Bilgisayar Bilimleri ve Mühendisliği Dergisi

Cilt-19
Sayı-1
2026

Sistem Çağrı Dizileri Kullanarak Ana Bilgisayar Tabanlı Saldırı Tespit Sistemlerinin Geliştirilmesi

Developing Host-Based Intrusion Detection Systems Using System Call Sequences

Hami SATILMIŞ
Ondokuz Mayıs Üniversitesi
Bilgisayar Mühendisliği Bölümü
Samsun, Türkiye
hami.satilmis@bil.omu.edu.tr
ORCID: 0000-0002-6611-7549

Sedat AKLEYLEK
İstinye Üniversitesi / Tartu Üniversitesi
Bilgisayar Mühendisliği Bölümü / Güvenlik ve Teorik Bilgisayar Bilimleri Kürsüsü
İstanbul, Türkiye / Tartu, Estonya
akleylek@gmail.com
ORCID: 0000-0001-7005-6489

Öz

Sistem çağrı dizilerine dayalı ana bilgisayar tabanlı saldırı tespit sistemleri, siber güvenlik alanında kritik bir araştırma konusu olmakla birlikte, bu dizilerin yüksek boyutluluğu ve sıralı yapısı, doğrudan işlenmelerini güçleştirmektedir. Bu çalışmada, sistem çağrı dizileri içeren UNM live LPR, MIT live LPR ve bu iki veri kümesinin birleşiminden oluşturulan UNM-MIT live LPR veri kümeleri üzerinde farklı ön işlem yöntemleri uygulanmaktadır. Bu ön işlem yöntemleri içerisinde, n-gram, çeşitli BoW ve özellik seçim yöntemleri yer almaktadır. Ön işlem sonucunda, her bir veri kümesinden türetilen BoW veri kümeleri elde edilmektedir. Daha sonra, seçilen özellikleri içeren BoW veri kümeleri kullanılarak hem bireysel hem de yığınlama topluluğu tabanlı makine öğrenimi modelleri/HIDS'leri geliştirilmektedir. Deneyler, geliştirilen bazı modellerin/HIDS'lerin, %0 yanlış pozitif oranı ve %100 doğruluk oranı ile yüksek başarım sergilediğini göstermektedir.

Anahtar sözcükler: Saldırı Tespit Sistemi, Ana Bilgisayar Tabanlı Saldırı Tespit Sistemi, Bilgi Güvenliği, Sistem Çağrı Dizileri, Makine Öğrenimi

Makale Bilgileri	**Article Info**
Türü: Araştırma	Type: Research
Geliş tarihi: 06.14.2025	Received date: 06.14.2025
Düzeltme tarihi: 10.12.2025	Revision date: 10.12.2025
Kabul tarihi: 17.12.2025	Accepted date: 17.12.2025

Atıf/ to Cite (IEEE): H. SATILMIŞ, S. AKLEYLEK, Sistem Çağrı Dizileri Kullanarak Ana Bilgisayar Tabanlı Saldırı Tespit Sistemlerinin Geliştirilmesi: Bilgisayar Bilimleri ve Mühendisliği Dergisi, Cilt 19 2026 Sayı-19-1. Sf: 67-81
DOI: 10.54525/bbmd.1797760

Abstract

Host-based intrusion detection systems based on system call sequences represent a critical research area in cybersecurity. However, these sequences' high dimensionality and sequential nature make their direct processing challenging. In this study, various preprocessing methods are applied to the UNM live LPR, MIT live LPR, and the combined UNM-MIT live LPR datasets containing system call sequences. These preprocessing methods included n-gram, various BoW, and feature selection methods. As a result of preprocessing, BoW datasets are derived from each dataset. Subsequently, both individual and stacking ensemble-based machine learning models/HIDS are developed using the BoW datasets containing the selected features. Experimental results indicated that some of the developed models/HIDS achieved high performance, with a 0% false positive rate and 100% accuracy.

Keywords: Intrusion Detection System, Host-based Intrusion Detection System, Information Security, System Call Sequences, Machine Learning

1. Giriş

Teknolojik gelişmeler doğrultusunda bilgisayar sistemleri giderek daha karmaşık bir yapıya kavuşmakta ve bu sistemler arasındaki veri akışı hızlı bir şekilde artmaktadır. Söz konusu dijital yoğunluk, kötü niyetli aktörlerin dikkatini çekmekte ve onları çeşitli siber tehditleri gerçekleştirmeye teşvik etmektedir. Saldırganlar, sistemler ve ağlardaki güvenlik açıklarını tespit etmek amacıyla farklı yöntemlerden yararlanmakta ve bu zayıf noktaları kullanarak çeşitli saldırı

türlerini hayata geçirmektedir. Özellikle kötü amaçlı yazılımlar, veri güvenliğinin temel bileşenleri olan gizlilik, bütünlük ve erişilebilirlik ilkelerini ciddi şekilde tehdit etmektedir. Bu bağlamda, siber saldırıların erken ve doğru bir şekilde tespit edilmesi, veri güvenliğinin sağlanması açısından kritik bir öneme sahiptir.

Veri güvenliğinin sağlanması amacıyla, siber saldırıların tespitine odaklanan çeşitli çözümler önerilmiştir. Bu çözümler arasında özellikle saldırı tespit sistemleri, bilgisayar sistemlerine ve ağlarına yönelik olası tehditleri tespit etmek için tasarlanmaktadır. Söz konusu sistemler genellikle ağ cihazlarına veya sunuculara entegre edilmekte ve bulundukları ortamda gerçekleşen dijital faaliyetleri sürekli olarak izlemektedir. İzlenen faaliyetlerle ilgili veriler, IDS tarafından kaydedilmekte ve çeşitli analiz yöntemleri aracılığıyla işlenmektedir [1], [2]. IDS'ler, gerçekleştirdikleri analizler sonucunda, söz konusu faaliyetlerin siber saldırı niteliği taşıyıp taşımadığına karar vermektedir. IDS'ler, kullandıkları verilerin kaynağına bağlı olarak, ağ tabanlı IDS'ler (network-based intrusion detection system - NIDS) ve HIDS'ler olmak üzere iki temel kategoriye ayrılmaktadır [2]–[4].

HIDS, bir ana bilgisayardaki çeşitli faaliyetleri izleyerek güvenliği sağlamayı hedefleyen bir savunma yaklaşımıdır. Bu tür saldırı tespit sistemleri, bulundukları ortamdan elde ettikleri sistem çağrıları, API çağrıları ve sistem günlükleri gibi çeşitli verileri toplamaktadır ve işlemektedir [4]. Toplanan veriler, farklı analiz yöntemleri aracılığıyla değerlendirildikten sonra olağandışı davranışlar saptanmakta ve potansiyel saldırılar açığa çıkarılmaktadır. HIDS çözümleri, saldırıların algılanma biçimlerine bağlı olarak iki temel kategoriye ayrılmaktadır: imza tabanlı yöntemleri kullanan HIDS'ler ve anomali tabanlı yaklaşımları esas alan HIDS'ler [5], [6].

İmza tabanlı HIDS çözümleri, saldırı örüntülerini barındıran bir imza veri tabanı üzerinden çalışmakta; gözlemlenen yeni veriler bu imzalarla karşılaştırılarak olası tehditler tespit edilmektedir. Ancak bu çözümün temel eksikliği, yalnızca bilinen saldırı türlerini tespit edebilmesidir. Buna karşılık, anomali tabanlı HIDS çözümleri, istatistiksel analiz tekniklerinden ya da makine öğrenmesi ve derin öğrenme tabanlı modellerden yararlanarak, daha önce tanımlanmamış saldırıları da ortaya çıkarabilmektedir. Çizelge-1'de, çalışma boyunca sıkça kullanılan kısaltmalar ve bunların açılımları yer almaktadır.

Çizelge-1: Kısaltmalar ve Açılımlar

Kısaltma	Açılım
IDS	Saldırı Tespit Sistemi (Intrusion Detection System)
HIDS	Ana Bilgisayar Tabanlı Saldırı Tespit Sistemi (Host-based Intrusion Detection System)
ML	Makine Öğrenimi (Machine Learning)
DL	Derin Öğrenme (Deep Learning)
FSVM	Bulanık Destek Vektör Makinesi (Fuzzy Support Vector Machine)
IoT	Nesnelerin İnterneti (Internet o Things)
SVM	Destek Vektör Makinesi (Support Vector Machine)
UNM	New Mexico Üniversitesi (University of New Mexico)
MIT	Massachusetts Teknoloji Enstitüsü (Massachusetts Institute of Technology)
LPR	Satır Yazıcı Uzak Protokolü (Line Printer Remote Protocol)
KVM	Çekirdek Sanal Makinesi (Kernel Virtual Machine)
VMI	Sanal Makine İç Gözlem (Virtual Machine Introspection)
DT	Karar Ağacı (Decision Tree)
RF	Rastgele Orman (Random Forest)
NB	Saf Bayes (Naive Bayes)
KNN	K-En Yakın Komşular (K-Nearest Neighbors)
XGBoost	Aşırı Gradyan Artırımı (Extreme Gradient Boosting)
AdaBoost	Uyarlamalı Artırma (Adaptive Boosting)
CNN	Evrişimsel Sinir Ağı (Convolutional Neural Network)
LSTM	Uzun Kısa Süreli Bellek (Long Short-Term Memory)
TF-IDF	Terim Sıklıkları ve Ters Belge Sıklıkları (Term Frequencies and Inverse Document Frequencies)
MI	Karşılıklı Bilgi (Mutual Information)
BoW	Sözcük Kesesi (Bag-of-Words)
ID	Kimlik Numarası (Identification Number)
PID	Proses Kimlik Numarası (Process Identification Number)
TP	Doğru Pozitif (True Positive)
TN	Doğru Negatif (True Negative)
FP	Yanlış Pozitif (False Positive)
FN	Yanlış Negatif (False Negative)
PRC	Tutturma (Precision)
REC	Bulma (Recall)
F1	F1-Değeri (F1-Score)
FPR	Yanlış Pozitif Oranı (False Positive Rate)

ACC	Doğruluk (Accuracy)

1.1 Kaynak Taraması

[7]'de, destek vektör veri tanımına (SVDD) dayalı bulanık destek vektör makinesi yöntemi önerilmiştir. Bu FSVM yöntemi, destek vektör makinesi yönteminin sistem çağrı dizilerindeki gürültülere olan duyarlılık sorununu azaltmaya yönelik geliştirilmiştir. Önerilen yöntem, gerçek zamanlı olarak çalışan anomali tabanlı IDS'lerde kullanılması amacıyla, kaynakları verimli harcayacak şekilde tasarlanmıştır. New Mexico Üniversitesi veri kümesinin alt kümelerinden olan UNM canlı satır yazıcı uzak protokolü kümesi (UNM live line printer remote protocol) üzerindeki deneylerde önerilen yöntem, %92,63 tespit oranı (DR) ve %6,16 yanlış alarm oranı değerlerine ulaşmıştır. UNM'nin bir diğer alt kümesi olan UNM posta gönderme (UNM sendmail) kümesi kullanılarak gerçekleştirilen deneylerde önerilen yöntem, %84,61 DR ve %4,51 FAR değerleri ile başarım göstermiştir.

