Fırat Üniversitesi Müh. Bil. Bil. Dergisi
34(1), 341-349, 2022

Araştırma Makalesi
https://doi.org/10.35234/fumbd.1026610

Arşimet Optimizasyon Algoritması ile Trafo Tabanlı Evrişimsel Sinir Ağı Modelini Kullanarak Yazılım Tanımlı Ağ Teknolojisi Verilerinde Saldırı Tespiti

Mesut TOĞAÇAR[1*]
[*1] Bilgisayar Teknolojileri Bölümü, Teknik Bilimler Meslek Yüksekokulu, Fırat Üniversitesi, Elazığ, Türkiye
[*1] mtogacar@firat.edu.tr

(Geliş/Received: 21/11/2021; Kabul/Accepted: 12/01/2022)

Öz: Son zamanlarda insanların teknoloji cihazları kullanarak günlük işlerini idame etmesindeki oran artmıştır. Akıllı cihazların birbirleriyle iletişim sağlayabildiği şu zamanda nesnelerin interneti kavramı ortaya çıkmıştır. Bütün bu gelişmeler insan hayatını daha da kolaylaştırırken diğer taraftan verilerin iletimini güvenli bir şekilde aktarılmasını sağlayabilen sistemlerin tasarlanmasını zorunlu hale getirmiştir. Bu çalışmada yazılım tanımlı ağ verilerinde saldırı tespitini gerçekleştirebilen yapay zekâ tabanlı hibrit bir yaklaşım geliştirilmiştir. Veri kümesi normal, dağıtılmış hizmet reddi, kaba kuvvet saldırıları, siteler arası betik çalıştırma ve SQL enjeksiyon ağ saldırı türlerini içermektedir. Önerilen yaklaşımda ön işlem adımı olarak Arşimet optimizasyon algoritması kullanılmıştır. Arşimet optimizasyon algoritması sayesinde veri kümesindeki verimli özelliklerin seçimi gerçekleştirilmiştir. Ardından trafo tabanlı evrişimsel sinir ağı modeli kullanılarak veri kümesi eğitilmiştir. Ağ trafiğinin normal veya saldırı tespitinde softmax yöntemi sınıflandırıcı olarak kullanılmıştır. Bu çalışmanın deneysel analizinde %98,94 genel doğruluk başarısı elde edilmiştir.

Anahtar kelimeler: Derin Öğrenme, Karar Destek Sistemi, Nesnelerin İnterneti, Veri Güvenliği, Yazılım Tabanlı Ağ.

Attack Detection in Software-Defined Network Technology Data Using A Transformer-Based Convolutional Neural Network Model with An Archimedean Optimization Algorithm

Abstract: Recently, there has been an increase in the number of people who do their daily work with the help of technological devices. During this time, the concept of the Internet of Things has emerged, where smart devices can communicate with each other. While all these developments make people's lives easier, on the other hand, they make it necessary to develop systems that can ensure secure transmission of data. In this study, a hybrid approach based on artificial intelligence was developed to detect attacks on software-defined network data. The dataset includes normal, denial of service, brute force, cross-site scripting and SQL injection network attacks. Archimedes optimization algorithm has been used as a preprocessing step in the proposed approach. Thanks to Archimedes optimization algorithm, the selection of efficient features in the dataset was done. Then, the dataset was trained using the transformer-based convolutional neural network model. The softmax method was used as a classifier for detecting normal or attack network traffic. The overall accuracy achieved in the experimental analysis of this study was 98.94%.

Key words: Deep Learning, Decision Support System, Internet of Things, Data Security, Software Defined Network.

1. Giriş

Nesnelerin interneti (IoT) ve yazılım tabanlı ağlar (SDN) kavramı son zamanlarda adından söz ettiren iki teknolojidir. IoT teknolojileri insanların geleneksel olarak gerçekleştirdiği iş veya eylemleri, dijital ortamda gerçekleştirilmesine olanak sağlar. Bunu gerçekleştirirken zamandan kazanç ve maliyetten tasarrufu ön planda tutar. IoT teknolojisindeki amaç akıllı cihazları internet üzerinden iletişimlerini sağlayabilmektir [1,2]. SDN, ağdaki verilerin daha hızlı bir şekilde aktarılmasını sağlar ve mevcut ağı merkezi bir sistem ile izlenmesini gerçekleştirir. Böylece ağ trafiğinde meydana gelen saldırıların tespiti daha kolay bir şekilde tespit edilir. Fakat SDN'ler ağda gerçekleşen dağıtılmış hizmet reddi saldırıları (DDoS) gibi türleri tespit etmede ise yetersizdir [3]. Bu tür durumlarda ağdaki veri güvenliğini koruyabilmek için birçok yöntem ve model geliştirilmiştir.

