
GAZİ ÜNİVERSİTESİ
Gazi
1926

ŞİFRELENMİŞ İNTERNET TRAFİĞİNİN MAKİNE ÖĞRENMESİ YAKLAŞIMI İLE SINIFLANDIRILMASI

Mesut UĞURLU

YÜKSEK LİSANS TEZİ
BİLGİ GÜVENLİĞİ MÜHENDİSLİĞİ ANA BİLİM DALI

GAZİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

ARALIK 2020

Mesut UĞURLU tarafından hazırlanan "ŞİFRELENMİŞ İNTERNET TRAFİĞİNİN MAKİNE ÖĞRENMESİ YAKLAŞIMI İLE SINIFLANDIRILMASI" adlı tez çalışması aşağıdaki jüri tarafından OY BİRLİĞİ ile Gazi Üniversitesi Bilgi Güvenliği Mühendisliği Ana Bilim Dalında YÜKSEK LİSANS TEZİ olarak kabul edilmiştir.

Danışman: Doç. Dr. İbrahim Alper DOĞRU

Bilgi Güvenliği Mühendisliği Ana Bilim Dalı, Gazi Üniversitesi

Bu tezin, kapsam ve kalite olarak Yüksek Lisans Tezi olduğunu onaylıyorum. ……..……

Başkan: Doç. Dr. Ahmet Murat ÖZBAYOĞLU

Yapay Zeka Mühendisliği Ana Bilim Dalı, TOBB Ekonomi ve Teknoloji Üniversitesi

Bu tezin, kapsam ve kalite olarak Yüksek Lisans Tezi olduğunu onaylıyorum. ……..……

Üye: Dr. Öğr. Üyesi Mehmet DEMİRCİ

Bilgisayar Mühendisliği Ana Bilim Dalı, Gazi Üniversitesi

Bu tezin, kapsam ve kalite olarak Yüksek Lisans Tezi olduğunu onaylıyorum. ……..……

Tez Savunma Tarihi: 31/12/2020

Jüri tarafından kabul edilen bu tezin Yüksek Lisans Tezi olması için gerekli şartları yerine getirdiğini onaylıyorum.

……..…

Prof. Dr. Cevriye GENCER

Fen Bilimleri Enstitüsü Müdürü

ETİK BEYAN

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

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

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

Mesut UĞURLU
31/12/2020

ŞİFRELENMİŞ İNTERNET TRAFİĞİNİN MAKİNE ÖĞRENMESİ YAKLAŞIMI İLE SINIFLANDIRILMASI

(Yüksek Lisans Tezi)

Mesut UĞURLU

GAZİ ÜNİVERSİTESİ

FEN BİLİMLERİ ENSTİTÜSÜ

Aralık 2020

ÖZET

İnternet kullanım oranı dünya genelinde %62'inin üzerindedir ve bu oran günden güne artmaktadır. Bu artış ile birlikte internet üzerinden akan trafikteki bilgilerin gizliliğini sağlamak önem kazanmaktadır. Bunun için şifreleme algoritmaları ve protokoller kullanılmaktadır. Kullanıcılar için faydalı olan bu durum saldırganlar tarafından gizlenmek amaçlı da kullanılmaktadır. Saldırganlar şifrelenmiş trafik ile IDS/IPS ve antivirüs sistemlerini atlatabilme yeteneği kazanmaktadırlar. Şifrelenmiş trafiğin deşifrelenmesi işlemi yapılmadan içerik analizi yapılamadığı için mevcut ticari güvenlik çözümleri bu durum karşısında yetersiz kalmaktadır. Bu çalışmada XGBoost, Karar Ağacı ve Rassal Orman sınıflandırma algoritmaları kullanılarak şifrelenmiş paketler üzerinden giden-gelen veriler analiz edilerek trafiğin sınıflandırılması amaçlanmıştır. Bu sayede deşifreleme yapılmadan sadece akan trafik üzerinde gelen giden paketlerin boyut, süre gibi bazı meta verileri kullanılarak sınıflandırılması ile ağ uzmanları ve siber güvenlik uzmanlarının ağ üzerindeki analiz yetenekleri artırılarak siber saldırıların tespiti ve saldırılara karşı önlem alınması mümkün olmaktadır. Bu çalışmada önerilen modelin test edilmesi için ISCX VPN-nonVPN veri seti kullanılmıştır. Oluşturulan yapı ile şifreli paketler yüksek başarı oranı ile sınıflandırılmış ve XGBoost sınıflandırma metodu kullanılarak %94,53 başarı yakalanmıştır.

Bilim Kodu	:	92403
Anahtar Kelimeler	:	Trafik sınıflandırılması, şifreli trafik tanımlama, makine öğrenmesi, siber güvenlik
Sayfa Adedi	:	66
Danışman	:	Doç. Dr. İbrahim Alper DOĞRU
İkinci Danışman	:	Dr. Öğr. Üyesi Recep Sinan ARSLAN

CLASSIFICATION OF ENCRYPTED INTERNET TRAFFIC USING MACHINE LEARNING APPROACH

(M. Sc. Thesis)

Mesut UĞURLU

GAZİ UNIVERSITY

GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

December 2020

ABSTRACT

The rate of internet usage in the world is over 62% and this rate is increasing day by day. With this increase, it becomes important to ensure the confidentiality of the information in the traffic flowing over the internet. Encryption algorithms and protocols are used for this purpose. This situation, which is beneficial for normal users, is also used by attackers to hide. Cyber attackers or hackers gain the ability to bypass security precautions such as IDS/IPS and antivirus systems with using encrypted traffic. Since payload analysis cannot be performed without deciphering the encrypted traffic, existing commercial security solutions fall short in this situation. In this study, it is aimed to classify the network traffic by analysing the outgoing and incoming data over the encrypted traffic using Extreme Gradient Boosting (XGBoost), Decision Tree and Random Forest classification methods. In this way, it is possible to detect cyber attacks and take measures against attacks by increasing the analysis capabilities of network experts and cyber security experts on the network by classifying the encrypted traffic using only some metadata such as the size and duration of incoming and outgoing packets on the flowing traffic without deciphering. ISCX VPN-NonVPN dataset was used to test the proposed model in this study. With the created framework, encrypted traffic was classified with a high success rate and 94,53% success was achieved by using the XGBoost classification method.

Science Code	:	92403
Key Words	:	Traffic classification, encrypted traffic identification, machine learning, cyber security
Page Number	:	66
Supervisor	:	Assoc. Prof. Dr. İbrahim Alper DOĞRU
Co-Supervisor	:	Assist. Prof. Dr. Recep Sinan ARSLAN

TEŞEKKÜR

Bu çalışmanın her aşamasında kıymetli katkıları ve değerli eleştirileri ile bana yol gösteren değerli danışmanlarım sayın Doç. Dr. İbrahim Alper DOĞRU ve Dr. Öğr. Üyesi Recep Sinan ARSLAN'a içtenlikle teşekkür ederim.

Maddi ve manevi desteklerini hiçbir zaman esirgemeyen ve beni hiçbir zaman yalnız bırakmayan aileme ve eşim Cemile UĞURLU'ya teşekkür ederim.

İÇİNDEKİLER

Sayfa

Sayfa

ÇİZELGELERİN LİSTESİ

Çizelge **Sayfa**

ŞEKİLLERİN LİSTESİ

Şekil **Sayfa**

Şekil **Sayfa**

SİMGELER VE KISALTMALAR

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

Kısaltmalar	Açıklamalar
CNN	Convolutional Neural Network
DARPA	Defense Advanced Research Projects Agency
DPI	Deep Packet Inspection
FT	File Transfer
FTP	File Transfer Protocol
FTPS	FTP over SSL
HTTP	Hypertext Transfer Protocol
IANA	Internet Assigned Numbers Authority
ICMP	Internet Control Message Protocol
IKE	Internet Key Exchange
IMAP	Internet Message Access Protocol
IP	Internet Protocol
IPsec	Internet Protocol Security
kNN	K-Nearest Neighbors
LSTM	Long Short-Term Memory
MAC	Message Authentication Code
MLP	Multilayer Perceptron
MLTAT	Machine Learning Traffic Analytics Tool
OSI	Open Systems Interconnection
P2P	Peer to Peer
POP	Post Office Protocol
QoS	Quality of Service
RBFN	Radial Basis Function Network
RIPPER	Repeated Incremental Pruning to Produce Error
SCP	Secure Copy
SFTP	Secure File Transfer Protocol
SMOTE	Synthetic Minority Oversampling Technique

Kısaltmalar	Açıklamalar
SMTP	Simple Mail Transfer Protocol
SMTPS	Simple Mail Transfer Protocol Secure
SSH	Secure Shell
SSL	Secure Sockets Layer
STNN	Stereo Transform Neural Network
SVM	Support Vector Machine
TCP	Transmission Control Protocol
TLS	Transport Layer Security
ToR	The Onion Router
UDP	User Datagram Protocol
VoIP	Voice Over Internet Protocol
VPN	Virtual Private Network
WebRTC	Web Real Time Communication
XGBoost	Extreme Gradient Boosting

1. GİRİŞ

İnternetin kullanım oranının yaygınlaşması ile birlikte siber saldırılar da çok büyük oranda artmaktadır. Özellikle internet kullanımında gizliliğin öneminin artması ile birlikte bilgi güvenliğinin de önemi artmaktadır. İnternet üzerinden kişisel, ticari ve askeri bilgiler kişi veya kurumlar arasında paylaşılmaktadır. Paylaşılan bu bilgilerin gizliliğinin sağlanması için şifreleme yöntem ve protokolleri kullanılmaktadır. İnternet üzerinde Güvenli Hiper Metin Transfer Protokolü (Hypertext Transfer Protocol Secure, HTTPS) trafik kullanım oranı %95 üzerindedir ve giderek artmaktadır [1]. Şifrelemenin olmadığı durumda kritik veriler hattı dinleyen siber saldırganlar tarafından ele geçirilebilmektedir ve kötü amaçlar için kullanılabilmektedir [2].

Ağ trafiğinin şifrelenmesi ve güvenli iletişim sağlanması amacı ile en çok kullanılan yöntem ve protokoller Özel Sanal Ağ (Virtual Private Network, VPN) [3] ve Güvenli Oturum Katmanı (Secure Sockets Layer, SSL) protokolleridir. SSL protokolü Netscape şirketi tarafından geliştirilmiş ve web tarayıcısı ile sunucu arasında güvenli iletişim kanalı kurulması amaçlı kullanılmaktadır [4]. Hem VPN hem de SSL protokolü güvenli iletişim sağlamak amacı ile kullanılmasına rağmen aralarında farklar bulunmaktadır. SSL şifreleme de paketin sadece içerik kısmı şifrelenirken, VPN'de tüm paket başka bir paket içerisine alınarak şifreleme işlemi gerçekleştirilir. VPN ile şifreleme işlemi, Open Systems Interconnection (OSI) [5] modelinin 3. katmanı olan ağ katmanında şifreleme işlemi yapmaktadır. SSL ise OSI sunum katmanı olan 6. katmanda çalışmaktadır. SSL çoğunlukla Hiper Metin Transfer Protokol (Hypertext Transfer Protocol, HTTP) trafiğinin şifrelenmesi amacı ile kullanılırken, VPN iki uç nokta arasındaki tüm trafiğin şifrelenmesinde yaygın olarak kullanılmaktadır [6].

Şifreleme faydalarına rağmen siber güvenlik uzmanları ve ağ uzmanlarının ağ üzerindeki trafik analiz kapasitesini azaltmakta ve siber saldırganların gizlenmesini sağlamaktadır [7]. Siber saldırganlar şifrelenmiş trafik kullanarak anti virüs, saldırı tespit sistemi, veri kaybı önleme sistemi ve uygulama güvenlik duvarı gibi güvenlik önlemlerinden gizlenerek saldırılarını yapabilmektedir [8]. Çünkü bu mevcut çözümler açık veri üzerinde analiz yapabilme yeteneklerine sahiptirler. Şifrelenmiş paketler deşifreleme yapılmadan derin paket analizi yapılamamaktadır ve trafiğin içinden geçen veriler analiz edilememektedir [9].

Saldırganlar bu sayede veri sızdırma, veri şifreleme, komuta ve kontrol merkezi ile iletişim, zararlı yazılım indirme ve diğer siber saldırı ve atakları yapabilmektedir. Yapılan analizlerde siber saldırganların gizlenebilmek için çok yüksek oranda şifreli trafiği tercih ettikleri görülmüştür [10]. Bu saldırılara önlem olarak şifrelenmiş paketler deşifreleme yapılarak analiz edilmesi gerekmektedir. Trafik boyutlarının yüksek olduğu ve deşifreleme-analiz-şifreleme süresinin uzun olması ile birlikte bu çözümün verimliliği düşmektedir. Bununla birlikte kullanıcıların şifreli trafiğinin deşifre edilmesi kullanıcı bilgilerinin gizliliği yasa ve politikalarını da ihlal etmektedir [11].

Ağ trafiğinin sınıflandırılması, bilinmeyen ağ sınıflarının tanımlanması ve sınıflandırılmasının ilk adımıdır. Ağ trafiğinin sınıflandırılması ağ yönetimi ve güvenliğinin sağlanmasında önemli bir rol oynar. Bu teknik sayesinde ağ uzmanları, ağ üzerinde izin vermedikleri veya istemedikleri trafikleri engelleyebilir. Ayrıca hangi uygulamanın ağ üzerinde ne kadar bant genişliği kullandığını tespit ederek ağ kaynak yönetimini de yapabilir.

1990'lı yılların başından itibaren belirli imza ve kalıplar kullanılarak şifrelenmemiş paketlerin sınıflandırılması yapılmıştır. Derin paket analizi yapabilmek için şifrelenmemiş paketlerin içeriğine ihtiyaç duyulmaktadır. Çünkü bilinen imza kalıpları ile yakalanan paketler içerisinde bulunan imza kalıplarının karşılaştırılması gerekmektedir. Paket içeriğinin şifreli olduğu durumlarda paket içeriği anlamsız karakterler içerdiği ve imza kalıplarına uymadığı için bu yöntemler kullanılamamaktadır. Bu durumlar için akış tabanlı sınıflandırma yöntemleri kullanılmaktadır. Akış ve zaman tabanlı sistemlerde paket içeriği yerine paket uzunluğu, paket sayısı, paket boyutu ve paket gidiş ve geliş zamanları gibi istatistiksel özellikler kullanılmaktadır. İnternet Protokolü (Internet Protocol, IP) ve port bilgileri şifrelenmiş trafik sınıflandırılmalarında özellik olarak kullanılmamaktadır. Çünkü şifreli trafik farklı portlar üzerinden kurulabilmektedir ve bu durum yapay zekâ temelli yaklaşımlarda başarı oranını düşürmektedir.

Deşifreleme yapmadan trafik üzerinde giden gelen paketlerin boyutları incelenerek trafik sınıflandırmasının yapılması alanında çalışmalar yapılmaktadır. Bu çalışmalarda yapay zekâ algoritmaları yoğun olarak tercih edilmektedir. Akış tabanlı yöntemler kullanılarak şifreli trafik üzerinden protokol ve uygulama tanımı yapılarak tanımlanamayan ağ trafikleri engellenebilmektedir. Özellikle virüs ve botnet cihazları komuta ve kontrol merkezi ile

şifreli iletişim kurdukları için trafik sınıflandırması ile bu uygulamalar tespit edilerek kontrol altında tutulabilmektedir.

Bu çalışmada, şifrelenmiş internet trafiğinin deşifrelenmeden çıkarılan özellikleri yardımı ile analiz edilmesi ve etiketlenmesi yapılmıştır. Bu sayede, ağ uzmanlarının ve siber güvenlik uzmanlarının analiz yetenekleri artırılarak güvenlik sistemlerinin güçlendirilmesi, bilgi sistemlerinin ve kullanıcıların güvenliğinin artırılmasının sağlanması amaçlanmıştır. Mevcut siber güvenlik çözümleri şifrelenmiş trafiği analiz edemediği için ağ uzmanları ve siber güvenlik uzmanları ağ üzerinden akan trafiği görememekte ve bu trafik üzerinden yapılan saldırılara karşı önlem alamamaktadır. Yapılan çalışma ile şifreli trafik sınıflandırılarak ağ uzmanları ve siber güvenlik uzmanlarının izin verdikleri trafik sınıfları haricinde bir sınıf için iletişim olduğunda alarm oluşturulması ve oluşturulacak korelasyonlar ile trafiğin engellenmesi mümkün olacaktır.

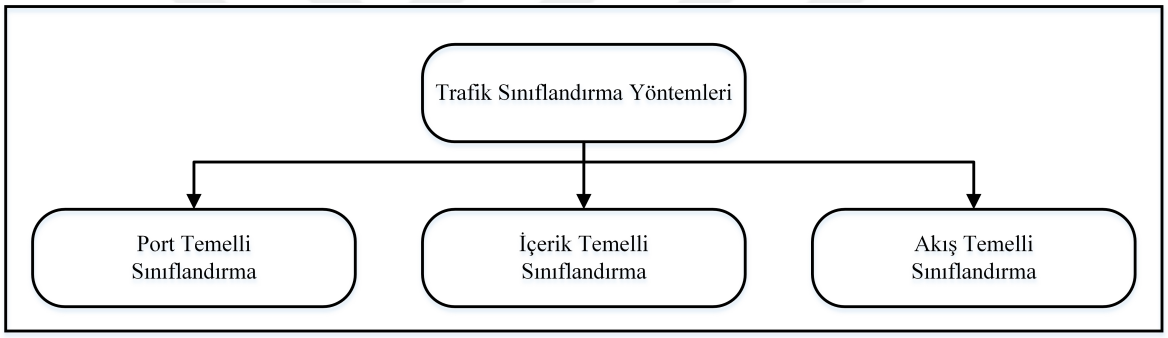
Yapılan çalışmada en büyük katkılarımız;

- Önerilen modelin test edilebilmesi için şifreli internet trafik verilerinin analizi ve düzenlenmesi,
- Elde edilen veri seti kullanılarak trafiğinin makine öğrenmesi teknikleri ile hem zaman hem de davranış tabanlı olarak sınıflandırılması,
- Farklı öğrenme algoritmaları kullanılarak problem çözümü üzerindeki etkisinin incelenmesidir.

Bu çalışmanın ikinci bölümünde trafik sınıflandırılmasında kullanılan yöntemler ve bu yöntemler kullanılarak yapılmış güncel çalışmalara yer verilmiştir. Üçüncü bölümde SSL ve VPN hakkında detaylı bilgilendirmelere, makine öğrenmesi yöntemlerine, çalışmada kullanılan veri setine ve performans metriklerine detaylı olarak yer verilmiş ve açıklanmıştır. Çalışma kapsamında kullanılan modele ait detaylar dördüncü bölümde sunulmuştur. Beşinci bölümde elde edilen veri seti ile önerilen modelin test edilmesi sonucunda elde edilen çıktılar detaylı olarak gösterilmiştir. Son bölümde ise çalışmanın genel bir değerlendirmesi yapılmış ve mevcut çalışmalar ile karşılaştırılmıştır. Ayrıca bu bölümde internet trafiğinin sınıflandırılmasına yönelik olarak gelecekte yapılabilecek çalışmalara ilişkin öneriler verilmiştir.

2. SINIFLANDIRMA YÖNTEMLERİ VE LİTERATÜR TARAMASI

Trafiğin sınıflandırılması ağ uzmanları ve siber güvenlik uzmanlarının ağ üzerinde analiz yapabilmeleri açısından oldukça önemlidir. Bu nedenle internetin kullanıma başlanmasından itibaren trafik sınıflandırma çalışmaları yapılmıştır. İlk başlarda port tabanlı sınıflandırma yapılırken daha sonraları belirli imza ve kalıplar kullanılarak şifrelenmemiş paketlerin içeriği incelenerek trafik sınıflandırılması yapılmıştır [12]. Paket içeriklerinin şifreli olduğu durumlarda ise akış tabanlı sınıflandırma yöntemleri kullanılmaktadır. Akış ve zaman tabanlı sistemlerde paket içeriği yerine paket uzunluğu, onay (acknowledge) paket sayısı ve paket gidiş ve geliş zamanları gibi özellikler kullanılmaktadır [13]. Şekil 2.1'de trafik sınıflandırma yöntemleri gösterilmiştir.



Şekil 2.1. Trafik sınıflandırma yöntemleri

2.1. Port Temelli Sınıflandırma Yöntemi

Port temelli sınıflandırma İletim Kontrol Protokolü (Transmission Control Protocol, TCP) ve Kullanıcı Veri Birimi Protokolü (User Datagram Protocol, UDP) port numaraları kullanılarak trafiğin sınıflandırılması amacı ile kullanılan en yoğun ve en eski yöntemdir [12]. Bu yöntemde paket başlığında bulunan port numarası bilgisine bakılarak sınıflandırma işlemi yapılır. Sınıflandırıcı port numarasını 80 gördüğünde HTTP, 22 gördüğünde Güvenli Kabuk (Secure Shell, SSH) veya 23 gördüğünde Telnet olarak sınıflandırır. Uygulamalar için port numaralar Internet Assigned Numbers Authority (IANA) tarafından verilmektedir [14].

Plonka [15] ağ trafiğini raporlayan ve görselleştiren FlowScan adlı bir araç geliştirmiştir. Bu araç ham trafik verisini toplayıp işleyebilme yeteneğine sahiptir. Trafiği yakaladıktan sonra IP üzerinden İnternet Kontrol Mesaj Protokolü (Internet Control Message Protocol, ICMP), TCP ve UDP olarak ayırabilmektedir. Bununla birlikte bilinen port numaralarına bakarak trafiği Dosya Transfer Protokol (File Transfer Protocol, FTP), Basit Posta Aktarım Protokolü (Simple Mail Transfer Protocol, SMTP) ve HTTP gibi kategorilere ayırabilmektedir. Ayrıca IP bilgisine bakarak yerel IP mi olup olmadığını anlayabilmektedir. Fakat araç trafiğin içeriğini yakalayamadığı için sadece port numarasına bakarak sınıflandırma yapabilmektedir. Bu da sınıflandırma güvenilirliğini düşürmektedir.

Moore [16] ve arkadaşları ağ uzmanlarının ağı izlemeleri ve sorunları çözebilmeleri için CoralReef adlı bir araç oluşturdular. Bu araç ağ üzerinden giden trafiği dinleyerek ağ üzerinden geçen gerçek trafiğin sınıflandırmasını ve ağ cihazlarının donanımsal performans bilgilerini ağ uzmanları tarafından görülebilmesini sağlamaktadır. CoralReef, AppPorts adlı protokol ve port bilgilerinin eşleştirildiği bir kütüphaneye sahiptir. Yakaladığı trafik üzerinden port bilgilerini alarak, bu kütüphane aracılığı ile sınıflandırma işlemi yapmaktadır.

Fraleigh [17] ve arkadaşları Sprint IP omurgasında trafik hesaplanmasının yapılabilmesi için gerçekleştirdikleri çalışmada port numaralarına bakarak uygulamaları sınıflandırmışlarıdır. Bu çalışmada benzer uygulamalar web, mail, dosya transferi, Eşten Eşe (Peer to Peer, P2P), Streaming gibi kategorilere göre sınıflandırılmıştır. Web bağlantıları portuna göre HTTP veya HTTPS olarak, mail SMTP veya Postane Protokolü (Post Office Protocol, POP3) olarak, dosya transferi ise Güvenli Kopyalama (Secure Copy, SCP) veya FTP olarak alt kategorilere ayrılabilmektedir. P2P bağlantıları farklı portları kullandığı için sınıflandırılamamış ve diğer adlı bir sınıf oluşturularak buraya koyulmuştur. Diğer kategorisi bilinmeyen portları kullanan uygulamaları içermektedir.

Sadece port numarasına bakılarak sınıflandırma yapıldığı için hesaplama maliyeti oldukça düşüktür. Fakat sadece port numarasına kullanıldığı için güvenilirliği de oldukça düşüktür. Çünkü çoğu uygulama standart olmayan port numaraları kullanabilmekte veya bilinen port numaralarından farklı uygulamalar çalıştırabilmektedir [18].

2.2. İçerik Temelli Sınıflandırma Yöntemi

Sadece port bilgilerine bakarak sınıflandırma işlemi kolay olsa da güvenilirliği oldukça düşüktür. Bunun için ağ paketlerinin içeriklerinin incelenerek trafiğin sınıflandırıldığı çalışmalar yapılmaktadır. Paket içeriğinin incelenerek analiz yapılması işlemine Derin Paket İncelenmesi (Deep Packet Inspection, DPI) adı verilmektedir [12]. Paket içeriği incelenerek yapılan sınıflandırma yöntemi port tabanlı sınıflandırma yöntemine göre maliyetli bir çözüm olmasına rağmen güvenilirliği çok daha yüksektir. Paket içerikleri bilinen kalıp veya imzalar ile karşılaştırılarak analiz ve sınıflandırma işlemi yapılmaktadır. Bu yöntem günümüz siber güvenlik çözümlerinde en yoğun kullanılan yöntemdir. Saldırı tespit sistemleri sahip oldukları imza veri tabanları ile paket içeriklerini karşılaştırarak zararlı tespiti ve uygulama sınıflandırması yapmaktadır. Bununla birlikte derin paket analizi için yapay zekâ algoritmaları da yoğun olarak kullanılmaktadır. Çünkü imza tabanlı sistemler sadece bilindik kalıpları kullandıkları için bilinmeyen durumlara karşı zayıftırlar.

