. Hesaplanan sonuçlar Tablo 4.6’da listelenmiştir.

İkinci durumda (SPA-IDS 6 sınıflı durumu), INCA kullanılarak en uygun öznitelik vektörünün uzunluğu 28 olarak hesaplanmıştır. INCA’nın bu model için öznitelik seçim süreci Şekil 4.8’de gösterilmektedir.



Şekil 4.8. Ana saldırı kategorileri için özellik seçimi

SPA-IDS veri setinin ikinci hali olan 6 sınıflı hali için de INCA’nın karmaşıklık üretici olarak yine DT kullanılmış ve en uygun öznitelik sayısı 28 olarak bulunmuştur. Daha sonra bu 28 özniteliğe dört adet geleneksel sınıflandırıcı uygulanmış ve sırasıyla %96.65, %98.52, %98.39 ve %99.11 doğruluk oranları elde edilmiştir. Alınan sonuçlar Tablo 4.6’da verilmiştir.

Tablo 4.6. Önerilen yöntemin 6 sınıflı SPA-IDS veri seti ile performansı

Performans metrikleri	KNN	SVM	DT	BT
Doğruluk (%)	96.65	98.52	98.39	99.11
CK	95.81	98.15	97.99	98.88
MCC	96.03	98.27	98.11	98.86
F-Ölçüsü	96.73	98.58	98.45	99.05
Hassasiyet	96.76	98.59	98.48	99.05
Duyarlılık	96.72	98.58	98.42	99.05
Geometrik Ortalama	96.70	98.58	98.41	99.05

SPA-IDS veri setinin 6 sınıflı hali için elde edilen karmaşıklık matrisleri Şekil 4.9’da gösterilmektedir. Kaba kuvvet, DHCP starvation, hizmet reddi, Zafiyet taraması, mac adres taşması ve normal trafik sınıfları sırasıyla 1’den 6’ya kadar numaralandırılmıştır.

Gerçek sonuçlar		1	2	3	4	5	6
	1	2620	0	0	39	3	37
	2	0	800	18	0	0	0
	3	0	10	1704	0	0	0
	4	78	0	0	1629	26	16
	5	2	0	0	7	847	5
	6	46	0	0	4	5	949
Tahmin edilen sonuçlar							

a) KNN sınıflandırıcısı karmaşıklık matrisi

Gerçek sonuçlar		1	2	3	4	5	6
	1	2649	0	0	27	0	23
	2	0	806	12	0	0	0
	3	0	5	1709	0	0	0
	4	23	0	0	1714	7	5
	5	0	0	0	1	860	0
	6	24	0	0	4	0	976
Tahmin edilen sonuçlar							

b) SVM sınıflandırıcısı karmaşıklık matrisi

Gerçek sonuçlar		1	2	3	4	5	6
	1	2647	0	0	22	0	30
	2	0	809	9	0	0	0
	3	0	7	1707	0	0	0
	4	24	0	0	1717	3	5
	5	0	0	0	1	860	0
	6	34	0	0	7	0	963
Tahmin edilen sonuçlar							

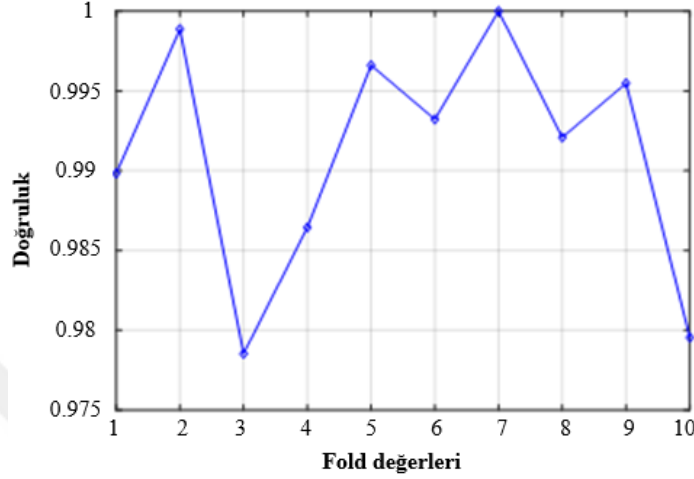
c) DT sınıflandırıcısı karmaşıklık matrisi

Gerçek sonuçlar		1	2	3	4	5	6
	1	2670	0	0	6	0	23
	2	0	806	12	0	0	0
	3	0	7	1707	0	0	0
	4	6	0	0	1740	2	1
	5	0	0	0	3	858	0
	6	14	0	0	5	0	985
Tahmin edilen sonuçlar							

d) BT sınıflandırıcısı karmaşıklık matrisi

Şekil 4.9. 6 sınıflı SPA-IDS veri seti için sınıflandırıcıların karmaşıklık matrisleri

Şekilden de görülebileceği gibi, en iyi sonuç veren sınıflandırıcı BT'dir ve on kat çapraz doğrulama ile %99.11 doğruluk elde etmiştir. Bu nedenle, BT'nin çapraz doğrulama sonuçları Şekil 4.10'da gösterilmektedir.



Şekil 4.10. 6 sınıflı SPA-IDS veri seti için BT sınıflandırıcısının çapraz doğrulama grafiği

4.3. Sonuçlar

Anahtarlama cihazları kullanıcıların yerel ağlarda ilk bağlandığı cihazlardır. Her saldırı türünün anahtarlama cihazı portları üzerinde farklı bir etkisi vardır. Bu çalışmada yerel ağlarda sıklıkla kullanılan 9 saldırı senaryosu kullanılmıştır. Her saldırı için anahtarlama cihazı portlarındaki anomali paket numaralarının değişimi alınmış ve SPA-IDS veri seti oluşturulmuştur. Veri seti, gerçek internet trafiğine sahip bir ağ üzerinde oluşturulmuştur. Anahtarlama cihazlarının portlarında meydana gelen paket değişikliğini almak, saldırının özelliklerini belirlemek için büyük avantajlar sağlar. Bu sayede ağ yöneticileri, ağlarındaki anahtarları izleyebilecek ve bağlantı noktalarındaki kötü amaçlı etkinlikleri çok hızlı bir şekilde tespit edebilecektir. Hafif bir model kullanarak paket trafiğini otomatik olarak sınıflandırabilmek için VMD ve INCA özellik seçimine dayalı bir yöntem önerilmiştir. Daha sonra veri seti KNN, SVM, DT ve BT sınıflandırıcıları ile sınıflandırılmıştır. 10 sınıflı SPA-IDS veri seti için sırasıyla %87.25, %95.98, %96.51 ve %97.74 doğruluk oranları elde edilmiştir. Önerilen yöntem 6 sınıflı SPA-IDS veri setine de uygulanmış ve sırasıyla %96.65, %98.52, %98.39 ve %99.11 başarımlarına ulaşılmıştır. Her iki senaryo için de en iyi performans BT sınıflandırıcı ile elde edilmiştir. Önerilen yöntem ile kritik ağ altyapılarında hizmet kesintilerinin önüne geçilmesi amaçlanmaktadır. Ayrıca önerilen yöntem, ağ yöneticileri için geniş bir ağ görünürlüğü sunmaktadır. Böylece kötü amaçlı ağ olaylarına tepki süresi en aza indirilir.

5. İNTERNET KULLANICILARININ DAVRANIŞ ANALİZİ

Kullanıcıların internet davranışlarının analizi internetin yaygın kullanımı ile birlikte önem kazanan bir diğer konudur. Kullanıcıların internetteki davranışları incelenerek ticaret, sağlık, ekonomi, eğlence gibi birçok alanda kullanıcıya özel öneriler çevrimiçi olarak sunulabilmektedir. Tez çalışmasının bu bölümünde kullanıcıların internetteki davranışlarını analiz edebilmek için otomatikleştirilmiş doğru bir sınıflandırma modeli ve internet of behavior (IoBe) veri seti önerilmiştir. Çalışmada önerilen IoBe veri setini oluşturmak için internette en sık kullanılan uygulama ve web sitelerinin trafikleri ele alınmıştır. Oluşturulan IoBe veri setini başarılı bir şekilde sınıflandırabilmek için yeni bir model geliştirilmiştir. Geliştirilen model havuzlama fonksiyonları kullanarak öznitelik artırma, ReliefF tabanlı öznitelik seçimi, sınıflandırma ve çoğunluk oylaması olmak üzere üç aşamadan oluşmaktadır. Öznitelikler, çalışmada önerilen koşullu havuzlama (conditional pooling - CP), ortalama havuzlama (average pooling - AP), maksimum havuzlama (maximum pooling - MP) ve maksimum mutlak havuzlama (maximum absolute pooling - MAP) havuzlama metodlarına dağıtılarak artırılmıştır. Geliştirilen model, davranışları tahmin etmek için toplanan IoBe veri seti ve CICDarknet2020 veri seti üzerinde test edilmiştir. Sunulan havuzlama tabanlı özellik artırma ve ReliefF seçici tabanlı modelin temel katkıları şunlardır:

- İnternet kullanıcılarının davranışlarını otomatik olarak sınıflandırmak için yeni ve orijinal bir veri seti oluşturulmuştur. Oluşturulan veri setinin, yapay zeka ve internet verilerinin analizi çalışmalarına katkı sağlaması için paylaşıma açılmıştır [176].
- Otomatik bir IoBe sınıflandırma modeli sunulmuştur. Önerilen model etkili yöntemleri bir arada kullanan oldukça basit ve performanslı bir modeldir. Havuzlama yöntemlerini kullanarak özellikleri artıran bu model, ReliefF ile en ayırt edici özellikleri seçmiş, yinelemeli BT kullanarak sınıflandırma yapılmış ve son aşamada sınıflandırıcı çıkışındaki veriler çoğunluk oylamasına tabii tutulmuştur. Önerilen model ile siber güvenlik ve kullanıcıların takibi için akıllı bir bilgi güvenliği modeli sunulmuştur.

5.1. Materyal ve Metod

İnternetin yaygınlaşmasıyla birlikte kullanıcıların davranış analizi önemli bir konu haline gelmiştir. Bu çalışmada, bir internet kullanıcısı internet trafiğini kullanarak davranış analizi yapılmaya çalışılmıştır. Çalışma için oluşturulan IoBe ve CICDarknet 2020 veri setleri kullanılmıştır.

5.1.1. IoBe Veri Seti

Veri seti hazırlanırken bilgisayarda normal bir internet kullanıcısının davranışları sergilenmiştir. Kullanıcıların internet trafiğini yansıtmak amacıyla 20 farklı internet uygulaması kullanılmış ve önerilen IoBe veri seti oluşturulmuştur. Uygulamaların her biri ayrı ayrı çalıştırılmış ve bu esnada wireshark ile uygulama trafiği alınmıştır. Elde edilen wireshark dosyaları, bir ağ analiz yazılımı olan CICFlowmeter aracıyla özelliklerine ayrıştırılmıştır. CICFlowmeter, java dilinde yazılmıştır ve çevrimiçi ağ trafiğinden veya çevrimdışı pcap dosyalarından Biflow'lar oluşturarak ağ akışlarından özellikler çıkarır. Uygulama ağ trafiğini 80 özelliğe ayrıştırır ve bu çıktıları CSV formatına dönüştürür. Bu yazılım çoğunlukla ağdaki saldırıları tespit etmek için üretilmiştir [56,177]. Ancak bu çalışmada internet kullanıcısının davranış analizi için kullanılmıştır. Tablo 5.1'de çalışmada kullanılan uygulamalar, uygulamaların ana kategorileri ve alınan veri miktarları belirtilmiştir. Her uygulamada çok fazla veri elde edildi. Ancak önerilen IoBe veri seti, her uygulama için ortalama 300 veri alınarak oluşturulmuştur.

Tablo 5.1. Önerilen IoBe veri seti özellikleri

Kategori	Uygulama	Veri miktarı	Web sitesi / uygulama adresi
Haber	Haber	301	https://edition.cnn.com/
Video	YouTube	306	https://www.youtube.com/
	BluTV	303	https://www.blutv.com/
	Udemy	304	https://www.udemy.com/
E-ticaret	Alibaba	309	https://www.alibaba.com/
	Amazon	307	https://www.amazon.com.tr/
	Ebay	305	https://www.ebay.com/
Uzak Masaüstü	Teamviewer	312	Teamviewer uygulaması
	Anydesk	310	Anydesk uygulaması
Akademik Araştırmalar	Science Direct	308	https://www.sciencedirect.com/
	Elsevier	310	https://www.elsevier.com/tr-tr
	IEEEExplore	311	https://ieeexplore.ieee.org/Xplore/home.jsp
Sosyal Medya	Facebook	309	https://tr-tr.facebook.com/
	Twitter	309	https://twitter.com/?lang=tr
	Instagram	309	https://www.instagram.com/
	Linkedin	309	https://tr.linkedin.com/
Forumlar	Eksi Sözlük	307	https://eksisozluk.com/
	Memurlar	308	https://www.memurlar.net/
Çevrimiçi Toplantılar	Zoom	314	Zoom uygulaması
	Google Meet	314	https://apps.google.com/meet/

5.1.2. CICDarknet 2020 Veri Seti

Çalışmada oluşturulan IoBe veri setinin performansını karşılaştırmak amacıyla literatürde kullanılan ve 2020 yılında oluşturulan CICDarknet veri seti kullanılmıştır. CICDarknet 2020 veri seti, gerçek zamanlı uygulama trafiğini sınıflandırmak için geliştirilmiştir. Darknet

trafiğinin analizi ile kötü niyetli internet trafiğinin tespit edilmesi amaçlanmaktadır. Veri setindeki uygulama trafiği hem normal ağ trafiği hem de Darknet trafiği kullanılarak elde edilmiştir [178,179]. Tablo 5.2’de hem Darknet trafik kategorileri hem de ağ trafiği kategorileri verilmiştir. Ayrıca bu çalışmada her bir uygulama kategorisinden alınan veri miktarları da Tablo 5.2’de verilmiştir.

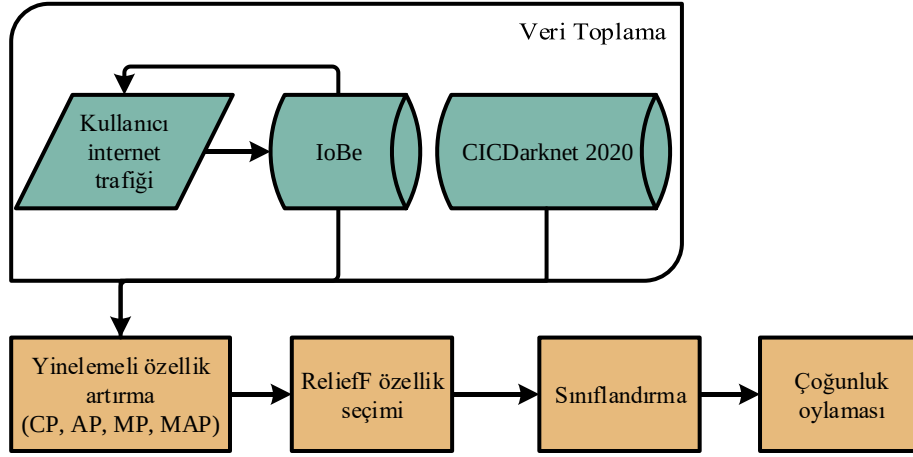
Tablo 5.2. CICDarknet 2020 veri seti trafik sınıfları

Kategori	Veri miktarı	Uygulamalar
Audio-Stream	1025	Vimeo
		Youtube
Transfer	1019	Skype
		SFTP
		FTPS
VOIP	873	Facebook
		Skype
		Hangouts
Browsing	1015	Firefox
		Chrome
E-mail	492	SMTPS
		POP3S
		IMAPS
Chat	343	Facebook
		ICQ
		AIM
		Skype
		Hangouts
P2P	1009	UTorrent
		BitTorrent
Video Stream	995	Vimeo
		Youtube

Bu çalışmada önerilen IoBe veri seti normal ağ trafiği kullanılarak oluşturulmuştur. Bu nedenle CICDarknet 2020 çalışmasından normal trafik sınıfları alınmıştır. Böylece benzer şekilde oluşturulan veri setleri karşılaştırılmıştır.

