T.C.
KASTAMONU ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ
MALZEME BİLİMİ VE MÜHENDİSLİĞİ ANA BİLİM DALI

YAPAY SİNİR AĞI VE METASEZGİSEL SINIFLANDIRICILARINI KULLANARAK NESNELERİN İNTERNETİNDE ÇAPRAZ ATEŞ VE DDOS SALDIRILARINI TESPİTİ

MUSTAFA AHMED ELBERRI

DOKTORA TEZİ

DR.ÖĞR.ÜYESİ ÜMİT TOKEŞER

ARALIK - 2025
KASTAMONU

TEZ ONAYI

MUSTAFA AHMED ELBERRI tarafından hazırlanan "**YAPAY SİNİR AĞI VE METASEZGİSEL SINIFLANDIRICILARINI KULLANARAK NESNELERİN İNTERNETİNDE ÇAPRAZ ATEŞ VE DDOS SALDIRILARINI TESPİTİ**" adlı tez çalışmasının savunma sınavı **26.12.2025** tarihinde yapılmış olup aşağıda verilen jüri tarafından oy birliği ile Kastamonu Üniversitesi Fen Bilimleri Enstitüsü **Malzeme Bilimi ve Mühendisliği Ana Bilim Dalı Doktora Tezi** olarak kabul edilmiştir.

Danışman	Dr.Öğr.Üyesi Ümit TOKEŞER Kastamonu Üniversitesi	.....
Jüri Üyesi	Prof. Dr. Fatih Vehbi Çelebi Ankara Yıldırım Beyazıt Universitesi	.....
Jüri Üyesi	Doç. Dr. Reza AGHAZADEH THK Üniversitesi	.....
Jüri Üyesi	Doç. Dr. Elif AŞIKUZUN TOKEŞER Kastamonu Üniversitesi	.....
Jüri Üyesi	Doç. Dr. Zafer ÜNAL Kastamonu Üniversitesi	.....

Jüri üyeleri tarafından kabul edilmiş olan bu tez Kastamonu Üniversitesi Fen Bilimleri Enstitüsü Yönetim Kurulunca onanmıştır.

Enstitü Müdürü Prof. Dr. Selahattin KAYMAKCI .....

TAAHHÜTNAME

Bu tezin tasarımı, hazırlanması, yürütülmesi, araştırmalarının yapılması ve bulgularının analizlerinde bütün bilgilerin etik davranış ve akademik kurallar çerçevesinde elde edilerek sunulduğunu; ayrıca tez yazım kurallarına uygun olarak hazırlanan bu çalışmada bana ait olmayan her türlü ifade ve bilginin kaynağına eksiksiz atıf yapıldığını, bilimsel etiğe uygun olarak kaynak gösterildiğini bildirir ve taahhüt ederim.

MUSTAFA AHMED ELBERRI

ÖZET

DOKTORA TEZİ

YAPAY SİNİR AĞI VE METASEZGİSEL SINIFLANDIRICILARINI KULLANARAK NESNELERİN İNTERNETİNDE ÇAPRAZ ATEŞ VE DDOS SALDIRILARINI TESPİTİ

MUSTAFA AHMED ELBERRI

KASTAMONU ÜNİVERSİTESİ FEN BİLİMLERİ ENSTİTÜSÜ
MALZEME BİLİMİ VE MÜHENDİSLİĞİ ANA BİLİM DALI
DANIŞMAN:DR.ÖĞR.ÜYESİ ÜMİT TOKEŞER

Kimlik avı saldırıları, hassas kullanıcı bilgilerini çalmak için sahte web siteleri kullanarak çevrimiçi güvenlik için önemli bir tehdit oluşturmaktadır. Derin öğrenme teknikleri, özellikle de evrişimsel sinir ağları (CNN'ler), oltalama saldırılarını tespit etmek için umut verici araçlar olarak ortaya çıkmıştır. Bununla birlikte, geleneksel CNN tabanlı görüntü sınıflandırma yöntemleri sahte sayfaları etkili bir şekilde tanımlamada sınırlamalarla karşı karşıyadır. Bu zorluğun üstesinden gelmek için, bir CNN-LSTM hibrit modeli kullanarak kimlik avı saldırılarını tespit etmek için görüntü tabanlı bir kodlama yaklaşımı öneriyoruz. Bu yaklaşım, veri kümesini dengelemek, bilgilendirici özellikleri seçmek ve gri tonlamalı görüntüler oluşturmak için Otomatik Kodlayıcı ağına dayalı gelişmiş bir GAN olan SMOTE ile sürü zekası algoritmalarını birleştirmektedir. Üç kıyaslama veri kümesi üzerinde yapılan deneyler, önerilen yöntemin diğer tekniklere kıyasla üstün doğruluk, kesinlik ve hassasiyet elde ettiğini, kimlik avı saldırılarını etkili bir şekilde tanımladığını ve çevrimiçi güvenliği artırdığını göstermektedir.

ANAHTAR KELİMELER: Sahte Sayfalar, Kimlik Avı Saldırıları, SMOTE, Derin Öğrenme, Oyun Teorisi, Evrişimli Sinir Ağları, LSTM, Özellik Seçimi, Afrika Akbaba Optimizasyon Algoritması (AVOA)

Ekim 2025, 66 Sayfa

ABSTRACT

PH.D THESIS

USE ARTIFICIAL NEURAL NETWORK AND METAHEURISTIC CLASSIFIERS TO DETECT CROSSFIRE AND DDOS ATTACKS ON THE INTERNET OF THINGS

MUSTAFA AHMED ELBERRI

KASTAMONU UNIVERSITY INSTITUTE OF SCIENCE
DEPARTMENT OF MATERIALS SCIENCE AND ENGINEERING
SUPERVISOR:DR. ÖĞR.ÜYESİ ÜMİT TOKEŞER

Phishing attacks pose a significant threat to online security, utilizing fake websites to steal sensitive user information. Deep learning techniques, particularly convolutional neural networks (CNNs), have emerged as promising tools for detecting phishing attacks. However, traditional CNN-based image classification methods face limitations in effectively identifying fake pages. To address this challenge, we propose an image-based coding approach for detecting phishing attacks using a CNN-LSTM hybrid model. This approach combines SMOTE, an enhanced GAN based on the Autoencoder network, and swarm intelligence algorithms to balance the dataset, select informative features, and generate grayscale images. Experiments on three benchmark datasets demonstrate that the proposed method achieves superior accuracy, precision, and sensitivity compared to other techniques, effectively identifying phishing attacks and enhancing online security.

KEYWORDS: Fake Pages, Phishing Attacks, SMOTE, Deep Learning, Game Theory, Convolutional Neural Networks, LSTM, Feature Selection, African Vulture Optimization Algorithm (AVOA)

June 2025, 66 Page

TEŞEKKÜR

Bu araştırmanın gerçekleşmesine yaptığı yardımdan dolayı danışmanım Dr.Öğr.Üyesi Ümit Tokeşer'na teşekkürlerimi sunarım, bu tez onun rehberliği ve desteği olmadan mümkün olamazdı. Ayrıca tez izleme komitesine ve Ana bilim dalı öğretim üyelerine teşekkür ediyorum. İyi ve kötü günlerimde yanımda olan ve başarılı olmama yardımcı olan maddi ve manevi desteklerini esirgemeyen tüm aile bireylerine teşekkürlerimi sunmak istiyorum. Bana Türkiye'de doktora eğitimim boyunca burs imkânı sağlayan Libya hükümetine ve sorumlu yetkililere teşekkür ederim.

MUSTAFA AHMED ELBERRI

Kastamonu, 2025

İÇİNDEKİLER

Sayfa

TEZ ONAYI ..... **ii**
TAAHHÜTNAME ..... **iii**
ÖZET ..... **iv**
ABSTRACT ..... **v**
TEŞEKKÜR ..... **vi**
İÇİNDEKİLER ..... **vii**
ŞEKİLLER DİZİNİ ..... **ix**
TABLOLAR DİZİNİ ..... **x**
SİMGELER VE KISALTMALAR DİZİNİ ..... **xi**
1. GİRİŞ **1**
1.1 Arkaplan 1
1.2 Problem Tanımı 4
1.3 Problem çözümü 5
1.4 Araştırma Amacı 5
1.5 Araştırma Yaklaşımı 5
1.6 Motivasyonu 6
1.7 Çalışmanın Sınırlılıkları (Avantaj ve Dezavantaj) 6
1.8 Araştırma Katkıları 7
1.9 Tezin Organizasyonu 7
2. LİTERATÜR TARAMASI **8**
2.1 IoT Güvenlik Zorluklarına Genel Bakış 8
2.2 Geleneksel Algılama Teknikleri 9
2.2.1 İmza Tabanlı Algılama 9
2.2.2 Anomali Tabanlı Algılama 9
2.2.3 Sezgisel ve Kural Tabanlı Sistemler 9
2.3 Makine Öğrenimi Yaklaşımları 10
2.3.1 Gözetimli Öğrenme Modelleri 10
2.3.2 Denetimsiz ve Yarı Denetimli Öğrenme 10
2.3.3 Pekiştirmeli Öğrenme 10
2.4 Derin Öğrenme Tabanlı Tespit 10
2.4.1 Evrişimli Sinir Ağları (CNN'ler) 11
2.4.2 Tekrarlayan Sinir Ağları (RNN'ler) ve LSTM 11
2.4.3 Hibrit Modeller (CNN-LSTM) 11
2.4.4 Otomatik kodlayıcılar ve GAN'lar 11
2.5 Özellik Seçimi için Metasezgisel Algoritmalar 11
2.5.1 Genetic Algoritma (GA) 12
2.5.2 Parçacık Sürü Optimizasyonu (PSO) 12
2.5.3 Gri Kurt Optimizasyonu (GWO) 12
2.5.4 Afrika Akbaba Optimizasyon Algoritması (AVOA) 12
2.6 Veri Dengeleme Teknikleri 12
2.6.1 SMOTE 12
2.6.2 GAN Tabanlı Aşırı Örnekleme 12
2.7 Mevcut Çalışmalardaki Sınırlamalar 13
2.8 İlgili Çalışmalar 13

3. MATERIALS AND METHODS 24
3.1 Metodoloji 24
3.2 Önerilen Çerçeve 25
3.2.1 Kimlik Avı Saldırısı Tespiti için Akış Şemasını Anlama 26
3.3 Veri tabanı 27
3.3.1 Veri Setinin Dengelenmesi 27
3.4 Ön İşleme 32
3.5 Özellik Çıkarımı ve Özellik Seçimi 33
3.6 CNN-LSTM Ağı ile Sınıflandırma 42
4. SONUÇLAR VE TARTIŞMA 46
4.1 Arka Plan 46
4.2 Veriseti 46
4.3 Deneylerin Analizi 47
4.4 Değerlendirme ve Sonuçlar 49
5. SONUÇLAR VE GELECEK ÇALIŞMALAR 57
5.1 Sonuç 57
5.2 Gelecek Çalışmalar 58
KAYNAKLAR 60
ÖZGEÇMİŞ 66

ŞEKİLLER DİZİNİ

Sayfa

Şekil 2.1 Kimlik avı saldırılarının döngüsü (Abdulghani Ali Ahmed vd., 2023) 14
Şekil 2.2 2019 ve 2020'de kimlik avı saldırılarının sayısında artış (Kalabarige vd., 2022) 14
Şekil 2.3 Oltalama saldırılarıyla başa çıkmak için yöntem türleri (Kaushik vd., 2021) 15
Şekil 3.1 Önerilen yöntemin çerçevesi 25
Şekil 3.2 İki sınıflı dengesiz veri seti 27
Şekil 3.3 GAN ağı ile yapay örnekler oluşturma 28
Şekil 3.4 SMOTE-VAE-GAN ağı ile yapay örneklerin oluşturulması 29
Şekil 3.5 SMOTE yöntemiyle sentetik örneklerin oluşturulması (A. Sharm vd., 2022) 30
Şekil 3.6 Veri setinin SMOTE yöntemiyle dengelenmesi (Kingma ve Ba, 2014) 30
Şekil 3.7 SMOTE yöntemi Sözde Kod 31
Şekil 3.8 Nüfusun iki değerli akbabasından birine doğru ilerlemek (Abdollahzadeh vd., 2021) 39
Şekil 3.9 Akbabanın yiyecek pozisyonuna doğru agresif hareketi (Abdollahzadeh vd., 2021) 40
Şekil 3.10 Önerilen özellik seçimi akış şeması 41
Şekil 3.11 Konvolüsyon sinir ağı yapısı (Divakaran ve Oest, 2022) 43
Şekil 3.12 LSTM Sinir Ağı Hücresinin Yapısı 44
Şekil 4.1 Özellik Seçimi Aşamasında Amaç Fonksiyonunun, Özellik Seçim Algoritmasının Yineleme Sayısına Göre Azalması 48
Şekil 4.2 Özellik Seçimi Algoritmasının Tekrarlanmasıyla, Özellik Seçimi Aşamasında Sınıflandırma Hatasının Azalması 48
Şekil 4.3 PhishTank Veri Kümesinde Önerilen Yöntemin Değerlendirme Ölçütlerinin Analizi 49
Şekil 4.4 UCI Veri Kümesinde Önerilen Yöntemin Değerlendirme Ölçütlerinin Analizi 50
Şekil 4.5 Tan Veri Kümesinde Önerilen Yöntemin Değerlendirme Ölçütlerinin Analizi 50
Şekil 4.6 UCI Veri Kümesinde Önerilen Yöntemin Özellik Seçim Yöntemleriyle Doğruluk Açısından Karşılaştırılması 53
Şekil 4.7 PhishTank Veri Kümesinde Önerilen Yöntemin Derin Öğrenme Yöntemleriyle Doğruluk Açısından Karşılaştırılması 54

TABLOLAR DİZİNİ

Sayfa

Tablo 2.1 Önemli Yöntem, ilgili çalışmaların avantajları ve dezavantajları 22
Tablo 3.1 CNN ve LSTM Hibrit Sinir Ağı Ayarları 45
Tablo 4.1 Tan Veri Kümesinde Önerilen Yöntemin Derin Öğrenme Yöntemleriyle Karşılaştırılması 52
Tablo 4.2 Önerilen Yöntemin Derin Öğrenme ve Meta-Sezgisel Yöntemlerle Karşılaştırılması 55

SİMGELER VE KISALTMALAR DİZİNİ

Kısaltmalar

CNN : Evrişimsel Sinir Ağları (Convolutional Neural Networks)
GA : Genetik Algoritma (Genetic Algorithm)
GWO : Gri Kurt Optimizasyonu (Gray Wolf Optimization)
HHO : Harris Şahinleri Optimizasyonu (Harris Hawks' Optimization)
LSTM : Uzun Kısa Süreli Bellek (Long Short-Term Memory)
MDF : Çok Ölçekli Veri Katmanı Birleşimi (Multi-scale Data-layer Fusion)
MFF : Çok Ölçekli Özellik Katmanı Birleşimi (Multi-scale Feature-layer)
MIF : Çok Ölçekli Derinlemesine Birleşim (Multi-scale In-depth Fusion)
PSO : Parçacık Sürü Optimizasyonu (Particle Swarm Optimization)
TCN : Zamansal Evrişimsel Ağ (Temporal Convolutional Network)
WOA : Balina Optimizasyon Algoritması (Whale Optimization Algorithm)

1. GİRİŞ

1.1 Arkaplan

Çevrimiçi finansal hizmetlerin, özellikle PayPal gibi platformların yaygın şekilde benimsenmesi, fidye yazılımı saldırılarının potansiyel etkisini artırmıştır (Greco vd., 2023). WannaCry her ne kadar bazı oltalama (phishing) benzeri unsurlar içerse de, esasen kullanıcı etkileşimine ihtiyaç duymadan İnternete bağlı bilgisayarları enfekte eden, Windows sistemlerindeki EternalBlue adı verilen bir güvenlik açığını istismar eden bir fidye yazılımı saldırısıdır.

Çevrimiçi hizmetler yalnızca para transferi ile sınırlı olmayıp, Amazon gibi çevrimiçi alışveriş platformları da İnternet'in sunduğu önemli hizmetlerden biridir (Buono vd., 2023). Son yıllarda İnternet kullanıcılarının sayısında önemli bir artış gözlemlenmektedir; bu durumun temel sebebi, web sayfalarının çeşitliliğidir. Web sayfaları, her yerden ve her zaman hizmet sunarak kullanıcıların günlük yaşamlarında bu platformları düzenli olarak kullanmalarını sağlamaktadır. İnternet ve web sayfalarının sayısız kullanım alanı ve avantajlarına rağmen, her teknolojide olduğu gibi bu alanda da çeşitli zorluklar bulunmaktadır. Sahte web sayfaları, İnternet üzerindeki temel güvenlik sorunlarından biri olup, siber saldırıların başlıca nedenlerindendir (Marin vd., 2023).