[8]'de, çekirdek sanal makinesi tabanlı bulut sistemlerinde kötücül yazılımları tespit edebilen KVMInspector tanıtılmıştır. KVMInspector'da, sanallaştırma katmanında sanal makine iç gözlemi ve ML modelleri birleştirilmiştir. KVMInspector'da ML modelleri olarak, karar ağacı, rastgele orman, saf Bayes, uyarlamalı artırma, k-en yakın komşular, SVM ve bu yöntemlerin çıktılarını birleştiren yumuşak oylama modelleri kullanılmıştır. Öte yandan, modeller geliştirilmeden önce veriler, n-gram ve yinelemeli özellik eleme (RFE) yöntemleriyle işlenmiştir. UNM veri kümesinin çeşitli alt kümelerinde yapılan deneylerde, UNM sentetik LPR (UNM synthetic LPR) üzerinde NB dışındaki modeller %99,29-%99,93 arasında doğruluk değerlerine ulaşmışlardır. Oluşturulan elog veri kümesinin çeşitli alt kümelerinde gerçekleştirilen deneylerde ise, zaman tabanlı kaçışlara ait örnekleri içeren alt küme üzerinde SVM, %98,17 ACC değeri ile başarım göstermiştir.

[9]'da, süreçlerde meydana gelen olağan dışı davranışları belirlemek amacıyla, gerçek zamanlı çalışabilecek bir HIDS tasarlanmıştır. Bu sistemde, sistem çağrıları n-gram yaklaşımı ile temsil edilmiş; ardından bu n-gram'ların önem ağırlıkları, terim sıklıkları ve ters belge sıklıkları tabanlı TFIDFVectorizer yöntemi aracılığıyla hesaplanmıştır. Elde edilen yüksek boyutlu özellik vektörleri ise, boyut indirgeme amacıyla sınırlandırılmış tekil değer ayrışımı (SVD) tekniği ile işlenmiştir. SVM, sinir ağı ve DT algoritmaları, tespit edici modeller olarak yapılandırılmıştır. Modeller, ADFA-LD ve ADFA-WD veri kümeleri kullanılarak eğitilmiş ve değerlendirilmiştir. ADFA-LD kullanılarak gerçekleştirilen deneylerde, SVM tabanlı HIDS; ikili sınıflandırmada %3,34, çoklu sınıflandırmada ise %9,12 yanlış pozitif oranı ile en iyi sonuçları vermiştir. ADFA-WD ile uygulanan deneylerde ise, NN tabanlı HIDS; ikili ve çoklu sınıflandırmalarda sırasıyla %8,63 ve %15,11 FPR değerleriyle en yüksek başarıyı göstermiştir.

[10]'da, sistem çağrı dizilerinin işlenmesi amacıyla ölçünlü kelime torbası, mantıksal BoW, olasılıksal BoW ve TF-IDF BoW teknikleri uygulanmıştır. Bu ön işleme yaklaşımlarıyla dönüştürülen sistem çağrısı verileri kullanılarak KNN, RIPPER, J48, SVM, RF ve NB gibi farklı makine öğrenimi algoritmaları ile modeller inşa edilip değerlendirilmiştir. Söz konusu modellerin oluşturulması ve test edilmesi süreçlerinde ADFA-LD veri kümesi ile sanal makine monitörü (VMM) kötücül yazılım veri kümesi kullanılmıştır. ADFA-LD verisiyle birlikte ölçünlü BoW yaklaşımıyla eğitilen RF modeli, %98,40 ACC ve %1,70 FPR ile en başarılı sonucu vermiştir. Öte yandan, VMM veri kümesinde TF-IDF BoW yöntemiyle geliştirilen J48 ve RF modelleri, %100 ACC ve %0 FPR elde ederek en yüksek başarıyı göstermiştir.

[11]'de, bulut ortamları için geliştirilen HIDS'lerdeki ML/DL modellerinin eğitimlerinde ve testlerinde kullanılmak üzere bir veri kümesi elde edilmiştir. Bu veri kümesi, VMI verilerine dayalı olan sistem çağrı dizilerini içerecek şekilde oluşturulmuştur. Oluşturulan veri kümesi, mevcut veri kümeleri ile karşılaştırmalı olarak analiz edilmiştir. Veri kümelerinin analizinde, J48, RF, RIPPER, NB, KNN ve SVM modelleri kullanılmıştır. Bu ML modelleri geliştirilmeden önce, veri kümelerindeki sistem çağrı dizileri vektör uzay modeline (VSM) dönüştürülmüştür. Deneylerde en başarılı ML modelleri, UNM veri kümesinde %99,60 ACC ile KNN, ADFA-LD veri kümesinde %97,53 ACC ile RF, LID-DS veri kümesinde %97,51 ACC ile J48 ve önerilen veri kümesi üzerinde %99,69 ACC ile KNN olmuştur. Deneyler sonucunda önerilen veri kümesinin, bulut platformlarındaki HIDS'ler için en uygun veri kümesi olduğu öne sürülmüştür.

[12]'de, kiracı sanal makinelerindeki (TVM) normal proseslere saldıran kötücül yazılımları tespit edebilen bir yaklaşım önerilmiştir. İç gözlem destekli evrimsel n-gram torbası (bag of n-gram) yaklaşımı (vServiceInspector) olarak adlandırılan bu yaklaşımda, günlükler yakalanmıştır ve günlükler içeresindeki davranışlar n-gram'lar olarak çıkartılmıştır. Ayrıca, optimum özellikleri elde etmek için genetik algoritma (GA) kullanılmıştır. Kötücül yazılımlar, evrişimsel sinir ağı modeli tarafından tespit edilmiştir. UNM veri kümesinin çeşitli alt kümelerinde yapılan deneylerde, GA tarafından seçilen özellikleri kullanan CNN, %83,13'ten %99,63'e kadar ACC değerlerine ulaşmıştır. Kaliforniya Üniversitesi veri kümesinin (Barecloud) çeşitli alt kümelerinde gerçekleştirilen deneylerde, GA'nın seçtiği özellikleri kullanan CNN, %97,8'den %99'a kadar ACC değerleri ile başarım göstermiştir.

[13]'te, IoT cihazlarında ana bilgisayar tabanlı anomali tespit yaklaşımı tanıtılmıştır. Önerilen yaklaşımda anomalilerin tespitinde, normal türdeki sistem çağrı dizilerini temsil eden Markov zincirleri kullanılmıştır. Öte yandan önerilen yaklaşımla, önceden belirlenmiş olasılık eşiği ve sabit uzunluk bölümleme sorunlarının çözümüne katkı sağlanması amaçlanmıştır. Gerçekleştirilen deneylerde önerilen yaklaşım, UNM veri kümesinin bir alt kümesi olan UNM LPR üzerinde

%99,93 ACC ve bir diğer alt kümesi olan UNM sendmail'de %100 ACC değerlerine ulaşmıştır. Bu çalışmada oluşturulan PiData veri kümesinde ise %99,90 ACC değeri ile başarım göstermiştir.

[14]'te, sistem çağrı dizilerine dayalı yeni bir saldırı tespit çerçevesi geliştirilmiştir. Bu çerçevede, sistem çağrılarının sıklık bilgisine dayanan anomali tespit yöntemi ile uzun kısa süreli bellek öğrenme yöntemi bir araya getirilerek melez bir yaklaşım benimsenmiştir. LSTM modeli, daha önce gözlemlenen sistem çağrı dizileri üzerinde saldırıların tanımlanmasında kullanılmıştır. Diğer yandan, sıklık tabanlı yöntemde, ön işleme sürecinde BoW, n-gram ve TF-IDF teknikleri uygulanmış; ardından sıklık tabanlı yöntem aracılığıyla farklı saldırı türleri sınıflandırılmıştır. Nihai sınıflandırma işlemi ise, ağırlıklı ortalamaya dayalı bir topluluk model aracılığıyla gerçekleştirilmiştir. ADFA-LD veri kümesi kullanılarak eğitilen saldırı tespit çerçevesi, %97,20 ACC ve %2,40 FPR ile yüksek başarım göstermiştir.

[15]'te, sistem çağrı dizilerini gruplandıran yeni bir risk temelli yöntem sunulmuştur. Bu yöntemde, fonksiyonların risk düzeylerine dayalı olarak risklilik puanları belirlenmiştir. Önerilen yöntemin uygulanması sonucunda, SVM ve DT modellerinin doğruluk oranlarında sırasıyla %23,4 ve %7,6 oranında artış sağlanmıştır. Bu risk tabanlı yöntem ile geliştirilen SVM ve DT modellerinin başarımları, AWSCTD veri kümesi üzerinde test edilmiştir. Elde edilen sonuçlara göre, SVM modeli, 1000 uzunluklu dizilerde %96,80 ACC değeriyle en yüksek başarıyı sergilemiştir.

[16]'da, takviyeli öğrenme (RL) temelli yeni bir HIDS olan RL-HIDS tanıtılmıştır. Tanıtılan bu tespit sisteminde, doğal dil işleme (DDİ) ve RL yöntemlerinden birlikte faydalanılmıştır. RL-HIDS, belirli saldırı günlüklerinden hareketle kural kümeleri türetecek şekilde yapılandırılmıştır. Bu kural kümeleri, NLP alanındaki çıkarımsal özetleme tabanlı yaklaşımlar ile RL yöntemleri tarafından elde edilmiştir. RL-HIDS geliştirilirken, ön işlem aşamasında Word2Vec ve TextRank algoritmaları kullanılmıştır. Bu tespit sisteminde tespit edici olarak, LSTM tabanlı Seq2Seq modeli içeren RL modeli yerleştirilmiştir. RL-HIDS'in etkinliği, ADFA-LD ve LID-DS 2021 veri kümeleri üzerinde değerlendirilmiş ve ADFA-LD veri kümesindeki saldırı örnekleri üzerinde ortalama %97,10 ACC elde edilmiştir.

[17]'de, IoT cihazlarına yönelik sistem çağrı dizilerine ve merkezi tabanlı vektör uzay modeline dayalı bir anomali tespit yöntemi sunulmuştur. Kaynak tüketimi açısından verimli olması amacıyla tasarlanan bu yöntemde, TF-IDF yöntemi ön işlem aşamasında kullanılmıştır. Deneylerde önerilen yöntem, UNM veri kümesinin bir alt kümesi olan UNM LPR üzerinde %99,66 ACC ve %0,6 FPR değerlerine ulaşmıştır. UNM'nin bir diğer alt kümesi olan UNM sendmail'de %99,72 ACC ve %0,28 FPR değerleri ile başarım göstermiştir. ADFA-LD üzerinde ise, %99,02 ACC ve %1,96 FPR değerlerine sahip olmuştur.

[18]'de, çeşitli ML modellerinden oluşan yığınlama topluluğu tabanlı HIDS'ler geliştirilmiştir. Bu tespit sistemleri geliştirilirken ML modellerinin ön işlem sürecinde, n-gram, ölçünlü BoW, ikili BoW, olasılıksal BoW ve TF-IDF BoW teknikleri kullanılmıştır. Ayrıca, özellik seçiminde karşılıklı bilgi ve k-ortalamalar kümeleme algoritmaları birlikte uygulanmıştır. Geliştirilen tespit sistemlerinde, KNN, DT, LR, RF, XGBoost ve AdaBoost modelleri, tespit ediciler olarak yerleştirilmiştir. Bireysel modeller olarak tanımlanan KNN, DT, LR ve RF modellerinin çıktıları, XGBoost ve AdaBoost modellerine girdi olarak verilmiş ve bu şekilde yığınlama topluluğu tabanlı modeller/HIDS'ler oluşturulmuştur. Söz konusu tespit sistemleri, ADFA-LD ve ADFA-WD veri kümeleri temelinde oluşturulan çeşitli BoW veri kümeleri ile eğitilmiş ve değerlendirilmiştir. ADFA-LD veri kümesiyle gerçekleştirilen deneylerde, ölçünlü BoW veri kümesi üzerinde oluşturulan yığınlama topluluğu tabanlı XGBoost modeli %97,47 ACC değeri elde etmiştir. ADFA-WD veri kümesindeki deneylerde, ölçünlü BoW veri kümesi ile eğitilmiş yığınlama topluluğu tabanlı XGBoost modeli, %91,63 ACC değerine sahip olmuştur. Çizelge 2, bu bölümde bahsedilen kaynaklardaki çalışmaları özetlemektedir.