Port tabanlı sistemlerin sınıflandıramadığı P2P uygulamalar için Sen [19] ve arkadaşları paket içeriği analizi ve belirli imza kalıpları kullanarak uygulama sınıflandırma çalışması yapmışlardır. İmzalar, dokümanlardan ve yakalanan paketler üzerinden analiz edilerek oluşturulmuştur. Oluşturulan yapı yüksek bant genişliğine sahip ağlar üzerinde denenmiş ve yanlış pozitif oranı %5 altında bir başarı sergilemiştir. Oluşturulan yapı ile port tabanlı çalışmalar ile karşılaştırıldığında oldukça başarılı sonuçlar elde edilmiştir. Kazaa P2P uygulaması port tabanlı sistemlere göre 3 kat daha başarılı şekilde sınıflandırılmıştır.

Moore [20] ve arkadaşları internet üzerinden topladıkları trafiklerin paket içeriklerini kullanarak port tabanlı sistemlerin dezavantajlarını gidermek amaçlı bir çalışma yapmışlardır. Veri seti, üniversite kampüs ağından 1000 araştırmacının internet trafiği yakalanarak hazırlanıyor. Oluşturulan veri setinin %94,81'i TCP, %3,58 ICMP ve %1,57 UDP'dir. Oluşturulan yapı ile %100 yakın doğruluk sağlanmıştır fakat canlı trafik üzerinde analiz gerçekleştirilmemiştir.

Finamore [21] ve arkadaşları UDP temelli İnternet Protokolü Üzerinden Ses (Voice Over Internet Protocol, VoIP) ve P2P uygulamaların sınıflandırılması amaçlı KISS adını verdikleri bir araç geliştirmişlerdir. KISS normal içerik inceleme araçlarına göre daha hızlı güncelleme, kolay uygulanabilirlik, az kaynak tüketimi avantajlarına sahiptir. KISS aracı

test veri seti ve canlı trafik üzerinde kullanarak performans ölçümü gerçekleştirilmiştir. Özellik çıkarımı için Destek Vektör Makinesi (Support Vector Machine, SVM) temelli karar mekanizması kullanılmıştır. Yapılan çalışmada doğru pozitif oranı %99,6 ve yanlış pozitif oranı ise %1'den küçüktür.

İçerik temelli sınıflandırma çok güvenilir bir yöntem olmasına rağmen bazı temel zayıflıkları ve eksikleri bulunmaktadır. Birincil olarak şifreli veya karıştırılmış verileri analiz edememektedir ve bu yüzden günümüz trafiğinin çoğunu analiz edememektedir. Ayrıca içerik analizi yapıldığı için gizlilik politika ve yasalarına aykırı olabilmektedir. Bunlara ek olarak çok fazla hesaplama gücü ve zaman gerektirdiği için yüksek hızlı trafiklerin analiz edilebilmesi çok zordur [22].

2.3. Akış Temelli Sınıflandırma Yöntemi

Derin paket analizi yapabilmek için şifrelenmemiş paketlerin içeriğine ihtiyaç duyulmaktadır ve analiz işlemleri fazla hesaplama gücü ve zaman gerektirmektedir. Paket içeriğinin şifreli olduğu durumlarda bu yöntemler kullanılamamaktadır. Bu durumlar için akış temelli istatistiksel yöntemler kullanılmaktadır. Akış temelli sistemlerde port bilgisi veya paket içeriği yerine paketin ağ üzerindeki istatistiksel verilerine bakılarak sınıflandırma işlemi yapılmaktadır. Bu yöntemde IP ve port bilgileri özellik olarak kullanılmamaktadır. Çünkü trafik farklı portlar üzerinden kurulabilmektedir ve bu durum yapay zekâ temelli yaklaşımlarda başarı oranını düşürmektedir [23].

Alshammari [24] ve arkadaşları SSH ve Skype trafiklerinin sınıflandırılması amaçlı bir çalışma yapmışlardır. Çalışma kapsamında SSH ile No-SSH ve Skype ile No-Skype olmak üzere ikili sınıflandırma uygulaması yapılmıştır. Problem çözümünde 5 farklı makine öğrenmesi algoritması kullanılmış ve C4.5 algoritmasının en iyi sonucu verdiği gözlemlenmiştir. C4.5 algoritması en kötü senaryoda %83,7 tespit oranı ve %1,5 yanlış pozitif oranı performansı, en iyi senaryoda ise %97 tespit oranı ve %0,8 yanlış tespit oranı performansı sergilemiştir.

Di Mauro [25] ve arkadaşları Web Tabanlı Gerçek Zamanlı İletişim (Web Real Time Communication, WebRTC) uygulamalar için sınıflandırma çalışması yapmışlardır. Veri seti yazarlar tarafından oluşturulmuş ve Tshark, MySQL ve Weka uygulamaları kullanılmıştır.

Çalışma kapsamında 4 farklı makine öğrenmesi algoritması kullanılmış ve ikili sınıflandırma yapılmıştır. Çalışma sonucunda Rassal Orman algoritmasının en başarılı sonucu verdiği belirtilmiştir. Rassal Orman algoritması %96.4 kesinlik oranı ve %96,4 F1-ölçüt oranına sahiptir.

2016 yılında Draper-Gil [26] ve arkadaşları akış temelli ve zamanla bağlantılı bir veri seti oluşturdular. Bu veri seti iki farklı senaryodan oluşmaktadır. Çalışma kapsamında VPN ile şifrelenmiş trafiğin ve VPN olmadan şifrelenmiş trafiğin sınıflandırılması yapılmıştır. Toplamda on dört sınıf etiketi bulunmaktadır. Çalışma kapsamında C4.5 ve K-En Yakın Komşu (K-Nearest Neighbors, kNN) algoritmaları kullanılmıştır. Hem ilk senaryo hem de ikinci senaryo için başarı oranları detaylı olarak verilmiştir. Çalışma sonucunda C4.5 algoritmasının kNN'den daha iyi sonuç verdiği tespit edilmiştir. C4.5 algoritması kullanılarak ilk senaryoda en yüksek keskinlik değeri %90,6 ve ikinci senaryoda en yüksek keskinlik değeri %80,9 olarak tespit edilmiştir.

Seddigh [27] ve arkadaşları yüksek hızlı ağlar üzerinde şifreli paketlerin sınıflandırılması amaçlı bir çalışma yapmışlardır. Veri seti kampüs ağından toplanmış ve iki yüz üzerinde özellik olmasına rağmen yüksek boyutlarda trafik olduğu için özellik seçimi yapılmıştır. Veri setinde 6 sınıf etiketi bulunmaktadır. Çalışma kapsamında 6 farklı makine öğrenmesi algoritması kullanılmıştır ve Makine Öğrenimi Trafik Analiz Aracı (Machine Learning Traffic Analytics Tool, MLTAT) adında bir yapı oluşturulmuştur. Hiper parametre seçimi MLTAT üzerinden yapılmış ve ikili ve çoklu sınıflandırma yapılmıştır. Yapılan testlerde tüm sınıflar için kesinlik değeri %88 üzerindedir fakat Web Browsing sınıfının başarı ortalamasını büyük oranda düşürdüğü tespit edilmiştir.

2018 yılında Caicedo-Muñoz [28] ve arkadaşları VPN ve VPN olmayan trafik üzerinde Hizmet Kalitesi (Quality of Service, QoS) amaçlı şifrelenmiş trafik sınıflandırma çalışması yapmışlardır. Çalışmada Draper-Gil [26] tarafından oluşturulan ISCX VPN-nonVPN veri seti kullanılmıştır. Yapılan çalışmada beş farklı makine öğrenme algoritması kullanılmıştır. İkinci senaryoda torbalama (bagging) ve yükseltme (boosting), diğer kullanılan algoritmalara göre daha iyi sonuçlar vermesine rağmen ilk senaryoda çok değişiklik olmadığı gözlemlenmiştir. Senaryo 1'in ikici kısmında VPN trafik için en yüksek doğruluk oranı %92,82 ve VPN olmayan trafik için en yüksek doğruluk oranı %94,42'dir. İkinci senaryoda ise %86,94 doğruluk oranı başarısı elde edilmiştir.

Saqib [29] ve arkadaşları paket içeriği yerine paketin özelliklerini kullanarak, statik analiz yöntemleriyle VoIP trafiğini sınıflandırmıştır. Çalışmada ağ üzerinden akan ses trafiğinin tespiti uygulama, güvenlik protokolleri ve şifreleme mekanizmalarından bağımsız olarak yapılmıştır. İnternette en çok kullanılan Skype, Yahoo ve Gmail gibi uygulamalar incelenmiştir. Analiz sonucunda paketlerin paket büyüklüğü dağılımı, paket çeşitliliği ve paket oranına özellikleri seçilerek sınıflandırma amacıyla kullanılmıştır. Ses trafiğinin düşük paket boyutuna, düşük paket çeşitliliğine ve yüksek paket oranı değerine sahip olduğu görülmüştür. Şifreli VoIP trafiğinde sınıflandırma işlemi daha zor olduğundan, paket sayısı, toplam akış süresi, aritmetik ortalama, standart sapma, paket sayısı ve maksimum-minimum paket boyutu gibi ek özellikler kullanılmıştır. Oluşturulan yapı ile çevrimdışı trafikte %93,6 tespit oranı, yazarlar tarafından toplanan VoIP trafiğinde %100 tespit oranı ve çevrimiçi trafikte ise % 95 tespit oranı elde edilmiştir.

Zhang [30] ve arkadaşları 2019 yılında Taşıma Katmanı Güvenliği (Transport Layer Security, TLS) veya SSL şifreli trafik üzerinden veri sınıflandırması için Stereo Dönüşüm Sinir Ağı (Stereo Transform Neural Network, STNN) adını verdikleri bir yapı oluşturmuşlarıdır. STNN, Uzun Kısa Süreli Bellek (Long Short-Term Memory, LSTM) ve Evrişimsel Sinir Ağları (Convolutional Neural Network, CNN) algoritmalarının birlikte kullanımı ile oluşturulmuştur. Veri seti, yazarlar tarafından oluşturulmuş ve 17 uygulama üzerinden toplanmıştır. Doğruluk oranı %99 üzerinde bir başarı elde etmişlerdir. Chari [31] ve arkadaşları J48 makine algoritmasını kullanarak şifreli trafik üzerinden sınıflandırma işlemi yapmışlardır. Veri seti olarak ISCXTor2016 veri seti kullanılmıştır ve paket uzunlukları tabanlı özellikler seçilmiştir. 6 sınıf üzerinden yapılan testlerde %91 doğruluk oranı elde edilmiştir.

Pradhan [32] ve arkadaşları şifreli SSH trafiğin ağ üzerinden tespit edilerek sınıflandırılması için bir çalışma yapmışlardır. Çalışmalarında hibrid Radyal Temel Fonksiyon Ağı (Radial Basis Function Network, RBFN) temelli makine öğrenme yaklaşımı kullanılmıştır. RBFN ileri beslemeli bir sinir ağı olup k-means kümeleme ve gradyan iniş (gradient descent) öğrenme tekniklerini kullanmaktadır. Yapılan testlerde RBFN ile elde edilen sonuçlar Adaboost, Karar Ağacı, Naive Bayes ve Rassal Orman algoritmaları ile karşılaştırılmıştır. RBFN tüm veri setleri için %99 üzerinde bir doğruluk oranı ile yüksek başarı sağladığı tespit edilmiştir.

2018 yılında Yang [33] ve arkadaşları TLS veya SSL ile şifrelenmiş trafiğin sınıflandırılması için Autoencoder Neural Network ve CNN algoritmaları kullanılarak derin öğrenme temelli bir yaklaşım kullanmışlardır. Autoencoder ve CNN için kullanılan parametreler detaylı olarak verilmiştir. Derin öğrenme için veri seti olarak Alexa [34] tarafından sağlanan en çok kullanılan 500 site içerisinden TLS/SSL trafik kullanan 10 ve 50 adet sayfa seçilerek iki adet veri seti oluşturulmuştur. Autoencoder ve CNN algoritmaları en başarılı sonuçlar veren beş farklı makine öğrenme algoritması ile karşılaştırılmıştır. CNN 10 web sayfa tabanlı veri setinde %96,46 doğruluk oranı ve 50 web sayfa tabanlı veri setinde %85 doğruluk oranı elde ederek en başarılı sonuçları vermiştir.

Al-Obaidy [35] ve arkadaşları şifreli kanal üzerinden haberleşen Skype, Whatsapp, Facebook, Netflix ve Youtube uygulamaları sınıflandırılması ve tespiti için makine öğrenmesi temelli bir yaklaşım kullanmışlardır. Çalışma kapsamında SVM, Çok Katmanlı Algılayıcılar (Multilayer Perceptron, MLP), Naive Bayes ve C4.5 makine öğrenme algoritmaları kullanılmıştır. Veri seti ekip tarafından oluşturulmuş ve Wireshark aracılığı ile son kullanıcı bilgisayarlarından toplanmıştır. Oluşturulan veri seti içerisinde 14 adet özellik bulunmaktadır. İlk önce Skype, Whatsapp, Facebook ve Youtube olmak üzere dört uygulama için sınıflandırma işlemi gerçekleştirilmiş ve %88,29 doğruluk oranı ile C4.5 algoritması en iyi sonucu verdiği tespit edilmiştir. Bu dört uygulamaya Netflix uygulaması ek olarak eklenerek beş uygulama üzerinde testler yapılmıştır. Yine C4.5 algoritması %86,33 doğruluk oranı ile en başarılı algoritma olduğu gözlemlenmiştir.

Khatouni [36] ve arkadaşları şifreli trafik kullanan sosyal medya, ses ve video uygulamalarının sınıflandırılması için bir çalışma yapmışlarıdır. Bu amaçla firefox ve chrome tarayıcıları üzerinden web sayfalarını otomatik olarak ziyaret eden ve uygulamayı kullanan bir araç ile veri seti oluşturmuşlardır. Çalışmada 13 farklı makine öğrenme algoritması kullanılmıştır. Rassal Orman algoritması en iyi sonucu vermesine rağmen Karar Ağacı algoritmasına göre daha fazla kaynak tükettiği tespit edilmiştir. Karar Ağacı algoritmasının Rassal Orman algoritmasına göre yedi kat daha hızlı çalıştığı belirtilmiştir. Çalışma sonucunda Karar Ağacı algoritması doğru pozitif oranı %85 ve yanlış pozitif oranı %5 olarak oluşturulan problemin çözümü için en güçlü algoritma olduğu belirtilmiştir.

Çizelge 2.1. Literatür özeti

Kaynak	Sınıflandırma Yöntemi	Sınıf Bilgileri	Veri Seti	Analiz Yöntemi	Sonuçlar
Plonka [15]	• Port temelli	• Çok bilinen port numaraları	• Canlı trafik	• Paket içerik analizi	• Sadece bilinen portlar
Moore [16] ve arkadaşları	• Port temelli	• Çok bilinen port numaraları	• Canlı trafik	• Paket içerik analizi	• Sadece bilinen portlar
Fraleigh [17] ve arkadaşları	• Port temelli	• Web • Mail • FT • P2P • Streaming	• Canlı trafik	• Paket içerik analizi	• Sadece bilinen portlar
Sen [19] ve arkadaşları	• İçerik temelli	• P2P	• Canlı trafik	• İçerik analizi	• Yanlış Pozitif Oranı: < %5
Moore [20] ve arkadaşları	• İçerik temelli	• TCP • UDP • ICMP	• Yazarlar tarafından oluşturulurmuş tur.	• İçerik analizi	• Doğruluk Oranı: %100
Finamore [21] ve arkadaşları	• İçerik temelli	• VoIP • P2P	• Canlı trafik	• SVM	• Doğru Pozitif Oranı: %99,6
Alshammari [24] ve arkadaşları	• Akış temelli	• SSH • No-SSH	• Dalhouise • AMP • MAWI • DARPHA99	• AdaBoost • SVM • Naive Bayes • RIPPER • C4.5	• Tespit Oranı: %97
Di Mauro [25] ve arkadaşları	• Akış temelli	• WebRTC • No-WebRTC	• gruveo.com üzerinden veri seti toplanıyor.	• J48 • SimpleCART • Naive Bayes • Rassal Orman	• Kesinlik Oranı: %96,4
Draper-Gil [26] ve arkadaşları	• Akış temelli	• VPN • Non-VPN • Browsing • Chat • FT • Mail • P2P • Streaming • VoIP	• Yazarlar tarafından oluşturulurmuş tur.	• C4.5 • kNN	• Senaryo A Kesinlik Oranı: %90,6 • Senaryo B Kesinlik Oranı: %80,9
Seddigh [27] ve arkadaşları	• Akış temelli	• Video • Video Chat • Audio • VoIP • File Transfer • Mail • Web browsing • P2P • Chat • ToR	• Yazarlar tarafından oluşturulurmuş tur.	• Logistic Regression • SVM • Karar Ağacı • Adaboost • Neural Network • Naive Bayes	• Kesinlik Oranı: %88 üzerinde

Çizelge 2.1. (devam) Literatür özeti

Caicedo-Muñoz [28] ve arkadaşları	• Akış temelli	• VPN • Non-VPN • Browsing • Chat • FT • Mail • P2P • Streaming • VoIP	• ISCX VPN-NonVPN	• C4.5 • kNN • Naive Bayes • MLP • Bagging ve Boosting	• Senaryo B Doğruluk Oranı: %86,94
Saqib [29] ve arkadaşları	• Akış temelli	• VoIP • No-VoIP	• Yazarlar tarafından oluşturulurmuş tur.	• Statik analiz	• Çevrimiçi Tespit Oranı: %93,6 • Çevrimdışı Tespit Oranı: %95
Zhang [30] ve arkadaşları	• Akış temelli	• 18 adet uygulama	• Yazarlar tarafından oluşturulurmuş tur.	• STNN	• Doğruluk Oranı: %99
Chari [31] ve arkadaşları	• Akış temelli	• Audio • Browsing • FTP • P2P • Video • VoIP	• ISCXTor2016	• J48	• Doğruluk Oranı: %91
Pradhan [32] ve arkadaşları	• Akış temelli	• SSH • No-SSH	• AMP • MAWI • DARPAA99 • NIMS4	• RBFN • Adaboost • Karar Ağacı • Naive Bayes • Rassal Orman	• Doğruluk Oranı: %99
Yang [33] ve arkadaşları	• Akış temelli	• Web Sayfaları	• Yazarlar tarafından oluşturulurmuş tur.	• Autoencoder • CNN • Naive Bayes • Logistic Regression • kNN • Karar Ağacı • Rassal Orman	• Veri seti 10 web sayfalı Doğruluk Oranı: %96,46 • Veri seti 50 web sayfalı Doğruluk Oranı: %85
Al-Obaidy [35] ve arkadaşları	• Akış temelli	• Skype • Whatsapp • Facebook • Netflix • Youtube	• Yazarlar tarafından oluşturulurmuş tur.	• SVM • MLP • Naive Bayes • C4.5	• Doğruluk Oranı: %86,33
Khatouni [36] ve arkadaşları	• Akış temelli	• Audio • Bank • Chat • Mail • News • Online • Social • Video • Weather • Web	• Yazarlar tarafından oluşturulurmuş tur.	• 13 adet makine öğrenme algoritması	• Doğru Pozitif Oranı: %85

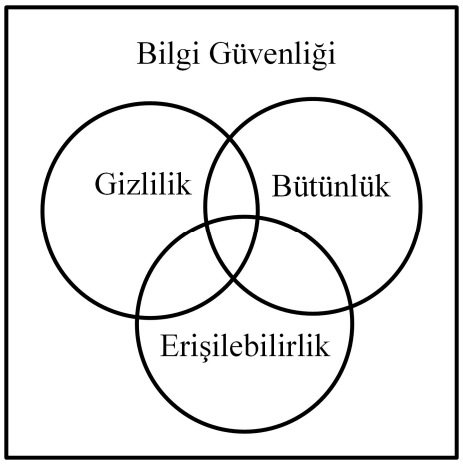
Literatürde yapılan çalışmalara ait kaynak, sınıflandırma metotları, sınıf bilgileri, kullanılan veri setleri, algoritmalar ve sonuçlar Çizelge 2.1'de gösterilmiştir.

3. KAPSAM, YÖNTEM VE ARAÇLAR

Günümüzde trafiğin şifrelenmesi amacı ile en çok SSL ve VPN protokol ve yöntemleri kullanılmaktadır. Şifrelenen trafiğin sınıflandırılması işlemi ağ üzerinde giden-gelen paketlerin boyutu ve gidiş-geliş zamanı özelliklerinden faydalanılarak yapılmaktadır. Her uygulamanın kendine özgü bir yapısı olduğu için giden-gelen paketlerin boyutu, sayısı ve zamanı gibi değerleri farklı olabilmektedir ve bu farklılıklar sınıflandırma amaçlı kullanılmaktadır. Bu çalışmada belirtilen özelliklerden faydalanılarak ve makine öğrenmesi algoritmaları kullanılarak VPN ile şifrelenmiş ve VPN kullanılmadan şifrelenmiş paketlerin sınıflandırılması yapılmıştır. Bu bölümde şifreleme yöntemleri, kullanılan makine öğrenmesi algoritmaları, veri seti ve performans metriklerinden bahsedilmektedir.

3.1 Şifreleme Yöntem ve Protokolleri

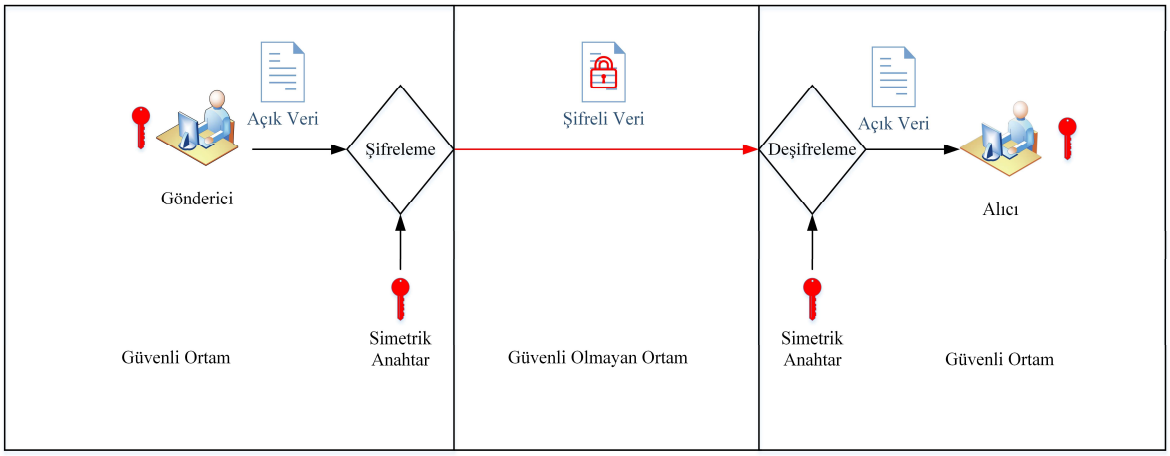
Bilgi güvenliği, bilginin yetkisiz kişi veya sistemler tarafından erişilmesi, kullanılması, yok edilmesi, bozulması veya değiştirilmesinin önlenmesi anlamına gelmektedir [37]. Verinin gizliliği, bütünlüğü ve erişilebilir olması bilgi güvenliğinin 3 temel unsurudur [38] ve Şekil 3.1'de gösterilmiştir. Gizlilik bilginin sadece yetkili kişiler tarafından erişilebilir olmasını sağlayan çok önemli bir unsurdur [39]. Bütünlük, verinin değiştirilmediği veya bozulmadığını kanıtlayan bir özelliktir [40]. Verinin istenilen anda sadece yetkili kişiler tarafından erişilebilmesinin sağlanmasına erişilebilirlik denilmektedir.



Şekil 3.1. Bilgi güvenliği 3 temel unsuru [39]

Kriptografi, verinin gizliliğini, bütünlüğünü, veri kaynağının kimliğinin doğrulanması gibi bilgi güvenliğiyle ilgili matematiksel tekniklerinin araştırılmasıdır. Bilgi güvenliği alanında kriptografi yoğun olarak kullanılmaktadır. Gizlilik, bütünlük, kimlik doğrulama ve inkâr edilememe özellikleri kriptografinin amaçları arasındadır [41]. Gizliliğin sağlanması amacı ile kriptografide şifreleme yöntemleri kullanılmaktadır. Şifreleme işlemi sonrasında veriler okunan tarafından anlaşılmayacak ve gizliliği bozulmayacak bir formata çevrilmektedir. Verinin bu haline şifreli veri denir. Şifreleme işlemi sırasında cipher adı verilen algoritmalar ve anahtarlar kullanılmaktadır [42].

