T.C.
SAKARYA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

AĞ ANOMALİSİ TESPİTİNDE MAKİNE ÖĞRENMESİ ALGORİTMALARININ KULLANIMI VE KARŞILAŞTIRMALI ANALİZİ

YÜKSEK LİSANS TEZİ

Mujibullah SHAMS

Enstitü Anabilim Dalı	**:**	**BİLGİSAYAR VE BİLİŞİM MÜHENDİSLİĞİ**
Tez Danışmanı	**:**	**Dr.Öğr.Üyesi Murat İSKEFİYELİ**

Ağustos 2020

T.C.
SAKARYA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

AĞ ANOMALİSİ TESPİTİNDE MAKİNE ÖĞRENMESİ ALGORİTMALARININ KULLANIMI VE KARŞILAŞTIRMALI ANALİZİ

YÜKSEK LİSANS TEZİ

Mujibullah SHAMS

Enstitü Anabilim Dalı : BİLGİSAYAR VE BİLİŞİM MÜHENDİSLİĞİ

Bu tez 28/08/2020 tarihinde aşağıdaki jüri tarafından oybirliği/oyçokluğu ile kabul edilmiştir.

Prof. Dr.	**Prof. Dr.**	**Dr.Öğr.Üyesi**
Resul KARA	**İbrahim ÖZÇELİK**	**Murat İSKEFİYELİ**
Jüri Başkanı	**Üye**	**Üye**

BEYAN

Tez içindeki tüm verilerin akademik kurallar çerçevesinde tarafımdan elde edildiğini, görsel ve yazılı tüm bilgi ve sonuçların akademik ve etik kurallara uygun şekilde sunulduğunu, kullanılan verilerde herhangi bir tahrifat yapılmadığını, başkalarının eserlerinden yararlanılması durumunda bilimsel normlara uygun olarak atıfta bulunulduğunu, tezde yer alan verilerin bu üniversite veya başka bir üniversitede herhangi bir tez çalışmasında kullanılmadığını beyan ederim.

Mujibullah Shams

28.08.2020

TEŞEKKÜR

Her şeyden önce, yüksek lisans tezimin danışmanlığını yapan Dr.Öğr.Üyesi. Murat İskefiyeli hocama içten teşekkürlerimi sunmak istiyorum. Çalışmamı üretken ve verimli hale getiren ayırdığı zaman, verdiği fikirleri ve yaptığı katkıları için takdir ediyorum. Onun değerli önerileri, yorumları ve rehberliği beni her geçen gün daha fazla öğrenmeye teşvik etmektedir. Onun derin anlayışları, araştırmamın çeşitli aşamalarında bana yardımcı olmuştur. Cömertliği, özverili desteği ve özellikle son üç yılda bana sağladığı mükemmel örnek ve sabır için de ona karşı borçluyum.

Ayrıca, jüri üyelerine, Prof.Dr. İbrahim Özçelik'e ve Prof.Dr. Resul Kara'ya, zor durumda bile olsa jüri olarak hizmet ettikleri için, savunmamın keyifli bir an olmasına izin verdikleri için ve parlak yorumları ve önerileri için teşekkür etmek istiyorum.

Türkiye Cumhuriyeti devletine, özellikle de Yurtdışı Türkler ve Akraba Topluluklar Başkanlığı'na Türkiye'deki en seçkin üniversitelerden birinde yüksek lisansımı sürdürme fırsatı ve finansal imkânlar sağladığı için çok minnettarım. Yardımları ve destekleri sayesinde edindiğim deneyimler için teşekkürlerimi sunmak istiyorum.

Kelimeler, hem çalışmamda hem de kariyerimde sürekli koşulsuz destekleri için, aileme duyduğum duyguları ifade edemez. Onlara da yaptığı her şey ve destekleri için teşekkür ederim.

Son olarak da, bu yaşam yolculuğuna çıkmama izin veren Yüce Rabbim'e şükrediyorum. Bu tezin her aşamasında rehberliğini her gün hissettim. Tanrım, bana sonsuz kutsaman için teşekkür ederim.

İÇİNDEKİLER

SİMGELER VE KISALTMALAR LİSTESİ

AB : AdaBoost
AFRL : Air Force Research Laboratory
AIN : Artificial Immune NetworK
ANN : Artificial Neural Network
AO : Algılama Oranı
AUC : Area Under Curve
C4.5 : Decision Tree Algorithm
CART : Classification And Regression Trees
CCI : Correctly Classified Instances
CIA : Confidentiality, Integrity, Availability
CIC : Canadian Institute for Cybersecurity
CNN : Convolutional Neural Network
CPU : Central Processing Unit
CRF : Correlation Ranking Filter
CSE : Communications Security Establishment
CSV : Comma-separated Values
CTF : Capture the Flag
DARPA : Defense Advanced Research Projects Agency
DDoS : Distributed Denial of Service
DL-4J : Deep Learning 4J Algorithm
DL-H2O : Deep Learning H2O Algorithm
DNN : Deep Neural Network
DoS : Denial of Service
DT : Decicion Tree
ELM : Extreme Learning Machine
FB-VQ : Fractal Technology and Vector Quantization

FC-ANN : Fuzzy Clustering Artifitial Neural Network
FCM : Fuzzy C-means
FL : Fuzzy Logic
FN : False Negative
FP : False Positive
FSVM : Fuzzy Support Vector Machine
FTP : File Transfer Protocol
GA : Genetic Algorithm
GAU : Gaussian Classifier
GB Trees : Gradient Boosting Trees
GPO : Gerçek Positif Oranı
GPU : Graphic Processing Unit
GRFE : Gain Ratio Feature Evaluator
HTTP : Hyper Text Transfer Protocol
HTTPS : Hyper Text Transfer Protocol Secure
hw-IBK : Heuristic Weight IBK
HYP : Hypersphere Algorithm
IBK : Instance Based Learner KNN
IBRF : Incremental RBF
ICO : Iterative Classifier Optimizer
ID : Identification
ID3 : Iterative Dichotomiser 3
IDS : Intrusion Detection System
IG : Information Gain
IMAP : Internet Message Access Protocol
IoT : Internet of Things
IP : Internet Protocol
IPv6 : Internet Protocol v.6
J48 : Decision Tree Algorithm
KDDCUP : Annual Data Mining and Knowledge Discovery competition
K-M : K-means
KNN : K-nearest Neighbor

LEA	: Leader Algorithm
LR	: Logistic Regression
LSSVM	: Least-Squares Support Vector Machine
LWL	: Locally Weighted Learning
MLP	: Multilayer Perceptron
MÖ	: Makine Öğrenmesi
NB	: Naive Bayes
NEA	: Nearest Clustering Algorithm
PCA	: Principal Component Analysis
PLSSVM	: Least Squares Support Vector Machine
POP3	: Post Office Protocol v.3
R2L	: Root to Local
RAM	: Random Access Memory
RBF	: Radial Basis Function
REP Tree	: Representative Tree
RF	: Random Forest
RFE	: Recursive Feature Elimination
RMSE	: Root Mean Square Error
ROC	: Receiver Operating Curve
SDG	: Stochastic Gradient Descent
SKL	: Scikit Learn
SLFN	: Single hidden Layer Feedforward Neural Networks
SMO	: Sequential Minimal Optimization
SMTP	: Simple Mail Transfer Protocol
SQL	: Structured Query Language
SSAD	: Dijkstra's Shortest Paths Algorithm
SSH	: Secure Shell
SSL / TLS	: Secure Socket Layer / Transport Layer Security
SVM	: Support Vector Machine
TCM-KNN	: Transductive Confidence Machines for K-Nearest Neighbor
TN	: True Negative
TP	: True Positive

TTLS	: Tunneled Transport Layer Security
U2R	: User to Root
UND	: Un-detection Rate
WEKA	: Waikato Environment for Knowledge Analysis
WMV	: Weigthed Majority Voting
XSS	: Cross-site scripting
YNO	: Yanliş Negatif Oranı
YOA	: Yanlış Alarm Oranı
YPO	: Yanlış Positif Oranı
YZ	: Yapay Zeka

ŞEKİLLER LİSTESİ

TABLOLAR LİSTESİ

ÖZET

Anahtar kelimeler: Anomali tespiti, makine öğrenmesi, denetimli öğrenme, CIC-IDS-2017, CSE-CIC-IDS-2018

Nesnelerin İnterneti ve 5G gibi teknolojilerin ortaya çıkmasıyla, internete bağlı kullanıcı ve cihaz sayısında büyük bir artış görülmüştür. Ancak, bu gelişmeye paralel olarak siber saldırıların sayısı da artmıştır. İnternet trafiği bağlamında, internette akan bu saldırılara ait veriler anomali olarak bilinmektedir. Bu anomalilere karşı önemli bir önlem, saldırıları önlemeye veya en azından etkilerini en aza indirmeye yardımcı olan Saldırı Tespit Sistemleri'dir. İzinsiz giriş (saldırı) tespiti için kullanılabilecek çeşitli yöntemler vardır, ancak son zamanlarda makine öğrenmesi teknikleri popülerlik kazanmıştır ve bu alanda başarılı sonuçlar göstermiştir.

Bu çalışmada ilk olarak ağ anomali tespiti bağlamında yapılan çalışmaların kapsamlı bir araştırması yapılmıştır. Dikkatle değerlendirildikten sonra, anomali tespiti için iyi sonuçlar elde ettiği kanıtlanmış yedi denetimli makine öğrenmesi algoritması ailesinden yedi algoritma seçilmiştir. Bu algoritmalar J48, Random Forest, K-nearest Neighbor, AdaBoost, Multilayer Perceptron, Support Vector Machines ve Naïve Bayes'tir. Bu algoritmaların performansları, CIC-IDS-2017 ve CSE-CIC-IDS-2018'in en güncel veri kümelerinden ikisini kullanarak doğrulukları, F-ölçüsü, Hassasiyeti, Geri Çağırma ve İşleme Süresi temelinde değerlendirilir. Özellik seçimi ve sınıflandırma yöntemlerinin rolünü değerlendirmek için Çok Terimli (saldırı tipine göre) ve İkili (anomali veya normal) olmak üzere iki tür sınıflandırma yapılmıştır.

Sonuçlar, J48, RF, KNN ve NB'nin başarılı sonuçlar elde edebildiğini ve bunları en güçlü sınıflandırıcılar olarak belirleyebildiğini göstermektedir. MLP-NN, SVM ve NB çoğu durumda iyi sonuçlar elde edememektedir. Ancak, daha dikkatli özellik çıkarma ile performanslarının geliştirilebileceği gösterilmektedir.

COMPARATIVE ANALYSIS OF MACHINE LEARNING ALGORITHMS IN NETWORK ANOMALY DETECTION

SUMMARY

Keywords: Anomaly detection, machine learning, supervised learning, CIC-IDS-2017, CSE-CIC-IDS-2018.

With the emergence of technologies such as Internet of Things and 5G, there has been a great boost in the number of users and the devices connected to the internet. However, parallel to this development so has increased the number of cyber-attacks. Within the context of internet traffic, the data belonging to these attacks flowing through internet are known as anomalies. A significant countermeasure against these anomalies Intrusion Detection Systems, which help avoid these attacks or at least minimize their effect. There are various methods that can be used for intrusion detection, however recently Machine Learning techniques has gained popularity and shown successful results in this area.

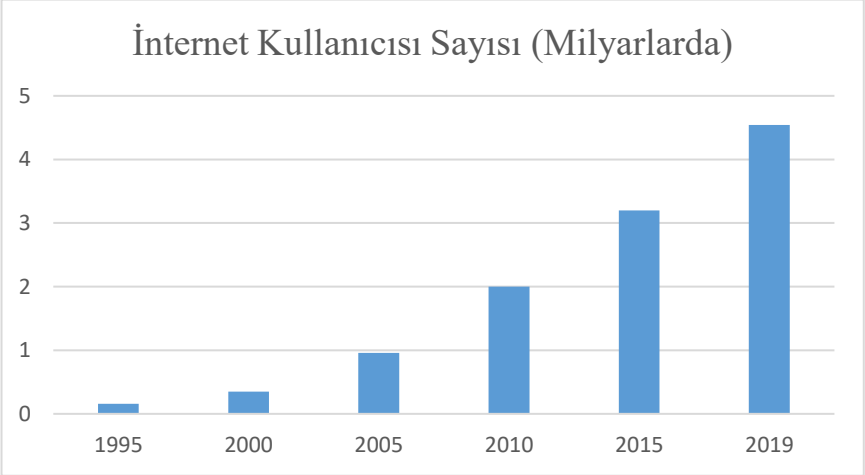
In this study firstly a comprehensive survey of the work done in the context of network anomaly detection is performed. After careful consideration seven algorithms from seven families of supervised ML algorithms, that have been proven to obtain good results for anomaly detection, are selected. These algorithms are J48, Random Forest, K-nearest Neighbor, AdaBoost, Multilayer Perceptron, Support Vector Machines and Naïve Bayes. The performance of these algorithms are evaluated based on their accuracy, F-measure, Precision, Recall and Processing Time using two of the most up to date datasets CIC-IDS-2017 and CSE-CIC-IDS-2018. In order to evaluate the role of feature selection and classification methods, two types of classification is performed Multinomial (attack-type-wise) and Binomial (anomaly or normal).

The results show that J48, RF, KNN and NB are able to achieve significant results and determine them as the strongest classifiers. MLP-NN, SVM and NB are not able to obtain good results in most cases. However we show that their performance can be improved with more careful feature extraction.

BÖLÜM 1. GİRİŞ

1.1. Motivasyon

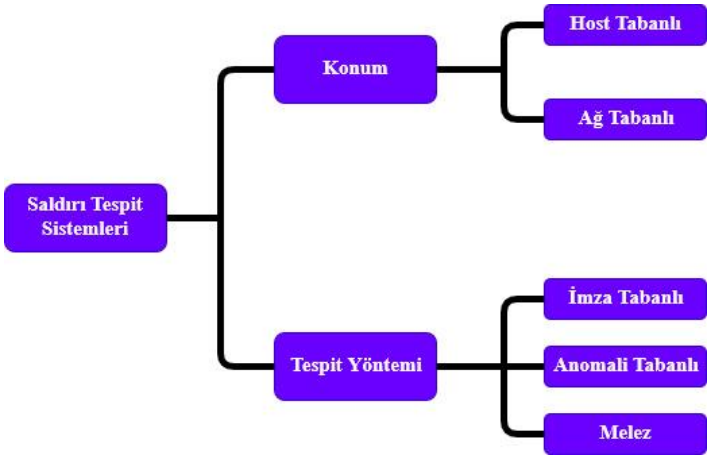
Son yıllarda, klasik yaşam biçimini dönüştüren ve günlük yaşamımızın önemli bir parçası haline gelen internet gibi çığır açan teknolojilerin ortaya çıkmasına şahit olmuşuzdur. İnternetin takibinde, Nesnelerin İnterneti (IoT) ve 5G gibi bilgisayar ağları dünyasındaki son trendler dünyayı yeni bir teknolojik devrim çağına getirmektedir. [1], internete bağlı cihaz sayısının 2025 yılına kadar 75.44 milyara ulaşacağını tahmin etmektedir. Şekil 1.1. 1955'ten günümüze internet kullanıcıların sayısının büyümesini gösteriyor.



Şekil 1.1. İnternet kullanıcılarının büyümesi 2005-2019 [2, 3].

Ancak bu gelişmelerle beraber internet üzerinden siber saldırıların sayısı da artmıştır. Bu saldırılar genellikle finansal kazanç, herhangi bir şekilde zarar verme, bilgi çalma veya meşru kullanıcılara hizmetleri düşürme amacıyla yapılmaktadır. Bu saldırıları tespit etmek veya önlemek için, araştırmacılar ve öncüler, Saldırı Tespit Sistemleri (IDS) olarak bilinen karşı önlemler geliştirmiştir. Bu sistemler saldırı trafiğini, onları

normal trafikten ayıran örüntüleri veya nitelikleri tanıyarak ayırt edebilmektedir [4]. Şekil 1.2.'de farklı IDS türleri açıklanmaktadır.



Şekil 1.2. Saldırı Tespit Sistemleri türleri

İmza tabanlı sistemler, daha önce bilinen saldırıların imzalarının veri tabanını oluşturarak saldırıları algılar. Bu sistemler genel olarak çok başarılıdır ve %100'e kadar algılama oranları sağlar, ancak sıfır gün saldırılarına (daha önce görülmemiş saldırılar) karşı etkisizdir. Bunun nedeni, günümüzde internet trafiğinin çoğunun şifrelenmiş olması (SSL / TLS trafiği) ve bu sistemlerin internet paketlerinin içeriğini izleyememesidir. Ancak anomali tabanlı sistemlerin paketlerin içeriğini izlemesi gerekmemektedir. Bu sistemler, paket sayısı, bağlantı süresi vb. özelliklerden yararlanıp ağ akışının davranışını analiz ederek saldırıları tespit edebilmektedir. Böylece bu yöntemi ağ trafiğinin olağandışı davranışlarına karşı ve sıfır gün saldırılarını tespit etmek için etkili kılmaktadır [4].

Bu çalışmada, anomali tabanlı tespit sistemlerinin artılarını ve eksilerini makine öğrenme (MÖ) tekniklerini kullanarak incelemeyi ve ağ anomalilerini hızlı ve etkin bir şekilde tespit edebilen bir sistem tasarlamayı amaçlamaktayız. Bu çalışma bu bağlamdaki ilgi alanları, anomali algılama yöntemlerinin, veri kümelerinin ve MÖ algoritmalarının incelenmesini içermektedir.

1.2. Araştırma Amaçları

Bu çalışmanın bitmesiyle, aşağıdaki amaçlar hedeflenmektedir.

- Bu alanda daha önce yapılan çalışmalar hakkında kapsamlı bir araştırma yapmak.
- Bilgisayar ağlarında anomalileri tespit etmek için en iyi yöntemi belirlemek.
- Bu çalışma bağlamında kullanılacak en uygun veri kümelerini belirlemek.
- Anomali tespiti için kullanılan MÖ tekniklerini incelemek ve bu çalışmanın amacı için en uygun algoritmaları seçmek.
- Ağ anomalilerini etkili ve hızlı bir şekilde tespit edebilen deneysel sistemler tasarlamak.
- Seçilen veri kümeler üzerinde MÖ algoritmalarını eğiterek ve test ederek, karşılaştırmalı bir şekilde bu algoritmalarının performansını değerlendirmek.
- Bu alanda daha önce yapılan çalışmalara yakın veya daha iyi sonuçlar elde ederek literatüre katkıda bulunmak.

1.3. Tez Yapısı

Bu belgenin geri kalanı aşağıdaki gibi düzenlenmiştir.

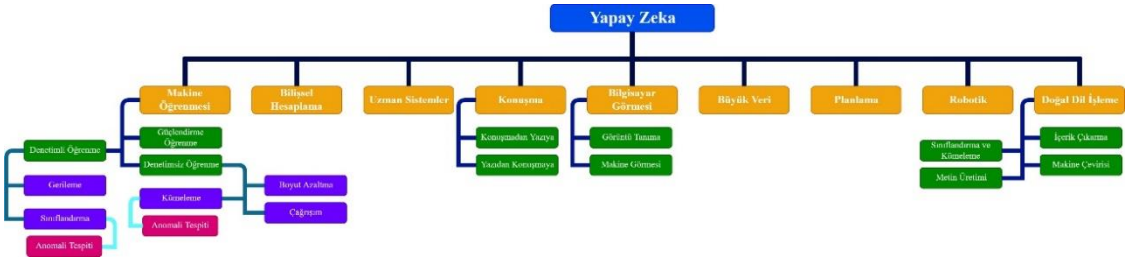
- Bölüm 2 Yapay Zeka (YZ), MÖ, anomali tespiti, ataklar ve veri kümeleri gibi gerekli terminolojiler ve literatür taraması hakkında bilgi sağlamaya odaklanmaktadır.
- Bölüm 3, bu çalışmanın amaçlarına ulaşmak için kullanılan metodolojiyi açıklamaktadır. Bu bölümdeki bilgiler, donanım ve yazılım platformlarının yanı sıra anomali algılama modellerini tasarlamak için izlenen adımlarla ilgilidir.
- Bölüm 4, deney sırasında elde edilen çalışmanın sonuçlarını sunmaktadır.
- Bölüm 5 çalışmayı özetlemekte olup, bu çalışma bağlamında gelecekteki araştırma yönergelerini sunar.

BÖLÜM 2. ARKA PLAN ÇALIŞMASI VE KAYNAK ARAŞTIRMASI

2.1. Yapay Zeka

YZ, Alan Turing'in (YZ'nin babası olarak da bilinir), bir varlığın zekasını belirlemek için ünlü Turing Testini tanıttığı, 1930'lardan bu yana bir akademik disiplin veya alan olarak biliniyordur [5]. O zamandan beri bu alan pek çok değişiklik geçirmiş olup, makine öğrenmesi, bilgisayar görmesi ve uzman sistemler gibi birkaç alt alanın ortaya çıkmasının nedeni olmuştur. Alanın birkaç öncüsü, yıllar boyunca çeşitli tanımlar sağlamıştır. Rich [6] YZ'yi "Yapay Zeka, bilgisayarların şu anda insanların daha iyi olduğu şeyleri nasıl daha iyi yapabilmelerinin araştırması" olarak tanımlamaktadır. Raphael'in YZ'nın [7] tanımı "Yapay Zeka, makineler tarafından insanların zekasını gerektiren şeyleri yapma bilimidir" demiştir. Bilgisayar bilimleri bakış açısına göre, YZ, ortamını algılayabilecek (dış verileri yorumlayabilecek), algılanan verilerden öğrenebilecek ve öğrendiklerini belirli hedeflere başarılı bir şekilde ulaşmak için kullanabilecek herhangi bir varlık olarak tanımlanabilir.

YZ'nin ana uygulama alanları sağlık, otomotiv, finans ve ekonomi, hükümet, video oyunları, askeri, denetim, reklam, sanat, bulut bilişim ve akıllı yönlendirmedir [8]. Ek olarak, son yıllarda YZ teknikleri, izinsiz giriş tespit ve önleme sistemleri başta olmak üzere, anomali ve sahtekarlık tespiti alanında yoğun bir şekilde kullanılmaktadır. Şekil 2.1. YZ'nin çeşitli dallarını ve Anomali Tespiti ile ilişkisini vurgulamaktadır.



Şekil 2.1. YZ'nin dalları

2.2. Makine Öğrenmesi

Makine Öğrenmesi, bilgisayarlara, veri ile çalışma esnasında desen tanımaya ve sonuç çıkarmaya dayanarak öğrenme yeteneği veren algoritmaların bilimsel çalışmasına odaklanan Yapay Zeka'nın bir uygulaması ve alt kümesidir. MÖ, bir makinenin belirli bir görevi etkin bir şekilde yerine getirmesi için açıkça programlanma yerine Eğitim Veriler olarak bilinen veriler üzerinde çalışarak öğrenilen deneyime dayalı olarak çalışmasını sağlar. MÖ süreci, eğitim veri kümelerindeki desenleri tanımlamayı ve yeni verilerle (test veri kümesi) ilgili tahmin ve karar almayı içerir [9].

MÖ, belirli bir görevi yerine getirmek için insan müdahalesinin ve açık talimatlarla bir algoritma geliştirmek pek mümkün olmadığı alanlarda Kullanılır. Bu alanlar arasında, bunlarla sınırlı olmamak üzere, ağ saldırı / güvenlik tehdidi tespiti, bilgisayar görmesi ve e-posta filtreleme yer almaktadır. Günümüz endüstrisinde, makine öğrenmesi finansal hizmetler, sağlık hizmetleri, perakende satış, petrol ve gaz firmaları ve taşımacılıkta yaygın olarak kullanılmaktadır [9].

ML yöntemleri aşağıdaki dört kategoriye ayrılır.

2.2.1. Denetimli öğrenme

Bu kategoriye giren algoritmalar, eğitim veri kümesinin içindeki etiketli örnekler kullanılarak eğitilir. Her örnek bir veya daha fazla girdi ve belirlenmiş bir çıktı içerir. Algoritma, optimal veya çıkarımlı bir fonksiyonun yinelenmesi yoluyla öğrenir ve daha sonra bu deneyimi, tahminler yapmak ve optimal bir sonuç veya çıktı üretmek için aynı özelliklere (Test veri kümesi) sahip yeni veri kümesinde kullanır.

Algoritmanın başarı oranını ve performansını değerlendirmek için sonuçlar daha sonra birincil çıktıyla karşılaştırılır. Sınıflandırma, regresyon, gradyan artırma ve kestirime dayanan algoritmalar denetimli öğrenme algoritmaları olarak sınıflandırılır [10].

Denetimli öğrenmede performans yüksektir ancak aynı zamanda manuel etiketleme gibi harici kaynaklar gerektirmesi nedeniyle maliyetli bir yöntemdir.

2.2.2. Denetimsiz öğrenme

Denetimli öğrenmenin aksine, bu kategoride veri kümesindeki örnekler yalnızca girdiler içerir ve çıktılar içermemektedir. Bu nedenle veriler hiçbir şekilde etiketlenmemiş ve sınıflandırılmamıştır. Bu yöntemde, algoritmalar çıktıların tahmini üzerinde işlem yapmaz, bunun aksine tek amaç girdileri çeşitli özelliklere göre gruplamak ve onların ilişkilerini gözlemlemektir. Başka bir deyişle, algoritmalar veri kümesi içindeki ortaklıklar olarak da bilinen veri noktalarının kümelenmesi veya gruplanması gibi gizli yapıları arar ve bu ortaklıkların yeni veri parçalarında bulunup bulunmadığına bakar [11].

Denetimsiz öğrenmenin birçok uygulama alanı vardır, ancak boyut küçültme, istatistiklerde yoğunluk tahmini ve kümelemede yaygın olarak kullanılır.

2.2.3. Yarı denetimli öğrenme

Bu yöntem, öğrenme doğruluğunu arttırmak için denetimli öğrenmenin yüksek performansını ve denetimsiz öğrenmenin düşük maliyetini birleştirir. Eğitim veri kümesi hem etiketli hem de etiketsiz verilerden oluşmaktadır. Genellikle etiketlenmemiş verilerin miktarı, etiketleme işlemiyle ilişkili maliyet nedeniyle etiketli verilerden daha büyüktür. Yarı denetimli öğrenmenin uygulama alanları, sınıflandırma, tahmin ve regresyon içeren denetimli öğrenim ile aynıdır [12].

2.2.4. Güçlendirme öğrenmesi

Bu kategorideki algoritmalar diğer üç kategoride yer alanlardan farklıdır. Bu yöntem, Yazılım Ajanı, Çevre ve Eylemleri içeren üç temel unsurdan oluşur. Ajan, eylemlerle çevre ile etkileşime girer. Öğrenme, ajan (algoritmanın) gerçekleştirdiği her eylem için bir ödül veya ceza aldığı ve hangi eylemin en iyi ödülü verdiğini öğrenmesini sağlayan bir deneme yanılma sürecini içerir. Bu şekilde, ajan kendi politikalarını ve kurallarını öğrenip inşa edebilir. Güçlendirme öğrenme, özerk araçlarda, robotlarda ve insan rakipleriyle oynamayı içeren oyunlarda yaygın olarak kullanılmaktadır [13].

2.3. Anomali Tespiti

Anomali Tespiti, normal ve beklenen veri davranışının standart ve iyi tanımlanmış özelliklerine uymayan olağandışı modelleri ve gözlemleri tanımlamak için kullanılan tekniği ifade eder. İzinsiz giriş tespiti, sahtekarlık tespiti, arıza tespiti, sağlık izleme sistemleri ve sensör ağlarında olay yönetim sistemleri dahil olmak üzere birçok uygulama alanına sahiptir. Bu beklenmedik veya olağandışı gözlemler, farklı alanlarda farklı terminolojilere sahiptir, ancak genellikle anomaliler olarak bilinir [14]. Genellikle anomaliler üç geniş kategoriye ayrılır [15].

- Nokta Anomalisi: Tek bir veri noktası veya örnek, tüm veri kümesinden farklı özelliklere sahipse veya verilerin geri kalanından uzakta konumlanmışsa, bir nokta anomalisi olarak kabul edilir. Örneğin, rastgele bir günde harcanan miktara göre kredi kartı sahtekarlığının tespit edilmesi, bir nokta anomalisidir.
- Bağlamsal Anomali: Bir veri noktasının anormalliği, iyi tanımlanmış bir bağlamda veya belirli şartlar yerine getirildiğinde meydana gelir. Örneğin, bir insanın bayram günü ya da tatil mevsimi dışında bir zamanda yiyecek için yüksek miktarda para alması bağlamsal anomali olarak kabul edilir.
- Toplu Anomali: Birbirine benzer bir dizi veri örneği, normal veri ile karşılaştırıldığında toplu olarak anormal özelliklere sahipse, toplu anomali olarak kabul edilir. Toplu anomali, genellikle bir siber saldırıya işaret eder.

Bu anomali tiplerinin birbirleriyle ilişkili olduğunu not etmek önemlidir. Bir nokta anomalisi, belli bir kritere uyması halinde, bağlamsal bir anomali haline gelebilir. Ayrıca, bir araya getirildiğinde bir nokta anomalisi topluluğu da toplu anomaliye yol açar.

Anomali tespiti için kullanılan çeşitli teknikler vardır, ancak en popüler olanlar denetimli anomali tespiti, yarı denetimli anomali tespiti ve denetimsiz anomali tespitidir. Bu tekniklerin her birinin farklı alanlarda kendi ve özel kullanım durumları vardır. Örneğin, denetimsiz anomali tespiti, derin öğrenme ve veri madenciliği gibi alanlarda popülerlik kazanmıştır, ancak ağa izinsiz giriş tespiti durumunda işe yaramaz ve yerine denetimli anomali tespiti kullanılmalıdır [16].

Ağ istismarı ve izinsiz giriş tespiti bağlamında, anomalilerin tespiti çok zor olabilir. Bunu yaparken bazı zorluklar aşağıda listelenmiştir [17].

- Normal bölgeleri tanımlamak çok zordur. Çoğu durumda, anomaliler ve normal veriler arasındaki sınırlar kesin değildir. Bu durumda, normal gözlemler anomaliler olarak kabul edilebilir ve bunun tersi de geçerlidir.
- Çoğu zaman saldırganlar eylemlerini normal davranışa uyarlamaya çalışırlar ve yine, bu durumda anomalileri tanımlamak çok kolay değildir.
- Bugün normal kabul edilenler gelecekte normal olmayabilir. İş sistemlerinin zamanla çeşitli faktörlerin etkisiyle değişir.
- Model eğitimi için, eğitim ve doğrulama verilerinin kullanılabilirliği büyük bir sorundur.

Bir modelin eğitimi sırasındaki bu zorluklar nedeniyle, eğitim ve test için kullanılan veri kümesi çok yeterli olmalıdır. Modelin normal ve neyin iyi olduğunu tanımlayan daha geniş bir tanımına sahip olabilmesi için veri kümesi ağ saldırı tipleri açısından çeşitlilik göstermelidir. Veri kümesi ayrıca doğru bir şekilde etiketlenmelidir.

2.4. Ağ Saldırıları

Bilgisayar ağları bağlamında bir saldırı, verilerin, sistemlerin, uygulamaların, trafiğin veya ağın diğer herhangi bir unsurunun gizliliğini, bütünlüğünü ve kullanılabilirliğini (CIA Üçgeni) ihlal etmeye yönelik herhangi bir girişimdir [18]. Daha basit bir deyişle, kaynaklara yetkisiz erişim sağlama, değiştirme, tahrip etme, çalma veya kazanma girişimlerinin bir saldırı olduğu düşünülmektedir. Saldırılar, maddi zarar, hizmetin engellenmesi ve hatta hayat kaybına yol açabilir. Bilgisayar ağlarını korumak için izinsiz giriş tespit ve önleme sistemleri (IDS) gibi önlemler alınmalıdır. Ağ saldırıları genellikle aşağıdaki dört kategoriye girer.

- DoS / DDoS saldırıları: Bu kategorideki saldırılar genellikle meşru ağ kullanıcılarının hizmetlerine erişimi engellemek amacıyla gerçekleştirilir. Saldırı, genellikle hedef cihazları sistemlerinin aşırı tamamlanmamış isteklerinin yüklenmesi ile ağ kaynaklarının ve hizmetlerinin geçici veya süresiz olarak bozulmasına neden olmaktadır. DDoS saldırısında, saldırı, kaynağı engelleyerek saldırıyı durdurmayı imkansız kılan birçok kaynaktan kaynaklanır [19].
- Prob saldırıları: Bu kategorideki saldırılar, hedef hakkında bilgi toplamak veya bir sistemdeki güvenlik açıklarını keşfetmek için gerçekleştirilir. Her ne kadar saldırıları bilgisayar ağlarına zarar vermese de, bunun sonucunda ağın performansı üzerinde yıkıcı bir etki yaratabilecek diğer saldırı türleri için zemin hazırlıyor. IPsweep ve Portsweep, bu kategoriye giren en ünlü saldırı türlerindendir [20].
- U2R (User to Root) saldırıları: Bu kategoride saldırgan ya da normal kullanıcı, bilgi çalmak ya da zarar vermek amacıyla bilgisayar sistemlerinde yönetici hakları kazanmaya çalışır. Bu kategoride kullanılan teknikler genellikle Brute Force veya sistem açıklarını keşfetmektir [21].
- R2L (Remote to Local) saldırıları: Bu kategorideki bir uzak fail, bir ayrıcalık elde etmek veya mağdur bilgisayardan mesaj göndermek için Brute Force gibi teknikleri kullanarak veya sistem güvenlik açıklarından yararlanarak yerel ağa erişim kazanmaktadır [22].