Çizelge-2: Kaynaklardaki Çalışmaların Çeşitli Özellikleri

Çalışma	Veri Kümesi	Tespit Etme Modeli/Yöntemi	En Yüksek Başarım
[7]	UNM live LPR UNM sendmail	FSVM	UNM live LPR: %92,63 DR ve %6,16 FAR UNM sendmail: %84,61 DR ve %4,51 FAR
[8]	UNM'nin alt kümeleri Elog veri kümesi	DT, RF, NB, AdaBoost, KNN, SVM ve yumuşak oylama	UNM sentetik LPR: %99,29–%99,93 ACC Elog: %98,17 ACC
[9]	ADFA-LD ADFA-WD	SVM, NN ve DT	ADFA-LD: İkili sınıflandırmada %3,34 FPR ADFA-WD: İkili sınıflandırmada %8,63 FPR
[10]	ADFA-LD VMM	KNN, RIPPER, J48, SVM, RF ve NB	ADFA-LD: %98,40 ACC ve %1,70 FPR ADFA-WD: %100 ACC ve %0 FPR
[11]	UNM ADFA-LD LID-DS [11]'de önerilen veri kümesi	J48, RF, RIPPER, NB, KNN ve SVM	UNM: %99,60 ACC ADFA-LD: %97,53 ACC LID-DS: %97,51 ACC [11]'de önerilen veri kümesi: %99,69 ACC
[12]	UNM'nin alt kümeleri Barecloud	CNN	UNM: %83,13–%99,63 ACC Barecloud: %97,8–%99 ACC
[13]	UNM LPR	Markov zincirleri	UNM LPR: %99,93 ACC

	UNM sendmail PiData		UNM sendmail: %100 ACC PiData: %99,90 ACC
[14]	ADFA-LD	LSTM	ADFA-LD: %97,20 ACC ve %2,40 FPR
[15]	AWSCTD	SVM ve DT	AWSCTD: %96,80 ACC
[16]	ADFA-LD LID-DS 2021	LSTM	ADFA-LD: %97,10 ACC LID-DS 2021: %95,80 ACC
[17]	UNM LPR UNM sendmail ADFA-LD	Merkezi tabanlı vektör uzay modeline dayalı yöntem	UNM LPR: %99,66 ACC ve %0,6 FPR UNM sendmail: %99,72 ACC ve %0,28 FPR ADFA-LD: %99,02 ACC ve %1,96 FPR
[18]	ADFA-LD ADFA-WD	KNN, DT, LR, RF, XGBoost ve AdaBoost	ADFA-LD: %97,47 ACC ADFA-WD: %91,63 ACC

1.2 Güdü ve Katkı

Sistem çağrı dizileri, kullanıcı ile işletim sistemi arasındaki etkileşimi ayrıntılı biçimde yansıttığından, HIDS'lerin etkinliğini artırmada kritik bir rol oynamaktadır. Ancak, sıralı verilerin yüksek boyutluluğu ve bağlamsal ilişkilerin karmaşıklığı, bu verilerin doğrudan ML modellerine aktarılmasını zorlaştırmaktadır. Kaynaklarda n-gram ve BoW gibi ön işlem yöntemleri kullanılmış olsa da bu yöntemlerin modellerin başarımına olan katkıları, farklı veri kümeleri üzerinden incelenmesi ve karşılaştırılması gerekmektedir. Ayrıca, düşük FPR ve yüksek ACC değerlerine aynı anda ulaşabilmek, gerçek zamanlı kullanılabilirlik açısından önemli bir gereklilik olarak öne çıkmaktadır. Bu bağlamda, çalışmamızın temel motivasyonu, farklı ön işlem yöntemlerini bir arada değerlendirerek sistem çağrı dizileri üzerinde etkili ve güvenilir ML modelleri/HIDS'leri geliştirmek olmuştur. Bu motivasyon kapsamında, [18]'de kullanılan ön işlem yöntemleri ve ML modelleri temel alınarak, çeşitli ML modellerinin/HIDS'lerinin geliştirilmesi amaçlanmaktadır. [18]'den farklı olarak, modellerin/HIDS'lerin, UNM ve MIT veri kümeleri kullanılarak geliştirilmesi hedeflenmektedir. Bu hedefler doğrultusunda gerçekleştirilen çalışmalar sonucunda elde edilen katkılar, aşağıda maddeler halinde verilmektedir.

- UNM live LPR, MIT live LPR ve bu iki veri kümesinin birleşiminden oluşan UNM-MIT live LPR veri kümeleri üzerinde, n-gram, ölçünlü BoW, ikili BoW, olasılıksal BoW, TF-IDF BoW ve özellik seçim yöntemleri uygulanmaktadır. Sonuç olarak, UNM, MIT ve UNM-MIT tabanlı BoW veri kümeleri elde edilmektedir.
- Seçilen özellikleri içeren BoW veri kümeleri kullanılarak, bireysel KNN, DT, LR, RF ve yığınlama topluluğu tabanlı XGBoost ve AdaBoost modelleri/HIDS'leri geliştirilmektedir.
- Geliştirilen modellerin başarımları, çeşitli metrikler aracılığıyla karşılaştırmalı olarak analiz edilmektedir. UNM tabanlı BoW veri kümeleri üzerinde geliştirilen en başarılı modeller, oldukça düşük FPR ve yüksek ACC değerleri ile kaynaklardaki mevcut yöntemlere kıyasla üstün başarım göstermektedir.

1.3 Organizasyon

Çalışmanın ikinci bölümünde, modellerin/HIDS'lerin geliştirilmesinde kullanılan veri kümelerine ait bilgilerden bahsedilmektedir. Üçüncü bölümde, n-gram, BoW ve özellik seçim yöntemlerinin, veri kümeleri üzerinde nasıl uygulandıkları açıklanmaktadır. Dördüncü bölümde, modellerin/HIDS'lerin geliştirilme süreci ifade edilmektedir ve modellere/HIDS'lere ait deney sonuçları sunulmaktadır. Son bölümde ise, bu çalışmaya kısa bir bakış açısı sağlanmaktadır.

2. Veri Kümeleri

UNM veri kümesi [19], Linux işletim sisteminde bulunan çeşitli programlardan toplanan gerçek ve sentetik sistem çağrı dizilerini içermektedir. Massachusetts Teknoloji Enstitüsü veri kümesinde [19] ise, LPR programına ait gerçek sistem çağrı dizileri bulunmaktadır. Bu veri kümelerinde her bir sistem çağrı dizisi, çağrılma sırasına göre sıralanmış sistem çağrılarından oluşmaktadır. Her bir sistem çağrısı, benzersiz kimlik numaraları ile temsil edilmektedir. Bu veri kümelerindeki ham veriler, proses kimlik numarasından ve prosesin çağırdığı sistem çağrı ID'lerinden oluşmaktadır. Şekil 1'de, UNM'de veya MIT'te ham olarak bulunan bir veri örneğinin kısaltılmış bir hali gösterilmektedir. Şekil 1'de virgülle ayrılmış sayı çiftlerinden birincisi PID'ye, ikincisi ise sistem çağrı ID'sine karşılık gelmektedir. Ham veriler içerisinden sistem çağrı dizilerini elde etmek için PID'lerin kaldırılması gerekmektedir. Daha sonra, her bir PID'ye ait olan sistem çağrılarından oluşan sistem çağrı dizileri elde edilebilmektedir. Şekil-2'de, Şekil-1'deki veri örneğinden elde edilen sistem çağrı dizisi yer almaktadır.

```
22799 4, 22799 2, 22799 66, 22799 66, 22799 4, 22799 138, 22799 66, 22799 5, 22799 23, 22799 45,
22799 4, 22799 27, 22799 66, 22799 5, 22799 4, 22799 2, 22799 66, 22799 66, 22799 66, 22799 5,
22799 4, 22799 2, 22799 66, 22799 66, 22799 5, 22799 5, 22799 105, 22799 104, 22799 104, 22799
106, 22799 105, 22799 104, 22799 104, 22799 106, 22799 105, 22799 104, 22799 104, 22799 106,...
```

Şekil-1: Ham veri örneği

```
4, 2, 66, 66, 4, 138, 66, 5, 23, 45, 4, 27, 66, 5, 4, 2, 66, 66, 66, 5, 4, 2, 66, 66, 5, 5, 105, 104, 104,
106, 105, 104, 104, 106, 105, 104, 104, 106,...
```

Şekil-2: Örnek sistem çağrı dizisi

UNM veri kümesi, çeşitli alt kümelerden oluşmaktadır. Bu alt kümeler, LPR, named, xlock, login, PS, inetd, stide, FTP ve sendmail olarak adlandırılan programların proseslerine ait sistem çağrı dizilerini içermektedir. Bu çalışmada, LPR programına ait sistem çağrı dizilerini içeren ve UNM canlı LPR (UNM live LPR) olarak adlandırılan alt küme kullanılmaktadır. Gerçek sistem çağrı dizilerinden oluşan UNM live LPR alt kümesinde, normal ve saldırı olmak üzere iki tür

bulunmaktadır. UNM live LPR alt kümesinin ham dosyalarının içerisinde, 1231 tane normal ve 1001 tane saldırı olmak üzere, toplam 2232 tane örnek kullanılabilir halde bulunmaktadır.

Öte yandan, MIT veri kümesi, MIT canlı LPR (MIT live LPR) olarak adlandırılmaktadır. Bu veri kümesi, LPR programına ait gerçek sistem çağrı dizileri içermektedir. Bu veri kümesinde kullanılabilir olarak bulunan sistem çağrı dizilerinin 2697 tanesi normal ve 1001 tanesi saldırı olarak kategorize edilmektedir. Dolayısıyla, MIT live LPR veri kümesinde toplam 3698 tane sistem çağrı dizisi bulunmaktadır. MIT live LPR ve UNM live LPR kümelerine ait istatistiki bilgiler, Çizelge 3'te verilmektedir.

Çizelge-3: UNM Live LPR ve MIT Live LPR Veri Kümelerine ait İstatistiki Bilgiler

Veri Kümesi	Normal Sistem Çağrı Dizilerinin Sayısı	Saldırı Sistem Çağrı Dizilerinin Sayısı	Sistem Çağrı Dizilerinin Toplam Sayısı
UNM live LPR	1231	1001	2232
MIT live LPR	2697	1001	3698

Bu çalışmada, UNM live LPR ve MIT live LPR veri kümelerine ait normal türdeki sistem çağrı dizileri içerisinden, rastgele 1001'er tane dizi seçilmektedir. Böylece, normal ve saldırı türündeki sistem çağrı dizilerinin sayıları dengelenmektedir. Netice olarak, UNM live LPR veri kümesine ait 2002 tane ve MIT live LPR veri kümesine ait 2002 tane sistem çağrı dizisi kullanılarak modeller/HIDS'ler geliştirilmektedir.