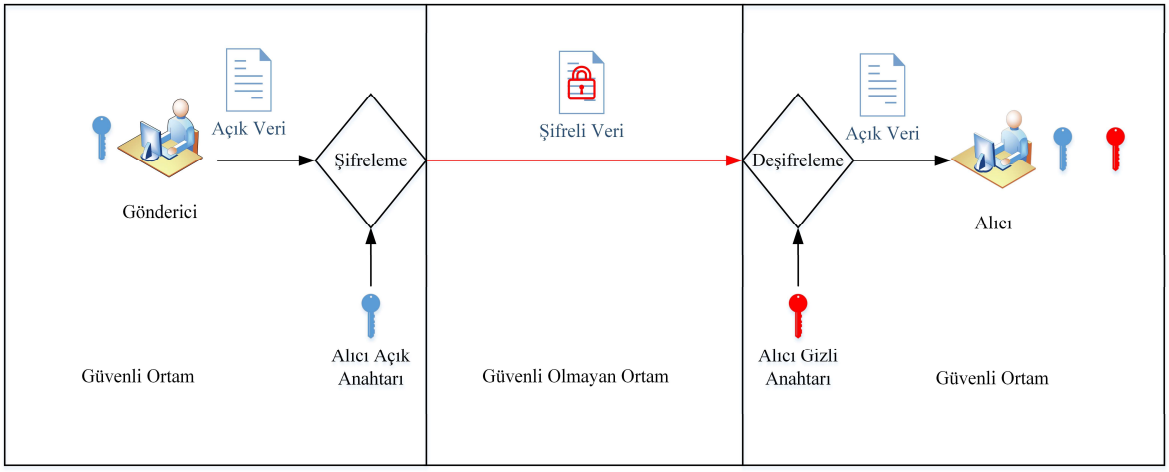
Şifreleme işlemi simetrik ve asimetrik olarak iki farklı biçimde yapılabilmektedir. Simetrik şifrelemede şifreleme ve deşifreleme işlemlerinde aynı anahtar kullanılmaktadır [43] ve en eski yöntem olarak bilinmektedir. Gönderen ve alan taraf aynı anahtara sahip olmak zorundadır. Şekil 3.2'de simetrik şifreleme yöntemi kullanılarak güvensiz ortam üzerinden güvenli veri iletimi gösterilmiştir. Gönderici göndermek isteği veriyi kendisi ve alıcı tarafından bilinen ortak simetrik anahtar yardımı ve şifreleme algoritmalarını kullanarak şifreler. Şifreli veriyi güvensiz ortam üzerinden alıcıya gönderir. Bu veri sadece şifreleme esnasında kullanılan anahtar ile çözülebildiği için güvensiz ortamda hattı dinleyen ve paketi yakalayan saldırganlar tarafından çözülemeyecektir. Simetrik şifreleme yönteminde simetrik anahtarın güvenli yollarla paylaşılması bu yöntemin en büyük problemidir. Bu sorunun üstesinden gelebilmek için asimetrik şifreleme ve anahtar çiftleri kullanılmaktadır [44].



Şekil 3.2. Simetrik şifreleme [45]

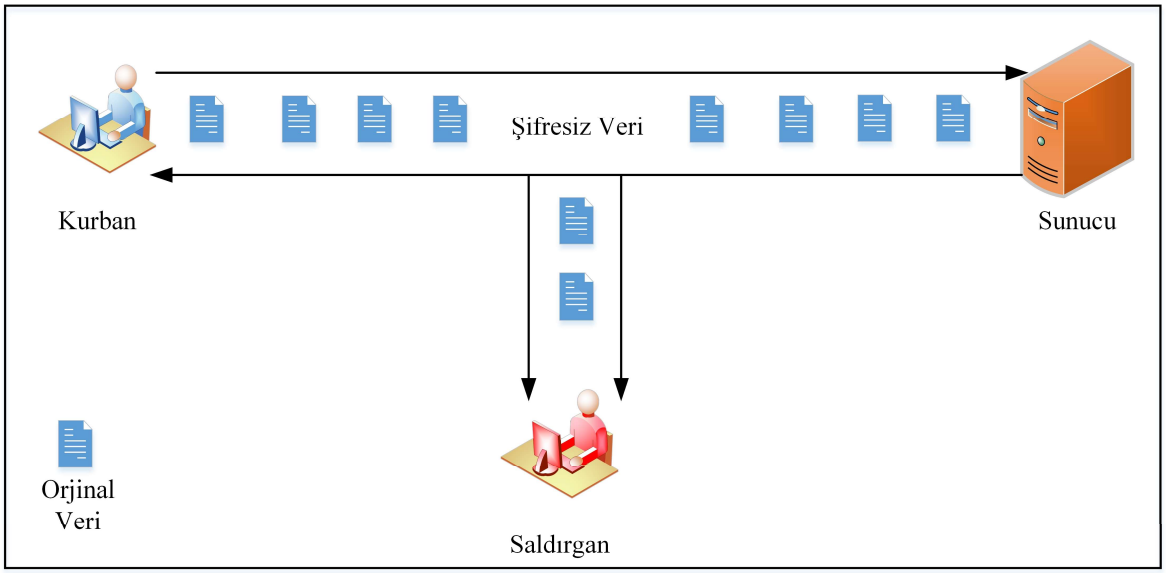
Asimetrik şifrelemede şifreleme ve deşifreleme işlemleri birbirleri ile matematiksel olarak bağlantılı iki farklı anahtarlar ile yapılmaktadır. Bu anahtarlara açık ve gizli anahtarlar adı

verilmektedir. Açık anahtar herkes tarafından erişilebilirken gizli anahtar sadece yetkili kişi veya sistemler tarafından erişilebilmektedir. Açık anahtar dağıtımının güvenli olarak sağlanabilmesi için sertifikalar kullanılmaktadır ve bu sertifikalar güvenilir makamlar tarafından üretilmekte ve dağıtılmaktadır. Şifreleme işlemi verinin gönderileceği kişi veya sisteme ait açık anahtar ile yapılır ve şifrelenmiş veri sadece verinin gönderileceği kişide bulunan gizli anahtar ile çözülebilir [46]. Asimetrik şifreleme ile güvenli veri iletimi Şekil 3.3'de gösterilmiştir. Asimetrik şifrelemede, simetrik şifrelemede bulunan anahtar paylaşım problemi çözülmesine rağmen şifreleme ve deşifreleme işlemleri simetrik şifrelemeye göre çok daha uzun sürmektedir [47].



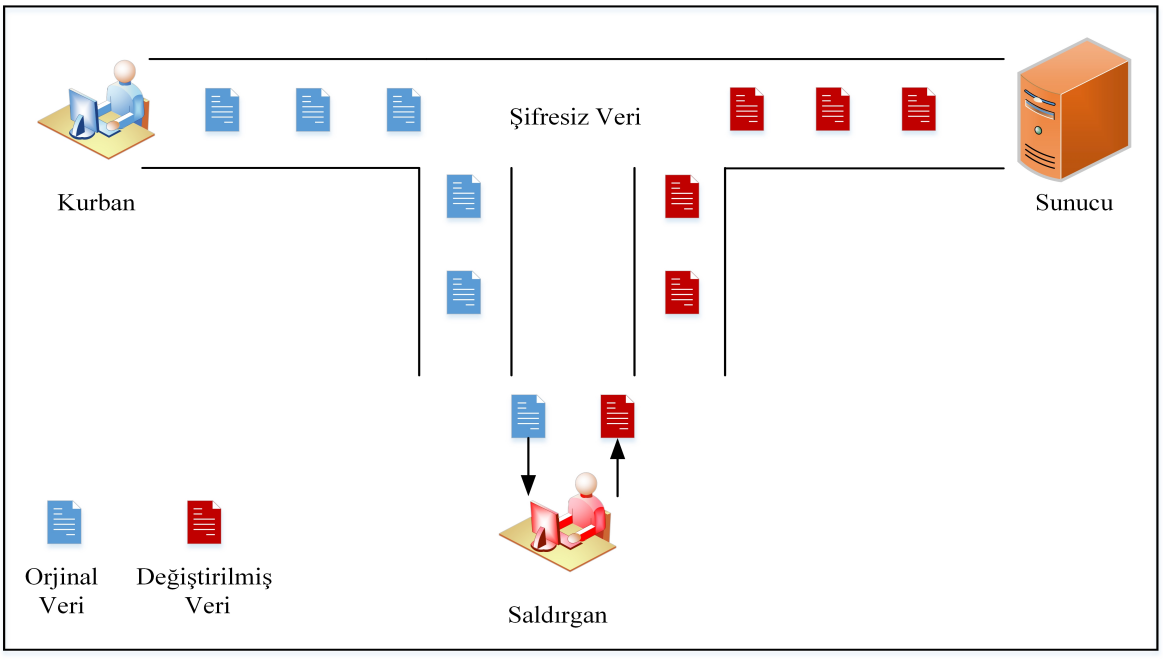
Şekil 3.3. Asimetrik şifreleme [45]

Şifrelemenin olmadığı durumlarda hattı dinleyen saldırgan trafiğe müdahale etmeden sadece dinleme yaparak verileri ele geçirirse bu saldırıya pasif dinleme (eavesdropping) saldırısı denilmektedir [48]. Şekil 3.4'de bu saldırı senaryosu gösterilmektedir. Kurban ile sunucu arasında bulunan hattı dinleme yeteneğine veya olanağına sahip saldırgan, gönderilen paket içerikleri şifreli olmadığı için paketlerin bir kopyasını edinerek tüm iletişimi öğrenebilmektedir.



Şekil 3.4. Pasif dinleme saldırısı [48]

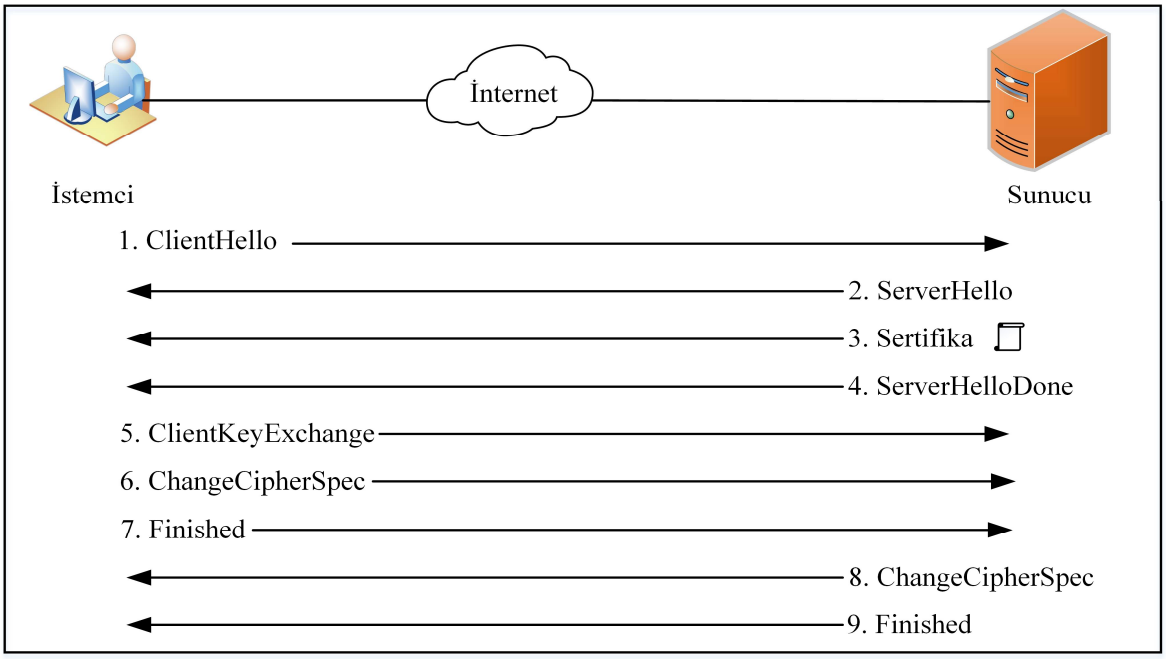
Eğer saldırgan ortadaki adam saldırısı yaparak araya girer ve paket içeriklerini değiştirerek trafiği manipüle ederse bu saldırıya manipülasyon (modification) saldırısı denilmektedir [49]. Bu saldırı senaryosu Şekil 3.5'de gösterilmiştir. Bu saldırıda eavesdropping saldırısından farklı olarak saldırgan amacı doğrultusunda paketlerin içeriklerini manipüle edebilmektedir. Şifrelemenin başarı bir şekilde uygulandığı durumlarda ise saldırganlar trafik üzerinde sadece anlamsız veriler görebilmektedir.



Şekil 3.5. Manipülasyon saldırısı [50]

SSL genellikle HTTP protokolü ile bütünleşik çalışmasına rağmen TCP tabanlı tüm protokoller ile çalışabilmektedir [51]. SSL şifreleme protokolü düzgün bir şekilde kullanıldığında ve doğru sunucu ile iletişim kurulduğunda hattı dinleyen saldırganlar tarafından ele geçirilen paketler anlamsız veriler içerecektir. SSL protokolünün tek amacı sadece kriptografik güvenlik sağlamak değildir. Bununla birlikte programcıdan bağımsız olarak çalışabilmekte ve diğer yaygın kullanılan kriptografik parametreler ile iletişim kurabilmektedir. Şifreleme ve deşifreleme işlem performansını artırarak veri iletişiminde verimlilik sağlamaktadır [52].

SSL, TCP protokolü üzerinden güvenli veri iletimi sağlamaktadır. SSL tünel bağlantısının oluşturulabilmesi için sunucunun SSL sertifikası olması gerekmektedir. Güvenli iletişime başlarken istemci, sunucuya bağlantı isteği gönderir. Gönderilen istek paketinin içeriğinde sürüm bilgisi, oturum numarası, rastgele bir sayı, kriptografik algoritmalar ve sıkıştırma metotları bulunmaktadır ve bu pakete "ClientHello" paketi adı verilmektedir. İstemciden bağlantı isteğini alan sunucu, istemciye bağlantının sağlanacağı sürüm numarasını ve kriptografik algoritma bilgilerini içeren "ServerHello" paketini gönderir. Bu paket içerisinde ayrıca istemcinin gönderdiği rastgele sayı ve oturum numarası da bulunmaktadır. Sunucu "ServerHello" paketine ek olarak anahtar değişimi için sertifikasında yüklü olan açık anahtar bilgisini de paylaşır. Sunucu tarafından gönderilen bilgileri alan istemci güvenli oturum bağlantısının sağlanabilmesi için oturum anahtarı adı verilen ve simetrik şifreleme için kullanılacak anahtara ait bilgileri sunucu açık anahtarı ile şifreleyerek sunucuya gönderir. Bu paketle birlikte güvenli oturum başlayabileceğini söyleyen bir başka paket daha gönderir. Aynı doğrulama paketi sunucu tarafından istemciye gönderildikten sonra SSL oturumu tamamlanır ve güvenli kanal üzerinden veri iletişimi başlar [53]. SSL bağlantısı için gerekli olan akış şeması Şekil 3.6'da gösterilmiştir. SSL'de hem asimetrik şifreleme hem de simetrik şifreleme altyapısı birlikte hibrit olarak kullanılmaktadır. Asimetrik şifreleme yöntemi, simetrik şifreleme anahtarının güvenli iletimi için kullanılmaktadır. İstemci sunucu arasındaki iletişimin hızlı olması için veri şifrelemesinde simetrik şifreleme yöntemi kullanılmaktadır.



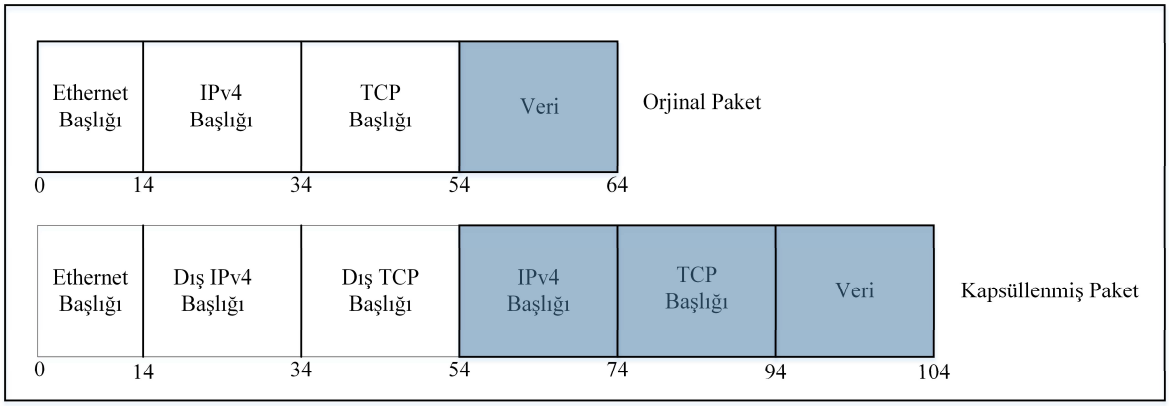
Şekil 3.6. SSL bağlantı akış şeması [53]

İnternet Protokolü Güvenliği (Internet Protocol Security, IPsec), IP üzerinden güvenli iletişimin sağlanabilmesi için oluşturulmuş açık bir standarttır. İnternet Anahtar Değişim (Internet Key Exchange, IKE) protokolü IPsec parametre ve şifreleme anahtarlarının güvenli olarak iletilebilmesi için kullanılmaktadır. IPsec trafik üzerinde gizlilik, bütünlük, uç doğrulama, tekrar saldırısına karşı koruma, trafik içeriği analizine karşı koruma, erişim kontrolü ve mükemmel anahtar gizliliği sağlayan özelliklere sahiptir [54]. IPsec, gizlilik için anahtar ve şifreleme algoritmaları kullanarak veriyi şifreler. Bütünlük kontrolü için gönderilen verinin özet değeri alınarak Mesaj Doğrulama Kodu (Message Authentication Code, MAC) üretilir ve alıcıya gönderilir. Alıcı tarafında aynı algoritmalar kullanılarak MAC değeri üretilir ve alınan paket içeriğindeki değer ile karşılaştırılarak verinin değişip değişmediği anlaşılır. Eğer MAC değerleri birbirinden farklı ise paketin yolda değiştirildiği, bozulduğu veya eksik geldiği anlaşılır [55]. IPsec her iki ucun kimlik bilgilerini doğrulayarak iletişimin doğru kişiler tarafından yapıldığını doğrular. Bu aşamada IPsec kullanan uç noktalar kendi belirledikleri yöntemler ile kimlik doğrulaması sağlarlar.

Aynı veri farklı zamanlarda gönderildiğine IPsec tarafından kabul edilmez. Bu sayede saldırganın şifreli trafiği kopyalayarak tekrar gönderilmesinin önüne geçilir. IPsec tünel modu kullanıldığında ağı dinleyen saldırganlar hangi iki uç noktanın iletişime geçtiğini, ne kadar veri gönderildiğini ve ne sıklıkla iletişim kurulduğunu öğrenemezler. IPsec tüm

paketlerin boyutunu sabit bir boyuta getirerek boyut üzerinden paket analizi yapılmasını ve arada gereksiz paket göndererek paket sıklığı üzerinden analiz yapılmasını engeller. Erişim kontrolü için IPsec filtreleme yöntemleri kullanarak sadece yetkili kullanıcı veya belirlenen trafik paket tiplerine izin verir. Mükemmel anahtar gizliliği, alıcının veya gönderenin uzun vadeli gizli anahtarlarının gizliliğinin bozulması durumunda önceki oturum anahtarlarının ele geçirilememesi ve gönderilen verilerin şifresinin çözülememesi anlamına gelmektedir. Bunun sağlanabilmesi için IPsec düzenli olarak yeni anahtar üretir ve eski anahtarları imha eder.

VPN, mevcut ve güvensiz fiziksel hat üzerinden IPsec kullanarak hem veriyi hem de IP paketini şifreleyerek iki uç arasında güvenli bir tünel kuran özel sanal bir ağdır [54]. Özel bir hat satın alınmasından uygun maliyetli olduğu için yoğun olarak kullanılmaktadır. VPN iki ağ arasında kurulabildiği gibi uzaktan sunucu veya istemci cihazlarına güvenli iletişim için de kullanılabilmektedir. VPN ağında gönderilecek olan tüm ağ paketi başka bir ağ paketi içine veri olarak alınır ve şifreleme işlemi yapılır. Bu işleme kapsülleme adı verilmektedir [56]. Kapsülleme işlemi Şekil 3.7'de gösterilmiştir. Orijinal paket içerisinde bulunan IP, TCP başlıkları ve veriler kapsülleme yöntemi ile başka bir paketin içerisine veri olarak eklenir ve şifrelenerek gönderilir. Bu sayede orijinal paket IP, TCP başlıkları ve veriler güvenli bir şekilde alıcıya iletilir.

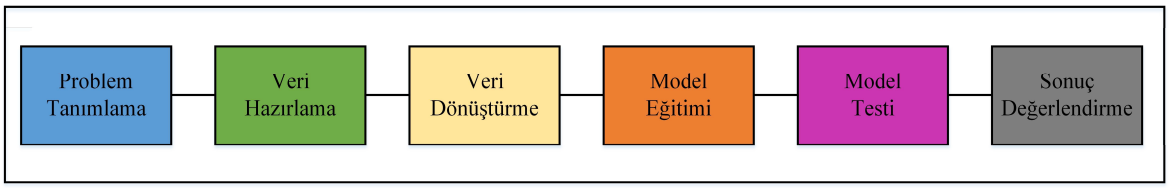


Şekil 3.7. Kapsülleme [56]

3.2 Makine Öğrenmesi ve Algoritmalar

Bilgisayar biliminin alt dalı olan yapay zekâ, nesneleri öğrenebilen, bilgi toplayabilen, iletişim kurabilen ve çevresindeki değişiklikleri algılayabilen akıllı makinelerin ve yazılımların incelenmesi ve geliştirilmesidir [57]. Yapay zekâ mühendislik, matematik, istatistik, psikoloji gibi birçok bilim dalı ile birlikte çalışmaktadır [58]. Yapay zekânın temeli 1900'lü yılların ortalarına dayanmaktadır. 1950 yılında A. M. Turing [59] yaptığı çalışmada makineler düşünebilir mi sorusu ile bilim dünyasına yeni bir yaklaşım getirmiştir. Bu tarihten itibaren makinelerin düşünebilmesi üzerine çalışmalar devam etmiştir. Günümüzde yapay zekâ ve makine öğrenmesi konuşma tanıma, doğal dil işleme, mühendislik, tıp ve siber güvenlik gibi çeşitli alanlarda ve zor problemlerin çözümlerinde yoğun olarak kullanılmaktadır [60]. Makinelerin öğrenmesi ve çevresindeki değişikliklere uyum sağlayabilmesi ihtiyacı makine öğrenmesi adı verilen alanın oluşmasına neden oldu [61]. Makine öğrenmesi, öğrenme yeteneğine sahip makineler yaratmak için ortaya çıkan ve yapay zekânın alt bir dalı olan bilim dalıdır [58]. Makine öğrenmesi doğru tahminler ve performans iyileştirmesi yapabilmek için deneyimler kullanarak matematiksel hesaplamalar yapar [62]. Deneyim, geçmiş verilerden öğrenilen bilgileri ifade etmektedir ve makineler öğrendikleri verileri deneyim haline getirirler.

Makine öğrenmesinin kullanılabilmesi için ilk olarak problemin belirlenmesi ve formüle edilmesi gerekmektedir. Bu aşamadan sonra verilerin toplanması ve toplanan verilerin analiz işlemleri yapılır. Veriler makinelerin anlayabileceği bir formata dönüştürülür. Sonrasında veriler eğitim ve test olarak ayrılır. Eğitim verisi ile model eğitildikten sonra test verileri kullanılarak model performans değerlendirmesi yapılır. Son olarak model istatistiksel yöntemler ile analizin edilir ve önemli özellikler ve parametreler belirlenerek model tekrar değerlendirilir [58]. Şekil 3.8'de makine öğrenmesi için gerekli adımlar gösterilmiştir.



Şekil 3.8. Makine öğrenmesi adımları [58]

Literatürde makine öğrenmesi uygulamaları için farklı tipte yöntemler kullanılmaktadır. En çok denetimli ve denetimsiz öğrenme kullanılmasına rağmen yarı denetimli ve pekiştirmeli öğrenme yöntemleri de kullanılmaktadır. Yöntem veya yöntemler probleme göre belirlenmektedir. Denetimli öğrenme yöntemi için kullanılan verilerin etiketli ve sınıflandırılmış olması gerekmektedir. Bu öğrenme yöntemi en çok sınıflandırma ve tahmin etme problemlerinde tercih edilmektedir [63]. Bu yöntemde en çok kNN, Karar Ağacı, Naive Bayes, SVM, Rassal Orman algoritmaları kullanılmaktadır. Bu yöntem için en büyük problem verilerin etiketlenmiş olması gerektiğidir.

Denetimsiz öğrenme yönteminde girdiler bulunmaktadır ve bu girdilere karşı çıktılar bulunmamaktadır. Bu yöntemin verilerinde etiket veya sonuç yoktur. Bu yöntemin amacı verilerin detaylı analiz edilerek verilerin yapısını ve dağılımını öğrenmektedir. Modeli denetleyen bir denetmen yoktur. Etiketsiz veri olması dezavantaj bir durum olarak gözükmesine rağmen gerçek hayatta tüm verilerin etiketsiz olması nedeni ile bu yöntem daha avantajlıdır. Bu yüzden gerçek hayatta daha çok kullanım olanağında sahiptir [64]. Bu yöntemin en büyük avantajı verilerin etiketlenme yükü olmamasıdır [65]. Denetimsiz öğrenmede amaç sınıflandırma veya tahmin ile verinin bir sınıfa ait olup olmadığını araştırılması değildir. Amaç veriler arasında bulunan benzerliklerin araştırılması ve verilerin kümelenmesidir [66]. Bu öğrenme yöntemi genellikle kümeleme ve boyut indirgeme amaçlı kullanılmaktadır [62].