5.1.3. Önerilen Metod

Önerilen yöntemde, IoBe ve CICDarknet 2020 veri setleri öznitelik seçimi ve sınıflandırma adımlarından geçirilmiştir. Önerilen modelin grafiksel özeti Şekil 5.1’de verilmiştir.

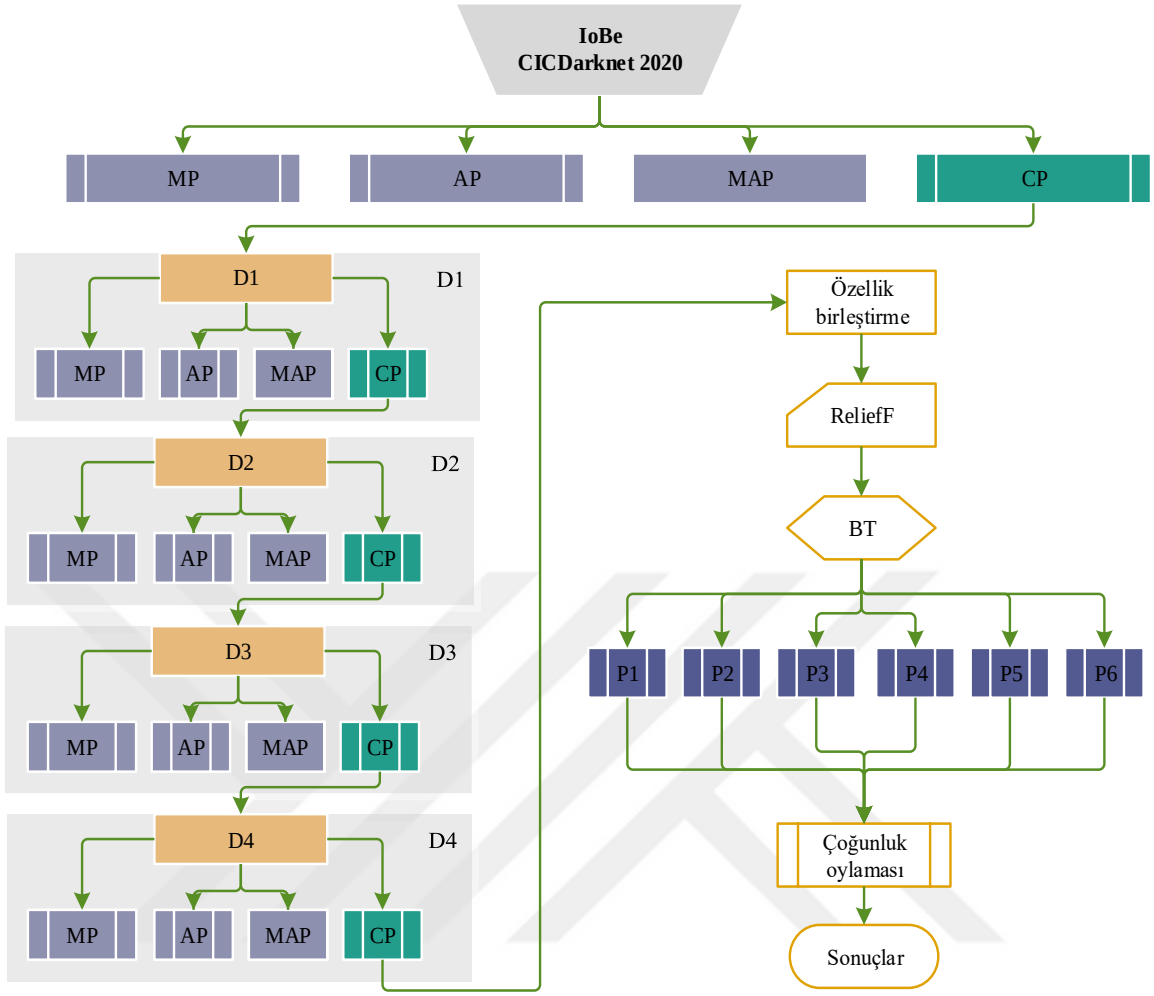


Şekil 5.1. Önerilen yöntemin akış diyagramı

Bu çalışmada, yeni nesil IoBe sınıflandırma modeli önerilmiştir. Önerilen model dört ana bölümden oluşmaktadır ve bu bölümler şunlardır:

- Özelliklerin havuzlama yöntemleriyle yinelemeli (iteratif) olarak artırılması
- ReliefF özellik seçimi ile en uygun özelliklerin seçilmesi
- BT'ye göre ayarlanmış yinelemeli parametrelerle sınıflandırma
- Ağırlıksız çoğunluk oylaması.

Çalışmada önerilen IoBe veri seti 8 kategori ile 6133 gözlemin 79 özellikten oluşmaktadır. İlk aşamada, CP, AP, MP, MAP olmak üzere dört adet havuzlama işlevi kullanılarak özellikler artırılmıştır. Bu havuzlama işlevleri, özellikleri artırmak için dört kez tekrarlanmış ve 363 adet özellik oluşturulmuştur. Özellik seçici olarak kullanılan ReliefF algoritması ile 363 özelliğin 50'si en en uygun özellik olarak seçilmiştir. Seçilen özellikler, altı bölünmüş kriter parametresi kullanılarak BT sınıflandırıcısına sokulmuştur. Böylece altı doğrulama tahmin vektörü hesaplanmıştır. Son aşamada, sunulan modelin genel sonucunu hesaplamak için oluşturulan altı sonuca ağırlıksız çoğunluk oylaması (mod fonksiyonu tabanlı) uygulanmıştır. Tanıtılan yinelemeli özellik artırma ve ReliefF tabanlı modelin grafiksel açıklaması Şekil 5.2'de gösterilmektedir. Ayrıca IoBe veri seti için uygulanan tüm işlemler CICDarknet 2020 veri seti için de uygulanmıştır.



Şekil 5.2. Önerilen sınıflandırma modelinin akış diyagramı.

Şekil 5.2’ deki, D1, D2, D3 ve D4 ayrıştırılmış verileri gösterirken P1, P2, ... P6, BT sınıflandırıcısı için tahmin vektörleridir. Sunulan modelin daha fazla detayı adımlar halinde verilmiştir.

Özellik Artırımı: Sunulan yinelemeli özellik artırma modeli, dört havuzlama modelini kullanır. Bu havuzlama yöntemleri CP, MP, AP ve MAP olarak adlandırılır. Çoklu havuza dayalı yinelemeli özellik artırma adımları aşağıda verilmiştir.

- Özellikler, önerilen sisteme yüklenmiştir.
- Veriler, ilk özellikler olarak atanmıştır.
- Özellikleri çıkarmak için CP, MP, AP ve MAP havuzlama metodları kullanılmıştır. Burada, ayrıştırma için iki boyutlu örtüşmeyen bloklar kullanılır. Bu havuzlama fonksiyonlarının tanımları aşağıda verilmiştir.

CP, koşullu havuzlamadır. CP'nin temel amacı minimum, maksimum ve ortalama havuzlamayı birlikte kullanmaktır. Bu nedenle, önerilen CP'de üç koşul kullanılır. CP'nin matematiksel gösterimi Denklem 5.1'de verilmiştir.

$$CP(in_1, in_2) = \begin{cases} \max(in_1, in_2), in_1 \geq 0 \text{ and } in_2 \geq 0 \\ \min(in_1, in_2), in_1 < 0 \text{ and } in_2 < 0 \\ \text{mean}(in_1, in_2), in_1 * in_2 < 0 \end{cases} \quad (5.1)$$

Burada in_1, in_2 CP'nin girdileridir. $\max(\cdot)$, $\min(\cdot)$ ve $\text{mean}(\cdot)$ maksimum, minimum ve ortalama hesaplama fonksiyonlarıdır. MP ve AP yöntemleri derin öğrenme modellerinde yaygın olarak kullanılan ayrıştırıcılarıdır. Matematiksel tanımları Denklem 5.2 ve Denklem 5.3'te verilmiştir.

$$MP(in_1, in_2) = \begin{cases} in_1, in_1 \geq in_2 \\ in_2, in_2 > in_1 \end{cases} \quad (5.2)$$

$$AP(in_1, in_2) = \frac{in_1 + in_2}{2} \quad (5.3)$$

Denklem 5.2 ve Denklem 5.3'te görüldüğü gibi, bu havuzlama yöntemleri yalnızca tepe ve ortalama değerleri yönlendirir. MAP, koşul başına hem maksimum hem de minimum değerleri yönlendirir ve Denklem 5.4'te tanımlanmıştır.

$$MAP(in_1, in_2) = \begin{cases} in_1, |in_1| \geq |in_2| \\ in_2, |in_2| > |in_1| \end{cases} \quad (5.4)$$

- CP, MP, AP ve MAP kullanarak ayrıştırılmış özellikler Denklem 5.5, Denklem 5.6, Denklem 5.7 ve Denklem 5.8'deki gibi hesaplanmıştır.

$$d_1^k = CP(veri), k \in \{1,2,3,4\} \quad (5.5)$$

$$d_2^k = MP(veri) \quad (5.6)$$

$$d_3^k = AP(veri) \quad (5.7)$$

$$d_4^k = MAP(veri) \quad (5.8)$$

Ayrıştırılmış özellikler hesaplandıktan sonra veriler Denklem 5.9'da gösterildiği gibi güncellenir.

$$veri = d_1^k \quad (5.9)$$

- Nihai özellikleri elde etmek için ayrıştırılmış özellikler ve özellik vektörü birleştirilmiştir.

Özellik Seçimi: Önerilen modelin ikinci aşaması özellik seçimidir. ReliefF seçici olarak kullanılmıştır. ReliefF, yaygın olarak tercih edilen özellik seçicilerden biridir ve ağırlıkları atamak için Manhattan mesafesini kullanır. ReliefF, Relief algoritmasının güçlendirilmiş bir uzantısıdır ve Relief algoritmasındaki gürültülü veriler ve eksik veriler gibi eksiklikleri çözmek için kullanılmaktadır. ReliefF algoritması, hem pozitif hem de negatif ağırlıklı özellikleri çıkarabilir. Negatif ağırlıklı özellikler, gereksiz olanları ifade eder ve daha büyük ağırlıklar, özelliklerin önemli derecesini gösterir [180–183]. Önerilen yöntemde artırılmış özelliklere ReliefF algoritması uygulanmış ve bunun sonucunda en iyi 50 özellik seçilmiştir.

Sınıflandırma: Torbalı ağaç sınıflandırıcısı, sınıflandırma sonuçlarını hesaplamak için seçilmiştir. Bu sınıflandırıcıyı seçmek için MATLAB classification learner toolbox (MCLT) kullanılmıştır. En iyi performans BT kullanılarak elde edildiğinden sınıflandırıcı olarak seçilmiştir. BT, bölme sayısı ve öğrenici sayısı olmak üzere iki ana parametreden oluşur. Çalışmada, bölme sayısı parametreleri yinelemeli olarak değiştirilmiştir. Bu parametre, altı değişken doğrulama tahmin vektörünü hesaplamak için yinelemeli olarak değiştirilmiştir. Ayrıca, doğrulama için on kat çapraz doğrulama kullanılmıştır. Maksimum bölünme sayısı 6000 olarak tanımlanmıştır. Bu nedenle Artma derecesi (Increasing degree - id), $\frac{6000}{6} = 1000$ olarak hesaplanır. Öğrenici sayısı 30 olarak kabul edilmiştir. Altı doğrulama tahminini yinelemeli olarak Denklem 5.10'daki gibi hesaplanır.

$$VP^j = BT(fseç, y, id * j, 30, 10), j \in \{1, 2, \dots, 6\} \quad (5.10)$$

Burada, VP^j j'inci doğrulama tahmin vektörünü, $fseç$ (Gözlem sayısı x 50) boyutunda seçilen özellikleri, y gerçek çıktı değerlerini, $id * j$ bölme sayısı parametresini tanımlar. Ayrıca çapraz doğrulama değeri 10 olarak seçilmiştir.

Çoğunluk Oylaması: Önerilen yöntemin son aşaması çoğunluk oylamasıdır ve çoğunluk oylamasının temel amacı, BT'nin sınıflandırma yeteneğini geliştirmektir. Bölüm 5.2.3'te görüldüğü gibi, altı doğrulama tahmini değeri hesaplanır. Bu doğrulama tahmin fonksiyonlarına mod fonksiyonu uygulanır ve genel bir doğrulama tahmin vektörü hesaplanır. Önerilen yöntemin son adımında oluşturulan, $VP^{j'}$ ye mod işlevini uygulanmış ve ortak bir VP hesaplanmıştır. Yapılan işlem Tablo 5.3'te açıklanmıştır.

Tablo 5.3. Mod işlevine dayalı çoğunluk oylaması

Giriş: Gözlem sayısı (Gs) uzunluğunda hesaplanan altı doğrulama tahmini (VP^j)
Çıkış: sonuçlar
00: VP^* yi yükle
01: for k=(1 – Gs) do
02: for j=1 to 6 do
03: $dizi(j) = VP^j(k)$;
04: end for i
05: $CVP(k) = mode(dizi)$;
// Ortak doğrulama tahmin vektörünü (CVP)sıralamak ve hesaplamak için mod fonksiyonunu uygulandı
06: end for k
07: Sonuçları CVP kullanarak hesapla

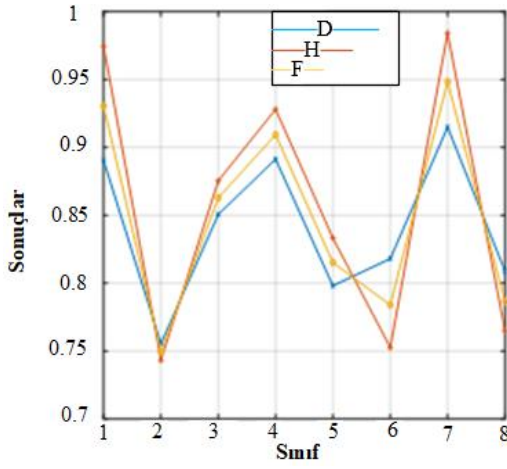
5.2. Bulgular

Bu bölüm, önerilen yinelemeli özellik artırma ve ReliefF tabanlı IoBe sınıflandırma modelinin sonuçlarını vermektedir. Yinelemeli özellik artırma, ReliefF tabanlı özellik seçimi ve çoğunluk oylama işlevleri m dosyaları kullanılarak kodlanmıştır. Sınıflandırma aşamasında BT sınıflandırıcısını çalıştırmak için MCLT kullanılmış ve BT'nin kaynak kodları MCLT kullanılarak oluşturulmuştur. Yinelemeli BT, bu kodlar değiştirilerek uygulanmıştır. Sonuçlar hesaplanırken 10 kat çapraz doğrulama kullanılmıştır. Bu nedenle, bu bölümde katlamalı ve genel sonuçlar sunulmuştur. Genel doğruluk (Overall accuracy - OA), genel ortalama kesinlik (overall average precision - OAP), ağırlıksız ortalama geri çağırma (unweighted average recall - UAR), Cohen's kappa (CK), geometrik ortalama (geometric mean - GM), genel F1 (overall F1 - OF1) ve Mathew korelasyon katsayısı (Mathew correlation coefficient - MCC) performans metrikleri kullanılmıştır. Önerilen IoBe sınıflandırma modelinin hesaplanan sonuçları Tablo 5.4'te verilmiştir.

