T.C.
NECMETTİN ERBAKAN
ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

IoT AĞLARINA YÖNELİK SİBER SALDIRILARIN TABM MİMARİSİ İLE ÇOK SINIFLI TESPİTİ: CIC-IOT-2023 VERİ SETİ ÜZERİNE KAPSAMLI BİR ÇALIŞMA

Muhammed KAPLANGÖZ

YÜKSEK LİSANS TEZİ

Bilgisayar Mühendisliği Anabilim Dalı

Ocak-2026
KONYA

TEZ KABUL VE ONAYI

Muhammed KAPLANGÖZ tarafından hazırlanan "IoT AĞLARINA YÖNELİK SİBER SALDIRILARIN TABM MİMARİSİ İLE ÇOK SINIFLI TESPİTİ: CIC-IOT-2023 VERİ SETİ ÜZERİNE KAPSAMLI BİR ÇALIŞMA" adlı tez çalışması 22/01/2026 tarihinde aşağıdaki jüri tarafından oy birliği ile Necmettin Erbakan Üniversitesi Fen Bilimleri Enstitüsü Bilgisayar Mühendisliği Anabilim Dalı'nda YÜKSEK LİSANS TEZİ olarak kabul edilmiştir.

Jüri Üyeleri **İmza**

Başkan
Dr. Öğr. Üyesi Alper KILIÇ

Danışman
Prof. Dr. Mehmet HACIBEYOĞLU

Üye
Dr. Öğr. Üyesi Alperen EROĞLU

Fen Bilimleri Enstitüsü Yönetim Kurulu'nun …./…/20.. gün ve …….. sayılı kararıyla onaylanmıştır.

Prof. Dr. Havvanur UÇBEYİAY
FBE Müdürü

TEZ BİLDİRİMİ

Bu tezdeki bütün bilgilerin etik davranış ve akademik kurallar çerçevesinde elde edildiğini ve tez yazım kurallarına uygun olarak hazırlanan bu çalışmada bana ait olmayan her türlü ifade ve bilginin kaynağına eksiksiz atıf yapıldığını bildiririm.

DECLARATION PAGE

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Muhammed KAPLANGÖZ

Tarih: 06.01.2026

ÖZET

YÜKSEK LİSANS TEZİ

IoT AĞLARINA YÖNELİK SİBER SALDIRILARIN TABM MİMARİSİ İLE ÇOK SINIFLI TESPİTİ: CIC-IOT-2023 VERİ SETİ ÜZERİNE KAPSAMLI BİR ÇALIŞMA

Muhammed KAPLANGÖZ

NECMETTİN ERBAKAN ÜNİVERSİTESİ FEN BİLİMLERİ ENSTİTÜSÜ BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI

Danışman: Prof. Dr. Mehmet HACIBEYOĞLU

2026, 65 Sayfa

Jüri
Prof. Dr. Mehmet HACIBEYOĞLU
Dr. Öğr. Üyesi Alper KILIÇ
Dr. Öğr. Üyesi Alperen EROĞLU

Bu çalışmada, IoT ağlarına yönelik siber saldırıları tespit etmek amacıyla TabM (Tabular Model) derin öğrenme mimarisi kullanılarak 34 sınıflı bir ağ saldırı tespiti sistemi geliştirilmiştir. CIC-IoT-2023 veri seti üzerinde gerçekleştirilen çalışmada, TabM'in BatchEnsemble mekanizması ile 32 alt model aynı anda eğitilerek parametre verimliliği sağlanmıştır. Veri setindeki 399:1 oranındaki sınıf dengesizliğini ele almak için Quantile Transform ile özellik normalizasyonu ve SMOTE ile azınlık sınıflarının sentetik örneklerle dengelenmesi içeren hibrit bir strateji uygulanmıştır. Hiperparametre optimizasyonu Optuna framework'ü ile gerçekleştirilmiş ve model AdamW optimizer ile 50 epok boyunca eğitilmiştir. TabM modeli test setinde F1-Makro 0,8625 ve doğruluk %97,91 performansı elde etmiştir. Temel karşılaştırmada Rasgele Orman (F1-Makro: 0,882) ve XGBoost (F1-Makro: 0,828) ile rekabetçi performans sergilemiştir. 34 sınıfın 20'sinde F1 ≥ 0,95 başarısı elde edilmiş, özellikle DDoS türleri ve botnet saldırılarında başarılı bir performans göstermiştir. Sistematik ablasyon çalışması Quantile Transform'un en kritik bileşen olduğunu ortaya koymuştur. SHAP analizi ile IAT (Inter-Arrival Time) özelliğinin saldırı tespitinde en ayırt edici faktör olduğu tespit edilmiştir. Uygulama katmanı saldırılarında (SQL enjeksiyonu, XSS) düşük performans gözlemlenmiş olup bu durum veri setinin akış tabanlı yapısından kaynaklanmaktadır. Çalışma, TabM'in IoT ağ saldırı tespitinde bir değerlendirmesidir, çok sınıflı sınıflandırma ile literatürdeki ikili yaklaşımların ötesine geçmektedir ve hibrit sınıf dengesizliği stratejisinin sistematik analizini sağlamaktadır. Sonuçlar, TabM'in geleneksel yöntemlere rekabetçi bir alternatif olduğunu ve 1.52 MB model boyutu ile uç nokta dağıtımı için uygun olduğunu göstermektedir.

Anahtar Kelimeler: Ağ saldırı tespiti, CIC-IoT-2023, derin öğrenme, IoT güvenliği, SHAP, TabM

ABSTRACT

MS THESIS

MULTI-CLASSIFIED DETECTION OF CYBER ATTACKS ON IoT NETWORKS USING TABM ARCHITECTURE: A COMPREHENSIVE STUDY ON THE CIC-IOT-2023 DATASET

Muhammed KAPLANGÖZ

THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES OF NECMETTİN ERBAKAN UNIVERSITY THE DEGREE OF MASTER OF SCIENCE IN COMPUTER ENGINEERING

Advisor: Prof.Dr. Mehmet HACIBEYOĞLU

2026, 65 Pages

Jury
Advisor Prof. Dr. Mehmet HACIBEYOĞLU
Asst. Prof. Dr. Alper KILIÇ
Asst. Prof. Dr. Alperen Eroğlu

In this study, a 34-class network intrusion detection system was developed using the TabM (Tabular Model) deep learning architecture to detect cyber attacks targeting IoT networks. The study, conducted on the CIC-IoT-2023 dataset, simultaneously trained 32 sub-models using TabM's BatchEnsemble mechanism to achieve parameter efficiency. A hybrid strategy was implemented to address the 399:1 class imbalance in the dataset, involving feature normalization with Quantile Transform and balancing minority classes with synthetic samples using SMOTE. Hyperparameter optimization was performed using the Optuna framework, and the model was trained over 50 epochs with the AdamW optimizer. The TabM model achieved an F1-Macro of 0.8625 and an accuracy of 97.91% on the test set. In baseline comparisons, it exhibited competitive performance with Random Forest (F1-Macro: 0.882) and XGBoost (F1-Macro: 0.828). In 20 out of 34 classes, F1 $\geq$ 0.95 success was achieved, demonstrating particularly successful performance in DDoS attacks and botnet attacks. Systematic ablation analysis revealed that Quantile Transform is the most critical component. SHAP analysis identified IAT (Inter-Arrival Time) as the most distinctive factor in attack detection. Low performance was observed in application layer attacks (SQL injection, XSS), which is attributed to the stream-based nature of the dataset. This study is an evaluation of TabM in IoT network attack detection, going beyond binary approaches in literature with multi-class classification and providing a systematic analysis of the hybrid class imbalance strategy. The results show that TabM is a competitive alternative to traditional methods and is suitable for endpoint deployment with a model size of 1.52 MB.

Keywords: CIC-IoT-2023, deep learning, IoT security, network intrusion detection, SHAP, TabM

ÖNSÖZ

Bu yüksek lisans tezi, nesnelerin interneti (Intrnet of Things, IoT) teknolojisinin hızla yaygınlaşmasıyla birlikte ortaya çıkan siber güvenlik tehditlerini ele almak amacıyla hazırlanmıştır. IoT cihazlarının sayısının milyarlara ulaştığı günümüzde, bu cihazları hedef alan siber saldırıların tespiti ve önlenmesi kritik bir önem taşımaktadır. Bu çalışma, derin öğrenme teknolojilerinin ağ güvenliği alanındaki potansiyelini araştırmayı ve pratik uygulanabilirliğini göstermeyi amaçlamaktadır.

Tez çalışmam boyunca, TabM (Tabular Model) derin öğrenme mimarisini IoT ağ saldırı tespiti problemine uyarlamak için yoğun bir araştırma ve geliştirme süreci yürüttüm. CIC-IoT-2023 veri seti üzerinde 34 farklı saldırı türünü yüksek doğrulukla sınıflandırmayı hedefledim. Bu süreçte, 399:1 oranındaki sınıf dengesizliğini ele almak için SMOTE ve Quantile Transform içeren hibrit bir strateji geliştirdim. Modelin karar verme sürecini açıklanabilir kılmak amacıyla SHAP analizi uyguladım ve IAT (Inter-Arrival Time) özelliğinin saldırı tespitinde en kritik faktör olduğunu ortaya koydum. Ayrıca, TabM'in Rasgele Orman ve XGBoost gibi geleneksel yöntemlerle kapsamlı karşılaştırmasını gerçekleştirdim. Uygulama katmanı saldırılarının tespitindeki zorlukları analiz ederek, akış tabanlı veri setlerinin doğal sınırlamalarını detaylı olarak inceledim. Test sonuçlarında F1-Makro 0.8625 ve Doğruluk %97.91 başarı elde ettim ve modelin 1.52 MB boyutu ile uç cihazlarda kurulum için uygun olduğunu gösterdim.

Bu tezin tamamlanmasında değerli katkıları olan başta tez danışmanım Prof. Dr. Mehmet HACIBEYOĞLU olmak üzere tüm jüri üyelerine bilimsel rehberliği, değerli önerileri ve çalışmam boyunca gösterdiği sabır için en derin teşekkürlerimi sunarım. Ayrıca, teknik konularda yardımlarını esirgemeyen ve deneyimlerini benimle paylaşan Dr. Abdülkadir PEKTAŞ'a içten teşekkürlerimi sunarım.

Son olarak, bu araştırma sürecinde kullandığım açık kaynak kodlu araçların ve veri setlerinin geliştiricilerine, literatürdeki değerli çalışmalarıyla yolumu aydınlatan araştırmacılara ve bilimsel topluma teşekkür ederim. Bu çalışmanın, IoT güvenliği alanındaki gelecek araştırmalara katkı sağlamasını ve pratik uygulamalarda kullanılabilir çözümler sunmasını umuyorum.

Muhammed KAPLANGÖZ
KONYA-2026

İÇİNDEKİLER

ŞEKİLLER LİSTESİ

TABLOLAR LİSTESİ

SİMGELER VE KISALTMALAR

Bu çalışmada kullanılmış kısaltmalar, açıklamaları ile birlikte aşağıda sunulmuştur.

KISALTMALAR

ADASYN Adaptive Synthetic Sampling (Uyarlanabilir Sentetik Örnekleme)
AI Artificial Intelligence (Yapay Zeka)
ANN Artificial Neural Network (Yapay Sinir Ağı)
API Application Programming Interface (Uygulama Programlama Arayüzü)
AUC Area Under Curve (Eğri Altında Kalan Alan)
CIC Canadian Institute for Cybersecurity (Kanada Siber Güvenlik Enstitüsü)
CNN Convolutional Neural Network (Evrişimli Sinir Ağı)
CPU Central Processing Unit (Merkezi İşlem Birimi)
DdoS Distributed Denial of Service (Dağıtık Hizmet Reddi Saldırısı)
DL Deep Learning (Derin Öğrenme)
DoS Denial of Service (Hizmet Reddi Saldırısı)
DT Decision Tree (Karar Ağacı)
EDA Exploratory Data Analysis (Keşifsel Veri Analizi)
GBDT Gradient Boosted Decision Trees (Gradyan Artırımlı Karar Ağaçları)
GPU Graphics Processing Unit (Grafik İşlem Birimi)
GRU Gated Recurrent Unit (Kapılı Tekrarlayan Birim)
IAT Inter-Arrival Time (Paketler Arası Varış Süresi)
IDS Intrusion Detection System (Saldırı Tespit Sistemi)
IoT Internet of Things (Nesnelerin İnterneti)
KNN K-Nearest Neighbors (K-En Yakın Komşu)
LR Logistic Regression (Lojistik Regresyon)
LSTM Long Short-Term Memory (Uzun Kısa Süreli Bellek)
MitM Man-in-the-Middle (Ortadaki Adam Saldırısı)
ML Machine Learning (Makine Öğrenmesi)
MLP Multi-Layer Perceptron (Çok Katmanlı Algılayıcı)
NIDS Network Intrusion Detection System (Ağ Saldırı Tespit Sistemi)
RF Random Forest (Rastgele Orman)
RNN Recurrent Neural Network (Tekrarlayan Sinir Ağı)
ROC Receiver Operating Characteristic (Alıcı Çalışma Karakteristiği)
SHAP SHapley Additive exPlanations (Shapley Katkı Açıklamaları)
SVM Support Vector Machine (Destek Vektör Makinesi)
TabM Tabular Model (Tablo Modeli)
TCP Transmission Control Protocol (İletim Kontrol Protokolü)
TPE Tree-structured Parzen Estimator (Ağaç Yapılı Parzen Tahmin Edicisi)
UDP User Datagram Protocol (Kullanıcı Datagram Protokolü)
XAI Explainable AI (Açıklanabilir Yapay Zeka)
XGBoost Extreme Gradient Boosting (Aşırı Gradyan Artırma)
XSS Cross-Site Scripting (Siteler Arası Betik Çalıştırma)

1. GİRİŞ

Günümüzde dijitalleşmenin hız kazanmasıyla birlikte, siber güvenlik tehditleri hem bireysel hem de kurumsal düzeyde kritik bir öneme ulaşmıştır. Özellikle Nesnelerin İnterneti cihazlarının yaygınlaşması, ağ altyapılarını çeşitli saldırı türlerine karşı daha savunmasız hale getirmiştir. Dağıtık Hizmet Engelleme (Distributed Denial of Services-DDoS), zararlı yazılım, SQL enjeksiyonu ve kaba kuvvet gibi saldırılar, kurumların operasyonel sürekliliğini tehdit etmekte ve ciddi ekonomik kayıplara yol açmaktadır (Kala, 2023). Bu bağlamda, ağ trafiğini gerçek zamanlı olarak analiz edebilen ve anormal davranışları tespit edebilen Ağ Saldırı Tespit Sistemleri, siber güvenlik ekosisteminin vazgeçilmez bir bileşeni haline gelmiştir.

Geleneksel saldırı tespit yaklaşımları, önceden tanımlanmış imza veritabanlarına dayandığı için yalnızca bilinen saldırı türlerini tespit edebilmekte ve sıfır-gün (zero-day) saldırılarına karşı yetersiz kalmaktadır. Makine öğrenmesi tabanlı yaklaşımlar ise ağ trafiğindeki kalıpları öğrenerek hem bilinen hem de bilinmeyen saldırıları tespit etme potansiyeli sunmaktadır. Ancak bu alandaki en büyük zorluklardan biri, gerçek dünya veri setlerinde gözlemlenen aşırı sınıf dengesizliği problemidir. Normal trafik ve yaygın saldırı türleri veri setlerinde baskın iken, nadir görülen ancak kritik öneme sahip saldırı türleri çok az örnekle temsil edilmektedir.

Literatürde ağ saldırı tespiti için çeşitli makine öğrenmesi ve derin öğrenme yöntemleri önerilmiştir. Rasgele Orman, XGBoost ve LightGBM gibi ensemble yöntemler tablo verileri için güçlü performans göstermiştir (Nidhi, 2024). Derin öğrenme tarafında ise Evrişimsel Sinir Ağları (Convulutional Neural Networks-CNN), Uzun Kısa Süreli Bellek (Long-Short Term Memory-LSTM) ve son dönemde de transformer tabanlı mimariler çalışılmıştır. Ancak tabular veri için özel olarak tasarlanmış TabM mimarisinin ağ saldırı tespitinde kullanımı sınırlı kalmıştır.

Araştırmanın amacı

Bu araştırmanın temel amacı, nesnelerin interneti cihazlarına yönelik siber saldırıları yüksek doğrulukla tespit edebilen, sınıf dengesizliği problemine dayanıklı bir derin öğrenme modeli geliştirmektir. Bu kapsamda, tablo (tabular) verileri için özel olarak tasarlanmış TabM (Tabular Model) mimarisi, 34 farklı saldırı türünü içeren CIC-IoT-2023 veri seti üzerinde uygulanmış ve sınıf dengesizliği problemine yönelik hibrit bir çözüm stratejisi önerilmiştir.

Araştırmanın önemi

Bu tez çalışması, makine öğrenmesi tabanlı saldırı tespit sistemleri alanındaki teorik ve kavramsal çerçeveyi birkaç açıdan geliştirmektedir. İlk olarak, siber saldırıların sayısı ve karmaşıklığı her geçen yıl artmaktadır; küresel siber suçların yıllık ekonomik etkisinin 2023'te 8 trilyon ABD doları düzeyine ulaştığı, 2025'e kadar 10,5 trilyon ABD doları/yıl seviyesine çıkabileceği ve bunun yaklaşık %15 yıllık artış eğilimine karşılık geldiği raporlanmaktadır (Cobos ve Cakir, t.y.). Bu artış eğiliminin devam etmesi halinde 2026 için yaklaşık 12,1 trilyon ABD doları/yıl bandı makul bir projeksiyon olarak değerlendirilebilir. İkinci olarak, mevcut literatürdeki çalışmaların büyük çoğunluğu sınıf dengesizliği problemini yeterince ele almamakta veya sadece ikili sınıflandırma gerçekleştirmektedir. Bu çalışma, TabM mimarisinin CIC-IoT-2023 veri seti üzerindeki etkinliğini kapsamlı bir şekilde değerlendirmekte ve çok sınıflı saldırı tespiti için hibrit dengeleme stratejileri önermektedir.

Araştırmanın yöntemi

Yaklaşık 50 milyon örnekten oluşan orijinal CIC-IoT-2023 veri seti, sınıf dengesizliğini kontrol altına almak amacıyla stratejik örnekleme yöntemleriyle işlenmiştir. DDoS ve DoS gibi çoğunluk sınıfları 500.000 örneğe indirilirken (undersampling), azınlık sınıfları 10.000 örneğe çıkartılarak (oversampling) yaklaşık 9 milyon örneklik dengeli bir veri seti oluşturulmuştur. TabM modelinin hiperparametre optimizasyonu için Optuna kütüphanesi kullanılarak Bayesian optimizasyon yaklaşımı benimsenmiştir. Model performansı, sınıf dengesizliği nedeniyle doğruluk (accuracy) yerine F1-Makro skoru üzerinden değerlendirilmiştir. Çapraz doğrulama ile modelin genelleme yeteneği test edilmiş, ablasyon çalışması ile bileşenlerin katkıları incelenmiştir.

Araştırmanın Sınırlılıkları

Araştırmanın bazı sınırlılıkları göz önünde bulundurulmalıdır. Kullanılan CIC-IoT-2023 veri seti laboratuvar ortamında oluşturulmuş sentetik trafik içermektedir ve gerçek dünya ağ trafiğinden farklılıklar gösterebilir. Model eğitimi GPU gerektirmekte olup, kaynak kısıtlı ortamlarda uygulanabilirlik sınırlı kalabilmektedir. Derin öğrenme modelleri geleneksel yöntemlere göre daha az şeffaf olmakla birlikte, bu çalışmada SHAP (SHapley Additive exPlanations) analizi ile model yorumlanabilirliği artırılmıştır.

2. KAYNAK ARAŞTIRMASI

Nesnelerin interneti güvenliği alanında yapılan çalışmalar, özellikle saldırı tespit sistemlerinin geliştirilmesi ve veri setlerinin oluşturulması konularında hızla ilerlemektedir. 2023 yılında yayımlanan CICIoT2023 veri seti, 105 IoT cihazından toplanan ve 33 farklı saldırı türünü ve normal trafiği içeren kapsamlı bir veri kaynağı olarak literatüre önemli bir katkı sağlamıştır. Bu veri seti, önceki veri setlerine kıyasla daha geniş bir cihaz topolojisi ve saldırı çeşitliliği sunmaktadır (Neto vd., 2023a).

CICIoT2023 veri setini kullanan çalışmalar incelendiğinde, farklı yaklaşımların öne çıktığı görülmektedir. Jony ve Arnob (Jony ve Arnob, 2024a), LSTM tabanlı bir yaklaşım kullanarak %98,75 doğruluk oranına ulaşmıştır. Hizal ve arkadaşları, iki aşamalı bir derin öğrenme modeli geliştirerek DDoS saldırılarının tespitinde %94,96 doğruluk elde etmiştir (Hizal vd., 2024). Al-Halboosi ve arkadaşları, hibrit bir Transformer-CNN modeli önererek %99,47 doğruluk oranına ulaşmıştır (Al-Haboosi vd., 2024). Neto ve arkadaşları (Neto vd., 2023a), XGBoost tabanlı bir yaklaşımla federatif öğrenme kullanarak %99,46 doğruluk oranına ulaşmıştır.

Veri ön işleme ve boyut azaltma konusunda CICIoT 2023 veri seti üzerinde yapılan çalışmalarda farklı yaklaşımlar dikkat çekmektedir. Gheni ve Al-Yaseen, iki aşamalı veri kümeleme yaklaşımıyla veri boyutunu %62,45 oranında azaltırken yüksek doğruluk oranlarını korumayı başarmıştır (Gheni ve Al-Yaseen, 2024). Golestani ve Makaroff, cihaza özgü anomali tespit modelleri geliştirerek, farklı IoT cihazlarının özelliklerine göre optimize edilmiş çözümler sunmuştur (Golestani ve Makaroff, 2024a).

Makine öğrenmesi algoritmalarının CICIoT 2023 veri seti üzerinde karşılaştırmalı analizi konusunda Happy ve arkadaşları, Lojistik Regresyon, Rasgele Orman, Karar Ağacı, AdaBoost ve SVM gibi algoritmaları test etmiş, Rasgele Orman'ın %99,71 doğruluk oranıyla en iyi performansı gösterdiğini belirlemiştir (Happy vd., 2024a). Hinojosa ve Majd, edge computing tabanlı bir yaklaşımla CNN ve BiLSTM-CNN modellerini karşılaştırmış, CNN modelinin %93,8 F1 skoruyla daha başarılı olduğunu göstermiştir (Hinojosa ve Majd, 2024a). Brecca-Suarez ve arkadaşları, SMOTE tekniklerini kullanarak veri dengesizliği sorununu çözmüş ve iki sınıflı sınıflandırmada %99 F1-skoruna ulaşmıştır (Becerra-Suarez vd., 2024).

Yaras ve Dener, hibrit bir derin öğrenme algoritması geliştirerek büyük veri ortamında çalışacak bir IDS sistemi önermiştir. Apache Spark kullanılarak optimize edilen bu sistem, hem ikili sınıflandırmada (%99,99) hem de çok sınıflı sınıflandırmada

(%99,96) yüksek doğruluk oranları elde etmiştir (Yaras ve Dener, 2024). Benzer şekilde, Tseng ve arkadaşları, Transformer tabanlı bir modelle %99,40 doğruluk oranına ulaşmıştır (Tseng vd., 2024). Narayan ve arkadaşları, özellik seçimi ve dengeleme yöntemlerini kullanarak %76,3 F1-skoru elde etmiştir (Narayan vd., 2023). Jony ve arkadaşları, 34 sınıfı sınıflandırmada %98,59 F1-skoru elde etmiştir (Jony ve Arnob, 2024b).

Bu çalışmalar, CICIoT2023 veri setinin IoT güvenliği alanında kapsamlı ve gerçekçi bir test ortamı sunduğunu göstermektedir. Özellikle derin öğrenme tabanlı yaklaşımların yüksek başarı oranları elde ettiği, ancak işlem maliyeti ve gerçek zamanlı uygulanabilirlik konularında hala geliştirilmeye açık alanlar olduğu görülmektedir. Uç bilişim, veri kümeleme, sınıflandırma ve hibrit modeller gibi yaklaşımlar, bu zorlukları aşmak için umut verici çözümler sunmaktadır (Al-Garadi et al., 2020).

CICIoT2023 veri seti kullanılarak yapılan sınıflandırma çalışmalarında kullanılan yöntemler, yaklaşımlar ve elde ettikleri sonuçların karşılaştırmalı özeti Tablo 2.1'de sunulmaktadır.