Milyarlarca akıllı cihazın internet ortamında kontrolü problem olabilmektedir. Ağ trafiğinin güvenliğini sağlayabilmek te bu problemlerden birisidir [4]. Karmaşık ağ yapılarında milyarlarca verinin eş zamanlı kontrolünü sağlayabilmek zor bir süreçtir. Son zamanlarda kötü amaçlı yazılımcılar saldırı yoğunluğunu IoT ve SDN teknolojilerini kullanan cihazlara yönlendirmiştir [5]. Bu teknolojileri kullanan cihazların heterojen bir

* Sorumlu yazar: mtogacar@firat.edu.tr. Yazarların ORCID Numarası: [1] 0000-0002-8264-3899

yapıya sahip olması, kullanıcılar tarafından ilgi görmesi, geleneksel ağ yapılarının sistem tarafından kullanılması gibi nedenler kötü amaçlı yazılımcıların yoğunlaşmasındaki faktörler arasındadır [6]. Bütün bunlara rağmen ağ saldırılarının güvenliğini sağlayabilecek teknolojiler de gelişmektedir. Son zamanlarda yapay zekâ tabanlı birçok çalışma gerçekleşmiştir. Bu çalışmalardan bazıları incelenirse;

Nugraha ve ark. [7] SDN verilerinde oluşabilecek ağ saldırılarının tespiti için evrişimsel sinir ağı (ESA) ve uzun kısa süreli bellek (LSTM) modellerini birleştiren bir model tasarladılar. Nugraha ve ark. [7] tasarladıkları model ile %99 genel doğruluk başarısı elde ettiler. Ahuja ve ark. [3] SDN ağlarında DDoS saldırılarının tespiti için hibrit bir makine öğrenme yöntemi tasarladılar. Önerdikleri yaklaşımda rastgele orman (RO) ile destek vektör makinelerini (DVM) birleştirdiler. Ahuja ve ark. [3] önerdikleri yaklaşım ile %98,8 genel doğruluk başarısı elde ettiler. Abdallah ve ark. [8] ağ saldırılarının tespiti için ESA-LSTM hibrit bir model tasarladılar. Ön işlem adımı olarak L_2 düzenlileştirme tekniğini kullandılar. Ön işlem adımı ile aşırı öğrenme probleminin oluşmasını engellediler. Analiz sonuçlarında elde ettikleri genel doğruluk başarısı %96,32'ydi. Polat ve ark. [9] SDN verilerinde ağ saldırılarının tespiti için makine öğrenme yöntemlerini kullandılar. SDN veri kümesini daha etkili kullanabilmek için özellik seçim yöntemi kullandılar. Ardından yapay sinir ağı (YSA), naif bayes (NB), DVM ve en yakın komşu (EYK) yöntemlerini kullanarak sınıflandırma işlemini gerçekleştirdiler. Sınıflandırma sürecinde en iyi performansı EYK yöntemi verdi ve elde ettikleri genel doğruluk başarısı %98,3'tü. Revathi ve ark. [10] SDN verilerinde gerçekleştirilen ağ saldırılarının tespiti için ayrık ölçeklenebilir bellek tabanlı DVM yöntemini kullandılar. DDoS saldırılarında elde ettiği genel doğruluk başarısı %99,7'ydi.

Bu çalışmada trafo tabanlı-ESA modeli mimarisini tasarlayarak SDN verilerindeki ağ saldırılarının tespitini başarılı bir şekilde gerçekleştirilmesi hedeflenmektedir. Ön işlem adımında arşimet optimizasyon algoritması (AOA) kullanılacaktır. AOA yöntemi ile SDN veri kümesinde verimli özelliklerin yer aldığı parametrelerin seçimi gerçekleştirilecektir. Bu çalışmanın diğer bölümleri hakkındaki bilgiler şu şekilde özetlenmiştir. SDN veri kümesi hakkında detaylı bilgiler Bölüm 2'de verilmiştir. Önerilen yaklaşımda kullanılan yöntemler ve model hakkındaki bilgiler Bölüm 3'te verilmiştir. Deneysel analizler ve bulgular Bölüm 4'te yer almıştır. Değerlendirme ve sonuç hakkındaki bilgiler Bölüm 5'te verilmiştir.

2. SDN Veri Kümesi

SDN veri kümesinde gerçekleştirilen ağ saldırıları trafik izleme kullanılarak tespit edilmesi mümkündür. Bu veri kümesi günlük olarak gerçekleştirilmiş gerçek zamanlı trafik kayıtlarını içermektedir. Veri kümesindeki kayıtlar paket yakalama arabirimlerinden elde edilmiştir ve ardından bir tablo formatına dönüştürülmüştür. Veri kümesinde 80 özellik yer almaktadır. Bu özelliklerin bir tanesi nitel özellik taşırken, diğer özellikler niceldir. Nicel özelliklerin 24 adeti "float/ondalık" veri tipi ve 55 adeti "int/tamsayı" veri tipidir. Veri kümesinin oluşturulma amacı, ağa gerçekleştirilen saldırıların tespitini gerçekleştirmek içindir. Ayrıca veri kayıtları çeşitli ağ türlerini içererek oluşturulmuştur ve erişime açıktır. Ağ trafiği verilerinde DDoS, sql enjeksiyon, kaba kuvvet, siteler arası betik saldırı türlerini ve normal trafik verilerini içermektedir. Veri kümesi türleri ve kayıt sayısı hakkında detaylı bilgiler Tablo 1'de verildi. Toplam 798.322 kayıt normal ağ trafiğini oluştururken, 390.011 kayıt ağ saldırısı trafiğini oluşturmaktadır [11].