Aşağıdaki bölümde, genellikle ağ anomalisi tespiti araştırmalarında kullanılan popüler veri kümelerinin bazıları incelenmiştir.

2.5. Eğitim Veri Kümeleri

Ağ anomalisi ve izinsiz giriş tespiti amacıyla makine öğrenme algoritmalarını eğitmek ve test etmek için büyük miktarda normal ve kötü niyetli ağ trafiğine ihtiyaç duyulur. Ancak, gizlilik sorunları nedeniyle, gerçek ağ trafiğinin kullanılması mümkün değildir. Bu sorunu çözmek için birçok veri kümesi oluşturulmuştur. İlerleyen bölümlerde, bu amaçla kullanılan birçok popüler veri seti ayrıntılı olarak incelenmiştir. Bu veri kümeleri, bu çalışma kapsamında kullanılması mümkün olan en iyi veri kümesini belirlemek için değerlendirilerek karşılaştırılır.

2.5.1. DARPA

DARPA, izinsiz giriş tespit sistemlerinin eğitimi ve test edilmesi için yaygın olarak kullanılan bir veri kümesidir. 1998 yılında DARPA tarafından sağlanan fonlarla birlikte MIT Lincoln laboratuvarı tarafından yaratılmıştır. Bu veri kümesinde, ağ trafiği Hava Kuvvetlerinin bilgisayar ağının bir simülasyonundan toplanmış olup 5 hafta eğitim verisi ve iki hafta test verisi içerir. Saldırı tespit sistemi başlangıçta Lincoln laboratuarında çevrimdışı olarak ve ardından Hava Kuvvetleri Araştırma Laboratuarı'ndaki (AFRL) gerçek zamanlı sistemler üzerinde test edilmiştir. Veri kümesi dört kategoriye giren 38 saldırı tipini içerir: DoS, U2R, Probe ve R2L. DARPA'98, yaklaşık 5 milyon bağlantı kaydında işlenebilen 7 haftalık ağ trafiğinin yaklaşık 4 gigabayt sıkıştırılmış ham (ikili) tcpdump verisidir. İki haftalık test verileri yaklaşık 2 milyon bağlantı kaydına sahiptir. İlk veri kümesi sadece binlerce simüle edilmiş UNIX sistemi içeriyordur. Ancak DARPA 1999 versiyonu, UNIX, Windows NT ve Cisco Router kurban sistemlerine karşı 57 saldırı örneği içermektedir [23].

DARPA veri kümeleri (1998, 1999 ve 2000), makine öğrenmesi ve ağ güvenliği araştırma topluluğundan yıllar boyunca birçok eleştiri almıştır. Çünkü bu veri kümesi, ağ saldırılarının gerçek özelliklerini doğru şekilde temsil edebilen gerçek ağ trafiği

değil, benzetilmiş ağ trafiğinin bir koleksiyonudur. Bir diğer önemli nokta, veri kümesinin gerçekten eski ve modası geçmiş olması, ve ağ kullanıcılarının davranışları ve ağ saldırılarının niteliğinin yaygın bir şekilde değiştiğidir. Bununla birlikte, DARPA veri kümesi ağ anomalisi tespiti için yeni bir geçit açması, KDDCUP 99 ve NSL-KDD gibi popüler ve çok yaygın kullanılan veri kümelerini oluşturmak için kaynak olarak kullanılmasından dolayı hala önemli bir değere sahiptir.

2.5.2. KDDCUP 99

KDDCUP 99, 1999'dan beri ağ anomalisi ve izinsiz giriş tespit araştırmasında en yaygın kullanılan veri kümelerinden biridir. Üçüncü Uluslararası Bilgi Keşfi ve Veri Madenciliği Araçları Yarışmasında Kaliforniya Üniversitesi Stoflo tarafından yaratılmıştır. DARPA 98 saldırı tespit değerlendirme araştırmasında elde edilen veriler kullanılarak inşa edilmiştir. KDDCUP 99 iki bölümden oluşur: eğitim veri kümesi ve test veri kümesi. Eğitim veri kümesinde, normal veya saldırı olarak etiketlenmiş 41 özellik içeren 4898431 veri akışı bulunur. Eğitim veri setindeki saldırılar DoS, U2R, Probe veya R2L saldırı tipleri altında sınıflandırılmıştır. Test veri kümesi 311029 veri akışına sahiptir ve eğitim veri kümesinde bulunmayan 14 saldırı tipini içerir. KDDCUP 99 veri kümesindeki toplam saldırı tiplerinin sayısını 38'e getiren eğitim veri kümesinde ek 24 saldırı tipi bulunmaktadır.

Yıllar boyunca KDDCUP 99 veri kümesinde, bazıları DARPA 1998 kullanımı nedeniyle bulunan ve bazıları bağımsız olan birçok eksiklik gözlemlenmiştir. KDDCUP 99 veri kümesindeki en önemli iki problem, kayıtların büyük miktarda tekrarlanması ve kayıtların zorluk seviyesidir. [24], eğitim ve test veri kümelerinin sırasıyla % 78 ve % 75 oranında tekrar kayıt içerdiğini açıklamaktadır.

2.5.3. NSL-KDD

NSL-KDD, 2009 yılında [24] tarafından oluşturulan KDDCUP 99 veri kümesinin geliştirilmiş ve rafine edilmiş bir versiyonudur. Bu veri kümesi, selefinin kayıt tekrarları ve zorluk seviyesi olan iki ana problemini çözmektedir. Ayrıca, KDDCUP

99 veri kümesinin boyutu oldukça büyüktür ve araştırmacılar veri kümesinin genel özelliklerini taşıyamayan veri kümesinden rastgele örnekleri seçmek zorunda kalmışlardır. NSL-KDD bu sorunu orijinal veri kümelerinin rastgele seçilmiş alt kümelerini oluşturarak ve bunu Eğitim ve Test olarak iki başlık altında dört bölüme ayırarak çözmektedir.

NSL-KDD, eğitim ve test kümelerinde, anormal tespit tekniklerinin karşılaştırılmasında kullanılacak bir referans veri kümesi olarak iyi bir seçim olmasını sağlayan makul sayıda kayda sahiptir. Bununla birlikte, veri kümesi hala mirasi sorunlardan muzdariptir ve mevcut bir gerçek ağı mükemmel bir şekilde temsil etmemektedir.

2.5.4. CAIDA

CAIDA, İnternet trafiği, performans, topoloji, yönlendirme ve güvenlikle ilgili olayların bilimsel analizi için verilerin toplanması ve paylaşılması konusunda temel amaçlara sahip bir kuruluş olan Uygulamalı İnternet Veri Analizi Merkezi anlamına gelir. Bu kuruluşun sağladığı veri kümeleri aynı adla bilinir. Bu veri kümesini oluşturan veriler San Jose şehrinin OC48'in omurga bağlantısının birkaç saatlik yakalanan ağ akışıdır. Bu veri kümesi ayrıca bir DDoS saldırısının saatlik simülasyonunu içemektedir [25].

CAIDA veri kümesi birden fazla saldırı senaryosuna sahip değildir ve içindeki veri akışları, örnekleme çeşitliliğini çok sınırlı hale getiren belirli saldırı tipleri ve özel uygulamalar ile örneklenmiştir. Ayrıca, veri kümesi de etiketlenmemiştir, bu da makine öğrenmesi için kullanılmasını oldukça zorlaştırmaktadır.

2.5.5. DEFCON

DEFCON, izinsiz giriş tespit uygulamalarında yaygın olarak kullanılan bir veri kümesidir. CTF (Bayrağı Yakala) hack ve anti-hack yarışmalar sırasında oluşan trafiği toplayarak yaratılır. CTF, ağ akışları yakalanan ve daha sonra veri kümesinin

oluşturulmasında kullanılan iki saldırgan ve savunma grubundan oluşur. DEFCON, kısıtlayıcı ortamdaki ağ akışları veya veri akışları toplamıdır; bu, yalnızca izinsiz ve rahatsız edici ağ trafiği içerdiğinden gerçek ağ akışlarından oldukça farklıdır. Bu nedenle DEFCON çoğunlukla alarm korelasyon teknikleri için kullanılır [25].

2.5.6. ISCX-UNB

Yıllar boyunca, birçok popüler ve standart veri kümesi ya güncelliğini yitirdikleri ya da bu veri kümelerini oluşturmak için kullanılan benzetilmiş verilerin gerçek dünya verilerine benzememesi nedeniyle topluluk tarafından eleştirilmektedir. Bu sorunları ele almak için, New Brunswick Üniversitesi'ndeki Bilgi Güvenliği Mükemmeliyet Merkezi (ISCX) 2012 yılında ISCX veri kümesini oluşturmuştur [25].

ISCX veri seti, test edilen Kanada Siber Güvenlik Enstitüsü'ndeki yakalanan 7 günlük internet akışını kullanarak oluşturulmuştur. Veri kümesi aşağıdaki olayları içermektedir.

- Gerçek bir ağ yapılandırması ve cihazları üzerinden inşa edilmiş olup, gerçek bir ağın olası anormal ve normal senaryolarını, bir grubun traffic üretmesiyle simüle edilerek oluşturulmuştur.
- HTTP, SMTP, FTP, SSH, POP3 ve IMAP protokolleri dahil olmak üzere gerçek normal ve zararlı akışları içermektedir.
- DoS, DDoS, Infiltration ve Brute Force SSH saldırıları dahil çok çeşitli saldırı tiplerine sahiptir.
- Toplam 2450324 kayda sahiptir.

ISCX uzun yıllar boyunca yararlı bir veri kümesi idi ve bu veri kümesi kullanılarak birçok araştırma yapılmıştır. Öte yandan, ISCX veri kümesinin temel dezavantajı, bugün internet trafiğinin % 50'sini oluşturan SSL / TLS trafiğinden yoksun olmasıdır, bu da bugünün gerçek ağ trafiğine artık yeterince benzeyemeyeceği anlamına gelmektedir [25].

2.5.7. CIC-IDS-2017

CICIDS, 2017 yılında New Brunswick Üniversitesi'ndeki Kanadalı Siber Güvenlik Enstitüsü tarafından oluşturulan izinsiz giriş tespit değerlendirme veri kümesidir. Bu veri kümesi, gerçek ağ trafiğini andıran hedefiyle, en yeni ve son zamanlarda oluşturulan izinsiz giriş tespit veri kümelerinden biridir. Veri yakalama süreci hem normal hem de saldırı trafiğini içeren beş gün (3-7 Temmuz 2017) sürmüştür [26].

CICIDS 2017 veri kümesinin oluşturulmasından önce, CIC ekibi kamuya açık ve en popüler izinsiz giriş tespit veri kümelerini değerlendirmiş olup, veri kümesi oluştururken izlenmesi gereken on kriteri belirlemişlerdir. Önceki veri kümelerinin hiçbiri on kriterin hepsini desteklememektedir. Bu kriterler CICIDS veri kümesinin avantajları olarak sayılır ve aşağıdakileri içerir [26].

- Komple ağ yapılandırması: Ağ topolojisi, Yönlendiriciler, Anahtarlar, Güvenlik Duvarı, Modem ve Windows, Mac OS X, Kali ve Ubuntu gibi çeşitli işletim sistemlerini içermektedir.
- Komple Trafik: Bir kullanıcı profili belirleme aracısından ve Victim-Network'teki 12 farklı makineden ve gerçek saldırılardan oluşmaktadır.
- Etiketli Veri Kümesi: Veri kümesi etiketleri dokümantasyonunda halka açık olan bilgileri içermektedir.
- Tam Etkileşim: LAN'lar iki farklı ağ ve Internet bağlantısı ile bağlanmaktadır.
- Komple Çekim: Tüm trafik ayna portu kullanılarak bir depolama ünitesinde yakalanmış ve kaydedilmiştir.
- Mevcut Protokoller: Veri kümesinde HTTP, HTTPS, FTP, SSH ve e-posta protokolleri gibi ortak protokoller bulunmaktadır.
- Saldırı Çeşitliliği: 2016 McAfee raporuna dayanarak, Web tabanlı, Brute force, DoS, DDoS, Infiltration, Heart-bleed, Bot ve Scan gibi en yaygın ve en son saldırılar bu veri kümesinde bulunmaktadır.

- Heterojenite: Saldırıların gerçekleştirilmesi sırasında ana şalterden ağ trafiği, hafıza dökümü ve tüm kurban makinelerden yapılan sistem çağrıları yakalanmıştır.
- Özellik Seti: Veri kümesi, ağ akışı veri kümesinde CSV dosyaları olarak 86 özellik içermektedir.
- Meta Veriler: Dokümantasyon, zaman, saldırı, akış ve etiketleri içeren veri kümesini tamamen açıklamaktadır.

2.5.8. CSE-CIC-IDS-2018

CICIDS-2018 AWS ortamında, ortak bir proje olarak İletişim Güvenlik Kuruluşu (CSE) ve Kanada Siber Güvenlik Enstitüsü (CIC) tarafından üretilen veri kümesidir. Bu veri kümesi, saldırılarda, özelliklerin ve örneklerin sayısında birkaç değişiklikle CICIDS-2017 veri kümesiyle aynı özellikleri taşımaktadır. Bu veri kümesi, bir ağda görülen olayları ve davranışları temsil eden ağ kullanıcı profillerine dayanan sistematik bir şekilde üretilmiştir. Veri kümesi ayrıca en son saldırılar dahil olmak üzere anomali saptamadaki en yeni eğilimleri içermektedir [27]. Veri kümesi aşağıdaki gibi kısaca tanımlanabilir.

- Veri kümesi dinamik olarak oluşturulmuş olup, değiştirilebilir, yeniden üretilebilir ve genişletilebilmektedir. Aynı zamanda en yeni trafik kompozisyonlarını ve izinsiz girişleri yansıtmaktadır.
- Saldırı altyapısında 50 makine bulunurken, mağdurlar toplamda 30 sunucu ve 420 makineden oluşan beş bölümden oluşmaktadır.
- Veri kümesinde yer alan saldırılar FTP-BruteForce, SSH-Bruteforce, DoS-GoldenEye, DoS-Slowloris, DoS-SlowHTTPTest, DoS-Hulk, DDoS attacks-LOIC-HTTP, DDoS-LOIC-UDP, DDOS-HOIC, Brute Force Web, Brute Force XSS, SQL Injection, Infiltration ve Bot'tur.
- Veriler üç ay boyunca 10 farklı günde toplanmıştır.
- CICFlowMeter kullanarak veri kümesinin yapımı için 80 özellik çıkarılmıştır.
- HTTP, HTTPS, FTP, SSH, SMTP, POP3 ve IMAP gibi çok çeşitli protokolleri içermektedir.

- Çok çeşitli Windows ve Linux makinelerini içermektedir.
- IPv6 trafiği içermiş olup, TTLS desteği vardır.
- Toplanan veriler gerçek dünya trafiğini andırıyor ve ağ anomalisi tespiti bağlamında kullanılmak üzere uygun bir veri kümesidir.

2.5.9. Veri kümeleri değerlendirmesi

Önceki bölümde, halka açık tüm izinsiz giriş tespit değerlendirme veri kümelerinin en yaygın kullanılan ve popüler olanları, artıları ve eksileri ile incelenmiştir. Tablo 2.1. tüm bu veri kümeleri kısaca karşılaştırmaktadır.

Tablo 2.1. a- Gerçek Ağ Yapılandırması, b- Gerçek Ağ Trafiği, c-Etiketli, d- Özellikler sayısı, e- Heterojen, f-Meta Veri Mevcut

								Protokol Çeşitliliği					
Veri Kümesi	a	b	c	d	e	f	Saldırı Çeşitliliği	HTTP	HTTPS	FTP	SSH	Mail	IPv6
DARPA	Evet	Hayır	Evet	NA	Hayır	Evet	Tüm saldırılar	Evet	Hayır	Evet	Evet	Evet	Hayır
KDDCUP-99	Evet	Hayır	Evet	41	Hayır	Evet	Tüm saldırılar	Evet	Hayır	Evet	Evet	Evet	Hayır
NSL-KDD	Evet	Hayır	Evet	41	Hayır	Evet	Tüm saldırılar	Evet	Hayır	Evet	Evet	Evet	Hayır
CAIDA	Evet	Evet	Hayır	NA	Hayır	Evet	Kısmi	-	-	-	-	-	-
DEFCON	Hayır	Hayır	Hayır	NA	Hayır	Hayır	Kısmi	Evet	Hayır	Hayır	Evet	Hayır	Hayır
ISCX-UNB	Evet	Evet	Evet	NA	Evet	Evet	Tüm saldırılar	Evet	No	Evet	Evet	Evet	Hayır
CICIDS-2017	Evet	Evet	Evet	86	Evet	Evet	Tüm saldırılar	Evet	Evet	Evet	Evet	Evet	Evet
CSE-CICIDS-2018	Evet	Evet	Evet	80	Evet	Evet	Tüm saldırılar	Evet	Evet	Evet	Evet	Evet	Evet

Bu veri kümelerinin dikkatli bir şekilde değerlendirilmesinden, avantaj ve dezavantajlarının araştırılmasından sonra CICIDS-2017 ve CSE-CIC-IDS-2018 bu

çalışma bağlamında en uygun seçenekler olarak görünmektedir. Bu karardaki en önemli faktörler şunlardır:

- Veri kümelerinin güncel olması ve gerçek ağın normal ve saldırı trafiğe benzemesi.
- Protokol ve saldırı havuzlarının çok çeşitli olması.
- Diğer veri kümelerinden farklı olarak, bu veri kümeleri kullanılarak pek fazla araştırma yapılmamıştır, bu nedenle bu çalışma literatüre iyi bir katkı olacaktır.

CICIDS-2017 ve CSE-CIC-IDS-2018 veri kümeleri nispeten yenidir ve muhtemelen bazı küçük eksiklikler içermektedir. Veri kümelerindeki problemler ve bu problemlerin nasıl çözüleceği konusundaki yöntemler 3. bölümde ele alınmaktadır.

2.6. Literatür Taraması

Ağ saldırı tespiti, özellikle anomali tespiti, alanında yapılan çalışmaları daha iyi anlayabilmek için bu çalışma süresince literatürde bulunan toplam elli çalışma incelenmiş olup analiz edilmiştir. Bu çalışmalar, Tablo 2.2.'de kısaca özetlenmişti

Tablo 2.2. İncelenen çalışmaların özeti.

NO	Tip	Algoritmalar	Öğrenme Yöntemi	Veri Kümesi	Değerlendirme Ölçütleri	Özellik Seçimi	Platform	Yorumlar
[28]	Karşılaştırma	J48, Decision Table, SVM, NB	Denetimli	KDDCUP-99	GPO, YPO, Hassasiyet, Doğruluk	-	WEKA	J48 en yüksek ve NB en düşük doğruluğa sahiptir.
[29]	Karşılaştırma	J48, NN-RBF, SVM-SMO	Denetimli	KDDCUP-99	Doğruluk	4 seçim yöntemi ile birleştirilen 3 özellik seçimi	WEKA	J48, diğer iki algoritmadan daha iyi performans gösterir. J48'in En Yüksek doğruluğu: 23-sınıflı: %99,9439 5-sınıflı: %99,9597 2-sınıflı: %99,969
[30]	Yeni ve karşılaştırma	Hierarchical Agglomerative Clustering, MLP, KNN	İki aşamalı: Denetimsiz + Denetimli ve Güçlendirme	NSL-KDD	Doğruluk	-	WEKA	Doğruluk: Tüm %99,75, MLP %99,95, KNN %99,94
[31]	İnceleme	J48, SMO, Winnow, NX, AIN+RBF, FC-ANN, FSVM and PLSSVM	Denetimli tek ve melez	-	Doğruluk	-	-	Doğruluk: J48% 99, SMO %99, Winnow %99, NX %94, AIN+RBF %97,28, FC-ANN %96,71, FSVM %91,21
[32]	Araştırma	ELM(SLFN-NN)	Denetimli	NSL-KDD	Algılama Oranı (GPO), YPO, ROC, F-ölçütü	-	-	30 nöron ve Gauss aktivasyon fonksiyonu ile SLFN-NN. Her saldırı için AO: DoS %86,89, Probe %99,06, U2R %99,99, R2L %99,94

Tablo 2.2. (Devam)

NO	Tip	Algoritmalar	Öğrenme Yöntemi	Veri Kümesi	Değerlendirme Ölçütleri	Özellik Seçimi	Platform	Yorumlar
[33]	Yeni ve karşılaştırma	NB, C4.5	İki Seviyeli Denetimli: Makro ve Mikro Seviyeler	KDDCUP-99	GPO, YPO, Ortalama GPO, Ortalama YPO	Info. Gain, Chi-Squared, Gain Ratio	WEKA	Makro level GPO: C4.5 %99, NB %98,6, Overall C4.5+Info. Gain %99,9. Mikro Level: C4.5+Info. Gain en iyi sonuçlara sahiptir.
[34]	Karşılaştırma	CNN: Shallow, Moderate ve Deep	Derin Öğrenme	NSL-KDD, Kyoto Honeypot and MAWILab	-	-	Tesorflow ve Keras	Shallow CNN diğer ikisinden daha iyi performans gösterir.
[35]	Araştırma	KNN, NB	Denetimli	Kasetsart University of Thailand	Hassasiyet, Geri Çağırma, F-ölçütü	-	-	Amaç, ağ trafiğinin aralık temelli özellikleri ile farklı anomaliler arasındaki ilişkiyi incelemektir. KNN NB'den daha iyi performans gösteriyor.
[36]	Yeni	K-means clustering	Yarı denetimli	NSL-KDD	Doğruluk	-	-	Doğruluk: %80,119
[37]	Araştırma ve karşılaştırma	Ensemble Learning, LR, CART, MLP	Denetimli Ensemble (WMV)	KDDCUP-99	Doğruluk	-	-	Test Doğruluğu: Ensemble %96,14, LR %96,13, CART %91,66, MLP %89,83.
[38]	Karşılaştırma	Ensemble Learning, RBF, KNN, NB, SVM, K-means, FCM	Denetimli ve Ensemble	Kyoto 2006+	Doğruluk, Hassasiyet, Geri Çağırma, ROC	-	Java ve R	Doğruluk: RBF %97,54, KNN %97,54, Ensemble %96,72, NB %96,72, SVM %94,26, K-means %83,6, FCM %83,6.

Tablo 2.2. (Devam)

NO	Tip	Algoritmalar	Öğrenme Yöntemi	Veri Kümesi	Değerlendirme Ölçütleri	Özellik Seçimi	Platform	Yorumlar
[39]	Karşılaştırma	RF, MLP(SDG), NB, DT	Denetimli	NSL-KDD	Doğruluk, Hata Algılama, Zaman	Correlation Ranking, Gain Ratio	WEKA	CRF ile Doğruluk: RF %99,32, MLP %96,75, NB %89,35, DT %87,72. GRFE ile Doğruluk: RF %99,77, MLP %96,52, DT %92,21, NB %89,08.
[40]	Yeni ve karşılaştırma	MLP, GAU, K-means, NEA, IRBF, LEA, HYP, Fuzzy ARTMAP, C4.5, MLP+GAU+ K-Means	Denetimli, Denetimsiz, olasılıksal, istatistiksel, bulanık-nöro sistemler	KDDCUP-99	Doğruluk	-	-	Hiçbir saldırı kategorisi için tek bir algoritma tatmin edici bir DP ve düşük YAO elde edemez ve belirli algoritmalar bir saldırı kategorisi için diğerlerine göre daha iyi performans gösterir. Her saldırı kategorisi için üstün algoritmalar arasında Prob için MLP, DoS ve U2R için K-M ve R2L için GAU bulunur. MLP + GAU + K - Çok sınıflandırıcı model daha iyi sonuç gösterir.
[41]	Yeni	FB-VQ	Denetimli	-	Algılama Oranı, YAO	FB algorithm	-	Yazarlar, daha yüksek AO, daha düşük YAO ve daha iyi gerçek zamanlı yetenek sağlayabilen Fraktal Teknolojisine ve Vektör Miktarlandırmasına dayanan ağ anomalisi tespiti önermektedir.

Tablo 2.2. (Devam)

NO	Tip	Algoritmalar	Öğrenme Yöntemi	Veri Kümesi	Değerlendirme Ölçütleri	Özellik Seçimi	Platform	Yorumlar
[42]	Yeni ve karşılaştırma	SVM, DT, NB	Denetimli	Customary	Doğruluk	10-fold cross validation: A+B and C+D feature groups	Apache (Hadoop, Kafka ve Storm), WEKA, HDFS	Üç sınıflandırıcının tamamı C + D özellik grubu ile daha iyi doğruluk sağlar. En iyi doğruluk: SVM %99,90 switch 3, DT %99,85 switch 3, NB %99,3 switch 4.
[43]	Araştırma	NB+AdaBoost	Denetimli - Melez	KDDCUP-99	Algılama Oranı, YPO	-	Matlab	Eğitim seti AO: %96,32. Test seti AO: %84,32.
[44]	Karşılaştırma	J48, RF, NB Tree, MLP, NB.	Denetimli	NSL-KDD: 3 subsets	Doğruluk, Hassasiyet, Geri Çağırma, F-ölçütü, AUC, Zaman	Consistency Subset Eval and CFs Subset Eval	WEKA	1 and 2 veri kümeleri: J48 diğerlerinden daha iyi performans gösteriyor 3 veri kümesi: RF diğerlerinden daha iyi performans gösteriyor.
[45]	Karşılaştırma	RF, C4.5, SVM, NB	Denetimli	UNSW-NB15	Doğruluk, Duyarlılık, Özgünlük, Zaman.	-	Apache Spark ve MLlib	Doğruluk: RF %97,49, C4.5 %95,82 , SVM %92,28, NB %74,19.
[46]	Karşılaştırma	LR, RF, MLP, NB, GB Trees, SVM	Denetimli: Çok sınıflı ve İkili sınıflandırma	NSL-KDD, KDDCUP-99	Doğruluk, Hassasiyet, Geri Çağırma, Zaman	Chi-Square	Apache Spark, Hadoop Yarn, Java API	Doğruluk: Çok sınıflı / İkili KDDCUP-99: LR %99,82, RF %99,39, MLP %99,00, NB %91,68. / GB Trees %99,98, SVM %99,64. Çok sınıflı / İkili NSL-KDD: RF %90,97, LR %81,27, NB %42,70 / GB Trees %98,72, SVM %15,5.

Tablo 2.2. (Devam)

NO	Tip	Algoritmalar	Öğrenme Yöntemi	Veri Kümesi	Değerlendirme Ölçütleri	Özellik Seçimi	Platform	Yorumlar
[47]	Karşılaştırma	RF, DL-H2O, LR, SVM, DL-4J, NB	Denetimli ve Derin Öğrenme, Binom ve Multinomial model.	NSL-KDD	Doğruluk, Hassasiyet, Geri Çağırma, F-ölçütü, ROC AUC, Zaman.	5-fold cross validation	AWS, WEKA, R.	Eğitim Setinde Binom Doğruluk: RF ve H20 %99,5'in üzerinde, NB %90,34. Eğitim + Test seti Binom Doğruluk: H2O %83,87, RF %80,25, SVM en zayıf olanıdır. Eğitim Setinde Multinomial Doğruluk: hepsi %99'un üzerinde, NB %93,66. Eğitim + Test seti Multinomial Doğruluk: LR ve H2O %85, SVM %58.
[48]	İnceleme	DNN, GA, NB, FL, RF, DT, KNN, SVM, ANN	Denetimli	KDDCUP-99	Doğruluk, Algılama Oranı, YAO	-	-	Doğruluk: DNN %99,5, GA %97,04, NB %94,9, FL %93,45, RF %92,58, DT %92,05, KNN %90,28, SVM %86,79, ANN %77.
[49]	Karşılaştırma	NN, SVM, DT, RF, NB, K-means.	Denetimli ve denetimsiz	KDDCUP-99, NSL-KDD, UNSW-NB15	Ağırlıklı F-ölçütü	SMOTE oversampling ve random undersampling	-	Sonuçlar UNSW-NB15-SMOTE'nin diğer iki veri setinden daha iyi sonuçlar elde edebileceğini ve literatürde kullanılan geleneksel veri setlerinin yerini alabileceğini göstermektedir.
[50]	Araştırma ve Karşılaştırma	C4.5+KNN, C4.5, KNN	Denetimli - Melez	NSL-KDD	Doğruluk, AO, YAO, Geri Çağırma	Info. Gain	-	Doğruluk: Melez %99,6, C4.5 %99,5, KNN %99,2.

Tablo 2.2. (Devam)

NO	Tip	Algoritmalar	Öğrenme Yöntemi	Veri Kümesi	Değerlendirme Ölçütleri	Özellik Seçimi	Platform	Yorumlar
[51]	Araştırma ve Karşılaştırma	KNN, K-means	Denetimli ve denetimsiz	CIDDS-001	Doğruluk, Algılama Oranı, YPO, Hassasiyet, F-ölçütü.	-	WEKA	Harici Sunucu için KNN Doğruluk: K = 5 ile en düşük %99,3 ve K = 2 ile en yüksek %99,6. Openstack sunucusu için KNN Doğruluk: K'nın tüm değerleri için %100. K-means Clustering YPO: harici sunucu %61,9 ve openstack sunucu %0,64.
[52]	Yeni ve Karşılaştırma	ELM, MLP, KNN, SVM, DT, LR	Denetimli	ISCX-2012, UNSW-NB15 Jan, UNSW-NB15 Feb, ISCX-2017, ve MAWILab-2018.	Algılama Oranı	Chi2, F-Score, SVMonline ve RFE.		- KNN, UDP için SVM'den daha iyi performans gösteriyor. - DT, TCP için LR'den daha iyi performans gösterir. - MLP normal, bilinmeyen ve diğer ataklar için ELM'den daha iyi performans gösterir. - SVMonline ile SVM, DoS saldırıları için en iyisidir ve SVMonline ile DT, diğer saldırı türlerini tespit etmek için en iyisidir. Platform için seçilirler.
[53]	Yeni	GA+SVM	Denetimli - Melez	KDDCUP-99	Doğruluk	GA-optimization	-	GA özellik optimizasyonu, SVM'nin performansını ve sınıflandırma doğruluğunu büyük ölçüde geliştirir.

Tablo 2.2. (Devam)

NO	Tip	Algoritmalar	Öğrenme Yöntemi	Veri Kümesi	Değerlendirme Ölçütleri	Özellik Seçimi	Platform	Yorumlar
[54]	Yeni	C5.0+LS-SVM	Denetimli - Melez	KDDCUP-99, UNSW-NB15	Doğruluk, GPO, YPO, ROC AUC	Aşama 1: Filter-based C5.0 ve Error Pruning, Aşama 2: Wrapper-based	LS-SVM Toolbox	KDDCUP 99 ile Doğruluk: 5 sınıfta %94,35 ile %99,89 arasında. UNSW-NB15 ile Doğruluk: 10 sınıfta %87,47 - %99,46 arasında.
[55]	Yeni	K-means+RF, K-means+DT	Yarı- denetimli	ISCX-2012	Doğruluk, Hassasiyet, Geri Çağırma, F1-ölçütü, YPO	Info. Gain	Apache Spark	Doğruluk: K-means+RF %99,5, K-means+DT %93,5.
[56]	Yeni ve karşılaştırma	K-M+ ID3,K-M+SVM,K-M+NB vs ID3, C4.5,NB, KNN ,SVM ve RF	Yarı- denetimli	NSL-KDD	GPO, YPO	Spearman Correlation	-	Spearman Correlation, R2L ve U2R saldırılarını tespit etmek için melez yöntemlerin GPO'sunu geliştirir.
[57]	Yeni	SVM, KNN, KNNGA	Denetimli: multitier	KDDCUP-99	Doğruluk, Duyarlılık, Özgünlük.	ID3+Info. Gain	Matlab	Saldırıların doğruluğu: DoS %99,48, Probe %98,27, U2R %98,48, R2L %99,96. - Sonuçlar, KNN'nin U2R için %99,49 sahip olduğu hariç, tüm durumlarda SVM ve KNN'den daha iyidir.
[58]	Araştırma ve karşılaştırma	Base learner: C4.5 Ensembles: Bagged Trees, GentleBoost, AdaBoost, LogitBoost ve RustBoost	Denetimli: Ensemble Learning	UNSW-NB15	Doğruluk, ROC AUC	-	Matlab	- Bagged Trees ve GentleBoost en iyi doğruluğu elde eder. - RustBoost en kötü performansa sahiptir.