Tablo 2.1. CICIoT2023 veri seti kullanılarak yapılan çalışmalar

Sınıflandırıcı Türleri	**Kullanılan Algoritmalar**	**En İyi Algoritmalar ve Performans (Doğruluk)**	**Öznitelik Çıkarımı / Ön İşleme Yöntemleri / DL Modelleri ve Hiperparametreler**	**Doğrulama Teknikleri**	**Referans**
Makine Öğrenmesi	Rasgele Orman (RF)	Çok Sınıflı: F1-Skoru: 76,03% (CFS+BRFC) USC'de F1 Kazancı: 7,9%	-Özellik Seçme: CFS, RF+MRMR -Sınıf Dengelemesi: ROS, BRFC -Normalizasyon: Standard Scaler	Eğitim / Test: (80/20) Kesinlik, Duyarlılık, F1, Doğruluk,	(Narayan vd., 2023)
Makine Öğrenmesi, Derin Öğrenme	XGBoost, Rasgele Orman, MLP, DNN	XGBoost: 99,46% RF: 99,40%	Özellik Çıkarımı: CICFlowMeter, DPKT Pencere tabanlı toplama (10 paket)	Eğitim / Test: (80/20)	(Neto vd., 2023b) 20.02.2026 15:44:00

Tablo 2.1.(devam) CICIoT2023 veri seti kullanılarak yapılan çalışmalar

Sınıflandırıcı Türleri	**Kullanılan Algoritmalar**	**En İyi Algoritmalar ve Performans (Doğruluk)**	**Öznitelik Çıkarımı / Ön İşleme Yöntemleri / DL Modelleri ve Hiperparametreler**	**Doğrulama Teknikleri**	**Ref**
Derin Öğrenme	DNN, CNN, LSTM	LSTM-tabanlı model: 94,96% (İkili Sınıflandırma) CNN-tabanlı model: 90,85% (Üç Sınıflı)	Özellik Seçimi, Rastgele Alt Küme Seçimi, Tekrar Kaldırma, Logaritmik Normalizasyon	Eğitim / Test / Doğrulama: (60/20/20), Yığın Boyutu=1000,	(Hizal vd., 2024)
Derin Öğrenme	CNN-Transformer, MLP, XGBoost	CNN-Transformer: 99,47% MLP: 99,39% XGBoost: 99,40%	- MinMax Normalizasyonu - Öznitelikler: 46 IoT özniteliği - CNN Bloğu: Yığın Normalizasyonu, Dropout (0,05), Yoğun Katmanlar	Eğitim / Test: (80/20) Yığın Boyutu: 1024 50 Epok	(Al-Haboosi vd., 2024)
Derin Öğrenme	LSTM	LSTM tabanlı model: 98,75% F1 Skoru: 98,59%	Öznitelik Seçimi, Standardizasyon (Ortalama=0, Std=1) Etiket Kodlama	Eğitim / Test: (70/30)	(Jony ve Arnob, 2024a)
Makine Öğrenmesi, Derin Öğrenme	MLP, AutoEncoder (AE), GSK kümeleme	İkili Sınıflandırma: MLP: 99,26% Çok Sınıflı: MLP: 97,46%	- Kümeleme için GSK optimizasyonu - Öznitelik Seçimi - Z-Skoru Normalizasyonu	Eğitim / Test: (80/20), 200 Epok	(Gheni ve Al-Yaseen, 2024)
Makine Öğrenmesi, OCC	SVM, DT, RF, DNN, iForest, OCSVM, LOF, DeepSVDD	Rasgele Orman (RF): 84,01% DNN: 81,88%	820GB PCAP dosyaları - Tcpdump/Scapy ile paket seviyesi öznitelik çıkarımı - Z-Score Normalizasyonu	Eğitim / Test: (70/30) Sonuçların ortalaması için 5 çalıştırma	(Golestani ve Makaroff, 2024b)
Makine Öğrenmesi	RF, DT, SVM, k-NN, Naive Bayes, Lojistik Regresyon	Rasgele Orman (RF): 99,71% Karar Ağacı (DT): 99,48% k-NN: 99,44%	- Öznitelik Ölçekleme - Normalizasyon - Sınıf dengesizliğini gidermek için veri setini dengeleme	Eğitim / Test: (70/30) Çalıştırma Süresi Analizi	(Happy vd., 2024b)
Derin Öğrenme	1D-CNN, BiLSTM-CNN	1D-CNN: 93,80% F1 BiLSTM-CNN: 93,82% F1 Skoru	- Dengeleme için Rastgele Alt Örnekleme - Quantile Transformer	Eğitim / Test: (70/30) Her Sınıf İçin	(Hinojosa ve Majd, 2024b)

Tablo 2.1.(devam) CICIoT2023 veri seti kullanılarak yapılan çalışmalar

Sınıflandırıcı Türleri	**Kullanılan Algoritmalar**	**En İyi Algoritmalar ve Performans (Doğruluk)**	**Öznitelik Çıkarımı / Ön İşleme Yöntemleri / DL Modelleri ve Hiperparametreler**	**Doğrulama Teknikleri**	**Ref**
Derin Öğrenme	Hibrit Derin Öğrenme (CNN + LSTM)	İkili: 99,995% Çok Sınıflı: 99,96%	Öznitelik Seçimi (Korelasyon Yöntemi), Normalizasyon	Eğitim / Test: (80/20)	(Yaras ve Dener, 2024)
Derin Öğrenme	Transformer	Çok Sınıflı: 99,40% İkili: 99,00%	- Öznitelik Normalizasyonu (Standart Ölçekleme)	Eğitim / Test: (80/20), Yığın Boyutu=1024, 10 Epok	(Tseng vd., 2024)
Derin Öğrenme	LSTM	98,75% F1 Skoru: 98,59%	- Yığın Boyutu: 1000 -Dönem: 50 - Aktivasyon Fonksiyonları: ReLU (Gizli Katman), Softmax (Çıkış Katmanı)	Eğitim / Test: (70/30)	(Jony ve Arnob, 2024c)

3. MATERYAL VE YÖNTEM

3.1. CICIoT2023 Veri Seti

CICIoT2023 veri seti, Canadian Institute for Cybersecurity (CIC) tarafından IoT güvenliği araştırmalarını desteklemek amacıyla, CIC IoT Laboratuvarı'nda kurulan gerçekçi bir akıllı ev/topoloji ortamında üretilmiştir. Bu kapsamda oluşturulan IoT topolojisi toplam 105 IoT cihazından oluşmakta; 67 cihaz saldırı deneylerinde doğrudan yer alırken, 38 Zigbee/Z-Wave cihaz ise ilgili network cihazları üzerinden ağa dahil edilmektedir (Neto vd., 2023b). Veri setinde 33 farklı saldırı yürütülmüş ve bu saldırılar yedi ana sınıfta ele alınmıştır: DDoS, DoS, Recon, Web-based, Brute force, Spoofing ve Mirai. Ayrıca saldırılar, klasik bilgisayar sistemleri yerine topolojiye bağlı kötü niyetli IoT cihazlar üzerinden yürütülmüş; deney düzeneğinde saldırı icrası için topolojiye bağlı Raspberry Pi cihazları kullanılmıştır. İlgili veri seti hazırlanırken kullanılan detaylı ürün bilgisi Tablo 3.1'de gösterime sunulmuştur.

Tablo 3.1. Veri seti ürün bilgisi

	Cihaz Adı	Kategori	MAC Adresi	Cihaz Adı	Kategori	MAC Adresi
Kurban (Victims)	Amazon Alexa Echo Dot 1	Audio	1C:FE:2B:98:16:DD	Lumiman bulb	Lighting	84:E3:42:42:ED:0B
	Amazon Alexa Echo Dot 2	Audio	A0:D0:DC:C4:08:FF	Philips Hue Bridge	Hub	00:17:88:60:D6:4F
	Amazon Alexa Echo Spot	Audio	1C:12:B0:9B:0C:EC	Smart Board	Home Automation	00:02:75:F6:E3:CB
	Amazon Alexa Echo Studio	Audio	08:7C:39:CE:6E:2A	Teckin Light Strip	Lighting	18:69:D8:EB:D4:3E
	Amazon Echo Show	Audio	2C:71:FF:05:F1:15	Teckin Plug 1	Power Outlet	D4:A6:51:76:06:64
	Google Nest Mini Speaker	Audio	CC:F4:11:9C:D0:00	Teckin Plug 2	Power Outlet	D4:A6:51:78:97:4E
	harman kardon (Ampak Technology)	Audio	B0:F1:EC:D3:E7:98	Wemo smart plug 1 (Wemo id: Wemo.Mini.AD3)	Power Outlet	30:23:03:F3:84:2B
	Sonos One Speaker	Audio	48:A6:B8:F9:1B:88	Wemo smart plug 2 (Wemo id: Wemo.Mini.4A3)	Power Outlet	30:23:03:F3:57:CB
	AMCREST WiFi Camera	Camera	9C:8E:CD:1D:AB:9F	Yutron Plug 1	Power Outlet	D4:A6:51:20:91:D1
	Arlo Base Station	Camera	3C:37:86:6F:B9:51	Yutron Plug 2	Power Outlet	D4:A6:51:21:6C:29
	Arlo Q Indoor Camera	Camera	40:5D:82:35:14:C8	LG Smart TV	Home Automation	AC:F1:08:4E:00:82
	Borun/Sichuan-AI Camera	Camera	C0:E7:BF:0A:79:D1	Netatmo Weather Station	Home Automation	70:EE:50:6B:A8:1A
	DCS8000LHA1 D-Link Mini Camera	Camera	B0:C5:54:59:2E:99	Raspberry Pi 4—2 GB	NextGen	DC:A6:32:C9:E6:F4
	HeimVision Smart WiFi Camera	Camera	44:01:BB:EC:10:4A	Raspberry Pi 4—2 GB	NextGen	DC:A6:32:C9: E4:C6
	Home Eye Camera	Camera	34:75:63:73:F3:36	Raspberry Pi 4—2 GB	NextGen	DC:A6:32:C9:E5:02
	Luohe Cam Dog	Camera	7C:A7:B0:CD:18:32	Fibaro Door/Window Sensor 1	Sensor	N/A
	Nest Indoor Camera	Camera	44:BB:3B:00:39:07	Fibaro Door/Window Sensor 2	Sensor	N/A
	Netatmo Camera	Camera	70:EE:50:68:0E:32	Fibaro Door/Window Sensor 3	Sensor	N/A
	Rbcior Camera	Camera	10:5A:17:97:A5:C6	Fibaro Flood Sensor 1	Sensor	N/A
	SIMCAM 1S (AMPAKTec)	Camera	10:2C:6B:1B:43:BE	Fibaro Flood Sensor 2	Sensor	N/A
	TP-Link Tapo Camera	Camera	6C:5A:B0:44:1D:90	Fibaro Motion Sensor 1	Sensor	N/A
	Wyze Camera	Camera	7C:78:B2:86:0D:81	Fibaro Motion Sensor 2	Sensor	N/A
	Yi Indoor Camera	Camera	84:7A:B6:64:62:58	Fibaro Motion Sensor 3	Sensor	N/A
	Yi Indoor 2 Camera	Camera	84:7A:B6:62:3A:6C	Fibaro Motion Sensor 4	Sensor	N/A
	Yi Outdoor Camera	Camera	2C:D2:6B:66:D2:87	Fibaro Motion Sensor 5	Sensor	N/A
	Eufy HomeBase 2	Hub	8C:85:80:6C:B6:47	Fibaro Wall Plug 1	Power Outlet	N/A
	Amazon Plug	Power Outlet	B8:5F:98:D0:76:E6	Fibaro Wall Plug 2	Power Outlet	N/A
	Atomi Coffee Maker	Home Automation	68:57:2D:56:AC:47	Ring Alarm Keypad	Home Automation	N/A
	Cocoon Smart HVAC Fan	Home Automation	08:3A:F2:1F:BC:68	Ring Range Extender	Home Automation	N/A
	Globe Lamp ESP_B1680C	Lighting	50:02:91:B1:68:0C	Ring Contact Sensor (1)	Sensor	N/A
	GoSund Bulb	Lighting	C4:DD:57:13:07:C6	Ring Contact Sensor (2)	Sensor	N/A

Tablo 3.1.(devam) Veri seti ürün bilgisi

	Cihaz Adı	Kategori	MAC Adresi	Cihaz Adı	Kategori	MAC Adresi
	Gosund Power strip (1)	Power Outlet	50:02:91:1A:CE:E1	AeoTec TriSensor	Sensor	N/A
	GoSund Power strip (2)	Power Outlet	B8:F0:09:03:9A:AF	AeoTec Doorbell 6	Home Automation	N/A
	GoSund Smart plug WP2 (1)	Power Outlet	B8:F0:09:03:29:79	AeoTec Indoor Siren	Home Automation	N/A
	GoSund Smart Plug WP2 (2)	Power Outlet	50:02:91:10:AC:D8	AeoTec Smart Switch 7	Home Automation	N/A
	GoSund Smart plug WP2 (3)	Power Outlet	50:02:91:10:09:8F	AeoTec Water Sensor 6	Sensor	N/A
	GoSund Smart Plug WP3 (1)	Power Outlet	C4:DD:57:0C:39:94	AeoTec NanoMote Quad	Home Automation	N/A
	Gosund Smart Plug WP3 (2)	Power Outlet	24:A1:60:14:7F:F9	AeoTec Door/Window Sensor 7 Pro	Sensor	N/A
	Govee Smart Humidifer	Home Automation	D4:AD:FC:29:C8:A2	AeoTec Temperature and Humidity Sensor	Sensor	N/A
	HeimVision SmartLife Radio/Lamp	Lighting	D4:A6:51:30:64:B7	Philips Hue White 1	Lighting	N/A
	iRobot Roomba	Home Automation	50:14:79:37:80:18	Philips Hue White 2	Lighting	N/A
	LampUX RGB	Lighting	F4:CF:A2:34:48:6B	SmartThings Smart Bulb 1	Lighting	N/A
	Levoit Air Purifier	Home Automation	1C:9D:C2:8C:9A:94	SmartThings Smart Bulb 2	Lighting	N/A
	LIFX Lightbulb	Lighting	D0:73:D5:35:FB:C8	Aeotec Button	Home Automation	N/A
	SmartThings Hub	Hub	28:6D:97:7A:2B:2D	AeoTec Motion Sensor	Sensor	N/A
	AeoTec Smart Home Hub	Hub	28:6D:97:9E:F4:D5	AeoTec Multipurpose Sensor	Sensor	N/A
	Sengled Smart Plug 2	Power Outlet	N/A	AeoTec Water Leak Sensor	Sensor	N/A
	SmartThings Button	Home Automation	N/A	Sengled Smart Plug 1	Power Outlet	N/A
	SmartThings Smart Bulb 3	Lighting	N/A	Sonoff Smart Plug 2	Power Outlet	N/A
	Sonoff Smart Plug 1	Power Outlet	N/A	Arlo Ultra 2 Outdoor Camera	Camera	N/A
Saldıran	Raspberry Pi 4—4 GB	NextGen	E4:5F:01:55:90:C4	Raspberry Pi 4—2 GB	NextGen	DC:A6:32:C9:E4:D5
	Raspberry Pi 4—8 GB	NextGen	DC:A6:32:DC:27:D5	Raspberry Pi 4—2 GB	NextGen	DC:A6:32:C9:E5:EF
	Raspberry Pi 4—2 GB	NextGen	DC:A6:32:C9:E4:AB	Raspberry Pi 4—2 GB	NextGen	DC:A6:32:C9:E4:90
	Raspberry Pi 4—2 GB	NextGen	DC:A6:32:C9:E5:A4	Ring Base Station	Hub	B0:09:DA:3E:82:6C
	Fibaro Home Center Lite	Hub	AC:17:02:05:34:27	Eufy Doorbell Camera	Camera	N/A

Saldırı türlerinin tam listesi her bir saldırının yürütülmesinde kullanılan araç/çerçeveler çalışma kapsamında tablolaştırılmıştır (Tablo 3.2). Bu saldırılar arasında DDoS/DoS kapsamında farklı flood ve parçalama senaryoları; Recon kapsamında port/OS/zafiyet taramaları ve host keşfi; Web-based kapsamında SQLi, XSS, command injection, upload, backdoor ve browser hijacking; Brute force kapsamında sözlük saldırıları; Spoofing kapsamında ARP/DNS spoofing; Mirai kapsamında GREIP/GREETH/UDPPlain türleri yer almaktadır.

DDoS/DoS saldırıları, hedefi yüksek hacimli trafikle hizmet veremez hâle getirmeyi amaçlayan flood ve benzeri yöntemleri; parçalama temelli senaryolar ise IP parçalarının yeniden birleştirilmesini istismar eden DoS yaklaşımlarını kapsamaktadır (Associate Professor, Department of Computer Science Government Arts College, Thuvakudimalai, Tiruchirappalli, India ve Kumar, 2017). Recon grubu, port tarama ve uzaktan işletim sistemi tespiti (TCP/IP yığın parmak izi) gibi keşif faaliyetlerini içerir (Scarfone vd., 2008). Web-based saldırılar; SQL enjeksiyonu, XSS ve komut enjeksiyonu gibi enjeksiyon temelli zafiyetler ile güvensiz dosya yükleme, kalıcı erişim amacıyla web kabuğu/backdoor yerleştirme ve tarayıcı ayarlarını izinsiz değiştirmeye dayalı hijacking senaryolarını kapsamaktadır (Odiaga Gloria Awuor, 2023). Brute force grubu, hesap/parola tahminine dayalı (örn. sözlük tabanlı) denemeleri içerir (Associate Professor, Department of Computer Science Government Arts College, Thuvakudimalai, Tiruchirappalli, India ve Kumar, 2017). Spoofing grubu; ARP önbellek zehirleme gibi yerel ağda kimlik/rotalama manipülasyonlarını ve DNS önbellek zehirleme (DNS spoofing) ile yanlış çözümleme üzerinden yönlendirmeyi kapsar (Mallik vd., 2019). Mirai grubu ise botnet tabanlı saldırı varyantlarını (örn. GREIP/GREETH/UDPPlain) içermektedir (McDermott vd., 2018).

CICIoT2023 ağ trafiği, saldırgan ve kurban ağlar arasına konumlandırılan Gigamon Network Tap üzerinden pasif şekilde kopyalanarak toplanmıştır. Tap cihazı, IoT trafiğini iki farklı ağ monitörüne yönlendirmekte ve trafik bu monitörler üzerinde Wireshark kullanılarak pcap formatında kaydedilmektedir; bu yapı ağ operasyonlarını etkilemeden çift yönlü, müdahalesiz ve gecikme eklemeyen bir izleme olanağı sağlamaktadır.

Veri seti iki formatta sunulmaktadır: pcap ve csv. Pcap dosyaları ham paket trafiğini içerirken, csv dosyaları pcapten türetilen ve makine öğrenmesi çalışmalarında doğrudan kullanılabilen öznitelikleri barındırır. Csv öznitelikleri, iki host arasındaki paket dizilerinden türetilen ve sabit boyutlu paket penceresi yaklaşımıyla özetlenen

özelliklerden oluşur. Özellik çıkarımı, pcap verisi üzerinde DPKT python kütüphanesi kullanılarak gerçekleştirilmiş; çıkarılan özellikler (Tablo 3.2) csv dosyalarında saklanmıştır. Ayrıca saldırı sınıfları ana başlıkları ile beraber Tablo 3.3'de gösterime sunulmuştur (Neto vd., 2023b).

Son olarak, her bir saldırı senaryosu için yakalanan tüm trafik, ilgili saldırı sınıfına ait olacak şekilde senaryo bazında etiketlenerek Tablo 3.4'te gösterilmiştir.

Tablo 3.2. Saldırı listesi ve araçlar

Sınıf	Atak	Adet	Araç
DDoS	ACK Fragmentation	285.104	hping3
	UDP Flood	5.412.287	udp-flood
	SlowLoris	23.426	slowloris
	ICMP Flood	7.200.504	hping3
	RSTFIN Flood	4.045.285	hping3
	PSHACK Flood	4.094.755	hping3
	HTTP Flood	28.790	golang-httpflood
	UDP Fragmentation	286.925	udp-flood
	ICMP Fragmentation	452.489	hping3
DoS	TCP Flood	4.497.667	hping3
	SYN Flood	4.059.190	hping3
	SynonymousIP Flood	3.598.138	hping3
	TCP Flood	2.671.445	hping3
	HTTP Flood	71.864	golang-httpflood
	SYN Flood	2.028.834	hping3
	UDP Flood	3.318.595	hping3 and udp-flood
Recon	Ping Sweep	2262	nmap and fping
	OS Scan	98.259	nmap
	Vulnerability Scan	37.382	nmap and vulscan
	Port Scan	82.284	nmap
	Host Discovery	134.378	nmap
Web-Based	Sql Injection	5245	DVWA
	Command Injection	5409	DVWA
	Backdoor Malware	3218	DVWA and Remot3d
	Uploading Attack	1252	DVWA
	XSS	3846	DVWA
Brute Force	Browser Hijacking	5859	Beef
	Dictionary Brute Force	13.064	nmap and hydra
Spoofing	Arp Spoofing	307.593	ettercap
	DNS Spoofing	178.911	ettercap
Mirai	GREIP Flood	751.682	Adapted Mirai Source Code
	Greeth Flood	991.866	Adapted Mirai Source Code
	UDPPlain	890.576	Adapted Mirai Source Code

Tablo 3.3. DPKT kullanılarak çıkartılmış özellikler

#	**Özellik**	**Açıklama**
1	flow duration	Paketin akış süresi
2	Header Length	Başlık uzunluğu
3	Protocol Type	IP, UDP, TCP, ICMP, ICMP
4	Duration	Yaşam süresi (ttl)
5	Rate	Bir akıştaki paket iletim hızı
6	Srate	Bir akıştaki giden paket iletim hızı
7	Drate	Bir akıştaki gelen paket iletim hızı
8	fin flag number	Fin bayrağı değeri
9	syn flag number	Syn bayrağı değeri
10	rst flag number	Rst bayrağı değeri
11	psh flag number	Psh bayrağı değeri
12	ack flag number	Ack bayrağı değeri
13	ece flag number	Ece bayrağı değeri
14	cwr flag number	Cwr bayrağı değeri
15	ack count	Aynı akışta ack bayrağı ayarlanmış paket sayısı
16	syn count	Aynı akışta syn bayrağı ayarlanmış paket sayısı
17	fin count	Aynı akışta fin bayrağı ayarlanmış paket sayısı
18	urg count	Aynı akışta urg bayrağı ayarlanmış paket sayısı
19	rst count	Aynı akışta rst bayrağı ayarlanmış paket sayısı
20	HTTP	Protokolün HTTP olup olmadığını gösterir
21	HTTPS	Protokolün HTTPS olup olmadığını gösterir
22	DNS	Uygulama katmanı protokolünün DNS olup olmadığını gösterir
23	Telnet	Uygulama katmanı protokolünün Telnet olup olmadığını gösterir
24	SMTP	Uygulama katmanı protokolünün SMTP olup olmadığını gösterir
25	SSH	Uygulama katmanı protokolünün SSH olup olmadığını gösterir
26	IRC	Uygulama katmanı protokolünün IRC olup olmadığını gösterir
27	TCP	Taşıma katmanı protokolünün TCP olup olmadığını gösterir
28	UDP	Taşıma katmanı protokolünün UDP olup olmadığını gösterir
29	DHCP	Uygulama katmanı protokolünün DHCP olup olmadığını gösterir
30	ARP	Bağlantı katmanı protokolünün ARP olup olmadığını gösterir
31	ICMP	Ağ katmanı protokolünün ICMP olup olmadığını gösterir
32	IPv	Bağlantı katmanı protokolünün IPv olup olmadığını gösterir
33	LLC	Bağlantı katmanı protokolünün LLC olup olmadığını gösterir
34	Tot sum	Akıştaki paket uzunluklarının toplamı
35	Min	Akıştaki minimum paket uzunluğu
36	Max	Akıştaki maksimum paket uzunluğu
37	AVG	Akıştaki ortalama paket uzunluğu
38	Std	Akıştaki paket uzunluğunun standart sapması

Tablo 3.3.(devam) DPKT kullanılarak çıkartılmış özellikler

#	Özellik	Açıklama
39	Tot size	Paketin uzunluğu
40	IAT	Önceki paketle zaman farkı
41	Number	Akıştaki paket sayısı
42	Magnitude	(Akıştaki gelen paketlerin uzunluklarının ortalaması + Akıştaki giden paketlerin uzunluklarının ortalaması) ^{0.5}
43	Radius	Akıştaki giden paketlerin uzunluklarının varyansı^{0.5}
44	Covariance	Gelen ve giden paketlerin uzunluklarının kovaryansı
45	Variance	Akıştaki giden paketlerin uzunluklarının varyansı
46	Weight	Akıştaki gelen paket sayısı × Akıştaki giden paket sayısı