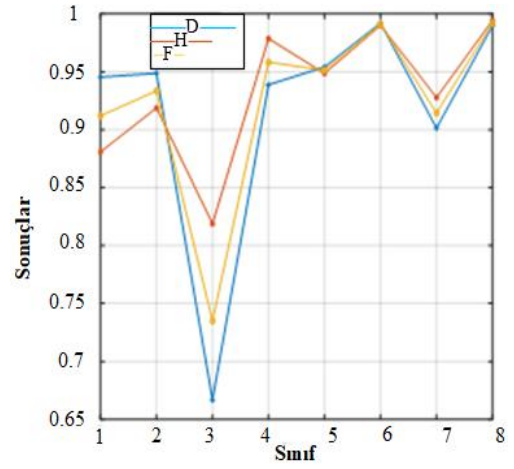
Tablo 5.4. Önerilen IoBe sınıflandırma modelinin hesaplanan sonuçları

Metrikler	IoBe veri seti sonuçları	CICDarkNet veri seti sonuçları
OA	83.01	93.90
OAP	85.70	93.23
UAR	84.10	91.71
CK	80.20	92.93
GM	83.94	91.09
OF1	84.82	92.35
MCC	82.36	91.54

Kullanılan veri seti sekiz sınıf içermektedir. Bu nedenle sınıf bazında duyarlılık, hassasiyet ve F-Ölçüsü değerleri hesaplanmış ve sonuçlar Şekil 5.3'te verilmiştir.



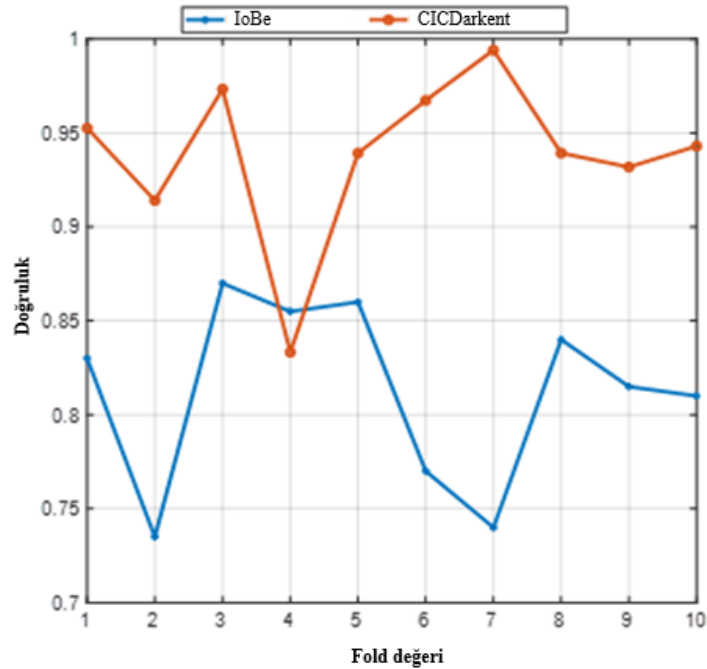
a) IoBE veri seti



b) CICDarknet veri seti

Şekil 5.3. Sınıf bazlı elde edilen sonuçlar

Şekil 5.3'te görüldüğü gibi en iyi sonuç veren sınıf yedinci sınıftır. Ayrıca, sonuçları on kat çapraz doğrulama kullanılarak elde edilmiş ve elde edilen sonuçlar Şekil 5.4'te verilmiştir.



Şekil 5.4. Önerilen modelin çapraz doğrulama sonuçları

Bu çalışmada veri setlerinin otomatik sınıflandırılması için, yeni bir model sunulmuştur. Bu model, temel ve etkili yöntemleri bir arada kullanır. Özellik artırmada dört havuzlama işlevi

kullanılmış ve 363 öznitelik elde edilmiştir. 363 özellikten 50'si ReliefF tarafından seçilmiştir. Yinelemeli BT, bölünmüş sayı parametresini değiştirerek uygulanmıştır. Altı bölünmüş sayı parametresi kullanılmış ve altı sonuç hesaplanmıştır. Bu modelin sınıflandırma yeteneğini geliştirmek için, bu sonuçlara ağırlıksız çoğunluk oylaması uygulanmıştır. Hesaplanan altı sonuç (çoğunluk oyu olmadan) ve ortak (çoğunluk oyu ile) Tablo 5.5'te listelenmiştir.

Tablo 5.5. BT'nin sonuçları

Sınıflandırıcı	IoBe veri seti			CICDarknet veri seti		
	OA	UAR	OF1	OA	UAR	OF1
BT (SN=1000)	82.52	83.73	84.40	93.37	90.98	91.60
BT (SN=2000)	82.39	83.55	84.19	93.38	91.11	91.70
BT (SN=3000)	82.41	83.56	84.16	93.40	91.14	91.73
BT (SN=4000)	82.55	83.70	84.44	93.38	91.27	91.86
BT (SN=5000)	82.60	83.71	84.45	93.36	91.18	91.76
BT (SN=6000)	82.54	83.63	84.31	93.36	91.45	91.96
Yinelemeli BT	83.01	84.10	84.82	93.90	91.71	92.35

Ayrıca bu sınıflandırıcıların karışıklık matrisleri Şekil 5.5'te verilmiştir.

		Tahmin edilen sonuçlar							
		1	2	3	4	5	6	7	8
Gerçek sonuçlar	1	268	5	6	1	7	11	1	2
	2	3	689	29	1	31	117	2	40
	3	0	35	780	3	24	57	3	15
	4	0	6	9	541	6	16	0	29
	5	1	61	21	7	739	70	0	27
	6	3	90	29	9	51	1007	3	39
	7	0	8	11	1	9	19	559	4
	8	0	33	6	20	20	41	0	508

		Tahmin edilen sonuçlar							
		1	2	3	4	5	6	7	8
Gerçek sonuçlar	1	954	5	23	4	8	2	12	1
	2	14	962	3	2	8	0	25	0
	3	69	3	226	1	14	6	18	2
	4	11	6	5	460	0	0	8	0
	5	15	17	10	0	978	0	4	1
	6	5	0	0	1	1	1001	0	1
	7	15	52	5	2	22	1	887	0
	8	0	2	4	0	0	1	2	863

a) IoBe veri seti iteratif BT karmaşıklık matrisi

b) CICDarknet veri seti iteratif BT karmaşıklık matrisi

Şekil 5.5. Yinelemeli BT kullanılarak hesaplanmış karmaşıklık matrisleri

Şekil 5.5 ve Tablo 5.5, yinelemeli BT ile sınıflandırma kapasitesini arttığı açıkça görülmektedir.

5.3. Sonuçlar

Günümüzde dijital hayat, gerçek hayatın vazgeçilmez bir unsuru haline gelmiştir. İnsanların internette geçirdikleri zaman, bu süre zarfında bıraktıkları ayak izleri çok önemli hale geldi.

Kullanıcıların internet üzerindeki davranışları çoğunlukla belirli kalıplarda ortaya çıkmaktadır. Bu çalışmada kullanıcıların internetteki davranışlarını analiz etmek için bir yöntem önerilmiştir. Bu amaçla IoBe ve CICDarknet 2020 veri setleri kullanılmıştır. Önerilen yöntemde en çok kullanılan uygulamalar ile yeni bir davranışsal veri seti oluşturulmuştur. Bu çalışmada, yeni bir IoBe veri seti ve IoBe sınıflandırma modeli sunulmuştur. Sunulan model, havuz tabanlı öznitelik artırma, ReliefF tarafından en bilgilendirici öznitelik seçimi, BT ile sınıflandırma ve çoğunluk oylamasından oluşmaktadır. Önerilen model, IoBe ve CICDarknet 2020 veri setleri üzerinde test edilmiştir. Önerilen yöntemden sonra veri setleri sırasıyla %83.01 ve %93.90 doğruluğa ulaşmıştır.



6. ÖZYİNELEMELİ ÖZELLİK ELEME VE ÇOK KATMANLI ALGILAYICI OPTİMİZASYONU İLE SAĞLIK SİSTEMLERİNDE OTOMATİK SALDIRI TESPİTİ

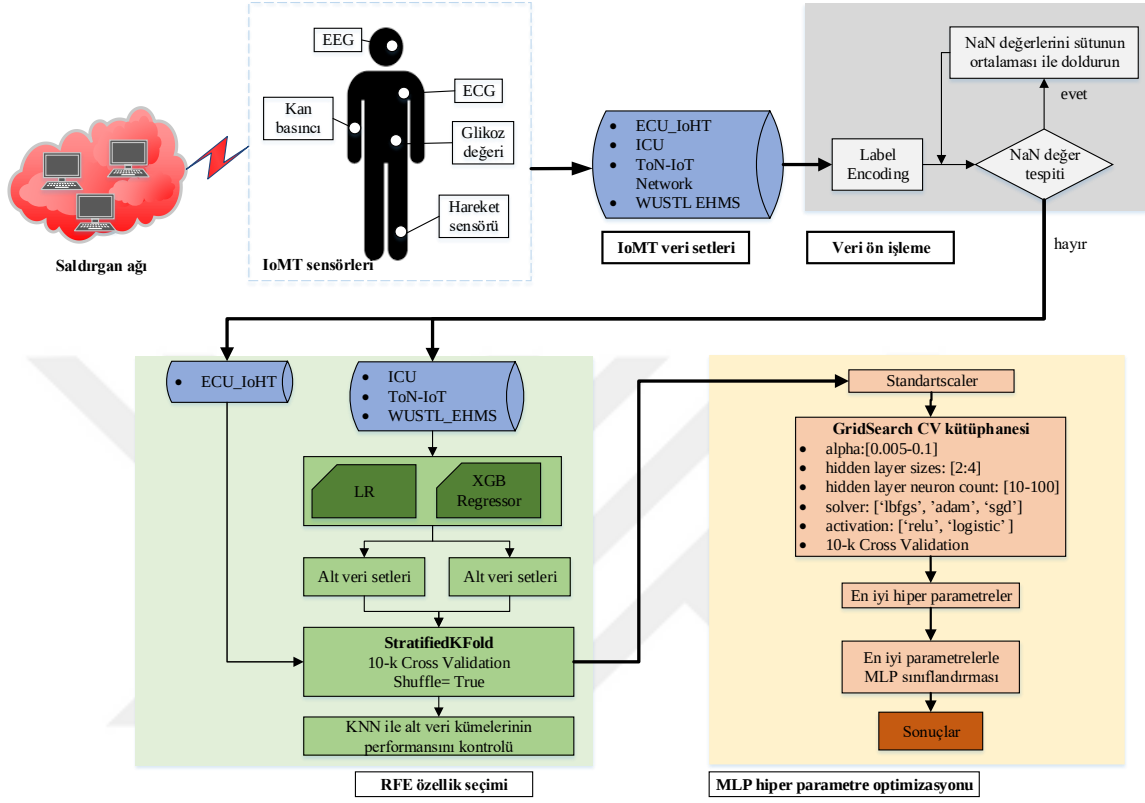
Nesnelerin interneti, temel altyapıdan günlük aktivitelere kadar hemen hemen her alanda kullanılmaktadır. İnsan sağlığı ile ilgili alanlarda da IoT teknolojisinin kullanımı giderek artmaktadır. Tıbbi nesnelerin interneti (IoMT) kavramı, IoT yeteneklerinin hasta izleme ve kritik bakım üniteleri gibi çok sayıda tıbbi disipline uygulanması ile doğmuştur. IoMT cihazlarının büyük bir kısmı internet saldırılarına dayanacak şekilde tasarlanmadığından güvenlikten ödün verme riskini artırmaktadır. Bu çalışmada, IoMT ağlarını siber saldırılara karşı korumak için RFE ve MLP tabanlı bir yöntem önerilmiştir. RFE yaklaşımında, LR ve XGBRegressor çekirdek fonksiyonları kullanılarak en uygun öznitelikler seçilmiştir. Sınıflandırma performansını iyileştirmek için, MLP sınıflandırıcısına hiper parametre optimizasyonu 10 kat çapraz doğrulama ile uygulanmıştır. Çalışmanın sunduğu katkılar aşağıdaki gibi özetlenmiştir.

- Literatürdeki IoMT çalışmalarına göre en fazla IoMT veri seti bu çalışmada ele alınmıştır. Bu da tıbbi alanlarda meydana gelebilecek saldırıların önlenmesinde geniş bir bakış açısı sağlamaktadır.
- Özellik seçimi için RFE sarmalayıcı (wrapper) metodu kullanılmıştır. Daha az bilgilendirici özellikler, yinelemeli olarak kaldırılmış ve en uygun özellikleri seçilmiştir. RFE algoritmasında sırasıyla LR ve XGBRegressor çekirdek işlevlerini kullanılmıştır. Böylelikle verilerin lojistik regresyona veya ağaç tabanlı yöntemlere uyumluluğu belirlenmiştir.
- Sınıflandırma için iki aşamalı doğrulama kullanılmıştır. İlk olarak, öznitelik seçim prosedürünün tutarlılığı, orijinal veri setleri ve RFE öznitelik seçimi ile oluşturulan alt veri setleri KNN sınıflandırıcısı kullanılarak değerlendirilmiştir.
- En uygun parametreler GridSearch CV kitaplığı kullanılarak belirlenmiş ve sınıflandırma performansı geliştirilmiştir. Hiper parametrelerin belirlenmesi, tanımlanması uzun zaman almasına rağmen, modelin performansını önemli ölçüde iyileştirmiştir. Modelin en büyük dezavantajı hiper parametreleri tanımlamak için geçen süredir, fakat bu parametreler sadece bir kez bulunacağı için önerilen model için bir sorun teşkil etmemektedir.

6.1. Materyal ve Metod

Anomali tespiti için kullanılan teknolojilerden bir olan STS sistemleri son zamanlarda medikal cihazlara yapılan saldırıların tespiti ve önlenmesinde sıklıkla kullanılmaktadır. Bu çalışmada sağlık alanında sıklıkla kullanılan IoMT cihazlarına yönelik saldırıların tespiti için yeni

bir yöntem önerilmiştir. Önerilen yöntem, veri ön işleme, özellik çıkarımı, sınıflandırma ve hiper parametre optimizasyonu adımlarından oluşmaktadır. Önerilen yönteme ait akış şeması Şekil 6.1’de verilmiştir.



Şekil 6.1. Önerilen yöntemin akış şeması

Veri ön işleme: Çalışmada önerilen IoMT veri setleri hem kategorik hem de numerik veriler içermektedir. Önerilen yöntemin en iyi performansta çalışabilmesi için verilerin tek bir formata getirilmesi gerekmektedir. Bu çalışmada kullanılan veri setlerindeki kategorik veriler Python Scikit-learn kütüphanesinde bulunan Label Encoder şifreleme metodu ile numerik değerlere çevrilmiştir. Veri setlerindeki veriler tek standarta çevrildikten sonra, veri setinde boş değer (NaN) kontrolü yapılmıştır. Böylelikle veri setlerinin herhangi bir yerinde bulunan NaN değerleri, bulundukları sütunun ortalaması ile değiştirilmiştir.