Tablo 1. Veri kümesinin trafik türleri ve kayıt sayısı.

Ağ trafik türleri	**Kayıt sayısı**
Normal	798.322
DDoS	383.439
Sql enjeksiyon	60
Kaba kuvvet	4.550
Siteler arası betik	1.962
Toplam	1.188.333

3. Yöntem ve Önerilen Yaklaşım

3.1. Arşimet Optimizasyon Algoritması

AOA yöntemi gerçek hayatta karşılaşılan problemlerin üstesinden gelebilen popülasyon temelli bir optimizasyon algoritmasıdır. AOA yöntemi Arşimet prensibine dayanmaktadır ve bir cismin sıvı içerisine daldırılması ile oluşan kuvvetin gösterdiği eylemden yola çıkarak optimizasyon algoritması geliştirilmiştir. Algoritma için gerçekleştirilen işlem adımları şu şekildedir;

- Cisimlerin/Nesnelerin başlangıç konumu rastgele belirlenir.
- Her bir cismin iterasyon esnasında güncellenen hacim, yoğunluk ve ivmesi vardır. Yoğunluk ve hacim güncellemesi gerçekleşir.
- Nesnelerin hareketleri esnasında birbiriyle çarpışması sonucu oluşan ivme değerleri, her bir iterasyonda güncellenir.
- Nesnelerin hareketsiz kabul edildiği durumlarda da ivme değerleri her bir iterasyonda güncellenir.
- Hareketli ve hareketsiz kabul edilen durumlar için değişim yüzdesi işlemi gerçekleşir ve hızlanma normalleşir.
- Son adımda konum güncellemesi gerçekleştirilir [12].

AOA yönteminin sözde kodu Tablo 2'de belirtildiği gibidir.

Tablo 2. AOA yönteminin sözde kodu [13].

Sözde kod-AOA(popülasyon boyutu, maksimum iterasyon, sabitler, değişkenler**)**

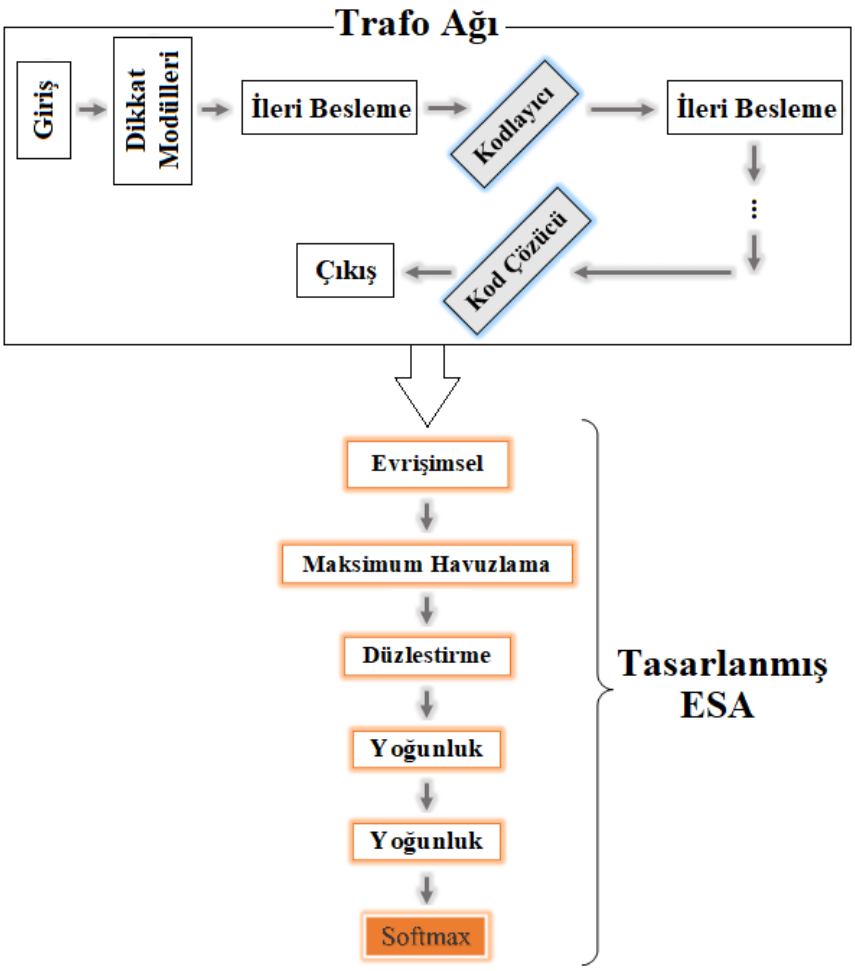
```
Başlangıç değerlerin rastgele seçimi (nesneler, hacim, yoğunluk, konum, uygunluk fonksiyonu)
iterasyon=1
  while (iterasyon <= maksimum iterasyon)
      for each nesne do
          güncelle (nesne_yoğunluk, nesne_hacim)
          hesapla (değişim_yüzdesi_faktörü)
          if (yüzde_faktörü<=0,5)
              normalizasyon (hız-ivme)
              güncelle (konum)
          else
              normalizasyon (hız-ivme)
              güncelle (hareket yönü)
              güncelle (konum)
          end if
      end for
      hesapla (en iyi uygunluk değeri)
      iterasyon=iterasyon+1
  end while
```