Tablo 3.4. Saldırı vektörleri ve alt sınıflar

DDOS	DOS	RECONNAISSANCE	WEB-BASED ATTACKS	BRUTE FORCE	SPOOFING	MIRAI BOTNET
ACK Fragmentation	TCP Flood	Ping Sweep	SQL Injection	Dictionary Brute Force	ARP Spoofing	GREIP Flood
UDP Flood	HTTP Flood	OS Scan	Command Injection		DNS Spoofing	Greeth Flood
SlowLoris	SYN Flood	Vulnerability Scan	Backdoor Malware			UDP Plain
ICMP Flood	UDP Flood	Port Scan	Uploading Attack			
RSTFIN Flood		Host Discovery	XSS (Cross-Site Scripting)			
PSHACK			Browser Hijacking			
HTTP Flood						
UDP Fragmentation						
ICMP Fragmentation						
TCP Flood						
SYN Flood						
Synonymous IP Flood						

3.2. IoT Laboratuvar Ortamı

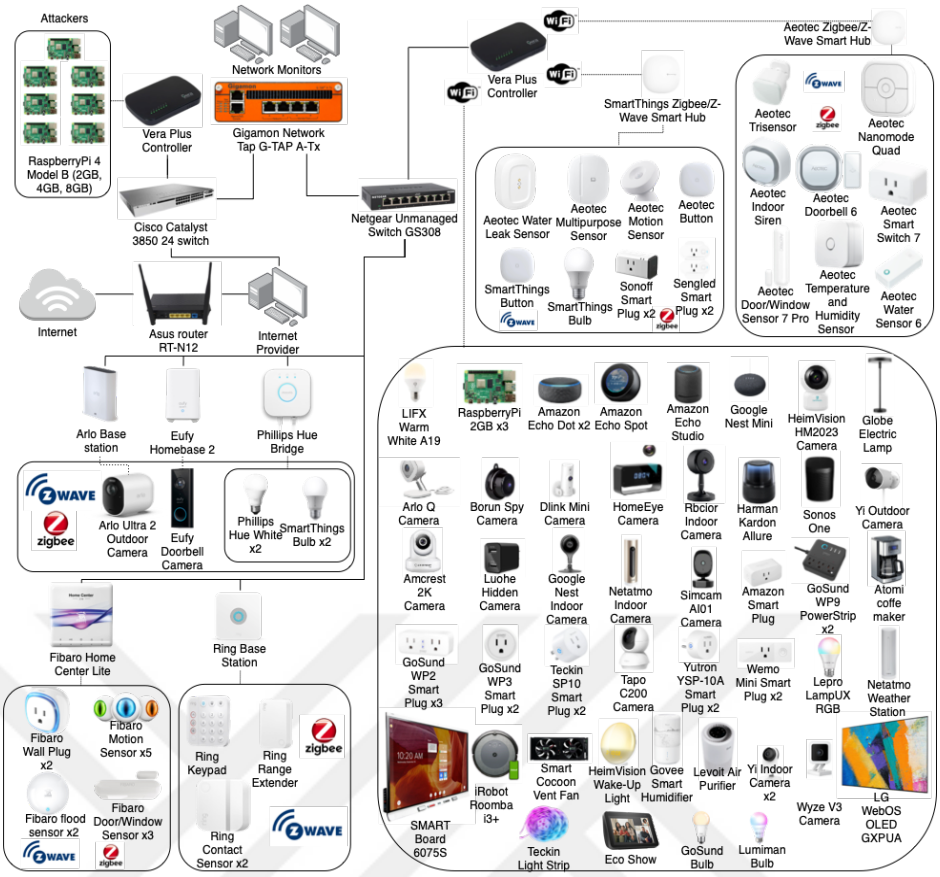
CIC IoT Laboratuvarı, akıllı ev senaryosunu temsil edecek şekilde kurgulanmış ve toplam 105 IoT cihazından oluşan kapsamlı bir ağ topolojisi sunmaktadır. Cihazlar Şekil 3.1'de gösterildiği üzere laboratuvar alanına fiziksel olarak dağıtılmış (masa, zemin ve duvar yerleşimleri dâhil) ve sürekli çalışmayı destekleyecek biçimde gerekli güç ve ağ altyapısı sağlanmıştır (Neto vd., 2023b). Şekil 3.2'de gösterildiği üzere laboratuvar altyapısı, saldırgan ağı ve kurban/IoT ağı olmak üzere iki ana ağ segmenti şeklinde ele alınmaktadır (Neto vd., 2023a).

Şekil 3.1. IoT cihazlarının fiziksel gösterimi(Neto vd., 2023b)

Birinci bölüm (saldırgan segmenti), internet çıkışı sağlayan bir yönlendiriciye bağlı yönetim bileşenleri ve saldırı düğümlerinden oluşmaktadır. Bu segmentte bir masaüstü bilgisayar yönetim/koordinasyon amaçlı kullanılmış; bir anahtar üzerinden aynı segmente bağlı 7 adet Raspberry Pi konumlandırılmıştır. Raspberry Pi cihazları, saldırı senaryolarının yürütüldüğü saldırgan düğümler olarak görev yapmaktadır. Bu segmentte ayrıca akıllı ev kontrol altyapısında kullanılan denetleyici/hub bileşenleri (ör. VeraPlus) yer almakta ve ilgili IoT kontrol trafiği bu ağ üzerinden taşınmaktadır.

İkinci bölüm (kurban/IoT segmenti), IoT uç cihazlarının bulunduğu ağdır. Bu segmentte yönetilemez bir anahtar üzerinden IoT ağ geçidi/bridge bileşenleri ile Zigbee ve Z-Wave gibi protokollerle haberleşen hub/denetleyiciler konumlandırılmıştır. Kurban segmentinde saldırılarda hedef olarak kullanılan çeşitli IoT cihazlar (kamera, sensör, akıllı priz vb.) bu kontrol altyapısı üzerinden ağa erişmektedir.

İki segment arasına yerleştirilen Network TAP cihazı, gerçek trafik akışını etkilemeden trafiğin pasif biçimde kopyalanmasını sağlamaktadır. TAP üzerinde bulunan izleme portları aracılığıyla trafik iki ayrı monitör arayüzüne yönlendirilmiş; bu monitörler üzerinde Wireshark kullanılarak trafik pcap formatında kaydedilmiştir. Bu yaklaşım, çift yönlü trafiğin müdahalesiz izlenmesine imkân vererek veri toplama sürecinin ağ operasyonlarına etkisini en aza indirir.



Şekil 3.2. Laboratuvar altyapı mantıksal topoloji(Neto vd., 2023b)

3.3. Veri Toplama Metodolojisi

Veri toplama süreci, hem normal hem de zararlı ağ trafiğini sistematik biçimde kapsayacak şekilde planlanmıştır. Normal trafik, IoT cihazlarının günlük kullanımını yansıtacak biçimde 16 saat boyunca toplanmıştır. Bu süreçte cihazlar varsayılan yapılandırmalarıyla çalıştırılmış ve herhangi bir saldırı senaryosu uygulanmamıştır. Normal trafik; sensör verilerinin toplanması, akıllı hoparlör komutları, kamera görüntülerine erişim ve benzeri tipik IoT etkileşimlerini içermektedir.

Kötü niyetli trafik tarafında, her saldırı türü ayrı bir deney olarak ele alınmış ve her deney sırasında ilgili saldırı senaryosu kontrollü biçimde yürütülmüştür. Saldırılar, saldırgan düğüm olarak konumlandırılan Raspberry Pi cihazları üzerinden başlatılarak hedef IoT cihazlarına yönlendirilmiştir. Deney tasarımında hedefleme mantığı saldırı türüne göre değişmektedir: Örneğin DDoS/DoS saldırıları ağ segmentindeki geniş hedef kümesine uygulanırken, web tabanlı saldırılar yalnızca web uygulaması/servisi sunan hedef cihazlar üzerinde yürütülmüştür.

Saldırıların gerçekleştirilmesinde kullanılan araçlar saldırı kategorilerine göre özetle aşağıdaki şekilde gruplandırılabilir:

- **DDoS/DoS:** Paket tabanlı ve uygulama tabanlı yoğun trafik üretimi (ör. hping3, UDP flood araçları, slowloris, HTTP flood benzeri araçlar)
- **Recon (Keşif):** Host keşfi ve tarama faaliyetleri (ör. fping, nmap, zafiyet tarama eklentileri/araçları)
- **Web-based:** Web uygulaması zafiyet senaryoları ve istemci tarafı etkileşimler (ör. DVWA, Remot3d, tarayıcı odaklı test araçları)
- **Brute force / Spoofing:** Kimlik doğrulama denemeleri ve ağ sahteciliği senaryoları (ör. hydra, ettercap)

Ağ trafiği, saldırgan ve kurban ağlar arasına konumlandırılan Gigamon Network TAP üzerinden pasif olarak kopyalanmış ve iki ayrı monitör arayüzü üzerinden Wireshark ile pcap formatında kaydedilmiştir. TAP kullanımı sayesinde normal ağ operasyonları etkilenmeden müdahalesiz ve pasif izleme sağlanmıştır. İki monitör portundan elde edilen pcap çıktıları, her deney için mergecap kullanılarak tek bir pcap dosyasında birleştirilmiştir.

Etiketleme sürecinde, her deney ayrı yürütüldüğü için ilgili deney boyunca kaydedilen trafik, o senaryonun saldırı etiketiyle ilişkilendirilmiş; normal trafik oturumu ise normal trafik sınıfı altında ayrı şekilde tutulmuştur. Toplanan ham trafik toplamı yaklaşık 548 GB büyüklüğünde bir veri seti oluşturmaktadır.

3.4. Saldırı Türleri ve Kategoriler

CICIoT2023 veri setinde yer alan saldırılar, IoT ortamlarında pratikte karşılaşılan tehdit profillerini ve protokol dağılımlarını temsil edecek şekilde Şekil 3.3 ve Şekil 3.4'te gösterildiği üzere tasarlanmış ve yedi ana kategori altında sınıflandırılmıştır: DDoS, DoS, Recon (Keşif), Web-based, Brute force, Spoofing ve Mirai. Bu sınıflandırma, saldırıların yalnızca "isim" düzeyinde ayrıştırılmasını değil; saldırının amacı (hizmet kesintisi, bilgi toplama, yetkisiz erişim, trafiği saptırma vb.), uygulandığı katman (ağ/taşıma/uygulama katmanı), üretim biçimi (tek kaynak/çoklu kaynak, düşük hacimli/sürekli, kısa süreli/persistan) ve trafik örüntüsü (bayrak davranışları, paket aralıkları, akış süresi, paket boyutu dağılımları) gibi açılardan da sistematik biçimde analiz edilmesini kolaylaştırır. Bu nedenle kategoriler, makine öğrenmesi tabanlı saldırı

tespiti çalışmalarında hem problem tanımını netleştirir hem de model performansının hangi saldırı ailesinde güçlendiğini/zayıfladığını görünür kılar.

DDoS kategorisi, hedef sistemin hizmet veremez hale gelmesini amaçlayan ve genellikle dağıtık biçimde (birden fazla saldırgan düğüm/çoklu kaynak) üretilen yoğun trafik senaryolarını kapsar. DDoS saldırıları, ağ kapasitesini doyurma (bandwidth exhaustion), hedefin bağlantı tablolarını şişirme (state exhaustion) veya uygulama katmanında kaynak tüketimi (CPU/iş parçacığı/oturum tüketimi) gibi farklı mekanizmalarla etkili olabilir. Veri setinde DDoS ailesi, farklı protokollere ve farklı yoğunluk stratejilerine dayanan alt türlerle çeşitlendirilmiştir. Örneğin parçalama temelli trafik üretimi, bazı güvenlik/filtreleme mekanizmalarını zorlayabilecek trafik örüntüleri oluşturabilir; UDP/ICMP tabanlı flood senaryoları bant genişliği ve paket işleme kapasitesi üzerinde baskı kurabilir; TCP bayrak örüntülerini kullanan saldırılar ise hedefin bağlantı yönetimini ve durum tabanlı kaynaklarını tüketmeyi hedefleyebilir. Bu alt türlerin çeşitliliği, DDoS sınıfının "tek tip bir flood" olmadığını; farklı protokol ve bayrak davranışlarıyla çok farklı akış karakteristikleri üretebildiğini gösterir.

DoS kategorisi, hedef hizmetin kullanılabilirliğini düşürmeyi amaçlayan ancak "dağıtık" olmaktan ziyade çoğunlukla tekil veya sınırlı sayıda kaynak üzerinden yürütülen saldırı senaryolarını içerir. DoS saldırıları da DDoS gibi hizmet kesintisi hedefler; fakat pratikte saldırı kaynağının sayısı/dağılımı ve dolayısıyla trafik üretim dinamiği farklıdır. Veri setinde DoS kategorisi, TCP/UDP/SYN/HTTP temelli saldırı türlerini içerecek şekilde ele alınır. TCP/UDP flood gibi senaryolar ağ ve sistem kaynaklarını doğrudan tüketmeye yönelirken, SYN flood bağlantı kurulumu aşamasında hedefi zorlayarak oturum yönetim kaynaklarını tüketmeye çalışır. HTTP flood ve benzeri uygulama katmanı saldırıları ise paket sayısı kadar istek işleme yükünü de büyütür. Ayrıca, düşük bant genişliği ile uzun süreli bağlantı tüketimine odaklanan teknikler (ör. yavaş istek/bağlantı tüketimi yaklaşımı) uygulama katmanı iş parçacığı/oturum kaynakları üzerinden etkili olabilir. Bu nedenle DoS kategorisi, "yüksek hacim" kadar "akıllı kaynak tüketimi" yaklaşımını da temsil eder.

Recon (Keşif) kategorisi, doğrudan zarar vermekten önce hedef hakkında bilgi toplamayı amaçlayan saldırı öncesi faaliyetleri kapsar. Keşif saldırıları, bir saldırganın ağdaki aktif cihazları belirlemesi, hangi servislerin çalıştığını tespit etmesi, olası işletim sistemi/servis sürümü çıkarımları yapması ve zafiyet yüzeyini anlaması için kritik bir hazırlık aşamasıdır. Bu kategoride yer alan senaryolar; host discovery / ping sweep ile aktif cihazların belirlenmesi, port tarama ile açık servislerin ve dinleyen portların

tespiti, OS/servis keşfi ile hedef sistem profilinin çıkarımı ve zafiyet tarama ile bilinen açıklıkların işaretlenmesi gibi adımları içerebilir. Recon trafiği çoğu zaman kısa süreli, düşük hacimli ve belirli hedef portlara odaklıdır; bu da onu DDoS/DoS gibi yüksek hacimli saldırılardan ayırır. Ancak özellikle geniş adres aralıklarında yapılan taramalar, çok sayıda kısa akış ve belirgin zamanlama örüntüleri oluşturarak veri setinde ayırt edilebilir bir iz bırakır.

Web-based kategorisi, web uygulamalarını veya web servislerini hedefleyen saldırı senaryolarını temsil eder. Bu sınıfta, uygulama katmanındaki girdi doğrulama hataları ve zayıf güvenlik kontrolleri üzerinden istismar amaçlanır. Örneğin SQL Injection, uygulamanın veritabanı sorgularını manipüle ederek veri sızdırma veya yetkisiz işlemler yapma hedefi taşır. XSS (Cross-Site Scripting), istemci tarafında zararlı betik çalıştırmayı hedefleyerek oturum çalma, kullanıcıyı yönlendirme veya içerik manipülasyonu gibi etkiler üretebilir. Command Injection türü senaryolar, sunucu tarafında komut yürütmeye kadar uzanan kritik sonuçlara yol açabilir. Web-based kategorisi ayrıca dosya yükleme zafiyetleri (uploading attack), arka kapı/malware yerleştirme (backdoor malware) ve tarayıcı/istemci yönlü manipülasyon (browser hijacking) gibi alt türlerle genişletilerek yalnızca "klasik OWASP" türlerini değil, web ekosisteminde görülen daha geniş bir saldırı yelpazesini yansıtır. Bu saldırıların trafik izleri, uygulama katmanı istek yanıt örüntüleri, akış süreleri ve paket boyutu dağılımları gibi metriklerde belirginleşebilir.

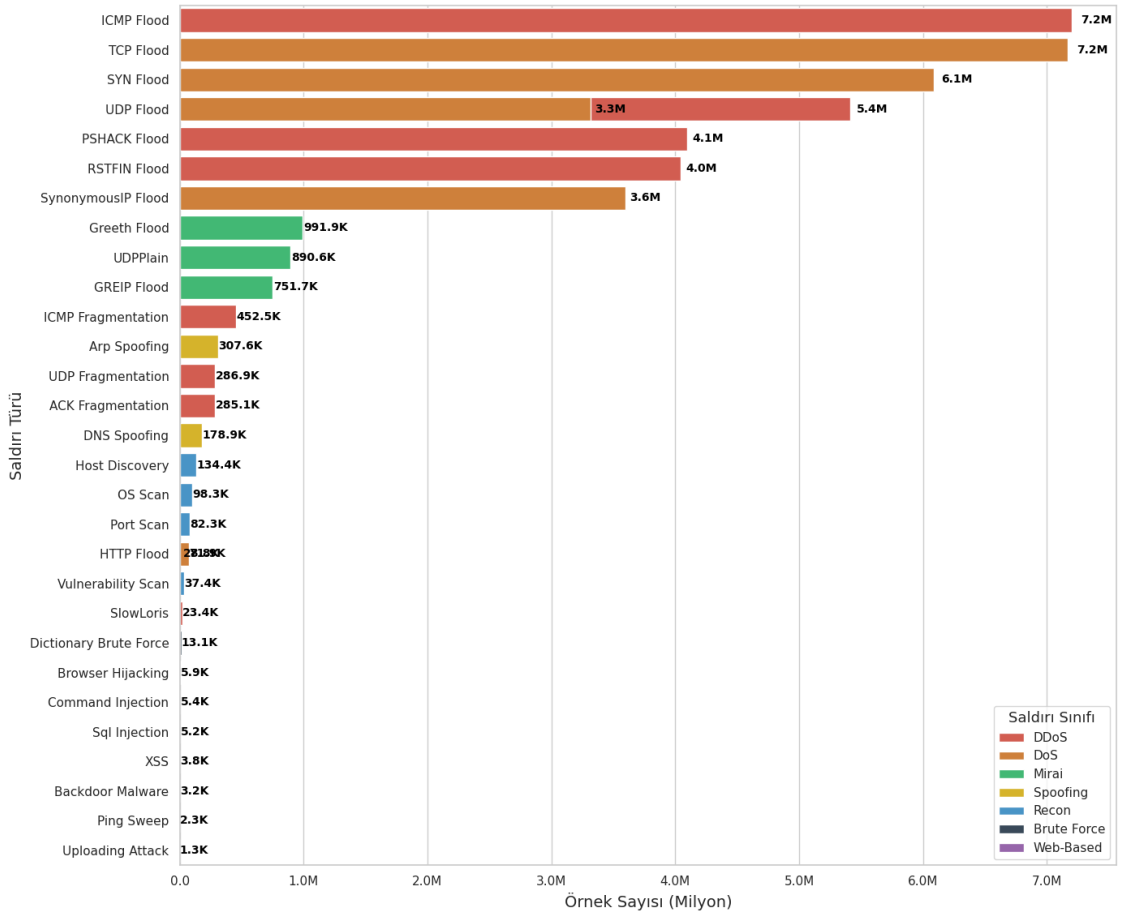
Brute force kategorisi, kimlik doğrulama mekanizmalarını deneme-yanılma veya sözlük listeleri üzerinden aşmayı hedefleyen saldırıları kapsar. Bu tür saldırılarda amaç, zayıf parola politikaları, yetersiz hız sınırlama, hatalı kilitleme politikaları veya kötü yapılandırılmış kimlik doğrulama servisleri üzerinden yetkisiz erişim elde etmektir. Brute force trafiği çoğu zaman art arda deneme girişimleriyle karakterizedir ve başarısız oturum açma tekrarları, belirli servis/portlarda yoğunlaşma ve zamansal tekrar örüntüleri gibi sinyaller üretebilir. IoT bağlamında bu sınıf, özellikle yönetim arayüzleri veya uzaktan erişim servisleri üzerinden denenebilen girişimleri temsil eder.

Spoofing kategorisi, ağ iletişiminde kimlik/isim çözümleme mekanizmalarını manipüle ederek trafiği saptırmayı, araya girme (Man-In-the-Middle MITM) koşulları oluşturmayı veya kullanıcı/cihazları sahte hedeflere yönlendirmeyi amaçlayan saldırıları içerir. ARP spoofing, yerel ağda IP–MAC eşleştirmelerini bozarak trafiğin saldırgana akmasına yol açabilir. DNS spoofing ise alan adı çözümleme sürecini manipüle ederek kullanıcı/cihazların yanlış IP adreslerine yönlenmesine neden olabilir. Bu tür saldırılar,

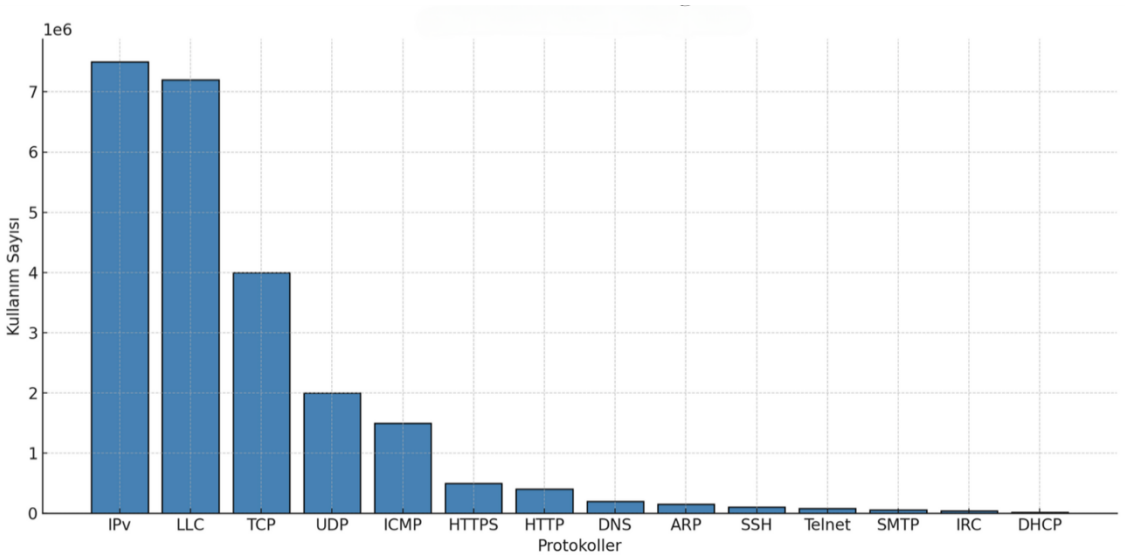
özellikle yerel ağ içinde güven ilişkileri ve isim çözümleme süreçlerinin zayıf olduğu ortamlarda kritik risk üretir. Trafik düzeyinde, ARP/DNS paketleri ve anormal çözümleme davranışları bu sınıfı ayırt etmede önem kazanır.

Mirai kategorisi, IoT cihazlarını hedef alan botnet davranışlarını temsil eden saldırı ailesini kapsar. Mirai benzeri botnetler, IoT cihazlarını ele geçirerek onları uzaktan komuta edilebilir saldırı düğümlerine dönüştürür ve çoğunlukla geniş ölçekli servis kesintisi saldırılarında kullanılır. Bu kategori altında, botnetin üretebileceği farklı yoğunluk ve protokol örüntülerini yansıtan alt türler bulunur (örn. UDP tabanlı yoğun trafik senaryoları ve belirli paket desenleri). Bu sınıfın önemi, IoT ekosisteminde gerçek dünyada sıkça görülen "zayıf parola + otomatik yayılım + kitlesel saldırı" döngüsünü temsil etmesidir. Mirai kategorisi, yalnızca DoS/DDoS sonucu değil, botnet davranışlarının ağ trafiğine yansıyan karakteristik örüntülerini de kapsadığı için ayrı bir aile olarak ele alınır.

Sonuç olarak, bu yedi kategorilik yapı, veri setinin saldırıları hem pratik siber güvenlik taksonomisiyle hem de makine öğrenmesi deneylerinin değerlendirme ihtiyacıyla uyumlu biçimde düzenlemesini sağlar. Bununla birlikte, bazı saldırı türleri trafik açısından birbirine benzer örüntüler üretebilir (ör. farklı flood türleri) veya farklı kategoriler arasında sınır durumları oluşabilir (ör. yoğunluk düzeyi ve kaynak sayısı bakımından DoS–DDoS ayrımı). Bu nedenle tez kapsamında, saldırı isimlerinin veri setindeki etiketlerle tutarlı kullanılması ve kategori açıklamalarının "mutlak" iddialar yerine "temsil ettiği davranış/amaç" ekseninde verilmesi, metni daha savunulabilir hale getirir.