Öte yandan, UNM live LPR ve MIT live LPR veri kümeleri, aynı sürüme ait işletim sisteminden ve LPR programından türetilmiştir. Buna bağlı olarak, bu iki veri kümesi içerisindeki sistem çağrı dizilerinin birlikte kullanılmasında bir engel bulunmamaktadır. Bu bakış açısıyla, bu iki veri kümesinin her birinden rastgele 1001'er tane normal türden sistem çağrı dizisi seçilmektedir. Toplam 2002 tane olan bu normal diziler ile iki veri kümesindeki toplam 2002 tane saldırı dizileri birleştirilmektedir. Bunun sonucunda, normal ve saldırı türleri eşit olan ve 4004 tane sistem çağrı dizisi içeren yeni bir veri kümesi elde edilmektedir. Bu veri kümesi, UNM-MIT live LPR olarak adlandırılmaktadır ve modellerin/HIDS'lerin geliştirilmesinde kullanılmaktadır.

3. Ön İşlem ve Özellik Seçimi

Bu bölümde ilk olarak, UNM live LPR ve MIT live LPR veri kümelerinde bulunan ham veriler içerisinden, sistem çağrı dizilerinin nasıl çıkartıldığı açıklanmaktadır. Daha sonra, çıkartılan sistem çağrı dizileri üzerine n-gram yönteminin uygulanması sonucu üretilen n-gram veri kümelerinden bahsedilmektedir. Bununla birlikte, üretilen n-gram veri kümelerine çeşitli BoW yöntemlerinin uygulanması neticesinde oluşturulan BoW veri kümeleri ifade edilmektedir. Son olarak ise oluşturulan BoW veri kümeleri üzerinde gerçekleştirilen özellik seçme işlemi açıklanmaktadır.

3.1 Sistem Çağrı Dizilerinin Çıkartılması

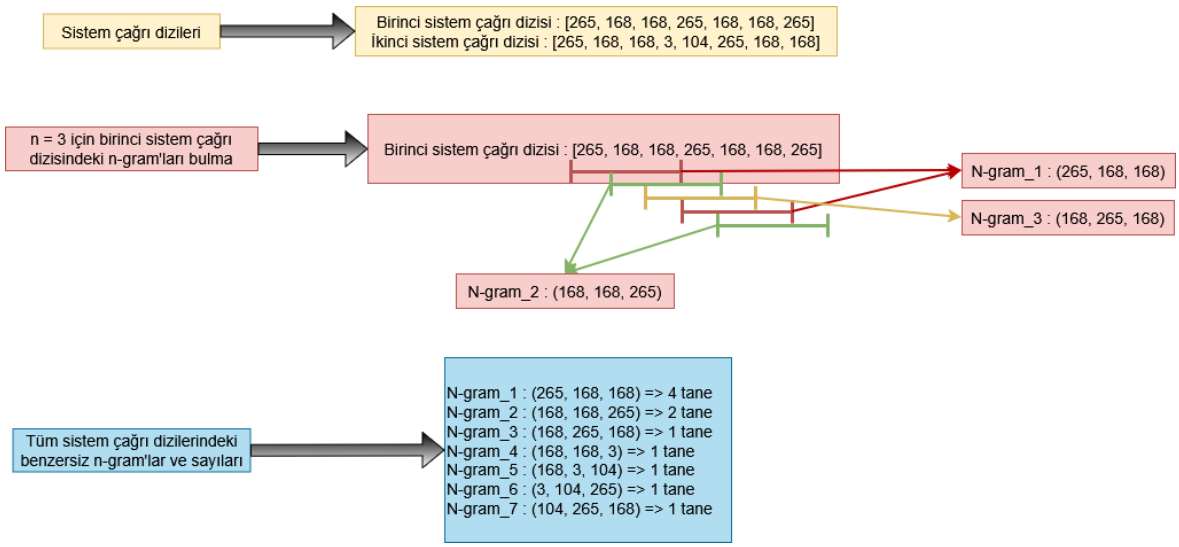
UNM live LPR ve MIT live LPR veri kümelerindeki normal türden olan ham veriler, sıkıştırılmış dosyalar içerisinde, ".log.int" uzantılı dosyalarda bulunmaktadır. Her bir ".log.int" uzantılı dosya, bir prosesin sıralı olarak çağırdığı sistem çağrılarını içermektedir. Netice olarak, bir ".log.int" uzantılı dosyadaki sıralı sistem çağrıları, bir sistem çağrı dizisini ifade etmektedir. Bu dosyalardaki ham verilere ait örnek Şekil 1'de ve çıkartılan örnek sistem çağrı dizisi Şekil 2'de verilmektedir.

Dosyaların her birinden sistem çağrı dizilerini çıkartmak için ilk olarak, PID'ler silinmektedir. Daha sonra, sistem çağrı ID'leri sıralarına göre, virgüllerle ayrılarak ".csv" uzantılı dosyalara kaydedilmektedir.

Öte yandan, veri kümelerindeki saldırı türündeki ham veriler, her bir veri kümesine ait tek bir ".int" uzantılı dosyada bulunmaktadır. Bu dosya, farklı proseslere ait sıralı sistem çağrılarından oluşmaktadır. Dosyadan proseslere ait sistem çağrı dizilerini çıkartmak için ilk olarak, PID'lere göre sistem çağrı ID'leri sırasıyla alınmaktadır. Daha sonra, sistem çağrı ID'leri virgüllerle ayrılarak, tek bir satır haline getirilmektedir. Böylece, her bir prosese ait sistem çağrı dizileri elde edilmektedir. Elde edilen saldırı türündeki sistem çağrı dizileri, normal türden olan sistem çağrı dizilerinin bulunduğu ".csv" dosyalarına kaydedilmektedir. Nihayetinde, UNM live LPR ve MIT live LPR veri kümelerine ait sistem çağrı dizileri elde edilmektedir.

3.2 Sistem Çağrı Dizileri Üzerinde N-Gram Yönteminin Uygulanması

Sistem çağrı dizilerindeki çağrılar arasındaki ilişkileri belirlemek amacıyla n-gram yöntemi kullanılmaktadır. Bu yöntem sayesinde, çağrılar arasındaki bağlamlar n-gram yapıları ile temsil edilmektedir. Bir sistem çağrı dizisine özgü n-gram'lar elde edilirken, uzunluğu önceden belirlenmiş olan n değerine sahip bir pencere kullanılmaktadır. Pencere, dizinin ilk sistem çağrısından başlayacak şekilde konumlandırılarak ilk n-gram oluşturulmaktadır. Sonraki n-gram'lar ise pencerenin her adımda bir çağrı ileri kaydırılmasıyla elde edilmektedir. Bu işlem sonucunda, ilgili dizideki tüm n-gram'lar ortaya çıkarılmaktadır. Örnek olarak, Şekil-3'te iki farklı sistem çağrı dizisine ait n-gram çıkarım süreci gösterilmektedir.



Şekil-3: Sistem çağrı dizilerine ait n-gram'ların çıkartılma süreci

UNM live LPR, MIT live LPR ve UNM-MIT live LPR veri kümelerine ait seçilmiş sistem çağrı dizilerindeki benzersiz n-gram'lar bulunurken, n değeri 5 olarak seçilmektedir. Bu değer, sistem çağrısı dizilerindeki kısa ve orta uzunluktaki bağlamları yeterince temsil ederek modellerin anlamlı örüntüleri öğrenmesini sağlamaktadır. Daha küçük n değerleri ile çok kısa bağlamlar ortaya çıkartılırken ve daha büyük n değerleri ile nadir görülen uzun bağlamlar yakalanmaktadır. Dolayısıyla, modellerin genelleme yetenekleri olumsuz olarak etkilenebilmektedir. Bu nedenle n değerinin 5 olması, hem bağlam bilgisini yeterli düzeyde sunmakta hem de veri temsiliyetini dengede tutmaktadır.

Sistem çağrı dizileri üzerine n-gram yöntemi uygulandıktan sonra, UNM live LPR veri kümesindeki sistem çağrı dizilerine ait benzersiz 311 tane 5-gram elde edilmektedir. MIT live LPR veri kümesindeki sistem çağrı dizilerinden, 391 tane benzersiz 5-gram çıkartılmaktadır. Son olarak, UNM-MIT live LPR veri kümesi içerisinden, 420 tane benzersiz 5-gram bulunmaktadır.

Sistem çağrı dizilerine ait 5-gram'lar bulunduktan sonra, bu 5-gram'lardan oluşan veri kümeleri oluşturulmaktadır. Oluşturulan veri kümelerinin özellikleri, 5-gram'lar olmaktadır. Bir sistem çağrı dizisinin özelliklerinin/5-gram'larının değerlerine ise, ilgili dizideki 5-gram'ların sıklık değerleri ayarlanmaktadır. Sonuç olarak, UNM live LPR, MIT live LPR ve UNM-MIT live LPR veri kümelerine ait 5-gram veri kümeleri elde edilmektedir. Şekil 3'te bulunan benzersiz n-gram'lardan oluşan örnek bir n-gram veri kümesi, Şekil-4'te gösterilmektedir.

Dizi	N-gram_1	N-gram_2	N-gram_3	N-gram_4	N-gram_5	N-gram_6	N-gram_7
Birinci	2	2	1	0	0	0	0
İkinci	2	0	0	1	1	1	1

Şekil-4: Örnek bir n-gram veri kümesi

3.3 Ölçünlü BoW Yönteminin Uygulanması

Metin madenciliğinde yaygın olarak kullanılan ölçünlü BoW yöntemi, metinlerde yer alan benzersiz kelimeleri belirlemekte ve bu kelimelerin sıklıklarını hesaplamaktadır. Bu doğrultuda, metinlerden oluşan bir veri kümesine ölçünlü BoW yöntemi uygulandığında, benzersiz kelimelerin özellik olarak temsil edildiği yeni bir veri kümesi elde edilmektedir. Ayrıca, bu veri kümesinde her bir özelliğin değeri, ilgili kelimenin sıklığı ile ifade edilmektedir.

Oluşturulan 5-gram veri kümelerine ölçünlü BoW yöntemi uygulandığında, her bir 5-gram benzersiz bir kelime olarak değerlendirilmektedir. Bu 5-gram'ların değerleri ise, metinlerdeki kelime sıklıklarına benzer biçimde, sistem çağrı dizileri içerisindeki görülme sıklıklarıyla belirlenmektedir. Başka bir ifadeyle, her bir benzersiz 5-gram'ın sistem çağrı dizilerinde ortaya çıkma sıklığı, ilgili 5-gram'ın değeri olarak atanmıştır. Dolayısıyla, 5-gram veri kümeleri, UNM live LPR, MIT live LPR ve UNM-MIT live LPR veri kümelerinden seçilen sistem çağrı dizilerine ölçünlü BoW yönteminin uygulanmasıyla elde edilmiş durumdadır. Bu doğrultuda, bu veri kümeleri sırasıyla, UNM tabanlı ölçünlü BoW veri kümesi, MIT tabanlı ölçünlü BoW veri kümesi ve UNM-MIT tabanlı ölçünlü BoW veri kümesi olarak adlandırılmaktadır.

3.4 İkili BoW Yönteminin Uygulanması

İkili BoW yöntemi, metin veri kümelerinde yer alan kelimelerin sıklıklarını dikkate almak yerine, bu kelimelerin metinlerde bulunup bulunmadığını esas almaktadır. Bir metni temsil eden özellikler olarak değerlendirilen kelimeler, metin içerisinde yer aldığında 1, yer almadığında ise 0 değeri ile gösterilmektedir. Böylece metinler, kelimelerin varlık durumlarını yansıtan ikili değerler (0 ve 1) ile temsil edilmektedir.