Yarı denetimli öğrenme yönteminde hem etiketli hem etiketsiz verilerden oluşan bir veri seti kullanılır ve tüm görünmeyen noktalar için tahminlerde bulunulur [62]. Bu yöntem etiketlenmemiş verinin çok olduğu ve etiketlenmemiş verinin az olduğu durumlarda kullanılmaktadır. Bu yöntem kümeleme, sınıflandırma ve tahmin etme gibi birçok problem için kullanılabilmektedir.

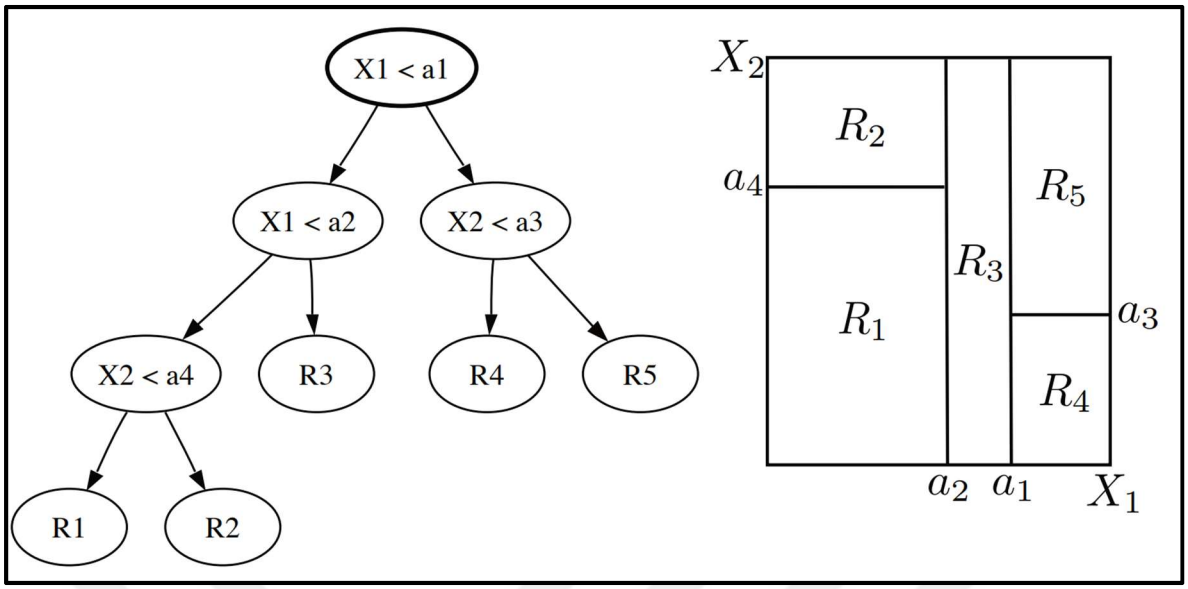
Pekiştirmeli öğrenme model ile çevrenin etkileşimde olduğu bir yöntemdir. Model çevreden öğrenirken, çevre ise modelin öğrendiği alandır [66]. Bu yöntemde eğitim ve test aşamaları birlikte yapılır. Öğrenme aşamasında bilgi toplamak için çevre ile iletişime geçilir ve bazı durumlarda çevreyi değiştirebilir. Model her bir eylemi sonucunda ödül veya ceza almaktadır. Modeli daha çok ödül alabilmesini sağlamak için yönlendirmek asıl amaçtır [67]. Model farklı stratejiler deneyerek hangisinin en iyi olduğunu öğrenmeye çalışmaktadır.

Çalışma kapsamında mevcut problemin çözümü için denetimli makine öğrenmesi yöntemi ve algoritmaları kullanılmıştır. Kullanılan veri seti sınıf etiketine sahip oluğu ve problem sınıflandırma işlemi olduğu için denetimli öğrenme yöntemi seçilmiştir. Çalışma kapsamında 3 farklı algoritma kullanılmıştır.

3.2.1. Karar ağacı algoritması

Quinlan [68] tarafından geliştirilen ve çok yaygın kullanılan bir makine öğrenmesi algoritmasıdır. Karar Ağacı algoritması sınıflandırma, kümeleme ve regresyon amaçlı kullanılabilmektedir [62]. Bu yöntem problem çözümünde birçok düğüm oluşturur ve bu düğümler arası ilişki kurar. İlk düğüme kök düğüm denir. Bu düğümler aracılığı ile bir karar ağacı oluşturur. Karar ağaçlarının eğitilmesi ve değerlendirilmesi hızlıdır ve yorumlanması kolaydır [62]. Desicion Tree algoritması kategorik veya sayısal veriler üzerinde işlem yapabilme yeteneğine sahiptir.

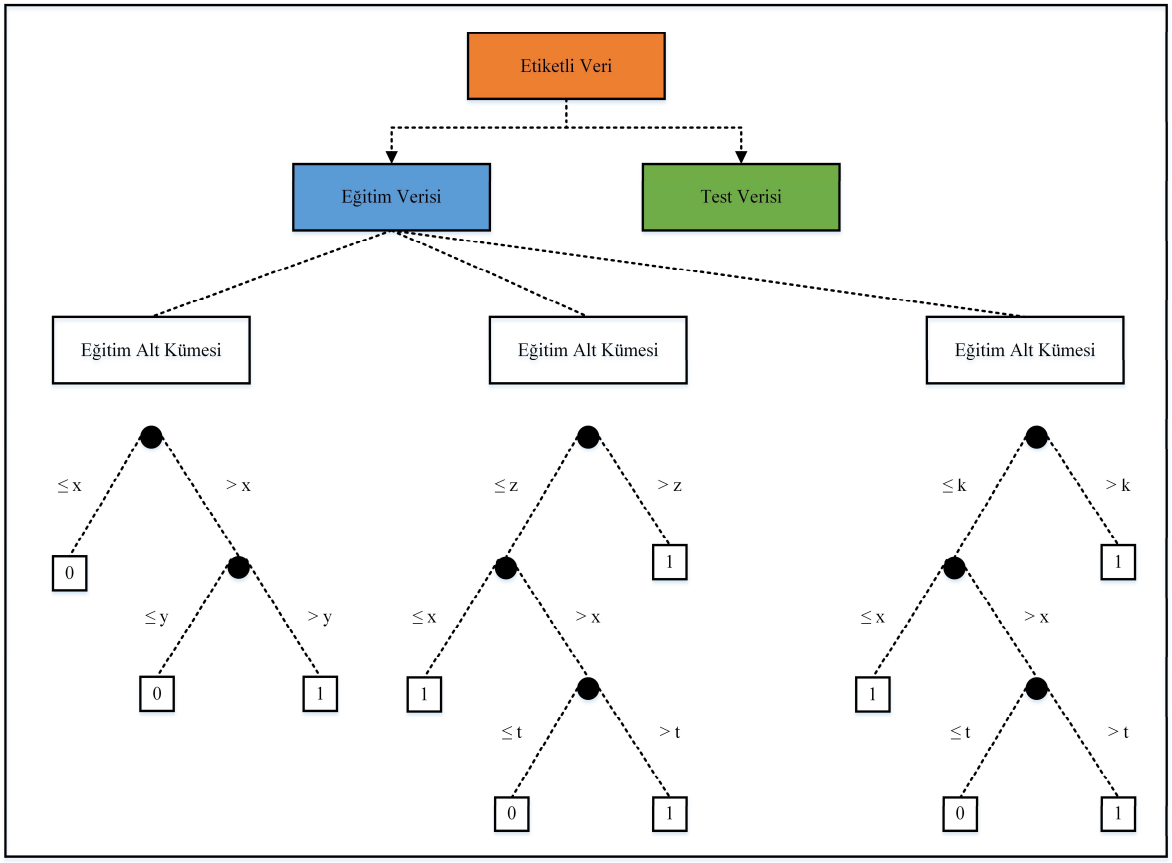
Kök düğüm, dallar ve yapraklardan oluşan karar ağaçları kök düğümden aşağıya doğru inerek veri kümesini alt gruplara böler. Düğümler bir özelliği analiz eder ve dallar aracılığı ile bir diğer düğüme veya sonuca ulaşır. Yapraklar sonuçları ifade etmektedir. Oluşan yapı bir ikili ağaca benzediğinden adını buradan alır. Şekil 3.9'da sol kısmında X1 ve X2 özelliklerine dayalı sorular içeren bir karar ağacı örneği ve sağ kısmında ise karar ağacının neden olduğu iki boyutlu uzay gösterilmektedir. X1 ve X2, özellikler ile bağlantılı a1, a2, a3 ve a4 ile matematiksel olarak değerlendirilir ve R eşik kümesine göre kıyaslanarak alt düğümlere veya yapraklara gönderilir [62].



Şekil 3.9. İki boyutlu karar ağacı [62]

3.2.2. Rassal orman algoritması

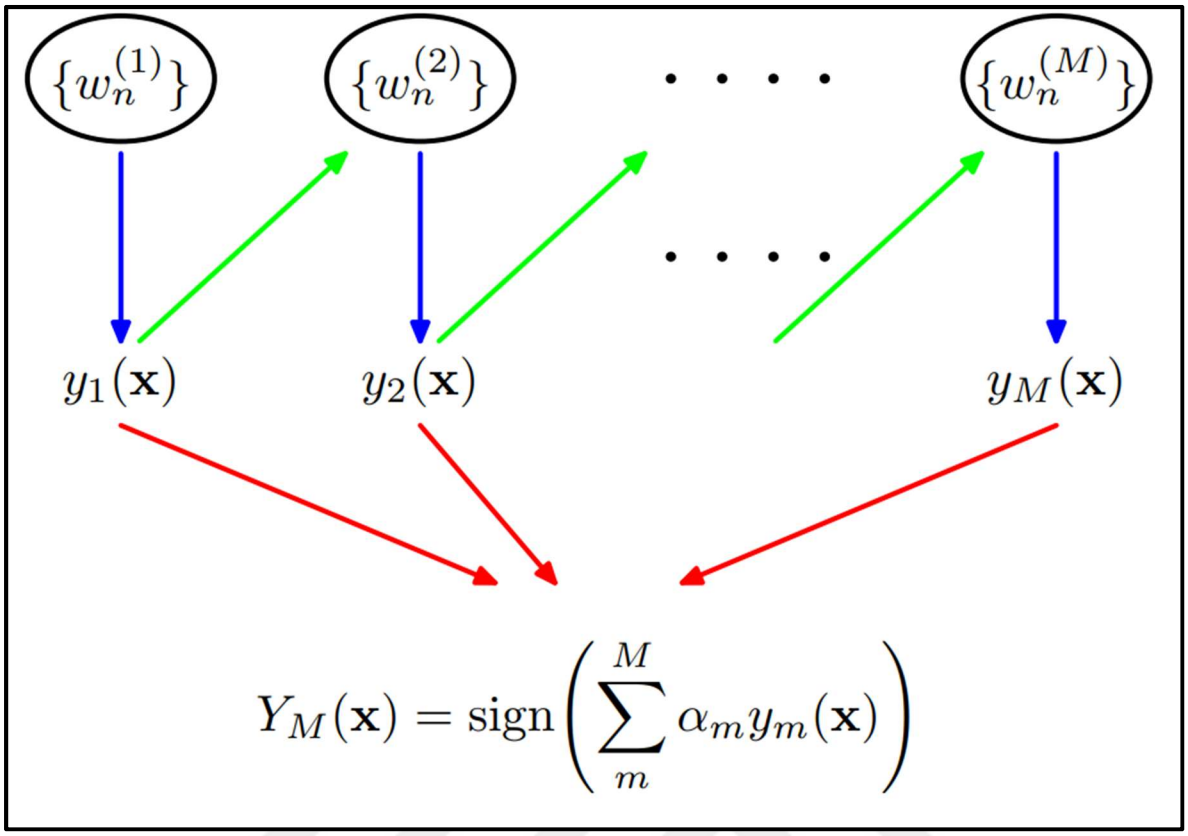
Topluluk (ensemble) yöntemi, daha doğru bir model oluşturulabilmesi için birkaç tahmin edicinin birleştirilmesi yöntemidir [62]. Rassal Orman algoritması L. Breiman [69] tarafından 2001 yılında geliştirilmiştir ve bir topluluk yöntemidir. Bu yöntemde ki fikir eğer bir ağaç iyi ise çok ağaç daha iyi olacaktır mantığından gelmektedir [66]. Bu algoritma karar ağaçları kullanarak rastgele bir orman oluşturur. Orman oluşturulurken her bir ağaç torbalama (bagging) yöntemi ile nispeten farklı veriler ile eğitilir. Torbalama yönteminde en önemli amaç farklı sınıflandırıcıların farklı veriler ile eğitilmesidir [66]. Her bir ağaç farklı eğitim kümeleri ile eğitildikleri için yapıları birbirlerinden farklıdır [70]. Fakat rastgeleliğin sağlanması için torbalama tek başına yeterli değildir. Buna ek olarak karar ağaçlarının yapabileceği seçimler sınırlandırılır. Her düğümde ağaca rastgele bir özellik atanarak sadece bu alt kümeden seçim yapabilmesi sağlanır. Her bir karar ağacı bir sınıf oluşturur ve en son tüm karar ağaçlarından gelen bilgiler değerlendirilerek sonuca varılır. Rassal Orman algoritması gücünü birçok karar ağacının birlikte kullanılmasından alır. Bu özelliği sayesinde büyük ölçekli problemlerde kullanılabilmektedir [71]. Şekil 3.10'da Rassal Orman algoritmasının yapısı gösterilmiştir. Eğitim kümesi içerisinden alt eğitim kümeleri oluşturulmaktadır ve bu alt kümeler farklı ağaçlar için eğitim kümeleri olmaktadır.



Şekil 3.10. Rassal orman yapısı [70]

3.2.3. XGBoost algoritması

Yükseltme (boosting), birleşik bir model oluşturabilmek için birçok öğrenme modelini birlikte kullanan toplu öğrenme modelidir [72]. Yükseltme yöntemi, torbalama yönteminden farklı olarak birbirlerini tamamlayan modeller arar ve oluşturur. Bu yöntem, zayıf algoritmaları yinelemeli ve sıralı olarak eğiterek onları güçlendirir. Bir öğrenme modeli bir önceki öğrenme modelinin neyi yapamadığını öğrenir. Sıralı olması torbalama yönteminin aksine modelleri paralel olarak aynı zamanda eğitilemediği anlamına gelmektedir. Bu torbalama yönteminde göre dezavantajlı olduğu bir durumdur. Şekil 3.11'de yükseltme yönteminin akış şeması gösterilmektedir. Her y temel sınıflandırıcısı w ağırlıklarına göre sıra ile eğitilmektedir ve bir önceki modelin performansı ile beslenmektedir. Tüm temel sınıflandırıcılar eğitildikten sonra nihai sonuç için birleştirilirler [73].



Şekil 3.11. Yükseltme sıralı eğitim [73]

Gradyan Yükseltme (Gradient Boosting), yükseltme temelli ve karar ağacı gibi güçsüz tahmin modellerinin sıralı toplu halde oluşturup eğiterek güçlü bir model üreten bir yöntemdir [74]. 2001 yılında Friedman [75] tarafından oluşturulmuştur. Her bir ağaç kayıp fonksiyonunu azaltmak için eklenmektedir ve sıralı olarak eklenen ağaçlar kayıp fonksiyonunu azaltmaktadır. XGBoost, Gradyan Yükseltme algoritması temelli fakat optimizasyon yapılarak performans iyileştirmesi sağlanmıştır. XGBoost açık kaynak kodlu bir algoritmadır ve diğer makine öğrenme algoritmalarına göre on kat daha hızlı çalışabilmektedir [76]. Paralel ve dağınık hesaplama yönetimi ile birlikte çok daha hızlı öğrenme işlemini tamamlamaktadır [77].

3.3 Veri Seti

Bu çalışmada önerilen modelin test edilebilmesi için gerçek trafik verilerinden üretilmiş şifreli trafik veri setine ihtiyaç duyulmuştur. Bunun için Draper-Gil [26] ve arkadaşları tarafından gerçek trafik verilerinden hazırlanan bir veri seti bu çalışmada kullanılmıştır. Oluşturulan veri setinde sınıf etiketi bulunmaktadır. Alice ve Bob adlı iki kullanıcı

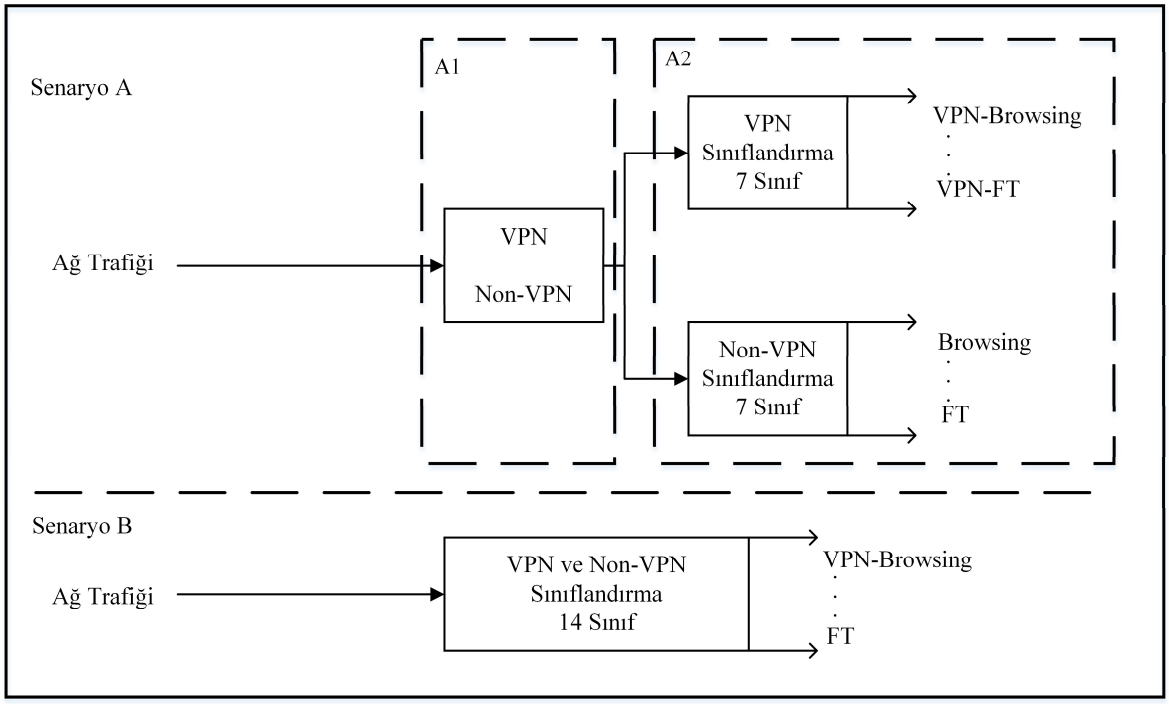
oluşturulmuş ve Simple Mail Transfer Protocol Secure (SMTPS), POP3, Browser, Skype ve benzeri uygulamalar kullanılarak paketler yakalanmıştır. Veri seti içerisinde toplamda 2 adet senaryo bulunmaktadır ve senaryo akışları Şekil 3.12'de gösterilmiştir. Senaryo A'da amaç VPN ile VPN olmayan trafiğin tanımlanması ve bu tanımla işlemi yapıldıktan sonra trafiğin alt sınıflara ayrılarak Browsing, Email, Chat, Streaming, File Transfer, VoIP ve P2P olarak sınıflandırılmasıdır. Bu işlemin yapılabilmesi için Senaryo A 2 farklı adımda yapılmaktadır. Senaryo B'de ise tüm işlem tek sınıflandırma işleminde yapılmakta ve toplamda 14 sınıftan oluşmaktadır.

Veri seti içerisinde 7 adet sınıf etiketi bulunmaktadır. Browsing sınıf etiketi için HTTPS web trafiği üzerinden veriler toplanmıştır. Thunderbird istemci uygulaması aracılığı ile Gmail üzerinden atılan postalar ile email sınıfı oluşturulmuştur. Bu sınıf oluşturulurken SMTPS, POP3/SSL ve İnternet Mesaj Erişim Protokolü (Internet Message Access Protocol, IMAP) protokolleri kullanılmıştır. Chat sınıfı için Facebook, Hangout, Skype ve ICQ uygulamaları kullanılarak mesajlaşma işlemi gerçekleştirilmiş ve veriler toplanmıştır. Youtube ve Vimeo web sayfalarından izlenen videolar üzerinden Streaming sınıf etiket verileri toplanmıştır. File transfer sınıfı için SSH Dosya Transfer Protokolü (Secure File Transfer Protocol, SFTP) ve SSL Dosya Transfer Protokolü (File Transfer Protocol over SSL, FTPS) protokolleri aracılığı ile dosya paylaşımı yapılarak veriler oluşturulmuştur. Yine Facebook, Hangouts ve Skype üzerinden yapılan sesli görüşmeler ile VoIP sınıfı oluşturulmuştur. P2P sınıfı, uTorrent ve Bittorrent uygulamaları kullanılarak paylaşılan veriler üzerinden hazırlanmıştır [26]. Kullanılan uygulamalar ve sınıf etiketleri Çizelge 3.1'de gösterilmiştir. Wireshark ve tcpdump uygulamaları üzerinden toplanan verilerin toplam 28 GB boyutundadır.

Çizelge 3.1: Veri seti sınıf ve uygulamaları

Sınıf	Uygulama
Browsing	Chrome ve Firefox
Email	SMTPS, POP3S ve IMAPS
Chat	ICQ, Skype, Facebook ve Hangout
Streaming	Youtube ve Vimeo
FT	Skype, FTPS, SFTP
VoIP	Facebook, Skype ve Hangout sesli çağrı
P2P	uTorrent ve Bittorent

Draper-Gil [26] ve arkadaşları tarafından hazırlanan veri seti içerisinde akış temelli yaklaşıma ek olarak zaman temelli bir yaklaşım da kullanılmıştır. Zaman temelli yaklaşım için her bir senaryo içerisinde 15, 30, 60 ve 120 saniyelik veri setleri oluşturmuştur. Bu zaman dilimleri Alice ve Bob arasında başlayan trafikten sonra geçen süreyi ifade etmektedir. Bu veri setleri aracılığı ile zamana dayalı sınıflandırma da yapılabilmektedir.



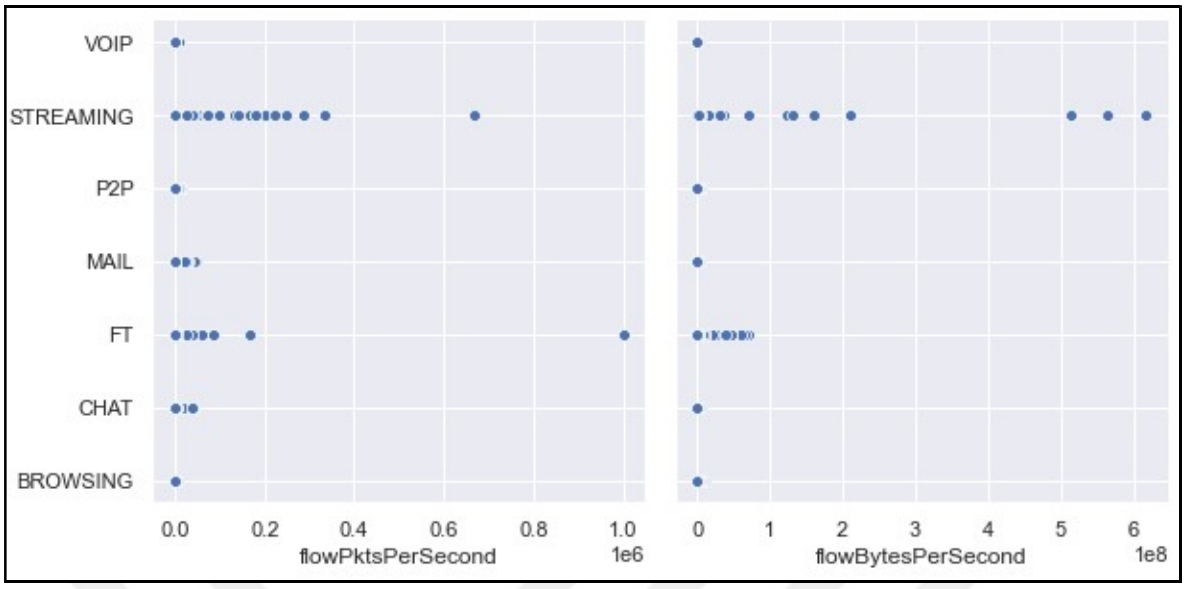
Şekil 3.12. Veri seti senaryoları [26]

Veri seti içerisinde 24 adet özellik olup bunlardan 1 tanesi etiket sınıfıdır. Diğer 3 özellik ise süre, bir saniyede oluşan bit sayısı ve bir saniyede oluşan paket sayısıdır. Geriye kalan 20 özellik ise 5 farkı özelliğin maksimum, minimum, ortalama, standart sapma veya toplam değerlerinden oluşmaktadır. Bu 5 özellik ise ileri yönlü varış zamanı, geri yönlü varış zamanı, akış arası varış zamanı, aktif olarak kullanılan süre ve boşta geçen süre olarak tanımlanmaktadır [26]. Kaynak ip adresi, hedef ip adresi, kaynak port ve hedef port bilgileri özellik olarak kullanılmamaktadır. Çalışma kapsamında veri seti özelliklerinin daha iyi anlaşılabilmesi için kullanılan özelliklerin adları ve açıklamaları Çizelge 3.2'de gösterilmiştir.