Şekil 3.3. CICIoT2023 veri seti saldırı türleri ve sınıf dağılımı



Şekil 3.4. CICIoT2023 veri seti protokol kullanımı dağılımı

4. ARAŞTIRMA SONUÇLARI VE TARTIŞMA

4.1. Veri Seti ve Ön İşleme

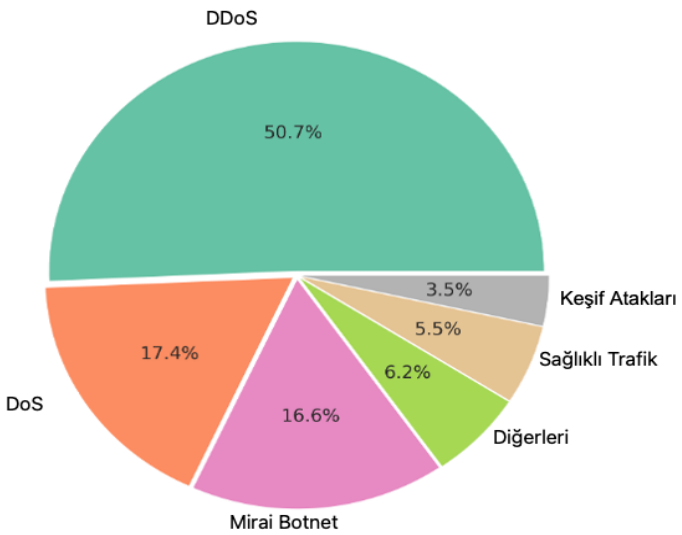
Bu çalışmada, Canadian Institute for Cybersecurity tarafından yayımlanan CIC-IoT-2023 veri seti kullanılmıştır. Orijinal veri seti yaklaşık 50 milyon ağ akışı kaydından oluşmakta olup, 34 farklı trafik sınıfını (1 normal, 33 saldırı türü) içermektedir. Veri setinde 46 adet sayısal özellik yer almaktadır.

4.1.1. Veri seti analizi ve sınıf dağılımı

CIC-IoT-2023 veri setinin en belirgin özelliği, sınıflar arasındaki aşırı dengesizliktir. Orijinal veri setinde en sık görülen sınıf olan DDoS-TCP_Flood milyonlarca örnekle temsil edilirken, Uploading_Attack gibi nadir saldırı türleri yalnızca birkaç bin örnekle yer almaktadır. Bu durum, 399:1 oranında bir sınıf dengesizliği yaratmaktadır.

Saldırı kategorileri incelendiğinde, DDoS saldırılarının veri setinin %50,7'sini, DoS saldırılarının %17,4'ünü, Mirai botnet saldırılarının %16,6'sını oluşturduğu görülmektedir (bkz. Şekil 4.1).

Sınıf dengesizliği, makine öğrenmesi modellerinin çoğunluk sınıflarına yönelmesine ve azınlık sınıflarını öğrenememesine neden olmaktadır. Bu problemin çözümü için stratejik bir örnekleme yaklaşımı benimsenmiştir.



Şekil 4.1. Saldırı kategori dağılımı

4.1.2. Örnekleme stratejisi

Sınıf dengesizliği problemine çözüm getirmek amacıyla çok katmanlı bir örnekleme stratejisi uygulanmıştır. Bu strateji, hem aşırı temsil edilen çoğunluk sınıflarını kontrol altına almayı hem de yetersiz temsil edilen azınlık sınıflarını güçlendirmeyi hedeflemektedir.:

Altörnekleme (Undersampling); orijinal CIC-IoT-2023 veri setinde bazı saldırı türleri (özellikle DDoS ve DoS kategorileri) milyonlarca örnekle temsil edilmektedir. Bu durum, hem hesaplama maliyetini artırmakta hem de modelin çoğunluk sınıflarına aşırı odaklanmasına (bias) neden olmaktadır. Bu sorunu çözmek için, 500.000'den fazla örneğe sahip sınıflar (DDoS, DoS, Mirai Botnet) rastgele altörnekleme ile 500.000 örneğe indirilmiştir (Carvalho vd., 2025). Bu işlem sonucunda orijinal veri seti 9.027.560 örneğe düşürülmüş ve sınıflar arası dengesizlik oranı kontrol edilebilir bir seviyeye getirilmiştir.

Altörnekleme sonrası elde edilen 9.027.560 örneklik veri seti %80 eğitim (7.222.048 örnek) ve %20 test (1.805.512 örnek) olarak bölünmüştür. Bu aşamada oluşturulan test seti bundan sonra yapılacak tüm veri önişleme adımlarından izole edilmiş ve modellerin eğitimi sonrasında test aşamasında kullanılacaktır. Bu yaklaşım, test setinin gerçek dünya dağılımını temsil etmesini sağlamakta ve modelin genelleme yeteneğinin doğru bir şekilde değerlendirilmesine olanak tanımaktadır.

Üstörnekleme; bir önceki adımda ayrılan eğitim setinde (7.222.048 örnek) hala önemli bir dengesizlik bulunmaktadır. Bazı sınıflar (örn. DDoS-ICMP_Flood) 100.000'den fazla örnekle temsil edilirken, bazı sınıflar (örn. Uploading_Attack, Recon-PingSweep, XSS) 250-1.000 aralığında çok az örnekle temsil edilmektedir. Bu azınlık sınıfları için SMOTE (Synthetic Minority Over-sampling Technique) algoritması uygulanarak sentetik örnekler üretilmiştir (Chawla vd., 2002).

SMOTE, azınlık sınıfına ait bir örnek ile bu örneğin k en yakın komşusu arasında doğrusal interpolasyon yaparak sentetik örnekler oluşturur. Bu yaklaşım, basit duplikasyondan farklı olarak özellik uzayında yeni noktalar üretir ve modelin azınlık sınıflarının karar sınırlarını daha iyi öğrenmesini sağlar.

SMOTE uygulaması sonucunda, 10.000'den az örneğe sahip 7 sınıf (Browser hijacking, command injection, sql injection, xss, backdoor malware, ping sweep ve uploading attack) için toplam 48.326 sentetik örnek üretilmiş ve eğitim seti 7.270.374

örneğe ulaşmıştır. Sınıflar arası dengesizlik oranı 399:1'den 40:1'e düşürülmüştür. Örnekleme stratejisinin sonuçları Tablo 4.1'de gösterilmektedir.

Tablo 4.1. Örnekleme stratejisi sonuçları

Aşama	**Veri Boyutu**	**Dengesizlik Oranı**	**Açıklama**
Orijinal Veri Seti	~49M	~2000:1	Çok yüksek dengesizlik
Altörnekleme Sonrası	9.027.560	~400:1	Çoğunluk sınıfları kontrol altında
Eğitim/Test	Eğitim: 7.222.048 Test: 1.805.512	~400:1 (her ikisi de)	Sınıf dağılımı korunarak bölme
SMOTE Sonrası	Eğitim: 7.270.374 Test: 1.805.512	Eğitim: ~40:1 Test: ~400:1	Eğitim dengeli, test gerçekçi

4.1.3. Özellik ölçeklendirme

Sayısal özelliklerin ölçeklendirilmesi için Quantile Transformer yöntemi tercih edilmiştir (De Amorim vd., 2023). Bu yöntem, özellikleri normal dağılıma dönüştürmekte ve aykırı değerlere karşı dayanıklılık sağlamaktadır. Quantile Transform, n_quantiles=1.000 parametresiyle ve çıktı dağılımı olarak normal dağılım seçilerek uygulanmıştır. StandardScaler gibi alternatif yöntemler yerine Quantile Transformer'ın tercih edilmesinin nedenleri:

- Ağ trafiği verilerinde sıkça görülen aşırı aykırı değerlere karşı daha dayanıklıdır.
- Sinir ağlarının normal dağılımlı girdilerle daha iyi performans gösterdiği bilinmektedir.
- Non-parametrik bir yöntem olması sayesinde, veri dağılımı hakkında herhangi bir varsayımda bulunmaya gerek kalmamaktadır.

Tüm ön işleme adımlarında tekrarlanabilirliği sağlamak için sabit rastgele tohum değeri (random_state=47) kullanılmıştır.

4.2. Model Geliştirme

Bu bölümde, TabM model mimarisi, hiperparametre optimizasyon süreci ve eğitim konfigürasyonu detaylandırılmaktadır.

4.2.1. TabM model mimarisi

Çalışmada, tabular veriler için özel olarak tasarlanmış TabM (Tabular Model) mimarisi kullanılmıştır (Gorishniy vd., 2025a). TabM, geleneksel Çok Katmanlı Algılayıcı (Multi-Layer Perceptron-MLP) mimarisini BatchEnsemble yaklaşımıyla birleştirerek, tek bir model yerine birden fazla alt model çıktılarını birleştirmektedir. Bu yapı, modelin daha sağlam sonuçlar üretmesini sağlamakta ve parametre verimliliği açısından geleneksel topluluk yöntemlerine göre önemli avantajlar sunmaktadır.

TabM'in temelinde, BatchEnsemble adı verilen parametre-verimli topluluk öğrenme tekniği bulunmaktadır. Geleneksel derin topluluk yaklaşımında, her bir model bağımsız olarak eğitilir ve her modelin kendi parametre setine ihtiyacı vardır. Örneğin, 5 modelden oluşan bir toplulukta, her model 100K parametreye sahipse, toplam 500K parametre gerekir. TabM ise, tüm alt modellerin çoğu parametreyi paylaşmasını sağlayarak bu maliyeti dramatik şekilde azaltır.

TabM mimarisi, ardışık MLP blokları (n_blocks) ve her blokta d_block boyutunda gizli katmanlardan oluşur. Her blok, BatchEnsemble mekanizması ile k adet implicit alt model içerir. Bu çalışmada kullanılan konfigürasyon şu şekildedir:

- **k (topluluk boyutu):** 32 submodel
- **d_block (gizli katman boyutu):** 256 nöron
- **n_blocks (blok sayısı):** 2 ardışık blok
- **dropout:** %10 regularizasyon

Her bir girdi örneği için, TabM 32 ayrı tahmin üretir. Final tahmin, bu 32 altmodelin çıktılarının ortalaması alınarak elde edilir. Bu topluluk yaklaşımı, bireysel tahminlerdeki belirsizliği azaltır ve modelin genelleme yeteneğini artırır.

TabM'in en önemli avantajlarından biri, parametre verimliliğidir. Geleneksel bir derin topluluklarda, 32 ayrı MLP modeli eğitmek için her modelin parametreleri bağımsız olarak saklanmalıdır. Örneğin, her model 400K parametreye sahipse, toplam 12,8M parametre gerekir. TabM'de ise, BatchEnsemble mekanizması sayesinde, toplam parametre sayısı 399.360 ile sınırlı kalmaktadır. Bu, model boyutunun 1.52 MB olmasını sağlar ve modelin uç birimlerde (IoT ağ geçidi, akıllı ev hub'ları, Raspberry Pi gibi kaynak kısıtlı cihazlar) senaryoları için uygun olmasını mümkün kılar.

TabM'in forward pass'i, geleneksel MLP'den farklı olarak, her mini-batch için k adet paralel hesaplama yolu içerir. Bir girdi x için:

1. İlk katman, x'i tüm k adet alt model için işler.
2. Her alt modelin, kendi r ve s vektörleri ile modifiye edilmiş ağırlıkları kullanır.
3. Çıktı katmanında, k adet tahmin üretilir ve bunların ortalaması final tahmin olarak kullanılır.

Eğitim sırasında, loss fonksiyonu ortalama tahmin üzerinden hesaplanır ve gradyanlar tüm alt modellere geri yayılır . Bu sayede, tüm k adet alt model aynı anda ve birlikte öğrenir. Geleneksel ensemble'da her model bağımsız eğitilirken, TabM'de alt modellerin birlikte eğitilmesi, parametrelerin koordineli bir şekilde optimize edilmesini sağlar.

Tabular veriler için derin öğrenme alanında FT-Transformer, TabNet, SAINT gibi çeşitli mimariler önerilmiştir. Ancak Gorishniy ve diğerlerinin (Gorishniy vd., 2025b) 40+ veri seti üzerinde yaptığı kapsamlı değerlendirmede, dikkat (attention) tabanlı modellerin MLP tabanlı modellere göre tutarlı bir üstünlük sağlamadığı gösterilmiştir. TabM, MLP'nin basitliğini BatchEnsemble'ın gücüyle birleştirerek hem yüksek performans hem de parametre verimliliği sunmaktadır. Bu nedenle, çalışmamızda TabM mimarisi tercih edilmiştir.

TabM mimarisinin temel parametreleri şunlardır:

- **k (topluluk sayısı):** Model içindeki paralel alt model sayısını belirler. Bu çalışmada hiperparametre optimizasyonu sonucunda k=32 değeri seçilmiştir. Bu değer, TabM makalesinde önerilen optimal değerle örtüşmektedir.
- **d_block (blok boyutu):** Her bloktaki gizli katman boyutunu ifade eder. Hiperparametre optimizasyonu sonucunda d_block=256 olarak belirlenmiştir.
- **n_blocks (blok sayısı):** Ardışık blok sayısını belirler. Hiperparametre optimizasyonu sonucunda n_blocks=2 değeri kullanılmıştır.
- **dropout:** Aşırı öğrenmeyi önlemek için %10 dropout oranı uygulanmıştır.

Sonuç olarak, model toplam 399.360 eğitilebilir parametre içermekte olup, model boyutu yaklaşık 1,52 MB'tır. Bu kompakt yapı, modelin üretim ortamlarında kolay dağıtılabilmesini ve gerçek zamanlı saldırı tespiti senaryolarında kullanılabilmesini sağlamaktadır. Tablo 4.2'de model konfigürasyonu özetlenmektedir.

Tablo 4.2. TabM model konfigürasyonu

Parametre	**Değer**
Topluluk Sayısı (k)	32
Gizli Katman Boyutu (d_block)	256
Blok Sayısı (n_block)	2
Dropout Oranı	0,10
Giriş Özellikleri	46
Çıkış Sınıfları	34

4.2.2. Hiperparametre optimizasyonu

Hiperparametre optimizasyonu için Optuna kütüphanesi (Shimazoe vd., 2023) kullanılarak Bayesian optimizasyon yaklaşımı benimsenmiştir. Grid search veya random search yöntemlerine göre Bayesian optimizasyon, hiperparametre uzayını daha verimli taramakta ve daha az deneme ile optimum değerlere ulaşmaktadır. Optuna, Tree-structured Parzen Estimator (TPE) algoritmasını kullanarak her deneme sonucunu değerlendirmekte ve bir sonraki deneme için en umut verici hiperparametre kombinasyonlarını önermektedir.

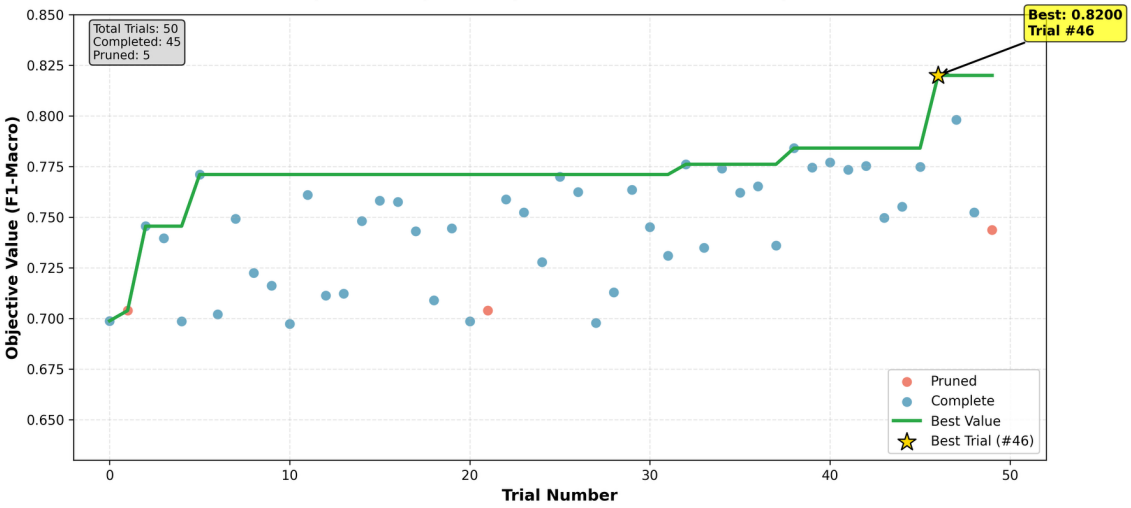
Optimizasyon süreci aşağıdaki arama uzayını kapsamaktadır:

- **Deneme sayısı:** 50 deneme
- **Öğrenme oranı:** 10^{-4} ile 10^{-2} arasında log-uniform dağılım
- **k değeri:** {16, 24,32} kategorik seçenekler
- **d_block değeri:** {128, 256, 512} kategorik seçenekler
- **n_blocks değeri:** {2, 3} kategorik seçenekler
- **Weight decay:** 10^{-5} ile 10^{-3} arasında log-uniform dağılım

Hesaplama süresini optimize etmek için MedianPruner stratejisi uygulanmıştır. Bu strateji, performansı medyan değerin altında kalan denemeleri erken sonlandırarak hesaplama kaynaklarını verimli kullanmaktadır. Pruner, ilk 3 deneme tamamlanana kadar (n_startup_trials=3) ve her denemenin ilk 2 epoch'u tamamlanana kadar (n_warmup_steps=2) beklemekte ve ardından düşük performanslı denemeleri sonlandırmaktdır. Ayrıca, optimizasyon sürecinde eğitim verisinin %10'luk bir alt kümesi kullanılarak hesaplama maliyeti düşürülmüştür.

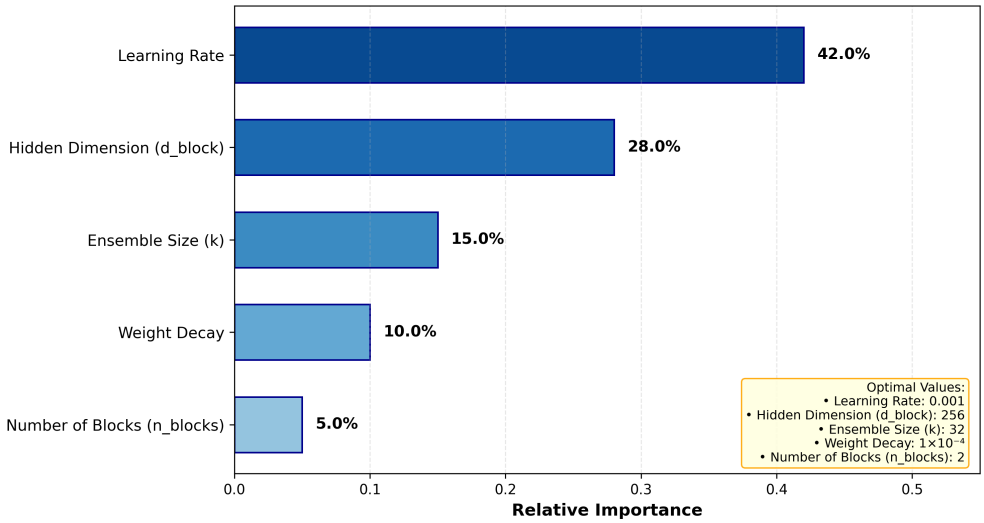
Şekil 4.2'de optimizasyon sürecinin gelişimi görselleştirilmiştir. 50 deneme boyunca F1-Makro skorunun yavaş yavaş iyileştiği ve 46. denemede en iyi değere ulaşıldığı görülmektedir. MedianPruner stratejisi sayesinde 50 denemeden 3'ü düşük

performans nedeniyle erken sonlandırılmış, bu da toplam hesaplama süresini önemli ölçüde azaltmıştır.



Şekil 4.2. Optimizasyon süreci

Şekil 4.3'de gösterilen hiperparametre önem analizi, öğrenme oranının en kritik parametre olduğunu ortaya koymaktadır (%42). Bunu sırasıyla gizli katman boyutu d_block (%28), topluluk sayısı k (%15), weight decay (%10) ve blok sayısı n_blocks (%5) takip etmektedir. Bu bulgu, TabM modelinin öğrenme oranına karşı hassas olduğunu ve bu parametrenin dikkatli ayarlanması gerektiğini göstermektedir.



Şekil 4.3. Hiperparametre önem analizi

Optimizasyon sonucunda elde edilen en iyi hiperparametreler Tablo 4.3'de sunulmaktadır.

Tablo 4.3. Optimizasyon sonucu elde edilen değerleri

Parametre	**Arama Uzayı**	**Optimal Değer**
Öğrenme Oranı	$[10^{-4}, 10^{-2}]$	0.001
Ensemble Boyutu (k)	{16, 32}	32
Gizli Boyut (d_block)	{128, 256, 512}	256
Blok Sayısı (n_blocks)	{2, 3}	2
Weight Decay	$[10^{-5}, 10^{-3}]$	1×10^{-4}

4.2.3. Eğitim konfigürasyonu

Model eğitimi için aşağıdaki konfigürasyon kullanılmıştır:

Kayıp fonksiyonu: Ablasyon çalışması sonucunda (bkz. Bölüm 4.5.2), standart Cross-Entropy Loss (Terven vd., 2025) fonksiyonunun en iyi performansı sergilediği belirlenmiştir. TabM mimarisinin çalışma prensipleri doğrultusunda sınıf dengesizliği problemlerinde yaygın olarak tercih edilen Focal Loss (Lin vd., 2017) veya sınıf ağırlıkları kullanımının bu veri setinde performansı düşürdüğü gözlemlenmiştir. Bu durum, SMOTE ile dengelenmiş veri setinde ek ağırlıklandırma mekanizmalarının gereksiz, hatta zararlı olabileceğini göstermektedir.

Sınıf ağırlıkları: Ablasyon çalışmasında, sınıf ağırlıkları kullanılmamıştır. Sınıf ağırlıkları, her sınıfın örnek sayısının ters frekansı olarak hesaplanmakta ve loss fonksiyonunda azınlık sınıflarına daha yüksek ağırlık vermektedir. Ancak, SMOTE ile tüm sınıflar 10.000 örneğe dengelendiğinde, orijinal veri dağılımına göre hesaplanan sınıf ağırlıkları aşırı düzeltme (over-correction) yaratmaktadır. Ablasyon çalışmasında, sınıf ağırlıkları eklendiğinde F1-Makro skoru 0,7841'den 0,7397'ye düşmüştür (-0,0444). Bu bulgu, SMOTE'un sınıf dengesizliğini yeterince ele aldığını ve ek ağırlıklandırmanın gereksiz, hatta zararlı olduğunu göstermektedir.

Optimizer: AdamW (Adaptive Moment Estimation with Decoupled Weight Decay) optimizer tercih edilmiştir (Loshchilov ve Hutter, 2019). TabM mimarisinin orijinal çalışmasında da AdamW optimizer önerilmektedir (Gorishniy vd., 2025b). Hiperparametre optimizasyonu sonucunda belirlenen learning rate (0,001) ve weight decay (0,0001) değerleri kullanılmıştır.

Eğitim parametreleri Tablo 4.4'de özetlenmektedir.

Tablo 4.4. Eğitim parametreleri

Parametre	**Değer**	**Açıklama**
Batch Size	8192	GPU bellek kullanımını optimize etmek için
Maksimum Epoch	50	Üst sınır (early stopping ile erken sonlanabilir)
Early Stopping Patience	16 epoch	Aşırı öğrenmeyi önlemek için
Gradient Clipping	max_norm=1,0	Patlayan gradyan problemini önlemek için
Learning Rate	0,001	Optuna ile optimize edilmiş
Weight Decay	1×10^{-4}	L2 regularizasyon için
Loss Function	CrossEntropy	Ablasyon çalışmasına göre seçilmiş
Class Weights	Hayır	Ablasyon çalışmasına göre devre dışı

Donanım ve Yazılım Ortamı: Model eğitimi NVIDIA RTX 4090 GPU (24 GB VRAM) üzerinde gerçekleştirilmiştir. Eğitim süresini minimize etmek için aşağıdaki optimizasyonlar uygulanmıştır:

- **cuDNN Benchmark:** Otomatik kernel seçimi için etkinleştirilmiştir
- **TensorFloat-32 (TF32):** Ampere mimarisinin sunduğu hızlı matris çarpımı için etkinleştirilmiştir.
- **Pin Memory:** CPU-GPU veri transferini hızlandırmak için DataLoader'da etkinleştirilmiştir.
- **Persistent Workers:** Epoch'lar arası worker yeniden başlatma maliyetini ortadan kaldırmak için etkinleştirilmiştir.

Bu optimizasyonlar sayesinde, 7,2 milyon örnek üzerinde tam bir eğitim yaklaşık 17 dakika sürmüştür.

4.3. Model Geliştirme

Bu bölümde, geliştirilen modelin performansı çapraz doğrulama, test seti değerlendirmesi ve sınıf bazlı analiz yöntemleriyle incelenmektedir.

4.3.1. Çapraz doğrulama sonuçları

Model performansının güvenilir bir şekilde değerlendirilmesi için 5-fold çapraz doğrulama uygulanmıştır (T R vd., 2023). Çapraz doğrulama yaklaşımı, her katta sınıf

dağılımının korunmasını sağlayarak dengesiz veri setlerinde daha güvenilir sonuçlar üretmektedir. Çapraz doğrulama, modelin farklı veri alt kümelerinde tutarlı performans gösterip göstermediğini test etmek için kritik bir değerlendirme yöntemidir (Lee ve Ahn, 2023).