Bu çalışmada AOA yöntemi ön işlem adımı olarak kullanılmıştır ve SDN veri kümesinin tamsayı parametrelerini içeren değerleri işleyerek en iyi niteliklerin seçimi gerçekleştirilmiştir. Böylece daha az ama daha etkili nitelikler seçilerek önerilen yaklaşımın performansına katkı sunması amaçlanmıştır. AOA yönteminin analiz kodları Python dili kullanılarak gerçekleştirilmiştir [14]. AOA yöntemi için tercih edilen önemli parametre değerleri şunlardır; iterasyon sayısı 1000, popülasyon boyutu 100, minimum hızlandırma parametre değeri 0,1 ve maksimum hızlandırma parametre değeri 0,95 tercih edilmiştir.

3.2. Trafo Tabanlı Evrişimsel Sinir Ağı Modeli ve Önerilen Yaklaşım

Trafo ağ mimarisi dikkat mekanizmasına dayalı bir modeldir ve doğal dil işleme uygulamalarında tercih edilmektedir. Trafo ağ mimarisi iki kısımdan oluşmaktadır. Bunlar, kodlayıcı ve kod çözücüdür. Her bir kodlayıcı

blok iki katmandan oluşmaktadır. Bunlar, dikkat modülü katmanı ve ileri besleme katmanıdır. Kod çözücü blok ise üç katmandan oluşmaktadır ve bunlar; kodlayıcı dikkat modülü, kod çözücü dikkat modülü ve ileri besleme katmanıdır. Burada dikkat modülleri benzerlik farkını hesaplayarak daha belirgin özelliklerin seçimine yardımcı olmaktadır ve bir kodlayıcı bloğundan çıkan girdiler ileri besleme katmanı sayesinde kod çözücüsüne aktarılmaktadır [15]. Trafo ağ mimarilerinin performansını artırabilmek için ESA yapılarındaki katmanlar ile hibrit bir model oluşturmak mümkündür. Bu çalışmada Trafo ağ mimarilerinden elde edilmiş veriler, tasarlanmış ESA modeline girdi olarak verilmektedir. Tasarlanan ESA modeli sırasıyla evrişimsel katman [16], havuzlama katmanı [17], düzleştirme katmanı ve yoğunluk katmanlarından [18] oluşmaktadır. ESA modelinin sınıflandırma işlemini gerçekleştirmek için son katmanda Softmax yöntemi [19] tercih edildi. Trafo Tabanlı Evrişimsel Sinir Ağı Modeli (TT-ESA) modelinin genel tasarımı Şekil 1'de gösterilmiştir.



Şekil 1. TT-ESA modelinin genel tasarımı.

Önerilen yaklaşım ön işlem adımı (AOA yöntemi) ve TT-ESA modelinin birleştirilmesi ile tasarlanmıştır. AOA yöntemi ile verimli özelliklerin seçimi gerçekleştirilirken, trafo ağı mimarisi ile seçilen veriler dikkat modüllerinden geçirilmiştir. Son işlem adımında ESA modeli ile eğitim-test verileri eğitilerek sınıflandırma işlemi gerçekleştirilmiştir. Bu amaçla SDN verilerinde gerçekleştirilen web tabanlı saldırıların tespiti başarılı bir şekilde gerçekleşebilecektir. Önerilen yaklaşımın işlem adımlarını gösteren tasarım Şekil 2'de belirtilmiştir.

Şekil 2. Önerilen yaklaşım ve işlem adımları.