Sahte sayfalar, orijinal ve güvenilir web sayfalarına görsel olarak oldukça benzemektedir. Bu tür sahte sayfalarda kullanıcıların kullanıcı adları ve parolaları gibi değerli bilgileri çalınmaktadır (Xu vd., 2023). Oltalama saldırıları, sosyal mühendislik tekniklerini kullanan, "phisher" olarak adlandırılan bireyler veya çevrimiçi suçluların kullanıcı bilgilerini dolandırıcılık yoluyla elde ettiği saldırılardır. Phisher'lar, sahte siteler aracılığıyla kullanıcıları kandırmakta ve kurbanlara e-posta, sosyal medya ya da kısa mesaj servisleri yoluyla oltalama bağlantıları göndermektedir (Lee vd., 2023) Oltalama saldırıları oldukça basit bir döngüye sahiptir ve yüksek düzeyde teknik uzmanlık gerektirmemektedir. Bu tür saldırılar genellikle banka bilgileri ve müşteri hesap bilgilerini çalmak amacıyla gerçekleştirilir. Oltalama saldırılarının etkinliği; web sitesinin benzerliği, sosyal mühendislik kullanımı ve kullanıcıların bilgi düzeyi

gibi çeşitli faktörlere bağlıdır (Syafitri vd., 2022). Oltalama saldırılarının tam sayısı kesin olarak bilinmemekle birlikte, güvenlik kurumlarının raporları bu saldırıların son yıllarda önemli ölçüde arttığını göstermektedir. Oltalama saldırıları yalnızca sahte web sayfalarıyla sınırlı değildir; bu sorun dijital para birimlerinde de mevcuttur. WannaCry, kripto para birimleri bağlamında gerçekleştirilen ve yaklaşık 4 milyar dolarlık zarara yol açan bir oltalama saldırısına örnektir. Bu siber saldırı, 150 ülkede 300.000'den fazla bilgisayarı etkilemiştir. Oltalama, Ethereum ekosistemi içerisinde gerçekleştirilen tüm siber suçların yarısından fazlasını oluşturmaktadır (Kim vd., 2023).

Sosyal ağların ve çevrimiçi finansal platformların kullanımının artması, hem bireyler hem de işletmeler açısından siber saldırılara karşı savunmasızlığı önemli ölçüde artırmıştır. Oltalama (phishing) saldırılarının en kritik türlerinden biri olan hedefli oltalama (spear phishing), belirli işletmelere ve kullanıcılarına yönelik ticari ve aldatıcı e-postaların gönderilmesiyle gerçekleştirilmektedir. Federal Soruşturma Bürosu (FBI) tarafından yapılan tahminlere göre, 2016 ile 2019 yılları arasında dünya genelindeki oltalama mağdurları, e-posta tabanlı oltalama saldırılarında 26 milyar ABD dolarından fazla kayıp yaşamıştır. 2018 yılında, yalnızca Avustralya'da hedefli oltalama saldırıları nedeniyle internet kullanıcılarının yaklaşık 60 milyon dolarlık kayba uğradığı bildirilmiştir. 2018–2019 yılları arasında oltalama saldırılarında Amerika Birleşik Devletleri'nin payı %39, Birleşik Krallık'ın %26 ve Avustralya'nın %11 olmuştur (Atlam ve Oluwatimilehin, 2022).

Oltalama saldırıları, insan doğasını ve iletişimi istismar ederek başarılı olmaktadır. Sosyal mühendislik kapsamında, oltacı ya da siber suçlu kullanıcılarla aldatma ve kandırma yoluyla iletişime geçmektedir. Örneğin, bir saldırgan kendisini bir banka çalışanı gibi tanıtarak kullanıcılara sahte e-posta gönderebilir. Yeterli bilgiye sahip olmayan kullanıcılar bu e-postalara inanarak değerli bilgilerini ifşa edebilir. Oltalama saldırılarıyla başa çıkmanın en temel yöntemi, bu saldırıların öncelikle tespit edilmesi ve ardından gerekli önlemlerin alınmasıdır. Bu kapsamda, oltalama saldırılarına karşı kullanılan yöntemler iki temel kategoriye ayrılmaktadır: kullanıcı eğitimi ve yazılımsal yöntemler (B. B. Gupta vd., 2017).

Kullanıcıları bilgilendirmeye yönelik eğitim yöntemleri uzun ve maliyetli bir süreçtir. Yazılımsal yaklaşımlar ise; liste tabanlı yöntemler (Jain ve Gupta, 2016), görsel benzerlik temelli yöntemler (A. Gupta vd., 2023), sezgisel (heuristic) yöntemler (da S vd., 2020), makine öğrenmesi tabanlı yöntemler (Divakaran ve Oest, 2022) ve derin öğrenme yöntemleri (Abdulrahman vd., 2023) olarak sınıflandırılmaktadır. Sezgisel yöntemler; adres içeriğindeki karakter türü, adres uzunluğu, nokta sayısı gibi çeşitli göstergelere dayanarak sahte sayfaları tespit etmeye çalışsa da, bu yöntemlerin hata oranı oldukça yüksektir. Kara liste (blacklist) temelli yöntemler, sahte sayfa adreslerinin veri tabanı ile karşılaştırma yaparak saldırıyı tespit eder; ancak bu yöntemler yüksek bellek kullanımı ve uzun arama süresi gerektirir. Kara liste ve sezgisel yaklaşımların aksine, makine ve derin öğrenme yöntemleri sıfırıncı gün saldırıları (zero-day attacks) gibi yeni tehditleri tespit etme yeteneğine sahiptir. Görsel temelli tespit yöntemleri, görseller ve logolar gibi sayfa üzerindeki görsel öğeleri kullanarak saldırıları tespit etmeye çalışır (Lin ed., 2022). Derin öğrenme yöntemleri, makine öğrenmesi yöntemlerine kıyasla daha güçlü bir desen tanıma kabiliyetine sahiptir ve daha yüksek öğrenme kapasitesi sunar. Ancak, Evrişimli Sinir Ağı (CNN) mimarisinin temel zorluklarından biri, giriş verisi olarak metin dizilerini (string) işlemesidir. Bu nedenle URL adreslerinin CNN ağına öğretilebilmesi için görsel biçime (image format) dönüştürülmesi gerekmektedir.

Bu çalışmanın temel amacı, oltalama saldırılarını tespit etmek için CNN ve LSTM mimarilerinin oyun kuramından ilhamla bütünleştirildiği yeni bir derin öğrenme yaklaşımını sunmaktır. Makalenin diğer bir amacı ise, CNN mimarisini AVOA (Afrika Akbabası Optimizasyon Algoritması) gibi sürü zekâsı yöntemleri ile geliştirerek performansını artırmaktır. Bu yöntemin önerilmesindeki motivasyon; oltalama saldırılarının tespitinde hata oranını azaltmak, CNN mimarisinin ağ güvenliği uygulamalarında etkin şekilde kullanılmasını sağlamak ve saldırıların zamanında tespiti ile zararı en aza indirmektir.

Önerilen oltalama saldırısı tespit yöntemi, ilk aşamada web sayfalarının ve bağlantıların temel özelliklerini çıkarmaktadır. İkinci aşamada ise SMOTE yöntemi (Feng vd., 2022), GAN ve Autoencoder tabanlı derin öğrenme yaklaşımları ile birleştirilerek oltalama ve yasal sınıflar arasındaki dengesizlik giderilmekte ve

öğrenme doğruluğu artırılmaktadır. AVOA (Abdollahzadeh vd., 2021), bu aşamada öznitelik seçiminde kullanılmakta olup, veri kümesinde boyutsal indirgeme sağlamaktadır. Boyut indirgeme, derin öğrenme modelinin giriş boyutunu azaltmakta ve öğrenme sürecini hızlandırmaktadır. Ayrıca, gereksiz verilerin elenmesi sayesinde model yalnızca önemli özniteliklere odaklanmakta ve CNN sınıflandırıcısının hata oranı azaltılmaktadır. Önerilen yöntem, boyutu indirgenmiş veriyi gri ölçekli görüntü biçimine dönüştürmekte ve CNN ile LSTM tabanlı derin öğrenme mimarisi bu görüntülerle eğitilmektedir. Bu çalışmada oltalama saldırılarını tespit etmeye yönelik olarak yazarların katkıları aşağıda sunulmaktadır:

- SMOTE tekniği ile veri kümesinin dengelenmesi ve GAN ile Otokodlayıcı (Autoencoder) ağına dayalı derin öğrenme yöntemiyle birleştirilmesi
- Kaos teorisi ile bütünleştirilmiş ikili (binary) bir AVOA varyantının tanıtılması
- Oltalama saldırılarının tespitinde Afrika akbabası algoritmasına dayalı sürü zekâsı yaklaşımı ile değişken seçimi yapılması
- Oltalama saldırılarının tespitinde CNN mimarisinin sürü zekâsı ile iyileştirilmesi
- CNN ile LSTM derin öğrenme mimarisinin entegrasyonu
- CNN ve LSTM ağları için giriş verisi olarak kullanılmak üzere dize (string) ve sayısal verilerin gri görüntü (grayscale image) biçiminde kodlanmasına yönelik yeni bir yöntemin sunulması

1.2 Problem Tanımı

Nesnelerin İnterneti (IoT) cihazlarının üstel artışı, özellikle Crossfire ve Dağıtık Hizmet Reddi (DDoS) gibi karmaşık siber saldırılar bağlamında, yeni güvenlik açıklarını beraberinde getirmiştir. İmza tabanlı (signature-based) ve sezgisel (heuristic) yaklaşımlar gibi geleneksel tespit yöntemleri, bu tehditlerin dinamik ve karmaşık yapısından dolayı, saldırıları doğru ve verimli bir şekilde tespit etmede yetersiz kalmaktadır. Buna ek olarak, derin öğrenme modelleri de dengesiz veri kümeleri, öznitelik fazlalığı ve yüksek boyutlu giriş verileri gibi zorluklarla karşı karşıyadır; bu durum, saldırıların doğru ve zamanında tespit edilmesini güçleştirmektedir.

1.3 Problem çözümü

Bu tez, oltalama (phishing) saldırılarını etkili bir şekilde tespit etmek amacıyla, derin öğrenme ve meta-sezgisel algoritmaların birleşiminden oluşan hibrit bir çerçeve önermektedir. Önerilen yaklaşımda, öznitelik öğrenimi ve ardışık veri modellemesi için CNN-LSTM mimarisi kullanılmakta; ayrıca, kaos teorisi ile güçlendirilmiş Afrika Akbabası Optimizasyon Algoritması (AVOA) temelli sürü zekâsı yöntemiyle öznitelik seçimi gerçekleştirilmektedir. Bunun yanı sıra, veri kümesindeki dengesizlik sorunları, sentetik veri üretimi amacıyla SMOTE yöntemi ile VAE-GAN modelinin entegrasyonu yoluyla giderilmekte; bu sayede tespit doğruluğu artırılmakta ve yanlış pozitif oranı azaltılmaktadır.

1.4 Araştırma Amacı

- IoT tabanlı oltalama (phishing) saldırılarını tespit etmek amacıyla, CNN-LSTM mimarisi kullanılarak hibrit bir derin öğrenme modeli geliştirmek.
- Özellik seçimini eniyilemek için kaotik haritalarla (chaotic maps) güçlendirilmiş geliştirilmiş meta-sezgisel algoritmalar (AVOA) uygulamak.
- SMOTE-VAE-GAN yöntemi ile dengesiz veri kümelerini dengeleyerek modelin genelleme performansını artırmak.

Önerilen sistemin doğruluk (accuracy), duyarlılık (sensitivity) ve kesinlik (precision) ölçütleri açısından birden fazla veri kümesi üzerinde değerlendirilmesini sağlamak ve mevcut yöntemlerle karşılaştırmak.

1.5 Araştırma Yaklaşımı

1. Önerilen yaklaşım, GS-A-CL olarak adlandırılmakta olup aşağıdaki aşamaları içermektedir:

2. Veri Ön İşleme: SMOTE ve GAN teknikleri kullanılarak sentetik örnekler üretilmekte ve veri kümesindeki sınıf dengesizlikleri giderilmektedir.
3. Özellik Çıkarımı ve Seçimi: Başlangıçta N-Gram yöntemleri ile öznitelikler çıkarılmakta; ardından, en bilgilendirici özniteliklerin seçimi için AVOA (Afrika Akbabası Optimizasyon Algoritması) uygulanmaktadır.
4. Özellik Temsili: Seçilen öznitelikler, derin öğrenme modeline giriş olarak kullanılmak üzere gri seviye (grayscale) görüntülere dönüştürülmektedir.
5. Model Eğitimi: Oltalama saldırı türlerinin etkin bir şekilde sınıflandırılması amacıyla, CNN-LSTM hibrit modeli eğitilmektedir.
6. Değerlendirme: PhishTank, UCI ve Tan gibi standart veri kümeleri kullanılarak, önerilen yaklaşımın doğruluğu, duyarlılığı, kesinliği gibi standart performans metrikleri ile geçerliliği test edilmektedir.

1.6 Motivasyonu

IoT ekosistemlerinde siber tehditlerin artan karmaşıklığı ve hacmi, yenilikçi ve akıllı tespit stratejilerine duyulan ihtiyacı giderek daha belirgin hâle getirmektedir. Mevcut yöntemler, veri dengesizliğini yönetme, sıfırıncı gün (zero-day) saldırılarına uyum sağlama ve öznitelik fazlalığını kontrol etme konusunda yetersiz kalmaktadır. Bu araştırma, IoT tabanlı sistemlerde siber güvenliği artırmak amacıyla, gerçek zamanlı tespit kabiliyetine ve yüksek doğruluk oranına sahip, sağlam, ölçeklenebilir ve zekâ temelli algoritmalar geliştirme gereksiniminden doğmuştur. Hedef, hem kullanıcıları hem de altyapıyı etkili bir şekilde koruyacak, gelişmiş bir güvenlik çözümüne katkı sunmaktır.

1.7 Çalışmanın Sınırlılıkları (Avantaj ve Dezavantaj)

Avantajlar:

- Birden fazla veri kümesinde kimlik avı saldırılarını tespit etmede yüksek doğruluk.

- Geliştirilmiş AVOA kullanarak sağlam özellik seçimi ve boyut azaltma.

- SMOTE ve GAN entegrasyonu ile dengesiz veri kümelerinin etkin kullanımı.

- Hibrit CNN-LSTM mimarisi sayesinde geliştirilmiş genelleme.

Dezavantajlar:

- Hibrit mimari ve kaotik AVOA nedeniyle artan hesaplama karmaşıklığı.

- Görüntü dönüşümü ve LSTM'nin sıralı yapısı nedeniyle daha uzun eğitim süresi.

- Her zaman gerçek dünya çeşitliliğini yansıtmayabilen sentetik verilere bağımlılık.

Performansı optimize etmek için hiper parametrelerde ince ayar yapma ihtiyacı.

1.8 Araştırma Katkıları

- Derin öğrenme ve sürü zekâsını birleştiren GS-A-CL adlı özgün bir oltalama saldırısı tespit çerçevesi geliştirilmiştir.
- Özellik seçimini iyileştirmek amacıyla, kaos teorisiyle güçlendirilmiş ikili (binary) bir Afrika Akbabası Optimizasyon Algoritması (AVOA) varyantı sunulmuştur.
- Gelişmiş veri artırma ve sınıf dengesi sağlama için SMOTE ve VAE-GAN modelleri entegre edilmiştir.
- Özellik vektörlerinin gri seviye görüntülere dönüştürülerek derin öğrenme modellerine giriş olarak kullanılmasına yönelik yeni bir yöntem önerilmiştir.

Benchmark veri kümelerinde, geleneksel ve derin öğrenme tabanlı yöntemlere kıyasla üstün tespit doğruluğu elde edilmiştir..

1.9 Tezin Organizasyonu

Bu araştırma çalışması beş bölüm hâlinde derlenmiş ve sunulmuştur. Çalışma, beş ana bölümden oluşacak şekilde hazırlanmış ve düzenlenmiştir: Bölüm 2, oltalama saldırılarına ilişkin araştırma geçmişi ve ilgili çalışmaların kapsamlı bir incelemesini içermektedir. Bölüm 3, oltalama saldırılarının tespiti için önerilen yaklaşımı ayrıntılı olarak sunmaktadır. Bölüm 4'te, önerilen stratejinin Python ve MATLAB ortamlarında gerçekleştirilmiş analizleri paylaşılmaktadır. Bölüm 5 ise, araştırma sonuçlarını ve gelecekteki çalışmalar için önerilen geliştirmeleri içermekte olup, oltalama saldırılarının tespit performansının artırılmasına yönelik ileriye dönük katkı alanlarını ele almaktadır.

2. LİTERATÜR TARAMASI

Nesnelerin İnterneti (IoT) cihazlarının günlük yaşama giderek daha fazla entegre edilmesi, özellikle Crossfire ve Dağıtık Hizmet Reddi (DDoS) saldırılarına karşı güvenlik açıklarında ciddi bir artışa yol açmıştır. Bu bölüm, IoT ağlarında saldırı tespiti konusundaki mevcut literatürü kapsamlı bir şekilde incelemektedir. Önceki çalışmalar kullanılan tekniklere göre sınıflandırılmakta, bu çalışmaların güçlü yönleri ve sınırlılıkları analiz edilmekte ve derin öğrenmenin meta-sezgisel algoritmalarla bütünleştirildiği gelişmiş hibrit yöntemlere olan gereksinim vurgulanmaktadır.