Tablo 2.2. (Devam)

NO	Tip	Algoritmalar	Öğrenme Yöntemi	Veri Kümesi	Değerlendirme Ölçütleri	Özellik Seçimi	Platform	Yorumlar
[59]	Araştırma ve karşılaştırma	K-means+RF vs RF, DT, KNN, NB, SVM and NN	Yarı-denetimli	ISCX-2012	Doğruluk, Algılama Oranı, YAO.	-	-	ISCX Uygulamaları için K-means+RF Doğruluğu: HTTPWeb %99,91, SSH %99,98, ICMP %100, FTP %99,97, DNS %99,99. - RF ve DT modelden sonra en yüksek hassasiyete sahiptir. - Saldırı doğruluğu hakkında bilgi yok.
[60]	Karşılaştırma	J48, ID3, NB	Denetimli	KDDCUP-99	Doğruluk, Hata Oranı, Zaman	10-fold cross validation	WEKA	Doğruluk: J48 %96,53, ID3 %96,53, NB %96,44.
[61]	Araştırma	PCA+CART	Denetimli	KDDCUP-99	-	PCA	SKL	PCA ile boyut azaltma, CART'ın performansını ve algılama doğruluğunu arttırır ve önerilen model hem ID3 hem de C4.5'ten daha iyi performans gösterir.
[62]	Araştırma ve karşılaştırma	Base Classifier: SVM Stacked Classifiers: RF, SimpleCART, J48, IBK (KNN), AdaBoost, BayesNet, LR, JRip and OneR.	Denetimli: Melez	NSL-KDD	Doğruluk, Duyarlılık, Özgünlük, Hassasiyet, Geri Çağırma, ROC AUC.	-	WEKA	Doğruluk: SVM+RF %97,50, SVM+SimpleCART %97,49, SVM+JRip %97,21, SVM+J48 %97,12, SVM+IBK %95,79, SVM+LR %92,58, SVM+BayesNet %92,14, SVM %91,81, SVM+OneR %91,71 ve SVM+AdaBoost %90,53.

Tablo 2.2. (Devam)

NO	Tip	Algoritmalar	Öğrenme Yöntemi	Veri Kümesi	Değerlendirme Ölçütleri	Özellik Seçimi	Platform	Yorumlar
[63]	Karşılaştırma	K-means+DT	Yarı-denetimli	KDDCUP-99	Hassasiyet, Geri Çağırma.	Online feature selection	-	Yarı-denetimli: %98,38 ve %96,28. Denetimsiz: %92,83 ve %87,03.
[64]	Yeni ve karşılaştırma	RF, Linear Regression	Denetimli: Kademeli çok katmanlı kategorizasyon	UNSW-2015	YAO, UND, Hata Oranı, Doğruluk	Best First	-	- RF, LR'den daha iyi performans gösterir. - Sınıflandırma Doğruluğu: Kademeli %93,35, tek-tip %80.
[65]	Araştırma ve karşılaştırma	RBF, MLP, Ensemble of MLP ve RBF	Denetimli: Ensemble learning	Customary	Doğruluk	-	-	Doğruluk: MLP için %98,88 normal ve %93,93 anormal, RBF için %94,20 normal ve %99,02 anormal, Ensemble modelinde MLP için %98,88 normal ve %94,31 anormal, ve Ensemble modelinde RBF için %94,21 normal ve %99,03 anormal.
[66]	Karşılaştırma	RF, LR, NB, SVM	Denetimli	NSL-KDD	Doğruluk, Hassasiyet, Geri Çağırma, F1-ölçütü, GPO ve YPO	-	SKL	Doğruluk: RF %99, LR %84, NB %79, SVM %75.
[67]	Karşılaştırma	TCM-KNN vs SVM, NN ve KNN	Denetimli: Aktif Öğrenme	KDDCUP-99	Doğruluk, YPO.	Chi-square, SVM Attribute Eval.	WEKA	Orijnal Set Doğruluğu: NN %99,8, TCM-KNN %99,7, SVM %99,5, KNN %99,2. Azaltılmış Set Doğruluğu: TCM-KNN %99,6, SVM %98,7, NN %98,3, KNN %97,7.

Tablo 2.2. (Devam)

NO	Tip	Algoritmalar	Öğrenme Yöntemi	Veri Kümesi	Değerlendirme Ölçütleri	Özellik Seçimi	Platform	Yorumlar
[68]	Yeni ve karşılaştırma	SSAD, NB	Yarı-denetimli İstatistiksel yaklaşım, Denetimli	NSL-KDD, Kyoto-2006+	GPO, YPO	-	-	- Modelin iki aşaması var.-YPO'yi azaltır. - NB'den daha iyi performans gösteriyor.
[69]	Yeni ve karşılaştırma	Fuzzy-based SLFN, J48, NB, NB Tree, RF, RTrees, SVM and MLP	Fuzzy tabanli Yarı-denetimli	NSL-KDD	Doğruluk	-	-	Doğruluk: KDDTest^{+} ile %84,2 ve KDDTest^{-21} ile %68,82. - J48, NB, NB Tree, RF, RTrees, SVM ve MLP'den daha iyi performans gösteriyor.
[70]	Yeni	NB, RF, KNN, MLP, SVM and ICO.	denetimli ve denetimsiz mesafeye dayalı Laplacian Eigenmap temelli doğrusal metrik öğrenme yaklaşımı.	NSL-KDD, Kyoto-2006+	GPO, YPO, YNO, F-ölçütü, CCI	-	-	Sonuçlar, önerilen metrik öğrenme yaklaşımının NB, RF, KNN, MLP, SVM ve ICO ile birlikte kullanıldığında performanslarını büyük ölçüde geliştirdiğini göstermektedir.
[71]	Araştırma, yeni ve karşılaştırma	hw-IBK, IBK, LWL	Denetimli: Tembel Öğrenme Algoritmaları	NSL-KDD	GPO, YPO, Hassasiyet, F-ölçütü, ROC AUC	-	WEKA	GPO: hw-IBK %97,6, IBK %92,3, LWL %90,7.
[72]	Yeni ve karşılaştırma	Non-linear invariant based LR vs RF, MLP, NB, C4.5 and SVM.	Denetimli: internet trafiğinin doğrusal olmayan değişmez özelliklerine dayalı.	ISCX-UNB, Tcpreplay replayed ISCX-UNB, ISCX+CAIDA 2007	Doğruluk, Özgünlük, Duyarlılık, Hassasiyet, F-ölçütü, AO…	Filtering+backward elimination wrapping	-	Doğruluk: RF %99,948, MLP %99,897, Önerilen metod %99,846, NB %99,795, C4.5 %99,795 ve SVM %97,801.

Tablo 2.2. (Devam)

NO	Tip	Algoritmalar	Öğrenme Yöntemi	Veri Kümesi	Değerlendirme Ölçütleri	Özellik Seçimi	Platform	Yorumlar
[73]	Yeni ve karşılaştırma	K-means+C4.5 vs K-means, SVM, ID3, NB, TCM-KNN and KNN.	Denetimli-Melez, Denetimsiz	KDDCUP-99	GPO, YPO, Hassasiyet, Doğruluk, F-ölçütü, ROC AUC	ANN based non-linear Component Analysis	WEKA	Doğruluk: K-means+C4.5 %95,8, TCM-KNN %95,7, SVM %95,5, NB %93,2, KNN %93, ID3 %93, K-means %89,4.
[74]	Yeni ve karşılaştırma	NeuroTree vs C4.5, RF, REP Tree, Random Tree ve Decision Stump	Denetimli: Decision Tree tabanlı hafif saldırı tespit	KDDCUP-99	Algılama Oranı, Hata Oranı.	Wrapper Approach: GA+NeuroTree	WEKA	AO: NeuroTree %98,38, C4.5 %92,27, RF %89,21, REP Tree %89,11, Random Tree %88,98, Decision Stump %79,73.
[75]	Yeni ve karşılaştırma	Multilevel SVM+ELM vs Multilevel SVM ve Multilevel ELM.	Yarı-denetimli: Hibrit ve çok seviyeli	KDDCUP-99	Doğruluk, Algılama Oranı, YAO	K-means Clustering	Java	Doğruluk: Önerilen metod %95,75, M-SVM %95,57, M-ELM %93,83.
[76]	Yeni	Level 1: CART rule base. Level 2: SVM, NN ya da NB Level 3: iPCA	Denetimli: Çok Düzeyli	NSL-KDD	Doğruluk	Discrete Wavelet Transform	-	Modelin bireysel Sınıflandırıcılar ile Doğruluğu: NN %96,67, SVM %95,55, NB %89,03. - SVM en uygun sınıflandırıcı olarak seçilmiştir.
[77]	Yeni	SVM ve KNN'nin Ensemble'i	Denetimli: Altı SVM ve altı KNN Ensemble'i	KDDCUP-99	-	-	-	12 uzmanın sonuçları PSO-WMV, PSO-LUS ve WMA kullanılarak birleştirilmiştir.

[29] Neural Network (RBF), SVM (SMO) ve Karar Ağaçları (J48) makine öğrenmesi algoritmalarının performansını ağ anomalisi tespiti bağlamında değerlendirip karşılaştırmaktadır. Çalışmada kullanılan 22 saldırı içeren %10 KDD99 veri kümesidir. Daha yeterli ve pratik sonuçlar elde edilmesi için, üç özellik değerlendirme yöntemi ve WEKA tarafından özellik seçimi için sağlanan dört arama yöntemi kullanılmaktadır. Bu yöntemler kullanılarak, deney için 23 sınıflı, 5 sınıflı ve 2 sınıflı üç senaryo oluşturulmuştur. İlk olarak, algoritmalar %10 KDD99 veri kümesinin rastgele seçilen 98804 örnekleri üzerinde değerlendirme ve arama yöntemlerinin 2 sınıflı ve 5 sınıflı kümeler üzerinde test edilmiştir. Her üç algoritma da %94'ün üzerinde bir yüksek doğruluk elde etmesine rağmen de, J48 diğer ikisinden daha iyi performans göstermektedir. Daha sonra J48 algoritması, tam %10 KDD99 veri kümesine uygulanır ve BestFirst + Cfssu bsetEval ile % 99,95 en yüksek doğruluğu elde etmektedir.

[38] altı makine öğrenmesi tekniğinin performansını Kyoto 2006+ veri kümesi ile inceliyor ve tüm altı algoritmanın sonuçlarının birleşimine ve çoğunluk oylamasının nihai tahmin için kullanılmasına dayanan bir Ensemble öğrenme yöntemi önermektedir. Altı makine öğrenme algoritması KM, FCM, KNN, NB, SVM ve RBF'yi içerir. Performans değerlendirme ölçütleri; hassasiyet, geri çağırma, doğruluk ve Receiver Operating Curve (ROC)'yi içerir. Hassasiyet, geri çağırma, doğruluğ’a dayalı sonuçlara göre, KNN tarafından takip edilen RBF en iyi performansı elde eder ve FCM ile KM çok düşük recall değeri nedeniyle en kötü performansa sahiptir. Bununla birlikte, eğer başlangıç merkezlerini seçmek için geliştirilmiş algoritmalar ile birlikte ikiden fazla küme seçilirse sonuçları değişebilir. Bununla beraber, bu çalışmada önerilen değerlendirme ölçütü, RBF'nin 0,9741 ROC değeri ile en iyi performansı gösterdiği, ardından 0,9639 ROC değeri ile ensemble öğrenmesinin gerçekleştirdiği ROC'dir. Yazarlar, ensemble öğrenmenin bireysel algoritmalardan daha iyi çalışabileceğini ve bu alanda daha fazla araştırma önerdiğini öne sürmektedir.

[39] dört makine öğrenmesi algoritmasının, NSL-KDD veri kümesi üzerinden ağ izinsiz giriş tespiti bağlamında, Correlation Ranking Filter (CRF) ve Gain Ratio Feature Evaluator (GRFE) dahil olmak üzere iki farklı özellik seçim yöntemi ile

performansını analiz eder. Bu çalışmada kullanılan MÖ algoritmaları Random Forest, Decision Tree, Naïve Bayes and Stochastic Gradient Descent'dir. Her algoritmanın performansını her iki özellik seçim yöntemiyle birlikte değerlendirirken, doğruluk, algılama hatası, hassasiyet, geri çağırma, F-ölçütü ve eğitim süresi gibi birçok ölçüm metriği kullanılır. Sonuçlar, Random Forest'in tüm algoritmalar arasında en iyisini yaptığını ve sırasıyla CRF ve GRFE ile %99,3172 ve %99,7618 doğruluk elde ettiğini göstermektedir. Sonuçlar ayrıca, tüm algoritmaların performans ve eğitim sürelerinin GRFE özellik seçim yöntemiyle geliştiğini göstermektedir.

[40], 9 farklı makine öğrenmesi algoritmasının KDDCUP 99 veri kümesi üzerindeki anomli tespiti bağlamında, performansını karşılaştırmaktadır. Bu algoritmalar, sinir ağları, olasılık modelleri, istatistiksel modeller, bulanık nöro sistemler ve karar ağaçları dahil olmak üzere çok çeşitli alanları temsil eder. Algoritmalar arasında Multilayer Perceptron (MLP), Gaussian Classifier (GAU), K-means clustering (K-M), Nearest clustering algorithm (NEA), Incremental RBF (IRBF), Leader algorithm (LEA), Hypersphere algorithm (HYP), Fuzzy ARTMAP ve C4.5 bulunmaktadır. Çalışmanın sonuçları, hiçbir algoritma saldırının tatmin edici bir tespit olasılığı ve tüm saldırı kategorileri (Probe, DoS, U2R, R2L) için düşük Yanlış Alarm Oranı (YAO) elde edemediğini ve bazı algoritmaların belirlenmiş bir saldırı kategorisi için diğerlerine göre daha iyi performans gösterdiğini sunmaktadır . Buna göre, her saldırı kategorisi için üstün algoritmalar arasında Prob için MLP, DoS ve U2R için K-M ve R2L için GAU bulunur. Sonuçlar ayrıca, neredeyse tüm algoritmaların Probe ve DoS saldırıları için iyi sonuçlara sahip olduğunu, ancak hiçbirinin U2R ve R2L için tatmin edici sonuçlar elde etmediğini göstermektedir. Bu nedenle, yazarlar dört ana saldırı kategorisinin tümünü ele almak için tek bir algoritma uygulama yaklaşımını sorgulamaktadır. Bunun yerine, her saldırı kategorisi için en iyi PD ve En Düşük YAO'ı üstün algoritmalara dayanan çoklu sınıflayıcı bir yaklaşım önermektedir. Çok sınıflandırıcı modelinde MLP, GAU ve K-Means bulunur. Bu model, daha yüksek PD ve daha düşük YAO açısından gelişmeler göstermektedir. Yazarlar ayrıca, U2R ve R2L saldırıları için daha iyi sonuçlar elde etmek için hala çok çalışma ve araştırmanın yapılması gerektiğini önermektedir.

[44], internet, IoT ve 5G ağları gibi modern ağlarda gerçek zamanlı uygulama yapabilen hafif bir algoritma bulma motivasyonu ile gerçekleştirilmiştir. Bu nedenle, NB, MLP, J48, NB Tree ve RF da dahil olmak üzere bir avuç algoritma performansı WEKA veri madenciliği aracı kullanılarak NSL-KDD veri kümesinin üç alt grubunda karşılaştırılır. Alt veri kümeleri, Consistency Subset Eval ve CFs Subset Eval özellik değerlendirme algoritmaları ile birlikte Best First Search ve Rack Search arama algoritmaları kullanılarak oluşturulmuştur. Alt veri kümeleri sırasıyla 13, 26 ve 11 özellik içerir. Çalışmanın sonuçları J48'in 1. ve 2. veri kümeleri üzerinde diğer algoritmalardan daha yüksek hassasiyet, geri çağırma ve AUC, ve RF'in 3. veri kümesi üzerinde daha yüksek performans gösterdiğini göstermektedir. Bu çalışma, verimli bir IDS'nin performansının en uygun özellik seçimine bağlı olduğunu göstermektedir. Çalışma ayrıca IoT bağlamında izinsiz giriş tespitinde, zamanın önemli olduğunu belirtiyor. Bunu göz önünde bulundurarak NB en iyi senaryo, NB Ağacı ise en kötüsüdür.

[45], gerçek dünya saldırı senaryoları ve mevcut ağ trafiğini temsil etmeyen KDD-99 ve NSL-KDD gibi yaygın kullanılan veri kümeleriyle ilgili problemi ele almaktadır. Bu nedenle yazarlar, Apache Spark aracını ve onun MLlib'sini kullanarak UNSW-NB15 veri kümesi üzerindeki NB, SVM, C4.5 ve Random Forest dahil olmak üzere dört popüler sınıflandırıcının performansını karşılaştırmak için bir deney yapmıştır. Algoritmaların performansını analiz etmek için doğruluk, duyarlılık, özgünlük ve zaman gibi çeşitli metrikler kullanılmıştır. İkili sınıflandırma sonuçları RF'nin %97,49'luk en yüksek doğruluğu elde ettiğini, ardından C4,5'in %95,82'lik bir doğruluğu elde ettiğini göstermektedir. SVM ve NB sırasıyla %92,28 ve %74,19 doğruluk oranına sahiptir. Sonuçlar aynı zamanda NB'yi eğitim ve tahmin süresi açısından en hızlı algoritma olarak, SVM'nin en yavaş olduğunu da göstermektedir.

[49], esas olarak, ağa izinsiz giriş tespiti ile ilgili çalışmalarda yoğun olarak kullanılan KDDCUP 99 veri kümesinde bulunan kusurları göstermeye çalışırken, bu veri kümesinin yerini alabilecek diğer uygun seçenekleri araştırmaya odaklanmaktadır. KDDCUP 99, Sinir Ağları, SVM, DT, RF, NB ve K-means ML algoritması kullanılarak NSL-KDD ve UNSW-NB15 veri kümeleri ile karşılaştırılmıştır. Her

algoritmanın performansını tüm veri kümeleriyle değerlendirmek için Ağırlıklı F-ölçütü kullanılmıştır. NSL-KDD ve UNSW-NB15 eğitim kümelerini daha da geliştirmek için SMOTE oversampling ve random undersampling teknikleri kullanılmıştır. Sonuçlar UNSW-NB15-SMOTE'nin diğer iki veri kümelerinden daha iyi sonuçlar elde edebileceğini ve literatürde kullanılan geleneksel ve eski veri kümelerinin yerini alabileceğini göstermektedir.

[51], KDD-CUP 99 ve NSL-KDD gibi geleneksel ağ saldırı veri kümelerinin eksikliklerini; ve KNN ve K-means algoritmalarını kullanarak daha güncel CIDDS-001 veri kümesinin yeteneklerini analiz etmektedir. Veri kümesi iki bölümden oluşur: Harici sunucu trafiği ve Openstack sunucu trafiği. Veri kümesindeki her kayıtta 14 özellik vardır ancak bu deneyde yalnızca 12 tanesi kullanılmıştır. Bu çalışmanın amacı doğrultusunda, veri kümesi eğitim ve test kümeleri için sırasıyla %66 ve %34 bölümlere ayrılmıştır. İlk olarak, veri kümesi, KNN algoritması kullanılarak, hem harici sunucu hem de Openstack sunucusu trafiği için 1-5 arasında değişen K ile analiz edilir. Harici sunucu trafiği beş sınıfta (Normal, Saldırgan, Mağdur, Şüpheli ve Bilinmeyen) analiz edilir ve K = 5 ile en düşük 0,993 ve K = 2 ile en yüksek 0,996 doğruluk oranına ulaşmaktadır. Openstack trafiği, üç sınıf (Normal, Saldırgan ve Kurban) üzerinde analiz edilir ve algoritmanın K'nın tüm değerleri için 1.000 doğruluk oranı elde etmektedir. Aynı işlem, harici sunucu trafiği için K-means Kümeleme algoritması için tekrarlanır. Bu çalışma, NID modellerinin çevrimdışı ve çevrimiçi eğitimi için CIDDS-001 veri setinin optimal bir seçenek olduğunu göstermektedir.

[66] 'nin amacı, NSL-KDD veri kümesini kullanarak ağ anomalisi tespiti bağlamında RF, NB, LR ve SVM de dahil olmak üzere denetimli makine öğrenme algoritmalarının performansını karşılaştırmaktır. Doğruluk, hassasiyet, Geri çağırma, F1-ölçütü, GPO ve YPO açısından değerlendirme yapılmaktadır. Algoritmaların her biri aşağıdaki doğruluk oranlarına ulaşır: RF %99, LR %84, NB %79 and SVM %75. Sonuçlar, RF'nin her üç algoritmayı da geride bıraktığını, SVM ise tüm değerlendirme durumlarında en düşük puanları aldığını göstermektedir.

2.7. Literatür Taramasının Değerlendirilmesi

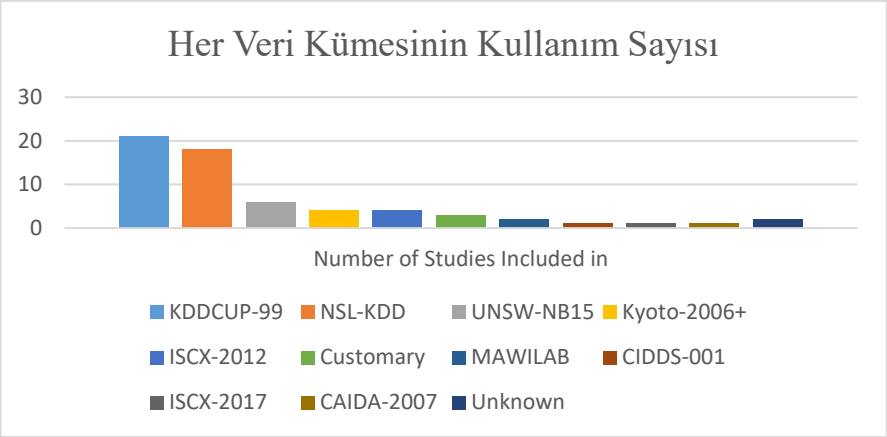
Bu bölümde, bu çalışma kapsamında hangi öğrenme yönteminin, algoritmalarının ve veri kümelerinin kullanılacağını belirlemek amacıyla önceki bölümde incelenen çalışmalar değerlendirilmektedeir. Çalışmanın yukarıda belirtilen maddelerini belirlemek için kullanılan kriterler veri kümeleri, çalışmaların sayısı ve performansı içermektedir.

2.7.1. Veri kümelerinin analizi

Aşağıdaki tablo ve şekil, tüm veri kümelerinin kullanım sayısını vurgulamaktadır. Bu veri kümelerinin hiçbiri, bazılarının modası geçmiş olması ve gerçek dünya trafiğine benzememesi ve bazılarının makine öğrenme algoritmalarını tam olarak eğitmek için gerekli sayıda saldırı trafiğine sahip olmaması nedeniyle kullanılmamıştır. Bu nedenle, daha güncel ve daha az araştırılan CIC-IDS-2017 ve CSE-CIC-IDS-2018 veri kümelerinin bu çalışmada kullanılması tercih edilmiştir.

Tablo 2.3. Literatür taramasında kullanılan veri setlerinin analizi

NO	Veri Kümesi	Dahil Edilen Çalışma Sayısı
1	KDDCUP-99	21
2	NSL-KDD	18
3	UNSW-NB15	6
4	Kyoto-2006+	4
5	ISCX-2012	4
6	Customary	3
7	MAWILAB	2
8	CIDDS-001	1
9	ISCX-2017	1
10	CAIDA-2007	1
11	Unknown	2



Şekil 2.2. Her veri kümesinin kaç kez kullanıldığı.

2.7.2. Öğrenme yöntemlerinin analizi

Tablo 2.4. her bir öğrenme yönteminin tüm çalışmalarda kaç kez kullanıldığını ve ortalama doğruluk oranını göstermektedir.

Tablo 2.4. Tüm çalışmalarda her bir öğrenme yönteminin ortalama doğruluğu.

NO	Öğrenme Yöntemi	Çalışmaların Sayısı	Ortalama Doğruluk (%)
1	Denetimli Öğrenme	41	94,3
2	Yarı Denetimli Öğrenme	9	91,2
3	Denetimsiz Öğrenme	5	80,3
4	Derin Öğrenme	2	95,6

Denetimli öğrenmenin tüm çalışmalarda en çok kullanıldığı ve diğer yöntemlerden daha iyi doğruluğa sahip olduğu kanıtlanmış olması, ağ anomali tespiti bağlamında bir öğrenme yöntemi olarak en iyi seçim olduğu anlamına gelmektedir. Bu nedenle, bu çalışma kapsamında, öğrenme modellerini eğitmek için denetimli öğrenme yöntemi kullanılacaktır.

2.7.3. Makine öğrenmesi algoritmalarının analizi

Bu bölümde tüm çalışmalarda kullanılan denetimli makine öğrenmesi algoritmaları incelenmiştir. Tablo 2.5. her algoritmanın çalışmalarda kaç kez kullanıldığını göstermektedir.

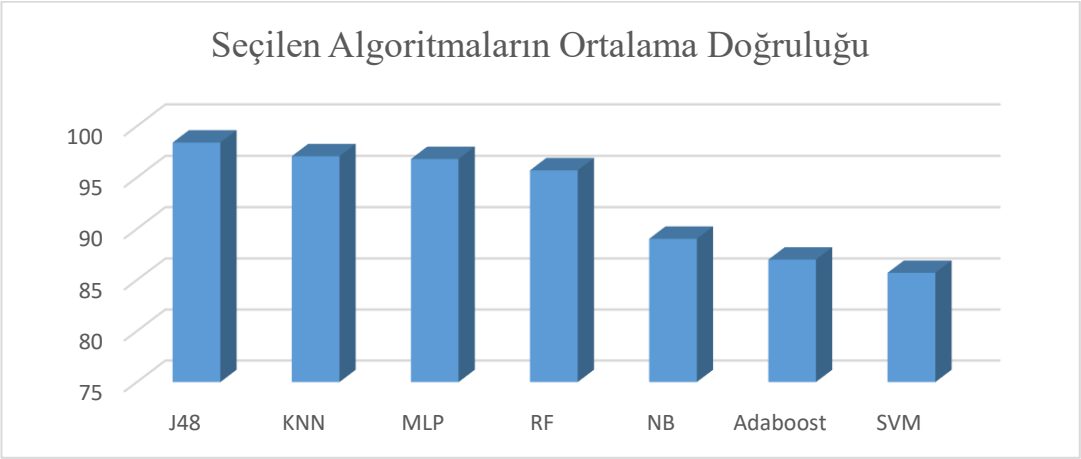
Tablo 2.5. Her denetimli algoritmanın dahil olduğu çalışma sayısı.

NO	Algoritmalar	Dahil Edilen Çalışma Sayısı
1	NB	23
Support Vector Machines		
2	SVM	18
3	SVM-SMO	2
4	FSVM	1
5	PLSSVM	1
6	M-SVM	1
Random Forest		
7	RF	15
Nearest Neighbors		
8	KNN	13
9	TCM-KNN	2
10	KNNGA	1
11	IBK	1
12	hw-IBK	1
13	LWL	1
Neural Networks		
14	MLP	10
15	ANN	5
16	RBF	4
17	FC-ANN	1
Decision Trees		
18	C4.5	8
19	J48	6
20	DT	6
21	ID3	3
22	CART	2
23	Random Tree	2
24	NB Tree	2
25	Decision Table	1
26	GB Tree	1
27	Bagged Trees	1
28	REP Tree	1
29	Decision Stump	1
Regressions		
30	LR	6
31	Linear Regression	1
Boostings		
32	Adaboost	1
33	Gentleboost	1
34	Logitboost	1
35	Rustboost	1
Diğer algoritmalar-Tek		
36	Winnow	1
37	NX	1
38	FCM	1
39	GAU	1
40	FB-VQ	1
41	GA	1
42	FL	1
43	ELM	1
44	M-ELM	1
Diğer algoritmalar-Melez		
45	AIN+RFB	1
46	MLP+RBF	1

Tablo 2.5. (Devamı)

NO	Algoritmalar	Dahil Edilen Çalışma Sayısı
47	Neuro Tree	1
48	GA+SVM	1
49	C5.0+LSSVM	1
50	M-ELM+SVM	1
51	SVM+RF	1
52	SVM+CART	1
53	SVM+J48	1
54	SVM+IBK	1
55	SVM+LR	1
56	SVM+BayesNet	1
57	SVM+OneR	1
58	SVM+JRip	1
59	SVM+Adaboost	1
60	SVM+KNN	1
61	NB+Adaboost	1
62	C4.5+KNN	1
63	PCA+CART	1

Tüm bu algoritmalar arasından, yedi tanesi bu çalışma kapsamında, kaç kez kullanıldıklarına ve ortalama doğruluklarına dayanılarak kullanılmak üzere seçilmiştir. Bu algoritmalar arasında Naïve Bayes (NB), Support Vector Machine (SVM), J48, Random Forest (RF), AdaBoost, Multilayer Perceptron (MLP-NN) ve K-Nearest Neighbor (KNN) bulunur. Bu algoritmalar, farklı kategorilerdeki geniş bir sınıflandırma algoritması ailesini temsil etmektedir. Şekil 2.3., incelenmiş çalışmalardaki bu algoritmaların ortalama doğruluklarını göstermektedir.



Şekil 2.3. Seçilen algoritmaların ortalama doğruluğu.

Aşağıdaki bölüm bu algoritmaların her birini kısaca açıklamaktadır.

2.8. Makine Öğrenmesi Algoritmaları

2.8.1. Naïve Bayes

Naïve Bayes (NB), bağımsız özelliklerin varsayımı ile basitleştirilmiş Bayes Teoremine dayanan bir sınıflandırma algoritmaları ailesidir. Başka bir deyişle, Naïve Bayes algoritmaları, belirli bir sınıfta birden çok özelliğin varlığının tamamen ilişkisiz olduğunu ve her özelliğin bağımsız olarak tahminin sonucuna katkıda bulunduğunu varsaymaktadır. Büyük veri kümeleriyle hızlı ve çok verimli çalışmakla beraber az eğitim süresine ihtiyaç duymaktadır. Buna rağmen, özelliklerin birbirinden bağımsız olduğunu varsayan bu olasılıksal sınıflandırıcının çok temel doğası nedeniyle, bazı uygulamalarda oldukça düşük bir doğruluk almaktadır. Bu dezavantaja rağmen, Naïve Bayes gerçek zamanlı tahmin, metin sınıflandırması, öneri sistemleri ve çok sınıflı tahmin gibi alanlarda hala popüler bir algoritmadır (bazı durumlarda daha gelişmiş algoritmalardan daha iyi performans göstermektedir) [78]. Naïve Bayes denklemi aşağıda açıklanmıştır.

$$P(c \mid x) = \frac{P(C|X)P(c)}{P(x)}$$

$$P(c|X) = P(x_1|c) \times P(x_2|c) \times \ldots \times P(x_n|c) \times P(c) \quad (2.1)$$

yukarıda,

- P (c | x), öngörücü (x, özellikler) verilen sınıf (c, hedef) 'in arka olasılığıdır.
- P (c), önceki sınıf olasılığıdır.
- P (x | c), öngörücünün verilen sınıfının olasılığıdır.
- P (x), önceki öngörücünün olasılığıdır.

2.8.2. Support vector machine

Support Vector Machine (SVM), en iyi Hiper Düzlemi bularak öğeleri ayırmaya dayanan sınıflandırma ve regresyon için kullanılan denetimli bir makine öğrenme

algoritmasıdır. En iyi hiper düzlem, en yakın veri noktalarından maksimum marjı tutan çizgidir. Bu veri noktaları aynı zamanda hiper düzlemin pozisyonunu belirleyen destek vektörleri olarak da bilinir. SVM, öğeleri N boyutlu bir alanda veri noktaları olarak çizmektedir; burada N, kullanılabilir özelliklerin sayısını temsil eder [79].

Başlangıçta SVM sadece hiper düzlemin düz bir çizgi olduğu doğrusal sınıflandırma veya regresyon için geliştirilmişti. Ancak özellik sayısı arttıkça hiper düzlemin yapısı da artmaktadır, örneğin özellik sayısı 3 ise hiper düzlem iki boyutlu bir boşluktur. SVM'yi çok boyutlu alanlarda kullanmak için Çekirdek Hilesi olarak bilinen bir teknik kullanılır. Çekirdek hilesi, Çekirdek, Düzenleme (C olarak temsil edilir), Gama ve Marjin dahil olmak üzere belirli parametreleri kullanmaktadır [80].

SVM, yüksek boyutlu alanlarda güçlü ve bellek açısından verimli bir sınıflandırıcıdır. Bununla birlikte, yüksek eğitim süresi ve gürültü, yani çakışan hedef sınıflar nedeniyle büyük veri kümeleriyle çalışırken iyi performans göstermez.

2.8.3. K-nearest neighbor

K-Nearest Neighbor (KNN), sınıflandırma ve regresyon problemlerinde kullanılan parametrik olmayan ve tembel bir makine öğrenmesi algoritmasıdır. Parametrik olmayan, verilerin temeldeki dağılımı hakkında herhangi bir varsayımda bulunmadığı ve modelin yapısının verilerden belirlendiği anlamına gelmektedir. Tembel, KNN'nin eğitim verileri üzerinde herhangi bir genelleme yapmadığını veya minimum genelleme gerçekleştirdiğini belirtir, bu da eğitim aşamasının gerçekten hızlı olduğu, ancak örneklerin veya veri noktalarının her zaman algoritma ile tutulduğu anlamına gelmektedir. Bu nedenle tahmin aşaması oldukça yavaştır ve çok fazla kaynak gerektirmektedir [81].

KNN, benzer veri sınıflarının birbirine yakın olduğu varsayımına dayanarak çalışır. Dolayısıyla, yeni bir numune veya veri noktasının sınıflandırılması gerektiğinde, algoritma en yakın komşular arasında benzerlikler arar. Yeni örnek, çoğunluğu olan komşuların sınıfına ait olacaktır. Burada K test edilecek yakın veya en yakın komşu

sayısını belirler. K, komşuların sınıflandırılmamış veri noktasından uzaklığına göre seçilir. K seçilirken hassasiyet şarttır çünkü K çok küçük seçilirse tahminin kararlılığı azalır ve çok büyük seçilirse hata sayısı artacaktır [81].