4. Bulgular

Çalışmanın deneysel analizleri Google Colab sunucuları kullanılarak gerçekleştirildi. Önerilen yaklaşımdaki tüm yöntem ve modeller (AOA, trafo ağı, ESA) Python dilinde tasarlandı. Kodların derlenmesinde Jupyter Notebook arayüz yazılımı kullanıldı. Analiz ölçümlerinde karmaşıklık matrisinin metrikleri kullanıldı ve tercih edilen metrikler ve hesaplama formülleri Denklem 1-4 arasında gösterilmiştir. Metrik değerlerinin hesaplanmasında kullanılan kısaltmalar; pozitif (P), negatif (N), doğru (D) ve yanlış (Y) ile temsil edilir [20,21].

$$\text{Duyarlılık} = \frac{DP}{DP+YN} \tag{1}$$

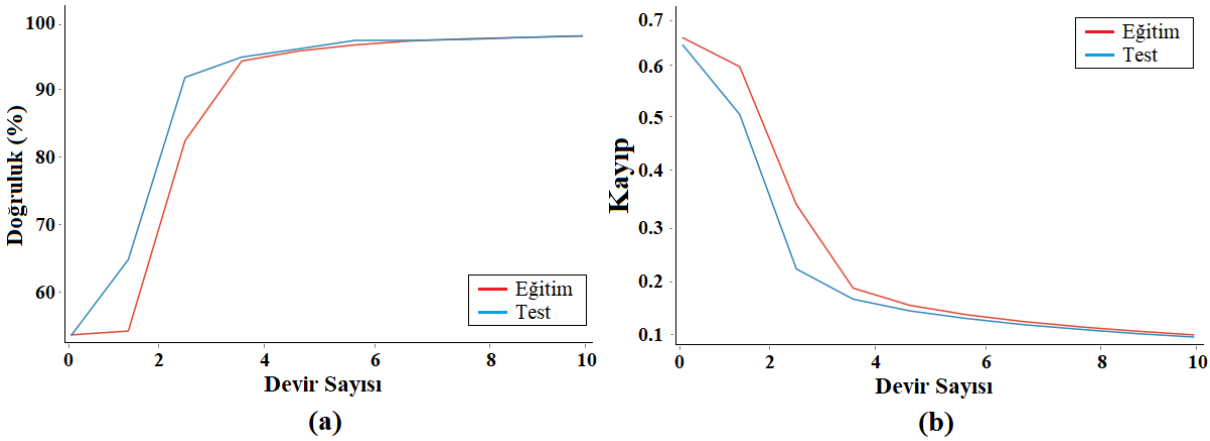
$$\text{Özgünlük} = \frac{DN}{DN+YP} \tag{2}$$

$$\text{F-skor} = \frac{2\times DP}{2\times DP+YP+YN} \tag{3}$$

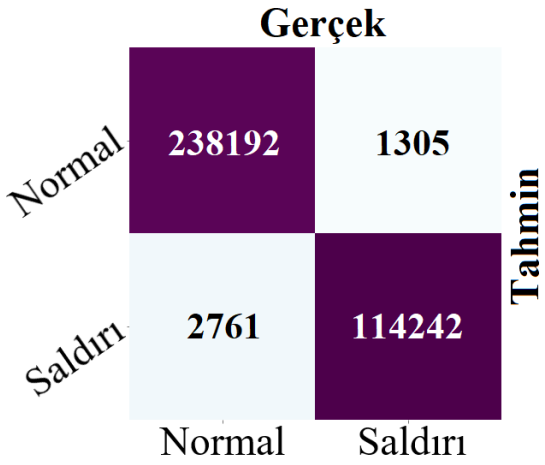
$$\text{Doğruluk} = \frac{DP+DN}{DP+DN+YP+YN} \tag{4}$$

Deney iki bölümden oluşmuştur ve tüm analizlerde veri kümesinin %70'i eğitim verisi olarak kullanılmıştır, %30'u test verisi olarak kullanılmıştır. Modelin eğitimi için tüm analizlerde devir sayısı 10 ve öğrenme oranı $1e-4$ tercih edilmiştir. Ayrıca derlenme esnasında aşırı öğrenmeyi engellemek için Keras kütüphanesinden erken durma (early stopping) parametresi [22] kullanılmıştır ve optimizasyon yöntemi için Adamax [23] tercih edilmiştir.

Deneyin birinci bölümünde veri kümesinin 55 özellik parametresi (integer değerler) kullanılarak analiz gerçekleştirilmiştir. Veri kümesinin 24 özellik parametresi (float değerler) önerilen yaklaşımın işlem süresini olumsuz etkileyeceğinden deney analizlerinde kullanılmamıştır. Birinci bölümde 55 özellik sütunu TT-ESA yaklaşım tarafından eğitilmiştir ve eğitim-test başarı grafikleri Şekil 3(a)'da, eğitim-test kayıp grafikleri Şekil 3(b)'de gösterilmiştir. Eğitim esnasında önerilen yaklaşımda AOA yöntemi kullanılmamıştır ve eğitim süresi yaklaşık 2097 saniye sürmüştür. Birinci bölümde elde edilen karmaşıklık matrisi Şekil 4'te gösterildi ve analiz sonuçları ise Tablo 3'te verilmiştir. TT-ESA modeli ile elde edilen genel doğruluk başarısı %98,86'ydı.