2.1 IoT Güvenlik Zorluklarına Genel Bakış

IoT ekosistemleri doğası gereği heterojen, kaynak açısından kısıtlı ve son derece dinamik yapılara sahiptir; bu da onları çeşitli siber saldırılara karşı savunmasız hâle getirmektedir. Bu ortamlarda karşılaşılan başlıca tehditler şunlardır:

- DDoS Saldırıları: Ağ kaynaklarını aşırı yükleyerek hizmetlerin kesintiye uğramasını hedefleyen saldırılardır.
- Crossfire Saldırıları: Düşük yoğunluklu trafik desenleriyle belirli ağ bağlantılarını tıkayarak gerçekleşen gizli bir DDoS saldırı türüdür.
- Oltalama (Phishing) ve Veri Sızdırma: Sosyal mühendislik yöntemlerinden faydalanarak hassas verilerin ele geçirilmesini amaçlamaktadır.
- Botnet Oluşumu: IoT cihazları, zayıf güvenlik önlemleri nedeniyle botnet ağına dahil edilmek için öncelikli hedeflerdir.

Atlam ve Wills (2021) gibi araştırmalar, geleneksel güvenlik mekanizmalarının bu tür gizli ve sürekli evrilen tehditleri tespit etmede yetersiz kaldığını göstermektedir.

2.2 Geleneksel Algılama Teknikleri

2.2.1 İmza Tabanlı Algılama

İmza tabanlı saldırı tespit sistemleri (IDS), bilinen saldırı türlerinden türetilmiş önceden tanımlanmış kurallara veya kalıplara dayanır. Bu sistemler, tanıdık tehditlerin tespitinde etkili olmakla birlikte, sıfırıncı gün (zero-day) veya polimorfik saldırılar gibi daha önce karşılaşılmamış ya da kendini değiştiren saldırı türlerini tanıma konusunda yetersiz kalmaktadır. Roesch (1999) tarafından geliştirilen SNORT gibi çalışmalar, bu tür imza tabanlı yaklaşımların sınırlılıklarını açıkça ortaya koymaktadır.

2.2.2 Anomali Tabanlı Algılama

Anomali tespiti, sistemin normal (temel) davranışından sapmaları belirlemeye dayanır. Bu alandaki erken dönem modellerden biri olan Denning (1987) tarafından önerilen yaklaşım, bu tür sistemlerin temelini oluşturmuştur. Ancak bu modeller, meşru davranışlardaki doğal değişkenlikleri de anomali olarak algılama eğiliminde olduklarından, yüksek düzeyde yanlış pozitif (false positive) üretme sorunuyla karşı karşıya kalmaktadırlar. Bu durum, anomali tabanlı tespit sistemlerinin güvenilirliğini ve uygulanabilirliğini sınırlamaktadır.

2.2.3 Sezgisel ve Kural Tabanlı Sistemler

Sezgisel (heuristic) teknikler, ağ trafiğini şüpheli kalıplar açısından analiz etmeye dayanır; örneğin, olağandışı port kullanımı veya TTL (Time-To-Live) değerlerindeki anomaliler gibi göstergeler değerlendirilir. Bu yöntemler, makine öğrenmesi tabanlı yaklaşımlara kıyasla daha hızlı çalışsa da, ölçeklenebilirlik ve dinamik tehdit ortamlarına uyum sağlama konusunda yetersiz kalmaktadır. Ayrıca, sabit kurallara dayandıkları için, yeni ve karmaşık saldırı türlerine karşı etkinlikleri sınırlıdır. Son yıllarda optimum parametrelerin bulunması, en önemli araştırma alanlarından biri haline gelmiştir (Abed vd., 2017; Ab-BelKhair vd., 2020; Elshara vd., 2024; Hazim Hameed vd., 2024; Hameed vd., 2023; Nureddin vd., 2020). Ayrıca, bu meta-sezgisel (metaheuristic) yöntemler, tıp bilimlerinde de uygulanmıştır (Abdullah vd., 2018; Abdullah vd., 2020; Aljanabi vd., 2018; Al Shalchi ve Rahebi, 2022; Javad vd., 2016;

Javad Rahebi ve Hardalaç, 2014; J Rahebi, 2011). En güncel meta-sezgisel yöntemler ise numaralı çalışmalarda ele alınmıştır.

2.3 Makine Öğrenimi Yaklaşımları

Makine öğrenimi (ML), verilerden kalıpları genelleştirme yeteneği nedeniyle IoT'de saldırı tespiti için ilgi görmüştür..

2.3.1 Gözetimli Öğrenme Modelleri

Panigrahi ve diğerleri (2021) gibi çalışmalar DDoS saldırılarını tespit etmek için Karar Ağaçları, Rastgele Ormanlar ve DVM'leri kullanmıştır. Bu modeller, uygulamada genellikle mevcut olmayan veya dengesiz olan etiketli veri kümeleri gerektirir.

2.3.2 Denetimsiz ve Yarı Denetimli Öğrenme

Kümeleme algoritmaları (örneğin K-Means) dahil olmak üzere denetimsiz yaklaşımlar, etiketli verilere ihtiyaç duymadan aykırı değerleri belirlemeye çalışır. Lakhina ve diğerleri (2004) istatistiksel akış analizi yoluyla anormallik tespitini göstermiştir.

Yarı denetimli modeller bir orta yol sunar ancak doğru başlangıç etiketlemesine bağlıdır.

2.3.3 Pekiştirmeli Öğrenme

Ortaya çıkan çalışmalar, ajanların ödüller yoluyla trafiği sınıflandırmayı öğrendiği RL tabanlı IDS çerçeveleri önermiştir. Ancak, bunlar çoğunlukla teoriktir ve eğitim karmaşıklığı nedeniyle yaygın olarak benimsenmemiştir.

2.4 Derin Öğrenme Tabanlı Tespit

Derin öğrenme, karmaşık örüntü tanıma ve otomatik özellik çıkarma kapasitesi nedeniyle izinsiz giriş tespiti için güçlü bir araç olarak ortaya çıkmıştır.

2.4.1 Evrişimli Sinir Ağları (CNN'ler)

CNN'ler, özellikle girdi özellikleri görüntü benzeri formatlara dönüştürüldüğünde, trafik sınıflandırması için etkili bir şekilde kullanılmıştır. Örneğin, Kim ve diğerleri (2022) DDoS modellerini tespit etmek amacıyla akış düzeyindeki trafik verilerini görüntü olarak analiz etmek için CNN'leri kullanmıştır.

2.4.2 Tekrarlayan Sinir Ağları (RNN'ler) ve LSTM

RNN'ler, özellikle de Uzun Kısa Süreli Bellek (LSTM) ağları, dizi modellemede ustadır ve zaman serisi tabanlı izinsiz giriş tespiti için kullanılmıştır. Yuwei Sun ve diğerleri (2022), başlık ve içerik modellerine dayalı olarak kimlik avı e-postalarını tespit etmek için LSTM kullanmıştı.

2.4.3 Hibrit Modeller (CNN-LSTM)

CNN ile LSTM'nin birleştirilmesi, sistemin hem uzamsal hem de zamansal özellikleri yakalamasını sağlar. Alsenani ve diğerleri (2023) böyle bir hibrit yapı kullanarak kimlik avı tespitinde doğruluğun arttığını göstermiştir.

2.4.4 Otomatik kodlayıcılar ve GAN'lar

Otomatik kodlayıcılar, normal örüntüleri yeniden oluşturma kapasiteleri nedeniyle anomali tespiti için kullanılır. GAN'lar eğitim verilerini artırmak için saldırı trafiğini simüle edebilir. Pham ve diğerleri (2021), veri dengesizliğini ele alarak GAN'ları kullanarak sentetik kimlik avı URL'leri oluşturmuştur.

2.5 Özellik Seçimi için Metasezgisel Algoritmalar

Optimum özelliklerin seçilmesi tespit doğruluğunu önemli ölçüde etkiler. Metasezgisel algoritmalar, IoT ağlarında bulunan yüksek boyutlu verileri işlemek için etkili bir yol sağlar.

2.5.1 Genetic Algoritma (GA)

GA, özellik seçimi için kullanılan en eski yaklaşımlardan biridir. Bununla birlikte, genellikle erken yakınsar ve büyük ölçekli problemlerle mücadele eder.

2.5.2 Parçacık Sürü Optimizasyonu (PSO)

PSO kuş sürüsü davranışını taklit eder ve IDS'de sıklıkla kullanılır. Basitliğine rağmen, genellikle yerel optimumlarda sıkışıp kalır.

2.5.3 Gri Kurt Optimizasyonu (GWO)

GWO küresel arama yeteneğini geliştirir ve orta derecede başarı ile ağ saldırı görevlerine uygulanmıştır (Mirjalili vd., 2014).

2.5.4 Afrika Akbaba Optimizasyon Algoritması (AVOA)

AVOA, sağlam küresel keşif ve yerel sömürü sergileyen yakın zamanda önerilen bir yöntemdir. Kaos teorisi ile entegre edildiğinde, (Abdollahzadeh vd., 2021) gösterildiği gibi, yakınsaması ve çeşitliliği önemli ölçüde artmaktadır.

2.6 Veri Dengeleme Teknikleri

IoT veri kümeleri genellikle dengesizdir ve normal verilerden çok daha az saldırı örneği içerir.

2.6.1 SMOTE

Sentetik Azınlık Aşırı Örnekleme Tekniği (SMOTE), azınlık sınıfının sentetik örneklerini oluşturur. Basitliği ve etkinliği nedeniyle yaygın olarak kullanılmaktadır.

2.6.2 GAN Tabanlı Aşırı Örnekleme

Gerçekçi sentetik örnekler oluşturmak için Üretken Çekişmeli Ağlar (GAN'lar) kullanılır. SMOTE-VAE-GAN yaklaşımında olduğu gibi SMOTE ile

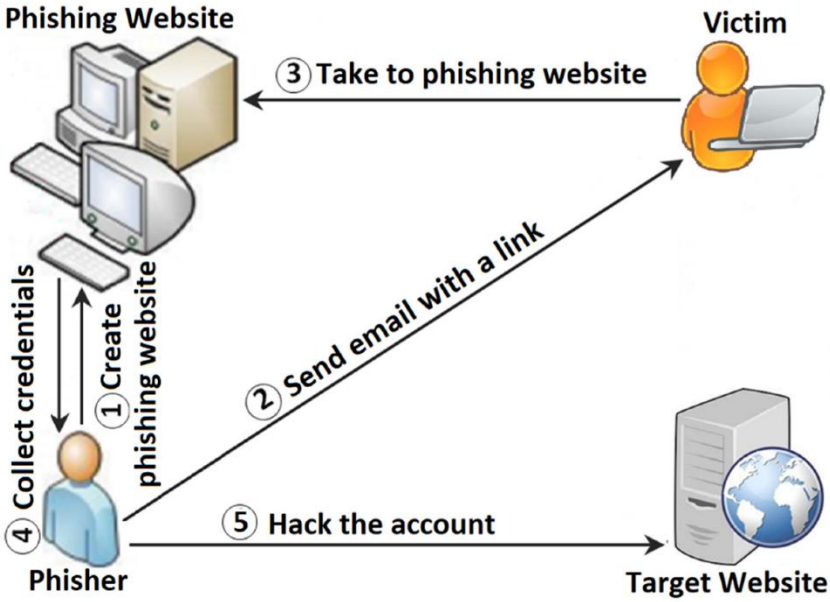
birleştirildiğinde, azınlık sınıfı örneklerinin kalitesi artarak eğitim dengesine yardımcı olur.

2.7 Mevcut Çalışmalardaki Sınırlamalar

- - Ölçeklenebilirlik: Birçok ML/DL modeli gerçek zamanlı IoT ortamları için iyi ölçeklenemez.
- - Veri Kıtlığı: Crossfire saldırıları için etiketlenmiş veri kümeleri nadirdir.
- - Genelleme: Statik özellik setleri ve zayıf doğrulama nedeniyle aşırı uyum yaygındır.
- - Yorumlanabilirlik: Derin modeller genellikle kara kutu gibi davranır.
- - Hibridizasyon Eksikliği: Çok az sayıda çalışma veri dengeleme, özellik seçimi ve hibrit derin ağları tek bir uyumlu sisteme entegre etmektedir.

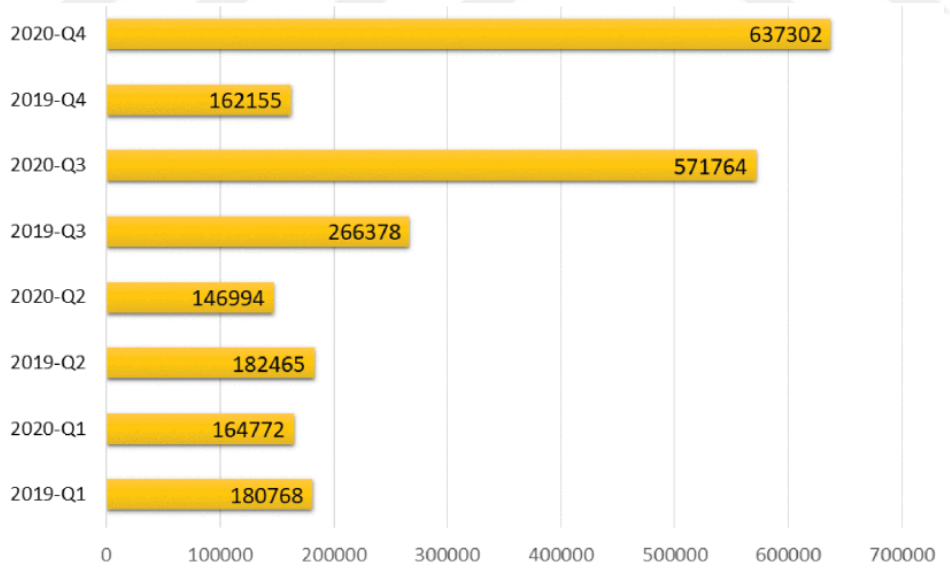
2.8 İlgili Çalışmalar

Şekil 2.1'e göre, oltalama (phishing) saldırıları çok aşamalı bir döngüye sahiptir. İlk aşamada, saldırgan özgün bir web sitesini taklit ederek benzer bir sahte site oluşturur. Bir sonraki aşamada, oluşturulan sahte web sitesi saldırgan tarafından internete yüklenir. Ardından saldırgan, kullanıcıları bu sahte siteye yönlendirmek amacıyla çeşitli iletişim kanalları aracılığıyla sahte site bağlantılarını kullanıcılara gönderir. Kullanıcılar bu bağlantıya tıkladığında, sahte web sitesine yönlendirilirler ve burada kimlik hırsızı kullanıcıya ait bilgileri ele geçirir.



Şekil 2.1 Kimlik avı saldırılarının döngüsü (Ahmed vd., 2023)

Hırsız, kullanıcıları kandırmak için özellikle e-posta olmak üzere sosyal mühendislik yöntemlerini kullanır. İnternet üzerindeki sahte sayfalara bir örnek Şekil (2.2)'de gösterilmektedir.

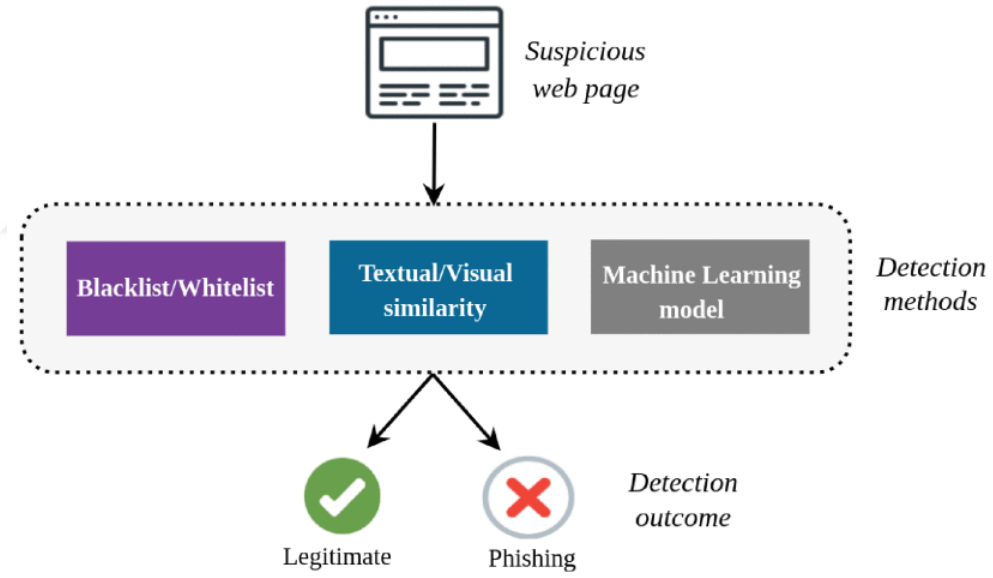


(Şekil 2.2) 2019 ve 2020'de kimlik avı saldırılarının sayısında artış (Kalabarige vd., 2022)

Sahte sayfalarda, web sitesinin görünümü orijinal web sitesine benzemektedir. Yine de, saçma bağlantıların sayısı da dahil olmak üzere web sitesinin kaynak kodunu analiz ederek, bunun kimlik avı olduğunu fark etti. Bu örnekte, web sayfasındaki birkaç boş bağlantı, web sitesinin kimlik avı olduğunu göstermektedir. Sahte siteler, oltalama

olduklarını kanıtlayan bir dizi özelliğe sahiptir. Birçok kimlik avı sitesinde JavaScript kodları kullanıcıların sağ tıklamasını engeller. Sahte siteler genellikle kısa ömürlüdür ve alan adları yakın zamanda kaydedilmiştir. Bazı özellikler web sayfası bağlantılarıyla ilgilidir. Örneğin, sahte web sayfaları @ karakterleri veya olağandışı port adresleri içerir. (Şekil2.2)'ye göre, Kimlik Avıyla Mücadele Çalışma Grubu'nun (APWG) raporlarına göre, 2020'deki kimlik avı web sitelerinin toplam sayısının 2019'daki dört sezonun tamamındaki rakamları aştığı görülmektedir. Bu rapora göre, 2020'nin ilk çeyreğinde 165.772 kimlik avı sitesi, 2020'nin dördüncü çeyreğinde yaklaşık 637.302'ye yükselmiştir.