Özellik seçimi: Bu adımda, kodlanmış ve boş değerlerinden arındırılmış veri setlerinin en iyi özellikleri yinelemeli olarak seçilmiştir. Özellik seçimi esnasında Python Scikit-learn kütüphanesinde bulunan RFE algoritması kullanılmıştır. RFE özellik seçimi algoritmasında kullanılmak istenen özellik sayısı “n_features_to_select” parametresi ile belirlenebilir. Bu parametre varsayılan olarak “None” değerindedir ve bu şekilde verilen özelliklerden en uygun yarısını seçer. Bu çalışmada, tüm veri setlerinde “n_features_to_select” parametresine farklı değerler verilmiş fakat en uygun özelliklerin varsayılan “None” değerinde seçildiği test edilmiştir.

RFE algoritmasında seçilen özellikler 1 ile ağırlıklandırılır. Diğer özellikler önem derecesine göre 1’den farklı değerler ile ağırlıklandırılır. ECU-IoHT veri seti çok az özellikten oluştuğu için bu veri seti özellik seçim işlemine tabii tutulmamıştır. ICU, ToN IoT ve WUSTL-EHMS veri setleri için seçilen en uygun özellikler Tablo 6.1’de verilmiştir.

Tablo 6.1. IoMT veri setleri en uygun özellikler

ICU		ToN_IoT		WUSTL-EHMS	
LR	XGB	LR	XGB	LR	XGB
tcp.len	frame.time_delta	src_port	src_port	Sport	Flgs
tcp.ack	frame.time_relative	dst_port	dst_port	Dport	Sport
tcp.connection.rst	frame.len	service	proto	DstBytes	SrcLoad
tcp.flags.ack	tcp.srcport	duration	service	SrcLoad	DstLoad
tcp.flags.fin	tcp.dstport	src_bytes	duration	DstLoad	DIntPkt
tcp.flags.push	tcp.flags	dst_bytes	src_bytes	SrcGap	SrcJitter
tcp.flags.reset	tcp.time_delta	conn_state	dst_bytes	SIntPkt	DstJitter
tcp.hdr_len	tcp.len	missed_bytes	conn_state	DIntPkt	Dur
tcp.pdu.size	tcp.ack	src_pkts	missed_bytes	SIntPktAct	Load
tcp.window_size_value	tcp.connection.fin	src_ip_bytes	src_pkts	dMaxPktSz	pLoss
tcp.checksum	tcp.hdr_len	dst_pkts	src_ip_bytes	sMinPktSz	Packet_num
mqtt.clientid_len	tcp.pdu.size	dst_ip_bytes	dst_pkts	dMinPktSz	Temp
mqtt.conack.flags	tcp.checksum	dns_qclass	dst_ip_bytes	Dur	SpO2
mqtt.hdrflags	mqtt.clientid_len	dns_AA	dns_qclass	Trans	Pulse_Rate
mqtt.kalive	mqtt.conack.flags	dns_rejected	dns_qtype	Load	SYS
mqtt.len	mqtt.hdrflags	ssl_cipher	dns_rcode	pLoss	DIA
mqtt.msgtype	mqtt.kalive	http_uri	dns_AA	pSrcLoss	Heart_rate
mqtt.qos	mqtt.msgtype	http_version	dns_rejected	SYS	Resp_Rate
mqtt.topic	mqtt.qos	http_user_agent	http_method	Heart_rate	ST
mqtt.topic_len	mqtt.retain	weird_addl	type		
ip.proto	mqtt.topic				
ip.ttl	mqtt.topic_len				
class	class				

Sınıflandırma: Hem orijinal çalışma veri setleri hem de LR veya XGBRegressor kullanılarak optimal özellik seçiminden sonra elde edilen alt veri setleri, 10 katlı çapraz doğrulama kullanılarak eğitim ve test verilerine ayrıştırılmıştır. Python Scikit-Learn StratifiedKFold kütüphanesi, 10 katlı çapraz doğrulama işleminin her yinelemesinde ilgili veri setlerinin farklı alt setlerini seçmek için karıştırma parametresi (Shuffle), “Doğru (True)” olarak ayarlanmıştır. Test ve eğitim verileri ayrıştırıldıktan sonra KNN sınıflandırıcısı kullanılarak performans testi

yapılmıştır. Orijinal veri setlerinin ve alt veri setlerinin sınıflandırma sonuçları Tablo 6.2’de özetlenmiştir. Genel olarak, optimal öznitelik seçimine sahip alt veri setleri daha yüksek doğruluk oranlarına ulaşmıştır.

Tablo 6.2. KNN sınıflandırıcısı performans sonuçları

Veri setleri	Doğruluk(%)
ICU orijinal veri seti	97.419
LR özellik seçiminden sonra ICU alt veri seti	99.528
XGB özellik seçiminden sonra ICU alt veri seti	97.417
ToN_IoT orijinal veri seti	95.496
LR özellik seçiminden sonra TON-IoT alt veri seti	95.513
XGB özellik seçiminden sonra TON-IoT alt veri seti	95.518
WUSTL-EHMS orijinal veri seti	89.826
LR özellik seçiminden sonra WUSTL-EHMS alt veri seti	91.215
XGB özellik seçiminden sonra WUSTL-EHMS alt veri seti	89.846

Tablo 6.2’den de görüleceği gibi en önemli özellikleri seçilen alt veri setleri KNN sınıflandırıcısı sonucunda daha iyi doğruluk değerlerine ulaşmıştır. Önerilen bu adım özellik seçimini performansı olumlu yönde etkileyip etkilemediği test edilmiştir. Önerilen yöntemin bir sonraki adımında, sınıflandırma performansının üst seviyelere çıkarılabilmesi için seçilen en uygun özellikler MLP algoritması ile sınıflandırılmıştır.

Hiper parametre Optimizasyonu ile MLP Sınıflandırması: KNN sınıflandırma adımında gözlemlenen özellik seçiminin olumlu etkilerine dayanarak, genel sınıflandırma performansını artırmak için son adımda MLP sınıflandırıcısına hiper parametre optimizasyonu uygulanmıştır. 10-kat çapraz doğrulama içeren hiper parametre optimizasyon işlemi sonrasında seçilen hiper parametreler ve değerleri Tablo 6.3’te listelenmiştir. Veri setleri en iyi performansı veren MLP parametreleri ile sınıflandırılmıştır. Alfa değeri [0.005-0.1] aralığında, gizli katman boyutu sayısı ve gizli katmanlardaki nöron sayısı sırasıyla [2-4] ve [10-100] aralığından seçilmiştir. MLP modelinin çözücü (solver) fonksiyonu sırasıyla “lbfgs”, “adam” ve “sgd” arasında seçilmiştir. Son olarak, aktivasyon fonksiyonu “relu” ve “lojistik” fonksiyonlar arasından seçilmiştir. Hiper parametre optimizasyonu sonucunda gizli katman sayısı üç, her bir gizli katmandaki nöron sayısı 100 olarak belirlenmiştir. Tablo 6.3’teki 100x100x100 ifadesi üç gizli katmanı ve her katmanda 100 nöron olduğunu göstermektedir.

Tablo 6.3. Hiper parametre optimizasyonu sonucu seçilen MLP parametreleri

Veri setleri	Aktivasyon fonksiyonu	Alpha değerleri	Gizli katman ve gizli katman nöron sayıları	Solver Fonksiyonu
ECU-IoHT	relu	0.002	100x100x100	adam
LR özellik seçiminden sonra ICU alt veri seti (ICU_LR)	relu	0.01	100x100x100	adam
XGB özellik seçiminden sonra ICU alt veri seti (ICU_XGB)	relu	0.01	100x100x100	adam
LR özellik seçiminden sonra TON-IoT alt veri seti (TON-IoT_LR)	relu	0.005	100x100x100	adam
XGB özellik seçiminden sonra TON-IoT alt veri seti (TON-IoT_XGB)	relu	0.005	100x100x100	adam
LR özellik seçiminden sonra WUSTL-EHMS alt veri seti (WUSTL-EHMS_LR)	relu	0.005	100x100x100	adam
XGB özellik seçiminden sonra WUSTL-EHMS alt veri seti (WUSTL-EHMS_XGB)	relu	0.005	100x100x100	adam

6.2. Bulgular

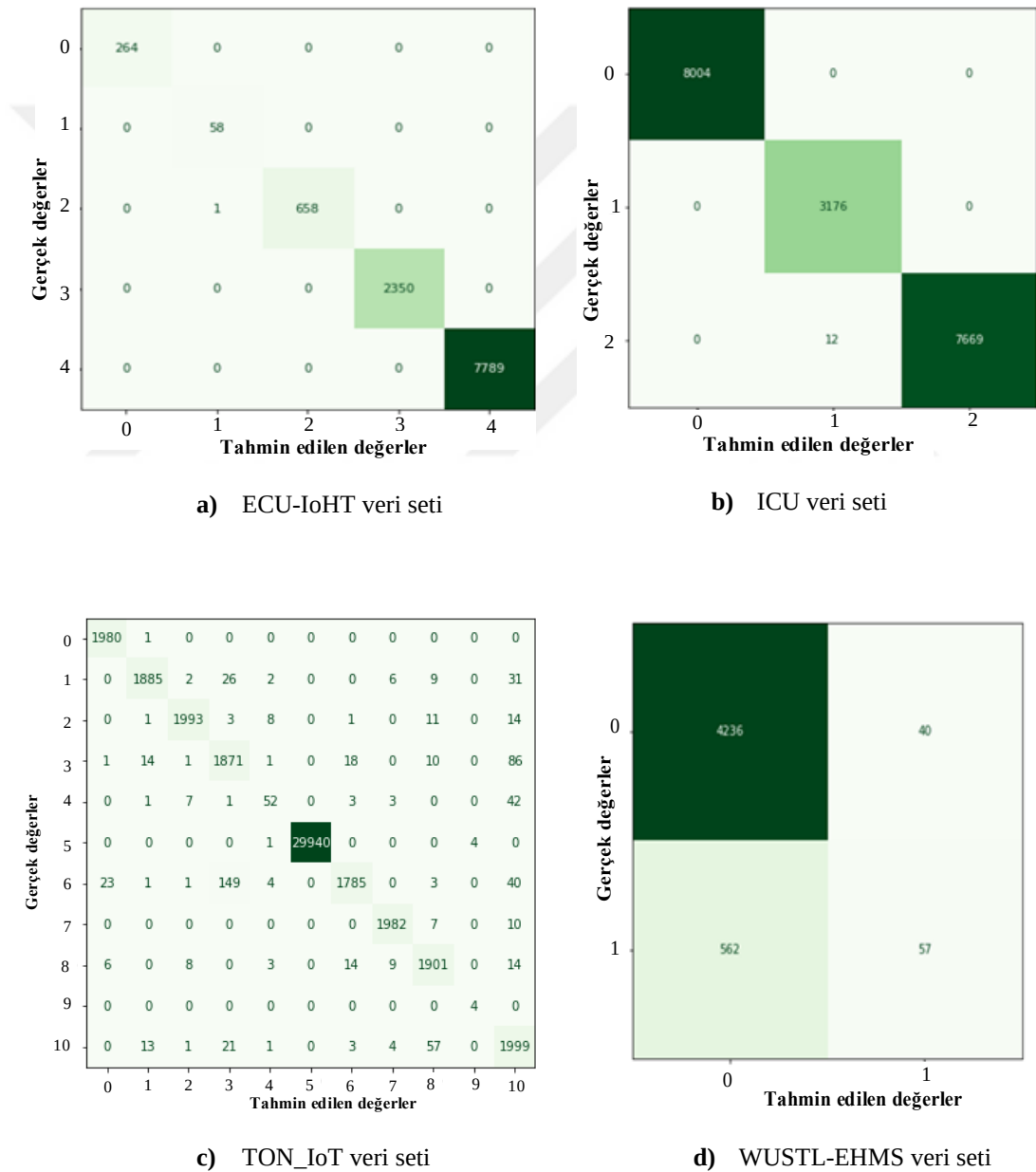
Önerilen yöntemde, en uygun özellikleri seçilen ICU, ToN_IoT, WUSTL-EHMS veri setleri ile herhangi bir özellik seçim işlemine tabii tutulmayan ECU-IoHT veri setleri MLP algoritması ile sınıflandırılmıştır. Elde edilen sonuçlar Tablo 6.4’ te verilmiştir.

Tablo 6.4. MLP sınıflandırıcısı sonuçları

Veri setleri	Doğruluk(%)
ECU-IoHT varsayılan MLP	99.8
ECU-IoHT optimize edilmiş MLP	99.99
ICU_LR varsayılan MLP	95.387
ICU_LR optimize edilmiş MLP	98.90
ICU_XGB varsayılan MLP	50.69
ICU_XGB optimize edilmiş tuned MLP	99.94
TON_IoT_LR varsayılan MLP	83.66
TON_IoT_LR optimize edilmiş MLP	96.90
TON_IoT_XGB varsayılan MLP	86.80
TON_IoT_XGB optimize edilmiş MLP	98.48
WUSTL-EHMS_LR varsayılan MLP	60.2
WUSTL-EHMS_LR optimize edilmiş MLP	88.47
WUSTL-EHMS_XGB varsayılan MLP	88.51
WUSTL-EHMS_XGB optimize edilmiş MLP	96.2

ECU-IoHT veri seti için, hiper parametre optimizasyonu olan ve olmayan sınıflandırma performansında çok az bir fark bulunmaktadır, bu da veri setindeki az sayıda kullanılabilir özellikten kaynaklanmaktadır. ICU veri seti için en iyi sonuç, XGBRegressor çekirdek işlevi tarafından seçilen özelliklere sahip alt veri setinde elde edilmiştir. Hiper parametre

optimizasyonundan sonra doğruluk oranı %50.69'dan %99.94'e yükselirken, ayarlamaların etkisi LR alt veri setinde çok daha zayıf kalmıştır. TON_IoT veri setinde, hem LR hem de XGB alt veri setleri için doğruluk oranları benzerdir, ancak hiper parametre optimizasyonundan sonra XGB alt veri setinde %98,48 doğruluk elde edilmiştir. WUSTL-EHMS veri setinde, hiper parametre optimizasyonundan sonra LR alt veri setlerine kıyasla XGB ile daha iyi sonuçlar elde edilmiştir. Genel olarak, XGBRegressor çekirdek fonksiyonu, LR çekirdek fonksiyonundan daha iyi performanslar sağlamıştır. Tablo 6.4'teki IoMT çalışma veri setleri için en iyi performans gösteren sonuçların karmaşıklık matrisleri Şekil 6.2'de verilmiştir.



Şekil 6.2. MLP sınıflandırıcısı sonucu karmaşıklık matrisleri

6.3. Tartışma

Bu bölümde IoMT cihazların güvenliği için önerilen yöntemin performansı literatürdeki çalışmalar ile karşılaştırılmıştır. Karşılaştırmalar Tablo 6.5’te verilmiştir.