Şekil 3. TT-ESA modeli ile gerçekleştirilen analiz grafikleri;
a) eğitim-test doğruluk başarısı grafiği, b) eğitim-test kayıp grafiği.



Şekil 4. TT-ESA modeli ile gerçekleştirilen analizin karmaşıklık matrisi.

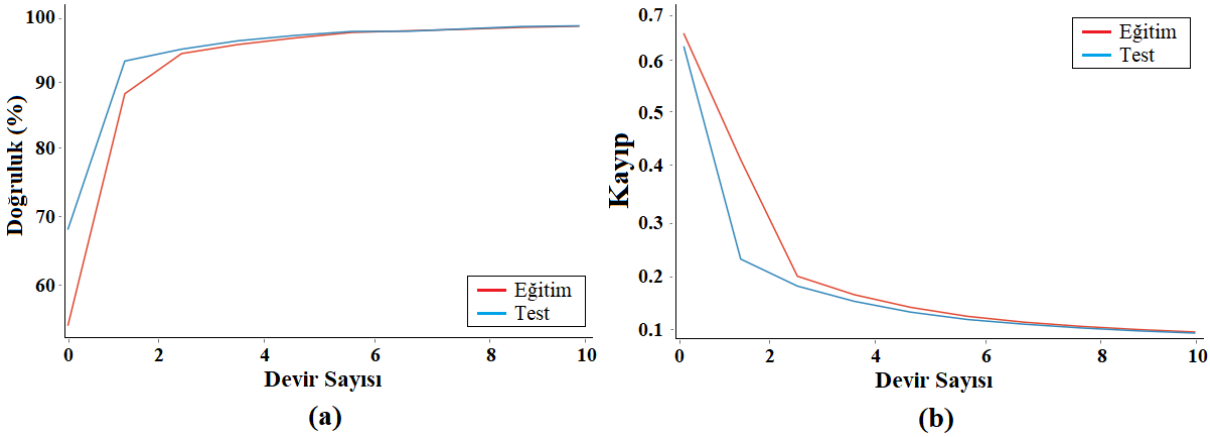
Tablo 3. TT-ESA modeli ile gerçekleştirilen analiz sonuçları (%).

Sınıf	Duyarlılık	Özgünlük	F-skor	Doğruluk
Normal & Saldırı	98,85	98,87	99,15	98,86

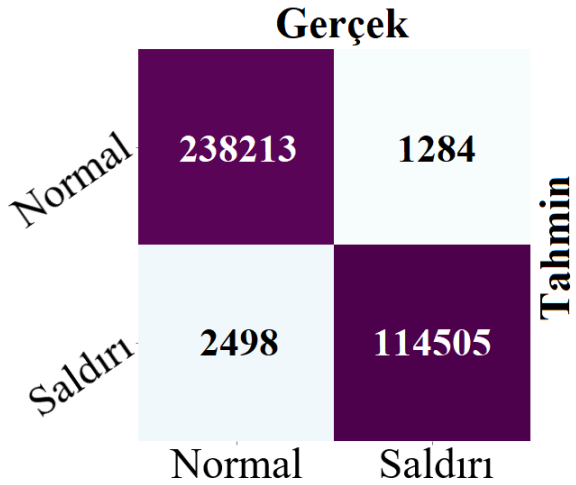
Deneyin ikinci bölümünde önerilen yaklaşım kullanılarak analizler gerçekleşti. Ön işlem adımı olarak verimli özellik seçimi gerçekleşti ve birinci bölümde kullanmış 55 özellik sütunu arasından en iyi 18 özellik sütunu AOA yöntemi kullanılarak belirlenmiştir. AOA yöntemi ile seçilmiş özellikler Tablo 4'te gösterilmiştir. Ardından veri kümesindeki 18 özellik sütunu ile TT-ESA modeli eğitilmiştir. Eğitim süresi yaklaşık 834 saniye sürmüştür. İkinci bölümün eğitim-test doğruluk başarı grafikleri Şekil 5(a)'da ve eğitim-test kayıp grafikleri Şekil 5(b)'de gösterilmiştir. Bu bölümden elde edilmiş karmaşıklık matrisi Şekil 6'da gösterilmiştir ve analiz sonuçları Tablo 5'te verilmiştir. Önerilen yaklaşım ile gerçekleştirilen analiz sonucunda %98,94 genel doğruluk başarısı elde edilmiştir. İkinci bölümde AOA yönteminin katkı sağladığı gözlemlenmiştir. Sonuç olarak birinci bölümdeki analiz sonuçlarına göre eğitim süresindeki tasarruf ve performans başarısındaki artış ikinci bölüm analizlerinde gerçekleşmiştir.

Tablo 4. AOA yöntemi kullanılarak seçilmiş verimli özellikler.