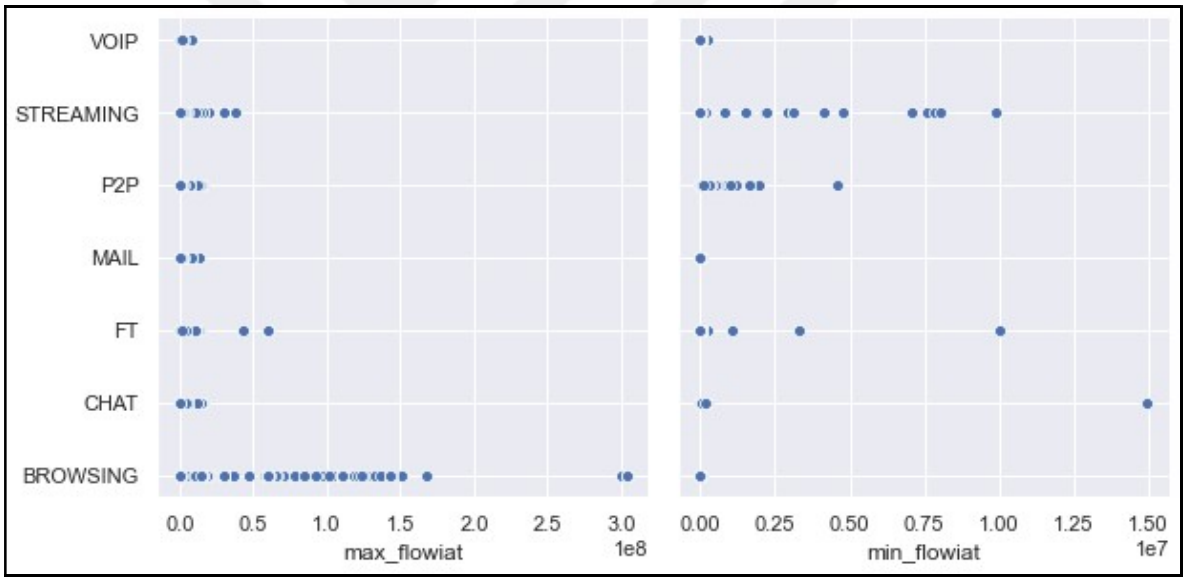
Çizelge 3.2. Özellik adları ve açıklaması

Adı	Açıklaması
duration	Süre
total_fiat	Toplam ileri yönlü varış zamanı
total_biat	Toplam geri yönlü varış zamanı
min_fiat	Minimum ileri yönlü varış zamanı
min_biat	Minimum geri yönlü varış zamanı
max_fiat	Maksimum ileri yönlü varış zamanı
max_biat	Maksimum geri yönlü varış zamanı
mean_fiat	Ortalama ileri yönlü varış zamanı
mean_biat	Ortalama geri yönlü varış zamanı
flowPktsPerSecond	Saniyede giden paket sayısı
flowBytesPerSecond	Saniyede giden bayt sayısı
min_flowiat	Minimum akış varış süresi
max_flowiat	Maksimum akış varış süresi
mean_flowiat	Ortalama akış varış süresi
std_flowiat	Akış arası varış süresinin standart sapması
min_active	Minimum aktif zaman
mean_active	Ortalama aktif zaman
max_active	Maksimum aktif zaman
std_active	Aktif zamanın standart sapması
min_idle	Minimum boşta kalma süresi
mean_idle	Ortalama boşta kalma süresi
max_idle	Maksimum boşta kalma süresi
std_idle	Boşta kalma süresi standart sapması
class1	Sınıf

Veri seti içerisinde bulunan her bir sınıfın kendine özgü bir yapısı bulunmaktadır. Özellikler sınıflara göre veya birbirlerine etkisine göre analiz edilerek veri seti daha iyi anlaşılabilmektedir. Bu amaçla veri seti içerisinde bulunan özelliklerin kendilerine göre ve sınıflara göre dağılımları incelenmiştir. Senaryo A2 VPN olmayan veri seti içerisinde bulunan trafiklerin bir saniyede oluşan paket sayısı ve verinin bit cinsinden değeri Şekil 3.13'de gösterilmiştir. Streaming sınıfı saniye başına en çok paket gönderen ve en çok veri gönderen sınıftır. Onu FT sınıfı izlemektedir. Streaming sınıfı, anlık veri gönderimine sahip olduğu için paket sayısı ve bit sayısının fazla olduğu anlaşılmaktadır. Gönderilen iki paketin zaman aralığının maksimum ve minimum değerlerinin sınıflara göre dağılımı Şekil 3.14'de gösterilmiştir. İki paket arasında geçen en uzun süre Browsing sınıfına ait iken en kısa süre Streaming sınıfına aittir.

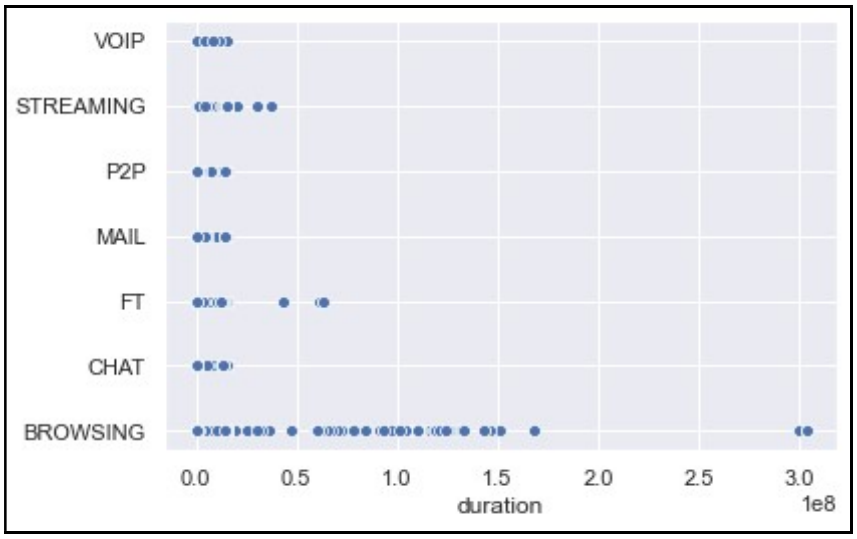


Şekil 3.13. Saniye başına veri ve paket gönderiminin analizi



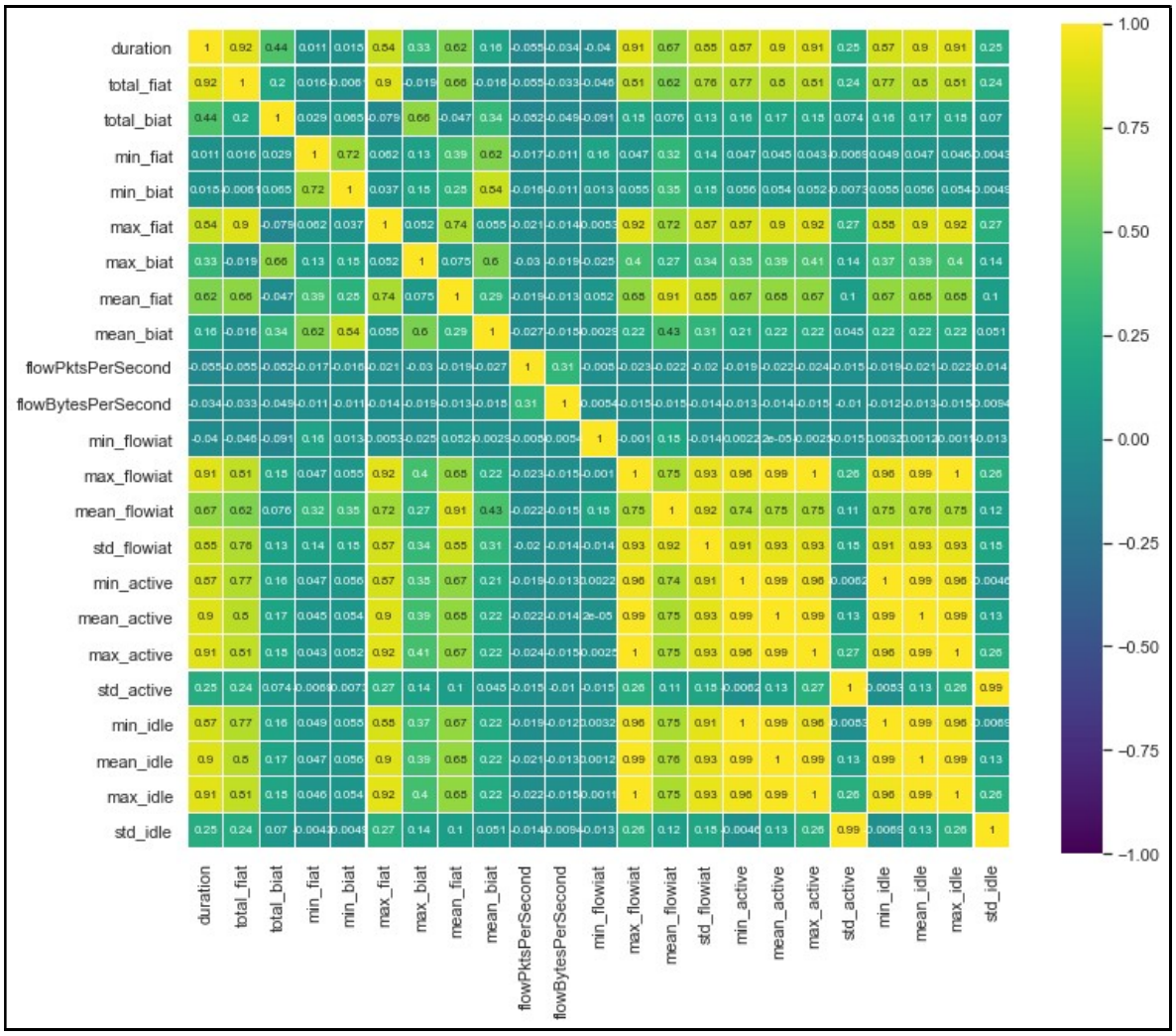
Şekil 3.14. Paketler arası süre analizi

Toplam bağlantı süresinin sınıflara göre analizi Şekil 3.15'de gösterilmiştir. Süre temelli analiz yapıldığında Browsing sınıfının en uzun süreli oturuma sahip olduğu görülmektedir. Browsing sınıfını FT ve Streaming sınıfları izlemektedir.



Şekil 3.15. Toplam süre analizi

Özelliklerin sınıflara göre analizi yapıldıktan sonra özelliklerin birbirleri ile analizi yapılarak aralarında bulunan ilişkiler görülebilmektedir. Bu amaçla özelliklerin sıcaklık harita analizi çıkarılmıştır ve Şekil 3.16'da gösterilmiştir. Bu harita ile birlikte hangi özelliğin hangi özellikler ile ne derece ilişkili olduğu görülebilmektedir. İlişki derecesi 1'den 0'a doğru azalmaktadır. Örnek olarak Total_biat özelliği analiz edildiğinde en yüksek ilişkisinin max_biat özelliği ile olduğu görülmektedir.



Şekil 3.16. Özellikler arasında bulunan ilişkilerin analizi

3.4 Performans Ölçümü

Literatürde önerilen modellerin performans ölçümünün yapılabilmesi için birçok metrik kullanılmaktadır. Bu metriklerin en yaygın kullanılanları doğruluk, hassasiyet, kesinlik ve F-Ölçütü ’dür. Bu metrikler Doğru Pozitif (DP), Yanlış Pozitif (YP), Doğru Negatif (DN) ve Yanlış Negatif (YN) değerleri üzerinden hesaplanmaktadır. Gerçekte anormal olan bir veri model tarafından anormal olarak sınıflandırılıyorsa bu DP ve gerçekte normal olan bir veri model tarafından anormal olarak sınıflandırılıyorsa bu YP anlamına gelmektedir. Normal olan bir veri model tarafından normal olarak sınıflandırılıyorsa bu DN ve anormal olan bir veri model tarafından normal olarak sınıflandırılıyorsa YN olarak kabul edilir.

Literatürde en yaygın kullanılan başarı ölçütü doğruluk ölçütüdür. Doğruluk, doğru sınıflandırılmış verilerin sayısının tüm verilerin sayısına bölümü ile hesaplanmaktadır [66]. Doğruluk ölçütünün hesaplanma formülü Eşitlik 4.1'de gösterilmektedir.

$$Doğruluk = \frac{DP+DN}{Tüm\ Veri\ Seti} \tag{4.1}$$

Doğruluk çok yaygın kullanılmasına rağmen tek başına model performansını ölçmek için yeterli değildir [66]. Eğer pozitif olarak bilinen verilerin sayısının oranı negatif olarak bilinen verilerin sayısına göre çok az olursa doğruluk çok anlamlı bir ölçüt olmayacaktır [78]. Bunun için literatürde doğruluk değerine ek olarak hassasiyet, kesinlik ve f-ölçütü kullanılmaktadır. Hassasiyet diğer bir adı ile doğru pozitif oranı, anormal olarak doğru sınıflandırılmış verilerin bütün anormal verilerin sayısına bölümü ile hesaplanmaktadır [79]. Hassasiyet ölçütünün hesaplanma formülü Eşitlik 4.2'de gösterilmektedir. Kesinlik, anormal olarak doğru sınıflandırılmış verilerin sayısının, model tarafından anormal olarak sınıflandırılmış bütün verilerin sayısına bölümü ile hesaplanmaktadır. Kesinlik ölçütünün hesaplanma formülü Eşitlik 4.3'de gösterilmektedir. F-ölçütü ise kesinlik ve hassasiyet ölçütlerinin harmonik ortalaması ile hesaplanmaktadır. F-ölçütü 'nün hesaplanma formülü Eşitlik 4.4'de gösterilmektedir.

$$Hassasiyet = \frac{DP}{DP+YN} \tag{4.2}$$

$$Kesinlik = \frac{DP}{DP+YP} \tag{4.3}$$

$$F - ölçütü = \frac{2*Hassasiyet*Kesinlik}{Hassasiyet+Kesinlik} \tag{4.4}$$

4. SINIFLANDIRMA MODELİ VE UYGULAMASI

Şifreli internet trafiğinin makine öğrenmesi yöntemi kullanılarak sınıflandırılması için verilerin hazırlanması ve modelin oluşturulması gerekmektedir. Verinin kalitesi ve modelin doğru oluşturulması sistem performansını büyük oranda etkilemektedir. Bu amaçla modelin oluşturulurken yapılan veri önişlemleri ve model oluşturma adımları detaylı olarak anlatılmıştır.

4.1 Veri Önişlemleri

Veri önişleme, verilerin artık makinenin kolayca ayrıştırabileceği bir duruma getirilmesi için dönüştürüldüğü veya kodlandığı adımdır. Veri ön işlemleri, makine öğrenme algoritmalarının sağlıklı çalışabilmesi ve performansının artırılabilmesi için gerekli ve önemlidir. Veri ön işlemlerinde veri temizleme, bütünleşme, azaltma, tamamlama ve veri dönüştürme işlemleri gerçekleştirilmektedir [80].

4.1.1 Özellik etiket kodlama

Veri önişlemleri kapsamında şifreli trafiğin sınıflandırılması için önerilen mimaride ilk olarak veri seti içerisinde bulunan verilere ön işlem yapılmıştır. XGBoost ve Rassal Orman algoritmaları kategorik verileri işleyememektedir. Bunun için veri seti içerisinde sayısal olmayan VPN-Browsing, VPN-FT gibi sınıf etiketleri sayısal değerlere kodlanmış ve dönüştürülmüştür. Örnek olarak bu işlem sonrasında VPN-Browsing için 0, VPN-FT için 6 değeri ataması yapılmıştır. Senaryo B için sayısallaştırma sonrası atama değerleri Çizelge 4.1'de gösterilmiştir.

Çizelge 4.1. Sayısallaştırma sonrası sınıf etiket değerleri

Sınıf Etiketi	Atanan Değer
BROWSING	0
CHAT	1
FT	2
MAIL	3
P2P	4

Çizelge 4.1. (devam) Sayısallaştırma sonrası sınıf etiket değerleri

STREAMING	5
VOIP	6
VPN-BROWSING	7
VPN-CHAT	8
VPN-FT	9
VPN-MAIL	10
VPN-P2P	11
VPN-STREAMING	12
VPN-VOIP	13

4.1.2 Veri normalizasyonu

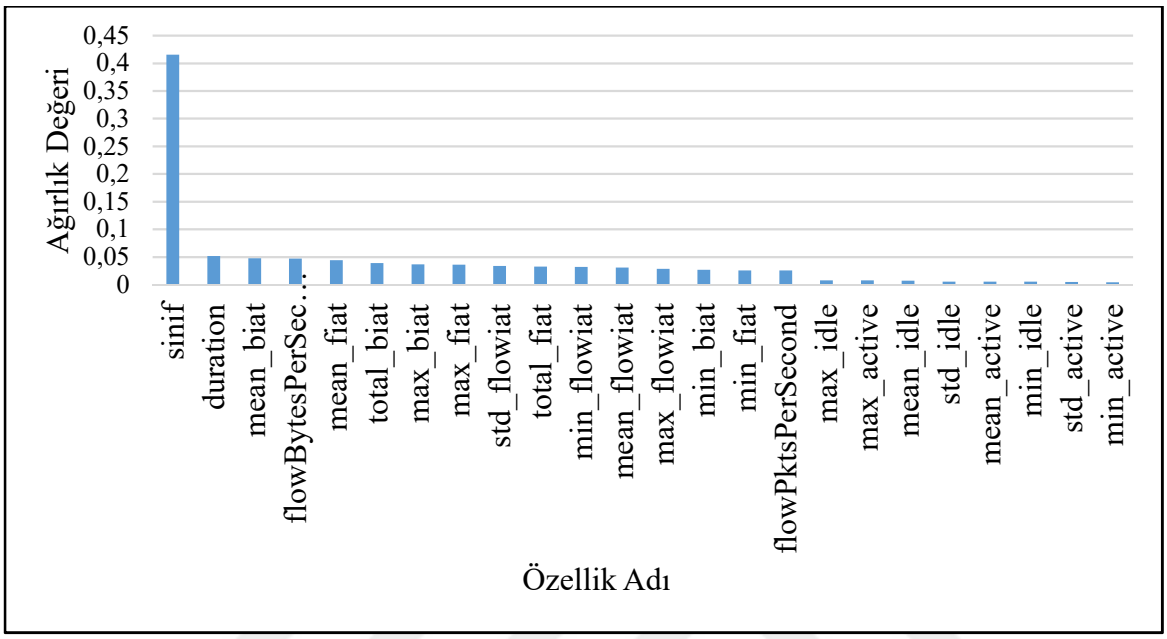
Makine öğrenme algoritmalarının başarısı veri seti içerisinde bulunan verilerin kalitesine bağlıdır [81]. Veri kalitesini ve performansı artırmak için yapılan işlemlerden biri veri normalizasyonudur. Veri normalizasyonu, büyük sayısal özelliklere sahip verilerin küçük sayısal özelliklere sahip veriler üzerinde hâkim olmasını engellemek için ortak aralıktaki bir özelliğe dönüştürülmesi işlemidir. Bu sayede veri seti içerisinde bulunan bütün özelliklerin sonuca etkileri artırılmaktadır [82]. Standart ölçeklendirme verilerin ortalamasını çıkarır ve birim varyansa göre ölçekleyerek özellikleri standart bir hale çevirir. Birim varyansı, bütün değerlerin standart sapmaya bölünmesi anlamına gelmektedir. Sayısallaştırma işlemi sonrasında standart ölçeklendirme yöntemi kullanılarak veri seti içerisinde bulunan değerlerin 0 ile 1 arasında bir değere indirgenmesi sağlanmıştır.

Veri önişleme tamamlandıktan sonra veriler kullanıma hazır duruma gelmiştir.

4.2 Özellik Seçimi

Özellik seçimi, verilerin özellikleri içerisinden en iyi alt kümeyi bularak gereksiz ve model için anlamları olmayan verilerin kaldırılmasıdır [83]. Özellik seçiminin amacı sistem başarı oranını değiştirmeden modelin hesaplama performansının iyileştirilmesidir [84]. Çalışmamızda sınıflandırma işlemi yapmadan önce özellik seçimi yapılmıştır. Bu kapsamda

kullanılan veri seti içerisindeki veriler önem derecesine göre ağırlıklandırılmıştır. Veri seti içerindeki özelliklerin ağırlık değerleri toplamı 1'e eşittir. Veri seti içerinde bulunan özelliklerin ağırlıklandırılmış grafiği Şekil 4.1'de gösterilmiştir.



Şekil 4.1. Veri seti özelliklerin ağırlık değerleri

Özellikler ağırlıklarına bakılarak analiz edildiğinde aktif geçen süre ve boşta bekleme sürelerinin sınıflandırma için çok önemli olmadığı görülmektedir. Bu tüm sınıflar için değerlerin birbirine benzediği ve düşük bilgi kazanımı sağlayarak modelin başarısını etkilemediği görülmüştür. En çok bilgi kazanımının toplam süre ve ileri veya geri yönlü varış zaman aralıkları özelliklerinin sağladığı görülmüştür.

Ağırlık değeri %2'nin altında olan ve sınıflandırmayı en az etkileyen 8 özellik çıkarılarak toplamda 16 adet özellik seçilmiştir. Seçilen özellikler ve ağırlık değerleri Çizelge 4.2'de gösterilmiştir.

Çizelge 4.2. Seçilen özellikler

Özellik Adı	Ağırlık Değeri
class	0,415323
duration	0,051761
mean_biat	0,047395
flowBytesPerSecond	0,047213
mean_fiat	0,044046
total_biat	0,038946
max_biat	0,036594
max_fiat	0,035938
std_flowiat	0,033617
total_fiat	0,032499
min_flowiat	0,032022
mean_flowiat	0,03083
max_flowiat	0,028706
min_biat	0,026907
min_fiat	0,025605
flowPktsPerSecond	0,025455

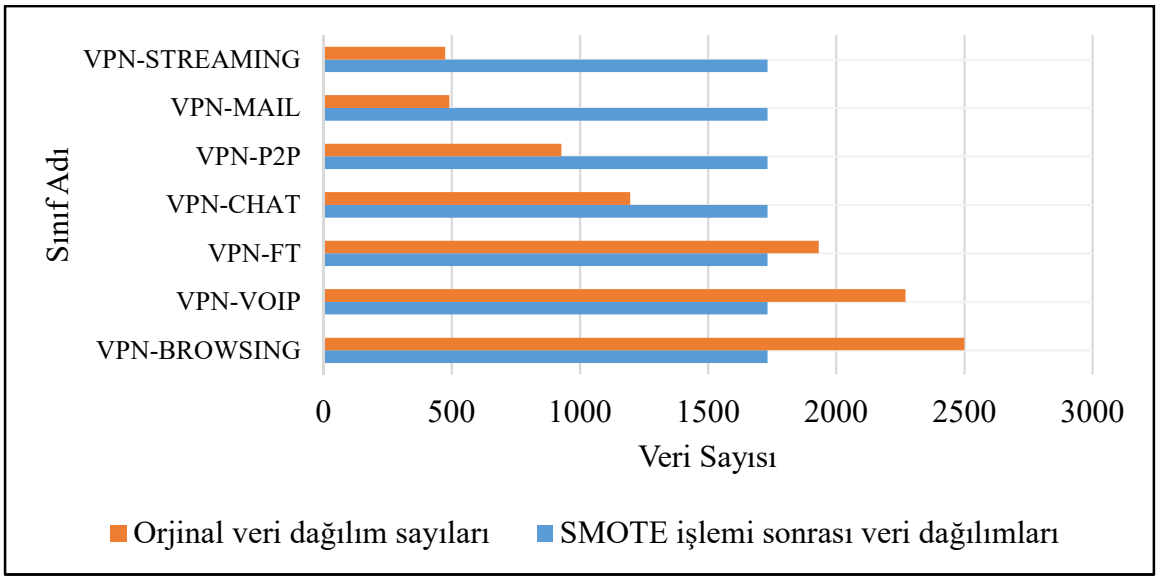
4.3 Eğitim ve Test Küme Ayrımı

Önerilen modelin mevcut problemi çözebilmesi için öğrenme işlemi gerçekleştirmesi gerekmektedir. Bu amaçla modeline eğitilebilmesi için eğitim verilerine ve modelin başarısının test edilebilmesi için test verilerine ihtiyaç duyulmaktadır. Bu kapsamda mevcut veri seti içerinden %30'luk kısmı test verisi olarak ayrılmış ve eğitim geriye kalan %70'lik veri üzerinden yapılmıştır. Test aşamasında sadece test verisi kullanılması gerekmektedir. Çünkü eğitim aşamasında kullanılan eğitim verileri model tarafından hatırlanmaktadır ve eğer bu veriler kullanılırsa sistem eğitim seti içerisinde bulunan verileri doğru etiketleyecektir. Bu da sistemin gerçek performansını etkileyecek ve yanlış sonuçlar verecektir [63].