İkili BoW yöntemi 5-gram veri kümelerine uygulandığında, her biri kelime olarak kabul edilen 5-gram'ların değerleri dikkate alınmaktadır. Bir sistem çağrı dizisine ait 5-gram'ın değeri 0'dan büyükse, bu değer 1 olarak güncellenmektedir. Buna karşılık, 5-gram'ın değeri 0 olduğunda, aynı şekilde 0 olarak kalmaktadır. Sonuç olarak, yalnızca 0 ve 1 değerlerinden oluşan, UNM tabanlı ikili BoW veri kümesi, MIT tabanlı ikili BoW veri kümesi ve UNM-MIT tabanlı ikili BoW veri kümesi elde edilmektedir.

3.5 Olasılıksal BoW Yönteminin Uygulanması

Olasılıksal BoW yöntemi, kelimelerin metinlerde yer alma olasılıklarını hesaplamaya dayanmaktadır. Bir kelimenin olasılık değeri, metin içerisindeki sıklığının, metindeki tüm kelimelerin sıklık değerleri toplamına bölünmesiyle elde edilmektedir. Bu şekilde, metinler özellik olarak kullanılan kelimelerin olasılık değerleri ile temsil edilmektedir.

Olasılıksal BoW yöntemi 5-gram veri kümelerine uygulandığında, kelime olarak kabul edilen 5-gram'ların olasılık değerleri Formül-1 kullanılarak hesaplanmaktadır. Bu hesaplamada, herhangi bir sistem çağrı dizisindeki 5-gram'ın değeri, tüm 5-gram'ların toplam değerine bölünmektedir. Her bir sistem çağrı dizisine ait 5-gram'ların olasılık değerleri belirlendikten sonra, bu değerleri içeren olasılıksal BoW veri kümeleri oluşturulmaktadır. Netice itibarıyla, UNM tabanlı olasılıksal BoW veri kümesi, MIT tabanlı olasılıksal BoW veri kümesi ve UNM-MIT tabanlı olasılıksal BoW veri kümesi elde edilmektedir.

$$P(5-gram_i) = \frac{Dizideki\ 5{-}gram_i'nin\ değeri}{Dizideki\ 5{-}gram'ların\ toplam\ değeri} \quad (1)$$

3.6 TF-IDF BoW Yönteminin Uygulanması

TF-IDF BoW yöntemi, metinlerdeki kelimelerin göreli önemini belirlemek için kullanılan bir yaklaşımdır. Bu yöntemde, bir kelimenin metin içerisindeki önemi, hem kelimenin o metinde ne sıklıkla geçtiğine hem de tüm belgeler arasındaki yaygınlığına bağlı olarak hesaplanmaktadır. Öncelikle metinlerde yer alan tüm benzersiz kelimeler belirlenmekte ve bunların TF değerleri elde edilmektedir. TF değeri, kelimenin belirli bir metin içerisindeki tekrar sayısını ifade etmektedir. Ardından, kelimelerin metinler arasındaki dağılımını ölçen IDF değerleri hesaplanmaktadır. IDF değerleri hesaplanırken, Formül 2 kullanılmaktadır. Son aşamada TF ve IDF değerlerinin çarpımı gerçekleştirilerek, her kelimenin TF-IDF değeri hesaplanmaktadır. Böylece, her metin benzersiz kelimelerin TF-IDF değerleri aracılığıyla temsil edilmektedir.

$$IDF(kelime) = log_{10}\left(\frac{Toplam\ metin\ sayısı}{Kelimeyi\ içeren\ metin\ sayısı}\right) \quad (2)$$

TF-IDF BoW yönteminin 5-gram veri kümelerine uygulanabilmesi için öncelikle, 5-gram'larla temsil edilen sistem çağrısı dizilerinin metin formatına dönüştürülmesi gerekmektedir. Bu dönüşüm sürecinde, bir sistem çağrı dizisini oluşturan 5-gram'lar arasından değeri sıfırdan farklı olanların adları kullanılmaktadır. Her bir 5-gram adı, sahip olduğu değer kadar tekrarlanarak aralarında boşluk bırakılmak suretiyle yan yana eklenmektedir. Bu işlem sonucunda, her bir sistem çağrı dizisi, 5-gram adlarından oluşan metinler haline getirilmektedir. Şekil-4'te sunulan n-gram veri kümesindeki birinci sistem çağrı dizisinin, n-gram tabanlı metin karşılığı Şekil-5'te gösterilmektedir.

Birinci sistem çağrı dizisinin metin hali =>	"N-gram_1 N-gram_1 N-gram_2 N-gram_2 N-gram_3"

Şekil-5: Örnek bir sistem çağrı dizisinin n-gram tabanlı metin hali

5-gram veri kümelerindeki her bir sistem çağrı dizisinin metin formatına dönüştürülmesinin ardından, bu metinler TF-IDF BoW yöntemiyle analiz edilmektedir. Bu aşamada, metinlerde yer alan kelimeler (5-gram adları) için TF-IDF değerleri, sklearn kitaplığında yer alan TfidfVectorizer yöntemi aracılığıyla hesaplanmaktadır. Hesaplamalarda, söz konusu yöntemin varsayılan parametreleri kullanılmaktadır. Böylece, TF-IDF değerlerine sahip 5-gram'ları özellik olarak içeren ve sistem çağrı dizilerinin bu özellikler aracılığıyla temsil edildiği TF-IDF BoW veri kümeleri elde edilmektedir. Başka bir ifadeyle, UNM tabanlı TF-IDF BoW veri kümesi, MIT tabanlı TF-IDF BoW veri kümesi ve UNM-MIT tabanlı TF-IDF BoW veri kümesi oluşturulmaktadır.

3.7 Özellik Seçimi

Saldırı tespiti için kullanılan ML modellerinin, yerleştirildikleri ortamlardaki kaynakları verimli bir şekilde tüketmeleri beklenmektedir. Bununla birlikte, bu tür ML modellerinin, saldırıları en erken sürede tespit edebilmeleri gerekmektedir. Bu bağlamda, ML modellerinin geliştirilmelerinde ve uygulanmalarında, az sayıda özellik kullanılması önemli bir avantaj olarak gösterilmektedir. Daha da ötesi, daha az özellik içeren ML modellerinin saldırıları tespit etme başarımları, daha fazla özellik içeren muadilleri ile benzer seviyelerde olması gerekmektedir. Bu gerekliliklerden ve nedenlerden dolayı, oluşturulan çeşitli BoW veri kümelerinin her biri üzerinde özellik seçimi yapılmaktadır. Özellik seçiminde, Şekil 6'daki algoritma [18] kullanılmaktadır. Özellik seçim işleminden sonra elde edilen özellikler ile çeşitli modeller/HIDS'ler geliştirilmektedir.

Girdi: Veri Kümesi
Çıktı: Seçili özelliklere sahip veri kümesi
1: k= 2 {küme sayısı}
2: Özelliklerin MI değerlerinin hesaplanması
3: k-means(k, özelliklerin indeksleriyle birlikte MI değerleri)
4: Büyük MI değerleri içeren kümedeki özelliklerin indekslerine göre seçilmesi

Şekil-6: Özellik seçim algoritması

Şekil-6'daki algoritma, MI yaklaşımı ile k-means algoritmasının birlikte kullanıldığı bir özellik seçim yöntemini ifade etmektedir. Bu yöntemde ilk aşamada, her bir özelliğin MI değeri hesaplanmaktadır. Hesaplanan değerler doğrultusunda, özellikler k-means algoritması aracılığıyla iki ayrı gruba ayrılmaktadır. Bu gruplar, görece yüksek MI değerine sahip özellikleri içeren küme ile düşük MI değerine sahip özellikleri içeren küme şeklinde tanımlanmaktadır. Sonuç olarak algoritma, yüksek MI değerine sahip özellikleri seçerek çıktı olarak sunmaktadır. Özellik seçimi tamamlandıktan sonra, BoW veri kümelerinin özellik sayılarına ait istatistiki bilgiler, Çizelge 4, 5 ve 6'da gösterilmektedir.

Çizelge-4: UNM Tabanlı BoW Veri Kümelerinin Özellik Seçiminden Önce ve Sonra Özellik Sayıları

	UNM Tabanlı Ölçünlü BoW Veri Kümesi	UNM Tabanlı İkili BoW Veri Kümesi	UNM Tabanlı Olasılıksal BoW Veri Kümesi	UNM Tabanlı TF-IDF BoW Veri Kümesi
Ham Özellik Sayısı	311	311	311	311
Seçilen Özellik Sayısı	51	48	140	137

Çizelge-5: MIT Tabanlı Bow Veri Kümelerinin Özellik Seçiminden Önce ve Sonra Özellik Sayıları

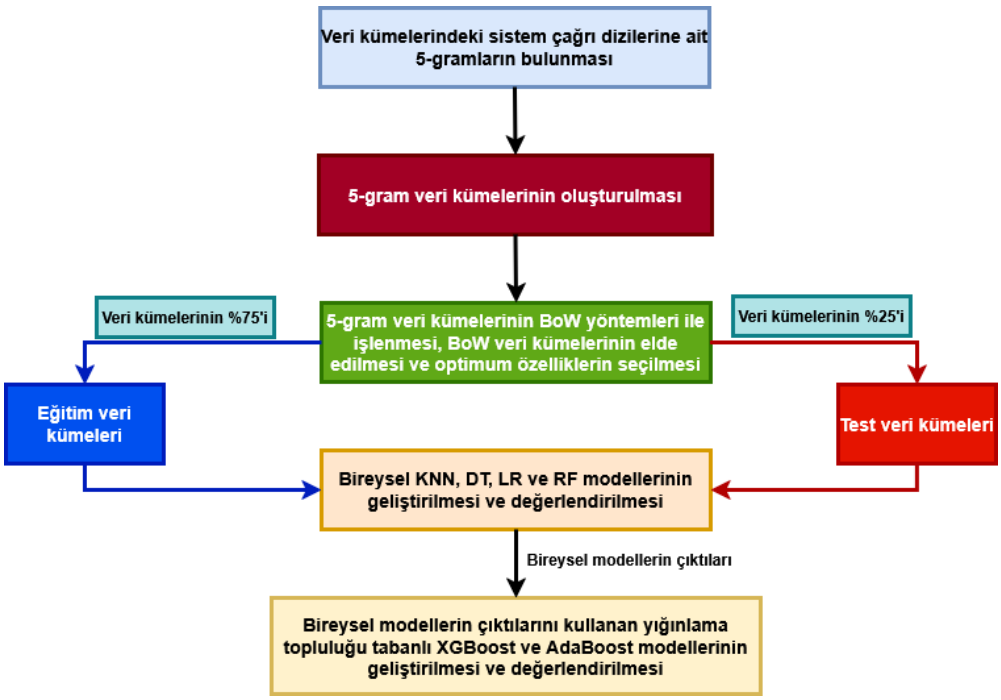
	MIT Tabanlı Ölçünlü BoW Veri Kümesi	MIT Tabanlı İkili BoW Veri Kümesi	MIT Tabanlı Olasılıksal BoW Veri Kümesi	MIT Tabanlı TF-IDF BoW Veri Kümesi
Ham Özellik Sayısı	391	391	391	391
Seçilen Özellik Sayısı	51	46	137	137

Çizelge-6: UNM-MIT Tabanlı Bow Veri Kümelerinin Özellik Seçiminden Önce ve Sonra Özellik Sayıları

	UNM-MIT Tabanlı Ölçünlü BoW Veri Kümesi	UNM-MIT Tabanlı İkili BoW Veri Kümesi	UNM-MIT Tabanlı Olasılıksal BoW Veri Kümesi	UNM-MIT Tabanlı TF-IDF BoW Veri Kümesi
Ham Özellik Sayısı	420	420	420	420
Seçilen Özellik Sayısı	41	37	131	130

4. Modellerin Geliştirilmesi ve Değerlendirilmesi

Modellerin geliştirilme sürecinde gerçekleştirilen işlemlerin akış diyagramı, Şekil 7'de verilmektedir. İlk olarak, UNM live LPR, MIT live LPR ve UNM-MIT live LPR veri kümelerine ait sistem çağrı dizilerinden benzersiz 5-gram'lar çıkartılmaktadır. Ardından, çıkartılan 5-gram'ların özellik olarak kabul edildiği 5-gram veri kümeleri oluşturulmaktadır. Oluşturulan 5-gram veri kümeleri üzerine, çeşitli BoW yöntemleri uygulanmaktadır. Bu uygulamadan sonra, UNM tabanlı, MIT tabanlı ve UNM-MIT tabanlı BoW veri kümeleri elde edilmektedir. Daha sonra, BoW veri kümeleri ile çeşitli bireysel ML modelleri geliştirilmektedir ve değerlendirilmektedir. Bireysel modeller, KNN, DT, LR ve RF modelleri olarak sıralanmaktadır. Bu bireysel modellerin çıktıları ise, yığınlama topluluğu tabanlı XGBoost ve AdaBoost modellerinde kullanılmaktadır. Netice olarak, bireysel ve topluluk olmak üzere, çeşitli ML tabanlı HIDS'ler geliştirilmektedir.