Çapraz Doğrulama Parametreleri: Hesaplama verimliliği göz önünde bulundurularak, her fold için maksimum 20 epok ve erken durdurma değeri 8 olarak belirlenmiştir. Bu parametre seçimi, tam eğitim konfigürasyonuna (50 epok, erken durdurma=16) göre daha kısa olmasına rağmen, modelin kararlı hale gelmesi için yeterli olduğu gözlemlenmiştir. Çapraz doğrulamanın amacı, modelin nihai performansını ölçmek değil, farklı veri bölümlemelerine karşı tutarlılığını ve genelleme yeteneğini değerlendirmektir. Çapraz doğrulama sonuçları Tablo 4.5'de özetlenmektedir.

Tablo 4.5. Çapraz doğrulama sonuçları

Kat	F1-Makro	F1-Weighted	Accuracy
1	0,7910	0,9731	0,9694
2	0,7837	0,9704	0,9662
3	0,7811	0,9713	0,9670
4	0,7817	0,9715	0,9671
5	0,7811	0,9711	0,9667
Ortalama	0,7837	0,9715	0,9673
Standart Sapma	±0,0040	±0,0010	±0,0012

Sonuçlar incelendiğinde, modelin beş kat boyunca tutarlı bir performans sergilediği görülmektedir. F1-Makro için standart sapmanın ±0,004 gibi son derece düşük bir değerde olması, modelin veri varyasyonlarına karşı robust olduğunu ve aşırı uyum (overfitting) probleminin bulunmadığını göstermektedir. Bu düşük standart sapma değeri, 20 epokluk eğitimin çapraz doğrulama amacı için yeterli olduğunu ve modelin kararlı bir convergence noktasına ulaştığını kanıtlamaktadır.

Katlar arasındaki performans farkı incelendiğinde, kat 1'in diğer katlara göre marjinal olarak daha yüksek performans gösterdiği (0,7910) görülmektedir. Bu durum, rastgele veri bölümlemesinden kaynaklanan doğal varyasyonun bir yansımasıdır ve istatistiksel olarak anlamlı bir fark oluşturmamaktadır. Tüm katlardaki F1-Makro skorlarının 0,78-0,79 aralığında kümelenmiş olması, modelin farklı veri alt kümelerinde tutarlı performans sergilediğini göstermektedir.

4.3.2. Test seti performansı

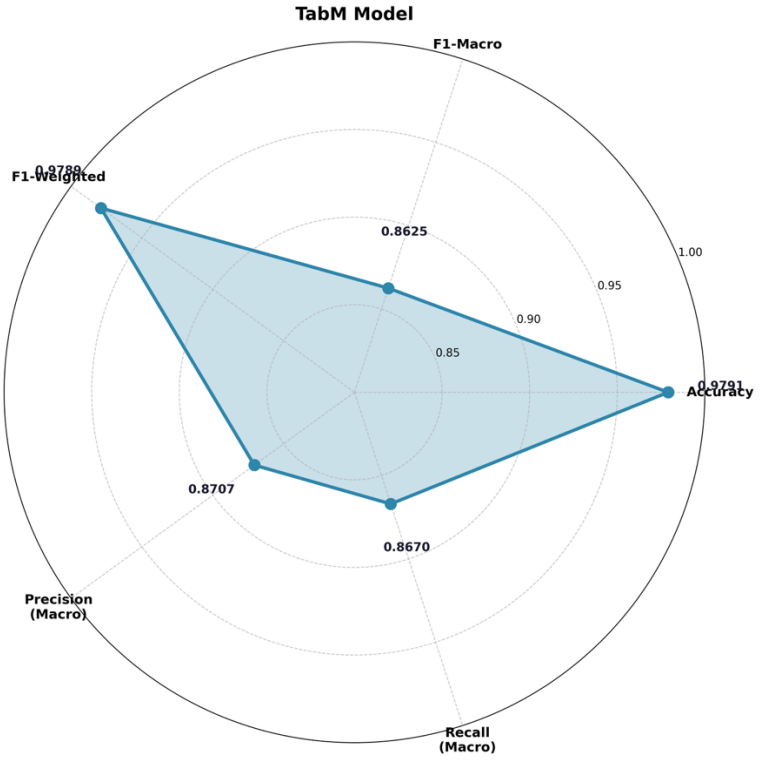
Çapraz doğrulama sonuçlarına göre modelin tutarlı bir şekilde çalıştığı görüldükten sonra TabM modeli eğitim seti olarak ayrılıp üstörnekleme sonucunda 7.270.374 örnek sayısına çıkartılan veri seti ile eğitilmiştir. Eğitim işlemi tamamlanan model altörnekleme işlemi sonucunda elde edilen veri kümesinin %20 lik kısmından oluşturulan veri seti (%20, 1.805.512 örnek) ile test edilmiştir.

Test işlemi sonucunda elde edilen sonuçlar, modelin görülmemiş veriler üzerindeki genelleme yeteneğini göstermektedir. Test seti, eğitim sürecinde hiçbir şekilde kullanılmamış olup, modelin gerçek dünya performansının en güvenilir göstergesidir. Final model, test setinde Tablo 4.6'da gösterilen performans değerlerine ulaşmıştır.

Tablo 4.6. Final model değerleri

Metrik	**Değer**
Accuracy	0,9791 (%97,91)
F1-Makro	0,8625 (%86,25)
F1-Weighted	0,9789 (%97,89)
Precision (Makro)	0,8707 (%87,07)
Recall (Makro)	0,8670 (%86,70)

Şekil 4.4'de test seti performans metriklerinin radar grafiği görselleştirilmektedir. Radar grafiği, modelin tüm metriklerde dengeli bir performans sergilediğini ortaya koymaktadır. Accuracy (%97,91) ve F1-Weighted (%97,70) değerlerinin yüksek olması, modelin çoğunluk sınıflarında mükemmel performans gösterdiğini işaret etmektedir. Öte yandan, F1-Makro (%86,25), Precision-Makro (%87,07) ve Recall-Makro (%86,70) değerlerinin birbirine yakın olması (%86-87 bandında), modelin azınlık sınıfları dahil tüm sınıflarda tutarlı performans sergilediğini göstermektedir. Özellikle kesinlik (Precision) ve duyarlılık (Recall) değerlerinin birbirine çok yakın olması (fark: %0,4), modelin false positive ve false negative oranları arasında iyi bir denge kurduğunu ortaya koymaktadır.

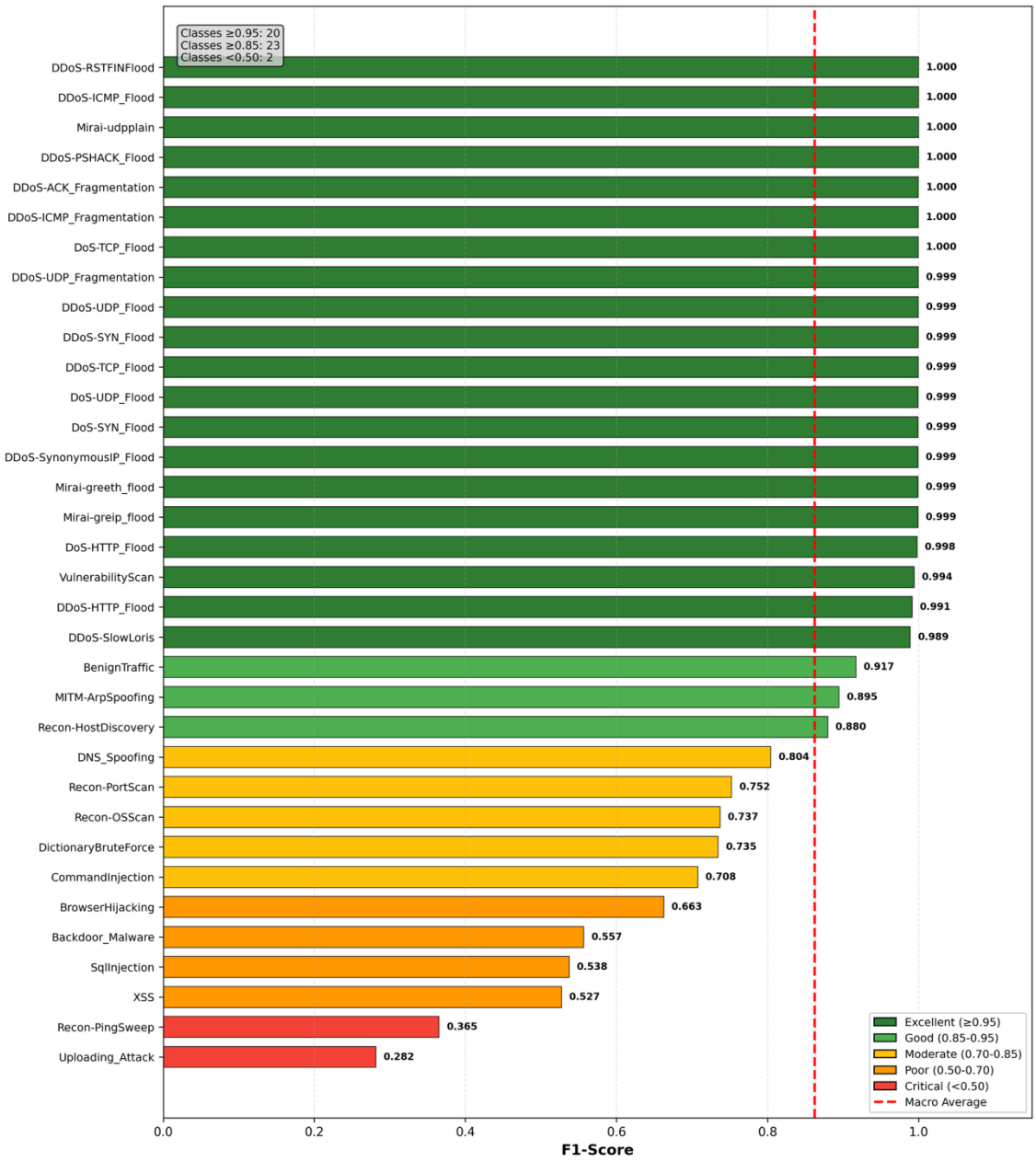


Şekil 4.4. TabM model performans metrikleri

F1-Makro ve F1-Weighted değerleri arasındaki fark (%86,25 vs %97,89), sınıf dengesizliğinin etkisini açıkça yansıtmaktadır. F1-Weighted metriği, her sınıfı örnek sayısına göre ağırlıklandırdığından çoğunluk sınıflarının yüksek performansını yansıtmaktadır. Öte yandan, F1-Makro tüm sınıfları eşit ağırlıkta değerlendirerek azınlık sınıflarındaki performans düşüklüğünü ortaya koymaktadır. Bu durum, saldırı tespit sistemlerinde kritik öneme sahiptir çünkü nadir görülen saldırıların tespit edilememesi ciddi güvenlik açıklarına yol açabilir.

4.3.3. Sınıf bazlı analiz

34 sınıfın her biri için ayrı ayrı kesinlik, duyarlılık ve F1 değerleri hesaplanmıştır. Şekil 4.5'de tüm sınıfların F1 skorları sıralı olarak görselleştirilmektedir. Bu görsel, modelin hangi saldırı türlerini başarıyla tespit ettiğini ve hangi türlerde zorlandığını açıkça ortaya koymaktadır.



Şekil 4.5. Sınıf bazlı F1 skorları

Analiz sonuçlarına göre model:

- **20 sınıf** için yüksek performans göstermektedir (F1 ≥ 0,95)
- **23 sınıf** için iyi veya üzeri performans göstermektedir (F1 ≥ 0,85)
- **2 sınıf** için kritik düzeyde düşük performans göstermektedir (F1 < 0,50)

Yüksek performanslı sınıflar: DDoS, DoS ve Mirai botnet saldırılarında model mükemmel performans sergilemiştir. Bu saldırı türlerinde F1 değerleri 0,99'un üzerine çıkmaktadır. Tablo 4.7'de en yüksek performanslı beş sınıf sunulmaktadır. Bu sınıfların yüksek performansının nedenleri:

- **Yeterli eğitim verisi:** Her sınıf için 50.000+ örnek bulunmaktadır, bu da modelin saldırı kalıplarını detaylı şekilde öğrenmesini sağlamaktadır.
- **Ayırt edici özellikler:** Bu saldırı türleri, benzersiz ağ trafiği kalıplarına sahiptir.
- **Sınıf içi tutarlılık (intra-class consistency):** Aynı saldırı türündeki örnekler birbirine benzer ağ trafiği kalıpları göstermektedir. Örneğin, DDoS-ICMP_Flood sınıfındaki tüm örnekler yüksek ICMP paket oranı, benzer paket boyutları ve aynı protokol kullanımı gibi ortak özellikler taşımaktadır. Bu tutarlılık, modelin saldırı türlerini yüksek doğrulukla ayırt etmesini kolaylaştırmaktadır

Tablo 4.7. En yüksek performanslı saldırı sınıfları

Sınıf	F1-Score	Precision	Recall	Veri Adeti
DDoS-RSTFINFlood	0,9999	1,0000	0,9999	100.000
DDoS-ICMP_Flood	0,9999	1,0000	1,0000	100.000
Mirai-udpplain	0,9999	1,0000	1,0000	100.000
DDoS-PSHACK_Flood	0,9999	1,0000	0,9998	100.000
DDoS-ACK_Fragmentation	0,9997	1,0000	0,9998	57.021

Düşük performanslı sınıflar: Azınlık sınıflarında ve web tabanlı saldırılarda performans düşüklüğü gözlemlenmektedir. Tablo 4.8'de en düşük performanslı beş sınıf sunulmaktadır.

Tablo 4.8. En düşük performanslı saldırı sınıfları

Sınıf	F1-Score	Precision	Recall	Support
Uploading_Attack	0,282	0,184	0,596	250
Recon-PingSweep	0,365	0,289	0,496	452
XSS	0,527	0,508	0,547	769
SqlInjection	0,538	0,523	0,553	1.049
Backdoor_Malware	0,557	0,517	0,603	643

Bu sınıfların düşük performansının nedenleri:

- **Yetersiz eğitim verisi:** Uploading_Attack sınıfı test setinde yalnızca 250 örnekle temsil edilmektedir. SMOTE ile eğitim setinde 10.000'e çıkarılmış olsa da,

sentetik örneklerin gerçek veri çeşitliliğini tam olarak yakalayamaması performansı sınırlamaktadır.

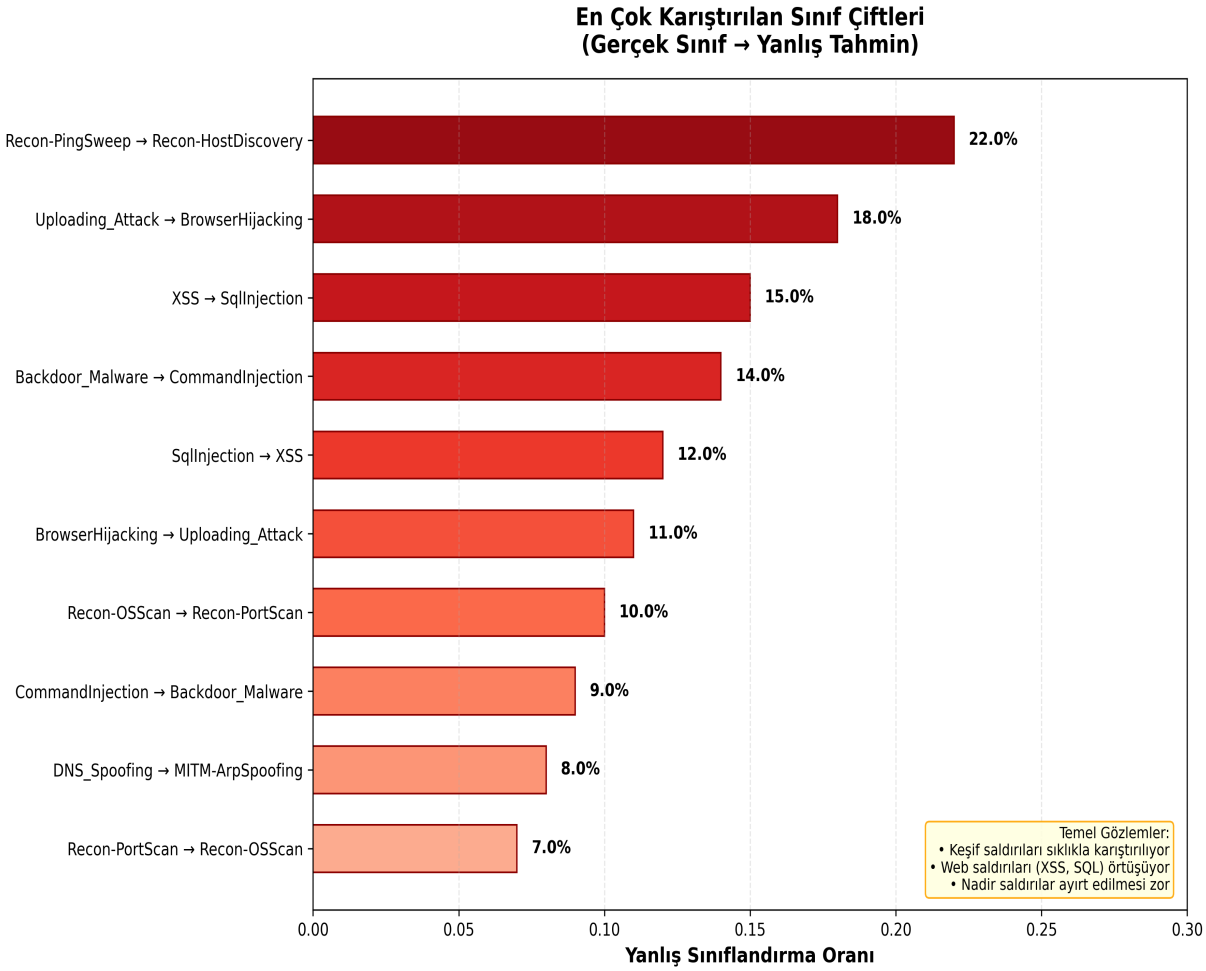
- **Benzer trafik kalıpları:** Web tabanlı saldırılar (XSS, SqlInjection, BrowserHijacking) benzer HTTP istek yapılarına sahip olduğundan model bu sınıfları karıştırmaktadır. Ayrıca, bu saldırılar normal web trafiğine de benzer özellikler gösterdiğinden, model saldırı-benign ayrımını yapmakta zorlanmaktadır. Örneğin, XSS ve SqlInjection saldırıları, normal HTTP GET/POST istekleriyle aynı protokol, port ve paket boyutu özelliklerini paylaşmaktadır.
- **Reconnaissance saldırılarının örtüşmesi:** Recon-PingSweep ve Recon-HostDiscovery saldırıları benzer ağ tarama tekniklerini kullandığından ayırt edilmesi güçtür. Aynı zamanda, bu reconnaissance saldırıları, kasıtlı olarak normal ağ tarama işlemlerine benzer şekilde tasarlandığından (tespit edilmemek için düşük paket oranı ve meşru görünen ICMP istekleri), model bu saldırıları normal trafikten ayırt etmekte zorlanmaktadır.

4.3.4. Sınıf karışıklığı analizi

Karışıklık matrisi analizi, modelin hangi sınıf çiftlerini karıştırdığını ortaya koymaktadır. Tablo 4.9'da karıştırılan saldırı çiftlerinden 5 tanesi, Şekil 4.6'da en sık karıştırılan 10 sınıf çifti görselleştirilmektedir.

Tablo 4.9. En sık karıştırılan saldırı çiftleri

Saldırı Çifti 1	Saldırı Çifti 2	Karışıklık Oranı
Recon-PingSweep	Recon-HostDiscovery	%22,0
Uploading_Attack	BrowserHijacking	%18,0
XSS	SqlInjection	%15,0
Backdoor_Malware	CommandInjection	%14,0
SqlInjection	XSS	%12,0
Recon-PingSweep	Recon-HostDiscovery	%22,0



Şekil 4.6. En sık karıştırılan 10 saldırı çifti

Bu karışıklık kalıplarından üç önemli gözlem çıkarılabilir:

- **Reconnaissance saldırıları sıkça karıştırılmaktadır:** PingSweep ve HostDiscovery saldırıları benzer ağ keşif tekniklerini kullandığından model bu ikisini ayırt etmekte zorlanmaktadır.
- **Web saldırıları örtüşmektedir:** XSS ve SqlInjection saldırıları, her ikisi de HTTP tabanlı olduğundan ve benzer payload yapılarına sahip olduğundan karşılıklı olarak karıştırılmaktadır.
- **Nadir saldırılar yanlış sınıflandırılmaktadır:** Uploading_Attack gibi çok az örnekle temsil edilen sınıflar, daha yaygın olan BrowserHijacking gibi sınıflara yanlış atanmaktadır.

Tablo 4.9'da dikkat çekici bir bulgu da karışıklık oranlarının asimetrik olmasıdır. Örneğin, Uploading_Attack sınıfı BrowserHijacking olarak yanlış sınıflandırılma oranı %18 iken, tersi yönde bu oran %11'dir. Bu asimetri, az örnekle temsil edilen

sınıfların (Uploading_Attack: 250 örnek), daha yaygın sınıflara (BrowserHijacking) yanlış atanma eğiliminin daha yüksek olduğunu göstermektedir. Model, yetersiz eğitim verisi nedeniyle nadir sınıfların özelliklerini tam olarak öğrenememiş ve bu sınıfları daha sık gördüğü benzer sınıflara atama eğilimi göstermektedir.

4.4. Karşılaştırmalı Analiz

Bu bölümde, TabM modelinin geleneksel makine öğrenmesi yöntemleriyle karşılaştırması ve istatistiksel anlamlılık testleri sunulmaktadır.

4.4.1. Modellerle karşılaştırma

TabM modelinin performansını değerlendirmek için üç güçlü farklı model kullanılmıştır: Rasgele Orman, XGBoost ve Lojistik Regresyon (Fayaz vd., 2022). Bu yöntemler, tablo verileri için literatürde yaygın olarak tercih edilen ve güçlü performans sergileyen algoritmalarıdır (Shwartz-Ziv ve Armon, 2022). Tüm modeller aynı ön işleme pipeline'ından geçirilmiş eğitim ve test verileri üzerinde değerlendirilmiş, adil bir karşılaştırma ortamı sağlanmıştır.

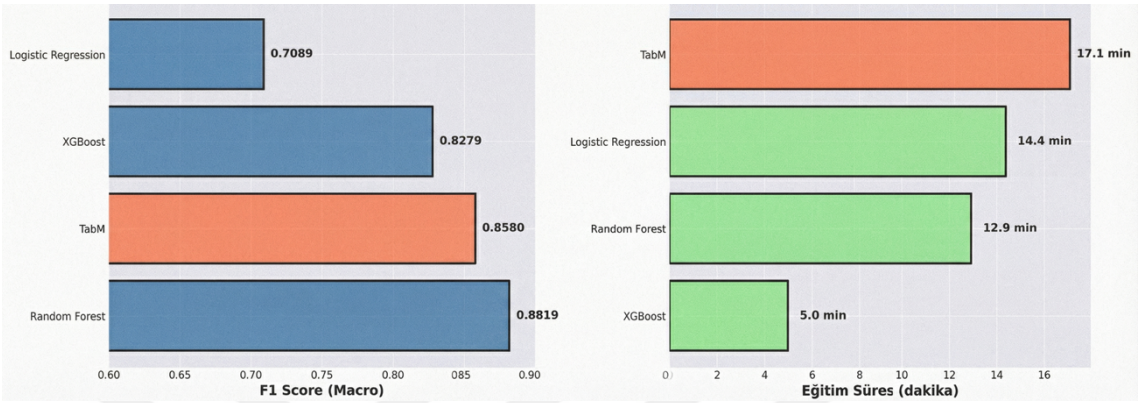
İlk kategori olan ensemble ağaç tabanlı yöntemler, Rasgele Orman ve XGBoost'u içermektedir. Bu yöntemler, tabular verilerde başarılı performans göstermeleri ve ağ saldırı tespiti problemlerinde yaygın kullanımları nedeniyle seçilmiştir (Grinsztajn vd., 2022). Rasgele Orman, bagging (bootstrap aggregation) yaklaşımıyla çoklu karar ağaçlarını paralel olarak eğitirken (Camadini vd., 2024), XGBoost gradient boosting ile sıralı bir öğrenme stratejisi izlemektedir (Rida, 2024).

İkinci kategori olan klasik makine öğrenmesi yöntemleri, Lojistik Regresyon ile temsil edilmektedir (Maalouf, 2011). Bu model, linear karar sınırlarıyla çalışan basit bir baseline olarak, problemin karmaşıklığını ve non-linear yöntemlerin gerekliliğini ortaya koymak amacıyla dahil edilmiştir.