Tablo 6.5. IoMT veri setleri için önerilen yöntem ve literatür karşılaştırması

Referans	Veri seti	Yöntem	Doğruluk (%)
[63]	WUSTL-EHMS	RF	92.13
		KNN	92.00
		SVM	92.40
		ANN	90.04
[184]	TON_IoT	Swarm-NN	99.5
[185]	TON_IoT	Ensemble-Anomaly detection system (E-ADS)	96.35
[62]	TON_IoT	Explainable simple recurrent units Internet of Medical Things (XSRU-IoMT)	99.38
[186]	TON_IoT	DT	99.97
		NB	34.44
		LR	98.70
		RF	99.96
		KNN	99.98
		SVM	98.73
[60]	ICU Dataset	NB	52.18
		KNN	99.48
		RF	99.51
		AB	99.50
		LR	99.50
		DT	99.47
[187]	ICU Dataset	NB	97.11
		KNN	99.92
		RF	99.99
Önerilen Yöntem	ECU-IoHT	MLP	99.99
	ICU		99.94
	TON_IoT		98.48
	WUSTL-EHMS		96.2

Tablo 6.5’te IoMT veri setleri ile yapılmış çalışmalar ile önerilen yöntem karşılaştırılmıştır. Literatürde NSL-KDD, CICIDS2018 gibi STS veri setleri ile yapılan IoMT çalışmaları mevcuttur. Fakat bu çalışmada gerçek anlamda medikal cihazlar üzerinden oluşturulan veri setleri ele alınmıştır. IoMT veri setlerinin literatürde kullanımı henüz yaygın olmadığı için önerilen yöntem çok fazla literatür çalışması ile karşılaştıramamıştır. Önerilen metod, ECU_IoHT veri seti için %99.99 başarımlık değeri yakalamıştır. ICU veri seti için yaklaşık %99.94’lük başarımlık değerine ulaşmıştır. ICU veri seti ile yapılan literatür çalışmalarında “environmentMonitoring” ve “patientMonitoring” sınıfları “Normal” sınıf altında birleştirilmiştir. Önerilen çalışmada bu sınıflar

ayrı ayrı düşünülmüş ve buna rağmen yüksek başarımlar elde edilmiştir. ToN_IoT veri seti için %98.48 başarımlarına ulaşılmıştır. Bu veri seti için de yine iyi bir başarımlarına ulaşılmış ve literatür çalışmaları ile benzer seviyeler yakalanmıştır. WUSTL_EHMS veri setinde %96.2'lük bir başarımlar elde edilmiştir. Bu veri seti için elde edilen sonuç literatürdeki en yüksek başarımdır. Ek olarak, hiper parametre optimizasyonunun sınıflandırma performanslarını önemli ölçüde iyileştirmiştir. ECU-IoHT, ICU, ToN_IoT ve WUSTL-EHMS veri setlerindeki doğruluk oranları, varsayılan MLP hiper parametre ayarlarıyla sırasıyla %99,8, %95,38, %83,66 ve %60,2 iken, hiper parametre optimizasyonundan sonra sırasıyla %99,99, %99,94, %98,48 ve %96,20 doğruluk değerlerine ulaşılmıştır. Çalışma, XGBRegressor tabanlı öznelik seçimi ve MLP hiper parametre optimizasyon işleminin, mevcut IoMT siber güvenlik araştırma veri setlerinde üstün performans sağladığını gösteren ilk çalışmadır.

Önerilen yöntemin en büyük dezavantajı, büyük veri setlerinde, örneğin 460.925 kayıt içeren ToN_IoT veri setinde pahalı olabilen hiper parametre ayarlama zamanıdır. Bu sınırlamayı aşmak için, veri setinin küçük bir alt seti rastgele örneklenebilir. Bu çalışmada, optimal hiper parametre ayarlarını belirlemek için ToN_IoT veri setinden rastgele 20.000 kayıt seçilmiştir. Daha sonra veri setinin tamamına MLP sınıflandırıcısı uygulanmıştır. Hiper parametre seçim süreci bir defalığına yapıldığı için, önemli ölçüde geliştirilmiş sınıflandırma performansı, zaman maliyetinden daha ağır basmaktadır.

6.4. Sonuçlar

Sağlık alanında kullanımı gittikçe artan IoMT cihazlarının güvenliğini sağlamak kritik bir konu haline gelmiştir. Bu sistemlere yapılabilecek siber saldırılar, insan hayatını olumsuz etkileyebilmektedir. Bu çalışmada, sağlık sistemlerinde sıklıkla kullanılan cihazlardan elde edilen ECU_IoHT, ICU, ToN_IoT ve WUSTL_EHMS IoMT veri setleri ele alınmıştır. Veri setlerinin sınıflandırılması için RFE tabanlı bir özellik seçim metodu ve MLP sınıflandırıcısı kullanılmıştır. Çalışmanın performansının artırılması için RFE özellik seçim metodunun çekirdek fonksiyonları LR ve XGBRegressor seçilmiştir. Sınıflandırma için kullanılan MLP algoritmasının sınıflandırma performansının artırılması için hiper parametre optimizasyonu yapılmıştır. Yapılan işlemler sonucunda ECU_IoHT, ICU, ToN_IoT ve WUSTL_EHMS IoMT veri setleri için sırasıyla %99.99, %99.94, %98.48 ve %96.2 doğruluk değerleri elde edilmiştir. Önerilen metod tüm IoMT veri setlerinin sınıflandırılmasında yüksek başarımlar değerlerine ulaşmıştır.

7. SONUÇLAR

İnternet, hem gündelik yaşamın hem de kurum ve kuruluşların vazgeçilmez bir unsuru haline gelmiştir. İnternetin bu kadar yaygın kullanımı beraberinde birçok güvenlik açığını da getirmektedir. Bu güvenlik açıkları, ağ ve sistemleri, bilgisayar korsanlarına, virüslere ve diğer saldırılara karşı savunmasız hale getirir. Bu güvenlik açıklarından gelebilecek tehditlerin önlenmesi amacıyla geliştirilen STS sistemleri günümüzdeki en popüler çalışma alanlarından biridir. Bu tez çalışmasında hem STS sistemlerinin geliştirilmesine hem de kullanıcıların internet davranış analizine yönelik yeni veri setleri ve yeni yöntemler geliştirilmiştir.

Çalışmanın ikinci bölümünde literatürde sıklıkla kullanılan CSE-CIC-IDS2018, UNSW-NB15, ISCX-2012, NSL-KDD ve CIDDs-001 veri setlerinin önerilen yöntem altında performansları ölçülmüştür. Önerilen yöntemde DT algoritması; CSE-CIC-IDS2018, ISCX-2012, NSL-KDD ve CIDDs-001 veri setleri için sırasıyla %99.9, %100, %99.92, %99.66 doğruluk değerleri elde edilmiştir. Kategorizasyon işlemi yapılan UNSW-NB15 veri seti için ise %99.84 doğruluk değeri elde edilmiştir.

Çalışmanın üçüncü bölümünde STS sistemlerin geliştirilmesi için CCiDD_A ve CCiDD_B veri setleri oluşturulmuştur. Önerilen veri setlerinin başarımlarının karşılaştırılması için CSE-CIC-IDS2018, UNSW-NB15 ve NSL-KDD veri setleri kullanılmıştır. Veri setlerinin performanslarını artırmak için en uygun özellikler Extra Tree algoritması ile seçilmiştir. Elde edilen uygun özellikler KNN, SVM, DT, NB, MLP, Adaboost, GBM, XGBoost, LGBM ve CatBoost sınıflandırıcıları ile sınıflandırılmıştır. Sınıflandırma sonucunda, LGBM ve XGBoost sınıflandırıcıları diğer makine öğrenmesi yöntemlerinden daha başarılı oldukları için bu sınıflandırıcılara hiper parametre optimizasyonu uygulanmıştır. Hiper parametre optimizasyonu sonrasında XGBoost ve LGBM benzer performans sonuçları elde etmesine rağmen, LGBM sınıflandırıcı sınıflandırma hızı ve performans açısından daha başarılı olmuştur. Tüm veri setleri için en iyi sonuçlar LGBM algoritması ile elde edilmiştir. LGBM algoritması NSL-KDD, UNSW-NB15, CSE-CIC-IDS2018, CCiDD_A ve CCiDD_B veri setleri için sırasıyla %99.84, %98.02, %99.94, %95.68 ve %99.98 doğruluk değerlerine ulaşmıştır. Ayrıca LGBM sınıflandırma hızı bakımından tüm veri setleri için, XGBoost sınıflandırıcısına göre yaklaşık 8 kat daha hızlıdır.

Çalışmanın dördüncü bölümünde yerel ağlarda meydana gelen saldırı senaryoları kullanılarak bir SPA-IDS veri seti oluşturulmuştur. Önerilen veri seti, anahtarlama cihazlarının portlarındaki anomali değişimleri dikkate alınarak oluşturulmuştur. Elde edilen SPA_IDS verileri bir dakikalık uzunluklara bölünmüş ve veriler bir sinyal halinde önerilen yönteme yüklenmiştir. Yöntemin ilk adımında, önerilen VMD modeli sinyale uygulanmış ve VMD katsayıları elde edilmiştir. Daha sonra elde edilen VMD katsayılarından INCA metodu ile en uygun özellikler

seçilmiş ve KNN, SVM, DT ve BT sınıflandırıcıları ile sınıflandırılmıştır. Sınıflandırıcı sonucunda en iyi başarımlar %97.74 ile BT sınıflandırıcısı ile sağlanmıştır.

Çalışmanın beşinci bölümünde günümüzde oldukça popüler hale gelen internet davranış analizi ele alınmıştır. Özellikle e-ticaret, reklam gibi alanlarda oldukça önemli yere sahip internet analizi için yeni bir IoBe veri seti önerilmiştir. Önerilen veri seti normal bir internet kullanıcısının kullanabileceği tüm uygulamaların trafiğinden oluşmaktadır. Çalışmada IoBe veri seti, önerilen MP, AP, MAP ve CP havuzlama işlemlerinden geçirildikten sonra ReliefF özellik seçicisine iletilmiştir. ReliefF özellik seçicisinden en uygun özellikler 6 bölünmüş kriter parametresi kullanılarak BT sınıflandırıcısı ile sınıflandırılmıştır. Böylece 6 doğrulama tahmin vektörü elde edilmiştir. Elde edilen bu 6 sonuca çoğunluk oylaması uygulanmış %83.01 başarımla değerlendirilmiştir. Ayrıca önerilen IoBe veri setinin başarımları CICDarknet 2020 veri seti ile karşılaştırılmıştır.

Çalışmanın altıncı ve son bölümünde tıbbi nesnelere yapılabilecek siber saldırılarının önlenmesine yönelik bir yöntem önerilmiştir. Önerilen yöntemde RFE özellik seçim algoritması ile en uygun özellikler seçilmiş ve elde edilen özellikler MLP sınıflandırıcısı ile sınıflandırılmıştır. Sınıflandırma performansının artırılması için MLP algoritması hiper parametre optimizasyonu ile desteklenmiştir. Önerilen yöntem sonucunda ECU_IoHT, ICU, ToN_IoT ve WUSTL_EHMS IoMT veri setleri için sırasıyla %99.99, %99.94, %98.48 ve %96.2 doğruluk değerleri alınmıştır.

Bu tez çalışmasında, ağ tabanlı STS sistemleri, IoMT tabanlı STS sistemleri ve kullanıcı davranış analizi konuları ele alınmıştır. Tez kapsamında birçok hazır veri seti incelendiği gibi, literatüre katkı sağlayacak CCIDD_A, CCIDD_B, SPA_IDS, IoBe veri setleri üretilmiştir. Çalışmanın her bir bölümünde veri setlerinin sınıflandırılması ve performanslarının artırılmasına yönelik orijinal çözümler sunulmuştur. Tez çalışması sunduğu yeni veri setleri ve sınıflandırma yöntemleri ile literatüre orijinal bir bakış açısı getireceği değerlendirilmektedir.

ÖNERİLER

Tez kapsamında birçok veri seti oluşturulmuş ve yeni yaklaşımlar önerilmiştir. Bunların yanı sıra tez kapsamına alınmayan fakat yaygın kullanım alanı olan birçok sistem için yeni çalışmalar yapılabilir. Bu kapsamda sunulan öneriler aşağıdaki maddeler halinde sunulmuştur.

- SCADA sistemlerine yönelik saldırıların tespiti ve önlenmesine yönelik veri setleri oluşturulabilir.
- IoT ağlarına yapılan saldırıların tespiti için kapsamlı bir IoT sensör ağı kurulup bu ağdaki sensörlere çeşitli saldırılar düzenlenip yeni bir IoT güvenlik çerçevesi önerilebilir.
- Çalışmanın 3. Bölümünde kullanılan ve veri trafiğini özelliklerine ayırtmak için kullanılan CICFlowmeter yazılımı gibi yeni bir yazılım geliştirilip, bu yazılım ticarileştirilebilir.
- Tıbbi alanlardaki saldırıların tespitini sağlamak için tıbbi sensörlerden oluşan bir ağ kurulup, bu alana yönelik daha kapsamlı çalışmalar üretilebilir.

KAYNAKLAR

- [1] Simon Kemp, DIGITAL 2022: GLOBAL OVERVIEW REPORT, (n.d.). <https://datareportal.com/reports/digital-2022-global-overview-report> (accessed May 18, 2022).
- [2] R. Security, 2021 Mid Year Report: Data Breach QuickView, (2021). [https://pages.riskbasedsecurity.com/hubfs/Reports/2021/2021 Mid Year Data Breach QuickView Report.pdf](https://pages.riskbasedsecurity.com/hubfs/Reports/2021/2021%20Mid%20Year%20Data%20Breach%20QuickView%20Report.pdf).
- [3] C.M. Research, Cyber Security Market by Component, Security Type, Deployment, Organization and Application - Global Industry Analysis and Forecast to 2022, (2017) 202. <https://www.crystalmarketresearch.com/report/cyber-security-market>.
- [4] D. Kiwia, A. Dehghantanha, K.K.R. Choo, J. Slaughter, A cyber kill chain based taxonomy of banking Trojans for evolutionary computational intelligence, J. Comput. Sci. (2018). <https://doi.org/10.1016/j.jocs.2017.10.020>.
- [5] T. Yadav, A.M. Rao, Technical aspects of cyber kill chain, in: Commun. Comput. Inf. Sci., 2015. https://doi.org/10.1007/978-3-319-22915-7_40.
- [6] T. Dargahi, A. Dehghantanha, P.N. Bahrani, M. Conti, G. Bianchi, L. Benedetto, A Cyber-Kill-Chain based taxonomy of crypto-ransomware features, J. Comput. Virol. Hacking Tech. (2019). <https://doi.org/10.1007/s11416-019-00338-7>.
- [7] E. Hutchins, M. Cloppert, R. Amin, Intelligence-driven computer network defense informed by analysis of adversary campaigns and intrusion kill chains, in: 6th Int. Conf. Inf. Warf. Secur. ICIW 2011, 2011.
- [8] S. Wang, Y. Gong, G. Chen, Q. Sun, F. Yang, Service vulnerability scanning based on service-oriented architecture in Web service environments, J. Syst. Archit. (2013). <https://doi.org/10.1016/j.sysarc.2013.01.002>.
- [9] H. Holm, Performance of automated network vulnerability scanning at remediating security issues, Comput. Secur. (2012). <https://doi.org/10.1016/j.cose.2011.12.014>.
- [10] Nmap Reference Guide, (n.d.). <https://nmap.org/book/man.html> (accessed May 10, 2021).
- [11] R. Daş, A. Karabade, G. Tuna, Common network attack types and defense mechanisms, in: 2015 23rd Signal Process. Commun. Appl. Conf. SIU 2015 - Proc., 2015. <https://doi.org/10.1109/SIU.2015.7130435>.
- [12] N. Hubballi, N. Tripathi, A closer look into DHCP starvation attack in wireless networks, Comput. Secur. (2017). <https://doi.org/10.1016/j.cose.2016.10.002>.
- [13] H. Mukhtar, K. Salah, Y. Iraqi, Mitigation of DHCP starvation attack, Comput. Electr. Eng. (2012). <https://doi.org/10.1016/j.compeleceng.2012.06.005>.
- [14] T. Girdler, V.G. Vassilakis, Implementing an intrusion detection and prevention system using Software-Defined Networking: Defending against ARP spoofing attacks and Blacklisted MAC Addresses, Comput. Electr. Eng. (2021). <https://doi.org/10.1016/j.compeleceng.2021.106990>.
- [15] R. Hofstede, M. Jonker, A. Sperotto, A. Pras, Flow-Based Web Application Brute-Force Attack and Compromise Detection, J. Netw. Syst. Manag. (2017). <https://doi.org/10.1007/s10922-017-9421-4>.
- [16] J. Park, J. Kim, B.B. Gupta, N. Park, Network Log-Based SSH Brute-Force Attack Detection Model, Comput. Mater. Contin. 68 (2021) 887–901. <https://doi.org/10.32604/cmc.2021.015172>.
- [17] XSSStrike, (n.d.). <https://github.com/s0md3v/XSSStrike> (accessed May 15, 2021).
- [18] C.L. Zhang, G.H. Yang, A.Y. Lu, Resilient observer-based control for cyber-physical systems under denial-of-service attacks, Inf. Sci. (Ny). 545 (2021) 102–117. <https://doi.org/10.1016/j.ins.2020.07.070>.
- [19] K.B. Virupakshar, M. Asundi, K. Channal, P. Shettar, S. Patil, D.G. Narayan, Distributed Denial of Service (DDoS) Attacks Detection System for OpenStack-based Private Cloud, in: Procedia Comput. Sci., 2020. <https://doi.org/10.1016/j.procs.2020.03.282>.