(Şekil 2.3), oltalama saldırılarını tespit etmek için kullanılan çeşitli teknikleri göstermektedir.



Şekil 2.3 Oltalama saldırılarıyla başa çıkmak için yöntem türleri (Kaushik vd., 2021)

Bu yöntemler, liste tabanlı, benzerlik tabanlı ve makine öğrenmesi tabanlı yaklaşımları içermektedir. Kara liste yöntemleri, oltalama saldırılarını tespit etmek için kuralları ve saldırı imzalarını içeren bir veritabanı kullanır. Bu yöntemler basit yapılıdır; ancak yüksek bellek gereksinimi duyarlar ve arama süreleri uzundur.

Görsel benzerlik ölçüm yöntemleri ise web sayfalarının görüntülerini ve logolarını analiz ederek sahte site tespiti yapar. Oltalama saldırılarının tespiti için kullanılan

görsel yöntemler, ileri düzey görüntü işleme araçları ve algoritmalar gerektirir. Bu tür yöntemler oldukça karmaşıktır ve hata oranları da azımsanmayacak düzeydedir.

Bir diğer yaklaşım olan sezgisel (heuristic) yöntemlerde ise keşifsel fonksiyonlar kullanılarak saldırılar tespit edilmeye çalışılır. Bu yöntemlerde; web sitesinin adres uzunluğu, alt alan adı (subdomain) sayısı, alan adının kullanım süresi ve özel karakterlerin varlığı gibi çeşitli özellikler kontrol edilir. Sezgisel yöntemler kabul edilebilir düzeyde doğruluk sunsa da, sıfırıncı gün (zero-day) saldırılarını tespit etmede yetersiz kalmaktadır.

Makine öğrenmesi ve derin öğrenme yöntemleri, oltalama saldırılarının tespitinde çok sayıda yaklaşım sunmaktadır. Bu yöntemler, özellikle sıfırıncı gün saldırılarını tespit etme açısından önemli avantajlar sağlar. Bu bölümün devamında, oltalama saldırılarının tespitine yönelik çeşitli çalışmalar incelenmiş ve değerlendirilmiştir.

Makine Öğrenmesi ve Derin Öğrenmenin Sıfırıncı Gün Saldırılarını Tespit Etmedeki Avantajları.

Geleneksel imza tabanlı saldırı tespit sistemleri, sıfırıncı gün saldırılarını tespit etmede başarısız olurken; makine öğrenmesi ve derin öğrenme tabanlı yöntemler, imzalara bağımlı olmadan yeni saldırı örüntülerine uyum sağlayarak umut verici çözümler sunmaktadır (Hindy vd., 2020; Soltani vd., 2023).

Derin Öğrenme ve Makine Öğrenmesinin Sıfırıncı Gün Saldırılarının Tespitindeki Avantajları (Hindy vd., 2020; Soltani vd., 2023).

1. Örüntü Tanıma: Derin öğrenme algoritmaları, sıfırıncı gün saldırılarının tespitinde hayati önem taşıyan karmaşık örüntüleri tanıma konusunda son derece başarılıdır; bu saldırılar genellikle alışılmadık ya da ince davranışlar sergileyebilir.
2. Uyarlanabilirlik: Makine öğrenmesi modelleri, sürekli olarak yeni verilerle öğrenme ve kendini güncelleme yeteneğine sahiptir. Bu da onları, ortaya çıkan tehditleri ve gelişen saldırı örüntülerini tespit etmede etkili kılar.

3. Özellik Çıkarımı: Derin öğrenme modelleri, ham veriden anlamlı özellikleri otomatik olarak çıkarabilir. Bu durum, manuel özellik mühendisliği gereksinimini azaltırken genel tespit başarımını da artırır.

Aşırı Öğrenme (Overfitting) ve Yanlış Pozitiflerin (False Positive) Azaltılması (Guo, 2023):

1. Derin öğrenme ve makine öğrenmesi yöntemleri her ne kadar yüksek potansiyele sahip olsa da, aşırı öğrenme (overfitting) problemi ve buna bağlı yüksek oranda yanlış pozitif üretme gibi zorluklarla karşı karşıyadır. Bu sorunları hafifletmeye yönelik başlıca stratejiler şunlardır:

2. Veri Çoğaltma (Data Augmentation): Eğitim verisinin miktar ve çeşitliliğinin artırılması, modelin genelleme yeteneğini artırarak yeni verilere karşı daha etkili hale gelmesini sağlar.

3. Düzenleme Teknikleri (Regularization): Dropout ve erken durdurma (early stopping) gibi yöntemler, aşırı karmaşık modellerin cezalandırılması yoluyla aşırı öğrenmenin önüne geçilmesini sağlar.

4. Topluluk Yöntemleri (Ensemble Methods): Birden fazla modelin bir araya getirilerek kullanılması, model performansını artırırken aynı zamanda aşırı öğrenme riskini azaltır.

5. Yanlış Pozitiflerin Azaltılması: Anomali tespiti ve itibar tabanlı filtreleme gibi son işlem (post-processing) teknikleri, yanlış pozitif oranlarının azaltılmasında etkilidir. Anomali tespiti, beklenen örüntülerden sapmaları tanımlar; itibar tabanlı filtreleme ise veri kaynağının güvenilirliğini dikkate alarak filtreleme yapar.

Derin Öğrenme ve Geleneksel Tekniklerin Birleştirilmesi (He vd., 2018; Alabandi, 2017; Xie vd., 2023).

Derin öğrenme ve makine öğrenimi, geleneksel güvenlik tekniklerinin yerini almak yerine onları tamamlar. Hibrit bir yaklaşım şunları sunar:

1) Tamamlayıcı Güçlü Yönler: Her iki yaklaşımın da güçlü yönlerinden yararlanarak kendi zayıflıklarını gidermek.

2) Geliştirilmiş Doğruluk: Farklı tespit yöntemlerini birleştirerek yanlış pozitifleri azaltma ve genel tespit doğruluğunu artırma.

3) Geliştirilmiş Uyarlanabilirlik: Hibrit sistem, yeni imzaları ve kuralları geleneksel bileşenlere dahil ederek değişen tehditlere uyum sağlarken, makine öğrenimi modellerinde yeni verilerden sürekli olarak öğrenir.

Rajeswary ve Thirumaran. (2023), numaralı çalışmada, oltalama saldırılarının tespiti yapay zekâ ve insan davranışlarının analizine dayanmaktadır. Bu çalışma, web sitelerinde kullanılabilir olan otomatik oltalama tespiti yöntemlerini, yapay zekâ ile insan davranışı temelli yöntemlerin birleşimiyle incelemektedir.

Pandey vd. (2023), numaralı çalışmada, makine öğrenmesi algoritmaları kullanılarak oltalama siteleri tespit edilmiştir. Karar ağacı, rastgele orman (Random Forest) ve gradyan artırmalı sınıflandırıcı (Gradient Boosting Classifier - GBC) gibi yöntemler ve üç farklı özellik seçme tekniği kullanılmıştır. 89 özellikten oluşan bir veri kümesiyle yapılan deneylerde, GBC algoritmasının saldırı tespitinde diğer yöntemlere göre daha yüksek doğruluk sağladığı gösterilmiştir.

Alsenani vd. (2023), numaralı çalışmada, Parçacık Sürü Optimizasyonu (Particle Swarm Optimization - PSO) algoritmasına dayalı bir özellik seçme yöntemi sunulmuştur. Deneysel sonuçlar, PSO tabanlı modelin makine öğrenmesi algoritmalarının doğruluğunu artırdığını ortaya koymuştur. En yüksek doğruluk oranına %97,81 ile yapay sinir ağı ulaşmıştır.

Yuwei vd. (2022), numaralı çalışmada, LSTM tabanlı bir e-posta oltalama saldırısı tespit yöntemi önerilmiştir. Bu yöntemde, federatif öğrenme ile LSTM modeli entegre edilmiştir. Elde edilen sonuçlara göre, önerilen sistem %83 doğruluk oranına ulaşmıştır.

Jha vd. (2022), numaralı çalışmada, derin öğrenme modelleri birleştirilerek bulut bilişim tabanlı bir oltalama saldırısı tespit sistemi geliştirilmiştir. URL analizinde

LSTM, logo analizinde YOLOv2 ve görsel benzerlik analizinde üçlü ağ modeli (triple network) kullanılmıştır. Sonuçlar, model kombinasyonlarının tekil modellere göre daha yüksek doğruluk sağladığını göstermiştir.

Almousa vd. (2022), numaralı çalışma, hiperparametre optimizasyonu ile birlikte derin öğrenme modelleri kullanarak oltalama saldırılarını tespit etmeyi amaçlamaktadır. Bu çalışmada kısa süreli bellek tabanlı modeller, tam bağlantılı sinir ağları (fully connected DNN) ve konvolüsyonel sinir ağları (CNN) olmak üzere üç farklı mimari test edilmiştir. Elde edilen sonuçlara göre, önerilen model yaklaşık %97,7 doğrulukla en iyi performansı göstermiştir. Ayrıca, ağ arama algoritması ve genetik algoritma, model doğruluğunu sırasıyla %0,1 ve %1 oranında artırmıştır.

S. R. Sharma vd. (2022), numaralı çalışmada, hibrit bir algoritma ile oltalama siteleri tespit edilmiştir. Veri ön işleme aşamasında, dengesiz veri ile başa çıkmak için uyarlamalı örnekleme yaklaşımı kullanılmıştır. İkinci aşamada, Rao'nun meta-sezgisel algoritması fazla özellikleri elemektedir. KNN sınıflandırıcısı, sahte ve gerçek sayfaları ayırt etmekte kullanılmıştır. UCI Makine Öğrenmesi Deposundan alınan verilerle yapılan deneylerde, %97,44 doğruluk elde edilmiştir.

S. Gupta ve Bansal. (2023), numaralı çalışma, sağlıkla ilgili web sitelerindeki oltalama sayfalarını görsel tekniklerle tespit etmeye odaklanmıştır. Karar ağacı, rastgele orman ve destek vektör makineleri kullanılmış; karar ağacı, en yüksek hassasiyeti sağlamıştır.

Yoo ve Cho. (2022), numaralı çalışmada, sosyal ağ servislerindeki (SNS) oltalama saldırıları, konvolüsyonel sinir ağları (CNN) ile tespit edilmiştir. Telegram chatbot ortamında uygulanan bu modelde, Text-CNN mimarisi kullanılmış ve LSTM'ye kıyasla daha yüksek doğruluk sağlanmıştır.

Trinh vd. (2022), numaralı çalışmada, görsel benzerliğe dayalı bir derin öğrenme yöntemiyle oltalama siteleri tespit edilmiştir. Görsel benzerliğe dayalı yöntemler, görsel özniteliklerin çıkarılmasında yüksek karmaşıklık içermektedir. Bu çalışmada, transfer öğrenme tekniği ile elde edilen öznitelikler, makine öğrenmesi algoritmalarına girdi olarak verilmiştir. VGG16 ve makine öğrenmesi kombinasyonu başarılı sonuçlar vermiştir.

Liu vd. (2022), numaralı çalışmada, oltalama sitelerini tespit etmek amacıyla derin anlamsal füzyon modelleri sunulmuştur. Çok ölçekli veri katmanı füzyonu (MDF), özellik katmanı füzyonu (MFF) ve derin füzyon modeli (MIF) önerilmiş ve en iyi sonuç MIF modeli ile elde edilmiştir. Model, %0,0047 düşük yanlış pozitif oranı ile dikkat çekmektedir.

Remmide vd. (2022), numaralı çalışmada, Zamanlayıcı Konvolüsyonel Ağ (TCN) kullanılarak phishing URL'leri tespit edilmiştir. TCN ile kelime gömme (word embedding) birleştirilmiş ve %98,95 doğruluk, %98 duyarlılık elde edilmiştir.

(Shaik vd. (2023), numaralı çalışmada, Bi-LSTM sinir ağı ile spam ve sahte e-postaların tespiti gerçekleştirilmiştir. Rastgele orman, yapay sinir ağı, destek vektör makineleri, LSTM ve Bi-LSTM gibi çeşitli algoritmalar denenmiş ve derin öğrenmenin daha yüksek doğruluk sunduğu gözlemlenmiştir.

Pham vd. (2021), numaralı çalışmada, GAN tabanlı sahte URL üretimiyle oltalama URL'lerinin tespiti amaçlanmıştır. WGAN-GP modeliyle dengeli ve sahte URL verileri üretilmiş, LSTM ve GRU sınıflayıcıları ile tespit gerçekleştirilmiştir.

Bu ve Cho. (2021), numaralı çalışmada, sıfırıncı gün oltalama saldırılarının tespiti için derin öğrenme temelli bir yaklaşım önerilmiştir. Derin öğrenme, mantıksal programlama ve klasik öğrenme yöntemleri bir araya getirilmiştir. Sonuçlar, duyarlılık indeksinin yaklaşık %3 oranında iyileştiğini göstermektedir.

Mohanty ve Acharya. (2023), numaralı çalışmada, URL analizine dayalı sağlam bir topluluk öğrenme modeli geliştirilmiştir. Özellik seçimi için çok değişkenli filtreleme, korelasyon analizi ve t-testi kullanılmış; ilk veri kümesinde %97, ikinci veri kümesinde %99,25 doğruluk elde edilmiştir.

Farida ve Mustopa. (2023), numaralı çalışmada, lojistik regresyon ve rastgele orman yöntemleri karşılaştırmalı olarak analiz edilmiştir. Özellik seçimi sonrası doğruluklar sırasıyla %92,718 ve %97,015 olarak raporlanmıştır.

(Thirumaran vd. (2023), numaralı çalışmada, doğal dil işleme ve derin öğrenme birleştirilerek hibrit bir model sunulmuştur. Rastgele orman ve CNN algoritmaları birleştirilerek, URL ve içerik analizine dayalı olarak otomatik sınıflandırma gerçekleştirilmiştir. Bu yaklaşım, standart makine öğrenmesi stratejilerine kıyasla daha yüksek doğruluk sağlamaktadır.

Genel Değerlendirme:

Tablo 2.1, incelenen oltalama saldırısı tespit çalışmalarının karşılaştırmasını sunmaktadır. İlgili çalışmaların analizi, saldırı tespitinde genellikle makine öğrenmesi ve derin öğrenme yöntemlerinin tercih edildiğini göstermektedir. Görsel yöntemler birçok sınırlamaya sahip olup, web sitesi görsellerinin analizinde zaman açısından maliyetlidir. Özellik seçimi yapılmayan makine öğrenmesi yöntemleri genellikle ortalama doğrulukla çalışmaktadır. Veri kümesi dengeleme yöntemleri, tespit doğruluğunu artırmada kritik rol oynamaktadır. Çalışmalar genellikle URL odaklı olsa da, içeriğin, alan adlarının, arama motorlarının ve kaynak kodun da önemli bilgiler taşıdığı unutulmamalıdır. Önerilen yöntem, sürü zekâsı ve veri dengeleme teknikleriyle bütünleştirilmiş bir derin öğrenme yaklaşımı sunmaktadır.

İlgili çalışmaların incelenmesi, aşağıdaki temel zorlukları ortaya koymaktadır:

- Dengesiz veriyle eğitilen derin öğrenme yöntemleri düşük doğruluk göstermektedir.
- Giriş verilerine boyut indirgeme uygulanmadığında eğitim süresi uzamaktadır.
- Derin öğrenme yöntemlerinde optimizasyon algoritmaları kullanılmadığında tespit doğruluğu düşmektedir.
- Birçok çalışmada basit GA ve PSO algoritmaları kullanılmıştır; ancak bu yöntemler problem uzayını yeterince etkili şekilde tarayamamaktadır. Yeni nesil sürü zekâsı algoritmaları, klasik sezgisel algoritmalara kıyasla daha karmaşık ve güçlü modelleme yeteneklerine sahiptir.