Şekil-7: Modellerin geliştirilme ve değerlendirilme süreci

Bireysel modeller geliştirilirken, BoW veri kümelerinin her biri %75 ve %25 oranlarında bölünmektedir. Kümelerin %75'lik bölümleri eğitimler için kullanılırken, %25'lik bölümler ile testler gerçekleştirilmektedir. Topluluk modelleri ise, bireysel modellerin çıktılarının %75'lik bölümleri ile eğitilirken, geriye kalan %25'lik bölümlerle değerlendirilmektedir. Bireysel ve topluluk modellerinin her biri, her seferinde farklı eğitim ve test kümeleri olacak şekilde beş kez eğitilip test edilmektedir. Modellerin nihai başarım değerleri ise, bu beş yinelemede elde edilen sonuçların ortalaması olarak hesaplanmaktadır. Modeller geliştirilirken, Google Colaboratory ortamı ve Python programlama dili tercih edilmiştir. Bununla birlikte, scikit-learn, XGBoost ve NumPy kitaplıkları kullanılmıştır.

Öte yandan, modellerin hiperparametreleri, yaygın olarak kullanılan ölçünlü değerler ve eğitim/test süresi dengesi göz önünde bulundurularak ayarlanmaktadır. Bireysel modellerde KNN için "n_neighbors = 3", DT için "criterion = entropy ve random_state = 42", LR için "solver = liblinear, max_iter = 1000 ve C = 0.1" ve RF için "n_estimators = 100 ve random_state = 42" hiperparametreleri tercih edilmektedir. Yığınlama topluluğu tabanlı XGBoost modellerinde "objective = binary:logistic" olarak ayarlanırken ve AdaBoost için ise varsayılan hiperparametreler kullanılmaktadır. Ayrıca, modeller üzerinde herhangi bir hiperparametre optimizasyonu uygulanmamıştır. Çizelge 7, modellerde kullanılan hiperparametreleri ve bu hiperparametrelere ait değerleri göstermektedir.

Çizelge-7: Modellerde Kullanılan Hiperparametreler ve Değerleri

Model	Hiperparametre	Değer
KNN	n_neighbors	3
DT	criterion	entropy
	random_state	42
LR	solver	liblinear
	max_iter	1000
	C	0,1
RF	n_estimators	100
	random_state	42
XGBoost	objective	binary:logistic
AdaBoost	Varsayılan parametre değerleri	

4.1 Değerlendirme Ölçüleri

Modellerin/HIDS'lerin başarımları, yaygın olarak kullanılan değerlendirme metrikleri ile ölçülmektedir. Bu ölçüler içerisinde, tutturma, bulma, f1-değeri, FPR, ACC, eğitim süresi ve sınama süresi yer almaktadır. PRC, REC, F1, FPR ve ACC metriklerinin değerleri hesaplanırken, doğru pozitif, doğru negatif, yanlış pozitif ve yanlış negatif değerleri kullanılmaktadır. Metriklerin ve metrikleri hesaplamak için kullanılan değerlerin tanımları, aşağıda maddeler halinde verilmektedir.

- **TP:** Doğru sınıflandırılan pozitif örnek sayısı.
- **TN:** Doğru sınıflandırılan negatif örnek sayısı.
- **FP:** Yanlış sınıflandırılan negatif örnek sayısı.
- **FN:** Yanlış sınıflandırılan pozitif örnek sayısı.
- **PRC:** Doğru sınıflandırılan pozitif örnek sayısının, pozitif olarak sınıflandırılan örnek sayısına oranı.

$$PRC = \frac{TP}{TP + FP} \quad (3)$$

- **REC:** Doğru sınıflandırılan pozitif örnek sayısının, pozitif örnek sayısına oranı.

$$REC = \frac{TP}{TP + FN} \quad (4)$$

- **F1:** PRC ve REC metrik değerlerinin harmonik ortalaması.

$$F1 = 2 * \frac{PRC * REC}{PRC + REC} \quad (5)$$

- **FPR:** Yanlış sınıflandırılan negatif örnek sayısının, negatif örnek sayısına oranı.

$$FPR = \frac{FP}{FP + TN} \quad (6)$$

- **ACC:** Doğru sınıflandırılan örnek sayısının, tüm örnek sayısına oranı.

$$ACC = \frac{TP + TN}{TP + TN + FP + FN} \quad (7)$$

- **Eğitim Süresi:** Modeli eğitmek için harcanan süre.
- **Test Süresi:** Modeli değerlendirmek için harcanan süre.

4.2 Modellerin Değerlendirilmesi

UNM tabanlı, MIT tabanlı ve UNM-MIT tabanlı BoW veri kümeleri üzerinde geliştirilen modellerin deney sonuçları, Çizelge-8, 9 ve 10'da verilmektedir. Bu çizelgelerde koyu kırmızı renkte belirtilen başarım sonuçları, ACC, FPR, eğitim süresi ve test süresi açısından en ayırt edici sonuçlara karşılık gelmektedir.

4.2.1. UNM Tabanlı BoW Veri Kümeleri Özelinde Değerlendirme

UNM tabanlı BoW veri kümeleri üzerinde geliştirilen modellerin başarım değerleri, Çizelge 8'de verilmektedir. Çizelge 8 incelendiğinde, aşağıda maddeler halinde sıralanan sonuçlar ortaya çıkmaktadır.

- Tüm veri kümelerinde, yığınlama topluluğu tabanlı XGBoost ve AdaBoost modelleri %100 ACC ve %0 FPR ile başarım sergilemiştir. Bu nedenle, genel olarak söz konusu modeller, ACC ve FPR metrikleri açısından en yüksek başarıyı göstermiştir.
- Bireysel modeller açısından, veri kümelerinden bağımsız olarak KNN modelleri %0 FPR'ye ulaşmıştır. Ölçünlü BoW veri kümesi dışındaki diğer kümeler üzerinde geliştirilen DT ve RF modelleri de benzer şekilde %0 FPR değeri elde etmiştir. Ayrıca, ikili ve olasılıksal BoW veri kümeleri üzerinde eğitilen DT modelleri %100 ACC oranı göstermiştir. Olasılıksal BoW veri kümelerindeki KNN ve RF modelleri de %100 ACC değerine ulaşmıştır.
- Eğitim sürelerine göz atıldığında, KNN modelleri tüm veri kümelerinde en yüksek verimliliği göstermiştir. Test süresi bakımından ise, her bir veri kümesi üzerinde DT modelleri en düşük süreleri elde etmiştir.

Genel olarak değerlendirildiğinde, yığınlama topluluğu tabanlı XGBoost ve AdaBoost modelleri ACC ve FPR açısından üstünlük sağlarken, KNN ve DT modelleri zaman etkinliği bakımından en verimli çözümleri sunmaktadır. Özellikle, ikili ve olasılıksal BoW veri kümeleri üzerinde geliştirilen DT modelleri, ACC, FPR ve test süresi açısından en uygun modeller olarak öne çıkmaktadır.

4.2.2. MIT Tabanlı BoW Veri Kümeleri Özelinde Değerlendirme

MIT tabanlı BoW veri kümeleri kullanılarak geliştirilen modellerin test sonuçları, Çizelge 9'da yer almaktadır. Çizelge 9 analiz edildiğinde elde edilen bulgular, aşağıda maddeler halinde verilmektedir.

- ACC ve FPR metrikleri açısından en üstün başarım, yığınlama topluluğu tabanlı XGBoost ve AdaBoost modelleri tarafından sergilenmiştir. Bu modeller %100 ACC ve %0 FPR değerine ulaşmıştır.
- Bireysel modeller değerlendirildiğinde, veri kümelerinin tamamında RF modelleri, %0 FPR ile başarım göstermiştir. Olasılıksal BoW veri kümesi dışındaki diğer veri kümelerinde LR modelleri de %0 FPR değerine ulaşmıştır. Benzer şekilde, TF-IDF BoW veri kümesi dışındaki KNN modelleri de %0 FPR oranı sergilemiştir.
- Eğitim süreleri açısından, KNN modelleri tüm veri kümelerinde en yüksek verimliliği göstermiştir. Test süreleri bakımından ise, DT ve LR modelleri veri kümesinden bağımsız olarak en düşük süreleri elde etmiştir.

Sonuç olarak, MIT tabanlı BoW veri kümelerinin tamamında yığınlama topluluğu tabanlı XGBoost ve AdaBoost modelleri ACC ve FPR açısından üstünlük sağlamaktadır. DT ve LR modelleri ise yüksek ACC ve düşük FPR değerlerinin yanı sıra test süreleri bakımından da dengeli bir başarım sunmaktadır.

4.2.3. UNM-MIT Tabanlı BoW Veri Kümeleri Özelinde Değerlendirme

UNM-MIT tabanlı BoW veri kümeleri ile geliştirilen modellerin deney sonuçları, Çizelge 10'da görülmektedir. Çizelge 10 incelendiğinde, aşağıda maddeler halinde belirtilen sonuçlara ulaşılmaktadır.

- DT modelleri ile yığınlama topluluğu tabanlı XGBoost ve AdaBoost modelleri, ACC ve FPR metrikleri bakımından tüm veri kümelerinde en yüksek başarımı sergilemiştir. Bu modeller, %100 ACC ve %0 FPR elde ederek üstün bir başarı göstermiştir.
- Bireysel modeller açısından, tüm veri kümelerinde RF modelleri %0 FPR değerine ulaşmıştır. KNN ve LR modelleri de, ölçünlü BoW veri kümesi dışındaki veri kümelerinde %0 FPR oranı elde etmiştir. Ayrıca, ikili ve TF-IDF tabanlı BoW veri kümeleri üzerinde eğitilen RF modelleri, %100 ACC oranına ulaşarak yüksek bir başarım sergilemiştir.
- Eğitim sürelerine bakıldığında, KNN modelleri en kısa sürede eğitilen modeller olmuştur. Öte yandan, DT ve LR modelleri test süresi açısından en düşük değerlere sahip olup, zaman etkinliği bakımından verimli sonuçlar elde etmiştir.