- [20] L. Barki, A. Shidling, N. Meti, D.G. Narayan, M.M. Mulla, Detection of distributed denial of service attacks in software defined networks, in: 2016 Int. Conf. Adv. Comput. Commun. Informatics, ICACCI 2016, 2016. <https://doi.org/10.1109/ICACCI.2016.7732445>.
- [21] L. Guo, H. Yu, F. Hao, Event-triggered control for stochastic networked control systems against Denial-of-Service attacks, *Inf. Sci. (Ny)*. (2020). <https://doi.org/10.1016/j.ins.2020.03.045>.
- [22] J.J. Kponyo, J.O. Agyemang, G.S. Klogo, J.O. Boateng, Lightweight and host-based denial of service (DoS) detection and defense mechanism for resource-constrained IoT devices, *Internet of Things (Netherlands)*. 12 (2020). <https://doi.org/10.1016/j.iot.2020.100319>.
- [23] V. Nagaraju, A. Raaza, V. Rajendran, D. Ravikumar, Deep learning binary fruit fly algorithm for identifying SYN flood attack from TCP/IP, *Mater. Today Proc.* (2021). <https://doi.org/10.1016/j.matpr.2021.07.171>.
- [24] Exploit (computer security), (n.d.). [https://en.wikipedia.org/wiki/Exploit_\(computer_security\)](https://en.wikipedia.org/wiki/Exploit_(computer_security)) (accessed November 1, 2020).
- [25] Exploit-db, Anatomy of Exploit - World of Shellcode, (n.d.). <https://www.exploit-db.com/papers/35646>.
- [26] Fuzzing, (n.d.). <https://en.wikipedia.org/wiki/Fuzzing> (accessed November 2, 2020).
- [27] C. Chen, B. Cui, J. Ma, R. Wu, J. Guo, W. Liu, A systematic review of fuzzing techniques, *Comput. Secur.* (2018). <https://doi.org/10.1016/j.cose.2018.02.002>.
- [28] K. Chen, C. Song, L. Wang, Z. Xu, Using memory propagation tree to improve performance of protocol fuzzer when testing ICS, *Comput. Secur.* (2019). <https://doi.org/10.1016/j.cose.2019.101582>.
- [29] 'Heartbleed' flaw leaves millions of websites, email servers and other services vulnerable to attack, *Netw. Secur.* (2014). [https://doi.org/10.1016/s1353-4858\(14\)70036-4](https://doi.org/10.1016/s1353-4858(14)70036-4).
- [30] Heartbleed bug leads to forking and funding, *Netw. Secur.* (2014). [https://doi.org/10.1016/s1353-4858\(14\)70045-5](https://doi.org/10.1016/s1353-4858(14)70045-5).
- [31] P. Tang, W. Qiu, Z. Huang, H. Lian, G. Liu, Detection of SQL injection based on artificial neural network, *Knowledge-Based Syst.* (2020). <https://doi.org/10.1016/j.knosys.2020.105528>.
- [32] P.R. McWhirter, K. Kifayat, Q. Shi, B. Askwith, SQL Injection Attack classification through the feature extraction of SQL query strings using a Gap-Weighted String Subsequence Kernel, *J. Inf. Secur. Appl.* (2018). <https://doi.org/10.1016/j.jisa.2018.04.001>.
- [33] S. Manimurugan, A. qdah Majdi, M. Mohmmmed, C. Narmatha, R. Varatharajan, Intrusion detection in networks using crow search optimization algorithm with adaptive neuro-fuzzy inference system, *Microprocess. Microsyst.* (2020). <https://doi.org/10.1016/j.micpro.2020.103261>.
- [34] Y. Al-Hadhrami, F.K. Hussain, Real time dataset generation framework for intrusion detection systems in IoT, *Futur. Gener. Comput. Syst.* (2020). <https://doi.org/10.1016/j.future.2020.02.051>.
- [35] P. García-Teodoro, J. Díaz-Verdejo, G. Maciá-Fernández, E. Vázquez, Anomaly-based network intrusion detection: Techniques, systems and challenges, *Comput. Secur.* 28 (2009) 18–28. <https://doi.org/10.1016/j.cose.2008.08.003>.
- [36] Z. Inayat, A. Gani, N.B. Anuar, M.K. Khan, S. Anwar, Intrusion response systems: Foundations, design, and challenges, *J. Netw. Comput. Appl.* 62 (2016) 53–74. <https://doi.org/10.1016/j.jnca.2015.12.006>.
- [37] A.S. Ashoor, S. Gore, Difference between Intrusion Detection System (IDS) and Intrusion Prevention System (IPS), in: *Commun. Comput. Inf. Sci.*, 2011: pp. 497–501. https://doi.org/10.1007/978-3-642-22540-6_48.
- [38] B. Subba, P. Gupta, A tfidfvectorizer and singular value decomposition based host intrusion detection system framework for detecting anomalous system processes, *Comput. Secur.* (2021). <https://doi.org/10.1016/j.cose.2020.102084>.
- [39] I.F. Kilincer, F. Ertam, A. Sengur, Machine learning methods for cyber security intrusion detection: Datasets and comparative study, *Comput. Networks.* (2021).

<https://doi.org/10.1016/j.comnet.2021.107840>.

- [40] M.A. Ferrag, L. Maglaras, S. Moschoyiannis, H. Janicke, Deep learning for cyber security intrusion detection: Approaches, datasets, and comparative study, *J. Inf. Secur. Appl.* 50 (2020). <https://doi.org/10.1016/j.jisa.2019.102419>.
- [41] M. Ring, S. Wunderlich, D. Scheuring, D. Landes, A. Hotho, A survey of network-based intrusion detection data sets, *Comput. Secur.* 86 (2019) 147–167. <https://doi.org/10.1016/j.cose.2019.06.005>.
- [42] G. Maciá-Fernández, J. Camacho, R. Magán-Carrión, P. García-Teodoro, R. Therón, UGR'16: A new dataset for the evaluation of cyclostationarity-based network IDSs, *Comput. Secur.* 73 (2018) 411–424. <https://doi.org/10.1016/j.cose.2017.11.004>.
- [43] Y. Xin, L. Kong, Z. Liu, Y. Chen, Y. Li, H. Zhu, M. Gao, H. Hou, C. Wang, Machine Learning and Deep Learning Methods for Cybersecurity, *IEEE Access.* 6 (2018) 35365–35381. <https://doi.org/10.1109/ACCESS.2018.2836950>.
- [44] M. Tavallaee, E. Bagheri, W. Lu, A.A. Ghorbani, A detailed analysis of the KDD CUP 99 data set, in: *IEEE Symp. Comput. Intell. Secur. Def. Appl. CISDA 2009*, 2009. <https://doi.org/10.1109/CISDA.2009.5356528>.
- [45] KDD Cup 1999 Data, (n.d.). <http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html>.
- [46] J. Song, H. Takakura, Y. Okabe, Description of Kyoto University Benchmark Data, (2010) 10–12. http://www.takakura.com/Kyoto_data/BenchmarkData-Description-v5.pdf.
- [47] N. Moustafa, J. Slay, UNSW-NB15: A comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set), in: *2015 Mil. Commun. Inf. Syst. Conf. MilCIS 2015 - Proc.*, 2015. <https://doi.org/10.1109/MilCIS.2015.7348942>.
- [48] N. Moustafa, J. Slay, The evaluation of Network Anomaly Detection Systems: Statistical analysis of the UNSW-NB15 data set and the comparison with the KDD99 data set, *Inf. Secur. J.* 25 (2016) 18–31. <https://doi.org/10.1080/19393555.2015.1125974>.
- [49] A. Verma, V. Ranga, Statistical analysis of CIDDS-001 dataset for Network Intrusion Detection Systems using Distance-based Machine Learning, in: *Procedia Comput. Sci.*, 2018: pp. 709–716. <https://doi.org/10.1016/j.procs.2017.12.091>.
- [50] A. Verma, V. Ranga, On evaluation of network intrusion detection systems: Statistical analysis of CIDDS-001 dataset using machine learning techniques, *Pertanika J. Sci. Technol.* 26 (2018) 1307–1332.
- [51] M. Ring, S. Wunderlich, M. Ring, Technical Report CIDDS-001 data set, 16 (2017) 361–369. https://www.hs-coburg.de/fileadmin/hscoburg/Forschung/WISENT_cidds_Technical_Report.pdf.
- [52] M. Ring, S. Wunderlich, D. Grödl, D. Landes, A. Hotho, Flow-based benchmark data sets for intrusion detection, in: *Eur. Conf. Inf. Warf. Secur. ECCWS*, 2017: pp. 361–369.
- [53] A. Shiravi, H. Shiravi, M. Tavallaee, A.A. Ghorbani, Toward developing a systematic approach to generate benchmark datasets for intrusion detection, *Comput. Secur.* 31 (2012) 357–374. <https://doi.org/10.1016/j.cose.2011.12.012>.
- [54] C. Kolas, G. Kambourakis, A. Stavrou, S. Gritzalis, Intrusion detection in 802.11 networks: Empirical evaluation of threats and a public dataset, *IEEE Commun. Surv. Tutorials.* 18 (2016) 184–208. <https://doi.org/10.1109/COMST.2015.2402161>.
- [55] U.S.K.P.M. Thanthrige, J. Samarabandu, X. Wang, Machine learning techniques for intrusion detection on public dataset, in: *Can. Conf. Electr. Comput. Eng.*, 2016. <https://doi.org/10.1109/CCECE.2016.7726677>.
- [56] I. Sharafaldin, A.H. Lashkari, A.A. Ghorbani, Toward generating a new intrusion detection dataset and intrusion traffic characterization, in: *ICISSP 2018 - Proc. 4th Int. Conf. Inf. Syst. Secur. Priv.*, 2018: pp. 108–116. <https://doi.org/10.5220/0006639801080116>.
- [57] Intrusion Detection Evaluation Dataset (CICIDS2017), (n.d.). <https://www.unb.ca/cic/datasets/ids-2017.html>.
- [58] CSE-CIC-IDS2018 on AWS, (n.d.). <https://www.unb.ca/cic/datasets/ids-2018.html>.

- [59] M. Ahmed, S. Byreddy, A. Nutakki, L.F. Sikos, P. Haskell-Dowland, ECU-IoHT: A dataset for analyzing cyberattacks in Internet of Health Things, *Ad Hoc Networks*. (2021). <https://doi.org/10.1016/j.adhoc.2021.102621>.
- [60] F. Hussain, S.G. Abbas, G.A. Shah, I.M. Pires, U.U. Fayyaz, F. Shahzad, N.M. Garcia, E. Zdravevski, A framework for malicious traffic detection in iot healthcare environment, *Sensors*. (2021). <https://doi.org/10.3390/s21093025>.
- [61] A. Alsaedi, N. Moustafa, Z. Tari, A. Mahmood, Adna N Anwar, TON-IoT telemetry dataset: A new generation dataset of IoT and IIoT for data-driven intrusion detection systems, *IEEE Access*. 8 (2020) 165130–165150. <https://doi.org/10.1109/ACCESS.2020.3022862>.
- [62] I.A. Khan, N. Moustafa, I. Razzak, M. Tanveer, D. Pi, Y. Pan, B.S. Ali, XSRU-IoMT: Explainable simple recurrent units for threat detection in Internet of Medical Things networks, *Futur. Gener. Comput. Syst.* (2022). <https://doi.org/10.1016/j.future.2021.09.010>.
- [63] A.A. Hady, A. Ghubaish, T. Salman, D. Unal, R. Jain, Intrusion Detection System for Healthcare Systems Using Medical and Network Data: A Comparison Study, *IEEE Access*. (2020). <https://doi.org/10.1109/ACCESS.2020.3000421>.
- [64] R. Patil, H. Dudeja, C. Modi, Designing an efficient security framework for detecting intrusions in virtual network of cloud computing, *Comput. Secur.* 85 (2019) 402–422. <https://doi.org/10.1016/j.cose.2019.05.016>.
- [65] V. Kanimozhi, D.T.P. Jacob, Calibration of Various Optimized Machine Learning Classifiers in Network Intrusion Detection System on the Realistic Cyber Dataset Cse-Cic-Ids2018 Using Cloud Computing, *Int. J. Eng. Appl. Sci. Technol.* 04 (2019) 209–213. <https://doi.org/10.33564/ijeast.2019.v04i06.036>.
- [66] C. Khammassi, S. Krichen, A NSGA2-LR wrapper approach for feature selection in network intrusion detection, *Comput. Networks*. 172 (2020) 107183. <https://doi.org/10.1016/j.comnet.2020.107183>.
- [67] J. Kim, Y. Shin, E. Choi, An Intrusion Detection Model based on a Convolutional Neural Network, *J. Multimed. Inf. Syst.* 6 (2019) 165–172. <https://doi.org/10.33851/jmis.2019.6.4.165>.
- [68] V. Kanimozhi, T.P. Jacob, Artificial Intelligence based Network Intrusion Detection with hyper-parameter optimization tuning on the realistic cyber dataset CSE-CIC-IDS2018 using cloud computing, *ICT Express*. 5 (2019) 211–214. <https://doi.org/10.1016/j.icte.2019.03.003>.
- [69] N. Moustafa, J. Slay, G. Creech, Novel Geometric Area Analysis Technique for Anomaly Detection Using Trapezoidal Area Estimation on Large-Scale Networks, *IEEE Trans. Big Data*. 5 (2019) 481–494. <https://doi.org/10.1109/TBDATA.2017.2715166>.
- [70] J. Zhang, Y. Ling, X. Fu, X. Yang, G. Xiong, R. Zhang, Model of the intrusion detection system based on the integration of spatial-temporal features, *Comput. Secur.* 89 (2020) 101681. <https://doi.org/10.1016/j.cose.2019.101681>.
- [71] F. Gottwalt, E. Chang, T. Dillon, CorrCorr: A feature selection method for multivariate correlation network anomaly detection techniques, *Comput. Secur.* 83 (2019) 234–245. <https://doi.org/10.1016/j.cose.2019.02.008>.
- [72] C. Khammassi, S. Krichen, A GA-LR wrapper approach for feature selection in network intrusion detection, *Comput. Secur.* 70 (2017) 255–277. <https://doi.org/10.1016/j.cose.2017.06.005>.
- [73] S.M. Kasongo, Y. Sun, A deep learning method with wrapper based feature extraction for wireless intrusion detection system, *Comput. Secur.* 92 (2020). <https://doi.org/10.1016/j.cose.2020.101752>.
- [74] V. Hajisalem, S. Babaie, A hybrid intrusion detection system based on ABC-AFS algorithm for misuse and anomaly detection, *Comput. Networks*. 136 (2018) 37–50. <https://doi.org/10.1016/j.comnet.2018.02.028>.
- [75] M.M. Hassan, A. Gumaei, A. Alsanad, M. Alrubaian, G. Fortino, A hybrid deep learning model for efficient intrusion detection in big data environment, *Inf. Sci. (Ny)*. (2020). <https://doi.org/10.1016/j.ins.2019.10.069>.
- [76] M. Awad, A. Alabdallah, Addressing imbalanced classes problem of intrusion detection system using