Tablo 2.1 Önemli Yöntem, ilgili çalışmaların avantajları ve dezavantajları

Araştırma	Yıl	Yöntem	Avantaj	Dezavantaj	Mevcut Yaklaşımlar
(Rajeswary ve Thirumaran, 2023)	2023	Yapay zekâ ve insan davranışı temelli yöntemler	İnsan davranışlarının kullanımı	Oltacı davranışı incelenmemiştir	Ek veri toplama ve analiz gerektirir
(Pandey vd., 2023)	2023	Karar ağacı, rastgele orman ve gradyan artırmalı sınıflayıcı	Birçok özelliğin incelenmesi	Akıllı özellik seçimi yok	Hesaplama karmaşıklığını artırabilir
(Alsenani vd., 2023)	2023	PSO tabanlı özellik seçimi	Yaklaşık %97.81 doğruluk	Dengesiz veri kümesi	Ek parametre ayarlaması gerekebilir
(Yuwei Sun vd., 2022)	2022	LSTM tabanlı oltalama e-posta tespiti	Sıfırıncı gün saldırılarının tespiti	Düşük doğruluk	Yüksek hesaplama maliyetine sahip olabilir
(Jha vd., 2022)	2022	Derin öğrenme modellerinin birleşimiyle bulut bilişim yaklaşımı	İçerik ve görsel analiz	Yüksek karmaşıklık	Gelişmiş altyapı ve uzmanlık gerektirir
(Almousa vd., 2022)	2022	Derin öğrenme ve hiperparametre optimizasyonu	CNN ve LSTM'ye göre daha yüksek doğruluk	Yüksek sistem yükü	Daha fazla zaman ve kaynak gerektirebilir
(S. R. Sharma vd., 2022)	2022	Rao algoritması ve KNN birleşimi	%97.44 doğruluk	KNN zayıf bir sınıflayıcıdır	Karmaşık veri ön işleme gerekebilir
(S. Gupta ve Bansal, 2023)	2023	Görsel teknikler ve makine öğrenmesi	Karar ağacı ile uygun hassasiyet	Zaman karmaşıklığı	Yorumlanabilirlik düşük olabilir
(Yoo ve Cho, 2022)	2022	Text-CNN	LSTM'ye göre daha yüksek doğruluk	Dengesiz veri	Ek eğitim verisi gerekebilir
(Trinh vd., 2022)	2022	Derin öğrenme ve görsel benzerlik	VGG yüksek doğruluk sağlar	İçerik kullanılmamış	Yüksek hesaplama maliyeti olabilir
(Liu vd., 2022)	2022	Derin anlamsal füzyon	%0.0047 yanlış pozitif oranı	Yüksek zaman maliyeti	Daha karmaşık model mimarisi gerekebilir
(Remmide vd., 2022)	2022	Zamansal konvolüsyonel sinir ağı	%98.95 doğruluk	Uzun eğitim süresi	Etkili eğitim için daha fazla veri gerekebilir
(Shaik vd., 2023)	2023	Bi-LSTM	LSTM'ye göre daha yüksek doğruluk	Dengesiz veri	Yüksek hesaplama maliyeti olabilir
(Pham vd., 2021)	2021	Oltalama URL tespiti için GAN ağı	Veri kümesinin dengelenmesi	İçerik özellikleri kullanılmamış	Karmaşık veri ön işleme gerekebilir
(Bu ve Cho, 2021)	2021	Derin öğrenme ve programlanmış bilgi	Hassasiyet indeksinde yaklaşık %3 artış	Özellik seçimi ve veri dengeleme eksikliği	Daha fazla optimizasyon süresi ve kaynak gerekebilir
(Mohanty ve Acharya, 2023)	2023	Sürdürülebilir grup öğrenmesi	Yüksek doğruluk	İçerik analizi eksik	Yüksek hesaplama maliyeti olabilir

Tablo 2.2 devamı

(Farida ve Mustopa, 2023)	2023	Korelasyona dayalı özellik seçimi	Boyut indirgeme ve öğrenme süresinde azalma	Dengesiz veri kümesi ve düşük doğruluk	Etkili eğitim için daha fazla veri gerekebilir
(Thirumaran vd., 2023)	2023	Doğal dil işleme ve derin öğrenme algoritmaları	Yüksek doğruluk	Özellik seçimi yok	Yüksek hesaplama maliyeti olabilir

- Sürü zekası davranışlarında kaos teorisinin kullanılmaması, küresel arama ve doğruluklarını ve yerel optimumlarda sıkışıp kalan meta sezgisel algoritmaları azaltacaktır.
- CNN ve LSTM gibi derin öğrenme mimarilerinin entegre edilmemesi, kimlik avı saldırılarını tespit etme doğruluğunu azaltır.

.

3. MALZEMELER VE YÖNTEMLER

3.1 Metodoloji

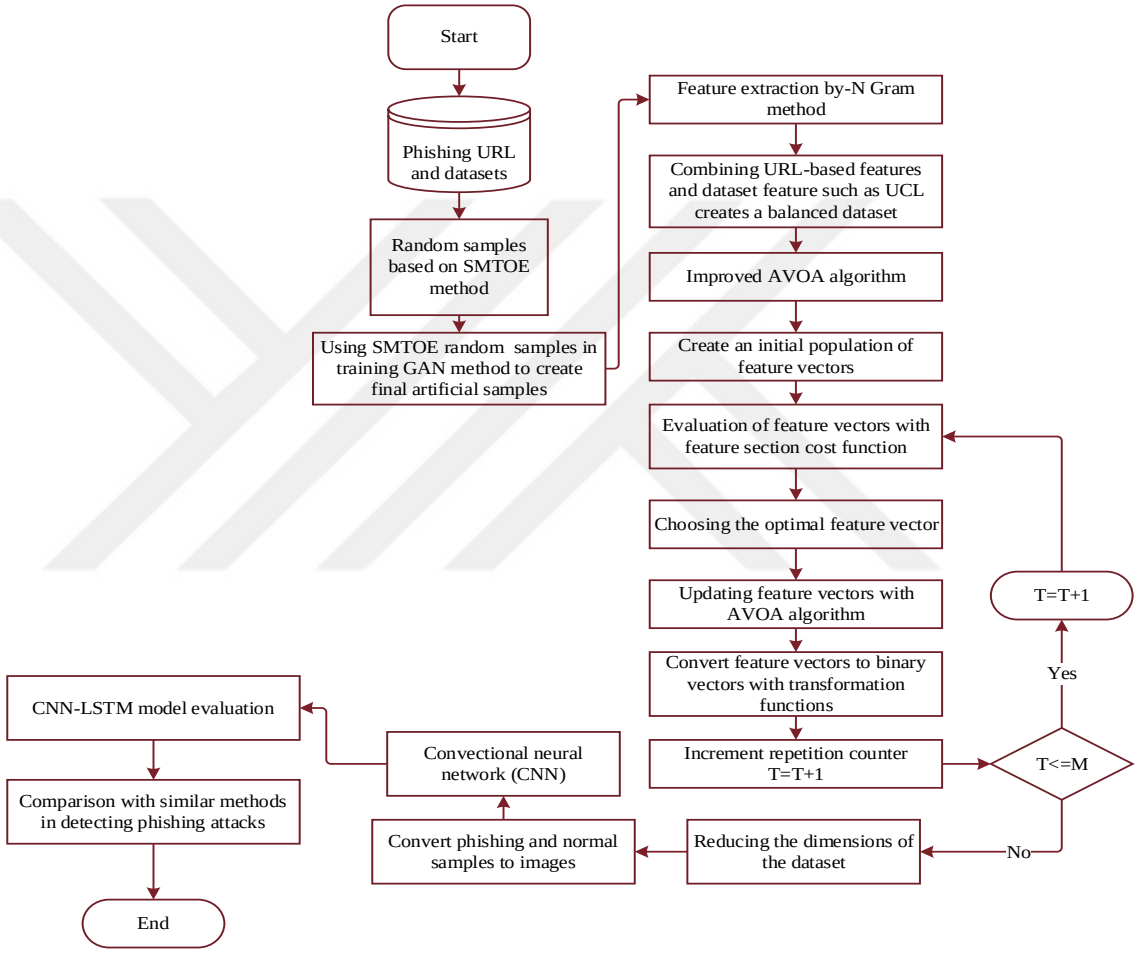
Önerilen oltalama saldırısı tespit yaklaşımı, önceki çalışmalarda karşılaşılan birçok zorluğu önemli ölçüde gidermektedir. İlgili çalışmaların incelenmesi, bazı araştırmalarda veri kümesinin dengelenmesi amacıyla GAN gibi derin öğrenme yöntemlerinin kullanıldığını ortaya koymuştur. Önceki çalışmalarda, veri kümesindeki dengesizlikleri gidermek için yaygın olarak SMOTE yöntemi uygulanmıştır. Bu çalışmada önerilen yöntem, GAN'ın geliştirilmiş bir sürümünü SMOTE ile birleştirerek veri kümesini daha dengeli hale getirmektedir.

Önceki çalışmalardan farklı olarak, önerilen model, derin öğrenme teknikleri ile birlikte sürü zekâsı yöntemlerinden yararlanarak özellik seçimi gerçekleştirmekte ve temel özellikleri belirlemektedir. Yeni geliştirilen AVOA (Adaptive Vulture Optimization Algorithm) algoritması, optimal çözümlerin hesaplanmasında yüksek doğruluk sağlamaktadır. Bu çalışmada, oltalama saldırılarının tespitinde CNN ve LSTM gibi derin öğrenme modellerinin performansını artırmak amacıyla sürü zekâsı algoritmaları kullanılmıştır. Bu çalışmanın özgün katkıları aşağıda özetlenmiştir:

- Olasılıksal türdeki GAN modeli ile veri kümesinin dengelenmesi,
- GAN modelinin SMOTE yöntemi ile iyileştirilmesi,
- N-Gram yöntemiyle özellik çıkarımı gerçekleştirilmesi,
- AVOA algoritmasının kaos teorisi ile geliştirilmesi,
- AVOA algoritmasının ikili (binary) uyarlamasının sunulması,
- Özellik seçiminin AVOA yöntemiyle gerçekleştirilmesi,
- Sahte sayfa ve bağlantıların tespitinde CNN ve LSTM'nin bütünleşik olarak kullanıldığı optimal bir yaklaşımın önerilmesi.

3.2 Önerilen Çerçeve

Bu makalede önerilen ve GS-A-CL (GAN ve SMOTE ile AVOA ve CNN-LSTM kombinasyonu) olarak adlandırılan metodolojinin şematik gösterimi (Şekil 3.1)'de gösterilmektedir.



Şekil 3.1 Önerilen yöntemin çerçevesi

Kimlik avı saldırılarını tespit etmek için önerilen yaklaşım aşağıdaki adımlardan oluşmaktadır:

- SMOTE ve GAN yöntemlerinin entegrasyonu ile veri kümesinin dengelenmesi
- N-Gram yöntemi ile özellik çıkarımı
- Veri kümesinin normalleştirilmesi
- Kaos teorisi ile Afrika akbabası algoritmasının performansının iyileştirilmesi

- Geliştirilmiş Afrika akbabası algoritması ile özellik seçimi
- Seçilen özelliklerin renkli görüntülere kodlanması
- LSTM sinir ağı ile birleştirilmiş CNN sinir ağı eğitimi
- Önerilen yöntemin test örnekleri ile değerlendirilmesi

3.2.1 Kimlik Avı Saldırısı Tespiti için Akış Şemasını Anlama

Bu akış şeması, oltalama saldırılarını etkili bir şekilde tespit etmek ve sınıflandırmak amacıyla geliştirilen derin öğrenme tabanlı bir yöntemi özetlemektedir.

Adım 1: Veri Artırma (Data Augmentation)
1.1. Veri kümesinde azınlık sınıfı olan oltalama örneklerinin sayısını artırmak için Sentetik Azınlık Aşırı Örnekleme Tekniği (SMOTE) kullanılmaktadır.
1.2. Oltalama saldırılarına ait sentetik örneklerin üretimi için Otoenkoder (Autoencoder) temelli geliştirilmiş Üretici Rekabetçi Ağ (GAN) modeli uygulanmaktadır.
Adım 2: Özellik Çıkarımı (Feature Extraction)
2.1. Oltalama sayfalarından anlamlı bilgi ve temel özellikleri çıkarmak amacıyla Afrika Akbabası Optimizasyon Algoritması (AVOA) adlı sürü zekâsı algoritması kullanılmaktadır2.2. Elde edilen bilgiler arasından kritik ve etkili özellikler seçilmektedir.
Adım 3: Özellik Temsili (Feature Representation)
3.1. Seçilen özellikler gri ölçekli (grayscale) görüntülere dönüştürülmektedir.
Adım 4: Derin Öğrenme Modeli Eğitimi
4.1. Gri ölçekli görüntüler kullanılarak Konvolüsyonel Sinir Ağı (CNN) ve Uzun Kısa Süreli Bellek (LSTM) modeli birlikte eğitilmektedir.
Adım 5: Değerlendirme
5.1. Eğitilen CNN-LSTM modeli, oltalama saldırılarını tespit etme başarımını değerlendirmek için standart karşılaştırmalı veri kümeleri üzerinde test edilmektedir.
Bulgular ve Sonuç:
Bu çalışmada önerilen derin öğrenme yaklaşımı; SMOTE, geliştirilmiş GAN, AVOA ve CNN-LSTM modellerinin birleşimiyle oltalama saldırılarını etkili biçimde tespit etmektedir. Yöntem, özellik seçimi konusunda mevcut yöntemleri geride bırakmakta;

doğruluk, duyarlılık ve kesinlik gibi ölçütlerde yüksek başarı sağlamaktadır. Önerilen model, çevrim içi güvenliği güçlendirme potansiyeline sahiptir.

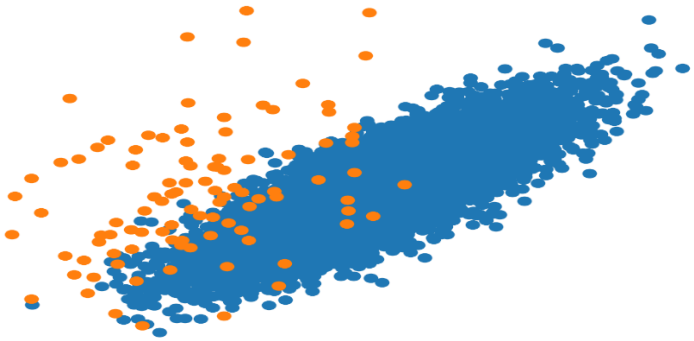
3.3 Veri tabanı

Önerilen yöntemin eğitimi ve değerlendirilmesi amacıyla, Phishtank gibi URL tabanlı veri kümeleri ile UCI gibi standart veri kümelerinde yer alan oltalama URL'leri kullanılmıştır. URL tabanlı veri kümeleri genellikle dengesiz yapıdadır ve UCI veri kümesinde oltalama sınıfına ait örnekler daha azdır.

Önerilen yöntemde, özellik çıkarımı aşaması URL'ler üzerinden gerçekleştirilmekte olup, elde edilen özellikler UCI veri kümesindeki özelliklerle birleştirilmektedir. Yeni oluşturulan veri kümesi, yalnızca çeşitli URL tabanlı özellikleri değil, aynı zamanda içerik, adres yapısı, kaynak kod, alan adı (domain) ve arama motoru ilişkili öznitelikleri de içerecek şekilde genişletilmiştir.

3.3.1 Veri Setinin Dengelenmesi

Farklı sınıflar arasındaki örnek sayısında önemli bir eşitsizlik dengesiz bir veri setini karakterize eder. (Şekil 3.2)'de dengesiz bir veri seti gösterilmektedir. Makine öğrenimi ve derin öğrenmede dengesiz veri seti, sınıflandırıcı modelin önemli bir hataya sahip olmasına neden olur.

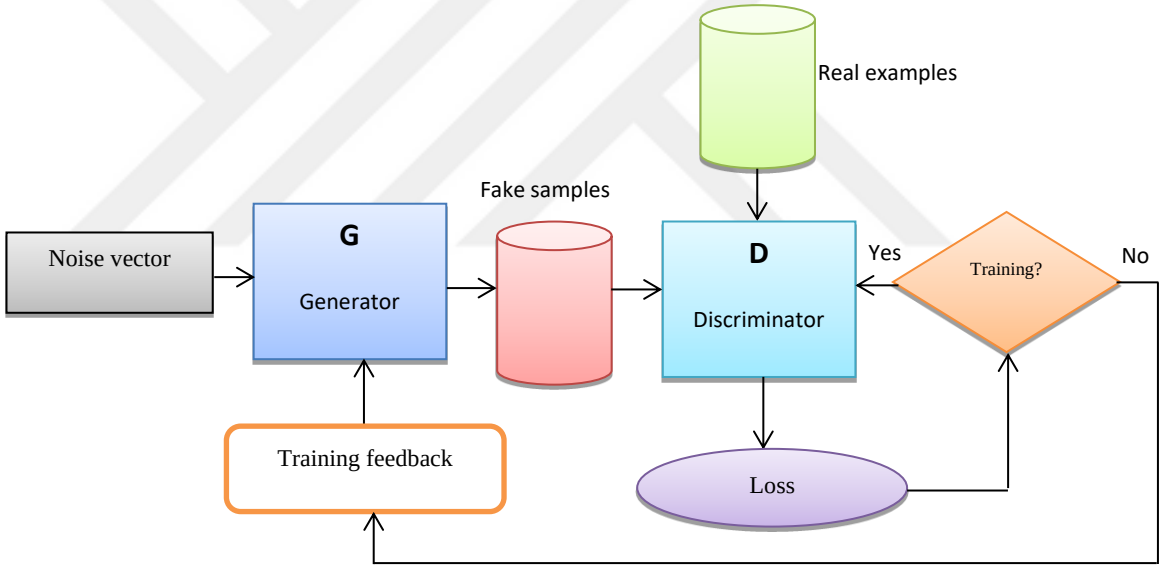


Şekil 3.2 İki sınıflı dengesiz veri seti

(Şekil 3.2) mavi ve turuncu renkte iki sınıf göstermektedir. Turuncu örneklerin sayısı, çoğunluk sınıfı olan mavi sınıftan çok daha azdır. Turuncu renkteki örneklerin sayısı daha azdır ve azınlık sınıfını oluştururlar. Veri setini dengelemek için etkili bir yöntem,

yapay veri oluşturma yöntemiyle azınlık örneklerinin sayısına ekleme yapmaktır. Çoğu araştırmada rastgele örnekler oluşturmak için GAN ve SMOTE yöntemleri kullanılmaktadır.