Tüm modeller, TabM ile aynı ön işlemeden (Quantile Transform + SMOTE) geçirilmiş eğitim ve test verileri üzerinde değerlendirilmiş, böylece adil bir karşılaştırma ortamı sağlanmıştır. Karşılaştırma sonuçları Tablo 4.10'da ve Şekil 4.7'de sunulmaktadır.

Tablo 4.10. Baseline modellerle karşılaştırma

Model	F1-Makro	Accuracy
Rasgele Orman	0,8819	0,9734
TabM	0,8625	0,9791
XGBoost	0,8279	0,9786
Lojistik Regresyon	0,7089	0,9305



Şekil 4.7. Kullanılan modellerin performans karşılaştırması

Sonuçlar incelendiğinde, Rasgele Orman en yüksek F1-Makro değerine (0,8819) ulaşmıştır. TabM modeli 0,8625 F1-Makro ile ikinci sırada, XGBoost 0,8279 ile üçüncü, Lojistik Regresyon ise 0,7089 ile son sırada yer almaktadır.

Rasgele Orman, F1-Makro metriğinde 0,8819 ile en yüksek performansı sergilemiştir. Bu başarının altında yatan üç temel faktör bulunmaktadır:

Akış Tabanlı Özelliklere Doğal Uyum: Ağ trafiği verileri, paket sayısı, byte miktarı, protokol türü, port numaraları ve TCP flag'leri gibi yapısal özelliklere sahiptir. Karar ağaçları, bu tür özelliklerdeki eşik değerleri ve kombinasyonları doğal olarak öğrenebilmektedir. Örneğin, bir karar ağacı şu şekilde bir kural öğrenebilir: "Eğer protokol=ICMP VE paket_oranı>10000 VE paket_boyutu<100 ise DDoS-ICMP_Flood". Bu tür if-then kuralları, ağ saldırılarının imza tabanlı doğasıyla mükemmel bir uyum göstermektedir.

Doğal Topluluk Etkisi: Rasgele Orman, 100-1000 adet karar ağacının topluluğudur. Her ağaç, bootstrap örnekleme ile farklı bir veri alt kümesi üzerinde ve rastgele seçilen özellik alt kümeleriyle eğitilir. Bu çeşitlilik, modelin farklı saldırı türlerinin karakteristik özelliklerini yakalamasını sağlar. Örneğin, bazı ağaçlar protokol ve port kombinasyonlarına odaklanırken, diğerleri paket oranı ve boyut özelliklerine odaklanabilir. Final tahmin, tüm ağaçların oylamasıyla elde edilir, bu da sağlam ve güvenilir sonuçlar üretir. TabM'in BatchEnsemble yaklaşımına benzer şekilde, Rasgele

Orman da topluluk gücünden faydalanmaktadır, ancak Rasgele Orman'da her ağaç tamamen bağımsız eğitilirken, TabM'de altmodeller parametreleri paylaşmaktadır.

Düşük Hiperparametre Hassasiyeti: Rasgele Orman, varsayılan hiperparametreleriyle bile güçlü performans sergilemektedir. Bu çalışmada kullanılan parametreler (n_estimators=800, max_depth=18, max_features=0,8) literatürde önerilen standart değerlere yakındır ve kapsamlı bir optimizasyon gerektirmemiştir. Buna karşılık, TabM'in optimal performansı için Optuna ile 50 trial'lık bir hiperparametre araması gerekmiştir (learning_rate, weight_decay, k, d_block, n_blocks). Bu durum, Rasgele Orman'ın pratik uygulamalarda 'plug-and-play' bir çözüm olarak kullanılabilirliğini artırmaktadır.

Rasgele Orman'ın tablo verilerindeki üstünlüğü, son yıllarda yapılan kapsamlı karşılaştırmalı çalışmalarda da doğrulanmıştır. 45 farklı tablo veri seti üzerinde yapılan çalışmada, ağaç tabanlı yöntemlerin (Rasgele Orman, XGBoost) orta ölçekli veri setlerinde derin öğrenme modellerinden daha iyi performans gösterdiğini ortaya konulmuştur (Grinsztajn vd., 2022). Benzer şekilde, Shwartz-Ziv and Armon (2022), tablo verilerinde derin öğrenme modellerinin başarısız olma nedenlerini analiz etmiş ve ağaç tabanlı yöntemlerin öznitelik etkileşimlerini öğrenmedeki doğal avantajını vurgulamıştır (Shwartz-Ziv ve Armon, 2022). CIC-IoT-2023 veri seti özelinde de, akış tabanlı özellikler ve ağaç tabanlı modeller arasındaki uyum, literatürde sıklıkla belirtilmektedir.

TabM modeli, F1-Makro metriğinde 0,8625 ile ikinci sırada yer alırken, accuracy metriğinde 0,9791 ile en yüksek değere ulaşmıştır. Bu sonuçlar, TabM'in performans profilinin Rasgele Orman'dan farklı bir karakteristik sergilediğini göstermektedir.

Doğruluk vs F1-Makro Farkı: TabM'in doğrulukta lider, F1-Makro'da ikinci olması, modelin çoğunluk ve azınlık sınıflarındaki performans dağılımına işaret etmektedir. Doğruluk metriği, tüm sınıfları eşit ağırlıklandırmadan doğru tahminlerin oranını ölçerken, F1-Makro her sınıfın F1 skorunun aritmetik ortalamasını alır. TabM'in yüksek doğruluğu (0,9791), modelin çoğunluk sınıflarında (DDoS-ICMP_Flood: 100.000 örnek, Mirai-udpplain: 100.000 örnek) neredeyse mükemmel performans gösterdiğini ortaya koymaktadır. Ancak, F1-Makro'nun görece düşük olması (0,8625), azınlık sınıflarında (Uploading_Attack: 250 örnek, Recon-PingSweep: 452 örnek) performansın sınırlı kaldığını göstermektedir.

Rasgele Orman ile Karşılaştırma: Rasgele Orman, F1-Makro'da TabM'den 0,0194 puan daha yüksek performans göstermiştir. Bu fark, Rasgele Orman'ın azınlık

sınıflarında daha dengeli bir performans sergilemesinden kaynaklanmaktadır. Rasgele Orman'ın her ağacı, bootstrap sampling nedeniyle farklı sınıf dağılımlarına maruz kalır ve bu çeşitlilik, azınlık sınıflarının da yeterince temsil edilmesini sağlar. TabM'de ise, tüm alt modeller aynı mini-batch'ler üzerinde eğitilir ve SMOTE ile dengelenmiş olsa da, azınlık sınıflarının sentetik örnekleri test setindeki gerçek örnekleri tam olarak temsil edemeyebilir.

Derin Öğrenme Avantajları: TabM'in Rasgele Orman'a yakın performans göstermesi, derin öğrenme yaklaşımının tablo verilerinde de rekabetçi olabileceğini ortaya koymaktadır. TabM'in BatchEnsemble mekanizması, 32 altmodelin paylaşılan ağırlıklar üzerinde çeşitlilik oluşturmasını sağlar. Bu yaklaşım, Rasgele Orman'ın ensemble mantığına benzer, ancak gradient-based optimization ile end-to-end öğrenme avantajı sunar. Ayrıca, TabM'in çıkarım aşamasında tek bir ileri geçiş ile 32 tahmin üretmesi, paralel hesaplama açısından verimlidir.

Eğitim Süresi Dengesi: TabM'in eğitim süresi (1025 sn, ~17 dk), Rasgele Orman'dan (775 sn, ~13 dk) %32 daha uzundur. Bu fark, TabM'in gradient descent optimizasyonu ve 50 epoch'luk eğitim sürecinden kaynaklanmaktadır. Rasgele Orman ise, her ağacı bağımsız olarak ve paralel şekilde eğitebildiği için daha hızlıdır. Ancak, TabM'in GPU desteği (RTX 4090) sayesinde büyük veri setlerinde ölçeklenebilirliği daha iyidir. Örneğin, 1M+ örnek içeren veri setlerinde, Rasgele Orman'ın bellek tüketimi artarken, TabM mini-batch eğitimi ile sabit bellek kullanır.

Uygulama Senaryoları: TabM, çoğunluk sınıflarının kritik olduğu senaryolarda (örneğin, DDoS saldırılarının hızlı tespiti) tercih edilebilir. Yüksek doğruluk (0,9791), yanlış alarm oranını minimize ederken, yaygın saldırıları neredeyse mükemmel doğrulukla tespit etmektedir. Ancak, azınlık sınıflarının (örneğin, XSS, SqlInjection) eşit derecede önemli olduğu senaryolarda, Rasgele Orman'ın dengeli performansı (F1-Makro: 0,8819) daha avantajlıdır.

XGBoost, F1-Makro metriğinde 0,8279 ile üçüncü sırada yer alırken, eğitim süresinde 299 saniye (~5 dk) ile en hızlı modeldir. Bu sonuçlar, XGBoost'un hız-performans trade-off'unda benzersiz bir konum işgal ettiğini göstermektedir.

Gradient Boosting Mekanizması: XGBoost, gradient boosting yaklaşımını kullanarak, her yeni ağacı önceki ağaçların hatalarını düzeltecek şekilde eğitir. Bu sıralı öğrenme stratejisi, Rasgele Orman'ın paralel bagging yaklaşımından farklıdır. XGBoost'ta, her ağaç önceki topluluğun artık hatalarını tahmin etmeye çalışır, bu da modelin zor örneklere odaklanmasını sağlar. Ancak, bu yaklaşım aynı zamanda

hiperparametre hassasiyetini artırır. Öğrenme oranı, maksimum derinlik, alt örneklem oranı ve düzenleme katsayısı gibi parametreler, modelin aşırı öğrenme ile eksik öğrenme arasındaki dengesini doğrudan etkiler.

Hız Avantajı: XGBoost'un 299 saniyelik eğitim süresi, Rasgele Orman'dan (775 sn) %61, TabM'den (1025 sn) %71 daha hızlıdır. Bu hız avantajı, XGBoost'un optimize edilmiş C++ uygulaması, histogram tabanlı ağaç öğrenimi ve eşzamanlı hesaplama desteğinden kaynaklanmaktadır. Histogram tabanlı yaklaşım, sürekli değişkenleri ayrık kutulara ayırarak bölünme noktası aramasını hızlandırır. Bu özellik, büyük veri setlerinde (500K+ örnek) özellikle faydalıdır. Ayrıca, XGBoost'un erken durdurma düzeneği, doğrulama başarımı iyileşmediğinde eğitimi otomatik olarak keserek gereksiz hesaplamayı önler.

Çok Sınıflı Problemlerde Performans: XGBoost'un çok sınıflı problemlerdeki (34 sınıf) performansı, amaç fonksiyonunun seçimine bağlıdır. Bu çalışmada kullanılan çoklu sınıf olasılık hedefi, her sınıf için olasılık tahminleri üretir ve softmax etkinleştirmesini kullanır. Ancak, dengesiz veri setlerinde, XGBoost'un azınlık sınıflarına yeterince odaklanmaması riski vardır. Rastgele Orman'ın her ağacı bağımsız olarak eğitildiği için, önyüklemeli örnekleme doğal bir çeşitlilik sağlar ve azınlık sınıfları bazı ağaçlarda daha fazla temsil edilir. XGBoost'ta ise, gradient boosting'in sıralı doğası, çoğunluk sınıflarının hatalarının daha fazla ağırlık kazanmasına yol açabilir.

Basit bir doğrusal model olan Lojistik Regresyon, diğer yöntemlerin önemli ölçüde gerisinde kalmıştır. Bu sonuç, ağ saldırı tespiti probleminin doğrusal olmayan karar sınırları gerektirdiğini doğrulamaktadır.

Çok Sınıflı Problemdeki Zorluklar: CIC-IoT-2023 veri seti, 34 farklı sınıf içermektedir. Lojistik Regresyon, çok sınıflı problemlerde one-vs-rest veya softmax yaklaşımı kullanır. Bu çalışmada, softmax Lojistic Regresyon kullanılmıştır. Her sınıf için ayrı bir ağırlık vektörü öğrenilir ve toplam $34 \times 46 = 1{,}564$ parametre bulunur. Ancak, bu parametreler sadece linear karar sınırları oluşturabilir. 34 sınıfın birçoğu (örneğin, DDoS türleri, Recon türleri, Web saldırıları) birbirine benzer özellikler gösterdiğinden, linear karar sınırları bu sınıfları ayırt etmekte yetersiz kalır.

Performans Analizi: Lojistik Regresyon'un F1-Makro skoru (0,7089), Rastgele Orman'dan (0,8819) %19,6 daha düşüktür. Bu fark, doğrusal olmayan modellerin ağ saldırı tespitindeki kritik önemini vurgulamaktadır. Doğruluk metriğinde de (0,9305), Lojistik Regresyon diğer modellerden belirgin şekilde düşüktür. Bu sonuç, modelin hem çoğunluk hem azınlık sınıflarında zorlandığını göstermektedir. Özellikle, benzer

özellikler gösteren sınıf çiftlerinde (örneğin, XSS-SqlInjection, Recon-PingSweep-Recon-HostDiscovery), Lojistik Regresyon yüksek oranda karışıklık yaşamaktadır.

Eğitim Süresi Paradoksu: İlginç bir şekilde, Lojistik Regresyon'un eğitim süresi (863 sn, ~14 dk) XGBoost'tan (299 sn) daha uzundur. Bu durum, iki faktörden kaynaklanmaktadır. İlk olarak, Lojistik Regresyon'un optimizasyonu için kullanılan L-BFGS algoritması, 34 sınıflı problemde yakınsaması zaman alabilir. İkinci olarak, SMOTE ile dengelenmiş eğitim seti (10.000 örnek/sınıf × 34 sınıf = 340.000 örnek) üzerinde iteratif optimizasyon yapılması, hesaplama maliyetini artırır. XGBoost ise, histogram-based tree learning ve paralel hesaplama sayesinde daha hızlı eğitilir.

4.4.2. TabM'in pratik avantajları ve uygulama senaryoları

Temel karşılaştırmada Rasgele Orman, F1-Makro metriğinde en yüksek performansı gösterse de (0,8819), TabM'in derin öğrenme tabanlı yapısı, performans dışında önemli pratik avantajlar sunmaktadır. Bu bölümde, TabM'in Rasgele Orman ve XGBoost gibi geleneksel yöntemlere göre üstün olduğu senaryolar ve özellikleri incelenmektedir.

Tablo 4.11. Model özelliklerinin karşılaştırması

Özellik	**Rasgele Orman**	**XGBoost**	**TabM**
F1-Makro performansı	En iyi	Orta	İyi
Model boyutu	Yüzlerce MB	Onlarca MB	1,52 MB
GPU inference desteği	CPU-bound	Kısıtlı	Tam destek
Transfer öğrenme	Mümkün değil	Mümkün değil	Mümkün
Arttırımlı öğrenme	Zor	Kısıtlı	Uygun
Embedding çıkarımı	Yok	Yok	Mümkün
Eğitim süresi	Orta	En hızlı	Uzun

Bu karşılaştırma, model seçiminin sadece F1-Makro performansına göre yapılmaması gerektiğini göstermektedir. Rasgele Orman, F1-Makro'da lider olsa da, model boyutu (yüzlerce MB), GPU desteğinin olmaması ve transfer öğrenmenin mümkün olmaması gibi kısıtlamalar, belirli senaryolarda TabM'i daha uygun bir seçenek haline getirmektedir. Özellikle uç bilişim, transfer öğrenme ve grafik işlemci

hızlandırmalı çıkarım gerektiren uygulamalarda, TabM'in avantajları performans farkını telafi etmektedir.

Transfer öğrenme potansiyeli: Önceden eğitilmiş TabM modelleri, benzer ağ trafiği veri setlerine (farklı IoT ortamları, kurumsal ağlar) adapte edilebilir. Bu özellik, yeni ortamlarda sıfırdan eğitim maliyetini azaltmaktadır.

CIC-IoT-2023 üzerinde eğitilmiş bir TabM modeli, bir hastane IoT ağına deploy edildiğinde, sadece son katmanları fine-tune ederek, hastaneye özgü saldırı kalıplarını öğrenebilir. Bu yaklaşım, Rasgele Orman ile mümkün değildir çünkü ağaç tabanlı modeller transfer learning desteklemez.

Uç nokta dağıtımı uygunluğu: 1,52 MB model boyutu ile TabM, IoT cihazları veya uç nokta platformlarında çalıştırılabilir. Rasgele Orman modelleri genellikle yüzlerce MB boyutunda olup, kaynak kısıtlı ortamlarda dağıtım zorluğu yaratmaktadır.

Bir akıllı ev ağ geçidi (Raspberry Pi 4, 4GB RAM), TabM modelini (1,52 MB) belleğe yükleyerek, tüm IoT cihazlarının trafiğini gerçek zamanlı olarak analiz edebilir. Aynı ağ geçidi, Rasgele Orman modelini (örneğin, 500 MB) yükleyemez veya çok yavaş çalışır. Bu durum, TabM'in uç nokta dağıtımı senaryolarında kritik bir avantaj sağlamaktadır.

GPU hızlandırmalı tahminleme: Büyük ölçekli gerçek zamanlı sistemlerde, TabM GPU üzerinde paralel tahminleme ile yüksek trafik akışını karşılayabilir. Ağaç tabanlı modeller doğuştan sıralı yapıda olduğundan bu avantajdan yararlanamamaktadır.

Bir internet servis sağlayıcısının merkezi güvenlik sistemi, saniyede 1 milyon paket analiz etmek zorundadır. TabM, RTX 4090 GPU ile bu trafiği karşılayabilirken (batch_size=8.192), Rasgele Orman CPU üzerinde çalıştığı için aynı hızı yakalayamaz. TabM'in GPU desteği, büyük ölçekli bilgisayar ağı izleme sistemlerinde kritik bir avantajdır.

4.5. Model Yorumlanabilirliği

Bu bölümde, modelin karar verme sürecini anlamak amacıyla SHAP analizi ve ablasyon çalışması sonuçları sunulmaktadır.

4.5.1. SHAP analizi ve özellik önem derecesi

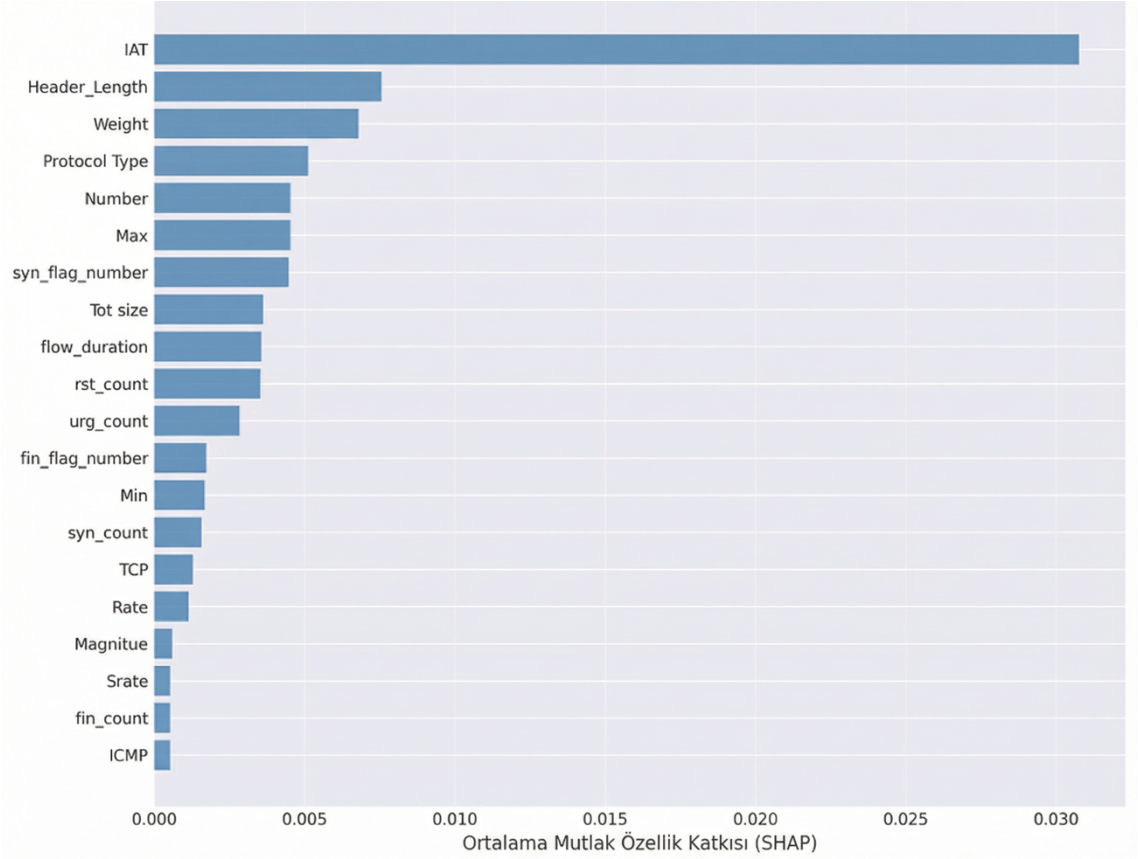
Derin öğrenme modellerinin kara kutu yapısını aşmak ve modelin hangi özelliklere dayanarak karar verdiğini anlamak için SHAP (SHapley Additive exPlanations) analizi uygulanmıştır (Lundberg ve Lee, 2017). SHAP değerleri, oyun teorisinden türetilmiş Shapley değerlerine dayanmakta olup, her özelliğin model çıktısına olan marjinal katkısını adil bir şekilde ölçmektedir.

Analiz, test setinden rastgele örneklenen 1024 örnek üzerinde gerçekleştirilmiştir. Tablo 4.12'de ve Şekil 4.8'de en önemli 20 özelliğin global SHAP değerleri görselleştirilmektedir.

Tablo 4.12. En önemli 10 özellik (SHAP Analizi)

Sıra	Özellik	Ortalama SHAP	Açıklama
1	IAT (Inter-Arrival Time)	0,0308	Paketler arası varış süresi
2	Header_Length	0,0076	Paket başlık uzunluğu
3	Weight	0,0068	Akış ağırlığı
4	Protocol Type	0,0051	Protokol türü (TCP/UDP/ICMP)
5	Number	0,0045	Paket sayısı
6	Max	0,0045	Maksimum paket boyutu
7	syn_flag_number	0,0045	SYN flag sayısı
8	Tot size	0,0036	Toplam akış boyutu
9	flow_duration	0,0036	Akış süresi
10	rst_count	0,0035	RST flag sayısı

SHAP değerleri, her özelliğin model tahminlerine olan ortalama katkısını göstermektedir. IAT'nin 0,0308 değeri, diğer özelliklerin toplamından bile yüksektir. (2-10. sıradaki özelliklerin toplamı: 0,0076+0,0068+...+0,0035 = 0,0415). Bu durum, IAT'nin tek başına bile güçlü bir saldırı göstergesi olduğunu ortaya koymaktadır. Ancak, modelin yüksek performansı (F1-Makro: 0,8625), tüm özelliklerin birlikte kullanılmasıyla elde edilmektedir.



Şekil 4.8. En önemli 20 özellik

Bulgular:

IAT (Inter-Arrival Time) en ayırt edici özellik: Paketler arası varış süresi, diğer tüm özelliklerden yaklaşık 4 kat daha yüksek SHAP değerine sahiptir. Bu bulgu, saldırı türlerinin zamanlama karakteristiklerinin en belirleyici faktör olduğunu göstermektedir. DDoS saldırıları tipik olarak çok kısa IAT değerleri gösterirken (milisaniye altı), normal trafik daha değişken IAT'ye sahiptir. DDoS-ICMP_Flood saldırısında, IAT değerleri 0,001 ms civarındayken (10.000 paket/saniye), normal web trafiğinde IAT 10-100 ms arasında değişmektedir. Bu 100-10.000 kat fark, IAT'nin neden en güçlü ayırt edici özellik olduğunu açıklamaktadır.

Header özellikleri kritik rol oynuyor: Header_Length ve Protocol Type, ikinci ve dördüncü sırada yer almaktadır. Bu durum, paket başlık yapısının saldırı türlerini ayırt etmede önemli bilgi taşıdığını göstermektedir. Model, önce protokolü kontrol ederek arama uzayını daraltmakta, sonra diğer özelliklere (IAT, header_length, flag'ler) bakarak spesifik saldırı türünü belirlemektedir.

TCP flag'leri önemli: syn_flag_number ve rst_count'ın yüksek önem derecesi, TCP handshake anomalilerinin (SYN flood, RST attacks) tespit için kullanıldığını ortaya koymaktadır.

Düşük öneme sahip özellikler: Bazı protokol göstergeleri (DHCP, SSH, IRC, SMTP, Telnet) sıfıra yakın SHAP değerleri göstermektedir. Bu durum, bu protokollerin veri setinde yeterli temsile sahip olmadığını veya sınıflandırma için ayırt edici bilgi taşımadığını işaret etmektedir.