- weighted Extreme Learning Machine, *Int. J. Comput. Networks Commun.* (2019). <https://doi.org/10.5121/ijcnc.2019.11503>.
- [77] Z. Tan, A. Jamdagni, X. He, P. Nanda, R.P. Liu, J. Hu, Detection of Denial-of-Service Attacks Based on Computer Vision Techniques, *IEEE Trans. Comput.* (2015). <https://doi.org/10.1109/TC.2014.2375218>.
- [78] I. Bouteraa, M. Derdour, A. Ahmim, Intrusion Detection using Data Mining: A contemporary comparative study, *Proc. - PAIS 2018 Int. Conf. Pattern Anal. Intell. Syst.* (2018). <https://doi.org/10.1109/PAIS.2018.8598494>.
- [79] W. Yassin, N.I. Udzir, Z. Muda, Anomaly-based Intrusion Detection Through K- Means Clustering and Naive Bayes Classification, in: *Proc. 4th Int. Conf. Comput. Informatics, ICOCI 2013*, 2013.
- [80] A Deeper Dive into the NSL-KDD Data Set, (n.d.). <https://towardsdatascience.com/a-deeper-dive-into-the-ns-l-kdd-data-set-15c753364657>.
- [81] A. Rashid, M.J. Siddique, S.M. Ahmed, Machine and Deep Learning Based Comparative Analysis Using Hybrid Approaches for Intrusion Detection System, (2020).
- [82] J. David, C. Thomas, Efficient DDoS flood attack detection using dynamic thresholding on flow-based network traffic, *Comput. Secur.* 82 (2019) 284–295. <https://doi.org/10.1016/j.cose.2019.01.002>.
- [83] R.P. Lippmann, D.J. Fried, I. Graf, J.W. Haines, K.R. Kendall, D. McClung, D. Weber, S.E. Webster, D. Wyschogrod, R.K. Cunningham, M.A. Zissman, Evaluating intrusion detection systems: The 1998 DARPA off-line intrusion detection evaluation, in: *Proc. - DARPA Inf. Surviv. Conf. Expo. DISCEX 2000*, 2000: pp. 12–26. <https://doi.org/10.1109/DISCEX.2000.821506>.
- [84] F. Ertam, I.F. Kiliçer, O. Yaman, Intrusion detection in computer networks via machine learning algorithms, in: *IDAP 2017 - Int. Artif. Intell. Data Process. Symp.*, 2017. <https://doi.org/10.1109/IDAP.2017.8090165>.
- [85] M. Injadat, F. Salo, A.B. Nassif, A. Essex, A. Shami, Bayesian Optimization with Machine Learning Algorithms Towards Anomaly Detection, in: *2018 IEEE Glob. Commun. Conf. GLOBECOM 2018 - Proc.*, 2018. <https://doi.org/10.1109/GLOCOM.2018.8647714>.
- [86] N. Dash, S. Chakravarty, S. Satpathy, An improved harmony search based extreme learning machine for intrusion detection system, *Mater. Today Proc.* (2021). <https://doi.org/10.1016/j.matpr.2021.01.619>.
- [87] J.K. Samriya, N. Kumar, A novel intrusion detection system using hybrid clustering-optimization approach in cloud computing, *Mater. Today Proc.* (2020). <https://doi.org/10.1016/j.matpr.2020.09.614>.
- [88] Y.N. Kunang, S. Nurmaini, D. Stiawan, B.Y. Suprpto, Attack classification of an intrusion detection system using deep learning and hyperparameter optimization, *J. Inf. Secur. Appl.* (2021). <https://doi.org/10.1016/j.jisa.2021.102804>.
- [89] S. Choudhary, N. Kesswani, Analysis of KDD-Cup'99, NSL-KDD and UNSW-NB15 Datasets using Deep Learning in IoT, in: *Procedia Comput. Sci.*, 2020. <https://doi.org/10.1016/j.procs.2020.03.367>.
- [90] P. Dahiya, D.K. Srivastava, A comparative evolution of unsupervised techniques for effective network intrusion detection in Hadoop, in: *Commun. Comput. Inf. Sci.*, 2018. https://doi.org/10.1007/978-981-13-1813-9_28.
- [91] S. Thakur, A. Chakraborty, R. De, N. Kumar, R. Sarkar, Intrusion detection in cyber-physical systems using a generic and domain specific deep autoencoder model, *Comput. Electr. Eng.* (2021). <https://doi.org/10.1016/j.compeleceng.2021.107044>.
- [92] N. Moustafa, G. Creech, E. Sitnikova, M. Keshk, Collaborative anomaly detection framework for handling big data of cloud computing, in: *2017 Mil. Commun. Inf. Syst. Conf. MilCIS 2017 - Proc.*, 2017. <https://doi.org/10.1109/MilCIS.2017.8190421>.
- [93] N. Moustafa, J. Slay, The evaluation of Network Anomaly Detection Systems: Statistical analysis of the UNSW-NB15 data set and the comparison with the KDD99 data set, *Inf. Secur. J.* (2016). <https://doi.org/10.1080/19393555.2015.1125974>.

- [94] K. Alrawashdeh, C. Purdy, Toward an online anomaly intrusion detection system based on deep learning, in: Proc. - 2016 15th IEEE Int. Conf. Mach. Learn. Appl. ICMLA 2016, 2017. <https://doi.org/10.1109/ICMLA.2016.167>.
- [95] H. Om, A. Kundu, A hybrid system for reducing the false alarm rate of anomaly intrusion detection system, in: 2012 1st Int. Conf. Recent Adv. Inf. Technol. RAIT-2012, 2012. <https://doi.org/10.1109/RAIT.2012.6194493>.
- [96] S. Naseer, Y. Saleem, S. Khalid, M.K. Bashir, J. Han, M.M. Iqbal, K. Han, Enhanced network anomaly detection based on deep neural networks, IEEE Access. (2018). <https://doi.org/10.1109/ACCESS.2018.2863036>.
- [97] T.A. Tang, L. Mhamdi, D. McLernon, S.A.R. Zaidi, M. Ghogho, Deep learning approach for Network Intrusion Detection in Software Defined Networking, in: Proc. - 2016 Int. Conf. Wirel. Networks Mob. Commun. WINCOM 2016 Green Commun. Netw., 2016. <https://doi.org/10.1109/WINCOM.2016.7777224>.
- [98] W. Wang, Y. Sheng, J. Wang, X. Zeng, X. Ye, Y. Huang, M. Zhu, HAST-IDS: Learning Hierarchical Spatial-Temporal Features Using Deep Neural Networks to Improve Intrusion Detection, IEEE Access. (2017). <https://doi.org/10.1109/ACCESS.2017.2780250>.
- [99] E.K. Viegas, A.O. Santin, L.S. Oliveira, Toward a reliable anomaly-based intrusion detection in real-world environments, Comput. Networks. (2017). <https://doi.org/10.1016/j.comnet.2017.08.013>.
- [100] R. Abdulhammed, M. Faezipour, A. Abuzneid, A. Abumallouh, Deep and Machine Learning Approaches for Anomaly-Based Intrusion Detection of Imbalanced Network Traffic, IEEE Sensors Lett. (2019). <https://doi.org/10.1109/LENS.2018.2879990>.
- [101] T.A. Tang, L. Mhamdi, D. McLernon, S.A.R. Zaidi, M. Ghogho, F. El Moussa, DeepIDS: Deep learning approach for intrusion detection in software defined networking, Electron. (2020). <https://doi.org/10.3390/electronics9091533>.
- [102] M. Al-Qatf, Y. Lasheng, M. Al-Habib, K. Al-Sabahi, Deep Learning Approach Combining Sparse Autoencoder with SVM for Network Intrusion Detection, IEEE Access. (2018). <https://doi.org/10.1109/ACCESS.2018.2869577>.
- [103] I. Ahmad, M. Basher, M.J. Iqbal, A. Rahim, Performance Comparison of Support Vector Machine, Random Forest, and Extreme Learning Machine for Intrusion Detection, IEEE Access. (2018). <https://doi.org/10.1109/ACCESS.2018.2841987>.
- [104] R. Singh, H. Kumar, R.K. Singla, An intrusion detection system using network traffic profiling and online sequential extreme learning machine, Expert Syst. Appl. (2015). <https://doi.org/10.1016/j.eswa.2015.07.015>.
- [105] M.R. Gauthama Raman, N. Somu, K. Kirthivasan, R. Liscano, V.S. Shankar Sriram, An efficient intrusion detection system based on hypergraph - Genetic algorithm for parameter optimization and feature selection in support vector machine, Knowledge-Based Syst. (2017). <https://doi.org/10.1016/j.knosys.2017.07.005>.
- [106] P. Tao, Z. Sun, Z. Sun, An Improved Intrusion Detection Algorithm Based on GA and SVM, IEEE Access. (2018). <https://doi.org/10.1109/ACCESS.2018.2810198>.
- [107] S. Huang, K. Lei, IGAN-IDS: An imbalanced generative adversarial network towards intrusion detection system in ad-hoc networks, Ad Hoc Networks. (2020). <https://doi.org/10.1016/j.adhoc.2020.102177>.
- [108] A. Ali Shah, M. Khuram Ehsan, K. Ishaq, Z. Ali, M. Shoaib Farooq, An Efficient Hybrid Classifier Model for Anomaly Intrusion Detection System, IJCSNS Int. J. Comput. Sci. Netw. Secur. (2018).
- [109] S. Dwivedi, M. Vardhan, S. Tripathi, A.K. Shukla, Implementation of adaptive scheme in evolutionary technique for anomaly-based intrusion detection, Evol. Intell. (2020). <https://doi.org/10.1007/s12065-019-00293-8>.
- [110] W.L. Al-Yaseen, Improving intrusion detection system by developing feature selection model based on firefly algorithm and support vector machine, IAENG Int. J. Comput. Sci. (2019).
- [111] O. Almomani, A feature selection model for network intrusion detection system based on pso, gwo,

- ffa and ga algorithms, *Symmetry (Basel)*. (2020). <https://doi.org/10.3390/sym12061046>.
- [112] T. Gu, H. Chen, L. Chang, L. Li, Intrusion detection system based on improved abc algorithm with tabu search, *IEEE Trans. Electr. Electron. Eng.* (2019). <https://doi.org/10.1002/tee.22987>.
 - [113] S. Aljawarneh, M. Aldwairi, M.B. Yassein, Anomaly-based intrusion detection system through feature selection analysis and building hybrid efficient model, *J. Comput. Sci.* (2018). <https://doi.org/10.1016/j.jocs.2017.03.006>.
 - [114] B.S. Arasu, B.J.B. Seelan, N. Thamaraiselvan, A machine learning-based approach to enhancing social media marketing, *Comput. Electr. Eng.* (2020). <https://doi.org/10.1016/j.compeleceng.2020.106723>.
 - [115] X. Xu, J. Wang, H. Peng, R. Wu, Prediction of academic performance associated with internet usage behaviors using machine learning algorithms, *Comput. Human Behav.* (2019). <https://doi.org/10.1016/j.chb.2019.04.015>.
 - [116] A. Devaraj, D. Murthy, A. Dontula, Machine-learning methods for identifying social media-based requests for urgent help during hurricanes, *Int. J. Disaster Risk Reduct.* (2020). <https://doi.org/10.1016/j.ijdr.2020.101757>.
 - [117] C.H. Ngejane, J.H.P. Eloff, T.J. Sefara, V.N. Marivate, Forensic Science International : Digital Investigation Digital forensics supported by machine learning for the detection of online sexual predatory chats, *Forensic Sci. Int. Digit. Investig.* 36 (2021) 301109. <https://doi.org/10.1016/j.fsidi.2021.301109>.
 - [118] S.H. Chin, C. Lu, P.T. Ho, Y.F. Shiao, T.J. Wu, Commodity anti-counterfeiting decision in e-commerce trade based on machine learning and Internet of Things, *Comput. Stand. Interfaces.* (2021). <https://doi.org/10.1016/j.csi.2020.103504>.
 - [119] D. Koehn, S. Lessmann, M. Schaal, Predicting online shopping behaviour from clickstream data using deep learning, *Expert Syst. Appl.* (2020). <https://doi.org/10.1016/j.eswa.2020.113342>.
 - [120] A.A. Mubarak, H. Cao, W. Zhang, W. Zhang, Visual analytics of video-clickstream data and prediction of learners' performance using deep learning models in MOOCs' courses, *Comput. Appl. Eng. Educ.* (2020). <https://doi.org/10.1002/cae.22328>.
 - [121] B. Mokhtar, M. Eltoweissy, Big data and semantics management system for computer networks, *Ad Hoc Networks.* (2017). <https://doi.org/10.1016/j.adhoc.2016.06.013>.
 - [122] X. Li, P. Lu, L. Hu, X.G. Wang, L. Lu, A novel self-learning semi-supervised deep learning network to detect fake news on social media, *Multimed. Tools Appl.* (2021). <https://doi.org/10.1007/s11042-021-11065-x>.
 - [123] R.K. Kaliyar, A. Goswami, P. Narang, FakeBERT: Fake news detection in social media with a BERT-based deep learning approach, *Multimed. Tools Appl.* (2021). <https://doi.org/10.1007/s11042-020-10183-2>.
 - [124] S. Hakak, M. Alazab, S. Khan, T.R. Gadekallu, P.K.R. Maddikunta, W.Z. Khan, An ensemble machine learning approach through effective feature extraction to classify fake news, *Futur. Gener. Comput. Syst.* (2021). <https://doi.org/10.1016/j.future.2020.11.022>.
 - [125] A. Agarwal, A. Dixit, Fake News Detection: An Ensemble Learning Approach, in: *Proc. Int. Conf. Intell. Comput. Control Syst. ICICCS 2020*, 2020, 2020. <https://doi.org/10.1109/ICICCS48265.2020.9121030>.
 - [126] A. Mulahuwaish, K. Gyorick, K.Z. Ghafoor, H.S. Maghdid, D.B. Rawat, Efficient classification model of web news documents using machine learning algorithms for accurate information, *Comput. Secur.* (2020). <https://doi.org/10.1016/j.cose.2020.102006>.
 - [127] A.I.E. Hosni, K. Li, Minimizing the influence of rumors during breaking news events in online social networks, *Knowledge-Based Syst.* (2020). <https://doi.org/10.1016/j.knosys.2019.105452>.
 - [128] Y. Wu, Y. Fang, S. Shang, J. Jin, L. Wei, H. Wang, A novel framework for detecting social bots with deep neural networks and active learning, *Knowledge-Based Syst.* (2021). <https://doi.org/10.1016/j.knosys.2020.106525>.
 - [129] L. Shang, D. (Yue) Zhang, M. Wang, S. Lai, D. Wang, Towards reliable online clickbait video