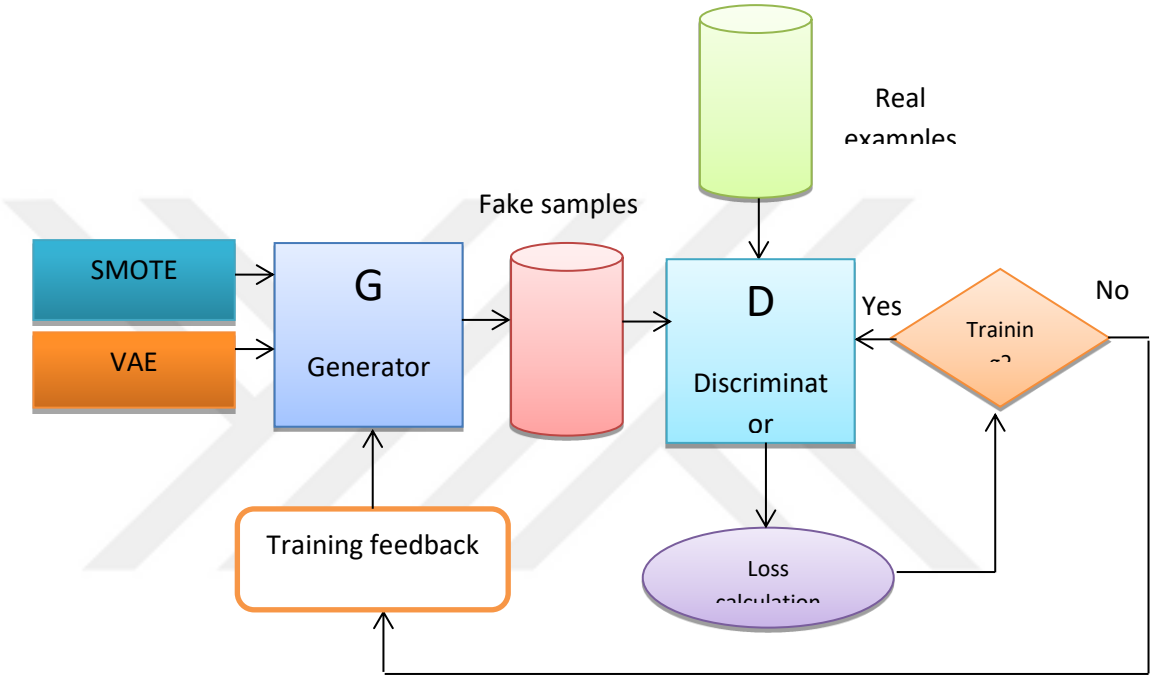
GAN ağları, jeneratör olarak adlandırılan bir derin öğrenme ağının bir ayırt edici içeren düşmanca bir süreçle rekabete girdiği bir oyun teorisi temeli üzerinde çalışır. Ayırıcı, üretici ağdan üretilen örnekleri orijinal verilerden ayırır. Üretecin rolü sahte örnekler oluşturmak ve çabalarını ayırt ediciyi kandırmak için kullanmaktır. Ayırıcı ise sahte örnekleri gerçek örneklerden ayıran bir sınıflandırıcıdır. Üreteç ve ayırt edici arasındaki oyun devam eder; üreteç ayırt ediciyi kandırmaya çalışır ve ayırt edici de üreteç tarafından kandırılmamaya çalışır. Rastgele örneklerin üretilmesi için GAN ağının yapısı (Şekil 3.3)'te gösterilmektedir.



Şekil 3.3 GAN ağı ile yapay örnekler oluşturma

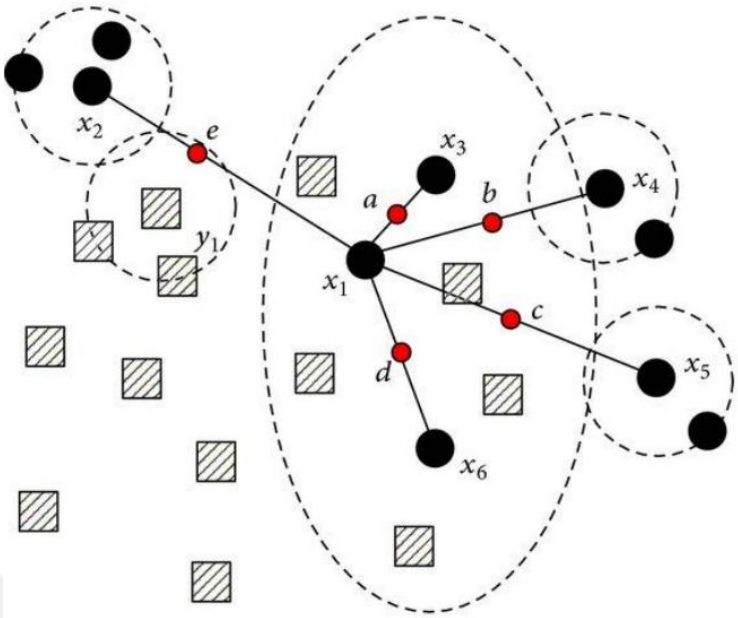
GAN'lar, eğitim veri kümesindeki verilere yakından benzeyen örnekler üretir ve özellikleri öğrenme kapasiteleri, oto kodlayıcılarla birleştirildiğinde daha da artar. Değişken oto kodlayıcı, eğitim veri kümesinde bulunan özellikleri içeren yeni örnek türleri üretir. GAN öğrenme aşamasında istikrarsızlık sergilerken, Varyasyonel Otomatik Kodlayıcı (VAE) GAN'a göre nispeten daha kararlıdır. GAN ve VAE'nin kombinasyonu, hem yüksek kalite hem de kararlılık ile karakterize edilen üretken bir modelle sonuçlanır. Bir Varyasyonel Otomatik Kodlayıcı Üretken Çekişmeli Ağ (VAE-GAN) tarafından üretilen sentetik verilerin kullanılması, kimlik avı saldırısı

tespit modelindeki dengesiz veri sorunlarını hafifletir. Rastgele örneklerin üretilmesi için önerilen yöntem aynı zamanda SMOTE algoritmasını kullanarak GAN'da bir üreteç girdisi olarak yapay örnekler üretmekte ve böylece üretecin ayırt edicinin aldatmacasını iyileştirme kabiliyetini artırmaktadır. (Şekil 3.4)'te, azınlık sınıfının yapay örneklerini üretmek için önerilen SMOTE-VAE-GAN modelinin yapısı gösterilmektedir.



Şekil 3.4 SMOTE-VAE-GAN ağı ile yapay örneklerin oluşturulması

SMOTE yönteminde, azınlık sınıflarından rastgele noktalar seçilir ve ardından (Şekil 3.5)'te gösterildiği gibi, en yakın komşularının K sayıları hesaplanır ve bunların kombinasyonu yeni örnekler elde etmek için bir çıktı olarak üretilir.



Şekil 3.5 SMOTE yöntemiyle sentetik örneklerin oluşturulması (A. Sharma vd., 2022)

(Şekil 3.5)'te çoğunluk örnekleri düz kareleri, azınlık örnekleri ise siyah daireleri göstermektedir. Yapay örnek sayısını artırmak için azınlık örnekleri arasındaki bağlantıdan kırmızı noktalar oluşturulmuştur. (Şekil 3.6), veri kümesi dengesini sağlamak için SMOTE yönteminde rastgele ve sentetik örnekler oluşturma prosedürünü göstermektedir.



Şekil 3.6 Veri setinin SMOTE yöntemiyle dengelenmesi (Kingma & Ba, 2014)

(Şekil 3.6) birincil verileri mavi ve yeşil örnekler olarak göstermektedir. SMOTE yöntemi, kırmızı dairelerle gösterilen azınlık örneklerinden veya yeşil dairelerden en yakın komşu yöntemine dayalı olarak birkaç yapay örnek oluşturur. Sentetik

örneklerin miktarının artırılması veri kümesi dengesine yol açar. Veri setinin dengelenmesi için SMOTE algoritmasının sözde kodu (Şekil 3.7)'de gösterilmektedir.

```
Algoritma: SMOTE algoritması Girdi: Xminor, Npentat, K
Çıktı: Xsynthetic_samples
Fonksiyon SMOTE (XMinor, XPercent, K)
Xsynthetic_samples={}
for i =1 to Size(XMinor) do
 M =K En Yakın Komşu (Xi, XMinor k)
 Per= [Npercent /100]
 while per! = 0 do
 Xneighbour = select random (M)
 Xsynthetic_samples= Xi+rand(0,1)*| Xneighbour - Xi |
 Per= Per-1
end while
end for
```

Şekil 3.7 SMOTE yöntemi Sözde Kod

Önerilen yöntemde, dengesiz veri sorununu azaltmak için VAE-GAN kullanılmaktadır. GAN ağı öğrenme sırasında istikrarsızlık sergilerken, VAE daha çeşitli örnekler üretir ve öğrenme boyunca göreceli istikrarı korur. Birkaç üretken modelin birleştirilmesi, yapay örnekler oluşturmak için her modelin avantajlarını sağlayabilir. VAE ağı bir kodlayıcı ve bir kod çözücü modülünden oluşur. Kodlayıcı, girişi gizli bir vektöre dönüştürür ve ardından kod çözücü bu gizli vektörü yaklaşık bir girişe yeniden yapılandırır. Kodlayıcı ve kod çözücü Denklem (3.1) ve (3.2)'de gösterilmiştir (Youngghyu Sun vd., 2023):

$$z \sim \text{Enc}\,(x) = q_{\phi}(z \mid x) \tag{3.1}$$

$$\hat{x} \sim \text{Dec}\,(z) = p_{\theta}(x \mid z) \tag{3.2}$$

Burada x, z ve $\hat{x}$ sırasıyla girdi, gizli vektör ve yaklaşık girdidir. φ ve θ kodlayıcı ve kod çözücü modellerinin parametreleridir. $q_\phi(z \mid x)$, $p_\theta(x \mid z)$'nin bir yaklaşımıdır. Denklem (3.3)'e göre yeniden yapılandırma hatasının toplamı olarak VAE kayıp fonksiyonu (Youngghyu Sun vd., 2023):

$$\begin{aligned} &J_{VAE} = J_{\text{recon}} + J_{\text{prior}}, \text{ with} \\ &J_{\text{recon}} = -\mathbb{E}_{q_\phi(z|x)}[\log\, p_\theta(x \mid z)] \\ &J_{\text{prior}} = D_{KL}\big(q_\phi(x \mid z) \parallel p_\theta(z)\big) \end{aligned} \tag{3.3}$$

GAN modeli, DKL ve $p_\theta(z)$'nin Kullback-Leibler ıraksamasını ve z'nin önceki dağılımını temsil ettiği bir üreteç ve bir ayrıştırıcıdan oluşur. Üreteç, gizli vektörü veri uzayına eşler ve ayrıştırıcıya v ve 1-v olasılığını atar. İkili çapraz entropili GAN'ın üreteç ve ayırıcıya göre kayıp fonksiyonu Denklem (3.4) ve (3.5)'te gösterilmiştir (Youngghyu Sun vd., 2023):

$$v = \text{Dis}\,(u) \in [0,1], u = \text{Gen}\,(w) \tag{3.4}$$

$$J_{GAN} = \log\,(\text{Dis}\,(u)) + \log\,(1 - \text{Dis}\,(\text{Gen}\,(w))) \tag{3.5}$$

Burada u gerçek bir örnektir ve w olasılık fonksiyonu p(w) olan bir rastgele değişkendir. VAE-GAN'ın amaç ve kayıp fonksiyonu denklem (3.6)'da formüle edilmiştir (Youngghyu Sun vd., 2023):

$$\begin{aligned} &J_{VAE-GAN} = J_{prior} + J_{Dis_l} + J_{GAN}, \text{ with} \\ &J_{Dis_l} = -\mathbb{E}_{q(z|x)}[\log\, p(Dis_l(x) \mid z)], \end{aligned} \tag{3.6}$$

3.4 Ön İşleme

Çoğu durumda, bir veri kümesinin özelliklerinin değerleri, makine öğrenimine zarar veren çeşitli ve farklı değerlere sahiptir. Makine öğreniminin doğruluğunu artırmanın pratik bir yolu, veri kümesinin tüm özellik değerlerini sıfır ile bir arasında dönüştürmektir. Önerilen yöntemde normalleştirme için Min-Max yöntemi kullanılmıştır. Kimlik avı saldırılarının veri seti için normalleştirme denklemi Denklem (3.7)'yi dikkate alır.

$$x' = \frac{x - min\ (x)}{max(x) - min\ (x)} \tag{3.7}$$

x normalleştirilmemiş değerdir ve x' bu özelliğin normalleştirilmiş değeridir. Normalleştirme bağlamında, "min(x)" ve "max(x)" sırasıyla kimlik avı veri kümesindeki özelliklerin minimum ve maksimum değerlerini temsil eder.

3.5 Özellik Çıkarımı ve Özellik Seçimi

Özellik çıkarımı için, URL özelliklerini çıkarmak için Ngram yöntemi kullanılır; bu tekniğin ayrıntıları (Bozkir vd., 2023)'de açıklanmıştır. Çıkarılan özellikler URL ile ilgilidir ve UCI gibi veri seti özellikleriyle birleştirilir. Bir sonraki adımda temel özellikleri seçmek için özellik seçimi aşaması gerçekleştirilir. Önerilen yaklaşımda, her özellik vektörü ya sıfır ya da bir olan elemanlardan oluşmaktadır. Bir özellik vektörünün i numaralı özelliği sıfıra eşitse bu veri setinde i numaralı özellik seçilmez, aksi takdirde bire eşitse seçilir. Önerilen yaklaşımda, her bir özellik vektörü denklem matrisinin (3.8) bir satırıdır ve her bir sütun o özellik vektörü ile ilişkili özelliklerdir. Bu matris n özellik vektörüne veya akbabaya sahiptir; her akbabanın d bileşeni vardır. UCI veri seti seçilirse, bu veri seti 30 özelliğe sahip olduğundan, bu matris denklem (3.9) gibi gösterilir. Bu denklemlerde P, kimlik avı saldırılarını tespit etmek için akbabaların veya özellik vektörlerinin başlangıç popülasyonudur.

$$P = \begin{bmatrix} \mathrm{p}_{1,1} & \cdots & \cdots & \mathrm{p}_{1,j} & \mathrm{p}_{1,d-1} & \mathrm{p}_{1,d} \\ \mathrm{p}_{2,1} & \cdots & \cdots & \mathrm{p}_{2,j} & \mathrm{p}_{2,d-1} & \mathrm{p}_{2,d} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \mathrm{p}_{n-1,1} & \cdots & \cdots & \cdots & \mathrm{p}_{n-1,d-1} & \mathrm{p}_{n-1,d} \\ \mathrm{p}_{n,1} & \cdots & \cdots & \mathrm{p}_{n,j} & \mathrm{p}_{n,d-1} & \mathrm{p}_{n,d} \end{bmatrix} \tag{3.8}$$

$$P = \begin{bmatrix} \mathrm{p}_{1,1} & \cdots & \cdots & \mathrm{p}_{1,j} & \mathrm{p}_{1,29} & \mathrm{p}_{1,30} \\ \mathrm{p}_{2,1} & \cdots & \cdots & \mathrm{p}_{2,j} & \mathrm{p}_{2,29} & \mathrm{p}_{2,30} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \mathrm{p}_{n-1,1} & \cdots & \cdots & \cdots & \mathrm{p}_{n-1,29} & \mathrm{p}_{n-1,30} \\ \mathrm{p}_{n,1} & \cdots & \cdots & \mathrm{p}_{n,j} & \mathrm{p}_{n,29} & \mathrm{p}_{n,30} \end{bmatrix} \tag{3.9}$$

Önerilen yöntemde, özellik vektörlerini değerlendirmek için bir amaç fonksiyonuna ihtiyaç vardır ve bir özellik vektörünü değerlendirmek için Denklem (3.10)'u kullanmak mümkündür:

$$Cost(P_i) = \alpha.\frac{1}{m}.\sum_{i=1}^{m}(\bar{Y}_i - Y_i)^2 + \beta.\frac{\|P_i\|}{\|d\|} \tag{3.10}$$

Bu denklemde indeks 1/m'dir. $\sum_{i=1}^{m}(\bar{Y}_i - Y_i)^2$ oltalama saldırılarının tespit hatasıdır ve $\frac{\|P_i\|}{\|d\|}$ koleksiyonun boyutlarının azaltılma miktarına eşittir. Bu iki bileşen, toplamı genellikle bire eşit olan alfa ve beta ağırlık katsayıları ile kullanılır ve alfa rastgele seçilir, beta ise alfa cinsinden belirlenir. Amaç fonksiyonu Denklem (3.11) olarak formüle edilmiştir:

$$Cost(P_i) = \alpha.\frac{1}{m}.\sum_{i=1}^{m}(\bar{Y}_i - Y_i)^2 + (1-\alpha).\frac{\|P_i\|}{\|d\|} \tag{3.11}$$

Bu denklemde ve amaç fonksiyonunda, Y_i gerçek sınıf değeri, $\bar{Y}_i$ ise sayfa türüne ilişkin tahmin edilen değeri ifade etmektedir. Oltalama (phishing) ve yasal (legal) sayfalar sırasıyla 1 ve 0 olarak kodlanmıştır. $\|P_i\|$, P_i özellik vektöründeki seçilmiş özniteliklerin sayısını; ‖d‖ ise veri kümesinin sütun sayısını ya da birincil özellik sayısını göstermektedir. Bu denklemde mmm, önerilen yöntemle oltalama saldırılarının değerlendirilmesi için kullanılan örnek sayısını temsil etmektedir. Genellikle, bu amaç fonksiyonunun değerini minimize edebilen herhangi bir özellik vektörü optimal olarak kabul edilmektedir.