KNN'nin avantajları aşağıdakileri içerir [82].

- Veriler hakkında varsayım yok - örneğin doğrusal olmayan veriler için yararlı.
- Basit algoritma - açıklamak ve anlamak / yorumlamak.
- Yüksek doğruluk (nispeten) - daha yüksek denetimli öğrenme modellerine kıyasla oldukça yüksek ancak rekabetçi değil.
- Çok yönlü - sınıflandırma veya regresyon için yararlı.

KNN'nin dezavantajları aşağıdakileri içerir [82]:

- Hesaplama pahalıdır - çünkü algoritma tüm eğitim verilerini depolar.
- Yüksek bellek gereksinimi.
- Eğitim verilerinin tümünü (veya neredeyse tümünü) saklar.
- Tahmin aşaması yavaş olabilir (büyük N ile).
- Alakasız özelliklere ve verilerin ölçeğine duyarlı.

2.8.4. J48

J48, Karar Ağacı tabanlı bir algoritmadır. Karar Ağacı algoritmaları anlamsız büyük veri kümelerini düz ileri ve anlaşılabilir kural tabanlı ağaçlara dönüştürür. Basit bir ifadeyle, bu algoritmalar, bir veri kümesindeki örnekleri bir bölme ve fethetme (divide and conquer) ilkesi uygulayarak küçük gruplara ayırır. Karar ağacı üç unsurdan oluşur: düğümler (kök düğüm ve alt düğümler), dallar ve yapraklar. Her düğümde, algoritmanın bir dal seçip bir sonraki düğüme geçtiği bir karar ifadesi vardır. Bu işlem, algoritma bir hedef özelliğin değerini tutan yapraklara ulaştığında sona erer [83].

J48, Ross Quinlan tarafından oluşturulan ve karar ağaçları üretmek için kullanılan C4.5 algoritmasının uygulanmasıdır. C4.5 tarafından üretilen karar ağaçları denetimli

sınıflandırma problemlerinde kullanılabilecek istatistiksel sınıflandırıcılardır. Bir karar ağacı oluşturmak için C4.5 algoritması, Bilgi Teorisinden Entropy ve Information Gain (IG) olarak bilinen yöntemleri kullanır. İlk adımda, veri kümesi içindeki etiketlerin entropisi ve tüm özniteliklerin veya özelliklerin IG'si hesaplanır ve en yüksek IG'ye sahip öznitelik kök düğüm olarak seçilir. Bu işlem özyinelemeli modda devam eder ve her özyinelemede yalnızca önceden kullanılmamış özellikler dikkate alınır. Aşağıdaki durumlardan biri meydana geldiğinde süreç sona erer [84].

- Alt kümedeki her öğe aynı sınıfa aittir; bu durumda düğüm bir yaprak düğümüne dönüştürülür ve örneklerin sınıfıyla etiketlenir.
- Seçilecek başka özellik yok, ancak örnekler yine de aynı sınıfa ait değildir. Bu durumda, düğüm bir yaprak düğümüne dönüştürülür ve alt kümedeki örneklerin en yaygın sınıfı ile etiketlenir.
- Alt kümede, üst kümedeki hiçbir örneğin seçilen özniteliğin belirli bir değeri ile eşleştiği bulunmadığında gerçekleşen örnek yoktur. Daha sonra üst düğüm kümesindeki örneklerin en yaygın sınıfıyla bir yaprak düğümü oluşturulur ve etiketlenir.

C4.5 algoritması hem ayrık hem de sürekli niteliklerin ele alınması, eksik değerlere sahip veri kümelerinin ele alınması ve üretim işleminden sonra ağaç budaması dahil birçok avantaj sağlar [84].

2.8.5. Random forest

Random Decision Forest olarak da bilinen, Random Forest, sınıflandırma ve regresyon problemlerinde kullanılan denetimli bir öğrenme algoritmasıdır. İsiminden anlaşıldığı gibi, RF modeli eğitmek için Bagging yöntemini kullanarak çok sayıda karar ağacı oluşturur. Basit bir ifadeyle, Rastgele orman, tahminlerde doğruluk ve istikrarı artırmak için birden fazla karar ağacı oluşturur ve bunları birleştirir [85].

Karar Ağaçlarından farklı olarak, RF'lerde, her karar ağacı eğitim seti verilerinin bir alt kümesiyle eğitilir ve her ağacın genel özelliklerin yalnızca bir alt kümesine erişimi

vardır. Alt kümeler her ağaç için rastgele seçilir ve en önemli özelliğe göre bölme yerine, düğümler alt kümedeki en iyi özelliğe göre ayrılır. Bu, öğrenme sürecinin çeşitliliğini arttırır. En yüksek derecelendirmeye sahip olan hedef, ya tüm ağaçların çıktılarının ortalamasıyla (regresyon durumunda) ya da oy çoğunlğu ile seçilir (sınıflandırma durumunda) [85].

RF'in önemli bir avantajı, bir özellik veya niteliğin tüm karar ağaçlarındaki genel öngörüye ne kadar katkıda bulunduğuna bağlı olarak veri kümesindeki özelliklerin göreceli önemini ölçme yeteneğidir. Bu avantajdan dolayı, bu çalışma bağlamında özellik seçimi için Rastgele Orman kullanılmaktadır [86].

RF, büyük veri kümeleriyle kolayca başa çıkabilen esnek bir algoritmadır. Harika bir performansa ve hızlı eğitim süresine sahiptir, ancak ağaç sayısı arttıkça tahmin süreci de yavaşlanır.

2.8.6. AdaBoost

AdaBoost veya uyarlamalı yükseltme, sınıflandırma performansını artırmak için geliştirilen bir Ensemble öğrenme yöntemidir. Yükseltme algoritmaları genellikle diğer sınıflandırma algoritmalarının performansını artırmak için kullanılır ve zayıf sınıflandırıcılardan güçlü sınıflandırıcılar oluştururlar. Bu algoritma ailesinin en uygun kullanımı, zayıf sınıflandırıcı olarak Karar Ağaçları ve güçlendirici algoritma olarak Ada-Boost ile görülür [87].

Ada-Boost, eğitim verilerinin bir modelini oluşturur ve sonra birincisinin hatalarını düzeltmeye çalışan başka bir model oluşturur. Algoritma, eğitim kümesinin mükemmel bir tahmini elde edilene veya maksimum model sayısı eklenene kadar model eklemeye devam edecektir. Ada-Boost ayrıca modellerin her birine veya zayıf sınıflandırıcılara bir ağırlıklandırma katsayısı atar. En düşük hata oranlarına sahip sınıflandırıcılar yüksek ağırlıklarla atanır. Bu ağırlıklar iki durumda önemlidir; birincisi, güçlü sınıflandırıcıyı oluştururken öncelik en yüksek ağırlıklara verilir ve ikinci olarak tahminin nihai çıktısı zayıf sınıflandırıcıların ağırlık ortalamasıdır [87].

Ada-Boost, aykırı değerlere ve gürültüye gerçekten duyarlıdır, bu nedenle eğitim veri kümesi doğru ve yüksek kalitede olmalıdır. Öte yandan, aşırı uydurma (Overfitting) problemi bu algoritmada nadiren görülmektedir.

2.8.7. Multilayer perceptron

Multilayer Perceptron (MLP-NN), İleri Beslemeli Yapay Nöron Ağlarının (ANNs) bir türüdür. ANN, bir hayvan beyninin çalışma şeklinden esinlenen bir makine öğrenme yöntemi veya platformudur. Biyolojik bir beyin nöronlar olarak bilinen bağlı hücrelerden oluştuğu için, ANN'ler birbirine bağlı yapay nöron katmanlarından oluşmaktadır. Bu ağların tüm amacı öğrenme ve karar verme gibi biyolojik bir beynin yeteneklerini taklit etmektir [88].

MLP, en az üç düğüm katmanından oluşur: giriş katmanı, gizli katmanlar ve çıkış katmanları. ANN'ler tamamen bağlı olduğu gibi, bir katmandaki her nöron bir sonraki katmandaki tüm nöronlara belirli bir ağırlıkla bağlanır. Giriş katmanı, sinyali veya bağımsız değişkenleri almaktan sorumludur. Bu katmanda herhangi bir veri işleme türü yoktur. Çıktı Katmanı, kararları veya tahminleri temsil etmekten sorumludur. Bunlar arasında MLP'nin motorları olan gizli katmanlar bulunur. Bu aşamadaki katman ve nöron sayısı sınıflandırma problemine göre değişebilir. Daha fazla sayıda katman ve nöron, algoritmanın performansını geliştirir ve daha karmaşık problemlerle gelmesine yardımcı olur, diğer yandan algoritmanın süresini de artırır. MLP, bir dizi girdi-çıktı çiftinde iki uç düğüm (girdi ve çıktı) arasındaki korelasyonu modellemeyi öğrenir. MLP'de öğrenme, çıktıdaki hata miktarının ve beklenen sonucun karşılaştırılmasına dayanarak, her veri parçası işlendikten sonra bağlantı ağırlıklarının değiştirilmesi ile gerçekleşir. Bu süreç, hataları en aza indirmek için modelin parametrelerinin (ağırlıklarının) ayarlanmasını içerir; bu, geri yayılım (Backpropagation) olarak bilinen denetimli bir öğrenme tekniğidir. Geri yayılım hataya göre tartım ve sapma ayarlamaları yapar ve hatanın kendisi kök ortalama kare hatası (RMSE) dahil olmak üzere çeşitli şekillerde ölçülebilir [88].

BÖLÜM 3. METODOLOJİ

Bu bölümde, CIC-IDS-2017 ve CSE-CIC-IDS-2018 veri kümelerine yedi makine öğrenme algoritmasının uygulanması için kullanılan yaklaşım ile birlikte yazılım ve donanım platformları açıklanmaktadır.

3.1. Platform

Bu çalışmanın hedeflerine ulaşabilmek için birçok yazılım ve donanım araçlarından yararlanılmaktadır. Tablo 3.1. bu araçlar hakkında bilgi vermektedir.

Tablo 3.1. Yazılım ve donanım platformu

NO	Araç	Açıklama
1	Windows 10 Enterprise 64 bit	İşletim Sistemi
2	Python 3.7	Programlama Dili
3	Scikit-learn	Python makine öğrenmesi kütüphanesi
4	Numpy	Matematiksel ve mantıksal işlemleri gerçekleştirmek için kullanılan Python kütüphanesi.
5	Pandas	Büyük veri kümeleriyle çalışmak için kullanılan Python veri analizi kütüphanesi.
6	Matplotlib	Python görselleştirme kütüphanesi.
7	Ms.Excel	Veri kümeleriyle çalışmak için kullanılır.
8	System	Dell Inc. Latitude E5440
9	CPU	Intel® Core(TM) i5-4300U CPU @ 1.90 GHz 2.50 GHz
10	RAM	4096 MB
11	GPU	Intel® HD Graphics Family, 2113 MB

3.2. Algoritma Performans Değerlendirme Metrikleri

Bu çalışmada makine öğrenmesi modellerinin performansını değerlendirmek için kullanılan kriterler Doğruluk (Accuracy), Hassasiyet (Precision), Geri Çağırma (Recall), F-ölçümü (F-measure) ve eğitim süresini içermektedir. Bu kriterlerin her birini açıklamadan önce Karışıklık Matrisi (Confusıon Matrix) kavramına aşina olmalıyız.

Karışıklık Matrisi, veri kümeleriyle çalışırken bir sınıflandırıcının (bir sınıflandırma modeli veya algoritması) performansını değerlendirmek için kullanılan bir referanstır. Bu tablo (Tablo 3.2.), veri kümesinin doğru sınıflandırılmış ve yanlış sınıflandırılmış örneklerinin sayısını içermektedir. Tablonun kendisi nispeten basit ve anlaşılması kolaydır ve doğruluk, hassasiyet, geri çağırma gibi diğer değerlendirme kriterleri tarafından kullanılan ana referanstır [89].

Tablo 3.2. Karışıklık Matrisi

Öngörülen Sınıf	Gerçek Sınıf	
	Attack	Benign
Attack	TP	FP
Benign	FN	TN

Tablo 3.3.'teki terimler aşağıdaki gibi açıklanmıştır,

- Gerçek Pozitif (TP) : Doğru saldırı olarak sınıflandırılan saldırı örneği sayısı.
- Yanlış Pozitif (FP) : Saldırı olarak sınıflandırılmış normal (Benign) örnek sayısı.
- Gerçek Negatif (TN) : Doğru şekilde normal olarak sınıflandırılmış normal örnek sayısı.
- Yanlış Negatif (FN) : Normal olarak sınıflandırılan saldırı örneklerinin sayısı.

Doğruluk, doğru sınıflandırılmış verilerin toplam verilere oranıdır ve aşağıdaki formül [90] ile açıklanmıştır,

$$\text{Doğruluk} = \frac{\text{TP}+\text{TN}}{\text{TP}+\text{FP}+\text{TN}+\text{FN}} \tag{3.1}$$

Hassasiyet, doğru sınıflandırılmış saldırı verilerinin tahmini saldırı verilerinin toplamına oranıdır. Hassasiyet aşağıdaki formül [90] kullanılarak karakterize edilir,

$$\text{Hassasiyet} = \frac{\text{TP}}{\text{TP}+\text{FP}} \tag{3.2}$$

Geri Çağırma, doğru sınıflandırılmış saldırı verilerinin gerçek saldırı verilerinin toplamına oranıdır ve aşağıdaki formül [90] kullanılarak hesaplanır,

$$\text{Geri Çağırma} = \frac{\text{TP}}{\text{TP+FN}} \tag{3.3}$$

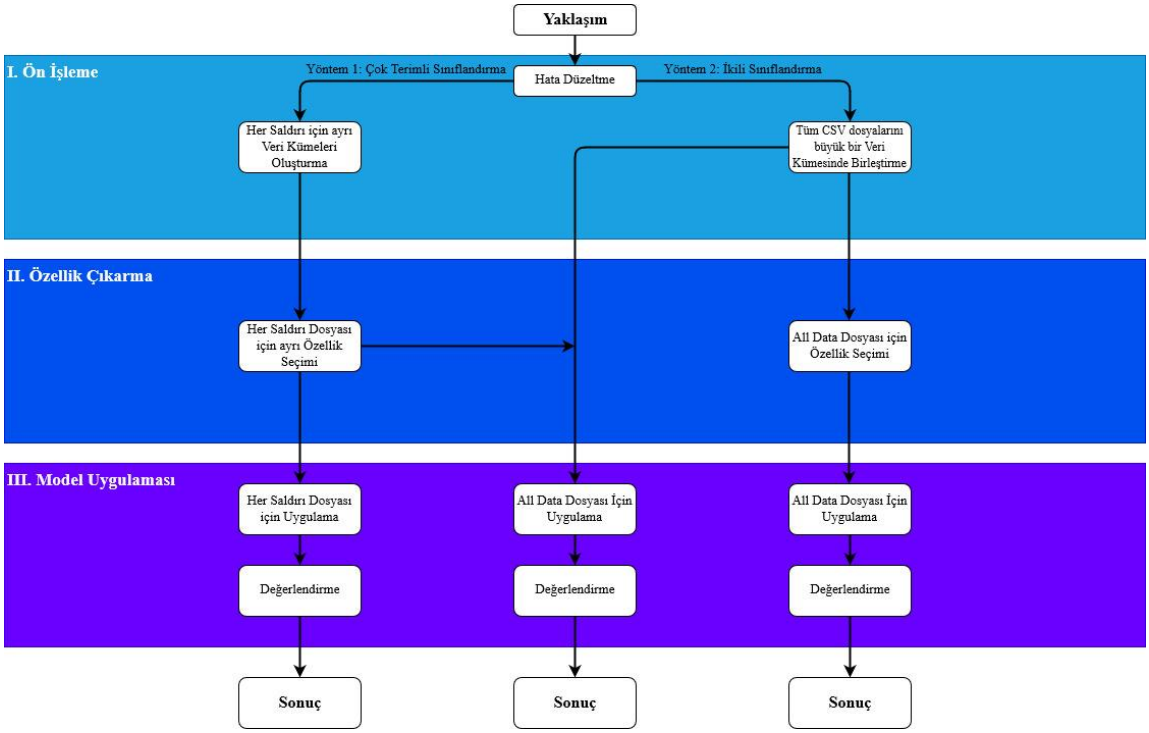
F-ölçütü (F-skoru), bir sınıflandırıcının genel başarısını temsil eder ve hassasiyet ile geri çağırmanın harmonik veya ağırlıklı ortalamasıdır [91]. F1-skoru aşağıdaki gibi gösterilebilir:

$$\text{f1} = \frac{2\times\text{Hassasiyet}\times\text{Geri Çağırma}}{\text{Hassasiyet+Geri Çağırma}} \tag{3.4}$$

Tüm bu metrikler 0 ile 1 arasında bir değer alır, değer ne kadar yüksek olursa (eğitim süresi hariç) sınıflandırıcının performansı o kadar iyi olur. Eğitim süresi, algoritmanın performans değerlendirmesi için önemli bir metrik değil, ancak nihai olanı seçmek için önemli bir kriterdir.

3.3. Yaklaşım

Bu bölümde, CIC-IDS-2017 ve CSE-CIC-IDS-2018 veri kümeleri kullanılarak seçilen denetimli MÖ algoritmalarını ağ anomali tespiti bağlamında eğitmek ve test etmek için bu çalışmada kullanılan yöntem açıklanmaktadır. Şekil 3.1. bu süreçte atılan adımları göstermektedir.



Şekil 3.1. Bu çalışmanın yaklaşımındaki adımlar.

3.3.1. Ön işleme

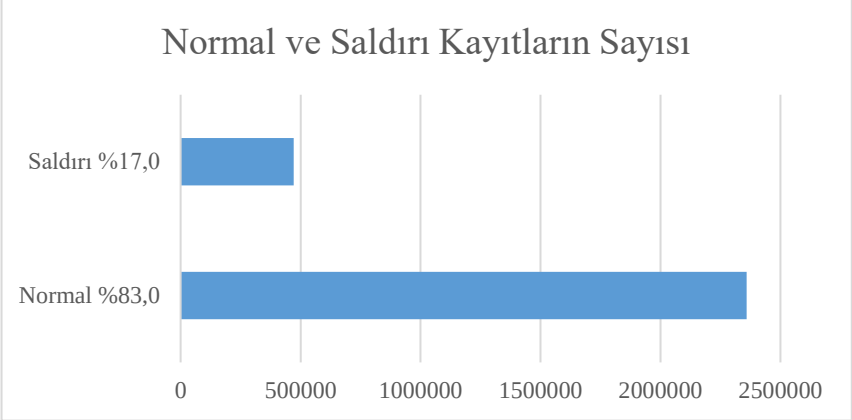
Veri ön işleme, her makine öğrenmesi uygulamasında başlangıç ve gerekli bir süreçtir [92]. İlk olarak bu adımda veri kümeleri, eğitim süreci sırasında hataya neden olabilecek mevcut hatalardan temizlenir. Bu işlem, veri kümelerinden boş kayıtların kaldırılmasını ve sayısal olmayan (dizeler veya kategorik) verileirn MÖ algoritmaları tarafından anlaşılabilen sayısal verilere dönüştürülmesini içeremektedir. Ardından, Benign ve Attack verilerinin ikili sınıflandırması için büyük bir veri kümesi oluşturmak amacıyla CSV dosyaları birleştirilmektedir. Bu çalışmada, benimsenen diğer bir yaklaşım, saldırı türlerine göre Çok Terimli Sınıflandırma gerçekleştirmek için veri kümesinde bulunan her saldırı trafiği türü için ayrı dosyalar oluşturmaktır. Bu tür bir sınıflandırma teorik olarak MÖ algoritmalarının doğruluğunu artırabilmektedir.

3.3.1.1. CIC-IDS-2017 veri kümesinin ön işlemesi

Önceki bölümde açıklandığı gibi CIC-IDS-2017 veri kümesinin ön işlemesi, hata düzeltme ile başlamaktar. Öğrenme sürecinde hatalar doğuracak aşağıdaki maddeler düzeltilmiştir.

- 'Thursday-WorkingHours-Morning-WebAttacks.pcap_ISCX.csv'dosyası 288602 boş / anlamsız kayıt içermektedir. Bu kayıtlar kaldırılmıştır.
- Sütun 41 ('Fwd Header Length' özelliği) sütun 61'de tekrarlanmıştır, bu nedenle sütun 61 veri kümesinden kaldırılmıştır.
- Sql Injection, Brute Force ve XSS gibi Saldırılardaki '–' (Unicode kodu: 8211) karakteri, hatalara yol açan Python-Pandas kütüphanesi tarafından tanınmıyor. Bu karakter "-" ile (Unicode kodu: 45) değiştirilmiştir.
- Sayısal olmayan veriler (dize veya kategorik) sayısal verilere dönüştürülmüştür.

CIC-IDS-2017 veri kümesindeki sekiz ayrı CSV dosyası, Benign and Attack verilerinin ikili sınıflandırması için tek bir büyük dosyada birleştirilmektedir. Şekil 3.2. ön işlemeden sonra veri kümesindeki saldırı ve benign verilerin dağılımını göstermektedir.



Şekil 3.2. Saldırı ve benign verilerin dağılımı.

Bu çalışmada kullanılan diğer bir yaklaşım, veri kümesinde bulunan saldırı türlerine dayanan Çok Terimli Sınıflamasıdır. Bu nedenle, büyük veri kümesi her saldırı türü için 10 ayrı veri kümesi oluşturmak üzere ön işlenir. Bu adımda web saldırıları (Web Attack - Brute Force, Web Attack - XSS ve Web Attack - Sql Injection) 'Web Attacks' olarak tek bir dosyada birleştirilmiştir. Bu işlem %30 saldırı verisi ve %70 rastgele seçilen benign veriler oranında ayrı CSV dosyaları oluşturmaktadır. Tablo 3.3. saldırı verilerinin veri kümesindeki dağılımını göstermektedir.

Tablo 3.3. Veri kümesindeki saldırı verilerinin dağılımı

NO	Saldırı Adı	Kayıt Sayısı
1	DoS Saldırısı (Hulk)	231074
2	PortScan Saldırısı	158931
3	DDoS Saldırısı	41836
4	DoS Saldırısı (GoldenEye)	10294
5	FTP Patator Saldırısı	7939
6	SSH Patator Saldırısı	5898
7	DoS Saldırısı (Slowloris)	5797
8	DoS Saldırısı (Slowhttptest)	5410
9	Bot Saldırısı	1967
10	Web Saldırısı (Brute Force)	1508
11	Web Saldırısı (XSS)	653
12	Infiltration Saldırısı	37
13	Web Saldırısı (Sql Injection)	22
14	Heartbleed Saldırısı	12

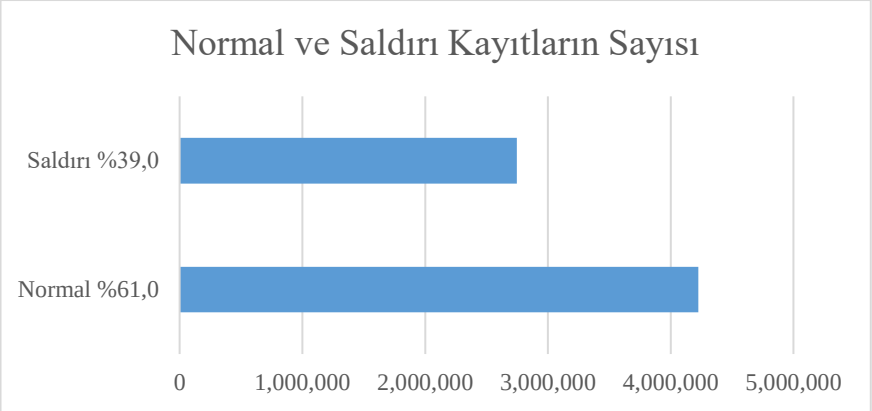
3.3.1.2. CSE-CIC-IDS-2018 veri kümesinin ön işlemesi

CSE-CIC-IDS-2018'in ön işlemesi, CIC-IDS-2017 veri kümesiyle hemen hemen aynı adımları izlemektedir. Öğrenme sürecinde hatalar doğuracak aşağıdaki maddeler düzeltilmiştir.

- 'Wendnesday-21-02-2018_TrafficForML_CICFlowMeter.csv' dosyası 71 boş / anlamsız kayıt içerimektedir. Bu kayıtlar kaldırılmıştır.
- Sql Injection, Brute Force ve XSS gibi Saldırılardaki '–' (Unicode kodu: 8211) karakteri, hatalara yol açan Python-Pandas kütüphanesi tarafından tanınmıyor. Bu karakter '-' (Unicode kodu: 45)ile değiştirilir.
- 'Flow Bytes / s' ve 'Flow Packets / s' sütunlarındaki bazı veriler sayısal değildir; bunlar sayısal verilere dönüştürülmüştür.

- Sayısal olmayan veriler (dize veya kategorik) sayısal verilere dönüştürülmüştür.
- Başlangıçta veri kümesinde yaklaşık 6 milyon benign kayıt bulunmaktadır. Hesaplama sınırlamaları nedeniyle bu sayı, 4.224.806 kayda indirilmiştir.

CSE-CIC-IDS-2018 veri kümesindeki 10 ayrı CSV dosyası, Benign and Attack verilerinin ikili sınıflandırması için tek bir büyük dosyada birleştirilmektedir. Şekil 3.3. ön işlemeden sonra veri kümesindeki saldırı ve benign verilerin dağılımını göstermektedir.



Şekil 3.3. Saldırı ve benign verilerin dağılımı

İkinci yaklaşımda veri kümesi, her saldırı türü için 12 ayrı veri kümesi oluşturmak üzere ön işlenir. Bu adımda web saldırıları (Brute Force -Web, Brute Force –XSS ve SQL Injection) bir araya getirilerek Web Attacks olarak kaydedilir. Bu işlem,% 30 saldırı verisi ve% 70 rasgele seçilen brnign veri oranı ile ayrı CSV dosyaları oluşturur. Tablo 3.4. veri kümesindeki saldırı verilerinin dağılımını göstermektedir.

Tablo 3.4. Veri kümesindeki saldırı verilerinin dağılımı

NO	Saldırı Adı	Kayıt Sayısı
1	DDOS Saldırısı (HOIC)	686011
2	DDoS Saldırısı (LOIC-HTTP)	576191
3	DoS Saldırısı (Hulk)	461912
4	Bot Saldırısı	286191
5	FTP BruteForce Saldırısı	193360
6	SSH Bruteforce Saldırısı	187589
7	Infilteration Saldırısı	161934
8	DoS Saldırısı (SlowHTTPTest)	139890
9	DoS Saldırısı (GoldenEye)	41508
10	DoS Saldırısı (Slowloris)	10990
11	DDOS Saldırısı (LOIC-UDP)	1730
12	Brute Force Saldırısı (Web)	611
13	Brute Force Saldırısı (XSS)	230
14	SQL Injection Saldırısı	87

3.3.2. Özellik çıkarma

Bu bölümde, her iki veri kümesinin özellikleri, hangi özelliklerin bir anomali türünü daha iyi belirlediğini belirlemek için değerlendirilmektedir. Diğer bir deyişle, her özellik, bir saldırıyı veri kümelerinde bulunan diğer özelliklerden daha iyi tanımlayıp tanımlayamayacağını görmek için incelenmektedir. Bunu başarmak için bir çeşit önem ölçülmelidir. Bu amaçla Random Forest Regressor algoritmasını kullanılmıştır. Bu algoritma karar ağaçlarından oluşmaktadır. Karar ağaçları, ağacın yapımında yararlılıklarını belirlemek için her özelliğe önem ağırlığı verir. Tüm özelliklerin önem ağırlıklarının toplamı, ağacın önem ağırlığını ve her bir özelliğin ağırlığının bu toplama oranı, özelliğin genel ağırlıklı önemini tanımlar [93].

Burada belirtilmesi gereken önemli bir nokta, klasik özellik çıkarma yaklaşımlarının veri kümesinde bulunan tüm özellikleri kullanmasıdır. Ancak bu çalışmada Source IP, Destination IP, Source Port, Destination Port, Flow ID, Protocol, External IP ve Timestamp gibi aralık tabanlı özellikler bu işlemden çıkarılmıştır. Bunun nedeni, saldıran bir tarafın bilinmeyen bağlantı noktaları (not-well-known ports) veya sahte IP adresleri kullanmayı tercih edebilmesidir. Genel özellikler ile değişmez veri şekli bir saldırıyı daha iyi tanımlayabilmektedir [94]. Tüm özelliklerin listesi ve açıklamaları Ek A'da bulunmaktadır.

Bu adımda, modellerin eğitim süreci için özellikler seçmek üzere iki yaklaşım izlenir.

3.3.2.1. Çok terimli sınıflandırma için özellik çıkarma

Bu yaklaşımda, en büyük öneme sahip özellikler her saldırı dosyasına göre seçilmektedir. Bu, her saldırının kendilerini daha iyi tanımlayan kendi alt özellikleri olduğu anlamına gelmektedir. Tablolar 3.5. ve 3.6. CIC-IDS-2017 ve CSE-CIC-IDS-2018 veri kümelerinde her saldırı türü için en önemli özellikleri göstermektedir.

Tablo 3.5. CIC-IDS-2017 veri kümesindeki her saldırı için en önemli özellikler

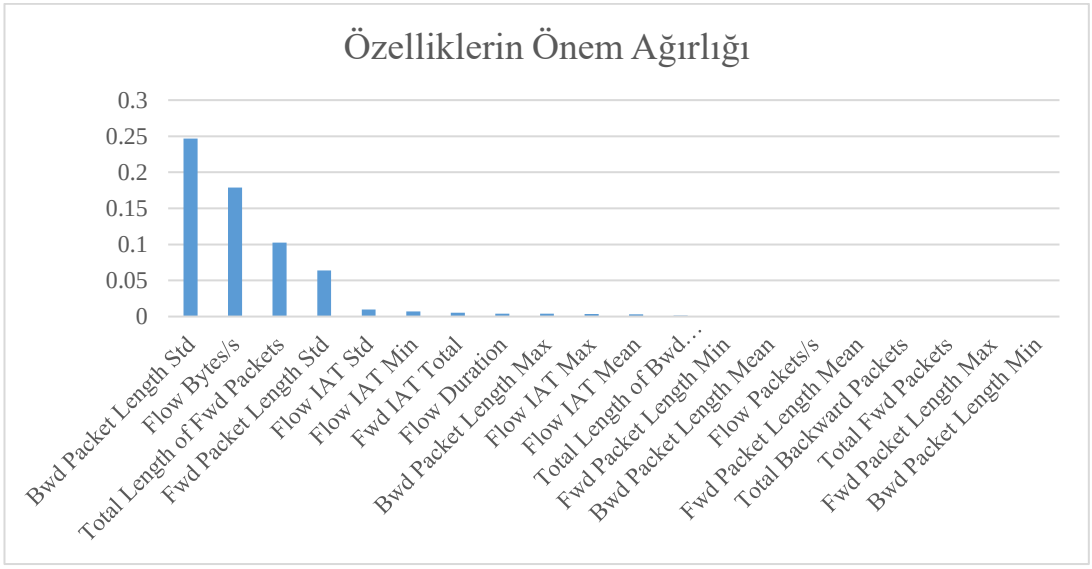
NO	Saldırı Adı	En Yüksek Öneme Sahip Özellikler
1	DoS Hulk	Bwd Packet Length Std, Fwd Packet Length Std, Fwd Packet Length Max, Flow Duration
2	PortScan	Flow Bytes/s, Total Length of Fwd Packets, Fwd Packet Length Max, Flow IAT Mean
3	DDoS	Bwd Packet Length Std, Total Backward Packets, Fwd IAT Total, Total Length of Fwd Packets
4	DoS GoldenEye	Flow IAT Max, Bwd Packet Length Std, Flow IAT Min, Total Backward Packets
5	FTP-Patator	Fwd Packet Length Max, Fwd Packet Length Std, Fwd Packet Length Mean, Bwd Packet Length Std
6	SSH-Patator	Fwd Packet Length Max, Flow IAT Mean, Flow IAT Max, Total Length of Fwd Packets
7	DoS slowloris	Flow IAT Mean, Total Length of Bwd Packets, Bwd Packet Length Mean, Total Fwd Packets
8	DoS Slowhttptest	Flow IAT Mean, Fwd Packet Length Std, Fwd Packet Length Mean, Fwd Packet Length Min
9	Bot	Bwd Packet Length Mean, Flow Duration, Flow IAT Std, Flow IAT Min
10	Web Attacks	Bwd Packet Length Std, Flow IAT Min, Total Length of Fwd Packets, Bwd Packet Length Max
11	Infiltration	Fwd Packet Length Mean, Total Backward Packets, Total Length of Fwd Packets, Flow Duration
12	Heartbleed	Bwd Packet Length Mean, Bwd Packet Length Max, Total Length of Fwd Packets, Fwd Packet Length Max

Tablo 3.6. CSE-CIC-IDS-2018 veri kümesindeki her saldırı için en önemli özellikler

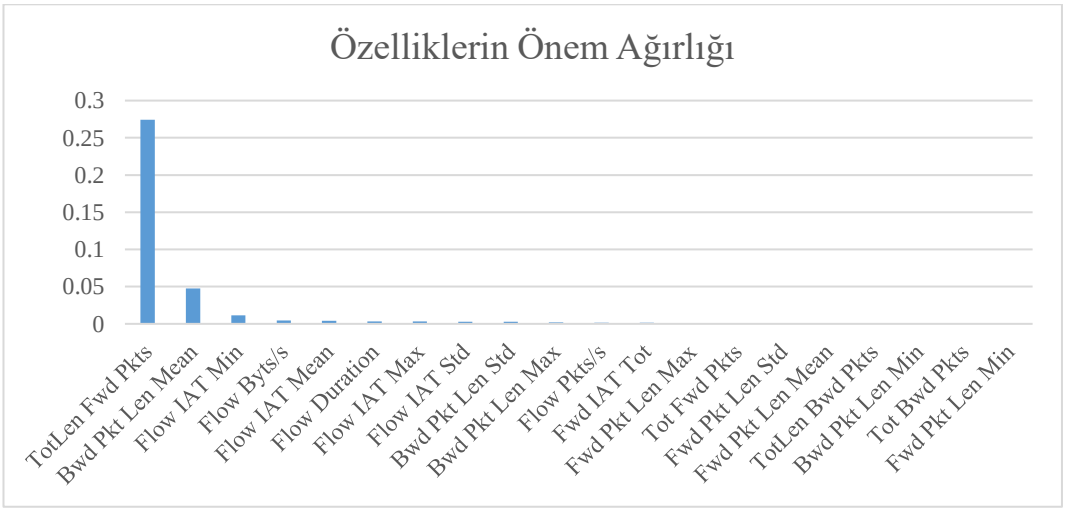
NO	Saldırı Adı	En Yüksek Öneme Sahip Özellikler
1	DDOS attack-HOIC	Flow Duration, Fwd IAT Tot, Bwd Pkt Len Std, Bwd Pkt Len Mean
2	DDoS attacks-LOIC-HTTP	Flow Duration, TotLen Fwd Pkts, Flow IAT Max, Flow IAT Std
3	DoS attacks-Hulk	Bwd Pkt Len Std, Flow IAT Min, Fwd Pkt Len Max, Fwd Pkt Len Std
4	Bot	Flow Pkts/s, Bwd Pkt Len Mean, Flow IAT Mean, Flow IAT Min
5	FTP-BruteForce	Tot Fwd Pkts, Flow Duration, Flow IAT Mean, Flow IAT Max
6	SSH-Bruteforce	Flow IAT Mean , Flow IAT Max, Flow Pkts/s, Flow Duration
7	Infilteration	Flow Byts/s, Fwd Pkt Len Std, Flow IAT Min, Flow IAT Max
8	DoS attacks-SlowHTTPTest	Flow IAT Max, Flow IAT Mean, Flow Pkts/s, Flow Duration
9	DoS attacks-GoldenEye	Fwd Pkt Len Max, Fwd IAT Tot, Flow Duration, Flow IAT Max
10	DoS attacks-Slowloris	Fwd Pkt Len Max, Flow Byts/s, Fwd Pkt Len Std, Tot Bwd Pkts
11	DDOS attack-LOIC-UDP	TotLen Fwd Pkts, Tot Fwd Pkts, Flow Duration, Bwd Pkt Len Mean
12	Web Attacks	Flow IAT Max, Fwd Pkt Len Mean, Flow Duration, Flow Byts/s

3.3.2.2. İkili sınıflandırması için özellik çıkarma

Bu yaklaşımda özellikler bir saldırıyı tanımlayabilme yeteneklerine göre, tüm veri kümesine kıyasıyla seçilmektedir. Bu yaklaşımın kullanılmasıyla, modelin veri kümesindeki örneklerin Benign veya Attack olmak üzere iki kategoriye ayrılacağı ikili sınıflandırmasını yapabileceği anlamına gelmektedir. Şekil 3.4. ve 3.5. sırasıyla CIC-IDS-2017 ve CSE-CIC-IDS-2018 veri kümeleri için seçilen özelliklerin ayrıntılarını ve bunların önemlerinin oranını temsil etmektedir.