Netice olarak, UNM-MIT tabanlı BoW veri kümelerinde tüm modeller genel olarak yüksek başarım sergilemektedir. DT, XGBoost ve AdaBoost modelleri, tüm veri kümelerinde %100 ACC ve %0 FPR elde ederek en başarılı modeller olarak öne çıkmaktadır. Bununla birlikte, DT modelleri, yüksek ACC ve

düşük FPR değerlerinin yanı sıra kısa test süreleriyle zaman etkinliği açısından pratik ve dengeli çözümler sunmaktadır.

Çizelge-8: UNM Tabanlı Bow Veri Kümeleri Üzerinde Geliştirilen Modellerin Başarım Değerleri

UNM Tabanlı Ölçünlü BoW Veri Kümesi							
Model	**PRC**	**REC**	**F1**	**FPR**	**ACC**	**Eğitim Süresi (s)**	**Test Süresi (s)**
KNN	%100	%99,84	%99,92	%0	%99,92	0,0006	0,0056
DT	%99,85	%99,84	%99,84	%0,16	%99,84	0,0019	0,0002
LR	%100	%99,84	%99,92	%0	%99,92	0,0060	0,0003
RF	%99,85	%99,84	%99,84	%0,16	%99,84	0,1100	0,0058
XGBoost	%100	%100	%100	%0	%100	0,1278	0,0129
AdaBoost	%100	%100	%100	%0	%100	0,1459	0,0157
UNM Tabanlı İkili BoW Veri Kümesi							
Model	**PRC**	**REC**	**F1**	**FPR**	**ACC**	**Eğitim Süresi (s)**	**Test Süresi (s)**
KNN	%100	%99,84	%99,92	%0	%99,92	0,0010	0,0085
DT	%100	%100	%100	%0	%100	0,0025	0,0003
LR	%99,84	%99,84	%99,84	%0,16	%99,84	0,0090	0,0004
RF	%100	%99,84	%99,92	%0	%99,92	0,1800	0,0097
XGBoost	%100	%100	%100	%0	%100	0,2093	0,0207
AdaBoost	%100	%100	%100	%0	%100	0,1959	0,0195
UNM Tabanlı Olasılıksal BoW Veri Kümesi							
Model	**PRC**	**REC**	**F1**	**FPR**	**ACC**	**Eğitim Süresi (s)**	**Test Süresi (s)**
KNN	%100	%100	%100	%0	%100	0,0048	0,0048
DT	%100	%100	%100	%0	%100	0,0740	0,0008
LR	%9,962	%100	%99,81	%0,41	%99,80	0,0781	0,0010
RF	%100	%100	%100	%0	%100	0,7262	0,0334
XGBoost	%100	%100	%100	%0	%100	1,1514	0,1020
AdaBoost	%100	%100	%100	%0	%100	0,8909	0,0985
UNM Tabanlı TF-IDF BoW Veri Kümesi							
Model	**PRC**	**REC**	**F1**	**FPR**	**ACC**	**Eğitim Süresi (s)**	**Test Süresi (s)**
KNN	%100	%99,83	%99,92	%0	%99,92	0,0028	0,0245
DT	%100	%99,83	%99,92	%0	%99,92	0,0413	0,0005
LR	%99,92	%99,83	%99,88	%0,08	%99,88	0,0378	0,0005
RF	%100	%99,83	%99,92	%0	%99,92	0,3372	0,0147
XGBoost	%100	%100	%100	%0	%100	0,4897	0,0422
AdaBoost	%100	%100	%100	%0	%100	0,4566	0,0442

Çizelge-9: MIT Tabanlı Bow Veri Kümeleri Üzerinde Geliştirilen Modellerin Başarım Değerleri

MIT Tabanlı Ölçünlü BoW Veri Kümesi							
Model	**PRC**	**REC**	**F1**	**FPR**	**ACC**	**Eğitim Süresi (s)**	**Test Süresi (s)**
KNN	%100	%99,84	%99,92	%0	%99,92	0,0013	0,0122
DT	%99,92	%99,68	%99,80	%0,08	%99,80	0,0041	0,0005
LR	%100	%99,84	%99,92	%0	%99,92	0,0115	0,0005
RF	%100	%99,68	%99,84	%0	%99,84	0,2442	0,0138
XGBoost	%100	%100	%100	%0	%100	0,3683	0,0290
AdaBoost	%100	%100	%100	%0	%100	0,3118	0,0320
MIT Tabanlı İkili BoW Veri Kümesi							
Model	**PRC**	**REC**	**F1**	**FPR**	**ACC**	**Eğitim Süresi (s)**	**Test Süresi (s)**
KNN	%100	%99,93	%99,96	%0	%99,96	0,0013	0,0090
DT	%99,85	%99,93	%99,89	%0,17	%99,88	0,0025	0,0004
LR	%100	%99,93	%99,96	%0	%99,96	0,0092	0,0005
RF	%100	%99,93	%99,96	%0	%99,96	0,1801	0,0102
XGBoost	%100	%100	%100	%0	%100	0,2236	0,0227

AdaBoost	%100	%100	%100	%0	%100	0,2177	0,0243
MIT Tabanlı Olasılıksal BoW Veri Kümesi							
Model	**PRC**	**REC**	**F1**	**FPR**	**ACC**	**Eğitim Süresi (s)**	**Test Süresi (s)**
KNN	%100	%99,84	%99,92	%0	%99,92	0,0010	0,0109
DT	%100	%99,76	%99,88	%0	%99,88	0,0244	0,0005
LR	%99,68	%99,84	%99,76	%0,32	%99,76	0,0222	0,0005
RF	%100	%99,76	%99,88	%0	%99,88	0,2774	0,0109
XGBoost	%100	%100	%100	%0	%100	0,3384	0,0242
AdaBoost	%100	%100	%100	%0	%100	0,3633	0,0273
MIT Tabanlı TF-IDF BoW Veri Kümesi							
Model	**PRC**	**REC**	**F1**	**FPR**	**ACC**	**Eğitim Süresi (s)**	**Test Süresi (s)**
KNN	%99,75	%99,84	%99,79	%0,22	%99,80	0,0011	0,0128
DT	%99,83	%99,91	%99,87	%0,15	%99,88	0,0316	0,0006
LR	%100	%99,84	%99,92	%0	%99,92	0,0231	0,0005
RF	%100	%99,68	%99,84	%0	%99,84	0,3164	0,0105
XGBoost	%100	%100	%100	%0	%100	0,3980	0,0257
AdaBoost	%100	%100	%100	%0	%100	0,3760	0,0248

Çizelge-10: UNM-MIT Tabanlı Bow Veri Kümeleri Üzerinde Geliştirilen Modellerin Başarım Değerleri

UNM-MIT Tabanlı Ölçünlü BoW Veri Kümesi							
Model	**PRC**	**REC**	**F1**	**FPR**	**ACC**	**Eğitim Süresi (s)**	**Test Süresi (s)**
KNN	%99,88	%99,80	%99,84	%0,12	%99,84	0,0025	0,0525
DT	%100	%100	%100	%0	%100	0,0073	0,0006
LR	%99,88	%99,80	%99,84	%0,12	%99,84	0,0233	0,0006
RF	%100	%99,84	%99,92	%0	%99,92	0,3571	0,0177
XGBoost	%100	%100	%100	%0	%100	0,4415	0,0745
AdaBoost	%100	%100	%100	%0	%100	0,3943	0,0723
UNM-MIT Tabanlı İkili BoW Veri Kümesi							
Model	**PRC**	**REC**	**F1**	**FPR**	**ACC**	**Eğitim Süresi (s)**	**Test Süresi (s)**
KNN	%100	%99,84	%99,92	%0	%99,92	0,0016	0,0268
DT	%100	%100	%100	%0	%100	0,0031	0,0004
LR	%100	%99,84	%99,92	%0	%99,92	0,0160	0,0005
RF	%100	%100	%100	%0	%100	0,2098	0,0101
XGBoost	%100	%99,84	%99,92	%0	%99,92	0,2456	0,0394
AdaBoost	%100	%100	%100	%0	%100	0,2341	0,0384
UNM-MIT Tabanlı Olasılıksal BoW Veri Kümesi							
Model	**PRC**	**REC**	**F1**	**FPR**	**ACC**	**Eğitim Süresi (s)**	**Test Süresi (s)**
KNN	%100	%100	%99,96	%0	%99,96	0,0024	0,0659
DT	%100	%100	%100	%0	%100	0,0549	0,0011
LR	%100	%99,96	%99,98	%0	%99,98	0,0618	0,0008
RF	%100	%99,92	%99,96	%0	%99,96	0,5872	0,0145
XGBoost	%100	%100	%100	%0	%100	0,8412	0,0866
AdaBoost	%100	%100	%100	%0	%100	0,7122	0,0838
UNM-MIT Tabanlı TF-IDF BoW Veri Kümesi							
Model	**PRC**	**REC**	**F1**	**FPR**	**ACC**	**Eğitim Süresi (s)**	**Test Süresi (s)**
KNN	%100	%99,96	%99,98	%0	%99,98	0,0013	0,0430
DT	%100	%100	%100	%0	%100	0,0612	0,0006
LR	%100	%99,88	%99,94	%0	%99,94	0,0516	0,0006
RF	%100	%100	%100	%0	%100	0,5526	0,0100
XGBoost	%100	%100	%100	%0	%100	0,6904	0,0557
AdaBoost	%100	%100	%100	%0	%100	0,6699	0,0547

4.3 Kaynaklardaki Yöntemlerle Karşılaştırma

Çizelge-11'de, UNM live LPR veri kümesinden türetilen BoW veri kümeleri üzerinde geliştirilen en başarılı modeller ile kaynaklarda önerilen yöntemlerin başarımları karşılaştırılmaktadır. [7]'de önerilen FSVM tabanlı yöntem, %6,16 oranında bir yanlış alarm üretmiştir. [13]'te sunulan Markov zinciri tabanlı yöntem ise %99,93 ACC ile oldukça başarılı sonuç elde etmiştir. Benzer şekilde, [17]'de geliştirilen merkezi tabanlı vektör uzay modeline dayalı yöntem, %0,60 FPR ve %99,66 ACC ile dikkat çekmektedir. Bununla birlikte, bu çalışmada UNM tabanlı BoW veri kümeleri kullanılarak geliştirilen yığınlama topluluğu tabanlı modellerin tamamı, %0 FPR ve %100 ACC oranları ile başarım göstermiştir. Bununla birlikte, UNM tabanlı ikili ve olasılıksal BoW veri kümelerindeki DT modelleri ve UNM tabanlı olasılıksal BoW veri kümesindeki KNN ve RF modelleri de %0 FPR ve %100 ACC elde etmiştir. Bu modellerin %0 FPR ve %100 göstermesi, veri kümelerinin sınıflar arasındaki farkları net biçimde yansıtması ve kullanılan n-gram, BoW ve özellik seçim yöntemlerinin güçlü ayrıştırıcı özellikler sunmasından kaynaklanmaktadır. Sonuç olarak bu modeller, kaynaklardaki bahsedilen yöntemlerden üstün bir başarım sergilemiştir. Bu sonuç, geliştirilen modellerin doğruluk açısından güçlü bir başarım sergilediğini göstermektedir. Bununla birlikte, yanlış pozitiflerin tamamen ortadan kaldırılması, IDS sistemlerinde sıkça karşılaşılan yanlış alarm sorununa etkili bir çözüm sunmaktadır.