4.4 Veri Dengeleme

Makine öğrenmesi çözümlerinin bir problemi de dengesiz veri setleridir. Veri seti içerisinde az ve düzensiz bulunan veriler az oldukları için tespit edilmeleri ve sınıflandırılmaları daha zordur [85]. Kullanılan veri seti üzerinde bulunan verilerin sınıf dağılımlarında dengesizlik bulunmaktadır. Bu dengesizlik nedeni ile az örneğe sahip sınıfların tespit edilmesi daha zor

olmaktadır. Bu nedenle veri dağılımında bulunan dengesizliğin giderilmesi için Synthetic Minority Oversampling Technique (SMOTE) [86] algoritması kullanılarak veri dengeleme işlemi yapılmıştır. SMOTE işlemi, az sayıda olan sınıflar için yukarı yönlü örnekleme ve çok sayıda bulunan sınıflar için aşağı yönlü örnekleme yaparak tüm sınıfları ortak bir sayıda olmasını sağlamaktadır. Veri dengeleme işlemi sonrasında az sayıda bulunan verilerin daha başarılı tespit edilerek model performansının arttığı görülmüştür. SMOTE algoritması kullanılmadan önce Senaryo 1 VPN veri sayıları ve SMOTE algoritması kullanıldıktan sonra oluşan veri sayıları Şekil 4.2'de gösterilmiştir.

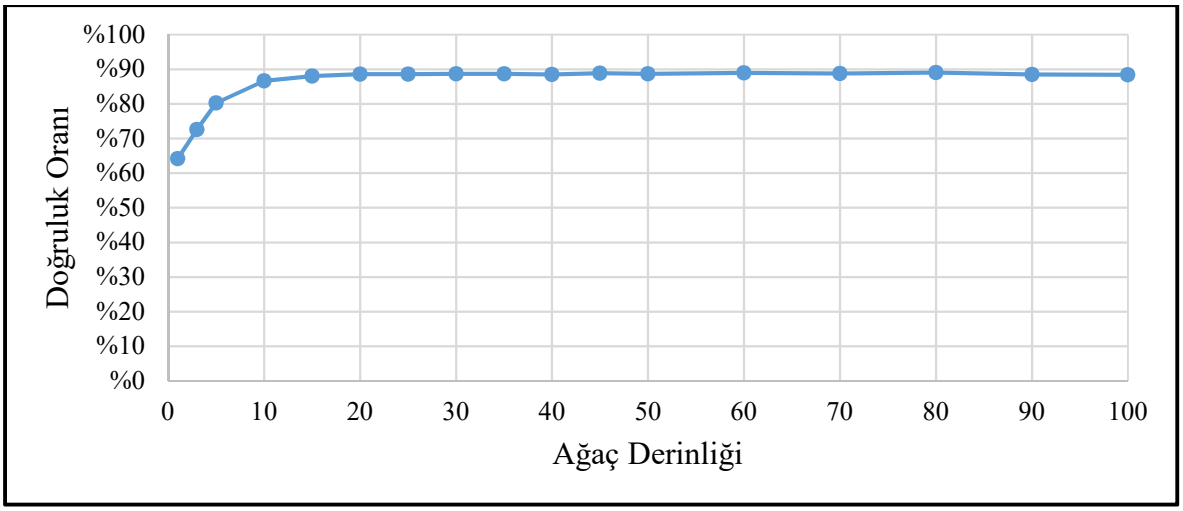


Şekil 4.2. SMOTE öncesi ve sonrası veri sayıları

4.5 Hiper Parametre Seçimi

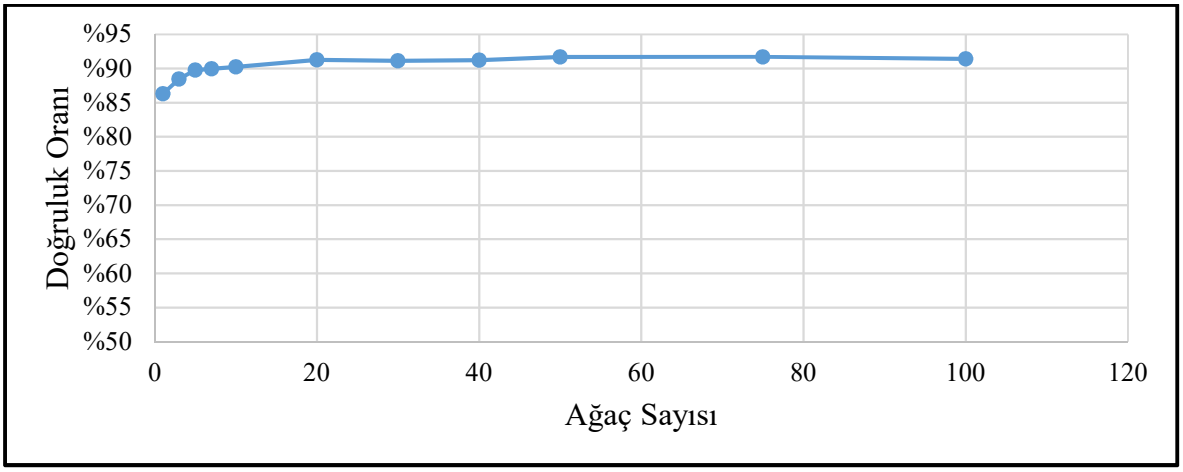
Hiper parametreler makine öğrenmesi modelini konfigüre etmek için veya kayıp fonksiyonunu minimize etmek amaçlı algoritmaları özelleştirmek için kullanılır [87]. Her makine öğrenmesi sistemi hiper parametrelere sahiptir ve bu parametrelerin belirlenmesi işlemi sistem başarısı ve performansı açısından çok önemlidir. Modelin optimum başarıya ulaşması için kullanılacak parametreler belirli aralıklar ile tanımlanır [87]. Aralıkta bulunan her bir parametreye aday parametre adı verilmektedir. Her bir aday parametrenin değerlendirilerek parametrelerin belirlenmesi için modelin sürekli eğitilmesi ve test edilmesi gerekmektedir. Bu işlem için yoğun hesaplama işlemleri ve zaman gerekmektedir [88]. Parametrelerin otomatik olarak belirlenmesi insan gücünü azaltmakla birlikte sistem performansını da artırmaktadır [89].

Çalışmamızda algoritmaların kullanımından önce kurulan sistemin en iyi performansı verebilmesi için hiper parametre optimizasyon işlemi gerçekleştirilmiştir. Hiper parametrelerin belirlenmesi için ızgara arama (grid search) yöntemi kullanılmıştır. Izgara arama yöntemi ile hiper parametreler belirlenirken parametreler için belirli bir aralık değeri girilir [90]. Bu amaçla çalışmamızda algoritmaların kullandığı parametre değerleri belirli bir aralıkta ve formatta verilerek en iyi performans sağlayan parametreler seçilmiştir. Her bir algoritma için parametre seçimi ayrı olarak yapılmıştır. Şekil 4.3'de Karar Ağacı algoritmasında ağaç derinliğinin doğruluk oranına etkisinin grafiği gösterilmiştir. Ağaç derinlik değerinin 20'nin üzerinde doğruluk oranını etkilemediği görülmektedir.

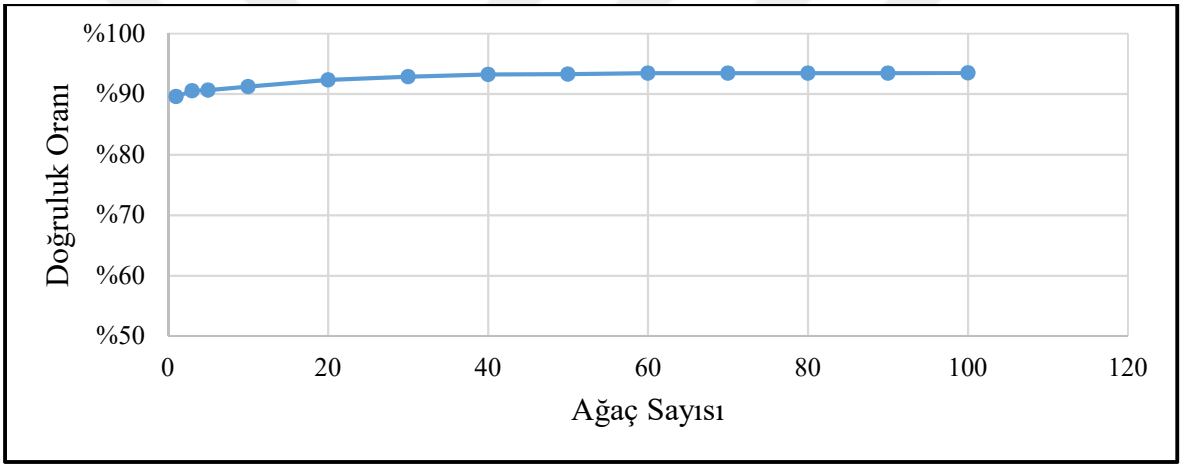


Şekil 4.3. Karar ağacı algoritması ağaç derinlik ve doğruluk analizi

Rassal Orman algoritmasında kullanılan ağaç sayısının doğruluk oranı üzerindeki etkisini gösteren grafik Şekil 4.4'de gösterilmiştir. Ağaç sayısı 50 değerine geldiğinde en yüksek başarı oranı elde edilmiştir. Ağaç sayısının XGBoost algoritmasının performansı üzerindeki etkisi Şekil 4.5'da gösterilmiştir. Ağaç sayısı 60 olduğunda en yüksek doğruluk oranı elde edilmiştir.



Şekil 4.4. Rassal Orman algoritması ağaç sayısı ve doğruluk analizi



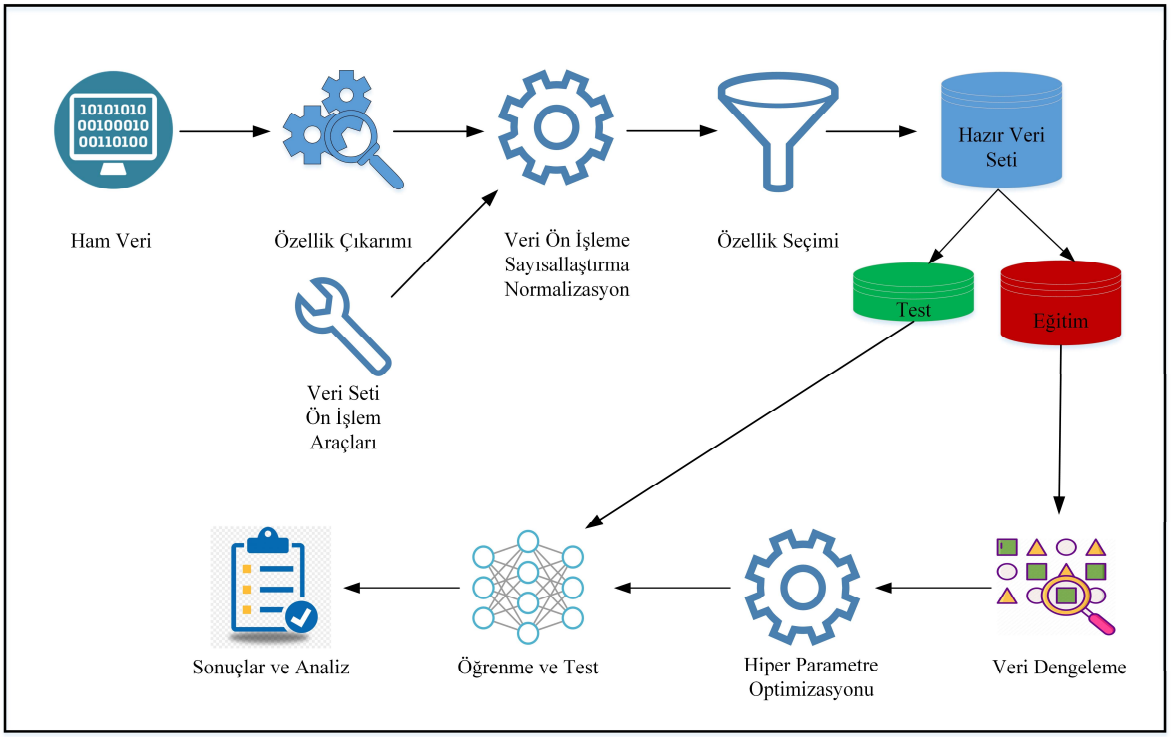
Şekil 4.5. XGBoost algoritması ağaç sayısı ve doğruluk analizi

Izgara yönteminde tek başına derinlik veya ağaç sayısı kullanılmamaktadır. Parametreler belirlenirken derinlik ile birlikte kullanılan parametreler ve işlem sonucunda seçilen değerler Çizelge 4.3'de gösterilmiştir. Bu parametreler haricinde kullanılan parametreler varsayılan değerleri ile kullanılmıştır.

Çizelge 4.3. Parametreler ve değerleri

Algoritma	Parametre Adları	Değerleri
Karar Ağacı	class_weight	balanced
	max_depth	20
	max_features	auto
	min_samples_leaf	1
	min_samples_split	2
	splitter	best
Rassal Orman	n_estimators	300
	criterion	entropy
	class_weight	balanced_subsample
	max_features	sqrt
XGBoost	eta	0.2
	max_depth	50
	gamma	2
	min_child_weight	4
	subsample	0.83
	max_bin	256
	objective	multi softmax
	tree_method	hist
	num_class	7
	learning_rate	0.1

Parametre optimizasyonu yapıldıktan sonra model eğitim verileri ile eğitilmiş ve test verileri ile başarısı test edilmiştir.



Şekil 4.6. Önerilen sistem mimarisi

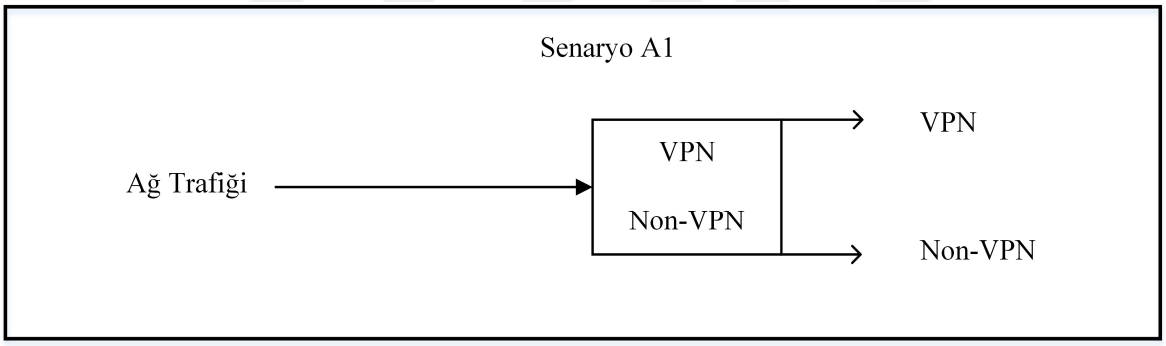
Önerilen modelin aşamaları Şekil 4.6'da gösterilmiştir. Ham verilerden özellik çıkarımı yapıldıktan sonra veriler sayısallaştırma ve normalizasyon süreçlerinden geçmektedir. Bu işlemlerden sonra özellik seçimi yapılır ve veri seti test ve eğitim kümelerine bölünür. Veri seti içerisinde bulunan dengesizlik örnekleme yöntemleri ile giderilmiştir. Son olarak, makine öğrenmesi algoritmaları için hiper parametreler seçilerek sınıflandırma işlemi gerçekleştirilir.

5. BULGULAR

Çalışma kapsamında kullanılan veri seti içerinde iki farklı senaryo bulunmaktadır. Bununla birlikte veri seti 15 saniye, 30 saniye, 60 saniye ve 120 saniye olarak kaydedilen verilerden oluşmaktadır. Üç farklı algoritma her bir senaryo için ayrı ayrı test edilmiş ve sonuçlar karşılaştırılmıştır. Çoklu sınıflandırmalarda kesinlik, hassasiyet ve f1-ölçüt metrikleri için ağırlıklı ortalama değerleri verilmiştir.

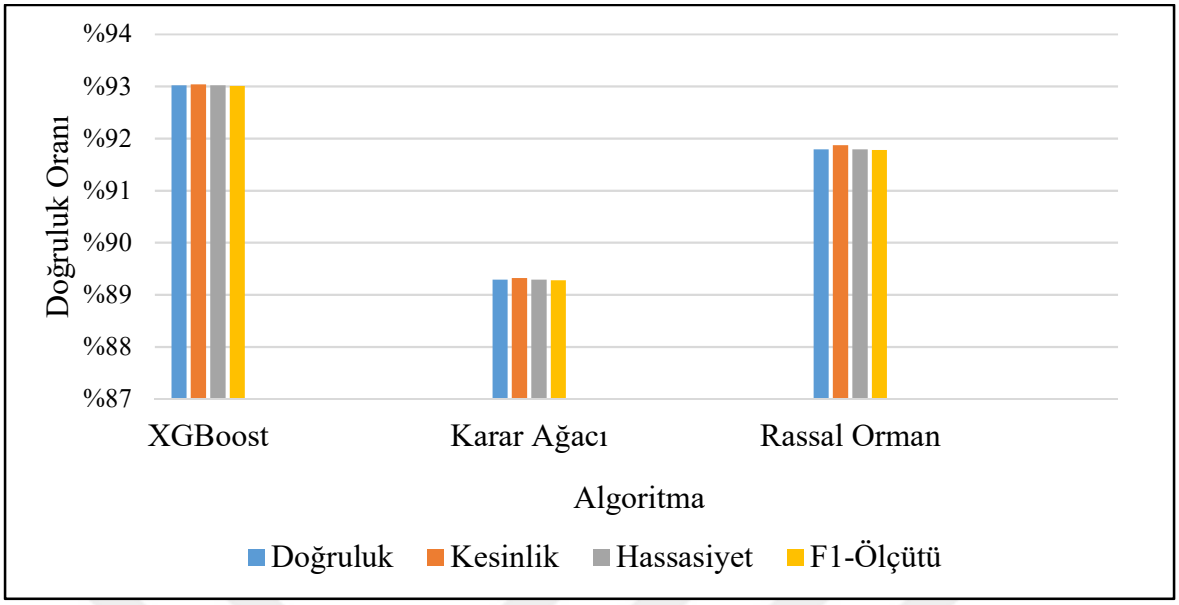
5.1. Senaryo A1

Senaryo A içerisinde iki farklı alt senaryo bulunmaktadır. Bu alt senaryolardan ilkinde gelen trafik paketlerinin VPN ile mi yoksa VPN olmadan mı şifrelendiğinin sınıflandırılması işlemi yapılmaktadır. Bu aşamada ikili sınıflandırma işlemi yapılmaktadır ve Şekil 5.1'de gösterilmektedir.



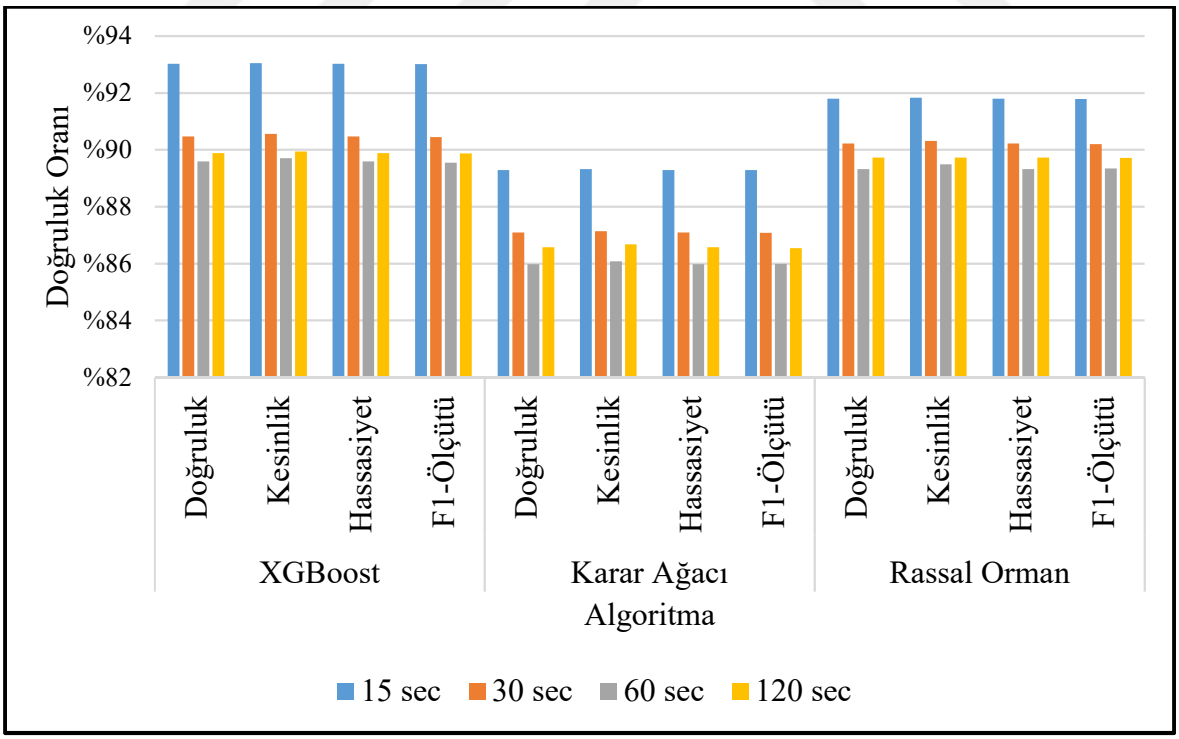
Şekil 5.1. Senaryo a1 mimarisi

Bu senaryo için veri setinde 15 saniye, 30 saniye, 60 saniye ve 120 saniyeden oluşan zaman temelli dört adet veri seti bulunmaktadır. Bu veri setlerinde VPN için 9793 adet ve Non-VPN için 8965 adet veri bulunmaktadır. Bu senaryonun tüm zaman dilimleri için XGBoost algoritması en başarılı sonucu verdiği görülmüştür. Şekil 5.2'de üç farklı algoritma için doğruluk, kesinlik, hassasiyet ve f1-ölçüt değerleri gösterilmektedir. En yüksek başarı oranı 15 saniyelik veri setinde yakalanmıştır. Bu veri setinde en iyi doğruluk oranı %93,02, kesinlik %93,04, hassasiyet %93,02 ve f1-ölçütü %93,01'dir.



Şekil 5.2. Senaryo a1 başarı oranları

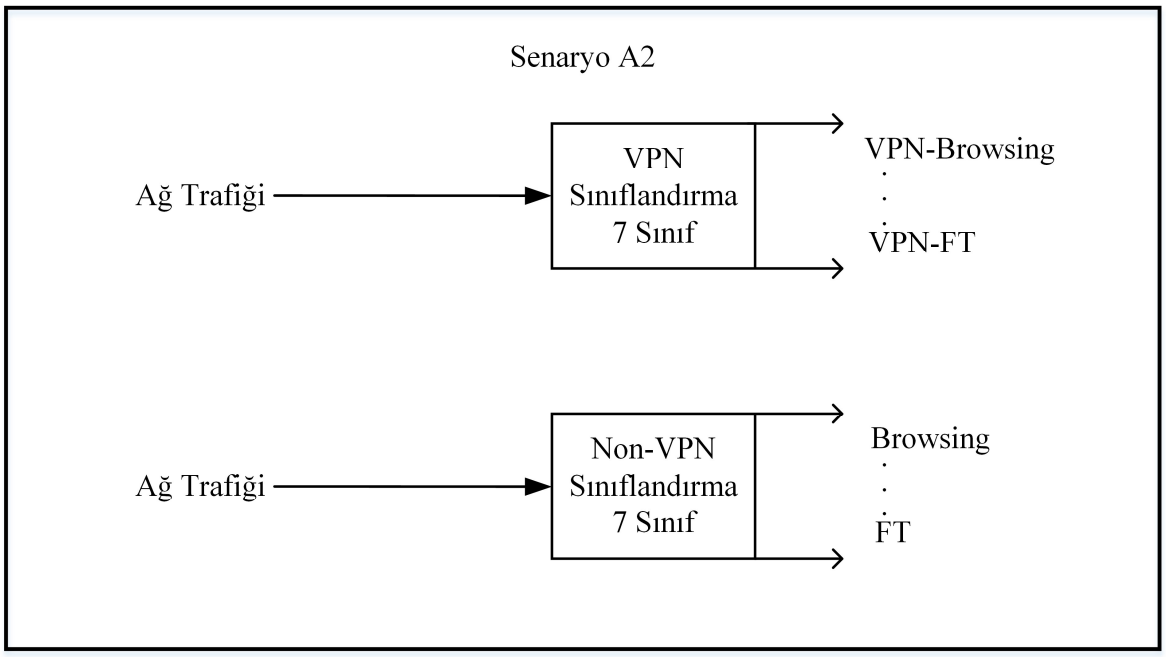
Zaman bazlı analiz yapıldığında ise 15 saniyeden 120 saniyeye doğru modelin başarı oranın azaldığı tespit edilmiştir. Zaman bazlı değişim grafiği Şekil 5.3'de gösterilmiştir.