Şekil 3.4. CIC-IDS-2017 için seçilen özellikler ve onların önemi



Şekil 3.5. CSE-CIC-IDS-2018 için seçilen özellikler ve onların önemi

3.3.3. Model uygulaması

Bu adımda yedi makine öğrenmesi algoritmasının tümü, önceki adımda seçilen özellikler kullanılarak veri kümeleri üzerinde eğitilip test edilmektedir. Burada karşılaşılan tek sorun, ne CIC-IDS-2017'nin ne de CSE-CIC-IDS-2018'in eğitim ve test için ayrı kümelerinin bulunmamasıdır. Bu nedenle, MÖ algoritmalarının (modellerinin) performansını değerlendirmek için veri kümeleri eğitim ve test bölümlerine ayrılmalıdır. Bu görev Python Scikit Learn kütüphanesinin train_test_split [95] ifadesinden yararlanarak gerçekleştirilmektedir. Bu ifade, veri kümelerini kullanıcı tarafından tanımlanan oranla iki parçaya ayırır. Tercih edilen oran genellikle %80 eğitim ve %20 testtir [96]. Aynı ilke bu çalışma bağlamında da takip edilmektedir.

Modelleri eğitmek için kullanılan yöntem, 10 kat çapraz doğrulama (10-fold cross validation) yöntemidir. Bu yöntem, yeterli ve daha iyi sonuçlar elde etmek için her algoritmayı veri kümelerine 10 kez uygulamaktadır [97]. Nihai sonuçlar bu sürecin aritmetik ortalamasıdır.
Modellerin performansını değerlendirmek için aşağıdaki üç yaklaşım izlenmektedir.

3.3.3.1. Çok terimli sınıflandırma

Modellerin performansını değerlendirmek için bu yaklaşımda tüm algoritmalar, önceki aşamadaki her saldırı tipi için üretilen veri kümelerine, ikinci aşamada bu veri kümeleri için çıkarılan özellikler kullanılarak uygulanmaktadır. Bu yaklaşım, farklı saldırı türlerine ilişkin olarak her bir algoritmanın etkinliğini gözlemlemek için takip edilmektedir.

3.3.3.2. Çok terimli yaklaşımı için çıkartılan özelliklerle ikili sınıflandırması

Bu yaklaşımda, tüm algoritmaların performansı, tüm veri kümesine göre değerlendirilmektedir. Başka bir deyişle, modeller tüm veri kümesine uygulanır ve çok-terimli yaklaşımındaki her saldırı tipi için çıkarılan özellikler kullanılarak verileri

ya saldırı ya da benign (ikili sınıflandırması) olarak sınıflandırmaktadır. Yani her saldırı için en büyük öneme sahip özellikler, ikili sınıflandırması için bir özellik havuzu elde etmek için birleştirilmektedir. Bunu yaparken her veri kümesi için 48 özelliğe sahip iki ayrı havuz elde edilir. Tekrarları kaldırdıktan sonra CIC-IDS-2017 veri kümesi için 18 özellik kalırken CSE-CIC-IDS-2018 veri kümesi için yalnızca 16 özellik kalmaktadır. Tablo 3.7. her iki veri kümesinin özelliklerini listelemektedir.

Tablo 3.7. İkinci yaklaşımda her iki veri kümesi için özelliklerin listesi

NO	Veri Kümesi	Özellikler
1	CIC-IDS-2017	Bwd Packet Length Mean, Flow Duration, Flow IAT Std, Flow IAT Min, Bwd Packet Length Std, Total Backward Packets, Fwd IAT Total, Total Length of Fwd Packets, Flow IAT Max, Fwd Packet Length Std, Fwd Packet Length Max, Flow IAT Mean, Fwd Packet Length Mean, Fwd Packet Length Min, Total Length of Bwd Packets, Total Fwd Packets, Bwd Packet Length Max, Flow Bytes/s
2	CSE-CIC-IDS-2018	Bwd Pkt Len Mean, Bwd Pkt Len Std, Flow Pkts/s, Flow IAT Min, Tot Bwd Pkts, Flow Duration, Fwd IAT Tot, TotLen Fwd Pkts, Tot Fwd Pkts, Flow IAT Max, Fwd Pkt Len Max, Flow IAT Mean, Fwd Pkt Len Std, Flow Byts/s, Flow IAT Std, Fwd Pkt Len Mean

3.3.3.3. İkili yaklaşımı için çıkartılan özelliklerle ikili sınıflandırması

İkinci yaklaşıma benzer şekilde, bu bölüm MÖ algoritmalarını tüm veri kümesine uygular ve burada özellik çıkarma safhasında bu yaklaşım için çıkarılan özelliklerle ikili bir sınıflandırma (Saldırı veya Benign) gerçekleştirilmektedir. Her iki veri kümesi için de en yüksek önem değerine sahip 20 özelliğin bir listesi Özellik Çıkarma bölümünde bulunmaktadır.

[98], bir veri kümesine MÖ algoritmaları uygularken, eğitim ve test için harcanan zamanın azalmasıyla ilgilenecek, özellik sayısının daha az olduğunu belirtmektedir. Bu göz önüne bulundurularak, bu yaklaşımdaki özellik sayısı azaltılmaya çalışılmıştır. Hesaplamalar, her iki veri kümesi için de en yüksek önem değerine sahip ilk yedi özelliğin toplam önemin %96'sından fazlasını oluşturduğunu göstermektedir. Bu nedenle, burada kullanılan özellik sayısı da bu hesaplamaya göre ayarlanmıştır.

BÖLÜM 4. ARAŞTIRMA BULGULARI ve TARTIŞMA

Bu bölüm, her iki veri kümesinin değerlendirilmesinin nihai sonuçlarını sunar ve önceki bölümde incelenen model uygulamalarının üç yaklaşımı açısından yedi MÖ algoritmasının hepsinin etkinliğini karşılaştırmaktadır. Her iki veri kümesi ile tüm yaklaşımlardaki tüm modellerin performans değerlendirmesinin sonuçları ve bıyık grafikleri hakkında daha fazla ayrıntı Ekler bölümünde bulunmaktadır.

4.1. CIC-IDS-2017

Bu bölümde, tüm algoritmaların performansı, CIC-IDS-2017 veri kümesi üzerinde, algoritmaların doğruluğu ana kriter olarak göz önüne bulundurarak analiz edilir, ancak hangi algoritmanın diğerlerinden daha iyi performans gösterdiğini anlamak için f-ölçü, hassasiyet, geri çağırma ve eğitim süresi de dikkate alınmaktadır.

4.1.1. Çok terimli sınıflandırması sonuçları

Burada, her modelin veri kümesinde bulunan saldırı türlerine göre etkinliği gösterilmektedir. Tablo 4.1. Çok Terimli Sınıflandırma Yaklaşımı için nihai sonuçları sunmaktadır.

Bu yaklaşımın sonuçlarında görülebileceği gibi, çoğu durumda J48, RF, AB ve KNN dahil dört algoritma %90'ın üzerinde bir doğruluk elde edebilirken, diğer üç algoritma çoğu durumda kötü bir doğruluk göstermektedir. J48, 9 saldırı tipinde diğer altı algoritmadan daha iyi performans elde ederek en iyi performansı göstermektedir. Bazı durumlarda J48, diğer algoritmalarla aynı doğruluğu paylaşmaktadır, ancak onu diğerlerinin önüne geçiren kriter düşük eğitim süresidir. RF, düşük işlem süresi olan J48 ile aynı kriterleri paylaşan algoritmalar arasında ikinci sırada yer almaktadır. RF,

Web Saldırıları ile en iyi performansı elde etmektedir. AB ve KNN, daha uzun işlem süresi haricinde J48 ve RF ile karşılaştırılabilir performanslara sahiptir.

Tablo 4.1. Çok terimli sınıflandırma yaklaşımı sonuçları

Saldırı Tipi	Model Doğruluğu						
	J48	RF	AB	KNN	SVM	MLP-NN	NB
DoS Hulk	%97	%95	%96	%96	%52	%77	%83
PortScan	%100	%100	%100	%100	%81	%77	%41
DDoS	%97	%97	%97	%93	%88	%79	%82
DoS GoldenEye	%99	%99	%98	%98	%51	%75	%51
FTP-Patator	%100	%100	%100	%100	%100	%100	%100
SSH-Patator	%96	%96	%96	%95	%55	%86	%44
DoS Slowloris	%96	%94	%95	%95	%49	%74	%42
DoS Slowhttptest	%98	%97	%97	%97	%94	%79	%93
Bot	%97	%97	%98	%96	%67	%79	%53
Web Attack	%95	%95	%95	%95	%50	%92	%49
Infiltration	%96	%95	%89	%92	%88	%49	%79
Heartbleed	%100	%100	%100	%100	%100	%82	%100

Burada önemli bir nokta, FTP-Patator saldırısı durumunda, tüm algoritmaların bu saldırı için %100 sınıflandırma doğruluğu elde etmesidir. Bu, seçilen özelliklerin etkisinin sonucudur. Bu özellikler, FTP-Patator saldırısı durumunda normal trafikten anormallikleri belirtmek için önemli bir araç olarak kullanılabilecektir. Aynı hipotez, J48, RF, KNN ve AB ile Portscan saldırısı için de geçerlidir.

Diğer bir önemli nokta, SVM'nin tüm algoritmalar arasında ikinci en kötü yere sahip olmasına rağmen, FTP-Patator saldırısıyla en iyi performansı göstermesidir. Bu, SVM'nin bu durumda en düşük işlem süresine sahip olmasından kaynaklanmaktadır.

Heartbleed saldırısı ile MLP-NN dışındaki tüm algoritmalar %100 sınıflandırma doğruluğuna ulaşır, ancak CIC-IDS-2017 veri kümesinde bulunan bu tür saldırılarla ilişkili az sayıda bulunan örnek nedeniyle bu saldırının sonuçları güvenilir değildir.
Ayrıca, tüm algoritmalar arasında SVM ve NB'nin en hızlı olduğunu belirtmek gerekir.

4.1.2. Çok terimli özelliklerle ikili sınıflandırmasının sonuçları

Bu bölümde, tüm veri kümesine göre ya anomali ya da normal olan örneklere göre sınıflandırma sonuçları sunulmaktadır. Bu adımda önceki bölümde belirtildiği gibi, kullanılan özellikler her saldırı türü için çıkarılan özelliklerle aynıdır. Tablo 4.2.'de

görüldüğü gibi, bu yaklaşımda KNN'nin %97'lik en iyi sınıflandırma doğruluğunu elde ettiği görünmektedir. AB, F-ölçütü değerinde küçük bir farkla J48'den daha iyi performans gösterir, ancak J48, AB'den çok daha hızlıdır. SVM her durumda en kötü performansa sahiptir. Tüm algoritmalar arasında NB ve SVM en hızlı, MLP-NN ve KNN en yavaştır.

Tablo 4.2. Çok terimli özellikler yaklaşımı ile CICIDS-2017 ikili sınıflandırması sonuçları

NO	Algoritma	Doğruluk	F-ölçütü	Geri Çağırma	Hassasiyet	Zaman (saniye)
1	KNN	%97	%93	%92	%96	1214,6431
2	AB	%95	%91	%87	%95	461,329
3	J48	%95	%90	%86	%97	29,3225
4	RF	%94	%88	%84	%96	31,3615
5	MLP-NN	%84	%50	%52	%62	1449,7386
6	NB	%78	%63	%64	%63	6,1811
7	SVM	%31	%31	%58	%58	9,9559

4.1.3. İkili yaklaşımı özelliklerle ikili sınıflandırması sonuçları

Burada, özellik çıkarma aşamasındaki ikili sınıflandırma yaklışımı için çıkarılan özelliklerle CIC-IDS-2017 veri kümesinin tümüne uygulanan ikili sınıflandırmasının sonuçları sunulmaktadır.

Tablo 4.3.'te görüldüğü gibi. bu yaklaşımda elde edilen sonuçlar, önceki bölümde elde edilen sonuçlardan çok farklı değildir. KNN yine tüm algoritmalar arasında ilk sıradaki yerini korumaktadır. NB ve SVM'nin sınıflandırma doğruluğunda küçük bir artış vardır.

Tablo 4.3. CICIDS-2017'in ikili özelliklerle sınıflandırması sonuçları

NO	Algoritma	Doğruluk	F-ölçütü	Geri Çağırma	Hassasiyet	Zaman (saniye)
1	KNN	%97	%94	%93	%95	1065,3153
2	J48	%95	%91	%89	%93	13,4433
3	AB	%94	%88	%85	%93	217,1472
4	RF	%94	%86	%81	%96	30,5459
5	MLP-NN	%84	%52	%54	%75	1355,3965
6	NB	%82	%65	%64	%66	3,8343
7	SVM	%38	%38	%61	%58	4,5624

Buradaki en önemli gelişme, tüm algoritmaların işlem süresindeki azalmadır. Bunun nedeni, burada kullanılan toplam özellik sayısının yedi'ye indirilmesidir. Bununla

birlikte, MLP-NN, NB ve SVM için elde edilen tatmin edici olmayan sonuçların da nedeni bu olabilir. Bu algoritmaların sınıflandırma doğruluğunu artırmak için ikili sınıflandırması için çıkarılan 20 özellik inceleniyor ve bu algoritmalar açısından önem değerleri ölçülmektedir. Bu süreçte, sadece önceden seçilen özelliklerden daha yüksek puan alan özellikler seçilmektedir. Bu şekilde, bu algoritmaların her biri için ayrı bir özellik alt kümesi çıkarılır. Yeni çıkarılan özellikler daha sonra her bir algoritma için ayrı ayrı yedi özellik havuzuyla birleştirilir. Tablo 4.4. bu özellikleri temsil etmektedir.

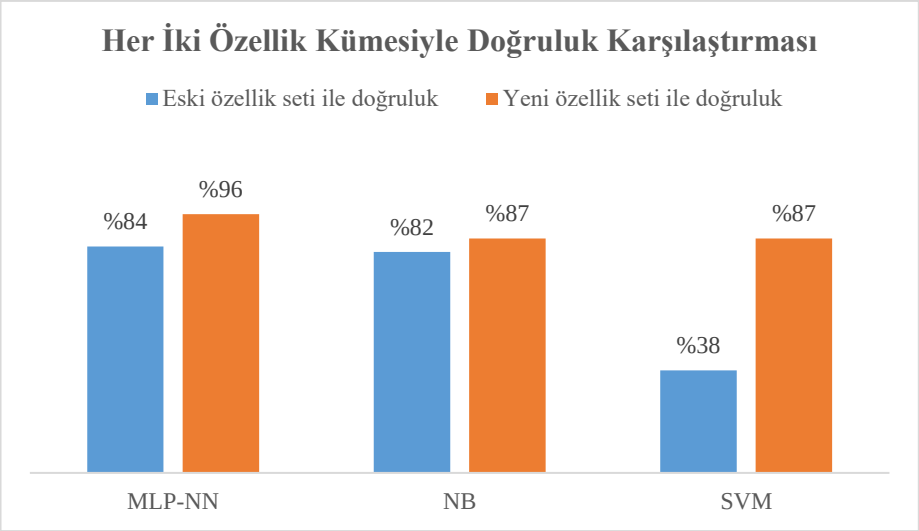
Tablo 4.4. MLP-NN, NB ve SVM için yeni özelliklerin listesi

NO	Algoritma	Yeni Seçilen Özellikler
1	MLP-NN	Total Length of Fwd Packets, Total Backward Packets, Flow IAT Min, Bwd Packet Length Max, Bwd Packet Length Mean, Fwd Packet Length Mean, Total Fwd Packets, Fwd Packet Length Max.
2	NB	Flow Packets/s, Bwd Packet Length Std, Flow IAT Min, Fwd Packet Length Min, Fwd Packet Length Mean, Total Length of Fwd Packets.
3	SVM	Flow Packets/s, Flow Bytes/s, Bwd Packet Length Std, Flow IAT Min,.

Tablo 4.5. bu yaklaşımdaki tüm algoritmaların yeni özellik alt kümeleriyle değerlendirilmesinin sonuçlarını göstermektedir. Bu yöntemin, MLP-NN başta olmak üzere üç zayıf algoritmanın hepsinin sonuçlarını büyük ölçüde geliştirdiği görülebilmektedir. MLP-NN'nin sınıflandırması J48, Rf ve AB'den daha iyi performans göstererek %84'ten %96'ya yükselmiştir. SVM'nin doğruluğu da %38'den %87'ye yükseldi. NB'ye gelince, %82'den %87'ye küçük bir artış görünmektedir.

Tablo 4.5. Geliştirilmiş özelliklerle CICIDS-2017 ikili sınıflandırmasının sonuçları

NO	Algoritma	Doğruluk	F-ölçütü	Geri Çağırma	Hassasiyet	Zaman (Saniye)
1	KNN	%97	%94	%93	%95	1070,9417
2	MLP-NN	%96	%92	%90	%95	5667,7409
3	J48	%95	%91	%89	%93	13,4761
4	RF	%94	%88	%84	%97	30,5569
5	AB	%94	%88	%85	%93	217,1866
6	NB	%87	%73	%70	%79	3,6389
7	SVM	%87	%73	%69	%79	3,5671



Şekil 4.1. MLP-NN, NB ve SVM'nin performansının eski ve yeni özellik havuzlarıyla karşılaştırılması.

Şekil 4.1.'de, özellik çıkarmanın algoritmaların doğruluğunda büyük bir rolü olduğu sonucuna varılabilir. Her algoritma farklı bir özellik alt kümesiyle daha iyi çalışabilmektedir. Bu, MÖ algoritmaları kullanılarak ağ anomali tespiti bağlamında akılda tutulması gereken çok iyi bir noktadır.

4.2. CSE-CIC-IDS-2018

Bu bölümde, tüm algoritmaların performansı, CSE-CIC-IDS-2018 veri kümesi üzerinde ana kriter olarak Doğruluk göz önünde bulundurularak analiz edilir, ancak hangi algoritmanın daha iyi performans gösterdiğini anlamak için f-ölçü, hassasiyet, geri çağırma ve eğitim süresi de dikkate alınır.

4.2.1. Çok terimli sınıflandırması sonuçları

Tablo 4.6. CSE-CIC-IDS-2018 veri kümesinde bulunan her bir saldırı türü için oluşturulan ayrı veri kümeleri üzerinde analiz edilen yedi makine öğrenmesi algoritmasının hepsinin çok terimli sınıflandırma ve değerlendirmesinin sonuçlarını sunmaktadır.

Tablo 4.6. CIC-IDS-2018 çok terimli sınıflandırma yaklaşımı sonuçları

Saldırı Tipi	Model Doğruluğu						
	J48	KNN	AB	RF	MLP-NN	SVM	NB
DDOS attack-HOIC	%98	%98	%98	%98	%95	%75	%59
DDoS attacks-LOIC-HTTP	%99	%99	%99	%99	%71	%45	%40
DoS attacks-Hulk	%98	%98	%98	%98	%96	%75	%30
Bot	%99	%100	%99	%99	%99	%78	%44
FTP-BruteForce	%98	%98	%98	%98	%98	%29	%61
SSH-Bruteforce	%99	%99	%99	%99	%93	%74	%57
Infilteration	%74	%74	%74	%74	%72	%48	%33
DoS attacks-SlowHTTPTest	%98	%98	%98	%98	%94	%28	%60
DoS attacks-GoldenEye	%98	%98	%98	%96	%70	%56	%37
DoS attacks-Slowloris	%93	%90	%93	%93	%90	%61	%57
DDOS attack-LOIC-UDP	%100	%100	%100	%100	%95	%30	%100
Web Attacks	%90	%91	%92	%90	%79	%55	%47

Sonuçlar, J48'in yedi saldırı türünde (DDOS attack-HOIC, DDoS attacks-LOIC-HTTP, DoS attacks-Hulk, FTP-BruteForce, SSH-Bruteforce, DoS attacks-SlowHTTPTest, DDOS attack-LOIC-UDP) diğer tüm algoritmalardan daha iyi performans elde ettiğini göstermektedir. Burada yine düşük eğitim süresi J48'i KNN, AB ve RF'nin önüne koymaktadır. KNN, üç saldırıda en yüksek doğruluğu elde ederek tüm algoritmalar arasında en iyi ikinci sırayı almaktadır. KNN, Bot saldırısında mükemmel bir puan almaktadır. AB, DoS Attacks-Slowloris ve Web saldırılarında en iyi performansı göstermektedir. RF, J48, KNN ve AB ile birlikte tüm durumlarda tatmin edici sonuçlar elde eder, ancak bunların hiçbirinden daha iyi performans göstermemektedir.

J48, KNN, AB ve RF, Infiltration saldırısı hariç tüm durumlarda %90'ın üzerinde bir doğruluk elde edebilmektedir. Bunun nedeni, bu saldırıyı tanımlayan özelliklerin normal trafiği tanımlayanlarla da çok benzer olması ve bu bağlamda daha belirgin bir özellik çıkarma işlemine ihtiyaç duyulmasıdır.

MLP-NN çoğu durum için iyi sonuçlar gösterir, ancak DDoS attacks-LOIC-HTTP, Infilteration, DoS attacks-GoldenEye ve Web saldırıları ile iyi bir performans elde etmemektedir.

SVM ve NB, tüm algoritmaların en zayıflarıdır, ancak aynı zamanda en hızlılarıdır. Bu iki algoritmanın kötü performansının arkasındaki neden, büyük veri kümelerini işleyememesidir. Veri kümesi ne kadar büyük olursa, bu algoritmaların

oluşturabileceği model o kadar karmaşık hale gelir ve böylece tatmin edici sonuçlar elde edilememektedir.

Bir başka ilginç nokta ise, MLP-NN ve SVM dışındaki tüm algoritmaların DDOS Attacks-LOIC-UDP saldırısı açısından %100'e ulaşmasıdır. Bunun nedeni, bu tür saldırılar için en uygun özellik seçimidir.

Tüm algoritmalar, diğer saldırı türlerine kıyasla Web saldırısında daha düşük bir doğruluk elde etmektedir. Bunun nedeni, CSE-CIC-IDS-2018 veri kümesinde bulunan bu tür saldırı örneklerinin az sayıda var olmasından kaynaklanmaktadır.

4.2.2. Çok terimli özelliklerle ikili sınıflandırmasının sonuçları

Bu bölümde, CIC-IDS-2018 veri kümesine ilişkin yedi algoritmanın performansı bir bütün olarak değerlendirilmektedir. Burada yapılan sınıflandırma, algoritmaların verileri saldırı veya normal olarak sınıflandırdığı ikili bir anlama sahiptir. Burada kullanılan özellik grubu, çok terimli sınıflandırma için her saldırı türü için çıkarılan özelliklerle aynıdır. Tekrarlar kaldırıldıktan sonra yalnızca 16 özellik havuzda kalmaktadır. Tablo 4.7., bu yaklaşımda KNN'nin %95'lik en iyi doğruluğu elde ettiğini göstermektedir. KNN, J48 ile tüm durumlarda aynı puanları paylaşırken, onun önüne koyan kriter daha yüksek Geri Çağırma değeridir. Ancak J48, KNN'den çok daha hızlıdır. AB ve RF'nin her ikisi de %94'lük bir doğruluk elde ediyor ve bu da onları bu bağlamda ikinci en güçlü algoritma yapmaktadır. En zayıf algoritmalar MLP-NN, NB ve SVM'dir. Bu algoritmalar bu yaklaşımda tatmin edici bir sonuç elde edememektedir. KNN tüm algoritmaların en yavaşı, NB ve SVM ise en hızlısıdır.

Tablo 4.7. Çok terimli özelliklerle CICIDS-2018 ikili sınıflandırması sonuçları

NO	Algoritma	Doğruluk	F-ölçütü	Geri Çağırma	Hassasiyet	Zaman (Saniye)
1	KNN	%95	%95	%95	%95	5534,6889
2	J48	%95	%95	%94	%95	68,1705
3	AB	%94	%94	%94	%94	1110,023
4	RF	%94	%94	%93	%94	105,3846
5	MLP-NN	%73	%64	%66	%72	2965,0478
6	NB	%46	%40	%55	%65	63,472
7	SVM	%37	%35	%35	%46	67,894

4.2.3. İkili yaklaşımı özelliklerle ikili sınıflandırması sonuçları

Bu bölümde, CSE-CIC-IDS-2018 veri kümesi için üçüncü model uygulama yönteminin sonuçları bulunmaktadır. Daha önce de belirtildiği gibi, burada kullanılan özellikler ikili sınıflandırma yaklaşımı için seçilen özelliklerdir ve sayıları yediye indirilmiştir. Tablo 4.8. bu yaklaşımın sonuçlarını göstermektedir.

Tablo 4.8. CICIDS-2018'in ikili özelliklerle sınıflandırması sonuçları

NO	Algoritma	Doğruluk	F-ölçütü	Geri Çağırma	Hassasiyet	Zaman (Saniye)
1	KNN	%95	%94	%94	%94	3940,106
2	J48	%93	%93	%93	%93	40,7142
3	AB	%93	%93	%93	%93	627,1437
4	RF	%92	%92	%92	%92	91,7004
5	MLP-NN	%70	%59	%63	%77	2005,3104
6	SVM	%46	%40	%55	%68	14,0871
7	NB	%45	%40	%54	%63	10,2669

Sonuçlar %95 doğrulukla KNN'nin en güçlü algoritma olduğunu, bunu sırasıyla %93, %93 ve %92 doğrulukla J48, AB ve RF izlediğini göstermektedir. J48 ve AB aynı sonuçlara sahiptir, ancak J48 çok daha hızlıdır, bu yüzden AB'den daha iyi performans elde etmektedir. Azalan özellik sayısı, önceki bölümle karşılaştırıldığında tüm algoritmaların işlem süresini önemli ölçüde azaltmıştır. Burada yine KNN en yavaş algoritma iken NB en hızlı algoritmadır.

MLP-NN, SVM ve NB iyi sonuçlar elde edemeyip, en zayıf algoritmalardır. Performansları artırmak için CIC-IDS-2017 ile aynı yaklaşım izlemenktedir. Tablo 4.9. her algoritma için havuza eklenen yeni özellikleri açıklamaktadır.

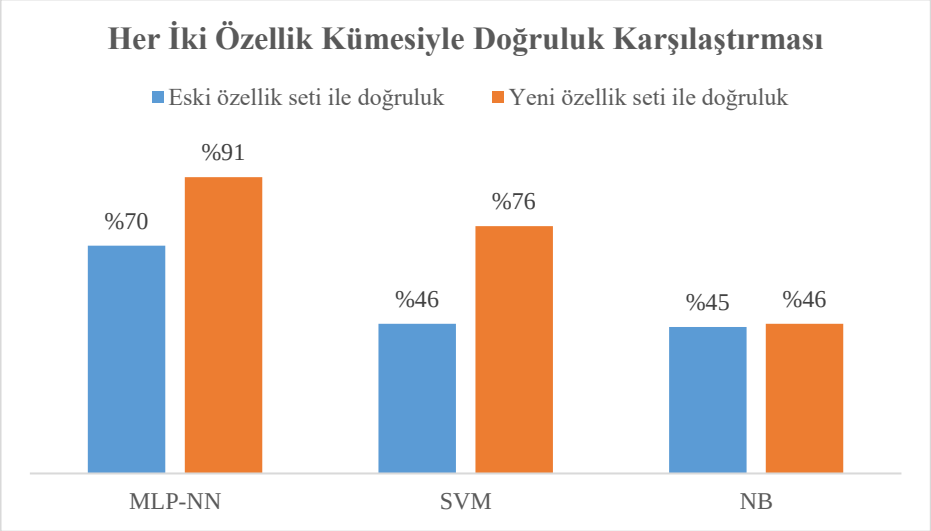
Tablo 4.9. MLP-NN, NB ve SVM için yeni özelliklerin listesi

NO	Algoritma	Yeni Özellikler
1	MLP-NN	Tot Len Fwd Pkts, Bwd Pkt Len Mean, Flow IAT Min, Bwd Pkt Len Max, Fwd Pkt Len Max, Flow Pkts/s, Tot Fwd Pkts.
2	NB	Flow Duration, Tot Len Fwd Pkts, Bwd Pkt Len Mean, Flow IAT Max, Bwd Pkt Len Max, Flow Pkts/s, Fwd IAT Tot, Fwd Pkt Len Max, Tot Fwd Pkts, Bwd Pkt Len Std, Fwd Pkt Len Std, Fwd Pkt Len Mean, TotLen Bwd Pkts, Fwd Pkt Len Min, Tot Bwd Pkts, Bwd Pkt Len Min.
3	SVM	Bwd Pkt Len Mean, Flow Pkts/s, Fwd Pkt Len Max, Bwd Pkt Len Max, Fwd Pkt Len Std, Fwd Pkt Len Mean, Fwd Pkt Len Min, Bwd Pkt Len Min.

Tablo 4.10. MLP-NN, SVM ve NB için bu yeni özelliklerin havuzlarıyla CIC-IDS-2018 veri kümesinin değerlendirilmesinin sonuçlarını sunmaktadır.