AVOA (Adaptive Vulture Optimization Algorithm) algoritması belirli varsayımlar ve varsayılan parametrelerle çalışır; bazıları aşağıda açıklanmıştır:

- Her bir iterasyonda, N çözüm ya da akbaba, en iyi çözüm ya da avı bulmaya çalışır.
- Her yineleme sırasında, akbaba popülasyonu iki gruba ayrılır. Birinci kategori, en değerli iki akbabayı ve ikinci en değerli akbabayı içerirken; kalan akbabalar, zayıf akbabalar olarak adlandırılan ikinci grubu oluşturur.

- Her grubun avı bulma stratejisi ve arama kapasitesi birbirinden farklıdır.
- Aç olan akbabaların arama ve uçma enerjisi daha düşüktür. Bu nedenle, avın bulunduğu noktaya yaklaşarak iki optimal akbabayla rekabete girerler ve burada yerel arama (local search) veya verimlilik odaklı bir keşif süreci gerçekleştirilir.
- Tok olan akbabalar daha az saldırgan davranış sergiler ve avdan daha uzak noktalarda yer alarak geniş problem uzayını tarar; bu da küresel ve keşifsel bir arama (global exploratory search) stratejisini temsil eder.

Akbaba Optimizasyon Algoritması dört temel aşamadan oluşur:

1. Popülasyonun güçlü ve zayıf gruplara ayrılması: Eğer nnn adet akbaba varsa, n−2n-2n−2 tanesi zayıf gruba, kalan iki tanesi ise en iyi çözümler olarak güçlü gruba atanır.
2. Başlangıç popülasyonunun rastgele oluşturulması ve ardından en uygun iki çözümün seçilmesi.
3. Bu iki optimal akbaba, sürünün lideri olarak kabul edilir.
4. Diğer akbabalar, bu lider akbabaların bulunduğu konumu bir avın tahmini olarak kabul eder ve o yöne doğru hareket eder.

Bir akbabanın (örneğin XiX_iXi) bu iki optimal akbabadan birine doğru hareket etme olasılığı, eşit olarak %50 kabul edilebilir ya da rulet tekerleği (roulette wheel) mekanizmasına göre hesaplanan olasılık değeriyle ifade edilir. Bu olasılık mekanizması, ilgili literatürde Denklem (3.12) olarak tanımlanmıştır (Abdollahzadeh vd., 2021):

$$p_i = \frac{F_i}{\sum_{i=1}^{n} F_i} \tag{3.12}$$

Bu denklemde F_i, akbaba benzeri çözüm olan P_i'nin uygunluk (fitness) değerini; p_i ise, bir akbabanın popülasyondaki iki optimal akbabadan birine yönelme olasılığını ifade etmektedir.

Akbaba Optimizasyon Algoritması, optimal akbabaların daha fazla besine erişimlerinin olması nedeniyle daha az aç olduklarını, daha az saldırgan davrandıklarını ve daha uzun mesafeler uçabildiklerini varsayar. Buna karşılık, aç olan akbabalar optimal çözüme ulaşmak için daha fazla çaba sarf eder, daha saldırgandır ve besin için mücadeleye daha yatkındır.

Bu akbabalar, rastgele olarak iki optimal çözümden birini seçer ve onun yönünde hareket eder. Akbabaların açlık düzeyindeki artış ve tokluk düzeyindeki azalma, (Abdollahzadeh vd., 2021) numaralı çalışmada tanımlanan Denklem (3.13) ve Denklem (3.14) ile modellenmiştir.

$$t = \mathrm{h} \times (sin^{w}\left(\frac{\pi}{2} \times \frac{iter}{MaxIter}\right) + \cos\left(\frac{\pi}{2} \times \frac{iter}{MaxIter}\right) - 1) \tag{3.13}$$

$$F = (2 \times \mathrm{rand} + 1) \times \mathrm{z} \times \left(1 - \frac{iter}{MaxIter}\right) + \mathrm{t} \tag{3.14}$$

Başka bir deyişle, Akbaba Optimizasyon Algoritması (AVOA)'nın temel amacı, algoritmanın ilerleyen iterasyonlarında çözümlerin küresel (global) aramayı bırakıp yerel (local) aramaya odaklanmasını sağlamaktır. AVOA algoritmasında bu durum, akbabaların iki optimal akbabaya doğru uçarak onlarla rekabete girmesi anlamına gelmektedir.

Bu bağlamda, FFF fonksiyonu, akbabaların tokluk düzeyini temsil etmektedir. Bu fonksiyon, algoritmanın iterasyon sayısına bağlı olarak azalan bir yapıya sahiptir. Yani, algoritmanın son aşamalarında popülasyon, yerel arama amacıyla optimal çözümlere yönlendirilir.

Bu denklemlerde:

- iter: AVOA algoritmasının o anki iterasyon numarasıdır,
- MaxIter: AVOA algoritmasının maksimum iterasyon sayısını ifade eder,
- z: [-1, +1] aralığında rastgele seçilen bir değişkendir,
- h: [-2, +2] aralığında eşit dağılımlı (uniform) bir rastgele sayıdır,
- w: t ve F grafiklerinin oluşturulmasında kullanılan bir parametredir.

Bu ifadelerde:

- t, uçuş için rastgele adım üreten bir fonksiyondur ve değeri -0,5 ile +0,5 arasında üretilmektedir.
- F'nin değeri, hem t hem de algoritmanın iterasyon sayısına bağlı olarak azalan bir davranış sergiler.
- Her iterasyonda AVOA algoritması, birinci ve ikinci en iyi çözümleri sırasıyla BestVulture_1 ve BestVulture_2 olarak belirler.

Önerilen yaklaşımda, F değerine bağlı olarak akbabanın arama tipi — yani yemek arama mı yoksa yeme saldırma mı gerçekleştireceği — belirlenmektedir:

- Eğer |F|>1 ise, akbaba aç değildir; yalnızca yiyecek aramakta ve uçmaktadır; bu durumda küresel (global) arama gerçekleştirilir.
- Eğer |F|≤1 ise, akbabalar açtır ve avlarına saldırmaktadır; bu durumda yerel ya da küresel arama uygulanmaktadır.

Akbabaların hareketi ve uçuş davranışlarını yönlendirmek için 0 ile 1 arasında değer alan üç parametre kullanılır: $P_1 = 0.6$, $P_2 = 0,4$, $P_3 = 0,6$.
Küresel arama modunda, akbabalar tok durumdadır ve |F|>1 koşulunu sağlamaktadır; bu durumda problem uzayında serbestçe dolaşabilirler. Her akbaba için rastgele bir sayı üretilir:

- Eğer bu sayı p1'den küçükse, Denklem (3.15) uygulanır.
- Eğer sayı p1'den büyükse, Denklem (3.16) kullanılır (Abdollahzadeh vd., 2021).

$$P(i+1) = R(i) - D(i) \times F \quad (3.15)$$

$$D(i) = |X \times R(i) - P(i)| \quad (3.16)$$

Bu denklemlerde, P(i) bir akbabanın mevcut konumunu, P(i+1) ise yeni konumunu temsil eder; X, 0 ile 2 arasında rastgele bir katsayıdır; R(i), akbabaların rastgele konumlarından biridir ve bunlar arasında BestVulture alt 1 ve BestVulture alt 2 de bulunmaktadır. i ise algoritmanın tekrar sayacını gösterir (Denklem (3.17)) (Abdollahzadeh vd., 2021).

Bu denklemlerde:

P(i), bir akbabanın mevcut konumunu,
P(i+1), akbabanın bir sonraki (yeni) konumunu,
X, 0 ile 2 arasında rastgele seçilen bir katsayıyı,
R(i), $\mathrm{BestVulture}_1$ veya $\mathrm{BestVulture}_2$ dâhil olmak üzere, akbabalardan birinin rastgele seçilen konumunu,i, algoritmanın yineleme sayacını (iterasyon numarasını) temsil etmektedir.

Bu ifade, Denklem (3.17) olarak tanımlanmış ve (Abdollahzadeh vd., 2021) numaralı çalışmada verilmiştir.

$$P(i+1) = R(i) - F + rand((ub - lb).rand + lb) \quad (3.17)$$

Bu ifadelerde, "ub" ve "lb" terimleri sırasıyla özellik uzayının veya optimizasyon probleminin üst (upper bound) ve alt (lower bound) sınırlarını temsil etmektedir.
Eğer akbabalar aç ise, bu durumda |F|≤1 koşulu sağlanır ve bu durumda iki farklı durum ortaya çıkar. Bu durumlar, yerel arama (local search) veya verimlilik (productivity) davranışlarını gösterir.
Eğer |F|≥0.5 ise, bu kez karar süreci P_2 değişkenine bağlı olarak ikiye ayrılır:

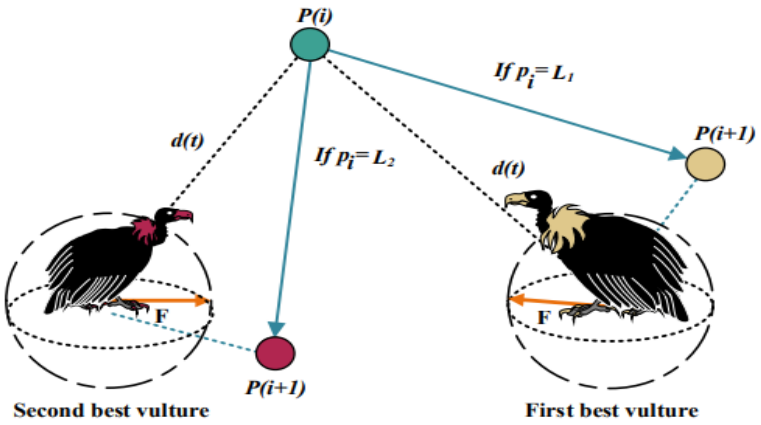
- Rastgele üretilen sayı p_2'den küçükse, Denklem (3.17) uygulanır.
- Rastgele sayı P_2'den büyükse, Denklem (3.18) devreye girer (Abdollahzadeh vd., 2021):

$$P(i+1) = D(i) \times (F + rand) - d(t) \quad (18)$$
$$d(t) = R(i) - P(i) \quad (19)$$

Bu denklemlerde, D(i) değeri Denklem (3.16)'ya benzer şekilde hesaplanmaktadır ve d(t), bir akbabanın $\mathrm{BestVulture}_1$ veya $\mathrm{BestVulture}_2$ olarak belirlenen iki optimal akbabadan birine olan mesafesini göstermektedir.

Başka bir ifadeyle, çok sayıda akbaba aynı konumda toplandığında, besin kaynağı üzerinde ciddi bir rekabet meydana gelebilir. Bu gibi durumlarda, fiziksel olarak güçlü akbabalar, diğerlerinin yiyeceğe ulaşmasını engellemeyi tercih edebilir. Bu durumda, aç olan akbaba, besine ulaşmak için diğer besin kaynağına, yani ikinci optimal

akbabanın bulunduğu tarafa doğru uçmayı dener. Bu davranış, (Şekil 3.8)'de görsel olarak temsil edilmiştir.



Şekil 3.8 Nüfusun iki değerli akbabasından birine doğru ilerlemek (Abdollahzadeh vd., 2021)

Eğer |F|≥0,5 ve rastgele sayı P_2'den daha anlamlı ise, çözümleri güncellemek için (3.20), (3.21) ve (3.22) denklemleri kullanılır (Abdollahzadeh et al., 2021):

$$S_1 = R(i) \times (\frac{rand \times P(i)}{2\pi}) \times Cos(P(i)) \tag{3.20}$$

$$S_2 = R(i) \times (\frac{rand \times P(i)}{2\pi}) \times sin(P(i)) \tag{3.21}$$

$$P(i+1) = R(i) - \frac{S_1 + S_2}{2} \tag{3.22}$$

Eğer |F|<0,5 ve rastgele sayı P_3'ten küçükse, çözümleri güncellemek için (3.23), (3.24) ve (3.25) denklemleri kullanılır (Abdollahzadeh vd., 2021):

$$A_1 = \text{BestVulture}_1(i) - (\frac{\text{BestVulture}_1 \times P(i)}{\text{BestVulture}_1 - P(i)^2}) \times F \tag{3.23}$$

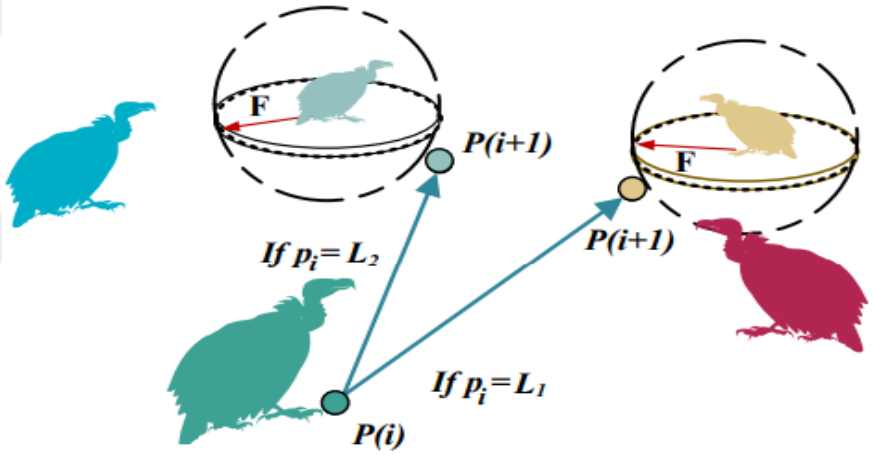
$$A_2 = \text{BestVulture}_2(i) - (\frac{\text{BestVulture}_2 \times P(i)}{\text{BestVulture}_2 - P(i)^2}) \times F \tag{3.24}$$

$$P(i+1) = \frac{(A_1 + A_2)}{2} \tag{3.25}$$

Eğer |F|<0,5 ve rastgele sayı P_3'ten daha anlamlı ise, çözümleri güncellemek için denklem (3.26) kullanılır (Abdollahzadeh vd., 2021):