No	Özellik	No	Özellik
1	Destination Port	10	Idle Max
2	Fwd PSH Flags	11	Idle Min
3	Bwd PSH Flags	12	act_data_pkt_fwd
4	FIN Flag Count	13	min_seg_size_forward
5	RST Flag Count	14	Flow Duration
6	ECE Flag Count	15	Fwd IAT Max
7	Down/Up Ratio	16	Fwd IAT Min
8	Init_Win_bytes_forward	17	SYN Flag Count
9	Init_Win_bytes_backward	18	Fwd Avg Bulk Rate



Şekil 5. Önerilen yaklaşım ile gerçekleştirilen analiz grafikleri;
a) eğitim-test doğruluk başarısı grafiği, b) eğitim-test kayıp grafiği.



Şekil 6. Önerilen yaklaşım ile gerçekleştirilen analizin karmaşıklık matrisi.

Tablo 5. Önerilen yaklaşım ile gerçekleştirilen analiz sonuçları (%).

Sınıf	Duyarlılık	Özgünlük	F-skor	Doğruluk
Normal & Saldırı	98,96	98,89	99,21	98,94

5. Tartışma ve Sonuç

Bu çalışmada SDN paradigmasına dayalı verileri kullanarak yapay zekâ tabanlı bir yaklaşım önerilmiştir. Önerilen yaklaşımda ESA modeli ile birlikte Trafo Ağı kullanılmıştır. Ön işlem adımı olarak popülasyon tabanlı AOA yöntemi tercih edilmiştir. AOA yöntemi bu çalışmada sürekli optimizasyon yaklaşımı ile probleme çözüm üretti ve SDN veri kümesindeki verimli özelliklerin seçilmesinde etkin bir rol aldı. Veri kümesi birkaç saldırı türlerini (SQL enjeksiyon, dağıtılmış hizmet reddi, kaba kuvvet vb.) içermektedir. Bu çalışmada saldırı türleri "saldırı" sınıfı olarak adlandırılmıştır ve ağ trafiğindeki diğer eylemler ise "normal" olarak sınıflandırılmıştır. İkili sınıflandırma ile gerçekleşen analizlerde %98,94 genel doğruluk başarısı elde edilmiştir. Sınıf türlerindeki dengesiz dağılımdan dolayı F1-skor başarıları da ölçülmüştür. Deney analizlerinde F1-skor değeri %99,15'ten %99,21'e arttığı gözlemlenmiştir.

Önerilen yaklaşımın avantajları;

- Yapay zekâ tabanlı bir yaklaşıma sahip olması,
- SDN, IoT teknolojilerinde ağ saldırılarının tespitinde başarılı bir model olması,
- AOA yönteminin web saldırıları tespitinde etkin bir özellik seçimine sahip olması,
- Önerilen yaklaşımın AOA yöntemi sayesinde zamandan tasarruf sağlayabilmesi,

Önerilen yaklaşımın dezavantajları;

- Uçtan uca bir model olmaması,
- Önerilen yaklaşımdaki yöntem veya modellerin el ile çalıştırılması (otomatik başla-bitiş olmaması).

Gelecek çalışmada, önerilen yaklaşıma alternatif olarak farklı meta-sezgisel optimizasyon yöntemleri ve diğer derin öğrenme yaklaşımları sisteme dahil edilerek uçtan uca bir model tasarlanacaktır. Ayrıca, IoT ve SDN paradigmalarından oluşmuş gerçek zamanlı veri kümeleri kullanarak analizler gerçekleştirilecektir.

Kaynaklar

[1] S.K. Tayyaba, M.A. Shah, O.A. Khan, A.W. Ahmed, Software Defined Network (SDN) Based Internet of Things (IoT), in: Proc. Int. Conf. Futur. Networks Distrib. Syst., ACM, New York, NY, USA, 2017: pp. 1–8. doi:10.1145/3102304.3102319.

[2] H. Polat, M. Turkoglu, O. Polat, Deep network approach with stacked sparse autoencoders in detection of DDoS attacks on SDN-based VANET, IET Commun. 14 (2020) 4089–4100. doi:10.1049/iet-com.2020.0477.

[3] N. Ahuja, G. Singal, D. Mukhopadhyay, N. Kumar, Automated DDOS attack detection in software defined networking, J. Netw. Comput. Appl. 187 (2021) 103108. doi:10.1016/j.jnca.2021.103108.

[4] I. Haque, D. Saha, SoftIoT: A resource-aware SDN/NFV-based IoT network, J. Netw. Comput. Appl. 193 (2021) 103208. doi:10.1016/j.jnca.2021.103208.

[5] S. Javanmardi, M. Shojafar, R. Mohammadi, A. Nazari, V. Persico, A. Pescapè, FUPE: A security driven task scheduling approach for SDN-based IoT–Fog networks, J. Inf. Secur. Appl. 60 (2021) 102853. doi:10.1016/j.jisa.2021.102853.