Şekil 5.3. Senaryo a1 zaman bazlı başarı oran değişimi

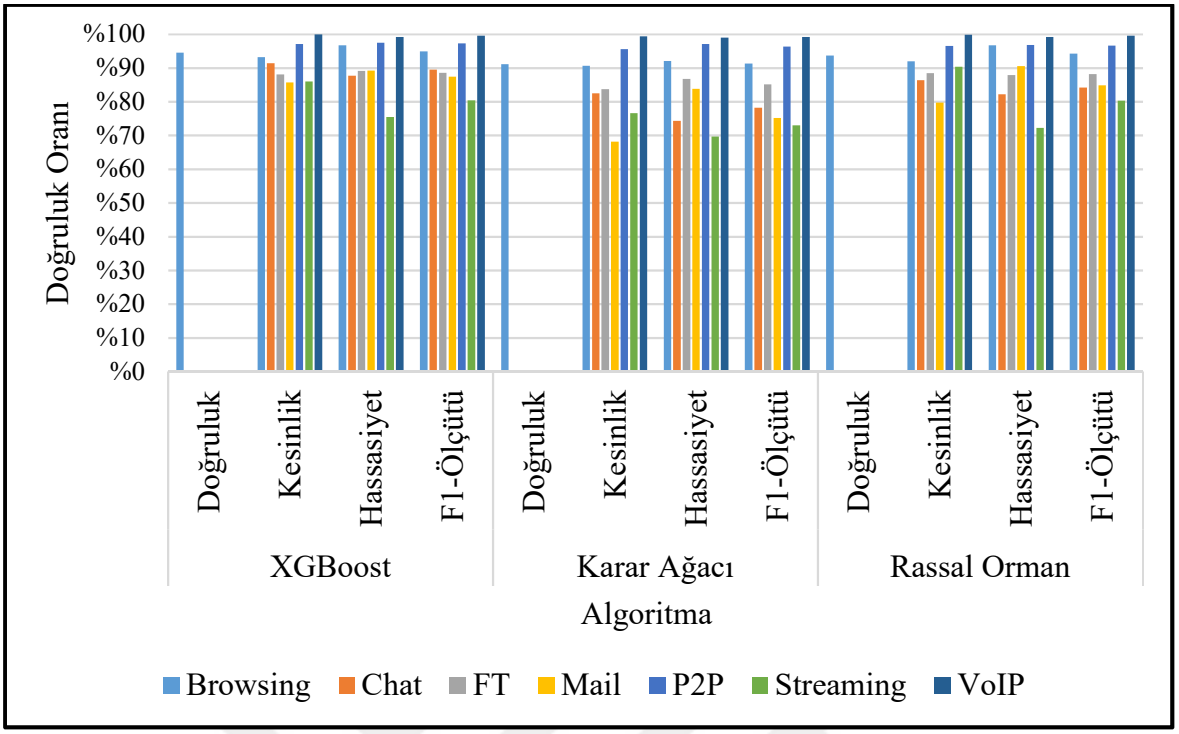
5.2. Senaryo A2

Senaryo A2'de, Senaryo A1'den gelen ve VPN ile VPN olmadan şifrelenmiş trafiklerin çoklu sınıflandırma işlemi ile 7 farklı kategoride sınıflandırma işlemi yapılmıştır. Senaryo A2 mimarisi şekil 5.4'de gösterilmiştir. Bu senaryoda VPN için başarı ölçütleri ve VPN olmadan şifrelenmiş trafiğin başarı ölçütleri ayrı ayrı verilmiştir.



Şekil 5.4. Senaryo a2 mimarisi

Senaryo A2 VPN olmayan şifreli trafik için en başarılı sonucu XGBoost algoritması verdiği tespit edilmiştir. XGBoost algoritması doğruluk oranı %94,53'dür. Ağırlıklı ortalama değeri alınmış kesinlik değeri %94,5, hassasiyet değeri %94,53 ve f1-ölçüt değeri %94,48'dir. Sınıf bazlı bakıldığında en yüksek başarı oranlı tespit edilebilen kategori VoIP kategorisidir. En düşük başarı oranına sahip kategori ise Mail kategorisi olduğu görülmüştür. Senaryo A2 15s VPN olmadan şifrelenen trafiğe ait başarı oranları Şekil 5.5'de gösterilmiştir.



Şekil 5.5. Senaryo a2 Non-VPN 15s başarı oranları

Senaryo A2 Non-VPN senaryosu için XGBoost algoritması kullanılarak oluşturulan karışıklık matrisi Çizelge 5.1'de verilmiştir. Bu çizelgeye göre analiz yapıldığında Chat sınıfının en çok Browsing sınıfı ile karıştırıldığı, yine en düşük başarı oranlarından birine sahip Streaming sınıfının FT ve Browsing sınıfları ile karıştırıldığı görülmektedir.

Çizelge 5.1. XGBoost senaryo a2 15s Non-VPN karışıklık matrisi

Sınıflar	Browsing	Chat	FT	Mail	P2P	Streaming	VoIP
Browsing	740	3	3	2	2	7	0
Chat	34	229	2	0	4	1	0
FT	10	8	267	2	1	6	1
Mail	2	0	5	62	0	0	0
P2P	1	3	0	0	283	0	0
Streaming	19	3	21	2	0	98	0
VoIP	0	1	7	0	0	0	861

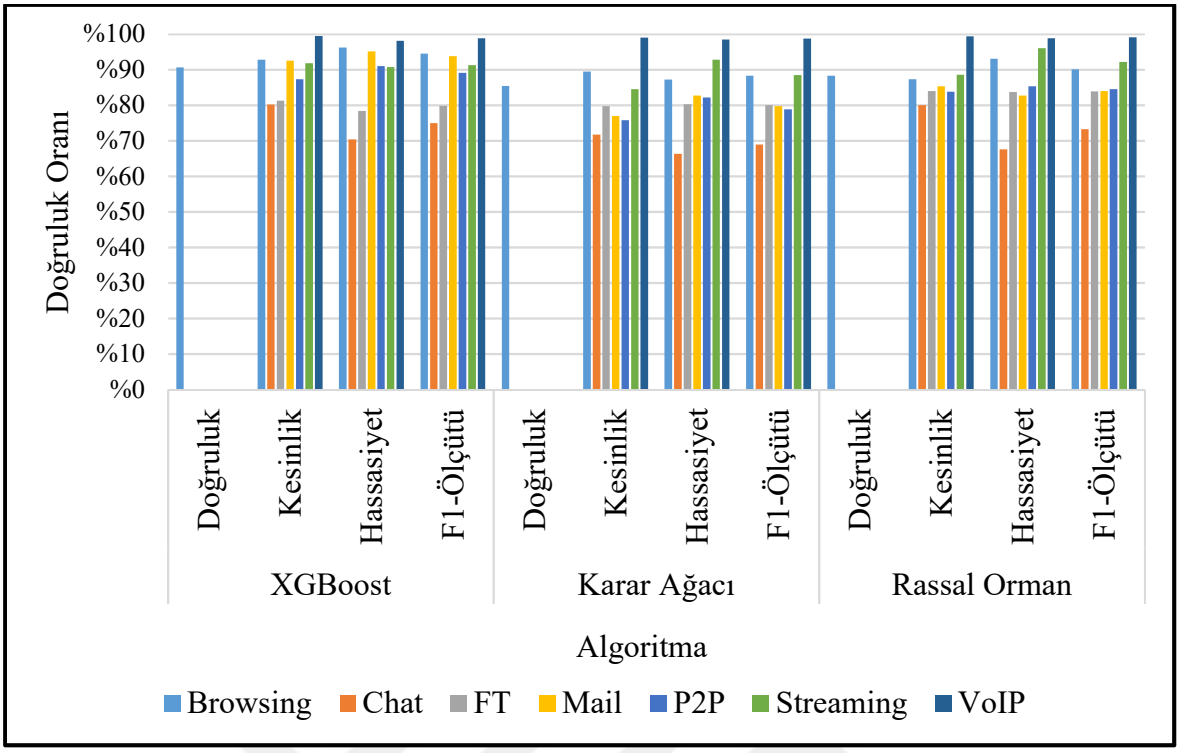
Zaman bazlı analiz yapıldığında ise 15 saniyeden 120 saniyeye doğru modelin başarı oranın azaldığı tespit edilmiştir. Zaman 15 saniyeden 120 saniyeye doğru giderken Browsing sınıfının başarı oranı %94,71'den %96,01'e artmaktadır. Browsing haricinde bulunan diğer tüm sınıfların ise başarı oranı zaman arttıkça düşmektedir. Chat sınıfı zaman arttıkça başarı

oranı en çok düşen sınıf olup 120 saniyedeki başarı oranı 15 saniye olan başarı oranına göre %13 daha düşüktür. Streaming sınıfı, Chat sınıfından sonra en çok başarı oranı düşen sınıf olup %7 daha düşük sonuç elde edilmiştir. Çizelge 5.2'de XGBoost algoritmasının zaman bazlı f1-ölçüt başarı metrikleri gösterilmiştir.

Çizelge 5.2. Senaryo a2 Non-VPN zaman bazlı sınıfların f1-ölçüt değerleri

Sınıflar	15 Saniye	30 Saniye	60 Saniye	120 Saniye
Browsing	0,9471	0,9517	0,9517	0,9601
Chat	0,8889	0,8689	0,8689	0,75
FT	0,8889	0,8627	0,8627	0,8717
Mail	0,8356	0,8627	0,8627	0,8291
P2P	0,9746	0,9373	0,9373	0,9477
Streaming	0,7826	0,783	0,783	0,7107
VoIP	0,9942	0,9913	0,9913	0,9558

Senaryo A2 VPN şifreli trafik için en başarılı sonucu XGBoost algoritması verdiği tespit edilmiştir. XGBoost algoritması doğruluk oranı %90,76'dir. Ağırlıklı ortalama değeri alınmış kesinlik değeri %90,75, hassasiyet değeri %90,76 ve f1-ölçüt değeri %90.73'dür. Sınıf bazlı incelendiğinde en yüksek başarı oranı ile tespit edilebilen kategori yine VoIP kategorisidir. Fakat bu senaryoda en düşük başarı oranına sahip kategori ise Chat kategorisidir. Senaryo A2 30s VPN ile şifrelenen trafiğe ait başarı oranları Şekil 5.6'da gösterilmiştir.



Şekil 5.6. Senaryo a2 VPN 30s başarı oranları

Senaryo A2 VPN senaryosu için XGBoost algoritması kullanılarak oluşturulan karışıklık matrisi Çizelge 5.3'de verilmiştir. Bu çizelgeye göre analiz yapıldığında Browsing sınıfının en çok Chat ve FT sınıfı ile karıştırıldığı, yine en düşük başarı oranlarından birine sahip Chat sınıfının Browsing, FT, Mail ve P2P sınıfları ile karıştırıldığı görülmektedir.

Çizelge 5.3. XGBoost senaryo a2 30s VPN karışıklık matrisi

Sınıflar	Browsing	Chat	FT	Mail	P2P	Streaming	VoIP
Browsing	786	33	26	1	6	0	0
Chat	15	187	26	10	12	1	2
FT	23	21	285	11	11	3	2
Mail	0	4	7	278	0	0	0
P2P	2	4	7	1	251	14	0
Streaming	0	3	0	0	1	83	0
VoIP	0	1	4	0	4	0	428

Zaman bazlı analiz yapıldığında ise modelin 30 saniyede en yüksek başarı oranını sağladığı tespit edilmiştir. Browsing sınıfı 30 saniye veri setinde %94,52 ile en yüksek başarı oranına sahipken 15 saniye veri setinde %91,13 ile en düşük başarı oranına sahiptir. Chat sınıfı için

en yüksek başarı oranına 15 saniye veri setinde, en düşük başarı oranına ise 120 saniye veri setinde elde edilmiştir. Zaman arttıkça Chat sınıfının tespiti büyük oranda düşmektedir. FT sınıfı %14 düşüş oranı ile en büyük farka sahiptir. Mail sınıfı için en yüksek tespit oranı 30 saniye veri setindedir ve 15 saniye veri setinde en düşük başarı oranı elde edilmiştir. Çizelge 5.4'de XGBoost algoritmasının zaman bazlı f1-ölçüt başarı metrikleri gösterilmiştir.

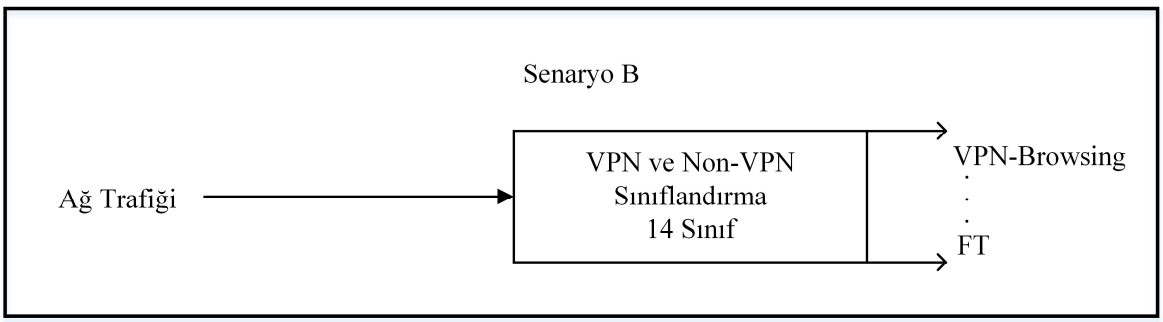
Çizelge 5.4. Senaryo a2 VPN zaman bazlı sınıfların f1-ölçüt değerleri

Sınıflar	15 Saniye	30 Saniye	60 Saniye	120 Saniye
Browsing	0,9113	0,9452	0,9217	0,9171
Chat	0,7531	0,7495	0,6712	0,6404
FT	0,8499	0,7983	0,7125	0,698
Mail	0,8704	0,9386	0,8136	0,915
P2P	0,8547	0,8912	0,8566	0,8616
Streaming	0,926	0,9133	0,8247	0,8462
VoIP	0,9925	0,9885	0,9448	0,9544

Senaryo A2 Non-VPN ile VPN sonuçları karşılaştırıldığında modelin Non-VPN trafikler için daha yüksek başarı oranına sahip olduğu açıkça görülebilmektedir. VPN ile şifrelenmiş trafik kapsüllemeye maruz kaldığı için sınıflandırma işlemi daha zor olmaktadır.

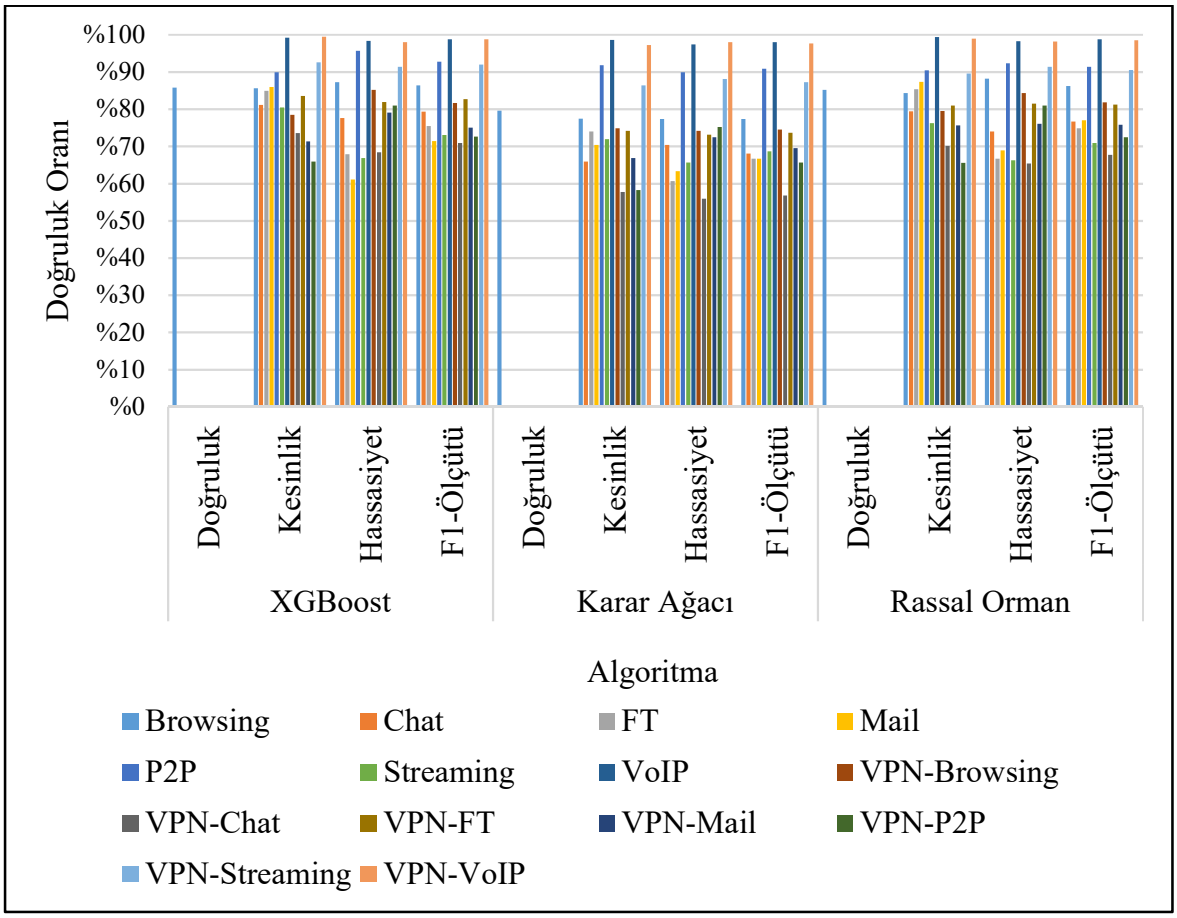
5.3. Senaryo B

Senaryo B'de VPN ve Non-VPN ayrımı yapılmadan 14 sınıf arasından çoklu sınıflandırma işlemi yapılmıştır. Senaryo B mimarisi Şekil 5.7'de gösterilmiştir.



Şekil 5.7. Senaryo b mimarisi

XGBoost algoritması diğer senaryolarda da olduğu gibi bu senaryo için de en başarılı sonuçları vermiştir. XGBoost algoritması doğruluk oranı %85,79, ağırlıklı ortalama değeri alınmış kesinlik değeri %86,07, hassasiyet değeri %85,76 ve f1-ölçüt değeri %85,75'dir. Senaryo B'de en düşük başarı oranı ile tespit edilen sınıf VPN-P2P ve en yüksek başarı oranı ile tespit edilen sınıf ise VPN-VoIP sınıfıdır. Senaryo B 15s VPN ile şifrelenen trafiğe ait başarı oranları Şekil 5.8'de gösterilmiştir.



Şekil 5.8. Senaryo b VPN 15s başarı oranları

Senaryo B 15 saniye veri seti için XGBoost algoritması kullanılarak oluşturulan karışıklık matrisi Çizelge 5.5'de verilmiştir. Bu çizelgeye göre analiz yapıldığında Browsing sınıfının en çok VPN-Browsing sınıfı ile karıştırıldığı görülmektedir. Browsing sınıfının hem VPN hem de VPN olmadan şifrelenen trafik için benzer davranışlar sergilemesinden kaynaklı birbirleri ile karıştığı tespit edilmiştir. En düşük başarı oranlarından birine sahip VPN-Mail sınıfının VPN-FT sınıfı ile ve VPN-P2P sınıfının ise P2P ve VPN-FT sınıfları ile karıştırıldığı görülmektedir.

Çizelge 5.5. XGBoost senaryo b 15s VPN karışıklık matrisi

Tür	Non-VPN							VPN						
Sınıflar	Browsing	Chat	FT	Mail	P2P	Streaming	VoIP	Browsing	Chat	FT	Mail	P2P	Streaming	VoIP
Browsing	716	5	7	2	2	3	0	77	7	2	0	0	0	0
Chat	14	215	0	0	1	3	0	22	12	5	1	3	1	0
FT	12	4	226	2	2	3	0	7	3	8	8	55	2	1
Mail	0	0	2	55	0	0	0	1	4	2	22	4	0	0
P2P	3	2	0	0	312	0	0	4	0	2	0	3	0	0
Streaming	15	0	2	1	0	107	0	3	0	6	1	22	3	0
VoIP	2	0	4	0	0	0	903	0	2	1	1	4	0	1
VPN-Browsing	52	10	3	0	4	0	0	685	26	22	1	1	0	0
VPN-Chat	7	16	7	0	0	3	1	38	273	31	5	16	1	1
VPN-FT	8	12	9	1	3	5	0	28	33	552	13	10	0	0
VPN-Mail	1	0	4	3	0	2	0	1	6	14	132	2	1	1
VPN-P2P	3	1	0	0	22	3	0	7	5	14	1	251	3	0
VPN-Streaming	3	0	0	0	0	4	0	0	0	0	0	6	138	0
VPN-VoIP	0	0	2	0	1	0	6	0	0	2	0	4	0	746

Zaman bazlı analiz yapıldığında ise 15 saniyeden 120 saniyeye doğru modelin başarı oranın azaldığı tespit edilmiştir. Genel model başarısının aksine Browsing, Mail, VPN-Browsing ve VPN-Mail sınıflarının 15 saniyeden 120 saniyeye doğru başarı oranı arttığı görülmektedir. Chat ve Streaming sınıfları 15 saniyeden 120 saniyeye doğru en yüksek başarı kaybı yaşanan sınıflardır. Çizelge 5.6'da XGBoost algoritmasının zaman bazlı f1-ölçüt başarı metrikleri gösterilmiştir.

Çizelge 5.6. Senaryo b VPN zaman bazlı sınıfların f1-ölçüt değerleri

Sınıflar	15 Saniye	30 Saniye	60 Saniye	120 Saniye
Browsing	0,8642	0,8836	0,8612	0,8812
Chat	0,7934	0,77	0,7598	0,5942
FT	0,7546	0,6907	0,8296	0,703
Mail	0,7143	0,8788	0,8112	0,8125
P2P	0,9272	0,905	0,9211	0,9065
Streaming	0,7304	0,7562	0,5909	0,537
VoIP	0,988	0,9829	0,9467	0,9382
VPN-Browsing	0,8169	0,8521	0,8479	0,8387
VPN-Chat	0,7091	0,704	0,5874	0,5909

Çizelge 5.6. (devam) Senaryo b zaman bazlı sınıfların f1-ölçüt değerleri

VPN-FT	0,827	0,7865	0,6843	0,6805
VPN-Mail	0,75	0,8821	0,7387	0,8828
VPN-P2P	0,7265	0,7281	0,684	0,6931
VPN-Streaming	0,92	0,9026	0,8286	0,8037
VPN-VoIP	0,9874	0,9839	0,8952	0,9407

Genel olarak sonuçlar incelendiğinde VOIP trafiğin tüm senaryolar içerisinde en yüksek başarı oranı ile tespit edildiği görülmüştür. Fakat en düşük tespit oranlı sınıf her senaryoda farklıdır. Önerilen model en yüksek başarı oranlarına VPN olmadan yapılan şifreli trafik üzerinde elde ederken, VPN trafiğin olduğu durumlarda başarı oranı %94,53'den %85'lere düşmektedir. Zaman bazlı inceleme yapıldığında Senaryo A2 VPN trafik haricinde 15 saniye zamanlı yakalanan veri setleri en iyi sonuçları vermektedir. Senaryo A2 VPN veri seti içinse 30 saniye zamanlı yakalanan veri seti en iyi sonucu vermiştir. Sınıf bazlı analiz yapıldığında ise Browsing sınıfının 15 saniyeden 120 saniyeye doğru başarı oranı artmaktadır fakat modelin genel başarı oranı 15 saniyeden 120 saniyeye doğru azalmaktadır.

Çizelge 5.7. Tüm senaryolar için doğruluk ve f1-ölçüt değerleri

Algoritmalar	XGBoost (%)		Karar Ağacı (%)		Rassal Orman (%)	
Senayolar	Doğruluk	F1-Ölçütü	Doğruluk	F1-Ölçütü	Doğruluk	F1-Ölçütü
Senaryo A1 15s	93,02	93,01	89,29	89,28	91,79	91,78
Senaryo A1 30s	90,47	90,44	87,09	87,08	90,3	90,27
Senaryo A1 60s	89,59	89,54	85,97	85,99	89,32	89,34
Senaryo A1 120s	89,88	89,87	87,52	87,51	90,03	90,02
Senaryo A2 No-VPN 15s	94,53	94,48	91,08	91,03	93,65	93,57
Senaryo A2 No-VPN 30s	92,99	92,86	88,57	88,49	92,69	92,58
Senaryo A2 No-VPN 60s	92,99	92,86	88,7	88,65	92,38	92,27
Senaryo A2 No-VPN 120s	92,59	92,43	89,29	89,35	92,35	92,13

Çizelge 5.7. (devam) Tüm senaryolar için doğruluk ve f1-ölçüt değerleri

Senaryo A2 VPN 15s	89,35	89,14	85,4	85,37	88,3	88,14
Senaryo A2 VPN 30s	90,76	90,73	85,47	85,36	90,64	90,6
Senaryo A2 VPN 60s	86,98	86,55	83,4	83,43	87,2	86,87
Senaryo A2 VPN 120s	87,04	86,57	83,16	82,79	86,98	86,61
Senaryo B 15s	85,79	85,75	79,57	79,59	85,17	85,12
Senaryo B 30s	85,46	85,28	79,09	78,99	84,72	84,65
Senaryo B 60s	83,3	83,43	77,4	77,53	82,64	82,59
Senaryo B 120s	80,75	80,92	76,59	76,48	80,98	80,75

Tüm senaryolar için doğruluk oranları ve ağırlıklı f1-ölçüt değerleri Çizelge 5.7'de gösterilmiştir.