- detection: A content-agnostic approach, *Knowledge-Based Syst.* (2019). <https://doi.org/10.1016/j.knosys.2019.07.022>.
- [130] A. Kumar, N. Sachdeva, A Bi-GRU with attention and CapsNet hybrid model for cyberbullying detection on social media, *World Wide Web.* (2021). <https://doi.org/10.1007/s11280-021-00920-4>.
 - [131] H. Fotouhi, A. Čaušević, K. Lundqvist, M. Björkman, Communication and Security in Health Monitoring Systems - A Review, in: *Proc. - Int. Comput. Softw. Appl. Conf.*, 2016. <https://doi.org/10.1109/COMPSAC.2016.8>.
 - [132] A.L. Buczak, E. Guven, A Survey of Data Mining and Machine Learning Methods for Cyber Security Intrusion Detection, *IEEE Commun. Surv. Tutorials.* (2016). <https://doi.org/10.1109/COMST.2015.2494502>.
 - [133] A. Antony Viswasa Rani, E. Baburaj, Secure and intelligent architecture for cloud-based healthcare applications in wireless body sensor networks, *Int. J. Biomed. Eng. Technol.* (2019). <https://doi.org/10.1504/IJBET.2019.097305>.
 - [134] A. Alabdulatif, I. Khalil, A.R.M. Forkan, M. Atiquzzaman, Real-Time Secure Health Surveillance for Smarter Health Communities, *IEEE Commun. Mag.* 57 (2019) 122–129. <https://doi.org/10.1109/MCOM.2017.1700547>.
 - [135] S. Chakraborty, S. Aich, H.C. Kim, A Secure Healthcare System Design Framework using Blockchain Technology, in: *Int. Conf. Adv. Commun. Technol. ICACT*, 2019. <https://doi.org/10.23919/ICACT.2019.8701983>.
 - [136] A. Mathews, What can machine learning do for information security?, *Netw. Secur.* (2019). [https://doi.org/10.1016/S1353-4858\(19\)30050-9](https://doi.org/10.1016/S1353-4858(19)30050-9).
 - [137] M.A.M.S. Omar Zakaria, N.W. Ahmad, M.A. Zaidi, A classification method for data mining using SVM-weight and Euclidean distance, *Aust. J. Basic Appl. Sci.* 5 (2011) 2053–2059.
 - [138] N.R. Sabar, X. Yi, A. Song, A Bi-objective Hyper-Heuristic Support Vector Machines for Big Data Cyber-Security, *IEEE Access.* 6 (2018) 10421–10431. <https://doi.org/10.1109/ACCESS.2018.2801792>.
 - [139] M.I. Gürsoy, A. Subaşı, DVM ile EEG işaretlerinin sınıflandırılmasında TBA, BBA ve DAA'nin performansının karşılaştırılması, in: *2008 IEEE 16th Signal Process. Commun. Appl. Conf. SIU*, 2008. <https://doi.org/10.1109/SIU.2008.4632748>.
 - [140] V. Jakkula, Tutorial on Support Vector Machine (SVM), *Sch. EECS, Washingt. State Univ.* (2011) 1–13.
 - [141] S.H. Bouazza, N. Hamdi, A. Zeroual, K. Auhmani, Gene-expression-based cancer classification through feature selection with KNN and SVM classifiers, in: *2015 Intell. Syst. Comput. Vision, ISCV 2015*, 2015. <https://doi.org/10.1109/ISACV.2015.7106168>.
 - [142] O. Yaman, F. Ertam, T. Tuncer, Automated Parkinson's disease recognition based on statistical pooling method using acoustic features, *Med. Hypotheses.* 135 (2020). <https://doi.org/10.1016/j.mehy.2019.109483>.
 - [143] A Comparative Study of Classification Techniques in Data Mining Algorithms, *Int. J. Mod. Trends Eng. Res.* 4 (2017) 58–63. <https://doi.org/10.21884/ijmter.2017.4211.vxayk>.
 - [144] C.E. Brodley, M.A. Friedl, Decision tree classification of land cover from remotely sensed data, *Remote Sens. Environ.* 61 (1997) 399–409. [https://doi.org/10.1016/S0034-4257\(97\)00049-7](https://doi.org/10.1016/S0034-4257(97)00049-7).
 - [145] V.Y. Kulkarni, P.K. Sinha, Pruning of random forest classifiers: A survey and future directions, in: *Proc. - 2012 Int. Conf. Data Sci. Eng. ICDSE 2012*, 2012. <https://doi.org/10.1109/ICDSE.2012.6282329>.
 - [146] E.E. Okoro, T. Obomanu, S.E. Sanni, D.I. Olatunji, P. Igbinedion, Application of artificial intelligence in predicting the dynamics of bottom hole pressure for under-balanced drilling: Extra tree compared with feed forward neural network model, *Petroleum.* (2021). <https://doi.org/10.1016/j.petlm.2021.03.001>.
 - [147] M.T. Uddin, M.A. Uddiny, Human activity recognition from wearable sensors using extremely randomized trees, in: *2nd Int. Conf. Electr. Eng. Inf. Commun. Technol. ICEEICT 2015*, 2015.

- <https://doi.org/10.1109/ICEEICT.2015.7307384>.
- [148] J.R. Quinlan, Bagging, boosting, and C4.5, in: Proc. Natl. Conf. Artif. Intell., 1996.
 - [149] R.E. Schapire, Explaining adaboost, in: Empir. Inference Festschrift Honor Vladimir N. Vapnik, 2013. https://doi.org/10.1007/978-3-642-41136-6_5.
 - [150] A. Yulianto, P. Sukarno, N.A. Suwastika, Improving AdaBoost-based Intrusion Detection System (IDS) Performance on CIC IDS 2017 Dataset, in: J. Phys. Conf. Ser., 2019. <https://doi.org/10.1088/1742-6596/1192/1/012018>.
 - [151] T.H. Tran, N.D. Hoang, Predicting algal appearance on mortar surface with ensembles of adaptive neuro fuzzy models: a comparative study of ensemble strategies, Int. J. Mach. Learn. Cybern. (2019). <https://doi.org/10.1007/s13042-018-0846-1>.
 - [152] Y. M. Banadaki, Detecting Malicious DNS over HTTPS Traffic in Domain Name System using Machine Learning Classifiers, J. Comput. Sci. Appl. 8 (2020) 46–55. <https://doi.org/10.12691/jcsa-8-2-2>.
 - [153] J.H. Friedman, Greedy function approximation: A gradient boosting machine, Ann. Stat. (2001). <https://doi.org/10.1214/aos/1013203451>.
 - [154] I. Baturynska, K. Martinsen, Prediction of geometry deviations in additive manufactured parts: comparison of linear regression with machine learning algorithms, J. Intell. Manuf. (2021). <https://doi.org/10.1007/s10845-020-01567-0>.
 - [155] S.S. Dhaliwal, A. Al Nahid, R. Abbas, Effective intrusion detection system using XGBoost, Inf. 9 (2018). <https://doi.org/10.3390/info9070149>.
 - [156] Z. Chen, F. Jiang, Y. Cheng, X. Gu, W. Liu, J. Peng, XGBoost Classifier for DDoS Attack Detection and Analysis in SDN-Based Cloud, in: Proc. - 2018 IEEE Int. Conf. Big Data Smart Comput. BigComp 2018, 2018. <https://doi.org/10.1109/BigComp.2018.00044>.
 - [157] Z. Salam Patrous, Evaluating XGBoost for User Classification by using Behavioral Features aExtracted from Smartphone Sensors, (2018) 67. <https://www.diva-portal.org/smash/record.jsf?pid=diva2%3A1240595&dsid=-6444>.
 - [158] D. Zhang, L. Qian, B. Mao, C. Huang, B. Huang, Y. Si, A Data-Driven Design for Fault Detection of Wind Turbines Using Random Forests and XGboost, IEEE Access. (2018). <https://doi.org/10.1109/ACCESS.2018.2818678>.
 - [159] Z. Zhang, Y. Zhang, D. Guo, M. Song, A scalable network intrusion detection system towards detecting, discovering, and learning unknown attacks, Int. J. Mach. Learn. Cybern. (2021). <https://doi.org/10.1007/s13042-020-01264-7>.
 - [160] E. Al Daoud, Comparison between XGBoost, LightGBM and CatBoost Using a Home Credit Dataset, Int. J. Comput. Inf. Eng. (2019).
 - [161] A.V. Dorogush, V. Ershov, A. Gulin, CatBoost: Gradient boosting with categorical features support, ArXiv. (2018).
 - [162] Beginner's Guide to Feature Selection in Python, (n.d.). <https://www.datacamp.com/community/tutorials/feature-selection-python> (accessed February 4, 2022).
 - [163] T.H. Jaya Hidayat, Y. Ruldeviyani, A.R. Aditama, G.R. Madya, A.W. Nugraha, M.W. Adisaputra, Sentiment analysis of twitter data related to Rinca Island development using Doc2Vec and SVM and logistic regression as classifier, Procedia Comput. Sci. 197 (2022) 660–667. <https://doi.org/10.1016/j.procs.2021.12.187>.
 - [164] X. Jin, J. Guo, Y. Shen, X. Liu, C. Zhao, Application of singular spectrum analysis and multilayer perceptron in the mid-long-term polar motion prediction, Adv. Sp. Res. (2021). <https://doi.org/10.1016/j.asr.2021.06.039>.
 - [165] M. Govindarajan, R. Chandrasekaran, Intrusion detection using neural based hybrid classification methods, Comput. Networks. (2011). <https://doi.org/10.1016/j.comnet.2010.12.008>.
 - [166] F. Ertam, Kurumsal Bilgisayar Ağlarındaki Trafik Bilgisinin Akıllı Sistemler ile Sınıflandırılması,

Fırat Üniversitesi, (2016).

- [167] CCIdd_Dataset, (n.d.). <https://github.com/ifkilincer/CCiDD-Datasets> (accessed June 14, 2022).
- [168] A.H. Lashkari, G.D. Gil, M.S.I. Mamun, A.A. Ghorbani, Characterization of tor traffic using time based features, in: ICISSP 2017 - Proc. 3rd Int. Conf. Inf. Syst. Secur. Priv., 2017. <https://doi.org/10.5220/0006105602530262>.
- [169] C. Khammassi, S. Krichen, A GA-LR wrapper approach for feature selection in network intrusion detection, *Comput. Secur.* (2017). <https://doi.org/10.1016/j.cose.2017.06.005>.
- [170] Z.B. Szuts, J.R. Blundell, M.P. Chidichimo, J. Marotzke, A vertical-mode decomposition to investigate low-frequency internal motion across the Atlantic at 26 N, *Ocean Sci.* (2012). <https://doi.org/10.5194/os-8-345-2012>.
- [171] C.M. Wu, Y.C. Chen, Statistical feature matrix for texture analysis, *CVGIP Graph. Model. Image Process.* (1992). [https://doi.org/10.1016/1049-9652\(92\)90025-S](https://doi.org/10.1016/1049-9652(92)90025-S).
- [172] H. Wang, W. Wang, S. Xiao, Z. Cui, M. Xu, X. Zhou, Improving artificial Bee colony algorithm using a new neighborhood selection mechanism, *Inf. Sci. (Ny)*. (2020). <https://doi.org/10.1016/j.ins.2020.03.064>.
- [173] T. Tuncer, A new stable nonlinear textural feature extraction method based EEG signal classification method using substitution Box of the Hamsi hash function: Hamsi pattern, *Appl. Acoust.* (2021). <https://doi.org/10.1016/j.apacoust.2020.107607>.
- [174] T. Tuncer, S. Dogan, F. Ozyurt, S.B. Belhaouari, H. Bensmail, Novel multi center and threshold ternary pattern based method for disease detection method using voice, *IEEE Access.* (2020). <https://doi.org/10.1109/ACCESS.2020.2992641>.
- [175] D. Chicco, M.J. Warrens, G. Jurman, The Matthews Correlation Coefficient (MCC) is More Informative Than Cohen's Kappa and Brier Score in Binary Classification Assessment, *IEEE Access.* (2021). <https://doi.org/10.1109/ACCESS.2021.3084050>.
- [176] I.F. Kilincer, IoBe_Dataset, (n.d.). https://github.com/ifkilincer/IoBe_Dataset (accessed June 14, 2022).
- [177] CICFlowMeter, (n.d.). <https://www.unb.ca/cic/research/applications.html> (accessed November 16, 2020).
- [178] CIC-Darknet2020, (n.d.). <https://www.unb.ca/cic/datasets/darknet2020.html> (accessed March 16, 2021).
- [179] S.A. Aswad, E. Sonuc, Classification of VPN Network Traffic Flow Using Time Related Features on Apache Spark, in: 4th Int. Symp. Multidiscip. Stud. Innov. Technol. ISMSIT 2020 - Proc., 2020. <https://doi.org/10.1109/ISMSIT50672.2020.9254893>.
- [180] F. Xiao, D. Yang, Z. Lv, X. Guo, Z. Liu, Y. Wang, Classification of hand movements using variational mode decomposition and composite permutation entropy index with surface electromyogram signals, *Futur. Gener. Comput. Syst.* (2020). <https://doi.org/10.1016/j.future.2019.11.025>.
- [181] I. Kononenko, M. Robnik-Šikonja, U. Pompe, ReliefF for estimation and discretization of attributes in classification, regression, and ILP problems, in: *Artif. Intell. Methodol. Syst. Appl.*, 1996.
- [182] K. Kira, L.A. Rendell, Feature selection problem: traditional methods and a new algorithm, in: *Proc. Tenth Natl. Conf. Artif. Intell.*, 1992.
- [183] Y. Chen, J. Fan, Z. Deng, B. Du, X. Huang, Q. Gui, PR-KELM: Icing level prediction for transmission lines in smart grid, *Futur. Gener. Comput. Syst.* (2020). <https://doi.org/10.1016/j.future.2019.08.002>.
- [184] S. Nandy, M. Adhikari, M.A. Khan, V.G. Menon, S. Verma, An Intrusion Detection Mechanism for Secured IoMT framework based on Swarm-Neural Network, *IEEE J. Biomed. Heal. Informatics.* (2021). <https://doi.org/10.1109/JBHI.2021.3101686>.
- [185] P. Kumar, G.P. Gupta, R. Tripathi, An ensemble learning and fog-cloud architecture-driven cyber-attack detection framework for IoMT networks, *Comput. Commun.* (2021).

<https://doi.org/10.1016/j.comcom.2020.12.003>.

- [186] G. Zachos, I. Essop, G. Mantas, K. Porfyraakis, J.C. Ribeiro, J. Rodriguez, An anomaly-based intrusion detection system for internet of medical things networks, *Electron.* (2021). <https://doi.org/10.3390/electronics10212562>.
- [187] S. Ghazanfar, F. Hussain, A.U. Rehman, U.U. Fayyaz, F. Shahzad, G.A. Shah, IoT-Flock: An Open-source Framework for IoT Traffic Generation, in: 2020 Int. Conf. Emerg. Trends Smart Technol. ICETST 2020, 2020. <https://doi.org/10.1109/ICETST49965.2020.9080732>.



ÖZGEÇMİŞ

İlhan Fırat KILINÇER

Category	Value
Category 1	Value 1
Category 2	Value 2
Category 3	Value 3
Category 4	Value 4
Category 5	Value 5
Category 6	Value 6
Category 7	Value 7

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED] [REDACTED] [REDACTED] [REDACTED] [REDACTED] [REDACTED] [REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]
 [REDACTED]
 [REDACTED]

■ [REDACTED]

[REDACTED]

[REDACTED] [REDACTED]
[REDACTED] [REDACTED]
[REDACTED] [REDACTED]
[REDACTED]

AKADEMİK FAALİYETLER

Makaleler:

1. Kilincer, I. F., Ertam, F., & Sengur, A. (2021). Machine learning methods for cyber security intrusion detection: Datasets and comparative study. *Computer Networks*, 188, 107840.
2. Kilincer, I. F., Ertam, F., & Sengur, A. (2022). A comprehensive intrusion detection framework using boosting algorithms. *Computers & Electrical Engineering*, 100, 107869.