Tablo 4.10. Geliştirilmiş özelliklerle CICIDS-2018 ikili sınıflandırması sonuçları

NO	Algoritma	Doğruluk	F-ölçütü	Geri Çağırma	Hassasiyet	Zaman (Saniye)
1	KNN	%95	%94	%94	%94	3942,8224
2	J48	%93	%93	%93	%93	40,6613
3	AdaBoost	%93	%93	%93	%93	626,5998
4	RF	%92	%92	%91	%92	89,9017
5	MLP-NN	%91	%90	%91	%90	5818,1916
6	SVM	%76	%76	%79	%78	13,3777
7	NB	%46	%41	%55	%66	53,2904



Şekil 4.2. MLP-NN, NB ve SVM'nin performansının eski ve yeni özellik havuzlarıyla karşılaştırılması

Şekil 4.2. bu yöntemin MLP-NN ve SVM performansını büyük ölçüde geliştirdiğini göstermektedir. MLP-NN'nin doğruluğu %70'ten % 91'e, SVM'nin ise %46'dan %76'ya artmıştır. Bununla birlikte, doğruluk %45'ten %46'ya değiştiğinden, NB'nin sonuçları fazla değişiklik göstermemektedir.

Daha önce de belirtildiği gibi, belirlenmiş algoritmalara özgü özellik çıkarma teknikleri ve hangi özelliklerin anomalileri daha iyi tanımladığı konusunda daha fazla araştırmaya ihtiyaç vardır.

4.3. Değerlendirme

Elde edilen sonuçları değerlendirmek ve karşılaştırmak için literatürden iki çalışma seçilmiştir.

Boukhamla [99], CIC-IDS-2017 veri kümesini optimize etmek için PCA algoritmasını kullanarak, böylece geri çağırma ve hassasiyette herhangi bir azalma olmadan veri kümesinin boyutsallığını (özelliklerini) ve boyutunu (örneklerini) azaltmıştır. Daha sonra, optimize edilmiş veri kümesi C4.5, KNN ve NB algoritmaları kullanılarak değerlendirilmiştir. Bu çalışmanın sonuçları, algoritma doğruluğu dikkate alınarak Tablo 4.11.'de bizimkilerle karşılaştırılmaktadır.

Tablo 4.11. CIC-IDS-2017 sonuçlarının literatürdeki başka bir çalışma ile karşılaştırılması

Saldırı Tipi	Boukhamla et al. [99]			Bizim çalışmamız		
	C4.5	KNN	NB	J48	KNN	NB
PortScan	%99,9	%99,9	%98	%100	%100	%41
DDoS	%87,9	%90,6	%99	%97	%93	%82
Bot	%97	%98,8	%51,8	%97	%96	%53
Web Attack	%57,7	%57,9	%61,1	%95	%95	%49
Infiltration	%47,2	%47,2	%75	%96	%92	%79

Karşılaştırma, çalışmamızın çoğu durumda J48 ve KNN ile daha iyi sonuçlar elde ettiğini göstermektedir. [99] 'da NB'nin daha yüksek doğruluğu, büyük olasılıkla bu algoritmayı eğitmek ve test etmek için kullanılan optimize edilmiş CIC-IDS-2017 veri kümesindeki daha az sayıda örnek ve özellik var olamsından kaynaklanmaktadır.

Çalışmamızın CSE-CIC-IDS-2018 veri kümesi ile çok terimli yaklaşımının sonuçları, literatürden bir başka çalışma ile karşılaştırılmıştır, Alsamiri [100]. Çalışmamız ve onların çalışması arasında birçok fark vardır; ilk olarak kullandıkları veri kümesi, CSE-CIC-IDS-2018 veri kümesini kullanarak IoT ortamında DoS ve DDoS saldırıları için oluşturulan bir veri kümesi olan Bot IoT'dur. Bununla birlikte, hem Bot IoT hem de CSE-CIC-IDS-2018'deki saldırıların çoğu aynı özelliklere sahiptir. Bir başka önemli fark ise, biz orijinal özellik setini kullanırken, onlar yeni bir özellik seti çıkarmak için CIC-FLOW-METER kullanmış olmalarıdır. Tablo 4.12. çalışmalar [100] arasındaki karşılaştırmayı, F ölçütü temel değerlendirme kriteri alarak,

göstermektedir. Tabloda görüldüğü gibi, çalışmamız NB ve MLP dışındaki tüm benzer algoritmalarda ve saldırı türlerinde daha iyi sonuçlar vermiştir.

Tablo 4.12. CSE-CIC-IDS-2018 sonuçlarının literatürdeki başka bir çalışma ile karşılaştırılması

Saldırı Tipi	Alsamiri et al.[100]					Bizim çalışmamız				
	KNN	AB	RF	MLP	NB	KNN	AB	RF	MLP	NB
DDOS (LOIC HTTP)	%96	%96	%96	%95	%72	%99	%99	%99	%71	%40
DDOS (LOIC UDP)	%98	%98	%98	%97	%73	%100	%100	%100	%95	%100
DoS (Slow HTTPTest)	%96	%95	%95	%95	%72	%98	%98	%98	%94	%60

BÖLÜM 5. SONUÇ

Bu çalışmanın amacı bilgisayar ağı bağlamında anomali tespiti için en etkili yolu belirlemektir. Bu hedefi gerçekleştirmek için daha önce bu alanda yapılan çalışmaların kapsamlı bir incelemesi yapılmıştır. Literatür araştırılmasından sonra ağ anomalisi tespitinde önemli olan aşağıdaki başlıklar için en uygun seçenekler belirlenmiştir.

- Tespit yöntemi: Tüm tespit yöntemleri arasında Makine Öğrenmesi yaklaşımları literatürde en başarılı ve en çok kullanılan yaklaşımdır.
- Öğrenme tekniği: Her üç öğrenme tekniğinden (Denetimli, Denetimsiz ve Güçlendirme) denetimli sınıflandırma ve denetimsiz kümeleme kategorileri anomali tespiti için kullanılabilir. Anomali tespiti bir sınıflandırma problemi olduğundan ve literatürde kümelenmeden daha fazla başarı gösterdiğinden, çalışmamızda bu yöntemi kullanmaya karar verilmiştir.
- Veri kümeleri: Önceki çalışmalarda kullanılan veri kümelerinin çoğu güncel değildir ve artık gerçek dünya verilerini temsil edememektedir. Araştırma camiasından birçok eleştiri almasının nedeni de budur. Bu nedenle, daha güncel ve genelleştirilmiş CIC-IDS-2017 ve CSE-CIC-IDS-2018 veri kümelerini kullanmaya karar verilmiştir.
- Değerlendirme metrikleri: algoritmaların performansını değerlendirmek için ‘Doğruluk’ ana değerlendirme metriği olarak seçilmiş olup, ancak F-ölçüsü, Geri Çağırma, Hassasiyet ve İşleme Süresi de dikkate alınmıştır.
- Algoritmalar: J48, KNN, AB, RF, NB, MLP-NN ve SVM.
- Ön işleme: Uygulama aşamasında hataya neden olabilecek küçük verimsizlikleri çözmek için veri kümeleri önişlenmiştir. Ayrıca sınıflandırmayı çok terimli (saldırı tipi) ve ikili (normal veya saldırı) olmak üzere iki şekilde yapmaya karar verildiğinden dolayı, tüm CSV dosyalarından bir büyük veri

kümesi ve her saldırı türü için ayrı bir veri kümesi grubu oluşturulmuştur. Bu işlem her iki veri kümesi için de yapılmıştır.

- Özellik seçimi: Her iki sınıflandırma yaklaşımının özelliklerini ayrı ayrı ayıklamak için Random Forest Regressor algoritması kullanılmıştır. En yüksek önem değerine sahip özellikler, her saldırı türü için ve ardından tüm veri kümesi için seçilmiştir.
- Sınıflandırma yaklaşımları: Uygulama aşamasında üç yaklaşım uygulanmıştır; Saldırı tipine göre sınıflandırma , çok terimli özelliklerle ikili sınıflandırma ve ikili yaklaşım için çıkarılan özelliklerle ikili sınıflandırma. Veri kümeleri %80 eğitim ve %20 test oranı ile iki parçaya ayrılmıştır. MÖ algoritmalarını eğitmek ve test etmek için 10 kat çapraz doğrulama (10-fold cross validation) yöntemi kullanılmıştır.

Bu çalışma sırasında J48, RF, KNN ve AB'nin her durumda yüksek doğruluk elde edebileceğini ve iyi performans gösterebileceği öğrenilmiştir. Çok terimli sınıflandırma yaklaşımıyla J48, binom sınıflandırmasında ise KNN öncülük ederken diğerlerinden daha iyi performans göstemektedir.

MLP-NN, SVM ve NB, her iki yaklaşımda da çoğu durumda kötü sonuçlar vermektedir, ancak bu algoritmalara daha iyi doğruluk sağlayan özellikleri yeniden değerlendirdikten ve özellik havuzlarını güncelledikten sonra, doğruluklarında önemli bir artış gözlenmektedir. Bu özellik çıkarma ve uygun özelliklerin seçilmesinin, ağ anomalisi tespiti bağlamında bir MÖ modelinin performansında büyük rol oynadığını kanıtlamaktadır.

Sonuçlar, ayrıca algoritmaların çok terimli sınıflandırma (saldırı tipine göre) yaklaşımında binom yaklaşımından daha iyi doğruluk sağlayabildiğini göstermektedir. Bu, daha yüksek TP (Gerçek Positif) ve doğruluk sağlayabilen anomali tabanlı bir IDS tasarlamak için saldırı tipine göre sınıflamanın daha iyi sonuçlar verdiğini göstermektedir.

Son olarak, anomali tespiti için mükemmel olan tek bir algoritma veya yaklaşım olmadığı sonucuna varılabilir. Bazı algoritmalar, belirli saldırı kategorileri için olağanüstü performans gösterirken, diğerleri farklı kategorilerde başarılıdır. Bu da ilerideki çalışmalarda yapılması gerekenleri andırmaktadır. Belki de gerçek zamanlı olarak herhangi bir anomaliyi tespit edebilen evrensel bir saldırı tespit sistemine ihtiyaç duyulmaktadır. Geliştirilecek olan bu sistem aşağıdaki özelliklere sahip olması gerekmektedir.

- Çok Katmalı: model birden çok katmandan (modüller) oluşacaktır. Bu katmanlar şunları içerebilir: ön işleme ünitesi, özellik çıkarma ünitesi, binom sınıflandırma ünitesi, çok terimli sınıflandırma ünitesi ve çevre.
- Gerçek Zamanlı: Verileri işlemek ve algoritmaları gerçek zamanlı olarak eğitmek için Apache Spark ve makine öğrenimi kütüphanesi gibi bulut teknolojilerin kullanılması.
- Kendi Kendine Öğrenme: Aktif öğrenme veya pekiştirme öğrenme (Ensemble Learning) gibi yaklaşımlar kullanılabilir, böylece sistem yeni tip anomali ile karşılaştığında insan müdahalesi olmadan modeli eğitebilir.
- Çoklu Sınıflandırma Katmanları: bir hipotez olarak, sistem ilk başta normal ve saldırı verileri arasında ayrım yapmak için ağ trafiği üzerinde ikili sınıflandırma yapacak şekilde çalışabilir. Bir anomali ile karşılaşılırsa, daha fazla analiz için çok terimli sınıflandırma katmanına aktarılmalıdır.
- Ensemble: Sınıflandırma katmanları, verileri sınıflandırmak için birden fazla algoritmanın kullanıldığı ve en iyi sonucun Çoğunluk Oylaması ile elde edildiği için, Ensemble öğrenmesi kullanılmalıdır. Ensemble öğrenmesi, algoritmaların performansında ve doğruluğunda büyük gelişme göstermiştir.
- Hızlı ve Verimli: düşük işlem süresine ve verimli bir modele sahip olmak için J48, SVM veya bunların bir grubu gibi hızlı ve güçlü sınıflandırıcılar ikili sınıflandırma ve RF, KNN, AB daha güçlü ama daha yavaş algoritmalar veya bunların bir topluluğu çok terimli sınıflandırma için kullanılabilir.

KAYNAKLAR

[1] https://www.statista.com/statistics/471264/iot-number-of-connected-devices-worldwide/., Erişim Tarihi: 06.08.2019.

[2] https://www.internetworldstats.com/stats.htm., Erişim Tarihi: 06.08.2019.

[3] https://www.internetadvisor.com/key-internet-statistics., Erişim Tarihi: 07.08.2019.

[4] Samrin, R., Vasumathi, D., Review on anomaly based network intrusion detection system. International Conference on Electrical, Electronics, Communication, Computer and Optimization Techniques (ICEECCOT), Mysuru, India, 141-147, 2017.

[5] Taulli, T., Artificial Intelligence basics: a non-technical introduction. İçinde: AI Foundation. 1. Baskı, Springer Science+Business Media, New York, USA, 1-19, 2019.

[6] Ertel, W., Introduction to Artificial Intelligence. İçinde: Introduction. 2. Baskı, Springer International Publishing AG, Cham, Switzerland, 1-20, 2017.

[7] Lucci, S., Kopec, D., Artificial Intelligence in the 21st Century. İçinde: Overview of Artificial Intelligence. 2. Baskı, Mercury Learning and Information, VA, USA, 3-36, 2016.

[8] https://en.wikipedia.org/wiki/Artificial_intelligence., Erişim Tarihi: 21.08.2019.

[9] Mohri, M., Rostamizadeh, A., Talwalkar, A., Foundations of machine learning. İçinde: Introduction. 1. Baskı, The MIT Press, London, England, 1-8, 2012.

[10] C.Müller, A., Guido, S., Introduction to machine learning with Python. İçinde: Supervised Learning. 1. Baskı, O'Reilly Media, CA, USA, 25-127, 2016.

[11] C.Müller, A., Guido, S., Introduction to machine learning with Python. İçinde: Unsupervised Learning and Preprocessing. 1. Baskı, O'Reilly Media, CA, USA, 131-208, 2016.

[12] Bonaccorso, G., Mastering machine learning algorithms. İçinde: Introduction to Semi-supervised Learning. 1. Baskı, Packt Publishing Ltd., Birmingham, UK, 2018.

[13] Bonaccorso, G., Mastering machine learning algorithms. İçinde: Introduction to Reinforcement Learning. 1. Baskı, Packt Publishing Ltd., Birmingham, UK, 2018.

[14] Thottan, M., Ji, C., Anomaly detection in IP networks. IEEE T SIGNAL PROCES, 51(8): 2191-2204, 2003.

[15] https://www.datascience.com/blog/python-anomaly-detection., Erişim Tarihi: 02.09.2019.

[16] Monowar H., Bhattacharyya, D. K., Kalita, J. K., Network anomaly detection: methods, systems and tools. IEEE COMMUN SURV TUT, 16(1): 303-336, 2013.

[17] https://en.wikipedia.org/wiki/Anomaly_detection., Erişim Tarihi: 02.09.2019.

[18] Moustafa, N., Hu, J., Slay, J., A holistic review of network anomaly detection systems: a comprehensive survey. J NETW COMPUT APPL, 128(): 33-55, 2019.

[19] https://en.wikipedia.org/wiki/Denial-of-service_attack., Erişim Tarihi: 04.09.2019.

[20] Ahmed, M., Mahmood, A. N., Hu, J., A survey of network anomaly detection techniques. J NETW COMPUT APPL, 60: 19-31, 2016.

[21] Hasan, D., Cost-sensitive access control for detecting Remote to Local (R2L) and User to Root (U2R) Attacks. INT J COMPUT APPL T, 43(2): 124-129, 2017.

[22] Paliwal, S., Gupta, R., Denial-of-Service, Probing & Remote to User (R2L) Attack Detection using Genetic Algorithm. INT J COMPUT APPL T, 60(19): 57-62, 2012.

[23] Haines, J. W., Rossev, L. M., Lippmann, R. P., Cunninghum, R. K., Extending the DARPA off-line intrusion detection evaluations. Proceedings DARPA Information Survivability Conference and Exposition II. DISCEX'01, Anaheim, CA, USA, 35-45, 2001.

[24] Tavallaee, M., Bagheri, E., Lu, W., Ghorbani, A. A., A detailed analysis of KDDCUP 99 data set. IEEE Symposium on Computational Intelligence for Security and Defense Applications, Ottawa, ON, Canada, 2009.

[25] Monowar H. B., Dhruba K. B., Jugal K. K., Network traffic anomaly detection and prevention: concepts, techniques and tools. 1. Baskı, Springer International Publishing AG, Cham, Switzerland, 71-82, 2017.

[26] https://www.unb.ca/cic/datasets/ids-2017.html., Erişim Tarihi: 17.09.2019.

[27] https://www.unb.ca/cic/datasets/ids-2018.html., Erişim Tarihi: 17.09.2019.

[28] Mehmood, T., Rais, H., Machine learning algorithms in context of intrusion detection. 3rd International Conference on Computer and Information Sciences (ICCOINS), Kuala Lumpur, Malaysia, 369-373, 2016.

[29] Meng, Y., The practice on using machine learning for network anomaly intrusion detection. 2011 International Conference on Machine Learning and Cybernetics, Guilin, China, 571-576, 2011.

[30] Divyatmika, Sreekesh, M., A two-tier network based intrusion detection system architecture using machine learning approach. International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT), Chennai, India, 42-47, 2016.

[31] Mohd, R., Zuhairi, M., Shadil, A., Hassan, D., Anomaly-based NIDS: a review of machine learning methods on malware detection. International Conference on Information and Communication Technology (ICICTM), Kuala Lumpur, Malaysia, 266-270, 2016.

[32] Imamverdiyev, Y., Sukhostat, L., Anomaly detection in network traffic using extreme learning machine. IEEE 10th International Conference on Application of Information and Communication Technologies (AICT), Baku, Azerbaijan, 2016.

[33] Khan, A., Khan, S., Two level anomaly detection classifier. International Conference on Computer and Electrical Engineering, Phuket, Thailand, 2008.

[34] Kwon, D., Natarajan, K., Suh, S., Kim, H., Kim, J., An empirical study on network anomaly detection using convolutional neural networks. IEEE 38th International Conference on Distributed Computing Systems (ICDCS), Vienna, Austria, 1595-1598, 2018.

[35] Limthong, K., Tawsook, T., Network traffic anomaly detection using machine learning approaches. IEEE Network Operations and Management Symposium, Maui, HI, USA, 542-545, 2012.

[36] Karsligil, M., Yavuz, A., Guvensan, M., Hanifi, K., Bank, H., Network intrusion detection using machine learning anomaly detection algorithms. 25th Signal Processing and Communications Applications Conference (SIU), Antalya, Turkey, 2017.

[37] Mirza, A., Computer network intrusion detection using various classifiers and ensemble learning. 26th Signal Processing and Communications Applications Conference (SIU), Izmir, Turkey, 2018.

[38] Zaman, M., Lung, C., Evaluation of machine learning techniques for network intrusion detection. NOMS 2018 - IEEE/IFIP Network Operations and Management Symposium, Taipei, Taiwan, 2018.

[39] Maniriho, P., Ahmad, T., Analyzing the performance of machine learning algorithms in anomaly network intrusion detection systems. 4th International Conference on Science and Technology (ICST), Yogyakarta, Indonesia, 2018.

[40] Sabhnani, M., Serpen, G., Application of machine learning algorithms to KDD intrusion detection dataset within misuse detection context. Proceedings of the International Conference on Machine Learning; Models, Technologies and Applications, Las Vegas, Nevada, USA, 2003.

[41] Liu, M., He, Y., Meng, Q., Wang, Z., Research on anomaly detection of network traffic based on fractal technology and vector quantization. Second International Workshop on Education Technology and Computer Science, Wuhan, China, 428-431, 2010.

[42] Zhao, S., Chandrashekar, M., Lee, Y., Medhi, D., Real-time network anomaly detection system using machine learning. 11th International Conference on the Design of Reliable Communication Networks (DRCN), Kansas City, MO, USA, 267-270, 2015.

[43] Li, W., Li, Q., Using Naive Bayes with AdaBoost to enhance network anomaly intrusion detection. Third International Conference on Intelligent Networks and Intelligent Systems, Shenyang, China, 486-489, 2010.

[44] Abedin, M., Siddiquee, K., Bhuyan, M., Karim, R., Hossein, M., Andersson, K., Performance analysis of anomaly based network intrusion detection systems. IEEE 43rd Conference on Local Computer Networks Workshops (LCN Workshops), Chicago, IL, USA, 1-7, 2018.

[45] Belouch, M., El Hadaj, S., Idhammad, M., Performance evaluation of intrusion detection based on machine learning using Apache Spark. Procedia Comput Sci, 127(2018):1-6, 2018.

[46] Dobson, A., Roy, K., Yuan, X., Xu, J., Performance evaluation of machine learning algorithms in apache spark for intrusion detection. 28th International Telecommunication Networks and Applications Conference (ITNAC), Sydney, NSW, Australia, 2018.

[47] Parampottupadam, S., Moldovann, A., Cloud-based real-time network intrusion detection using deep learning. International Conference on Cyber Security and Protection of Digital Services (Cyber Security), Glasgow, UK, 2018.

[48] Shashank, K., Balachandra, M., Review on network intrusion detection techniques using machine learning. IEEE Distributed Computing, VLSI, Electrical Circuits and Robotics (DISCOVER), Mangalore, India, 104-109, 2018.

[49] Divekar, A., Parekh, M., Savla, V., Mishra, R., Shirole, M., Benchmarking datasets for anomaly-based network intrusion detection: KDD CUP 99 alternatives. IEEE 3rd International Conference on Computing, Communication and Security (ICCCS), Kathmandu, Nepal, 1-8, 2018.

[50] Foroushani, Z., Li, Y., Intrusion detection system by using hybrid algorithm of data mining technique. Proceedings of the 2018 7th International Conference on Software and Computer Applications - ICSCA, Kuantan Malaysia, 119-123, 2018.

[51] Verma, A., Ranga, V., Statistical analysis of CIDDS-001 dataset for network intrusion detection systems using distance-based machine learning. Procedia Comput Sci, 125(2018): 709-716, 2018.

[52] Monshizadeh, M., Khatri, V., Atli, B., Kantola, R., Yan, Z., Performance evaluation of a combined anomaly detection platform. IEEE Access, 7:100964-100978, 2019.

[53] Shah, A. A., Ehsan, M. K., Ishaq, K., Ali, Z., Farooq, M. S., An efficient hybrid classifier model for anomaly intrusion detection system. International Journal of Computer Science and Network Security, 18(11): 127-136, 2018.

[54] Serkani, E., Gharaee, H., Mohammadzadeh, N., Anomaly detection using SVM as classifier and Decision Tree for optimizing feature vectors. INT J INF SECUR, 11(2): 159–171, 2019.

[55] Jain, M., Kaur, G., A novel distributed semi-supervised approach for detection of network based attacks. 9th International Conference on Cloud Computing, Data Science & Engineering (Confluence), Noida, India, 120-125, 2019.

[56] Singh, V., Puthran, S., Tiwari, A., Intrusion detection using data mining with correlation. 2nd International Conference for Convergence in Technology (I2CT), Mumbai, India, 620-625, 2017.

[57] Singh, P., Tiwari, A., An efficient approach for intrusion detection in reduced features of KDD99 using ID3 and classification with KNNGA. Second International Conference on Advances in Computing and Communication Engineering, Dehradun, India, 445-452, 2015.

[58] Timcenko, V., Gajin, S., Ensemble classifiers for supervised anomaly based network intrusion detection. 13th IEEE International Conference on Intelligent Computer Communication and Processing (ICCP), Cluj-Napoca, Romania, 13-19, 2017.

[59] Soheily-Khah, S., Marteau, P., Bechet, N., Intrusion detection in network systems through hybrid supervised and unsupervised machine learning process: a case study on the ISCX dataset. 1st International Conference on Data Intelligence and Security (ICDIS), South Padre Island, TX, USA, 219-226, 2018.

[60] Panda, M., Patra, M., A comparative study of data mining algorithms for network intrusion detection. First International Conference on Emerging Trends in Engineering and Technology, Nagpur, Maharashtra, India, 504-507, 2008.

[61] Li, M., Application of CART decision tree combined with PCA algorithm in intrusion detection. 8th IEEE International Conference on Software Engineering and Service Science (ICSESS), Beijing, China, 38-41, 2017.

[62] Chand, N., Mishra, P., Krishna, C., Pilli, E., Govil, M., A comparative analysis of SVM and its stacking with other classification algorithm for intrusion detection. International Conference on Advances in Computing, Communication, & Automation (ICACCA) (Spring), Dehradun, India, 2016.

[63] Cataltepe, Z., Ekmekci, U., Cataltepe, T., Kelebek, I., Online feature selected semi-supervised decision trees for network intrusion detection. NOMS 2016 - IEEE/IFIP Network Operations and Management Symposium, Istanbul, Turkey, 1085-1088, 2016.

[64] Salman, T., Bhamare, D., Erbad, A., Jain, R., Samaka, M., Machine learning for anomaly detection and categorization in multi-cloud environments. IEEE 4th International Conference on Cyber Security and Cloud Computing (CSCloud), New York, NY, USA, 2017.

[65] Govindarajan, M., Chandrasekaran, R., Intrusion detection using neural based hybrid classification methods. COMPUT NETW, 55(8): 1662-1671, 2011.

[66] Belavagi, M., Muniyal, B., Performance evaluation of supervised machine learning algorithms for intrusion detection. Procedia Comput Sci., 89(2016): 117-123, 2016.

[67] Li, Y., Guo, L., An active learning based TCM-KNN algorithm for supervised network intrusion detection. Comput Secur, 26(7-8): 459-467, 2007.

[68] Aissa, N., Guerroumi, M., Semi-supervised statistical approach for network anomaly detection. Procedia Comput Sci., 83(2016) :1090-1095, 2016.

[69] Ashfaq, R., Wang, X., Huang, J., Abbas, H., He, Y., Fuzziness based semi-supervised learning approach for intrusion detection system. Inf Sci (N Y), 378(2017): 484-497, 2017.

[70] Aliakbarisani, R., Ghasemi, A., Felix Wu, S., A data-driven metric learning-based scheme for unsupervised network anomaly detection. COMPUT ELECTR ENG, 73(2019): 71-83, 2019.

[71] Chellam, A., L, R., S, R., Intrusion detection in computer networks using lazy learning algorithm. Procedia Comput Sci., 132(2018): 928-936, 2018.

[72] Palmieri, F., Network anomaly detection based on logistic regression of nonlinear chaotic invariants. J NETW COMPUT APPL, 148(2019): 1-14, 2019.

[73] Muniyandi, A., Rajeswari, R., Rajaram, R., Network anomaly detection by cascading K-Means Clustering and C4.5 Decision Tree algorithm. Procedia Eng, 30(2012): 174-182, 2012.

[74] Sivatha Sindhu, S., Geetha, S., Kannan, A., Decision tree based light weight intrusion detection using a wrapper approach. Expert Syst Appl, 39(1):129-141, 2012.

[75] Al-Yaseen, W., Othman, Z., Nazri, M., Multi-level hybrid support vector machine and extreme learning machine based on modified K-means for intrusion detection system. Expert Syst Appl, 67(2017): 296-303, 2017.

[76] Ji, S., Jeong, B., Choi, S., Jeong, D., A multi-level intrusion detection method for abnormal network behaviors. J NETW COMPUT APPL, 62(2016): 9-17, 2016.

[77] Aburomman, A., Ibne Reaz, M., A novel SVM-KNN-PSO ensemble method for intrusion detection system. Appl Soft Comput, 38(2016): 360-372, 2016.

[78] Albon, C., Machine learning with Python cookbook: practical solutions from preprocessing to deep learning. İçinde: Naïve Bayes. 1. Baskı, O'Reilly Media, Inc., CA, USA, 279-284, 2018.

[79] Albon, C., Machine learning with Python cookbook: practical solutions from preprocessing to deep learning. İçinde: Support Vector Machines. 1. Baskı, O'Reilly Media, Inc., CA, USA, 267-277, 2018.

[80] https://towardsdatascience.com/support-vector-machine-introduction-to-machine-learning-algorithms-934a444fca47., Access on: 16.11.2019.

[81] Albon C., Machine learning with Python cookbook: practical solutions from preprocessing to deep learning. İçinde: Support Vector Machines. 1. Edition, O'Reilly Media, Inc., CA, USA, 251-257, 2018.

[82] https://towardsdatascience.com/machine-learning-basics-with-the-k-nearest-neighbors-algorithm-6a6e71d01761., Erişim Tarihi: 12.11.2019.

[83] Keller, J. D., Namee, B. M., D'arcy, A., Fundamentals of machine learning for predictive data analytics. İçinde: Information Based Learning. 1. Baskı, The MIT Press, London, England, 168-164, 2015.

[84] https://en.wikipedia.org/wiki/C4.5_algorithm., Erişim Tarihi: 16.11.2019.

[85] Shalev-Shwartz, S., Ben-David, S., Understanding machine learning from theory to algorithms. İçinde: Decision Trees. 1. Baskı, Cambridge University Press, NY, USA, 212-218, 2014.

[86] https://towardsdatascience.com/the-random-forest-algorithm-d457d499ffcd., Erişim Tarihi: 22.11.2019.

[87] Shalev-Shwartz, S., Ben-David, S., understanding machine learning from theory to algorithms. İçinde: Boosting. 1. Baskı, Cambridge University Press, NY, USA, 101-112, 2014.

[88] https://skymind.ai/wiki/multilayer-perceptron., Erişim Tarihi: 29.11.2019.

[89] Warzyński, A., Kołaczek, G., Intrusion detection systems vulnerability on adversarial examples. 2018 Innovations in Intelligent Systems and Applications (INISTA), Thessaloniki, Greece, 2018.

[90] Hossen, S., Janagam, A., Analysis of network intrusion detection system with machine learning algorithms (deep reinforcement learning algorithm). Blekinge Institute of Technology, Faculty of Computing, Yüksek Lisans Tezi, 2018.

[91] https://towardsdatascience.com/understanding-confusion-matrix-a9ad42dcfd62., Erişim Tarihi: 11.12.2019.

[92] Pacheco, F., Exposito, E., Gineste, M., Baudoin, C., Aguilar, J., Towards the deployment of machine learning solutions in network traffic classification: a systematic survey. IEEE COMMUN SURV TUT, 21(2): 1988 - 2014, 2019.

[93] Resende, P. A. A., Drummond, A. C., A survey of Random Forest based methods for intrusion detection systems. ACM COMPUT SURV, 51(3): 1-36, 2018.

[94] Limthong, K., Performance of interval-based features for anomaly detection in network traffic. IEEE Conference on Communications and Network Security (CNS), National Harbor, MD, USA, 361-362, 2013.

[95] https://towardsdatascience.com/why-and-how-to-cross-validate-a-model-d6424b45261f., Erişim Tarihi: 26.12.2019.

[96] https://en.wikipedia.org/wiki/Pareto_principle., Erişim Tarihi: 26.12.2019.

[97] https://machinelearningmastery.com/k-fold-cross-validation/., Erişim Tarihi: 26.12.2019.

[98] Mishra, P., Varadharajan, V., Tupakula, U., Pilli, E. S., A detailed investigation and analysis of using machine learning techniques for intrusion detection. IEEE COMMUN SURV TUT, 21(1): 686-728, 2019.

[99] Boukhamla, A., Gaviro, JC., CICIDS2017 dataset: performance improvements and validation as a robust intrusion detection system testbed. International Journal of Information and Computer Security, (in press), 2019.

[100] Alsamiri, J., Alsubhi, K., Internet of Things cyber attacks detection using machine learning. International Journal of Advanced Computer Science and Applications (IJACSA), 10(12): 627-634, 2019.

[101] https://www.unb.ca/cic/datasets/ids-2018.html., Erişim Tarihi: 17.08.2019.