6. SONUÇ VE ÖNERİLER

Trafik sınıflandırması, trafik üzerindeki uygulamaların ve protokollerin tanımlandığı bir süreçtir. Güvenlik uygulamaları ve çözümleri ağ üzerinden geçen trafiğin sınıfına bakarak belirlenen politikalara göre izin verir, engeller veya alarm oluşturur. Belirlenen politikaların uygulanabilmesi için ilk adım olarak trafiğin tanımlanması ve sınıflandırılması gerekmektedir. Siber saldırıların büyün çoğunluğu ağ üzerinden geçmektedir ve trafiğin şifrelendiği durumlarda mevcut güvenlik çözümleri ağ trafiğini analiz etmekte yetersiz kalmaktadır. Ağ trafiği analiz edilemediği durumlarda mevcut güvenlik çözümleri siber saldırıları tespit edememekte ve engel olamamaktadır. Ayrıca siber saldırganlar trafiği şifreleyerek kendilerini ve ağ üzerindeki izlerini gizleyebilmektedir.

Bu çalışmada şifreli trafiğin sınıflandırılarak ağ uzmanları ve güvenlik analistlerinin ağ üzerindeki analiz yeteneklerinin artırılarak güvenlik sistemlerinin güçlendirilmesi, bilgi sistemlerinin ve kullanıcıların güvenliğinin artırılmasının sağlanması amaçlanmıştır. Şifreli trafiğin sınıflandırılması ile birlikte ağ uzmanları belirledikleri politikalara göre istenmeyen sınıflara ait trafikleri tespit ederek engelleyebilmektedir. Bu amaçla şifreli internet trafiğinin sınıflandırılması için XGBoost, Karar Ağacı ve Rassal Orman algoritmaları kullanılarak makine öğrenmesi tabanlı bir yaklaşım kullanılmıştır. Veri seti olarak açık kaynak olan ISCX VPN-NonVPN veri seti kullanılmıştır. Veri seti gerçek internet trafiği üzerinden elde edilmiş ve içerisinde günümüz ağ trafiğinde en çok kullanılan sınıflar bulunmaktadır. Bu durum yapılan çalışmanın mevcut güvenlik çözümleri ile entegrasyonunu artırmaktadır. Makine öğrenmesi kullanılmadan önce verilere önişlem uygulanmış ve özellik seçimi yapılmıştır. Makine öğrenme algoritmaları kullanılmadan önce ızgara metodu ve GridSearchCV kütüphanesi kullanılarak hiper parametre seçimi yapılmıştır.

XGBoost algoritması her senaryo için daha yüksek doğruluk oranına sahip olup bu problem için Karar Ağacı ve Rassal Orman algoritmalarından daha başarılı sonuçlar vermiştir. Senaryo A1'de en yüksek kesinlik oranı 15 saniye veri setinde elde edilmiş olup %93,04'dür. Bu sonuç, Draper-Gil [26] tarafından yapılan çalışma ile kıyaslandığında %3 daha başarılıdır. Senaryo A2 Non-VPN senaryosunda en iyi sınıflandırma 15 saniye veri setinde ve %94,53 doğruluk oranı ile elde edilmiştir. Senaryo A2 Non-VPN senaryoda elde ettiğimiz sonuçlar, Draper-Gil [26] tarafından yapılan çalışmadan daha yüksek bir başarı oranı sağlasa

da, Caicedo-Muñoz [28] tarafından yapılan çalışmadan nispeten biraz daha başarılıdır. Senaryo A2 VPN senaryosunda No-VPN senaryosuna göre modelin başarı oranı düşmüş ve 30 saniye veri setinde %90,76'lık doğruluk oranı elde edilmiştir. 14 sınıftan oluşan Senaryo B'de ise en yüksek başarı oranı %85,79'luk doğruluk oranı ile 15 saniye veri setinden elde edilmiştir. Önerilen model, Draper-Gil [26] tarafından yapılan çalışmadan daha yüksek bir başarı oranı elde etmesine rağmen, A2 ve B senaryosunda Caicedo-Muñoz [28] tarafından sunulan çalışma ile karşılaştırıldığında benzer sonuçlar elde edilmiştir. Önerilen model şifreli trafiğin sınıflandırılması işlemini her bir senaryo için yüksek başarı oranları ile gerçekleştirdiği görülmüştür.

Çoklu sınıflandırma işlemi yapılan Senaryo A2 ve Senaryo B içinde sınıf bazlı değerlendirme yapıldığında en yüksek başarı oranı ile tespit edilen sınıf VoIP sınıfıdır. VoIP şifreli trafikleri %100'e yakın başarı oranları ile tespit edilebilmektedir. Önerilen model, Saqib [15] tarafından yapılan çalışmaya kıyasla çok daha yüksek bir başarı oranıyla VoIP trafiğini tespit etmektedir. Senaryo A2 No-VPN veri setinde en düşük başarı oranına sahip sınıf Streaming sınıfı olup doğruluk oranı %86,03'dür. Senaryo A2 VPN veri setinde ise en düşük başarı oranına sahip sınıf Chat sınıfı olup Doğruluk oranı %84,64'dür. VPN-P2P sınıfı %65,88 doğruluk oranı ile Senaryo B içinde en düşük başarı oranına sahiptir ve Senaryo B için model performansını yüksek oranda düşürmektedir.

Derin paket analizi yapılamayan şifreli trafikler için yapay zekâ tabanlı sınıflandırma işlemleri 1990'lı yıllardan başlayan çalışmalar günümüzde de devam etmektedir. Derin paket analizi yapılamadığı için şifreli trafiklerin sınıflandırılma işlemi şifrelenmemiş trafiklerin sınıflandırılmasına göre çok daha zor olmaktadır. İnternetin kullanım oranının artması ile birlikte yeni uygulama ve yeni protokoller ortaya çıkmaktadır. Gizli ve güvenli iletişimin sağlanabilmesi için uygulamalar internet üzerinden giden trafiği şifreleme yöntem ve protokolleri ile şifrelemektedir. Gelecek çalışmalarda yeni çıkan uygulama ve protokollerin tespiti için yeni bir veri seti oluşturulması ve bu veri seti üzerinden sınıflandırma işlemi yapılabilir. Bu alanda çalışmalar devam ettikçe hem trafiği izleyen ağ uzmanlarının ve hem de siber güvenlik uzmanlarının şifreli trafik üzerinden analiz yapabilme yetenekleri artacaktır.

KAYNAKLAR

1. İnternet: Google Şeffaflık Raporu, URL: https://transparencyreport.google.com/https/overview Son Erişim Tarihi: 14 Kasım 2020.

2. Devi, R., T. (2013). *Importance of Cryptography in Network Security*. International Conference on Communication Systems and Network Technologies, Gwalior, 462-467.

3. Wood, D., Stoss, V., Chan-Lizardo, L., Papacostas, G., S., and Stinson, M., E. (1988). *Virtual private networks*. International on Private Switching Systems and Networks, London, 132-136.

4. Mao, H., Zhu, L., and Qin, H. (2012). *A Comparative Research on SSL VPN and IPSec VPN*. 8th International Conference on Wireless Communications, Networking and Mobile Computing, Shanghai, 1-4.

5. Piscitello, D., and Chapin, A., L. (1993). *Open Systems Networking: Tcp/Ip and Osi*. Massachusetts: Addison-Wesley, 33-43.

6. Stanton, R. (2005). Securing VPNs: comparing SSL and IPsec. *Computer Fraud & Security*, 17-19.

7. Nguyen, N. (2019). SSL/TLS Interception Challenge from the Shadow to the Light. *SANS Information Security Reading Room*, 1-6.

8. Radivilova, T., Kirichenko, L., Ageyev, D., Tawalbeh, M., and Bulakh, V. (2018, May). *Decrypting SSL/TLS Traffic for Hidden Threads Detection.* IEEE 9th International Conference on Dependable Systems, Services and Technologies, Kiev, 143-146.

9. El-Maghraby, R., T., Elazim, N., M., A., and Bahaa-Eldin, A., M. (2017). *A survey on deep packet inspection*. 12th International Conference on Computer Engineering and Systems, Cairo, 188-197.

10. İnternet: Cisco 2018 Annual Cyber Security Report, URL: https://www.cisco.com/c/dam/m/hu_hu/campaigns/security-hub/pdf/acr-2018.pdf Son Erişim Tarihi: 02 Ocak 2021.

11. Taylor, A. (2019). Decrypting SSL traffic: best practices for security, compliance and productivity. *Network Security*, 2019(8), 17-19.

12. Finsterbusch, M., Richter, C., Rocha, E., Muller, J., and Hanssgen, K. (2014). A Survey of Payload-Based Traffic Classification Approaches. *IEEE Communications Surveys & Tutorials,* 16(2), 1135-1156.

13. Wang, P., Chen, X., Ye, F., and Sun, Z. (2019). A Survey of Techniques for Mobile Service Encrypted Traffic Classification Using Deep Learning. *IEEE Access*, 2019 (7), 54024-54033.

14. İnternet: Service Name and Transport Protocol Port Number Registry, URL: https://www.iana.org/assignments/service-names-port-numbers/service-names-port-numbers.xhtml Son Erişim Tarihi: 16 Kasım 2020.

15. Plonka, D. (2000, December). FlowScan: *A Network Traffic FlowReporting and Visualization Tool.* Proceedings of the14th Systems Administration Conference, Louisiana, 305-317.

16. Moore, D., Keys, K., Koga, R., Lagache, E., and Claffy, K. (2001). *CoralReef software suite asa tool for system and network administrators.* Proceedings of the LISA 2001 15th Systems Administration Conference, Washington, 1-10.

17. Fraleigh, C., Moon, S., Lyles, B., Cotton, C., Khan, M., Moll, D., Rockell, R., Seely, T., and Diot, S., C. (2003). Packet-level traffic measurements from the Sprint IP backbone. *IEEE Network,* 17(6), 6-16.

18. Valenti, S., Rossi, D., Dainotti, A., Pescapè, A., Finamore, A., and Mellia, M. (2013). *Reviewing Traffic Classification.* Berlin: Springer, 123-147.

19. Sen, S., Spatscheck, O., and Wang, D. (2004). *Accurate, Scalable In-Network Identification of P2P Traffic using Application Signatures.* International World Wide Web Conference, New York, 512-521.

20. Moore., A., W., and Papagiannaki, K. (2005). *Lecture Notes in Computer Science.* Berlin: Springer, 41-54.

21. Finamore, A., Mellia, M., Meo, M., and Rossi, D. (2010). KISS: Stochastic Packet Inspection Classifier for UDP Traffic. *IEEE/ACM Transactions on Networking*, 18(5), 1505-1515.

22. Khater, N., A., and Overill, R., E. (2015). *Network Traffic Classification Techniques and Challenges.* Tenth International Conference on Digital Information Management, Jeju, 43-48.

23. Alshammari, R., and Zincir-Heywood, A., N. (2011). Can encrypted traffic be identified without port numbers, IP addresses and payload inspection?. *Computer Networks,* 55(6), 1326-1350.

24. Alshammari, R., and Zincir-Heywood, A., N. (2009). *Machine learning based encrypted traffic classification: Identifying SSH and Skype.* IEEE Symposium on Computational Intelligence for Security and Defense Applications, Ottowa, 1-8.

25. Di-Mauro, M., and Longo, M. (2015). *Revealing encrypted WebRTC traffic via machine learning tools.* 12th International Joint Conference on e-Business and Telecommunications, Colmar, 259-266.

26. Draper-Gil, G., Lashkari, A., H., Mamun M., S., I., and Ghorbani, A., A. (2016). *Characterization of Encrypted and VPN Traffic using Time-related Features.* Proceedings of the 2nd International Conference on Information Systems Security and Privacy, Rome, 407-414.

27. Seddigh, N., Nandy, B., Bennett, D., Ren, Y., and Dolgikh, S. (2019). *A Framework & System for Classification of Encrypted Network Traffic using Machine Learning.* 15th International Conference on Network and Service Management, Halifax, 1-5.

28. Caicedo-Muñoz, J., A., Espino, A., L., Corrales, J., C., and Rendón, A. (2018). QoS-Classifier for VPN and Non-VPN traffic based on time-related features. *Computer Networks,* 2018(144), 271-279.

29. Saqib, N., A., Shakeel, Y., Khan, M., A., Mahmood, H., and Zia, M. (2017). An effecitve empirical approach to VoIP traffic classification. *Turkish Journal of Electrical Engineering & Computer Sciences*, 25(2), 888-900.

30. Zhang, Y., Zhao, S., Zhang, J., Ma, X., Huang, F. (2019). *STNN: A Novel TLS/SSL Encrypted Traffic Classification System Based on Stereo Transform Neural Network.* 2019 IEEE 25th International Conference on Parallel and Distributed Systems, Tianjin, 907-910.

31. Chari, M., Srinidhi, H., and Somu, T., E. (2019). *Network Traffic Classification by Packet Length Signature Extraction.* IEEE International WIE Conference on Electrical and Computer Engineering, Bangalore, 1-4.

32. Pradhan, A., Behera, S., and Dash, R. (2018). *Hybrid RBFN Based Encrypted SSH Traffic Classification.* 5th International Conference on Signal Processing and Integrated Networks, Noida, 1-4.

33. Yang, Y., Kang, C., Gou, G., Li, Z., and Xiong, G. (2018). *TLS/SSL Encrypted Traffic Classification with Autoencoder and Convolutional Neural Network.* IEEE 20th International Conference on High Performance Computing and Communications, Exeter, 362-369.

34. İnternet: The top 500 sites on the web. Alexa, URL: https://www.alexa.com/topsites Son Erişim Tarihi: 14 Kasım 2020.

35. Al-Obaidy, F., Momtahen, S., Hossain, M., F., and Mohammadi, F. (2019). *Encrypted Traffic Classification Based ML for Identifying Different Social Media Applications.* IEEE Canadian Conference of Electrical and Computer Engineering, Edmonton, AB, Canada, 1-5.

36. Khatouni, A., S., and Zincir-Heywood N. (2019). *Integrating Machine Learning with Off-the-Shelf Traffic Flow Features for HTTP/HTTPS Traffic Classification.* IEEE Symposium on Computers and Communications, Barcelona, Spain, 1-7.

37. Nieles, M., Dempsey, K., and Pillitteri, V., Y. (2017). An Introduction to Information Security. *NIST Special Publication 800-12,* 1, 1-91.

38. Tokdemir, E., B. (2019). *An information security management system approach and technical security best practices for the enterprise companies*, Yüksek Lisans Tezi, Bahçeşehir Üniversitesi Fen Bilimleri Enstitüsü, İstanbul, 6-11.

39. Andress J. (2014). *The Basics of Information Security* (Second edition). United Kingdom: Syngress, 5-21.

40. Vigil, M., Buchmann, J., Cabarcas D., Weinert, C., and Wiesmaier, A. (2015). Integrity, Authenticity, Non-Repudiation, and Proof of Existence for Long-term Archiving: A survey. *Computers & Security*, 50, 6-32.

41. Menezes, A., Oorschot, P., V., and Vanstone, S. (1997). *Handbook of Applied Cryptography.* Florida: CRC Press, 2-5.

42. Aumasson, J., P. (2018). *Serious Cryptography Practical Introduction to Modern Encryption*. San Francisco: William Pollock, 1-12.

43. Ciğer, İ. (2012). *Data şifreleme algoritmaları ve performans analizi,* Yüksek Lisans Tezi, İstanbul Üniversitesi Fen Bilimleri Enstitüsü, İstanbul, 2-4.

44. Lozupone, V. (2018). Analyze encryption and public key infrastructure (PKI). *International Journal of Information Management*, 38(1), 42-44.

45. İnternet: SQL Server Best Practices: Using Asymmetric Keys to Implement Column Encryption, URL: https://alibaba-cloud.medium.com/sql-server-best-practices-using-asymmetric-keys-to-implement-column-encryption-bead4e15f548 Son Erişim Tarihi: 18 Kasım 2020.

46. Barker, E. (2016). Guideline for Using Cryptographic Standards in the Federal Government: Cryptographic Mechanisms. *NIST Special Publication 800-175B*, 22-28.

47. Katz, J., and Lindell, Y. *Introduction to Modern Cryptography* (Second edition). Boca Raton: CRC Press, 375-428.

48. Tzemos, I., Fournaris, A., P., and Sklavos, N. (2016). *Security and Efficiency Analysis of One Time Password Techniques.* Proceedings of the 20th Pan-Hellenic Conference on Informatics, Patras, Greece, 10-12.

49. York, D. (2010). *Seven Deadliest Unified Communications Attacks*, United States: Syngress, 41-69.

50. İnternet: Man-in-the-Middle Attack: Security and Privacy Concerns, URL: https://wikisites.cityu.edu.hk/sites/netcomp/articles/Pages/Man-in-the MiddleAttack.aspx Son Erişim Tarihi: 18 Kasım 2020.

51. Vega, J., Messier, M., and Chandra, P. (2002). *Network Security with OpenSSL*. Sebastopol: O'Reilly Media, 1-23.

52. Ristić, I. (2014). *Bulletproof SSL and TLS.* London: Feisty Duck Limited, 1-22.

53. Thomas, S., A. (2000). *SSL & TLS Essentials Securing the Web.* Canada: John Wiley & Sons, 37-66.

54. Barker, E., Dang, Q., Frankel, S., Scarfone, K., and Wouters, P. (2020). Guide to IPsec VPNs. *NIST Special Publication 800-77 Revision 1*, 3-8.

55. Malinowski, C., and Noble, R. (2007). Hashing and data integrity: Reliability of hashing and granularity size reduction. *Digital Investigation*, 4(2), 98-104.

56. Aydoğdu, N. (2014). *Sanal özel ağ (VPN) bağlantı mantığı (VPN teknolojisi) ve token güvenliğinin pin kodu ile arttırılması,* Yüksek Lisans Tezi*,* Beykent Üniversitesi Fen Bilimleri Enstitüsü, İstanbul, 4-10.

57. Pannu, A. (2015). Artificial Intelligence and its Application in Different Areas. *International Journal of Engineering and Innovative Technology*, 4(10), 79-84.

58. Mechelli, A., and Vieira, S. (2019). *Machine Learning Methods and Applications to Brain Disorders*. London: Academic Press, 1-43.

59. Turing, A., M. (1950). Computing Machinery and Intelligence. *Mind*, 59(236), 433-460.

60. Stamp, M. (2018). *Introduction to Machine Learning with Applications in Information Security.* Boca Raton: CRC Press, 1-9.

61. Shi, B., and Iyengar, S., S. (2020). *Mathematical Theories of Machine Learning*. Switzerland: Springer, 3-11.

62. Mohri, M., and Rostamizadeh, A., and Talwalkar, A. (2018). *Foundations of Machine Learning*, Cambridge: The MIT Press, 1-8, 224-230.

63. Müller, A., C., and Guido, S. (2016). *Introduction to Machine Learning with Python*. Sebastopol: O'Reilly Media, 17-23.

64. Kızılkaya, B. (2019). *Unsupervised anomaly detection algorithms,* Yüksek Lisans Tezi, Dokuz Eylül Üniversitesi Fen Bilimleri Enstitüsü, İzmir, 8-10.

65. Kakışım, A. (2019). *Ağ bağlantılı veriler için gözetimsiz ikili öznitelik oluşturma yöntemi*, Doktora Tezi, Gebze Teknik Üniversitesi Fen Bilimleri Enstitüsü, Kocaeli, 7-9.

66. Marsland, S. (2015). *Machine Learning An Algorithmic Perspective.* Boca Raton: CRC Press, 281-305.

67. Güçkıran, K. (2020). *Pekiştirmeli öğrenme problemlerinde keşif ve genelleştirme yöntemleri*, Yüksek Lisans Tezi, Yıldız Teknik Üniversitesi Fen Bilimleri Enstitüsü, İstanbul, 3-6.

68. Quinlan, J. (1986). Induction of Decision Trees. *Machine Learning*, 1, 81-106.

69. Breiman, L. (2001). Random Forest. *Machine Learning*, 45, 5-32.

70. Guo, H., and Goodchild M., F. (2020). *Manual of Digital Earth.* Singapore: Springer, 360-361.

71. Biau, G., and Erwan Scornet, E. (2016). A Random Forest Guided Tour. *TEST*, 25(2), 197-227.

72. Wittek, P. (2014). *Quantum Machine Learning.* Oxford: Academic Press, 89-95.

73. Bishop, C., M. (2006). *Pattern Recognition and Machine Learning.* New York: Springer, 657-663.

74. Callens, A., Morichon, D., Abadie, S., Delpey, M., and Liquet, B. (2020). Using Random Forest and Gradient Boosting Trees to Improve Wave Forecast at a Specific Location. *Applied Ocean Research*, 104, 1-9.

75. Friedman, J., H. (2001). Greedy Function Approximation: A Gradient Boosting Machine. *The Annuals of Statistics*, 29, 1-39.

76. Chen, T., and Guestrin, C. (2016). *XGBoost: A Scalable Tree Boosting System.* Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, 785-894.

77. Dev, V., A., and Eden, M., R. (2019). Gradient Boosted Decision Trees for Lithology Classification. *Computer Aided Chemical Engineering*, 74, 113-118.

78. Uddin, M. (2019). *Addressing Accuracy Paradox Using Enhanched Weighted Performance Metric in Machine Learning.* Sixth HCT Information Technology Trends, Ras Al Khaimah, 319-324.

79. Powers, D. (2007). Evaluation: From Precision, Recall and F-Factor to ROC Informedness, Markedness & Correlation. *Technical Report School of Informatics and Engineering*, Adelaide, 1-24.

80. Svec, P., Benko, L., Kadlecik, M., Kratochvil, J., and Munk, M. (2020). Web Usage Mining: Data Pre-processing Impact on Found Knowledge in Predictive Modelling. *Procedia Computer Science*, 171, 168-178.

81. Gudivada, V., Apon, A., and Ding, J. (2017). Data Quality Considerations for Big Data and Machine Learning: Going Beyond Data Cleaning and Transformations. *International Journal on Advances in Software*, 10, 1-20.

82. Singh, D., and Singh, B. (2019). Investigating the impact of data normalization on classification performance. *Applied Soft Computing*, 1-23.

83. Kuhn, M., and Johnson, K. (2013). *Applied Predictive Modelling.* New York: Springer, 488-502.

84. Zheng, A., and Casari, A. (2018). *Feature Engineering for Machine Learning.* Sebastopol: O'Reilly Media, 38-40.

85. Rustogi, R., and Prasad, A. (2019). *Swift Imbalance Data Classification using SMOTE and Extreme Learning Machine.* International Conference on Computational Intelligence in Data Science, Chennai, 1-6.

86. Chawla, N., V., Bowyer, K., W., Hall, L., O., and Kegelmeyer, W., P. (2002). SMOTE: Synthetic Minority Over-sampling Technique. *Journal of Artificial Intelligence Research*, 16, 321-357.

87. Yang, L., and Shami, A. (2020). On Hyperparameter Optimization of Machine Learning Algorithms: Theory and Practice. *Neurocomputing*, 415, 295-316.

88. Tran, N., Schneider, J., Weber, I., and Qin, A., K. (2020). Hyper-parameter Optimization in Classification: To-do or Not-to-do. *Pattern Recognition*, 103, 1-12.

89. Hutter, F., Kotthoff, L., and Vanschoren, J. (2019). *Automated Machine Learning - Methods, Systems, Challenges*. Switzerland: Springer, 3-5.

90. Tanyıldızı, E., and Demirtaş, F. (2019). *Hiper Parametre Optimizasyonu.* 1st International Informatics and Software Engineering Conference, Ankara, 1-5.

ÖZGEÇMİŞ

Kişisel Bilgiler

Soyadı, adı	: UĞURLU, Mesut
Uyruğu	: T.C.
Doğum tarihi ve yeri	: 01.12.1990, Şiran
Medeni hali	: Evli
Telefon	: 0 (542) 341 06 29
Faks	: 0 (262) 648 10 00
e-mail	: mesut.ugurlu@gazi.edu.tr

Eğitim

Derece	Eğitim Birimi	Mezuniyet Tarihi
Yüksek Lisans	Gazi Üniversitesi / Bilgi Güvenliği Mühendisliği	Devam ediyor
Lisans	Erciyes Üniversitesi / Elektrik Elektronik Mühendisliği	2013
Lise	Nermin Mehmet Çekiç Anadolu Lisesi	2008

İş Deneyimi

Yıl	Yer	Görev
2015-Halen	TÜBİTAK BİLGEM Kamu SM	Siber Güvenlik Uzmanı

Yabancı Dil

İngilizce

Yayınlar

Uğurlu, M., and Doğru, İ., A. (2019). *A Survey on Deep Learning Based Intrusion Detection System.* 4th International Conference on Computer Science and Engineering, Samsun, Turkey, 223-228.

Hobiler

Teknoloji, Bilim Kurgu, Doğa, Kamp

GAZİ GELECEKTİR...