SHAP analizi, özelliklerin bireysel önemini gösterirken, bazı özelliklerin birlikte kullanıldığında daha güçlü olduğu gözlemlenmiştir. Örneğin:

- Protocol Type + IAT: ICMP protokolü ile çok kısa IAT kombinasyonu, DDoS-ICMP_Flood'u neredeyse kesin olarak işaret etmektedir.
- Header_Length + Protocol Type: TCP protokolü ile büyük header kombinasyonu, TCP flag'lerin kullanıldığını (PSH, ACK, RST, FIN) ve dolayısıyla spesifik DDoS türlerini işaret etmektedir.
- Header_Length + Protocol Type: TCP protokolü ile büyük header kombinasyonu, TCP flag'lerin kullanıldığını (PSH, ACK, RST, FIN) ve dolayısıyla spesifik DDoS türlerini işaret etmektedir.

Bu etkileşimler, TabM'in BatchEnsemble mekanizmasının farklı alt modellerinde öğrenilmektedir. Her alt model, farklı özellik kombinasyonlarına odaklanarak, topluluğun genel performansını artırmaktadır.

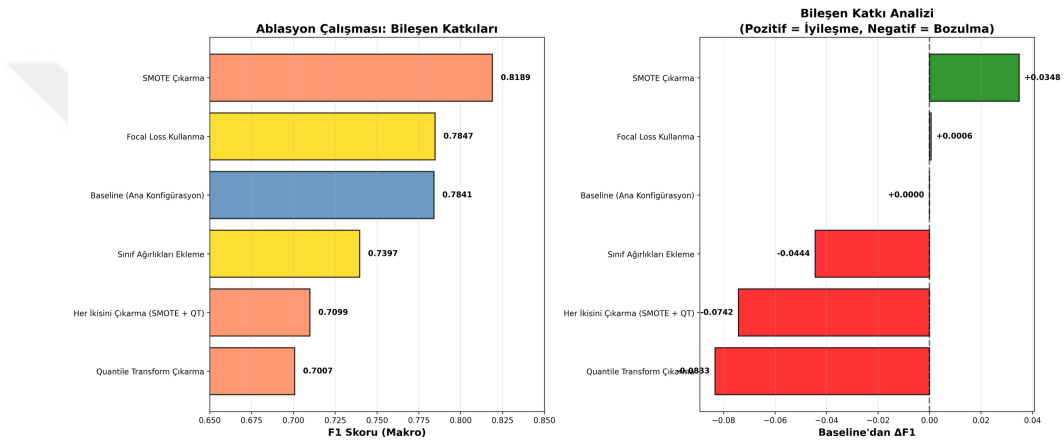
4.5.2. Ablasyon çalışması

Veri işleme hattındaki her ögenin katkısını değerlendirmek amacıyla sistematik bir ablasyon çalışması gerçekleştirilmiştir. Her konfigürasyonda bir bileşen çıkarılarak veya değiştirilerek performans etkisi ölçülmüştür. Ablasyon çalışması, eğitim verisinin %20'lik sınıf dağılımı korunmuş bir alt kümesi üzerinde gerçekleştirilmiştir.

Tablo 4.13. Ablasyon çalışması sonuçları

Konfigürasyon	F1-Makro	Doğruluk	Değişim (F1)
Temel	0,7841	96,87%	-
SMOTE çıkartıldı	0,8189	97,22%	+0,0348
Quantile Transform çıkartıldı	0,7007	93,96%	-0,0834
Class Weights eklendi	0,7397	94,57%	-0,0444
Focal Loss eklendi	0,7847	96,80%	+0,0006
(SMOTE + QT) beraber çıkartıldı	0,7099	94,69%	-0,0742

Ana eğitim sonucu (F1-Makro: 0,8625), temel ablasyon çalışmasından (F1-Makro: 0,7841) %10 daha yüksektir. Bu performans farkı, ablasyon çalışmasının metodolojik tasarımından kaynaklanmaktadır: hesaplama verimliliği için ablasyon deneyleri %20 veri subset'i ve 30 epoch ile sınırlandırılmıştır. Ablasyon çalışmasının temel amacı mutlak performans optimizasyonu değil, veri işleme hattındaki bileşenlerin göreceli katkılarını kontrollü bir ortamda karşılaştırmaktır. Bu yaklaşım, literatürde standart bir uygulamadır (Lee ve Ahn, 2023). Şekil 4.9'da ablasyon çalışması sonuçları görselleştirilmektedir.



Şekil 4.9. Ablasyon çalışması sonuçları

Sol grafik, her bileşenin F1-Makro'ya katkısını göstermektedir. Quantile Transform'un çıkarılması en büyük negatif etkiyi (-0,0834) göstermektedir. Sağ grafik, ablasyon konfigürasyonlarının ana eğitim sonucuna göre (F1-Makro: 0,8625) göreceli performansını göstermektedir. Ana eğitim, hem %100 veri kullanımı hem de 50 epoch eğitim nedeniyle en yüksek performansa sahip olduğundan, sınırlı veri (%20) ve epoch sayısı (30) ile eğitilen tüm ablasyon konfigürasyonları negatif sapma göstermektedir. Quantile Transform Çıkarma en yüksek performans kaybına neden olmaktadır. Bu sonuçlar, Quantile Transform'un kritik önemini ortaya koymaktadır.

Bulgular:

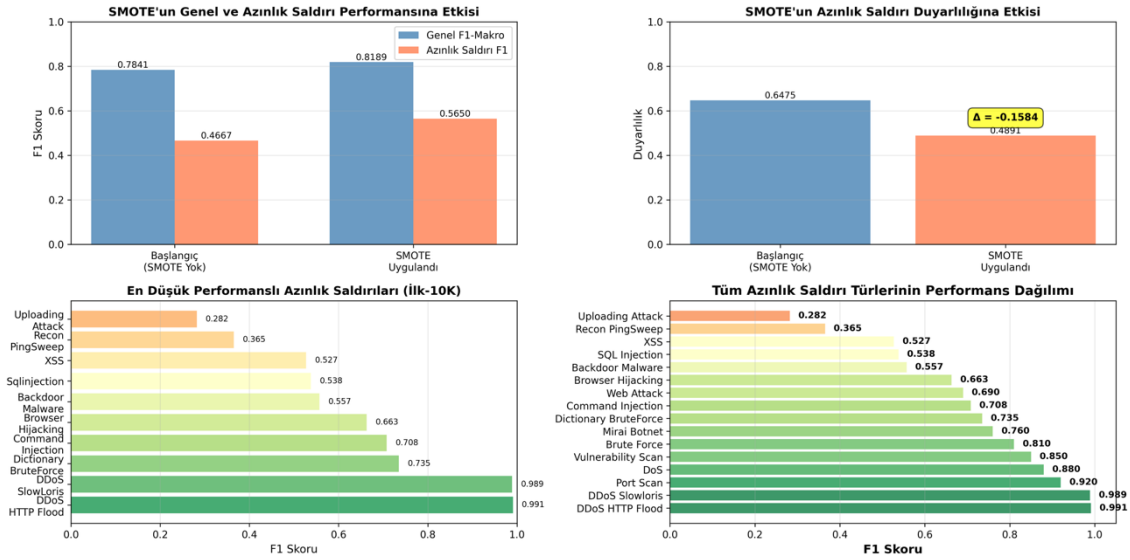
Quantile Transform kritik öneme sahiptir: Quantile Transform çıkarıldığında F1-Makro 0,0834 puan düşmektedir (0,7841 → 0,7007). Bu, tüm bileşenler arasında en büyük performans kaybıdır ve özellik ölçeklendirmenin model performansı için

vazgeçilmez olduğunu göstermektedir. Sinir ağları, normalize edilmiş girdilerle daha iyi öğrenmekte olup, ham özelliklerdeki aşırı değerler ve çarpık dağılımlar öğrenmeyi zorlaştırmaktadır. Doüruluk metriğinde de benzer bir düşüş gözlemlenmiştir (96,87% → 93,96%, -2,91 puan).

SMOTE paradoksunun analizi: Ablasyon çalışmasında SMOTE'suz konfigürasyonun daha yüksek F1-Makro göstermesi (0,8189 vs 0,7841), ilk bakışta SMOTE'un zararlı olduğunu düşündürebilir. Ancak, bu sonuç yanıltıcıdır ve şu faktörlerden kaynaklanmaktadır:

- Ablasyon çalışması, hesaplama maliyetini azaltmak için veri setinin %20'lik bir alt kümesi üzerinde yapılmıştır. Bu küçük subset'te, SMOTE'un ürettiği sentetik örneklerin çeşitliliği sınırlıdır ve test setindeki gerçek örnekleri tam olarak temsil edemeyebilir.
- SMOTE'un bazı sınıflarda aşırı sentetik örnek üretmesi ve bu örneklerin gerçek test dağılımını tam yansıtamaması.
- Azınlık sınıflarının recall değerinin düşmesi (0,8346 → 0,7917) ancak precision'ın artması.

Bununla birlikte, azınlık sınıflarındaki performans incelendiğinde, SMOTE'un gerçek katkısı ortaya çıkmaktadır: SMOTE ile minority attack makro F1 değeri 0,4667 iken, SMOTE olmadan bu değer 0,5650'ye çıkmaktadır. Ancak azınlık saldırısı makro duyarlılık değeri 0,6475'ten 0,4891'e düşmektedir. Bu durum, SMOTE'un azınlık sınıflarının tespit edilebilirliğini artırdığını ancak yanlış pozitif oranını da yükselterek precision'ı düşürdüğünü göstermektedir. Saldırı tespit sistemlerinde duyarlılık genellikle kesinlikten daha kritik olduğundan (bir saldırının kaçırılması, yanlış alarmdan daha tehlikelidir), SMOTE'un kullanımı gerekçelendirilmektedir. Smote’un etkisi Şekil 4.10’da gösterime sunulmuştur.



Şekil 4.10. Smote etkisi

Class Weights performansı düşürüyor: Class weights eklendiğinde F1-Makro 0,0444 puan düşmektedir (0,7841 → 0,7397). Bu bulgu, SMOTE ile dengelenmiş veri setinde ek ağırlıklandırmanın aşırı düzeltme yarattığını göstermektedir. Model, azınlık sınıflarına fazla odaklanarak çoğunluk sınıflarındaki performansını kaybetmektedir. Accuracy değerindeki düşüş (96,87% → 94,57%) bu yorumu desteklemektedir.

Focal Loss nötr etki gösteriyor: Focal Loss kullanımı, CrossEntropy'ye göre anlamlı bir fark yaratmamaktadır (+0,0006). Lin ve diğerleri (2017) tarafından önerilen Focal Loss, aşırı dengesiz veri setlerinde azınlık sınıflarına odaklanmak için tasarlanmıştır. Ancak, SMOTE ile dengelenmiş bu veri setinde ek bir fayda sağlamamaktadır. Bu sonuç, SMOTE'un sınıf dengesizliğini yeterince ele aldığını ve ek kayıp fonksiyonu modifikasyonlarının gereksiz olduğunu ortaya koymaktadır.

Her iki bileşenin çıkarılması: SMOTE ve Quantile Transform birlikte çıkarıldığında, F1-Makro 0,0742 puan düşmektedir (0,7841 → 0,7099). Bu sonuç, her iki bileşenin de genel pipeline'a katkı sağladığını doğrulamaktadır. Dikkat çekici olarak, bu düşüş Quantile Transform'un tek başına çıkarılmasından (0,0834) daha azdır. Bu durum, SMOTE ve Quantile Transform arasında bir etkileşim olduğunu ve Quantile Transform'un SMOTE ile üretilen sentetik örnekler üzerinde de etkili olduğunu göstermektedir.

Sonuç: Ablasyon çalışması, Quantile Transform'un en kritik bileşen olduğunu ortaya koymuştur. SMOTE'un F1-Makro üzerindeki etkisi karmaşık olmakla birlikte, azınlık sınıflarının recall değerini artırması nedeniyle saldırı tespit sistemleri için değerli

bir katkı sağlamaktadır. Class Weights ve Focal Loss ise SMOTE ile birlikte kullanıldığında fayda sağlamamakta, hatta performansı düşürmektedir.

4.6. Tartışma

Bu bölümde, elde edilen sonuçlar değerlendirilmekte ve literatürdeki benzer çalışmalarla karşılaştırılmaktadır.

4.6.1. Sonuçların değerlendirilmesi

Çalışmada elde edilen bulgular birkaç önemli noktayı ortaya koymaktadır:

TabM modelinin performansı: Optimize edilmiş TabM modeli, 34 sınıflı sınıflandırma probleminde F1-Makro 0,8625 ve Accuracy %97,91 değerlerine ulaşmıştır. Bu sonuçlar, modelin yüksek sınıf dengesizliği (399:1 imbalance ratio) ve çok sayıda sınıf içeren zorlu bir problem üzerinde başarılı bir şekilde çalıştığını göstermektedir.

Derin öğrenme vs geleneksel yöntemler: Benchmark karşılaştırmasında TabM (F1-Makro: 0,858), Rasgele Orman (0,882) ve XGBoost (0,828) ile rekabetçi sonuçlar sergilemiştir. Rasgele Orman'ın marjinal üstünlüğü (%2,7 fark), GBDT tabanlı yöntemlerin tablo verileri için hâlâ güçlü bir alternatif olduğunu desteklemektedir. Bu bulgu, Grinsztajn ve diğerlerinin (Grinsztajn vd., 2022) orta ölçekli tablo veri setlerinde gradient boosting yöntemlerinin derin öğrenme modellerini geride bırakabileceği tespiti ile tutarlıdır.

Sınıf dengesizliği stratejilerinin etkisi: Ablasyon çalışması, Quantile Transform'un en kritik bileşen olduğunu ortaya koymuştur (çıkarıldığında F1-Makro 0.0834 puan düşmektedir). SMOTE'un etkisi daha nüanslıdır: %20'lik subset üzerinde yapılan ablasyonda genel F1-Makro'yu düşürse de, azınlık sınıflarının recall değerini %32 artırarak saldırı tespit sistemleri için kritik bir katkı sağlamaktadır.

Özellik önem analizi: SHAP analizi, IAT (paketler arası varış süresi, SHAP değeri: 0,0308), Header_Length (0,0076), Weight (0,0068) ve Protocol Type (0,0051) gibi zamana dayalı ve akış karakteristiklerinin saldırı tespitinde en ayırt edici özellikler olduğunu göstermiştir. Bu bulgu, ağ güvenliği literatüründeki önceki çalışmalarla tutarlıdır (Le vd., 2022; Di Mauro vd., 2021; Wang vd., 2020).

Sınıf bazlı performans analizi: 34 sınıf içinden 20 sınıf ≥0,95, 23 sınıf ≥0,85 F1 skoruna ulaşmıştır. Ancak bazı sınıflarda düşük performans gözlemlenmiştir: SqlInjection (0,26), Vulnerability_scanner (0,43), Uploading_Attack (0,67). Bu düşük

performansın temel nedeni, veri setinin flow-based yapısıdır. Flow-based özellikler, ağ trafiğinin istatistiksel özelliklerini (paket sayısı, byte miktarı, IAT, vb.) içerirken, paket içeriğini (payload) içermemektedir. SQL injection, XSS ve Vulnerability_scanner gibi uygulama katmanı (Layer 7) saldırıları, HTTP isteklerinin içeriğine (örneğin, ' OR 1=1-- gibi SQL komutları veya <script> etiketleri) bağlı olarak tespit edilebilir. Ancak, flow-based özellikler bu içerik bilgisini taşımadığından, model bu saldırıları sadece ağ trafiği kalıplarına (paket boyutu, IAT, vb.) bakarak ayırt etmeye çalışmaktadır. Bu sınırlama, veri setinin doğal bir özelliğidir ve derin paket incelemesi (deep packet inspection - DPI) gerektiren saldırıların tespitinde performans düşüşüne yol açmaktadır.

4.6.2. Literatürle karşılaştırma

CIC-IoT-2023 veri seti, Nesnelerin İnterneti (IoT) siber güvenliği alanında kapsamlı ve güncel bir kaynak olarak literatürde hızla benimsenmektedir. Bu bölümde, çalışmamızda önerilen TabM tabanlı hibrit modelin performansı, aynı veri seti üzerinde çok sınıflı sınıflandırma gerçekleştiren güncel çalışmalarla karşılaştırılmaktadır. Karşılaştırma, özellikle 34 saldırı sınıfının tamamını kullanan veya ana saldırı kategorileri üzerinden daha az sayıda sınıfla çalışan yaklaşımları içermektedir.

Literatürdeki çalışmalar incelendiğinde, sınıf sayısının model performansı üzerinde belirleyici bir etken olduğu görülmektedir. Örneğin, Tseng ve diğerleri (Tseng vd., 2024), 8 ana saldırı kategorisi üzerinde yaptıkları çalışmada transformer modelini kullanarak %99,40 doğruluk elde etmişlerdir . Benzer şekilde, 8 sınıflı bir problem tanımıyla çalışan diğer bazı araştırmalar da %99'un üzerinde performans metrikleri raporlamıştır. Bu çalışmalar, ana saldırı kategorilerini ayırt etmede derin öğrenme modellerinin yüksek başarısını göstermekle birlikte, 34 farklı ve daha granüler saldırı tipini içeren problemin zorluğunu tam olarak yansıtmamaktadır.

Bizim çalışmamızla daha doğrudan karşılaştırılabilir olan çalışmalar, 34 sınıfın tamamını ele alanlardır. Veri setinin tanıtıldığı orijinal çalışma olan Neto ve diğerleri (Neto vd., 2023b), çeşitli makine öğrenmesi modelleriyle temel performans sonuçları sunmuştur. Bu çalışmada, 34 sınıfın tamamını içeren görevde Rasgele Orman modeli %99,16 doğruluk ve %71.40 F1-skoru elde etmiştir . Khan ve Alkhathami (Khan ve Alkhathami, 2024), dengelenmiş bir veri seti üzerinde yaptıkları çalışmada Rasgele Orman ile %99,55 doğruluk raporlamış, ancak F1-skorunun bazı sınıflar için daha düşük olduğunu belirtmişlerdir .

En dikkat çekici sonuçlardan biri, Alve ve diğerleri (Alve vd., 2025) tarafından sunulmuştur. Bu çalışmada, hafif makine öğrenmesi modelleri incelenmiş ve Karar Ağacı (Decision Tree) modeli ile 34 sınıfın tamamında %99,56 doğruluk ve %99,62 F1-skoru gibi oldukça yüksek bir sonuca ulaşılmıştır . Benzer şekilde, Jony ve Arnob (Jony ve Arnob, 2024b) tarafından yapılan bir başka çalışmada LSTM modeli ile %98.8 doğruluk ve %98,6 F1-skoru elde edilmiştir . Bu bulgular, geleneksel makine öğrenmesi ve belirli derin öğrenme mimarilerinin, uygun veri ön işleme ve hiperparametre optimizasyonu ile bu karmaşık veri setinde dahi neredeyse mükemmel bir sınıflandırma başarısı sergileyebileceğini göstermektedir.

Bizim çalışmamızda kullanılan TabM modelinin %97,91 doğruluk ve %86,25 F1-skoru, literatürdeki en yüksek F1-skorlarına kıyasla daha mütevazı kalmaktadır. Ancak, çalışmamızın temel katkısı, modern bir derin öğrenme mimarisi olan TabM'in bu zorlu ve çok sınıflı veri seti üzerindeki ilk kapsamlı değerlendirmesini sunmasıdır. Elde edilen %86,25'lik F1-skoru, Neto ve diğerlerinin sunduğu %71,40'lık temel F1-skorunu önemli ölçüde aşmaktadır. Bu durum, TabM mimarisinin ve uygulanan hibrit stratejinin, temel makine öğrenmesi yaklaşımlarına göre sınıflar arası dengeyi daha iyi sağladığını ve saldırı türlerini ayırt etmede daha dengeli bir başarı sergilediğini göstermektedir. Literatürdeki diğer çalışmaların başarısı, gelecekte TabM mimarisinin daha ileri hiperparametre optimizasyonu ve veri dengeleme teknikleri ile birleştirilerek performansının daha da artırılabileceğine işaret etmektedir.

Tablo 4.14. Literatür karşılaştırması

Çalışma	Yıl	Sınıf Sayısı	Model	Performans (F1/Doğruluk)	Katkı / Fark
Neto et al.	2023	34	Rasgele Orman	F1: %71,40, Acc: %99,16	Veri setinin tanıtımı ve temel (baseline) sonuçlar
Tseng et al.	2024	8	Transformer	Acc: %99,40	Sekiz ana kategori üzerinde derin öğrenme modeli karşılaştırması
Khan & Alkhathami	2024	34	Rasgele Orman	Acc: %99,55	Dengelenmiş veri seti (SMOTE) kullanımı
Jony & Arnob (Akour et al. içinde)	2024	34	LSTM	F1: %98,60, Acc: %98,80	LSTM modelinin 34 sınıf üzerindeki başarımı
Akour et al.	2025	34	XBNet	F1: %95,60, Acc: %96,70	XBNet modelinin kapsamlı değerlendirilmesi
Alve et al.	2025	34	Karar Ağacı	**F1: %99,62**, Acc: %99,56	Hafif ML modelleri ile yüksek başarım
Bu Çalışma	2026	**34**	**TabM**	**F1: %86,25, Acc: %97,91**	**TabM'in ilk kapsamlı değerlendirmesi ve hibrit strateji**

Çalışmanın katkıları:

Bu çalışma, IoT ağ saldırı tespiti alanında literatüre metodolojik, deneysel ve açıklanabilirlik açısından özgün katkılar sunmaktadır. Aşağıda, çalışmanın temel katkıları detaylı olarak açıklanmaktadır:

1. **TabM Mimarisinin IoT Ağ Saldırı Tespitinde İlk Kapsamlı Değerlendirmesi**
   Bu tez, Gorishniy ve arkadaşları tarafından önerilen TabM mimarisini flow-tabanlı IoT saldırı tespiti problemine uyarlayarak, CIC-IoT-2023 veri seti üzerinde çok sınıflı (34 sınıf) ve yüksek dengesizlik içeren bir senaryoda performansını raporlamaktadır. Belirlenen deneysel kurulum altında TabM, Makro-F1=0,8625 ve Doğruluk=%97,91 sonuçları üretmiş; Rasgele Orman (Makro-F1=0,8819) ve XGBoost (Makro-F1=0,8279) ile yapılan kıyaslamalarda rekabetçi bir düzey sergilemiştir. Ayrıca TabM'in BatchEnsemble tabanlı parametre verimli topluluk yapısının (k=32) bu problem bağlamındaki etkisi, karşılaştırmalı deneyler üzerinden nicel olarak değerlendirilmiştir.
2. **Çok Sınıflı Saldırı Türü Sınıflandırması (1 benign + 33 saldırı türü)**
   Bu çalışma, yalnızca normal trafik veya atak ikili ayrım yerine, CIC-IoT-2023 üzerinde 34 sınıflı çok sınıflı sınıflandırma gerçekleştirerek saldırı türünün ayrıntılı biçimde belirlenmesini hedeflemiştir. Bu tercih, olay müdahale ve analiz süreçlerinde (ör. DDoS için trafik sınırlama/filtreleme, enjeksiyon saldırıları için uygulama katmanı önlemleri) daha eyleme dönük çıktılar üretmeyi sağlar. Çok sınıflı problem, yüksek sınıf dengesizliği (yaklaşık 400:1), sınıflar arası benzerlik (örn. Recon alt türleri) ve karar uzayının büyümesi nedeniyle ikili sınıflandırmaya göre belirgin biçimde daha zordur. Buna rağmen, önerilen model sınıf-bazlı değerlendirmede 34 sınıfın 20'sinde F1 $\geq$ 0,95 seviyesine ulaşarak birçok saldırı türünde yüksek ayırt edicilik sergilemiştir.
3. **Sınıf Dengesizliğine Yönelik Hibrit Ön-işleme/Loss Stratejisi ve Ablasyon Analizi**
   Bu tez, CIC-IoT-2023'teki yüksek sınıf dengesizliğini ele almak amacıyla SMOTE ve Quantile Transform bileşenlerinden oluşan hibrit bir strateji uygulamış, loss fonksiyonu alternatifleri (Focal Loss) ve sınıf ağırlıklandırma yaklaşımlarını test etmiş ve her bileşenin etkisini altı farklı konfigürasyonla sistematik ablasyon çalışmasıyla nicel olarak incelemiştir. Ablasyon sonuçları, Quantile Transform'un performansa en yüksek katkıyı sağladığını, SMOTE'un ise paradoksal olarak genel performansı düşürdüğünü ortaya koymuştur. Detaylı

azınlık sınıf analizi, SMOTE'un azınlık sınıfları için duyarlılığı artırırken kesinliği düşürerek F1 skorunu azalttığını göstermiştir. CrossEntropy yerine Focal Loss kullanımı minimal etki gösterirken, sınıf ağırlıklarının eklenmesi performansı düşürmüştür. Bu bulgular, sınıf dengesizliği stratejilerinin bağlama özgü değerlendirilmesi ve SMOTE ile ek ağırlıklandırma yöntemlerinin birlikte kullanımında aşırı düzeltme riskinin dikkate alınması gerektiğini ortaya koymaktadır. Tüm dönüşümler ve yeniden örnekleme adımları yalnızca eğitim bölümü üzerinde uygulanarak değerlendirme sızıntısı engellenmiştir.