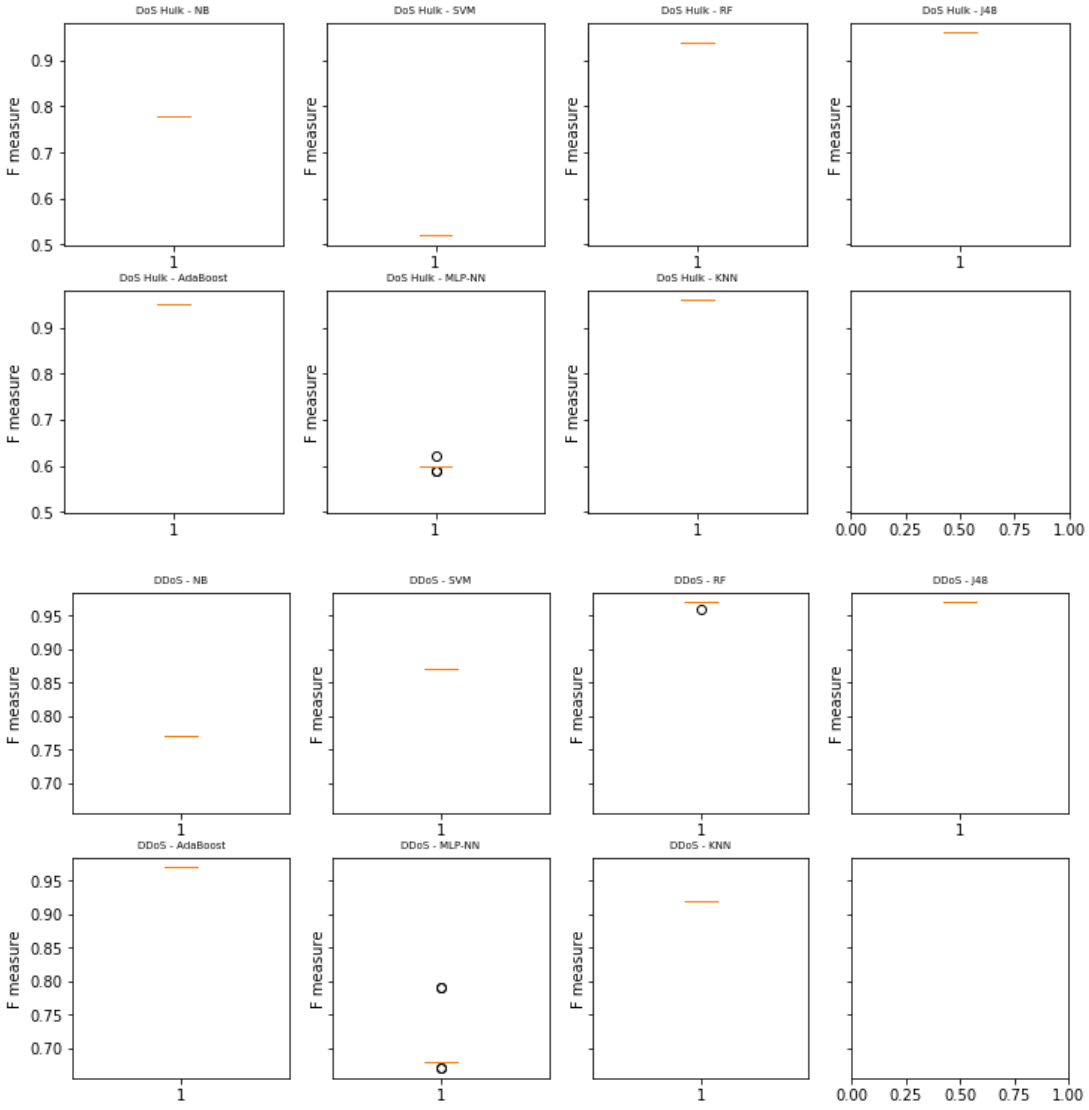
EKLER

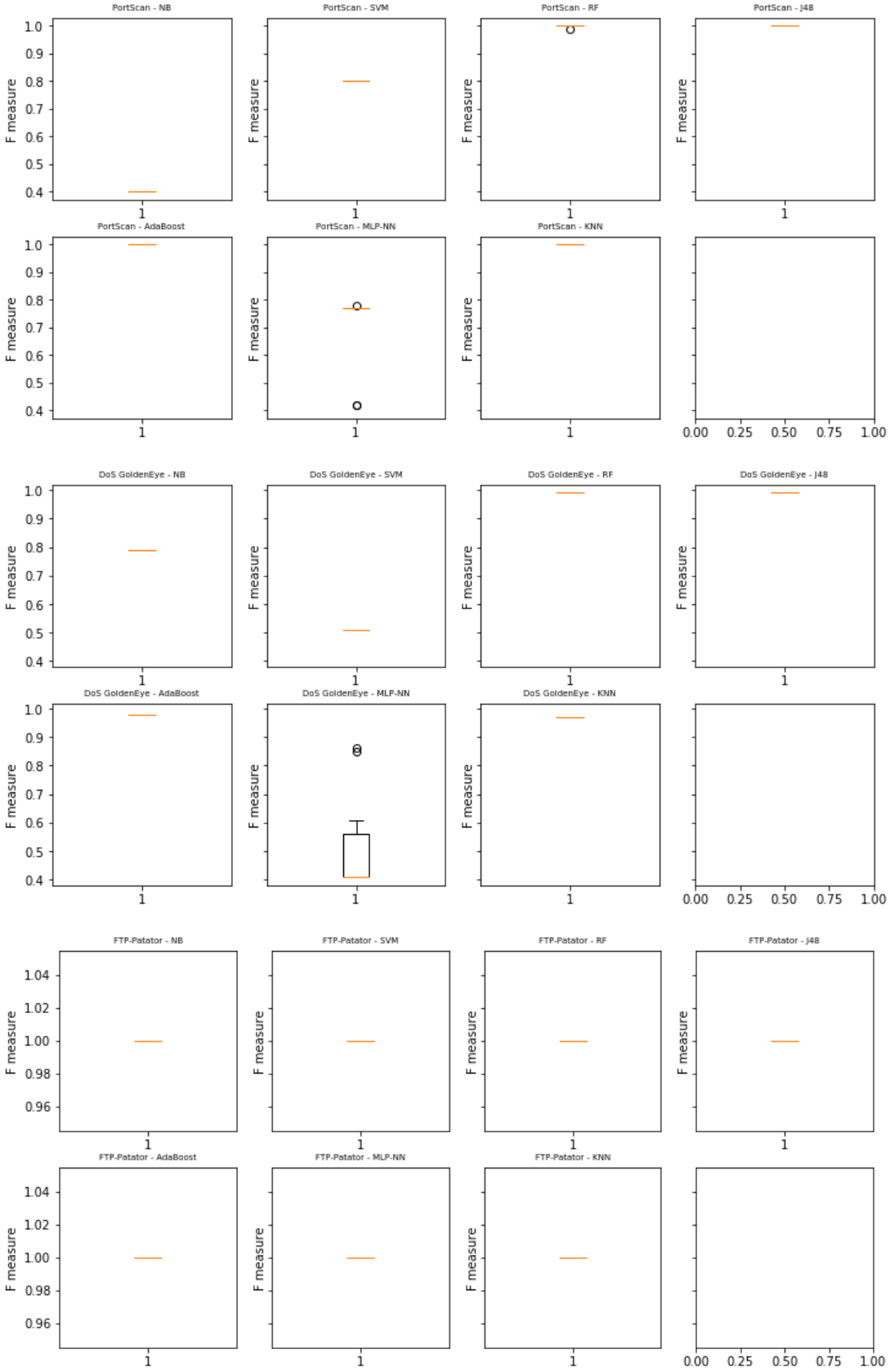
Ek 1: Tüm özelliklerin listesi ve açıklamaları

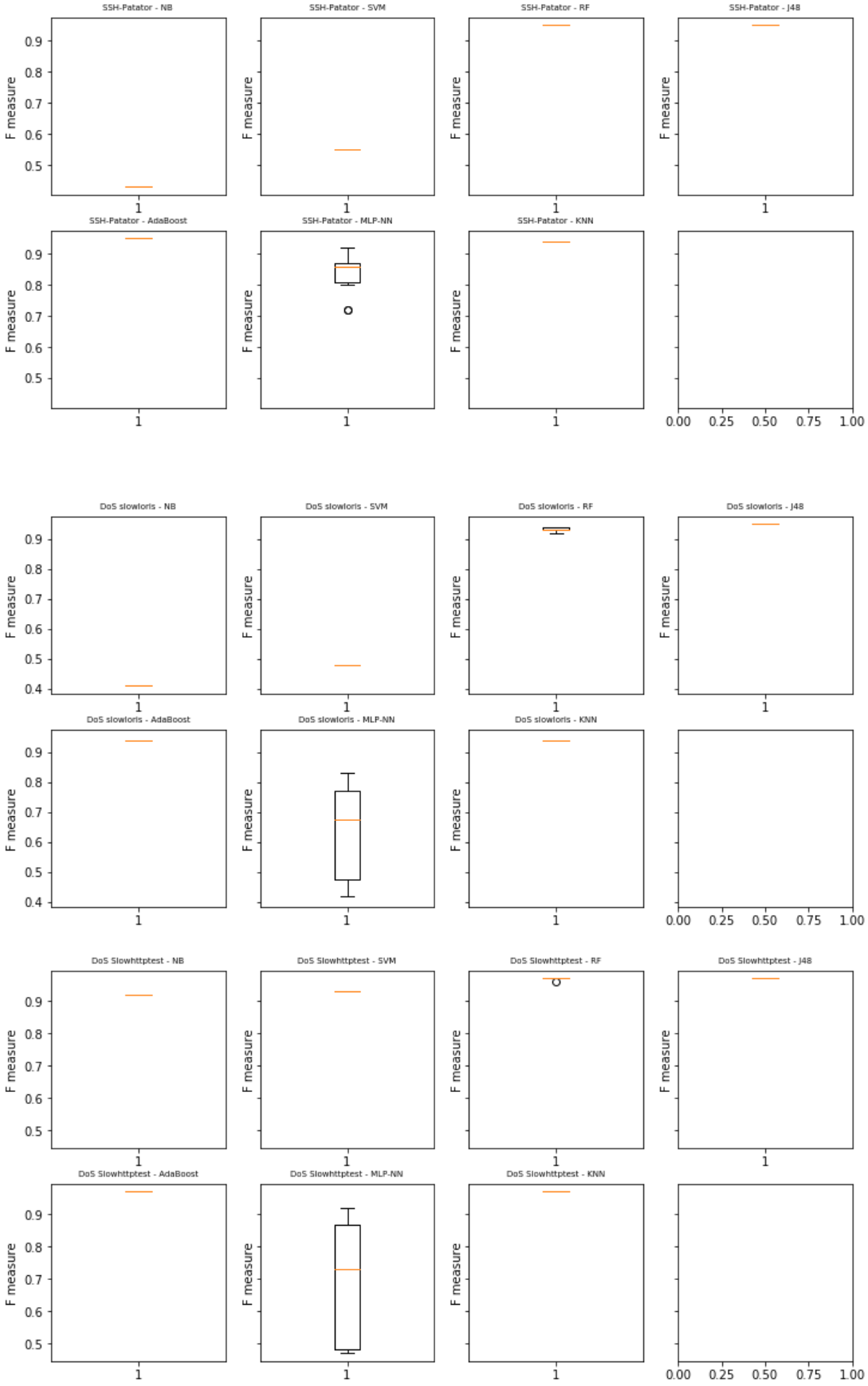
NO	Özellik	Açıklama
1	fl_dur	Akış süresi
2	tot_fw_pk	İleri yönde toplam paket
3	tot_bw_pk	Geri yönde toplam paket
4	tot_l_fw_pkt	Paketin ileri yönde toplam boyutu
5	fw_pkt_l_max	İleri yönde maksimum paket boyutu
6	fw_pkt_l_min	İleri yönde minimum paket boyutu
7	fw_pkt_l_avg	Paketin ileri yönde ortalama boyutu
8	fw_pkt_l_std	Paketin ileri yönde standart sapma boyutu
9	Bw_pkt_l_max	Geri yönde maksimum paket boyutu
10	Bw_pkt_l_min	Minimum paket boyutu geriye doğru
11	Bw_pkt_l_avg	Paketin geriye doğru ortalama boyutu
12	Bw_pkt_l_std	Paketin geriye doğru standart sapma boyutu
13	fl_byt_s	saniyede aktarılan paket sayısı olan akış bayt oranı
14	fl_pkt_s	saniyede aktarılan paket sayısı olan akış paket hızı
15	fl_iat_avg	İki akış arasındaki ortalama süre
16	fl_iat_std	Standart sapma süresi iki akış
17	fl_iat_max	İki akış arasındaki maksimum süre
18	fl_iat_min	İki akış arasındaki minimum süre
19	fw_iat_tot	İleri yönde gönderilen iki paket arasındaki toplam süre
20	fw_iat_avg	İleri yönde gönderilen iki paket arasındaki ortalama süre
21	fw_iat_std	İleri yönde gönderilen iki paket arasındaki standart sapma süresi
22	fw_iat_max	İleri yönde gönderilen iki paket arasındaki maksimum süre
23	fw_iat_min	İleri yönde gönderilen iki paket arasındaki minimum süre
24	bw_iat_tot	Geri yönde gönderilen iki paket arasındaki toplam süre
25	bw_iat_avg	Geri yönde gönderilen iki paket arasındaki ortalama süre
26	bw_iat_std	Geri yönde gönderilen iki paket arasındaki standart sapma süresi
27	bw_iat_max	Geri yönde gönderilen iki paket arasındaki maksimum süre
28	bw_iat_min	Geri yönde gönderilen iki paket arasındaki minimum süre
29	fw_psh_flag	İleri yönde seyahat eden paketlerde PSH bayrağının ayarlanma sayısı (UDP için 0)
30	bw_psh_flag	PSH bayrağının geriye doğru giden paketlerde ayarlanma sayısı (UDP için 0)
31	fw_urg_flag	URG bayrağının ileri yönde giden paketlerde ayarlanma sayısı (UDP için 0)
32	bw_urg_flag	URG bayrağının geriye doğru giden paketlerde ayarlanma sayısı (UDP için 0)
33	fw_hdr_len	İleri yönde başlıklar için kullanılan toplam bayt sayısı
34	bw_hdr_len	Geri yönde başlıklar için kullanılan toplam bayt sayısı
35	fw_pkt_s	Saniyede iletilen paket sayısı
36	bw_pkt_s	Saniyedeki geriye doğru paket sayısı

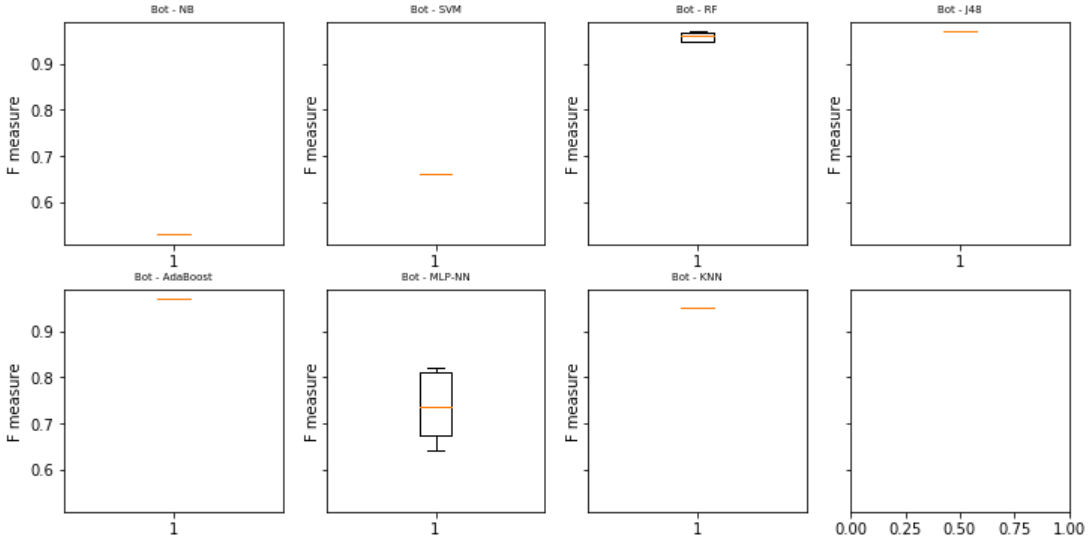
NO	Özellik	Açıklama
37	pkt_len_min	Minimum akış uzunluğu
38	pkt_len_max	Maksimum akış uzunluğu
39	pkt_len_avg	Ortalama akış uzunluğu
40	pkt_len_std	Bir akışın standart sapma uzunluğu
41	pkt_len_va	Minimum paket varış zamanı
42	fin_cnt	FIN'li paket sayısı
43	syn_cnt	SYN'li paket sayısı
44	rst_cnt	RST'li paket sayısı
45	pst_cnt	PUSH'li paket sayısı
46	ack_cnt	ACK'li paket sayısı
47	urg_cnt	URG'li paket sayısı
48	cwe_cnt	CWE'li paket sayısı
49	ece_cnt	ECE'li paket sayısı
50	down_up_ratio	İndirme ve yükleme oranı
51	pkt_size_avg	Ortalama paket boyutu
52	fw_seg_avg	İleri yönde gözlenen ortalama boyut
53	bw_seg_avg	Geri yönde gözlemlenen ortalama boyut
54	fw_byt_blk_avg	İleri yönde ortalama bayt yığın oranı
55	fw_pkt_blk_avg	İleri yönde ortalama paket toplu hızı
56	fw_blk_rate_avg	İleri yönde ortalama yığın oranı
57	bw_byt_blk_avg	Geri yönde ortalama bayt yığın oranı
58	bw_pkt_blk_avg	Geri yönde ortalama paket yığın sayısı
59	bw_blk_rate_avg	Geri yönde ortalama yığın oranı
60	subfl_fw_pk	İleri yönde bir alt akıştaki ortalama paket sayısı
61	subfl_fw_byt	İleri yönde bir alt akıştaki ortalama bayt sayısı
62	subfl_bw_pkt	Geri akıştaki bir alt akıştaki ortalama paket sayısı
63	subfl_bw_byt	Geri yönde bir alt akıştaki ortalama bayt sayısı
64	fw_win_byt	İlk pencerede ileri yönde gönderilen bayt sayısı
65	bw_win_byt	İlk pencerede geriye doğru gönderilen bayt sayısı
66	Fw_act_pkt	İleri yönde en az 1 bayt TCP veri yükü olan paket sayısı
67	fw_seg_min	İleri yönde gözlemlenen minimum segment boyutu
68	atv_avg	Bir akışın boşta kalmadan önce aktif olduğu ortalama süre
69	atv_std	Bir akış boşta olmadan önce aktif olan standart sapma süresi
70	atv_max	Bir akışın kullanılmadan önce etkin olduğu maksimum süre
71	atv_min	Boşta kalmadan önce bir akışın etkin olduğu minimum süre
72	idl_avg	Bir akışın aktif hale gelmeden önce boşta kaldığı ortalama süre
73	idl_std	Bir akış aktif olmadan önce boşta kalan standart sapma süresi
74	idl_max	Bir akışın etkin olmadan önce kullanılmadığı maksimum süre
75	idl_min	Bir akışın etkin hale gelmeden önce en az boşta kaldığı süre

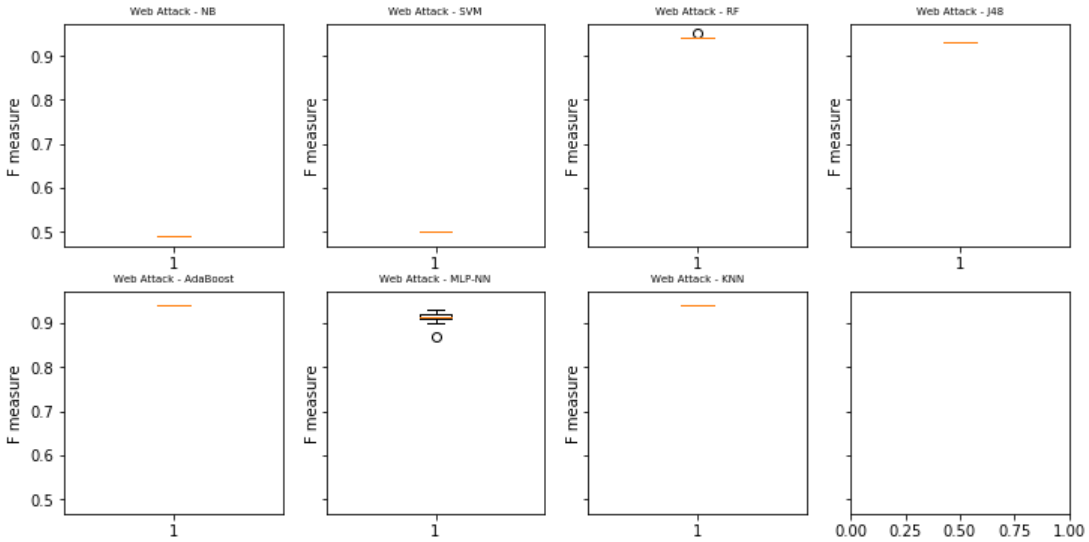
Ek 2: A. CIC-IDS-2017 çok terimli sınıflandırma sonuçları.

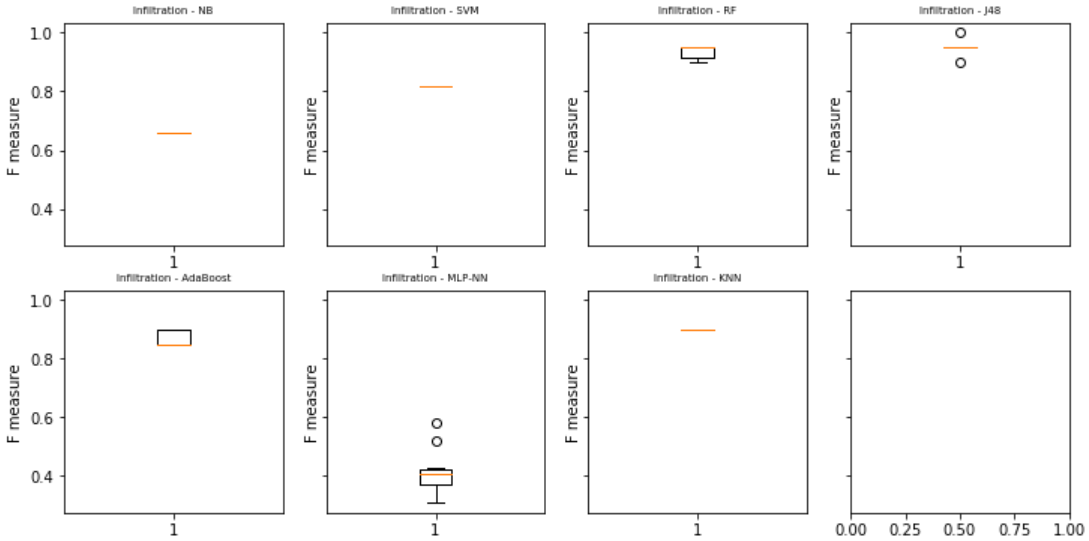


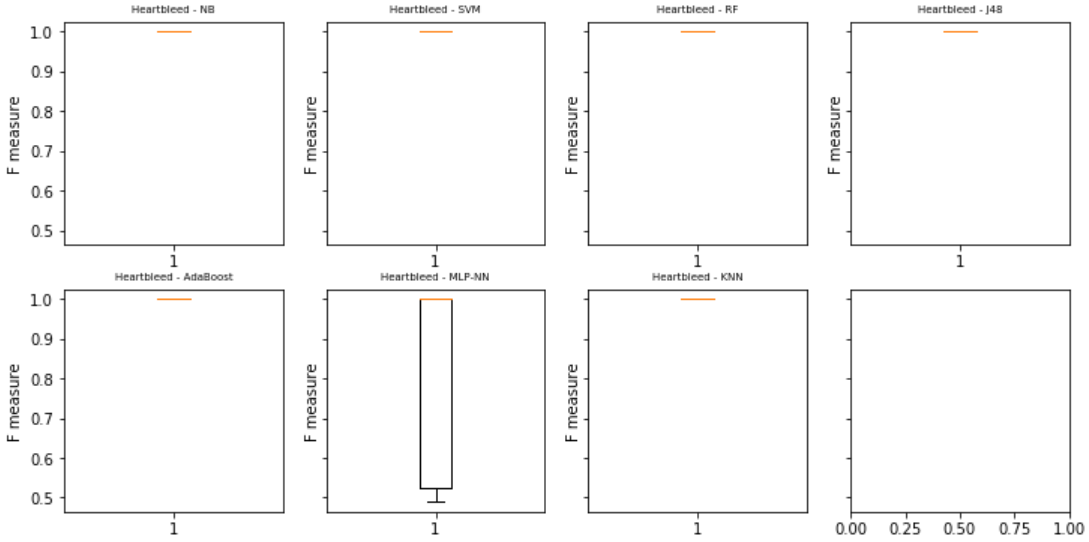

PortScan - NB
PortScan - SVM
PortScan - RF
PortScan - J48
PortScan - AdaBoost
PortScan - MLP-NN
PortScan - KNN
DoS GoldenEye - NB
DoS GoldenEye - SVM
DoS GoldenEye - RF
DoS GoldenEye - J48
DoS GoldenEye - AdaBoost
DoS GoldenEye - MLP-NN
DoS GoldenEye - KNN
FTP-Patator - NB
FTP-Patator - SVM
FTP-Patator - RF
FTP-Patator - J48
FTP-Patator - AdaBoost
FTP-Patator - MLP-NN
FTP-Patator - KNN
F measure


SSH-Patator - NB
SSH-Patator - SVM
SSH-Patator - RF
SSH-Patator - J48
SSH-Patator - AdaBoost
SSH-Patator - MLP-NN
SSH-Patator - KNN
DoS slowloris - NB
DoS slowloris - SVM
DoS slowloris - RF
DoS slowloris - J48
DoS slowloris - AdaBoost
DoS slowloris - MLP-NN
DoS slowloris - KNN
DoS Slowhttptest - NB
DoS Slowhttptest - SVM
DoS Slowhttptest - RF
DoS Slowhttptest - J48
DoS Slowhttptest - AdaBoost
DoS Slowhttptest - MLP-NN
DoS Slowhttptest - KNN
F measure

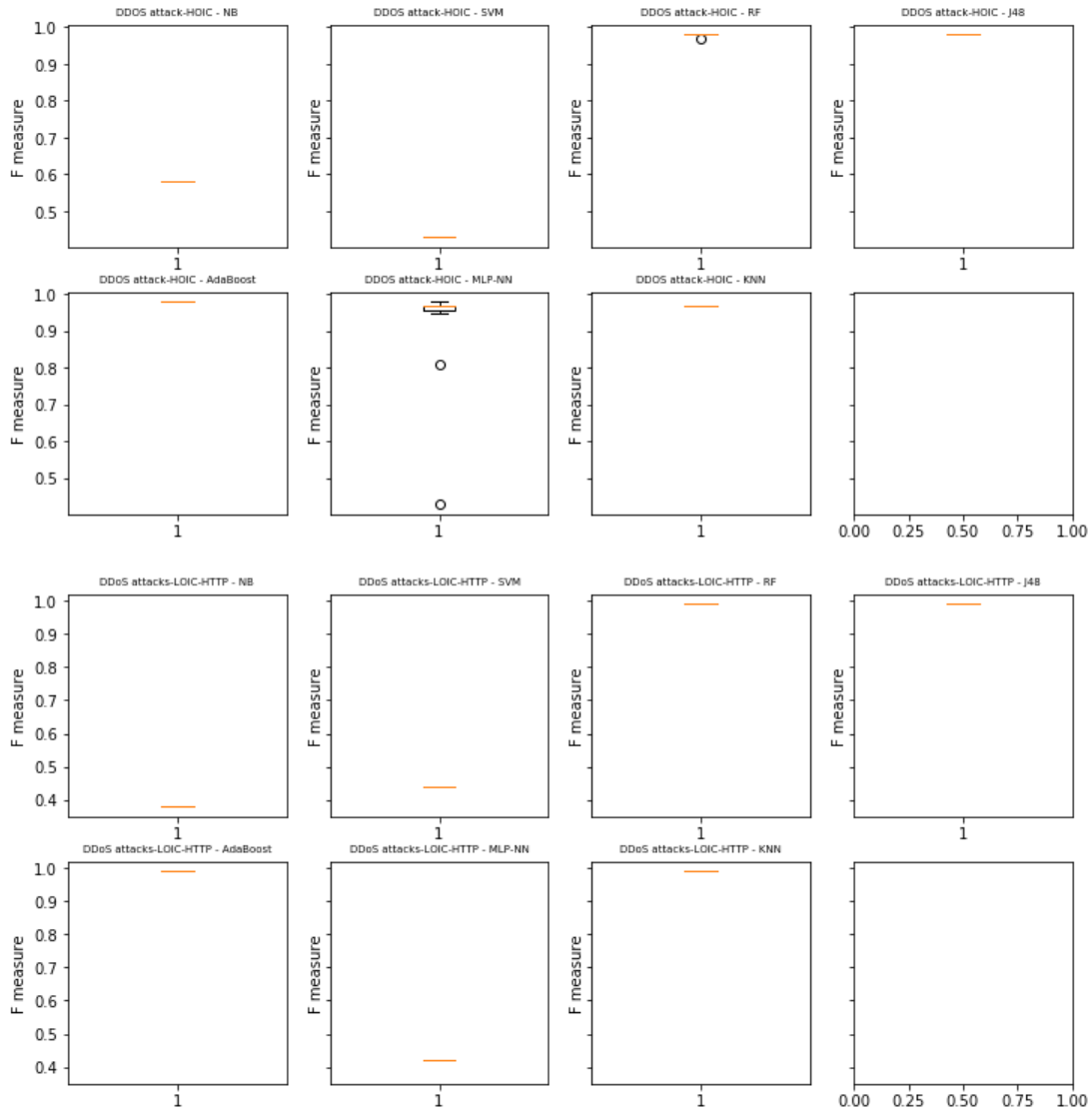

Bot - NB
Bot - SVM
Bot - RF
Bot - J48
Bot - AdaBoost
Bot - MLP-NN
Bot - KNN
F measure

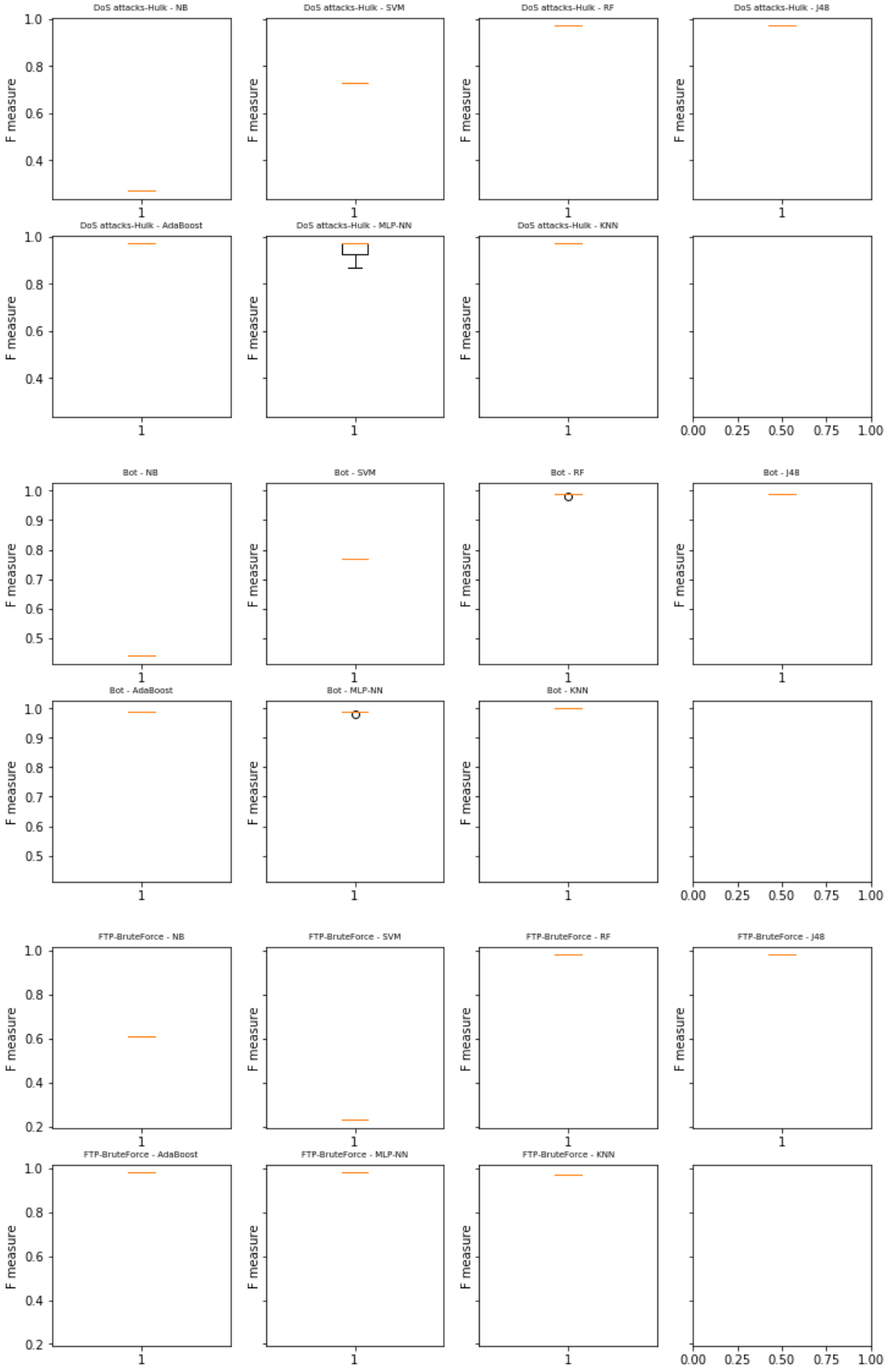

Web Attack - NB
Web Attack - SVM
Web Attack - RF
Web Attack - J48
Web Attack - AdaBoost
Web Attack - MLP-NN
Web Attack - KNN
F measure