Çizelge-11: Kaynaklardaki Modellerle veya Yöntemlerle Karşılaştırma

Model/Yöntem	FPR	ACC
[7]'de önerilen FSVM yöntemi	%6,16	-
[13]'te önerilen Markov zinciri tabanlı anomali tespit yöntemi	-	%99,93
[17]'de önerilen merkezi tabanlı vektör uzay modeline dayalı bir anomali tespit yöntemi	%0,60	%99,66
Bu çalışmada geliştirilen modeller	%0	%100

4.4 Tartışma

UNM tabanlı, MIT tabanlı ve UNM-MIT tabanlı BoW veri kümeleri üzerinde geliştirilen modellerin tamamı, tüm metrikler açısından yüksek başarı göstermiştir. Dolayısıyla, bu veri kümeleri üzerinde n-gram, ölçünlü BoW, ikili BoW, olasılıksal BoW, TF-IDF BoW ve özellik seçim yöntemlerinin, modellerin başarımını anlamlı ölçüde artırdığı sonucuna varılmaktadır. Bununla birlikte, söz konusu ön işlem yöntemlerinin sistem çağrı dizileri gibi sıralı veriler üzerinde uygulanmasının, ML modellerinin başarımını artırabileceği sonucuna ulaşılmaktadır. Diğer yandan, eğitim ve test sürelerinin görece kısa olması, söz konusu modellerin veri kümelerinin üretildiği işletim sistemi ve program bağlamında gerçek zamanlı kullanım için uygunluk potansiyeli taşıdığını göstermektedir.

Öte yandan, bazı modellerde gözlemlenen %0 FPR ve %100 ACC değerleri dikkat çekmektedir. Bu değerler, kullanılan veri kümelerinin belirli işletim sistemleri ve programlara özgü homojen bir yapıya sahip olmalarından dolayı ortaya çıkmıştır. Bununla birlikte, elde edilen tüm sonuçlar, beş yinelemenin ortalaması alınarak hesaplanmıştır. Böylece, rastgelelikten kaynaklanan sapmalar minimize edilmiştir.

Geliştirilen modellerin bazı sınırlılıkları da bulunmaktadır. Elde edilen yüksek başarım sonuçları, kullanılan veri kümelerinin belirli işletim sistemleri ve programlara özgü olması nedeniyle genellenebilirlik açısından sınırlı kalabilmektedir. Özellikle, kullanılan veri kümeleri, günümüzdeki gelişmiş ve karmaşık saldırı türlerini tam olarak temsil etmemektedir. Bu durum, elde edilen sonuçların, modern ağ ve sistem ortamlarında aynı derecede geçerli olmayabileceğini ortaya koymaktadır. Bu nedenle, modellerin genellenebilirliğini daha kapsamlı biçimde değerlendirmek için, daha modern ve çeşitli veri kümeleri üzerinde testlerin gerçekleştirilmesi gerekmektedir. Başka bir ifadeyle, farklı ortamlar veya daha karmaşık programlar üzerinde benzer başarıların elde edilip edilemeyeceği, gerçek dünya senaryoları üzerinden titizlikle incelenmesi gerekmektedir. Ayrıca, eğitim ve test sürelerinin görece kısa olması gerçek zamanlı kullanım için bir avantaj sağlasa da daha büyük ve çeşitlendirilmiş veri kümeleri üzerinde bu sürelerin artabileceği göz önünde bulundurulmalıdır. Öte yandan, n-gram ve BoW yöntemlerinin, sıralı verilerdeki uzun bağımlılık ilişkilerini tam olarak yansıtamama ihtimali de, olası bir sınırlılık olarak değerlendirilmektedir.

5. Sonuç ve Gelecek Çalışmalar

Bu çalışmada, UNM live LPR, MIT live LPR ve her iki veri kümesinin birleşiminden oluşturulan UNM-MIT live LPR veri kümeleri kullanılarak çeşitli modeller/HIDS'ler geliştirilmiştir. İlk aşamada, söz konusu veri kümeleri üzerinde n-gram, ölçünlü BoW, ikili BoW, olasılıksal BoW ve TF-IDF BoW yöntemleri uygulanmış ve her birine özgü BoW tabanlı veri kümeleri elde edilmiştir. Daha sonraki aşamada, bu veri kümeleri üzerinde özellik seçim süreci gerçekleştirilmiş, seçilen özellikleri içeren BoW veri kümeleri kullanılarak hem bireysel hem de yığınlama topluluğu tabanlı modeller/HIDS'ler geliştirilmiştir. Deneysel sonuçlar, geliştirilen modellerin oldukça düşük FPR ve yüksek ACC değerleri ile başarım gösterdiğini ortaya koymuştur. Ayrıca, elde edilen bulgular, UNM tabanlı BoW veri kümeleri ile geliştirilen en başarılı modellerin, kaynaklarda önerilen yöntemlere kıyasla ilgili metrikler açısından daha üstün başarım sergilediğini göstermektedir.

Öte yandan, pratik kullanımlar, gerçek zamanlı senaryolar ve kaynak kısıtlı sistemler açısından geliştirilen modeller değerlendirildiğinde, deney sonuçları bu tür durumlarda modellerin uygulanabilir nitelikte olduklarını göstermektedir. Ancak, kullanılan veri kümelerinin görece eski olması, modellerin güncel saldırı türlerine karşı ve dinamik ortamlarda doğrudan uygulanabilirliğini sınırlamaktadır. Buna bağlı olarak, güncel veri kümeleri üzerinde gerçekleştirilecek deneyler aracılığıyla, modellerin gerçek zamanlı ya da kaynak kısıtlı sistemlerde kullanımlarına ilişkin daha kapsamlı sonuçların elde edilmesi beklenmektedir.

Gelecek çalışmalarda, farklı ve güncel veri kümeleri üzerinde aynı ön işlem yöntemleri uygulanarak çeşitli ML tabanlı modellerin/HIDS'lerin geliştirilmesi planlanmaktadır. Bu sayede, daha geniş kapsamlı kullanım alanlarına sahip modellerin/HIDS'lerin elde edilmesi amaçlanmaktadır. Ayrıca, mevcut çalışmada kullanılan veri kümeleri ve güncel veri kümeleri kullanılarak, LSTM, GRU ve transformer gibi DL tabanlı modellerin/HIDS'lerin geliştirilmesi ve bunların kapsamlı bir şekilde değerlendirilmesi hedeflenmektedir.

Kaynakça

[1] Liao, H.-J. Lin, C.-H. R. Y.-C. Lin, ve Tung, K.-Y. 'Intrusion detection system: A comprehensive review', *J. Netw. Comput. Appl.*, vol. 36, no. 1, pp. 16–24, 2013.

[2] Martins, I. Resende, J. S. Sousa, P. R. Silva, S. Antunes, L. ve Gama, J. 'Host-based IDS: A review ve open issues of an anomaly detection system in IoT', *Futur. Gener. Comput. Syst.*, vol. 133, pp. 95–113, 2022.

[3] Thakkar A. ve Lohiya, R. 'A survey on intrusion detection system: feature selection, model, performance measures, application perspective, challenges, ve future research directions', *Artif. Intell. Rev.*, vol. 55, no. 1, pp. 453–563, 2022.

[4] Satilmiş, H. Akleylek, S. ve Tok, Z. Y. 'A systematic literature review on host-based intrusion detection systems', *Ieee Access*, vol. 12, pp. 27237–27266, 2024.

[5] Jose, S. Malathi, D. Reddy, B. ve Jayaseeli, D. 'A survey on anomaly based host intrusion detection system', in *Journal of Physics: Conference Series*, 2018, vol. 1000, p. 12049.

[6] Shin Y. ve Kim, K. 'Comparison of anomaly detection accuracy of host-based intrusion detection systems based on different machine learning algorithms', *Int. J. Adv. Comput. Sci. Appl.*, vol. 11, no. 2, 2020.

[7] Liu, W. Ci, L. L. ve Liu, L. P. 'A new method of fuzzy support vector machine algorithm for intrusion detection', *Appl. Sci.*, vol. 10, no. 3, 2020, doi: 10.3390/app10031065.

[8] Mishra, P. Verma, I. ve Gupta, S. 'KVMInspector: KVM Based introspection approach to detect malware in cloud environment', *J. Inf. Secur. Appl.*, vol. 51, p. 102460, 2020, doi: 10.1016/j.jisa.2020.102460.

[9] Subba B. ve Gupta, P. 'A tfidfvectorizer ve singular value decomposition based host intrusion detection system framework for detecting anomalous system processes', *Comput. & Secur.*, vol. 100, p. 102084, 2021.

[10] Melvin, A. A. R. Kathrine, G. J. W. Pasupathi, S. Shanmuganathan, V. ve Naganathan, R. 'An AI powered system call analysis with bag of word approaches for the detection of intrusions ve malware in Australian Defence Force Academy ve virtual machine monitor malware attack data set', *Expert Syst.*, p. e13029, 2022.

[11] Melvin A. A. R. ve ark., 'Dynamic malware attack dataset leveraging virtual machine monitor audit data for the detection of intrusions in cloud', *Trans. Emerg. Telecommun. Technol.*, vol. 33, no. 4, pp. 1–19, 2022, doi: 10.1002/ett.4287.

[12] Mishra, P. Gupta, A. Aggarwal, P. ve Pilli, E. S. 'vServiceInspector: Introspection-assisted evolutionary bag-of-ngram approach to detect malware in cloud servers', *Ad Hoc Networks*, vol. 131, no. April 2021, p. 102836, 2022, doi: 10.1016/j.adhoc.2022.102836.

[13] Shamim, N. Asim, M. Baker, T. ve Awad, A. I. 'Efficient Approach for Anomaly Detection in IoT Using System Calls', *Sensors*, vol. 23, no. 2, pp. 1–24, 2023, doi: 10.3390/s23020652.

[14] Chaudhari, A. Gohil, B. ve Rao, U. P. 'A novel hybrid framework for cloud intrusion detection system using system call sequence analysis', *Cluster Comput.*, vol. 27, no. 3, pp. 3753–3769, 2024.

[15] Vyšniunas, T. Čeponis, D. Goranin, N. ve Čenys, A. 'Risk-based system-call sequence grouping method for malware intrusion detection', *Electronics*, vol. 13, no. 1, p. 206, 2024.

[16] Kim, Y. Hong, S.-Y. Park, S. ve Kim, H. K. 'Reinforcement Learning-based Generative Security Framework for Host Intrusion Detection', *IEEE Access*, 2025.

[17] Shamim, N. Asim, M. Ismail Awad, A. ve Khurram Khan, M. 'Anomaly Detection in Internet of Things System Calls Using a Centroid-Based Vector-Space Model', *IEEE Internet Things J.*, vol. 12, no. 14, pp. 26868–26881, 2025, doi: 10.1109/JIOT.2025.3562675.

[18] Satilmiş, H. Akleylek, S. ve Yüce Tok, Z. 'Development of Various Stacking Ensemble-Based HIDS Using ADFA Datasets', *IEEE Open J. Commun. Soc.*, vol. 6, pp. 1170–1189, 2025, doi: 10.1109/OJCOMS.2025.3538101.

[19] Hofmeyr, S. A. Forrest, S. ve Somayaji, A. 'Intrusion detection using sequences of system calls', *J. Comput. Secur.*, vol. 6, no. 3, pp. 151–180, 1998, doi: 10.3233/JCS-980109.