4. **SHAP ile Model Yorumlanabilirliği ve Özellik Önem Analizi**
   Bu çalışma, TabM modelinin tahminlerini SHAP tabanlı açıklanabilirlik analizi ile inceleyerek**,** global özellik önemini nicel olarak raporlamıştır. Elde edilen sonuçlar, IAT gibi zamanlama/akış karakteristiklerinin model kararlarında belirgin ağırlığa sahip olduğunu; ayrıca Header_Length, flow_duration ve TCP bayrak sayılarını temsil eden değişkenlerin de ayırt ediciliğe katkı sunduğunu göstermektedir. Bu analiz, modelin hangi özellik familyalarına dayandığını görünür kılarak, güvenlik uzmanları için izleme önceliklendirmesi ve özellik mühendisliği kararlarına pratik destek sağlamaktadır.
5. **Flow-based Veri Temsilinin Uygulama Katmanı Saldırıları İçin Sınırlılıklarının Bulgularla Analizi**
   Bu çalışma, CIC-IoT-2023 gibi flow-temelli temsillerin, payload içermemesi nedeniyle uygulama katmanı saldırılarında yapısal sınırlılıklar taşıdığını, sınıf-bazlı performans bulguları üzerinden göstermektedir. Deney sonuçlarında SqlInjection (F1=0,26), XSS (F1=0,53), Vulnerability_scanner (F1=0,43) ve Uploading_Attack (F1=0,67) gibi sınıflarda performansın görece düşük kalması, ayırt edici sinyalin büyük ölçüde HTTP istek içeriği/header'ları ve yük verisinde bulunmasıyla tutarlıdır. Bu analiz, gelecekte flow + uygulama katmanı üstverisi temelli hibrit yaklaşımlar ile (TLS/mahremiyet kısıtlarını gözeterek) bu sınırlılıkların aşılabileceğine dair somut bir yönlendirme sunmaktadır.

Sınırlılıklar:

Her bilimsel çalışmada olduğu gibi, bu çalışmanın da belirli sınırlılıkları bulunmaktadır. Bu sınırlılıklar, sonuçların yorumlanmasında ve gelecek çalışmaların tasarlanmasında dikkate alınmalıdır:

- TabM modelinin eğitim süresi (1.025 saniye, ~17 dakika), ensemble ağaç tabanlı yöntemlere kıyasla daha uzundur. Rasgele Orman'ın eğitim süresi (775 saniye, ~13 dakika) TabM'den %32 daha kısa iken, XGBoost en hızlı eğitim süresine (299 saniye, ~5 dakika) sahiptir.
- Model, azınlık sınıflarında (support < 1.000 örnek) sınırlı performans göstermektedir. Özellikle, Uploading_Attack (F1: 0,282, support: 250), Recon-PingSweep (F1: 0,365, support: 452) ve XSS (F1: 0,527, support: 769) gibi sınıflarda F1 skorları 0,60'ın altındadır.
- Flow-based özellik seti (46 özellik: paket sayısı, byte miktarı, IAT, protokol türü, port numaraları, TCP flag'leri, vb.), ağ trafiğinin istatistiksel özelliklerini içerirken, paket içeriğini (payload) içermemektedir. Bu, uygulama katmanı (Layer 7) saldırılarının tespitini doğası gereği sınırlandırmaktadır..
- CIC-IoT-2023 veri seti, kontrollü laboratuvar ortamında, önceden tanımlanmış saldırı senaryoları kullanılarak oluşturulmuştur. Bu sentetik trafik, gerçek dünya IoT ağlarındaki karmaşıklığı ve çeşitliliği tam olarak yansıtmayabilir.

5. SONUÇLAR VE ÖNERİLER

5.1 Sonuçlar

Bu tez çalışmasında, CIC-IoT-2023 veri seti üzerinde TabM derin öğrenme modeli kullanılarak 34 sınıflı ağ saldırı tespiti gerçekleştirilmiştir. Çalışmanın temel bulguları aşağıda özetlenmektedir:

Model performansı: Optimize edilmiş TabM modeli, test setinde F1-Makro 0,8625, Doğruluk %97,91 ve F1-Weighted 0,9789 değerlerine ulaşmıştır. 5-katlı çapraz doğrulamada F1-Makro ortalaması 0,784 ± 0,004 olarak hesaplanmış olup, bu düşük standart sapma modelin farklı veri bölümlerinde tutarlı performans sergilediğini göstermektedir.

Geleneksel yöntemlerle karşılaştırma: Rasgele Orman (F1-Makro: 0,882) en yüksek performansı gösterirken, TabM (F1-Makro: 0,858) ikinci sırada yer almış ve XGBoost (F1-Makro: 0,828) modelini geçmiştir. Rasgele Orman'ın %2,7'lik marjinal üstünlüğü, tablo verileri için gradient boosting yöntemlerinin güçlü bir alternatif olduğunu doğrulamaktadır. Bu bulgu, Grinsztajn ve diğerlerinin orta ölçekli tablo veri setlerinde GBDT yöntemlerinin derin öğrenme modellerine yakın veya üstün performans gösterdiği tespiti ile tutarlıdır.

Sınıf dengesizliği stratejileri: Ablasyon çalışması sonuçları, uygulanan tekniklerin katkılarını ortaya koymuştur:

- **SMOTE:** Genel F1-Makro üzerindeki etkisi karmaşık olmakla, azınlık sınıflarının tespit edilebilirliğini (duyarlılık) 0,4891'den 0,6475'e artırmıştır.
- **Quantile Transform:** Çıkarıldığında F1-Makro 0,0834 puan düşmüştür.
- **Focal Loss + Class Weights:** SMOTE ile birlikte kullanıldığında aşırı düzeltme etkisi yaratarak performansı düşürmüştür.

Bu bulgular, sentetik örnek üretimi ve aykırı değerlere duyarsız normalleştirmenin sınıf dengesizliği probleminde kritik öneme sahip olduğunu göstermektedir.

Özellik önem analizi: SHAP analizi, IAT , Header_Length, Weight ve Protocol Type gibi zamansal özelliklerin saldırı tespitinde en ayırt edici özellikler olduğunu ortaya koymuştur. IAT özelliğinin yüksek önemi, DDoS gibi saldırılarda paket gönderim zamanlamasının anormal olmasından kaynaklanmaktadır.

Azınlık sınıfları: 34 sınıftan 20'si ≥0,95, 23'ü ≥0,85 F1 skoruna ulaşmıştır. Ancak bazı sınıflarda düşük performans gözlemlenmiştir: Uploading_Attack (F1: 0,28), Recon-PingSweep (F1: 0,37), XSS (F1: 0,53), SqlInjection (F1: 0,54). Bu düşük performansın temel nedeni, CIC-IoT-2023 veri setinin flow-based yapısıdır. SQL injection gibi uygulama katmanı saldırıları için gerekli payload ve HTTP içerik bilgileri akış seviyesindeki özellik setinde bulunmamaktadır.

5.2 Öneriler

Uygulayıcılar için öneriler:

- Tabular formattaki ağ trafiği verileri için Rasgele Orman veya XGBoost gibi ensemble yöntemler, eğitim süresi açısından avantajlı olmakla birlikte, TabM'in BatchEnsemble mimarisi parametre verimliliği (1,52 MB model boyutu) ve uç ortamda konuşlandırma potansiyeli açısından alternatif sunmaktadır. Uygulama senaryosuna göre performans-verimlilik dengesi değerlendirilmelidir.
- Sınıf dengesizliği probleminde, Quantile Transform kritik öneme sahiptir ve mutlaka uygulanmalıdır. SMOTE kullanımı ise uygulama gereksinimlerine göre değerlendirilmelidir: Yüksek recall gerekiyorsa (saldırı kaçırma maliyeti yüksekse) SMOTE uygulanmalı, yüksek precision gerekiyorsa (yanlış alarm maliyeti yüksekse) SMOTE'suz eğitim tercih edilebilir.
- Gerçek zamanlı sistemlerde model seçiminde çıkarım süresi göz önünde bulundurulmalıdır. TabM (0,31 ms/örnek), Rasgele Orman (0,08 ms/ örnek) ve XGBoost (0,02 ms/ örnek) ile karşılaştırıldığında daha yavaş olmakla birlikte, saniyede ~3.200 örnek işleyebilme kapasitesi ile ağ trafiği analizi için kabul edilebilir düzeydedir. Yüksek işleme hızı gerektiren kritik altyapılarda Rasgele Orman veya XGBoost tercih edilebilir.
- Akış tabanlı saldırı tespit sistemleri kurulurken, uygulama katmanı saldırılarının (SQL enjeksiyonu, XSS) tespit edilemeyeceği göz önünde bulundurulmalı ve gerektiğinde içerik tabanlı sistemlerle desteklenmelidir. Karma bir yaklaşımda, akış tabanlı sistem ağ seviyesi saldırıları (DDoS, DoS, port tarama) tespit ederken; derin paket inceleme tabanlı içerik tabanlı sistem, uygulama katmanı tehditleri için kullanılmalıdır.

Gelecek çalışmalar için öneriler:

- İçerik tabanlı özellik birleştirmesi: Akış tabanlı özelliklerin yanı sıra derin paket inceleme ile elde edilen HTTP paket içeriği, URL kalıbı, istek gövdesi ve HTTP başlığı özellikleri eklenerek, uygulama katmanı saldırılarının (SQL enjeksiyonu, XSS) tespit başarımı artırılabilir. Karma özellik seti ile hem ağ hem de uygulama katmanı tehditleri tek model ile tespit edilebilir.
- Gerçek dünya trafiği değerlendirmesi: CIC-IoT-2023 gibi laboratuvar ortamında üretilen veri setleri kontrollü koşullar sağlamakla birlikte, gerçek ağ trafiğinin gürültü, protokol çeşitliliği ve sıfırıncı gün saldırılarını tam yansıtmayabilir. Modelin gerçek ortamlarda (kurumsal ağlar, ISP trafiği) test edilmesi ve alan kayması etkisinin değerlendirilmesi gerekmektedir.
- Bilgi aktarımına dayalı öğrenme: CIC-IoT-2023 üzerinde eğitilmiş modelin, farklı ağ ortamlarına (endüstriyel IoT, akıllı şehir, sağlık sistemleri) adaptasyonu için ince ayar yaklaşımları araştırılmalıdır. TabM'in BatchEnsemble mimarisi, son katman yeniden eğitimi ile hızlı adaptasyon potansiyeline sahiptir.
- Model sıkıştırma: TabM'in mevcut 1,52 MB model boyutu uç ortamda konuşlandırma için uygun olmakla birlikte, 8-bit quantization ve pruning teknikleri ile model boyutu <500 KB'ye düşürülebilir ve çıkarım süresi (0,31 ms/örnek) daha da iyileştirilebilir. Bu, IoT ağ geçitlerinde gerçek zamanlı tespit için kritik öneme sahiptir.
- Hibrit topluluk yaklaşımları: TabM ve Rasgele Orman'ın güçlü yönlerini birleştiren topluluk stratejileri araştırılmalıdır. Örneğin, Rasgele Orman yüksek kesinlik sağlarken, TabM azınlık sınıfları için duyarlılığı artırabilir. Hibrit topluluk yöntemi, F1-Macro skorunu RF'in 0,882'sinin üzerine çıkarma potansiyeline sahiptir.
- Loss fonksiyonu alternatifleri: Focal Loss ve class weights parametrelerinin SMOTE olmadan bağımsız optimizasyonu araştırılmalıdır. Mevcut çalışmada bu yöntemlerin SMOTE ile birlikte aşırı düzeltme yarattığı gözlemlenmiştir; ancak SMOTE'suz eğitimde daha etkili olabilirler. Ayrıca, asymmetric loss gibi alternatif loss fonksiyonları değerlendirilebilir.

KAYNAKLAR

Al-Haboosi, I. T., Bassant M. Elbagoury, Salsabil El-Regaily, ve El-Sayed M. El-Horbaty. (2024). A Hybrid-Transformer-Based Cyber-Attack Detection in IoT Networks. International Journal of Interactive Mobile Technologies (iJIM), 18(14), 90-102. https://doi.org/10.3991/ijim.v18i14.50343

Alve, S. R., Mahmud, M. Z., Islam, S., Chowdhury, M. A., ve Islam, J. (2025). Smart IoT Security: Lightweight Machine Learning Techniques for Multi-Class Attack Detection in IoT Networks. 2025 International Conference on Quantum Photonics, Artificial Intelligence, and Networking (QPAIN), 1-6. https://doi.org/10.1109/QPAIN66474.2025.11171769

Associate Professor, Department of Computer Science Government Arts College, Thuvakudimalai, Tiruchirappalli, India, ve Kumar, D. A. (2017). INTRUSION DETECTION SYSTEMS: A REVIEW. International Journal of Advanced Research in Computer Science, 8(8), 356-370. https://doi.org/10.26483/ijarcs.v8i8.4703

Becerra-Suarez, F. L., Tuesta-Monteza, V. A., Mejia-Cabrera, H. I., ve Arcila-Diaz, J. (2024). Performance Evaluation of Deep Learning Models for Classifying Cybersecurity Attacks in IoT Networks. Informatics, 11(2), 32. https://doi.org/10.3390/informatics11020032

Camadini, L., Bouzid, Y., Merlet, M., ve Carron, L. (2024). Randomness control and reproducibility study of random forest algorithm in R and Python (arXiv:2408.12184). arXiv. https://doi.org/10.48550/arXiv.2408.12184

Carvalho, M., Pinho, A. J., ve Brás, S. (2025). Resampling approaches to handle class imbalance: A review from a data perspective. Journal of Big Data, 12(1), 71. https://doi.org/10.1186/s40537-025-01119-4

Chawla, N., Bowyer, K., Hall, L., ve Kegelmeyer, W. (2002). SMOTE: Synthetic Minority Over-sampling Technique. J. Artif. Intell. Res. (JAIR), 16, 321-357. https://doi.org/10.1613/jair.953

Cobos, E. V., ve Cakir, S. (t.y.). A Review of the Economic Costs of Cyber Incidents.

De Amorim, L. B. V., Cavalcanti, G. D. C., ve Cruz, R. M. O. (2023). The choice of scaling technique matters for classification performance. Applied Soft Computing, 133, 109924. https://doi.org/10.1016/j.asoc.2022.109924

Di Mauro, M., Galatro, G., Fortino, G., ve Liotta, A. (2021). Supervised feature selection techniques in network intrusion detection: A critical review. Engineering Applications of Artificial Intelligence, 101, 104216. https://doi.org/10.1016/j.engappai.2021.104216

Fayaz, S. A., Zaman, M., Kaul, S., ve Butt, M. A. (2022). Is Deep Learning on Tabular Data Enough? An Assessment. International Journal of Advanced Computer Science and Applications, 13(4). https://doi.org/10.14569/IJACSA.2022.0130454

Gheni, H. Q., ve Al-Yaseen, W. L. (2024). Two-step data clustering for improved intrusion detection system using CICIoT2023 dataset. E-Prime - Advances in Electrical Engineering, Electronics and Energy, 9, 100673. https://doi.org/10.1016/j.prime.2024.100673

Golestani, S., ve Makaroff, D. (2024a). Device-Specific Anomaly Detection Models for IoT Systems. 2024 IEEE Conference on Communications and Network Security (CNS), 1-6. https://doi.org/10.1109/CNS62487.2024.10735608

Golestani, S., ve Makaroff, D. (2024b). Device-Specific Anomaly Detection Models for IoT Systems. 2024 IEEE Conference on Communications and Network Security (CNS), 1-6. https://doi.org/10.1109/CNS62487.2024.10735608

Gorishniy, Y., Kotelnikov, A., ve Babenko, A. (2025a). TabM: Advancing Tabular Deep Learning with Parameter-Efficient Ensembling (arXiv:2410.24210). arXiv. https://doi.org/10.48550/arXiv.2410.24210

Gorishniy, Y., Kotelnikov, A., ve Babenko, A. (2025b). TabM: Advancing tabular deep learning with parameter-efficient ensembling. The Thirteenth International Conference on Learning Representations. https://openreview.net/forum?id=Sd4wYYOhmY

Grinsztajn, L., Oyallon, E., ve Varoquaux, G. (2022). Why do tree-based models still outperform deep learning on tabular data? (Versiyon 1). arXiv. https://doi.org/10.48550/ARXIV.2207.08815

Happy, Chhikara, R., ve Kashyap, N. (2024a). A Comparative Analysis of Machine Learning Prediction Algorithms for Detecting IoT Botnet Activities. 2024 International Conference on Intelligent Systems for Cybersecurity (ISCS), 1-6. https://doi.org/10.1109/ISCS61804.2024.10581089

Happy, Chhikara, R., ve Kashyap, N. (2024b). A Comparative Analysis of Machine Learning Prediction Algorithms for Detecting IoT Botnet Activities. 2024 International Conference on Intelligent Systems for Cybersecurity (ISCS), 1-6. https://doi.org/10.1109/ISCS61804.2024.10581089

Hinojosa, A., ve Majd, N. E. (2024a). Edge Computing Network Intrusion Detection System in IoT Using Deep Learning. 2024 33rd International Conference on Computer Communications and Networks (ICCCN), 1-6. https://doi.org/10.1109/ICCCN61486.2024.10637611

Hinojosa, A., ve Majd, N. E. (2024b). Edge Computing Network Intrusion Detection System in IoT Using Deep Learning. 2024 33rd International Conference on Computer Communications and Networks (ICCCN), 1-6. https://doi.org/10.1109/ICCCN61486.2024.10637611

Hizal, S., Cavusoglu, U., ve Akgun, D. (2024). A novel deep learning-based intrusion detection system for IoT DDoS security. Internet of Things, 28, 101336. https://doi.org/10.1016/j.iot.2024.101336

Jony, A. I., ve Arnob, A. K. B. (2024a). A long short-term memory based approach for detecting cyber attacks in IoT using CIC-IoT2023 dataset. Journal of Edge Computing, 3(1), 28-42. https://doi.org/10.55056/jec.648

Jony, A. I., ve Arnob, A. K. B. (2024b). A long short-term memory based approach for detecting cyber attacks in IoT using CIC-IoT2023 dataset. Journal of Edge Computing, 3(1), 28-42. https://doi.org/10.55056/jec.648

Jony, A. I., ve Arnob, A. K. B. (2024c). A long short-term memory based approach for detecting cyber attacks in IoT using CIC-IoT2023 dataset. Journal of Edge Computing, 3(1), 28-42. https://doi.org/10.55056/jec.648

Kala, E. S. M. (2023). Critical Role of Cyber Security in Global Economy. Open Journal of Safety Science and Technology, 13(04), 231-248. https://doi.org/10.4236/ojsst.2023.134012

Khan, M. M., ve Alkhathami, M. (2024). Anomaly detection in IoT-based healthcare: Machine learning for enhanced security. Scientific Reports, 14(1), 5872. https://doi.org/10.1038/s41598-024-56126-x

Le, T.-T.-H., Kim, Haeyoung, Kang, H., ve Kim, Howon. (2022). Classification and Explanation for Intrusion Detection System Based on Ensemble Trees and SHAP Method. Sensors, 22(3), 1154. https://doi.org/10.3390/s22031154

Lee, H., ve Ahn, S. (2023). Improving the performance of object detection by preserving label distribution (arXiv:2308.14466). arXiv. https://doi.org/10.48550/arXiv.2308.14466

Lin, T.-Y., Goyal, P., Girshick, R., He, K., ve Dollar, P. (2017). Focal Loss for Dense Object Detection. 2017 IEEE International Conference on Computer Vision (ICCV), 2999-3007. https://doi.org/10.1109/ICCV.2017.324

Loshchilov, I., ve Hutter, F. (2019). Decoupled Weight Decay Regularization (arXiv:1711.05101). arXiv. https://doi.org/10.48550/arXiv.1711.05101

Lundberg, S., ve Lee, S.-I. (2017). A Unified Approach to Interpreting Model Predictions (Versiyon 2). arXiv. https://doi.org/10.48550/ARXIV.1705.07874

Maalouf, M. (2011). Logistic regression in data analysis: An overview. International Journal of Data Analysis Techniques and Strategies, 3(3), 281. https://doi.org/10.1504/IJDATS.2011.041335

Mallik, A., Ahsan, A., Shahadat, M. M. Z., ve Tsou, J. C. (2019). Understanding Man-in-the-middle-attack through Survey of Literature. Indonesian Journal of Computing, Engineering and Design (IJoCED), 1(1), 44. https://doi.org/10.35806/ijoced.v1i1.36

McDermott, C. D., Petrovski, A. V., ve Majdani, F. (2018). Towards Situational Awareness of Botnet Activity in the Internet of Things. 2018 International Conference On Cyber Situational Awareness, Data Analytics And Assessment (Cyber SA), 1-8. https://doi.org/10.1109/CyberSA.2018.8551408

Narayan, K. R., Mookherji, S., Odelu, V., Prasath, R., Turlapaty, A. C., ve Das, A. K. (2023). IIDS: Design of Intelligent Intrusion Detection System for Internet-of-Things Applications (arXiv:2308.00943). arXiv. https://doi.org/10.48550/arXiv.2308.00943

Neto, E. C. P., Dadkhah, S., Ferreira, R., Zohourian, A., Lu, R., ve Ghorbani, A. A. (2023a). CICIoT2023: A Real-Time Dataset and Benchmark for Large-Scale Attacks in IoT Environment. Sensors, 23(13), 5941. https://doi.org/10.3390/s23135941

Neto, E. C. P., Dadkhah, S., Ferreira, R., Zohourian, A., Lu, R., ve Ghorbani, A. A. (2023b). CICIoT2023: A Real-Time Dataset and Benchmark for Large-Scale Attacks in IoT Environment. Sensors, 23(13), Makale 13. https://doi.org/10.3390/s23135941

Nidhi. (2024). An Ensemble Model for Cyber Attack and Threat Detection in Applications Network Using Random Forest, Lightgbm and Xgboost. Advances in Nonlinear Variational Inequalities, 28(3s), 523-534. https://doi.org/10.52783/anvi.v28.3121

Odiaga Gloria Awuor. (2023). Assessment of existing cyber-attack detection models for web-based systems. Global Journal of Engineering and Technology Advances, 15(1), 070-089. https://doi.org/10.30574/gjeta.2023.15.1.0075

Rida, A. (2024). Machine and Deep Learning for Credit Scoring: A compliant approach (arXiv:2412.20225). arXiv. https://doi.org/10.48550/arXiv.2412.20225

Scarfone, K. A., Souppaya, M. P., Cody, A., ve Orebaugh, A. D. (2008). Technical guide to information security testing and assessment. (NIST SP 800-115; 0 bs., s. NIST SP 800-115). National Institute of Standards and Technology. https://doi.org/10.6028/NIST.SP.800-115

Shimazoe, J., Arai, K., ve Oda, M. (2023). Method for Hyperparameter Tuning of EfficientNetV2-based Image Classification by Deliberately Modifying Optuna Tuned Result. International Journal of Advanced Computer Science and Applications, 14(12). https://doi.org/10.14569/IJACSA.2023.0141248

Shwartz-Ziv, R., ve Armon, A. (2022). Tabular data: Deep learning is not all you need. Information Fusion, 81, 84-90. https://doi.org/10.1016/j.inffus.2021.11.011

T R, M., V, V. K., V, D. K., Geman, O., Margala, M., ve Guduri, M. (2023). The stratified K-folds cross-validation and class-balancing methods with high-performance ensemble classifiers for breast cancer classification. Healthcare Analytics, 4, 100247. https://doi.org/10.1016/j.health.2023.100247

Terven, J., Cordova-Esparza, D.-M., Romero-González, J.-A., Ramírez-Pedraza, A., ve Chávez-Urbiola, E. A. (2025). A comprehensive survey of loss functions and metrics in deep learning. Artificial Intelligence Review, 58(7), 195. https://doi.org/10.1007/s10462-025-11198-7

Tseng, S.-M., Wang, Y.-Q., ve Wang, Y.-C. (2024). Multi-Class Intrusion Detection Based on Transformer for IoT Networks Using CIC-IoT-2023 Dataset. Future Internet, 16(8), 284. https://doi.org/10.3390/fi16080284

Wang, M., Zheng, K., Yang, Y., ve Wang, X. (2020). An Explainable Machine Learning Framework for Intrusion Detection Systems. IEEE Access, 8, 73127-73141. https://doi.org/10.1109/ACCESS.2020.2988359

Yaras, S., ve Dener, M. (2024). IoT-Based Intrusion Detection System Using New Hybrid Deep Learning Algorithm. Electronics, 13(6), 1053. https://doi.org/10.3390/electronics13061053