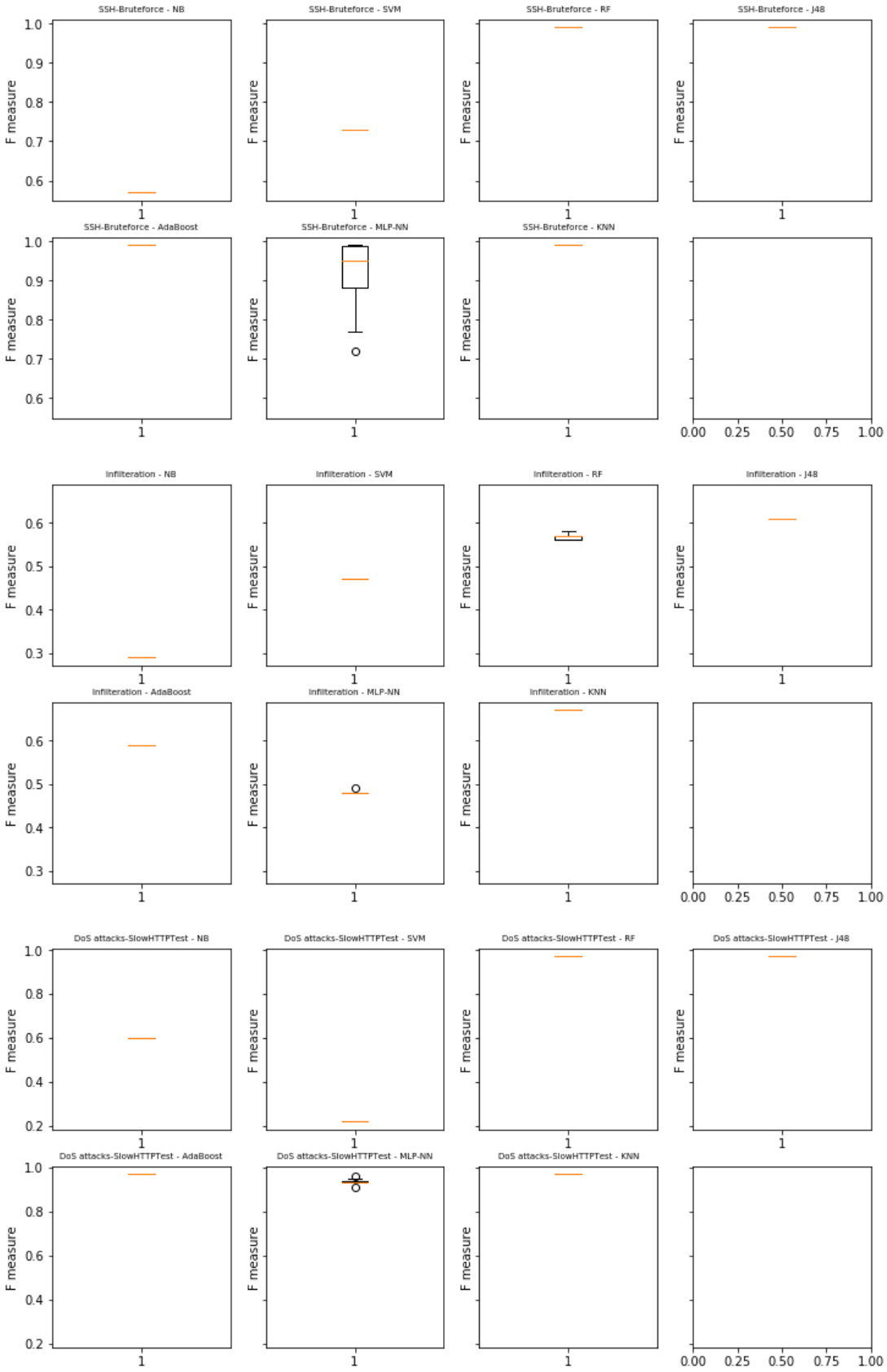

Infiltration - NB
Infiltration - SVM
Infiltration - RF
Infiltration - J48
Infiltration - AdaBoost
Infiltration - MLP-NN
Infiltration - KNN
F measure

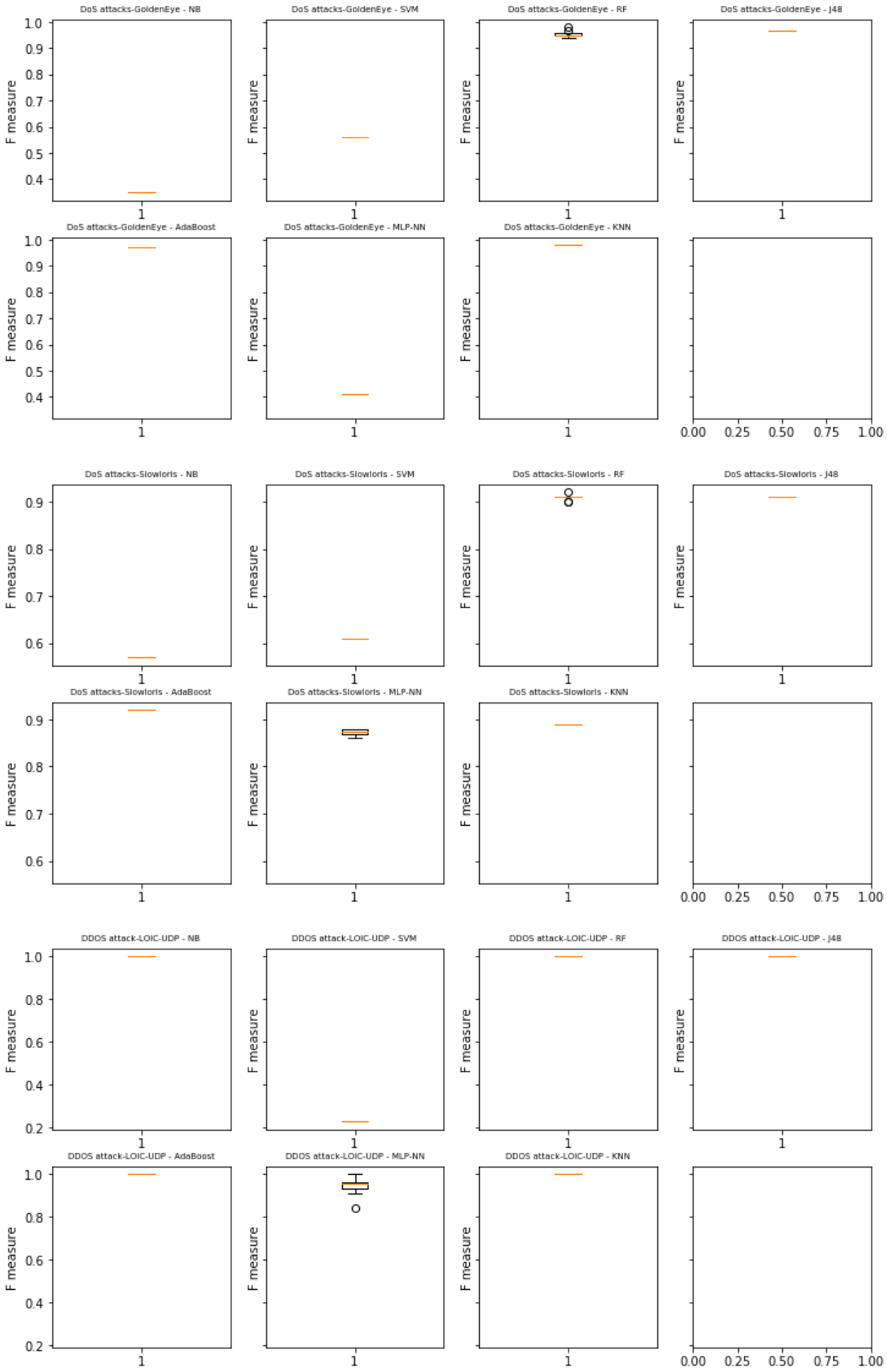

Heartbleed - NB
Heartbleed - SVM
Heartbleed - RF
Heartbleed - J48
Heartbleed - AdaBoost
Heartbleed - MLP-NN
Heartbleed - KNN
F measure
1.0
0.9
0.8
0.7
0.6
0.5
1
0.00
0.25
0.50
0.75
1.00

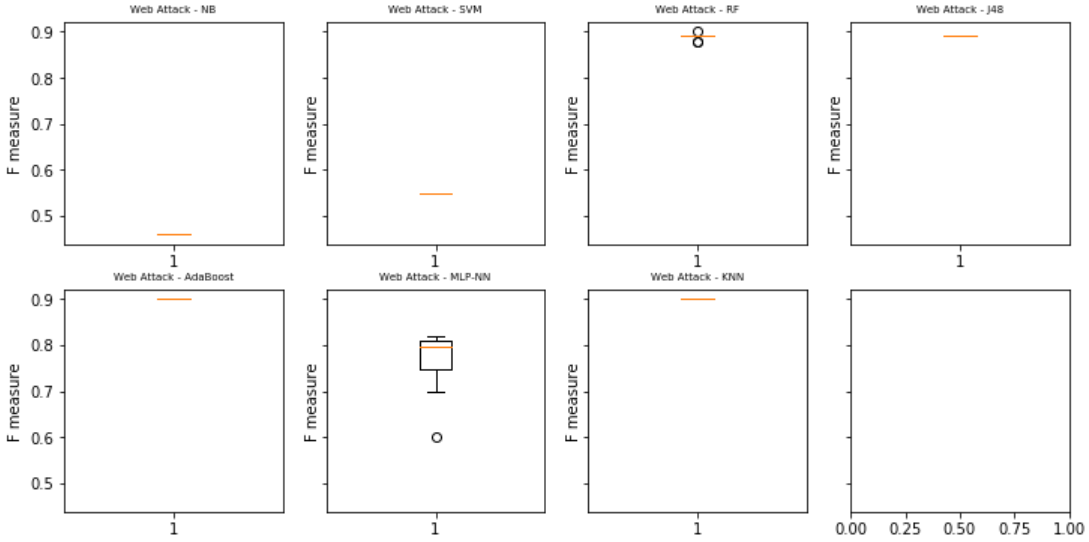
Ek 3: CSE-CIC-IDS-2018 çok terimli sınıflandırma sonuçları.



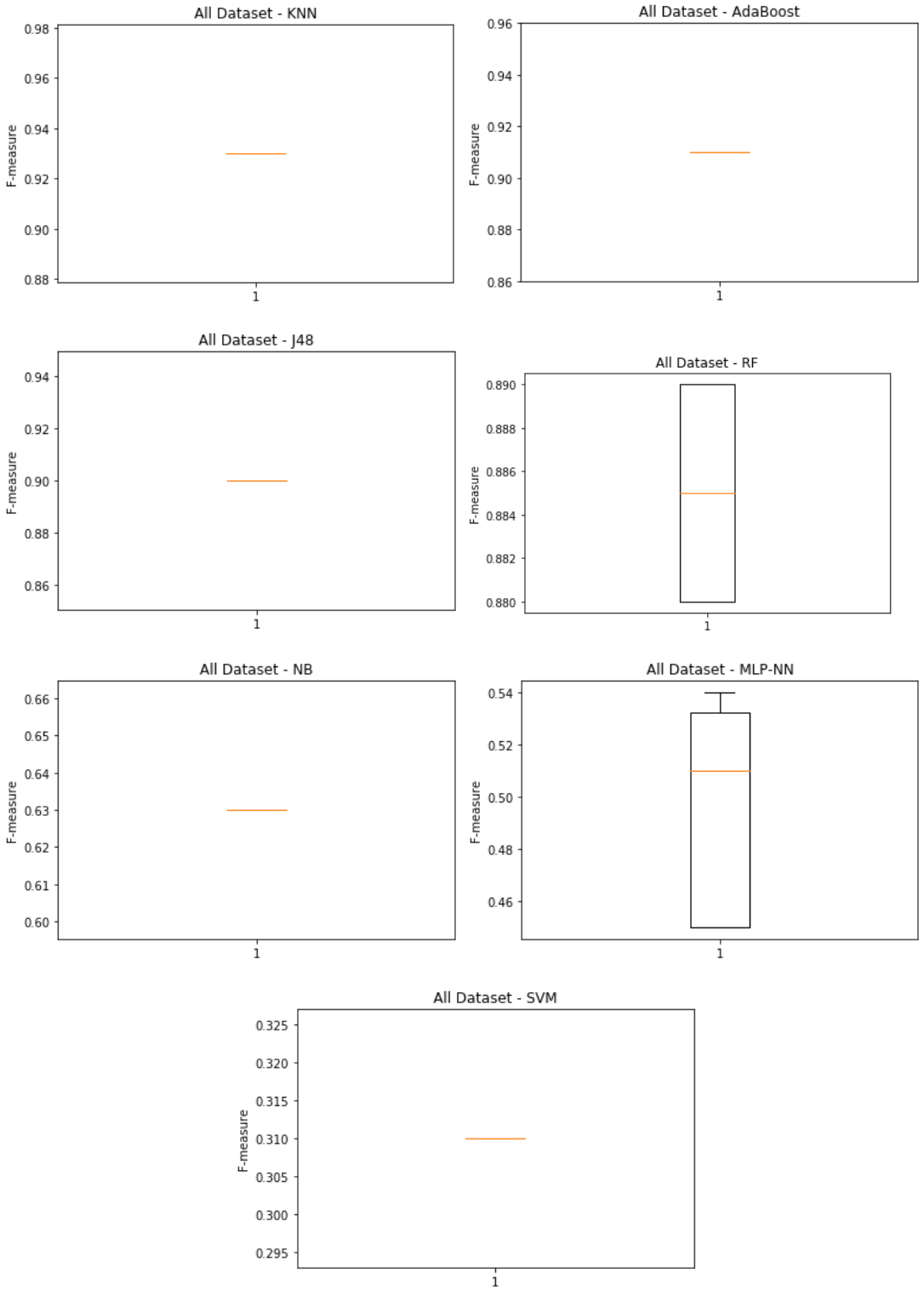

DoS attacks-Hulk - NB
DoS attacks-Hulk - SVM
DoS attacks-Hulk - RF
DoS attacks-Hulk - J48
DoS attacks-Hulk - AdaBoost
DoS attacks-Hulk - MLP-NN
DoS attacks-Hulk - KNN
Bot - NB
Bot - SVM
Bot - RF
Bot - J48
Bot - AdaBoost
Bot - MLP-NN
Bot - KNN
FTP-BruteForce - NB
FTP-BruteForce - SVM
FTP-BruteForce - RF
FTP-BruteForce - J48
FTP-BruteForce - AdaBoost
FTP-BruteForce - MLP-NN
FTP-BruteForce - KNN
F measure


SSH-Bruteforce - NB
SSH-Bruteforce - SVM
SSH-Bruteforce - RF
SSH-Bruteforce - J48
SSH-Bruteforce - AdaBoost
SSH-Bruteforce - MLP-NN
SSH-Bruteforce - KNN
Infiltration - NB
Infiltration - SVM
Infiltration - RF
Infiltration - J48
Infiltration - AdaBoost
Infiltration - MLP-NN
Infiltration - KNN
DoS attacks-SlowHTTPTest - NB
DoS attacks-SlowHTTPTest - SVM
DoS attacks-SlowHTTPTest - RF
DoS attacks-SlowHTTPTest - J48
DoS attacks-SlowHTTPTest - AdaBoost
DoS attacks-SlowHTTPTest - MLP-NN
DoS attacks-SlowHTTPTest - KNN
F measure

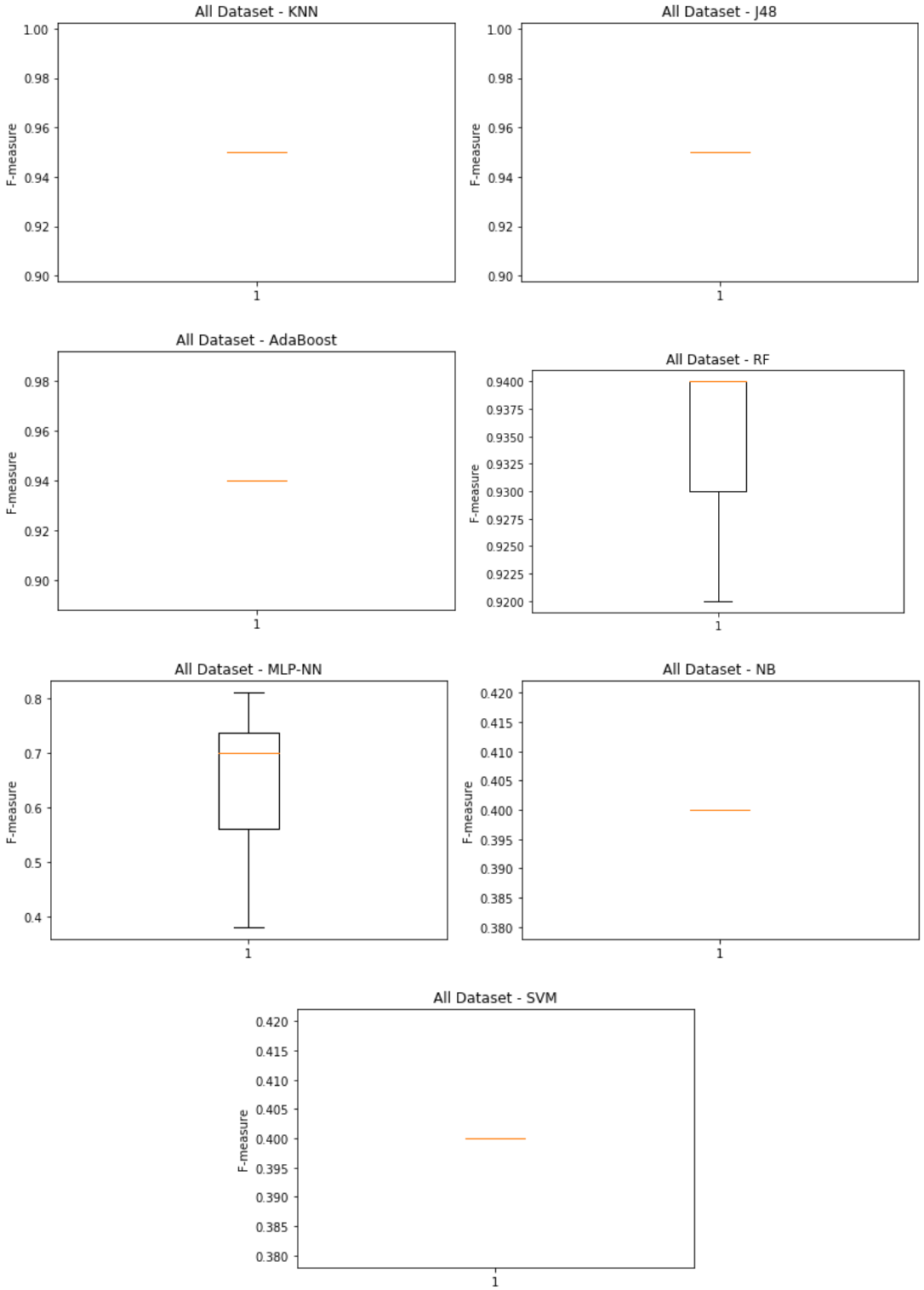

DoS attacks-GoldenEye - NB
DoS attacks-GoldenEye - SVM
DoS attacks-GoldenEye - RF
DoS attacks-GoldenEye - J48
DoS attacks-GoldenEye - AdaBoost
DoS attacks-GoldenEye - MLP-NN
DoS attacks-GoldenEye - KNN
DoS attacks-Slowloris - NB
DoS attacks-Slowloris - SVM
DoS attacks-Slowloris - RF
DoS attacks-Slowloris - J48
DoS attacks-Slowloris - AdaBoost
DoS attacks-Slowloris - MLP-NN
DoS attacks-Slowloris - KNN
DDOS attack-LOIC-UDP - NB
DDOS attack-LOIC-UDP - SVM
DDOS attack-LOIC-UDP - RF
DDOS attack-LOIC-UDP - J48
DDOS attack-LOIC-UDP - AdaBoost
DDOS attack-LOIC-UDP - MLP-NN
DDOS attack-LOIC-UDP - KNN
F measure


Web Attack - NB
Web Attack - SVM
Web Attack - RF
Web Attack - J48
Web Attack - AdaBoost
Web Attack - MLP-NN
Web Attack - KNN
F measure
0.9
0.8
0.7
0.6
0.5
1
0.00
0.25
0.50
0.75
1.00

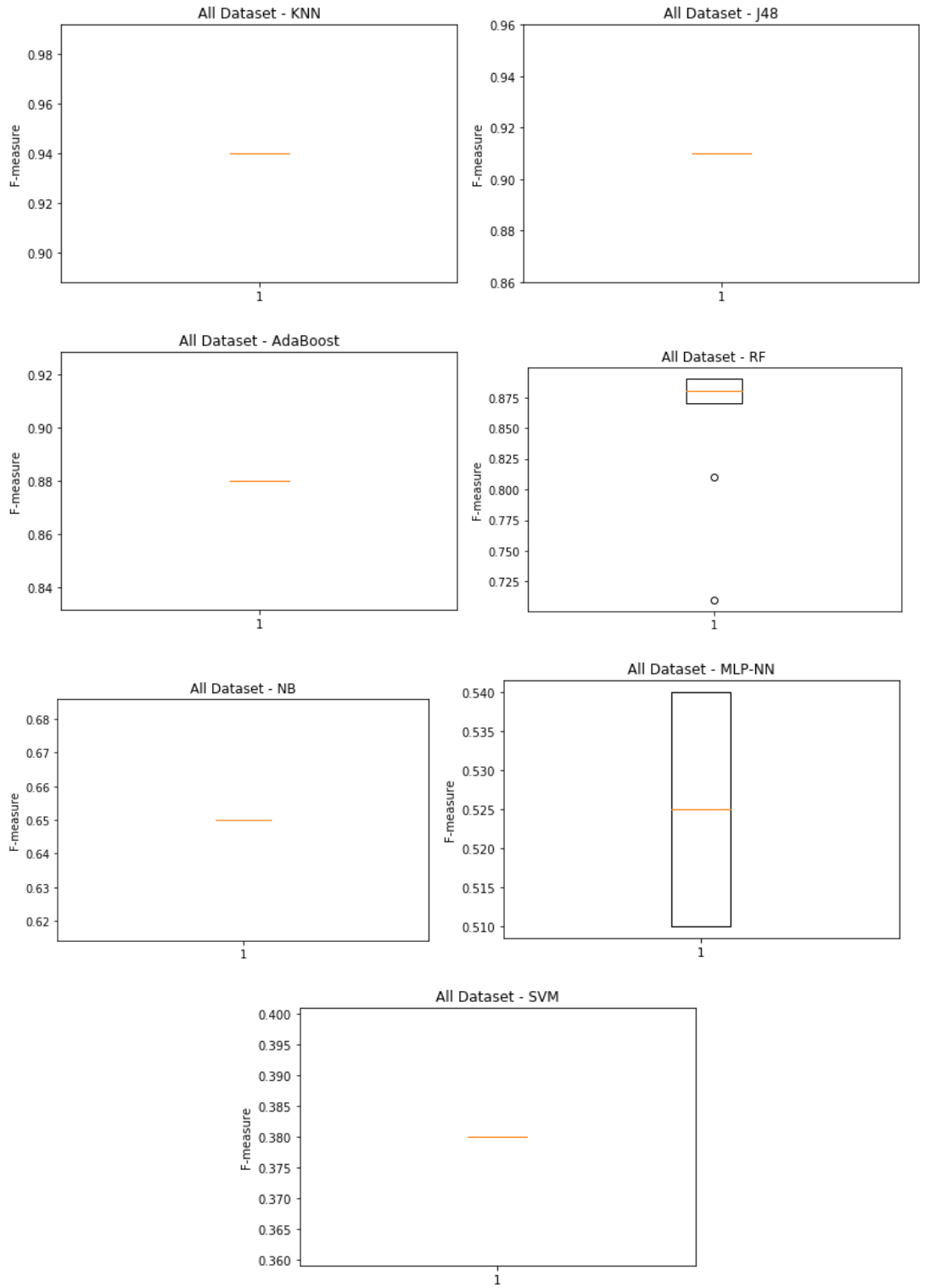
Ek 4: CIC-IDS-2017 çok terimli özelliklerle ikili sınıflandırma sonuçları



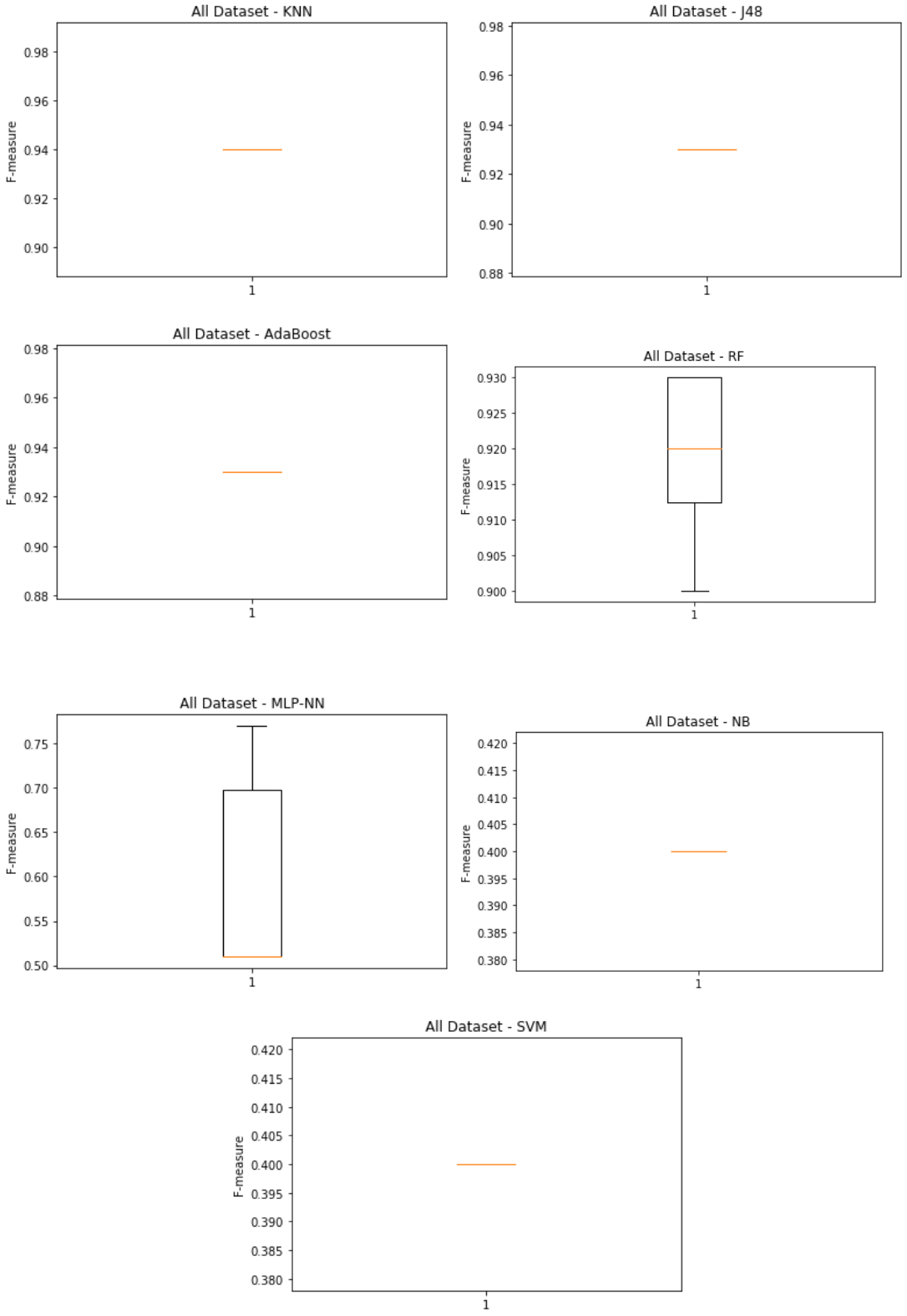
Ek 5: CSE-CIC-IDS-2018 çok terimli özelliklerle ikili sınıflandırma sonuçları.



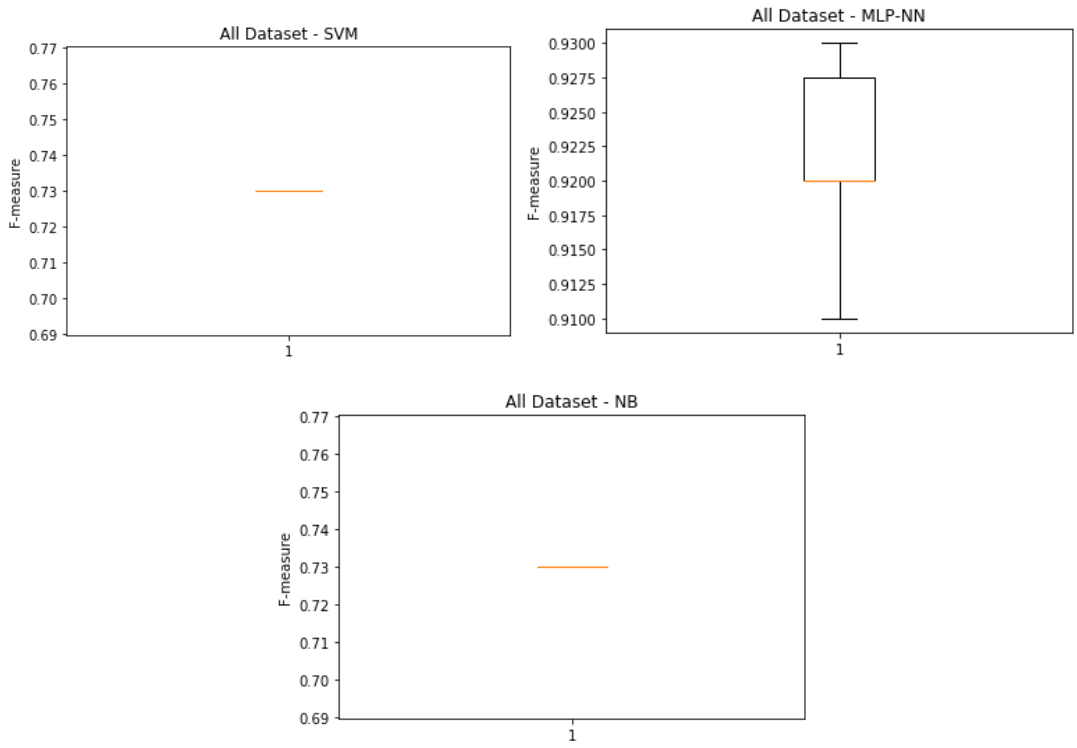
Ek 6: CIC-IDS-2017 ikili özelliklerle ikili sınıflandırma sonuçları.



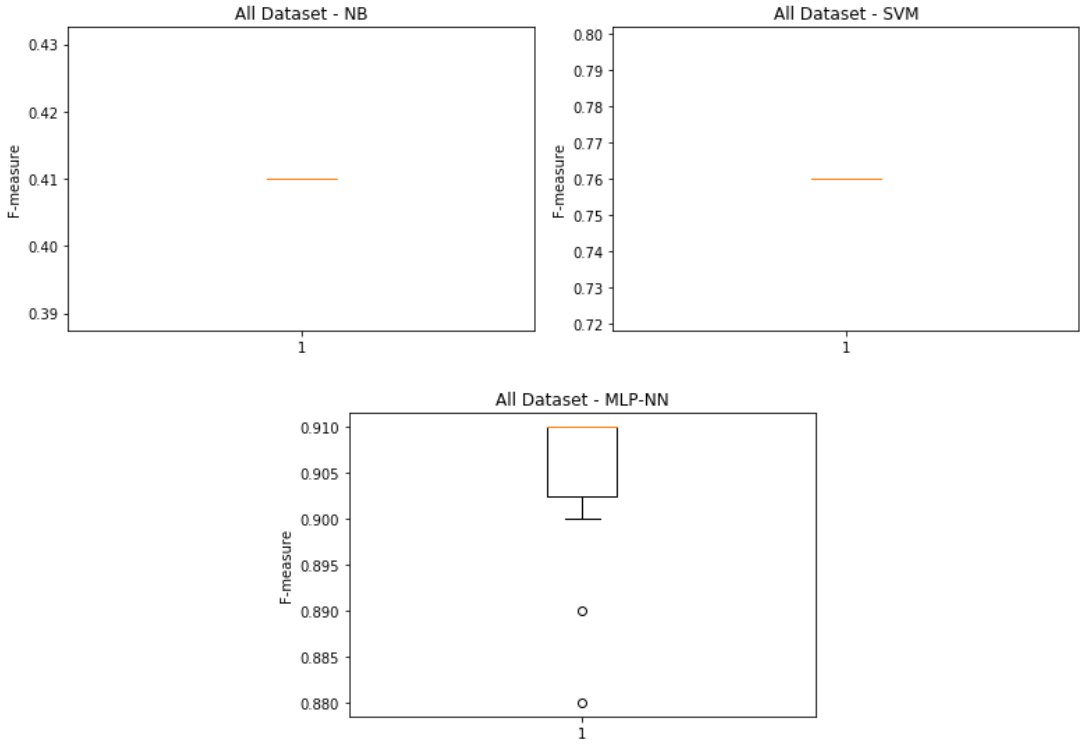
Ek 7: CSE-CIC-IDS-2018 ikili özelliklerle ikili sınıflandırma sonuçları.



Ek 8: CIC-IDS-2017 geliştirilmiş özelliklerle ikili sınıflandırma sonuçları.



Ek 9: CSE-CIC-IDS-2018 geliştirilmiş özelliklerle ikili sınıflandırma sonuçları.



ÖZGEÇMIŞ

Mujibullah Shams, 08.09.1994'da Afganistan'da doğdu. İlk, orta ve lise eğitimini Kabil'de tamamladı. 2011 yılında Khwaja Abdullah Ansari Lisesi'nden mezun oldu. 2012 yılında başladığı Kabul Üniversitesi Bilgisayar Bilimleri Bölümü'nü 2015 yılında bitirdi. 2016 yılında Sakarya Üniversitesi, TÖMER'de bir yıl Türkçe dili hazırlığı gördükten sonra, 2017 yılında Bilgisayar ve Bilişim Mühendisliği Bölümü'nde yüksek lisans eğitimine başladı. 2020 yılında yüksek lisans eğtimini başarılı bir şekilde bitirdi.