[6] M.V.O. de Assis, L.F. Carvalho, J.J.P.C. Rodrigues, J. Lloret, M.L. Proença Jr, Near real-time security system applied to SDN environments in IoT networks using convolutional neural network, Comput. Electr. Eng. 86 (2020) 106738. doi:10.1016/j.compeleceng.2020.106738.

[7] B. Nugraha, R.N. Murthy, Deep Learning-based Slow DDoS Attack Detection in SDN-based Networks, in: 2020 IEEE Conf. Netw. Funct. Virtualization Softw. Defin. Networks, IEEE, 2020: pp. 51–56. doi:10.1109/NFV-SDN50289.2020.9289894.

[8] M. Abdallah, N. An Le Khac, H. Jahromi, A. Delia Jurcut, A Hybrid CNN-LSTM Based Approach for Anomaly Detection Systems in SDNs, in: 16th Int. Conf. Availability, Reliab. Secur., ACM, New York, NY, USA, 2021: pp. 1–7. doi:10.1145/3465481.3469190.

[9] H. Polat, O. Polat, A. Cetin, Detecting DDoS Attacks in Software-Defined Networks Through Feature Selection Methods and Machine Learning Models, Sustainability. 12 (2020) 1035. doi:10.3390/su12031035.

[10] M. Revathi, V. V. Ramalingam, B. Amutha, A Machine Learning Based Detection and Mitigation of the DDOS Attack

by Using SDN Controller Framework, Wirel. Pers. Commun. (2021). doi:10.1007/s11277-021-09071-1.
[11] S. Chakraborty, SDN Intrusion Detection, 2021. (2021). https://www.kaggle.com/subhajournal/sdn-intrusion-detection (accessed November 16, 2021).
[12] F.A. Hashim, K. Hussain, E.H. Houssein, M.S. Mabrouk, W. Al-Atabany, Archimedes optimization algorithm: a new metaheuristic algorithm for solving optimization problems, Appl. Intell. 51 (2021) 1531–1551. doi:10.1007/s10489-020-01893-z.
[13] E.H. Houssein, B.E. Helmy, H. Rezk, A.M. Nassef, An enhanced Archimedes optimization algorithm based on Local escaping operator and Orthogonal learning for PEM fuel cell parameter identification, Eng. Appl. Artif. Intell. 103 (2021) 104309. doi:https://doi.org/10.1016/j.engappai.2021.104309.
[14] N. Van Thieu, Archimedes optimization algorithm code, 2021. (2021). https://github.com/thieu1995/mealpy/blob/master/mealpy/math_based/AOA.py (accessed November 19, 2021).
[15] H. Wang, W. Li, DDosTC: A Transformer-Based Network Attack Detection Hybrid Mechanism in SDN, Sensors. 21 (2021) 5047. doi:10.3390/s21155047.
[16] V. Tümen, B. Ergen, Intersections and crosswalk detection using deep learning and image processing techniques, Phys. A Stat. Mech. Its Appl. 543 (2020) 123510. doi:10.1016/j.physa.2019.123510.
[17] E. Başaran, Z. Cömert, A. Şengür, Ü. Budak, Y. Çelik, M. Toğaçar, Chronic Tympanic Membrane Diagnosis based on Deep Convolutional Neural Network, in: 2019 4th Int. Conf. Comput. Sci. Eng., 2019: pp. 1–4. doi:10.1109/ubmk.2019.8907070.
[18] M. Liu, F. Li, H. Yan, K. Wang, Y. Ma, L. Shen, M. Xu, A multi-model deep convolutional neural network for automatic hippocampus segmentation and classification in Alzheimer's disease, Neuroimage. 208 (2020) 116459. doi:https://doi.org/10.1016/j.neuroimage.2019.116459.
[19] A. Ahmed, K. Shaalan, S. Toral, Y. Hifny, A Multimodal Approach to Improve Performance Evaluation of Call Center Agent, Sensors (Basel). 21 (2021) 2720. doi:10.3390/s21082720.
[20] N. Tötsch, D. Hoffmann, Classifier uncertainty: evidence, potential impact, and probabilistic treatment, PeerJ Comput. Sci. 7 (2021) e398. doi:10.7717/peerj-cs.398.
[21] A. Arı, Ö.F. Alçin, D. Hanbay, Brain MR Image Classification Based on Deep Features by Using Extreme Learning Machines, Biomed. J. Sci. Tech. Res. 25 (2020). doi:10.26717/bjstr.2020.25.004201.
[22] Y. Bai, E. Yang, B. Han, Y. Yang, J. Li, Y. Mao, G. Niu, T. Liu, Understanding and Improving Early Stopping for Learning with Noisy Labels, (2021). http://arxiv.org/abs/2106.15853.
[23] M.K. Bohmrah, H. Kaur, Classification of Covid-19 patients using efficient fine-tuned deep learning DenseNet model, Glob. Transitions Proc. 2 (2021) 476–483. doi:10.1016/j.gltp.2021.08.003.