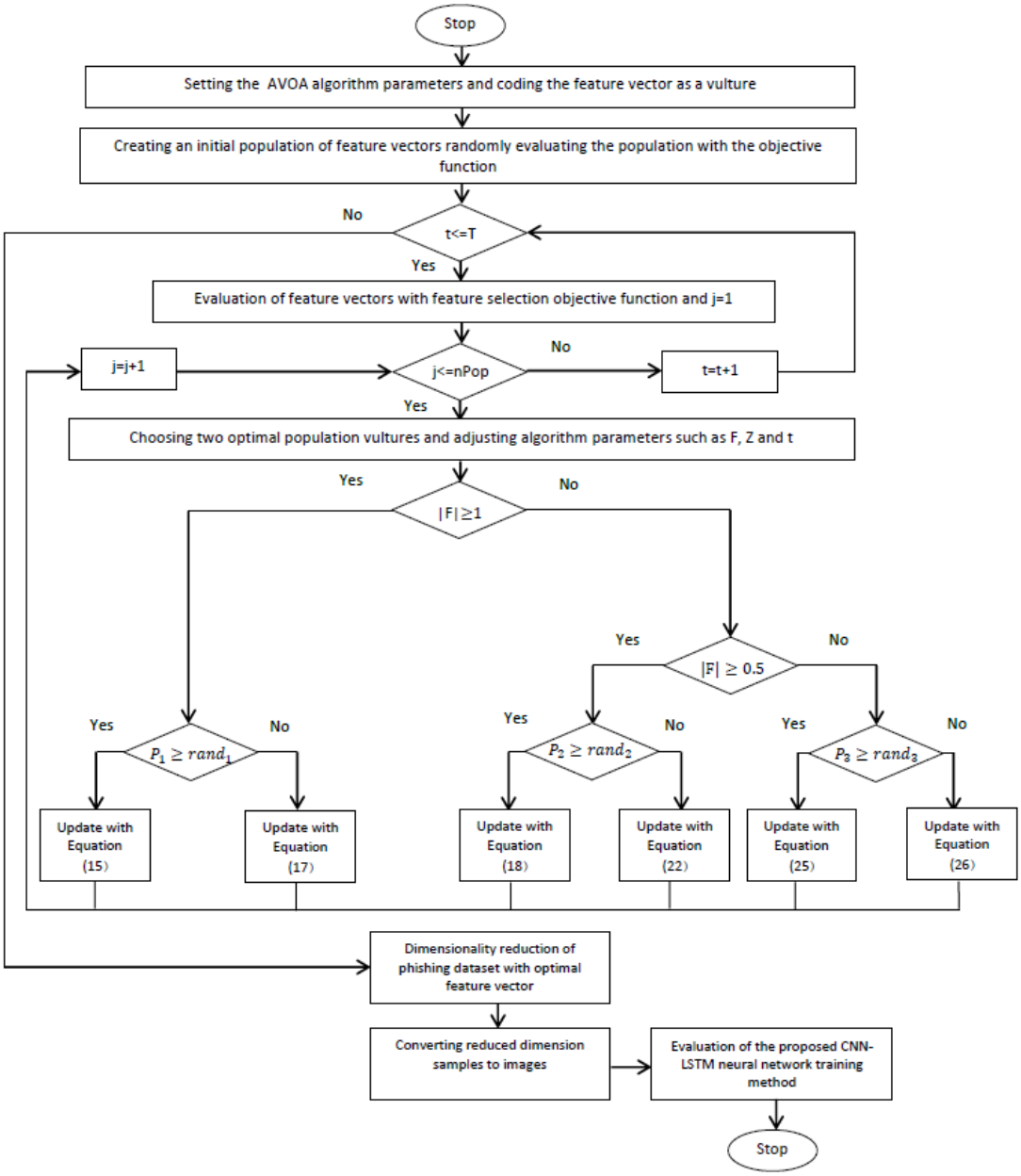
$$P(i+1) = R(i) - |d(t)| \times F \times Levy(d) \quad (3.26)$$

Bu denklemde Levy(d), d bileşenleri için rastgele bir uçuş fonksiyonudur ve |d(t)|, bir akbabanın optimum akbabalardan birine olan uzaklığına karşılık gelen mutlak değeri ifade eder. (Şekil 3.9), avına veya F arama yarıçapına en uygun cevaba doğru hareket eden bir akbabanın agresif davranışını göstermektedir. Algoritmanın tekrarı arttıkça, F'nin değeri azalır ve akbaba en uygun çözüm etrafında daha fazla arama yapar (Abdollahzadeh vd., 2021).



Şekil 3.9 Akbabanın yiyecek pozisyonuna doğru agresif hareketi (Abdollahzadeh vd., 2021)

(Şekil 3.10)'da, oltalama saldırılarını tespit etmek için önerilen yöntemi gösteren akış şeması gösterilmektedir.



Şekil 3.10 Önerilen özellik seçimi akış şeması

Meta sezgisel algoritmaların doğruluğunu artırabilecek şeylerden biri, rastgele bir değişken için algoritmanın her iterasyonu için kullanılan bir dizi rastgele sayı oluşturan kaos teorisine dayalı rastgele davranışların kullanılmasıdır. Denklem (3.27)'de Lojistik rastgele ve kaotik fonksiyonlar tanıtılmaktadır. Burada n, önerilen algoritmanın iterasyon sayısıdır.

$$X_{n+1} = a.X_n(1 - X_n) \tag{3.27}$$

Bu fonksiyonda X_0 (0,1) aralığında bir sayıdır, ancak 0, 0,25, 0,5, 0,75 ve 1 sayıları olamaz ve diğer yandan a 2'ye eşit olarak ayarlanmıştır. Denklem (3.28)'de Çadır rastgele ve kaotik fonksiyonu tanıtılmıştır. Burada n önerilen algoritmanın iterasyon sayısıdır.

$$X_{n+1} = \begin{cases} X_n/0{,}7 & X_n < 0{,}7 \\ 10/3X_n(1 - X_n) & otherwise \end{cases} \quad (3.28)$$

Bu fonksiyonda X_0 (0,1) aralığında bir sayıdır, ancak 0, 0,25, 0,5, 0,75 ve 1 sayıları olamaz ve diğer yandan a 4'e eşit olarak ayarlanmıştır. Denklem (3.29)'da Sinüzoidal rastgele ve kaotik fonksiyonlar tanıtılmıştır. Burada n önerilen algoritmanın iterasyon sayısıdır.

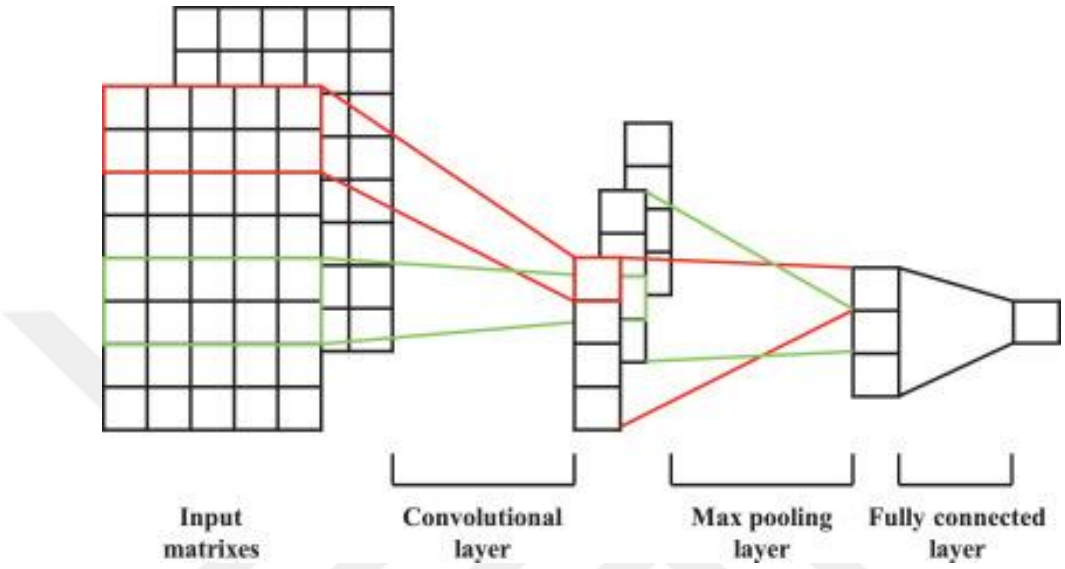
$$X_{n+1} = ax_n^2 \sin(\pi X_n) \quad (29)$$

Bu fonksiyonda X_0 0,7'ye ve a 2,3'e eşittir. Önerilen yöntemde, bu kaotik haritaların her biri AVOA'yı iyileştirmek ve AVOA'nın parametrelerini bu fonksiyonlar yardımıyla optimize etmek için kullanılır.

3.6 CNN-LSTM Ağı ile Sınıflandırma

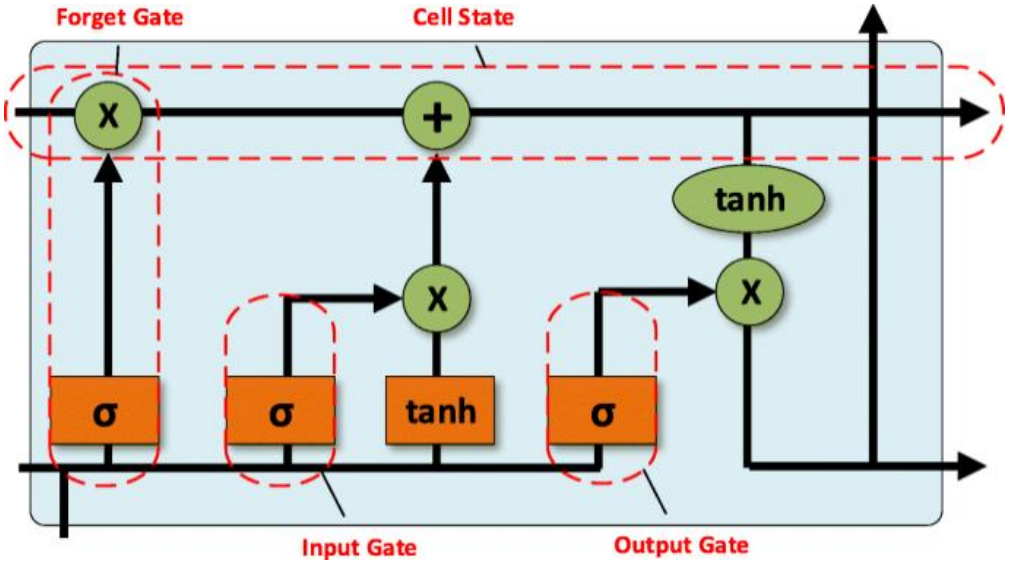
Seçilen özellikler, boyutluluğunu azaltmak için veri kümesine uygulanır. Özellik seçimi aşamasında k özellik seçildiğini, ardından veri setinden m kimlik avı örneğinin seçildiğini ve k içinde m matrisinin gri bir görüntü oluşturduğunu varsayalım. Gri tonlamalı bir görüntü oluşturmak için istenen matristeki sayılar 0 ile 255 arasında normalleştirilir. Kimlik avı olmayan örnekler için de benzer şekilde gri görüntüler oluşturulur ve birleşik CNN-LSTM modelini eğitmek için kullanılır. Önerilen yöntemde, 224 x 224 görüntülerin boyutları araştırmaya göre ayarlanmıştır (Thirumaran vd., 2023). Bu durumda, 224-k sütunları boş ve sıfıra ayarlanmış 224 x 244 matris vardır. M değeri veya görüntü matrisinin satır sayısı 224 olarak ayarlanmıştır. Derin öğrenme mimarilerinin aksine, bir CNN ağı, temel bir sinir ağının karmaşık özellikleri öğrenme yeteneği sınırlı olduğundan, özel bir çok katmanlı algılayıcı sinir ağı türüdür. CNN'ler, görüntü sınıflandırma ve tıbbi görüntü analizi de dahil olmak üzere çok sayıda uygulamada mükemmeldir. Önerilen yöntem, CNN sinir

ağına girdi olarak kimlik avı ve yasal örneklerle ilgili görüntüleri seçer. CNN, (Şekil 3.11)'de gösterildiği gibi konvolüsyon, entegrasyon ve tam bağlı katmanları içerir.



Şekil 3.11 Konvolüsyon sinir ağı yapısı (Divakaran ve Oest, 2022)

LSTM, zaman serisi analizi ve sınıflandırma gibi görevler için özel olarak tasarlanmış bir RNN (Yinelenen Sinir Ağı) derin öğrenme mimarisi çeşididir. LSTM, eğitim sürecinde ortaya çıkan kaybolan gradyan problemini çözmek için etkili bir şekilde kapı mekanizması kullanır. LSTM bellek hücresi; unutma kapısı (f), giriş kapısı (i), kontrol kapısı (c) ve çıkış kapısı (o) olmak üzere dört kapıya sahiptir. LSTM hücresinin temel yapısı (Şekil 3.12)'de sunulmuştur ve bu yapı, bir önceki bellek hücresinin çıktısı olan C_{t-1}'i içerir (Burgess vd., 2021).



Şekil 3.12 LSTM Sinir Ağı Hücresinin Yapısı

Bu sinir ağı, her zaman adımındaki giriş sinyali Xt, mevcut bellek hücresinin çıktısı C_t , önceki gizli katman durumu H_{t-1} ve mevcut gizli katman durumu HtH_tHt gibi bileşenleri kullanır. Unutma kapısı, bir önceki zaman adımındaki bilginin ne ölçüde dikkate alınacağını belirler ve bu işlem sonucunda C_{t-1}'deki her veri noktası için 0 ile 1 arasında bir değer elde edilir. Giriş kapısı, mevcut zaman adımından bellek hücresine ne kadar bilginin aktarılacağını düzenler. Bu sırada kontrol kapısı, bellek hücresinin içeriğini C_{t-1}'den C_t'ye günceller.

Çıkış kapısı, mevcut zaman adımında içsel durumun dışsal durumu ne ölçüde etkileyeceğini belirler.

Burada kullanılan:

- ⊗ sembolü, vektör elemanlarının bileşen bazında çarpımını,
- ⊕ sembolü ise vektörlerin toplamını ve ardından σ (sigma) aktivasyon fonksiyonunun uygulanmasını ifade eder.

LSTM yapay sinir ağının matematiksel formülasyonu için Denklemler (3.30), (3.31), (3.32), (3.33) ve (3.34) kullanılmaktadır .

$$f_t = \sigma(W_f . X_t + U_f . h_{t-1} + b_f) \quad (3.30)$$

$$O_t = \sigma(W_o . X_t + U_o . h_{t-1} + b_0) \quad (3.31)$$

$$\widetilde{C_t} = \tanh(W_c . X_t + U_c . h_{t-1} + b_c) \quad (3.32)$$

$$C_t = f_t . C_{t-1} + i_t . \widetilde{C_t} \quad (3.33)$$

$$h_t = \tanh(C_t) + O_t \quad (3.34)$$

Önerilen yaklaşım, özellik analizinde CNN sinir ağını ve oltalama ile yasal örneklerin sınıflandırılmasında LSTM ağını kullanmaktadır. Oltalama örneklerinin tespiti için kullanılan sinir ağı yapısı (Tablo 3.1)'de gösterilmiştir.

Tablo 3.1 CNN ve LSTM Hibrit Sinir Ağı Ayarları

Katman	**Tür**	**Çekirdek Boyutu (Kernel Size)**	**Adım (Stride)**	**Çekirdek Sayısı (Kernel)**	**Girdi Boyutu (Input Size)**
1	Convolution2D	3 × 3	1	64	224 × 224 × 3
2	Convolution2D	3 × 3	1	64	224 × 224 × 64
3	Pool	2 × 2	2	–	224 × 224 × 64
4	Convolution2D	3 × 3	1	128	112 × 112 × 64
5	Convolution2D	3 × 3	1	128	112 × 112 × 128
6	Pool	2 × 2	2	–	112 × 112 × 128
7	Convolution2D	3 × 3	1	256	56 × 56 × 128
8	Convolution2D	3 × 3	1	256	56 × 56 × 256
9	Pool	2 × 2	2	–	56 × 56 × 256
10	Convolution2D	3 × 3	1	512	28 × 28 × 256
11	Convolution2D	3 × 3	1	512	28 × 28 × 512
12	Convolution2D	3 × 3	1	512	28 × 28 × 512
13	Pool	2 × 2	2	–	28 × 28 × 512
14	Convolution2D	3 × 3	1	512	14 × 14 × 512
15	Convolution2D	3 × 3	1	512	14 × 14 × 512
16	Convolution2D	3 × 3	1	512	14 × 14 × 512
17	Pool	2 × 2	2	–	14 × 14 × 512
18	LSTM	–	–	–	49 × 512
19	FC	–	–	64	25,088
20	Output	–	–	3	64

4. SONUÇLAR VE TARTIŞMA

4.1 Arka Plan

Bu bölümde, önerilen oltalama saldırısı tespit yaklaşımı Python ve MATLAB ortamlarında uygulanmıştır. Veri kümesinin dengelenmesi ve derin öğrenme aşaması Python programlama dili ile gerçekleştirilmiş; özellik seçimi aşaması ise MATLAB 2023 yazılımında uygulanmıştır..

4.2 Veriseti

Bu araştırmada, URL kümesi Phishtank veritabanından toplanmış; testlerin tamamlanabilmesi için UCI ve Tan veri kümeleri kullanılmıştır. Önerilen yöntem, oltalama (phishing) ve yasal (legal) URL sayısını dengeler ve ardından CNN-LSTM modeli bu dengelenmiş veri kümesi üzerinde eğitilmektedir.

4.2. Değerlendirme Ölçütleri

Doğruluk (accuracy), duyarlılık (sensitivity) ve kesinlik (precision) gibi değerlendirme ölçütlerinin formülasyonu için Denklemler (4.35), (4.36) ve (4.37) kullanılmaktadır.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{4.35}$$

$$Sensitivity = Recall = \frac{TP}{TP + FN} \tag{4.36}$$

$$Precision = \frac{TP}{TP + FP} \tag{4.37}$$

- Doğruluk, duyarlılık ve kesinlik değerlerini hesaplamak için kullanılan TP (Doğru Pozitif), TN (Doğru Negatif), FP (Yanlış Pozitif) ve FN (Yanlış Negatif) parametreleri aşağıdaki şekilde tanımlanmıştır:
- TP (Doğru Pozitif): Örnek bir oltalama verisidir ve önerilen yöntem bu örneği doğru şekilde oltalama kategorisine sınıflandırmıştır.

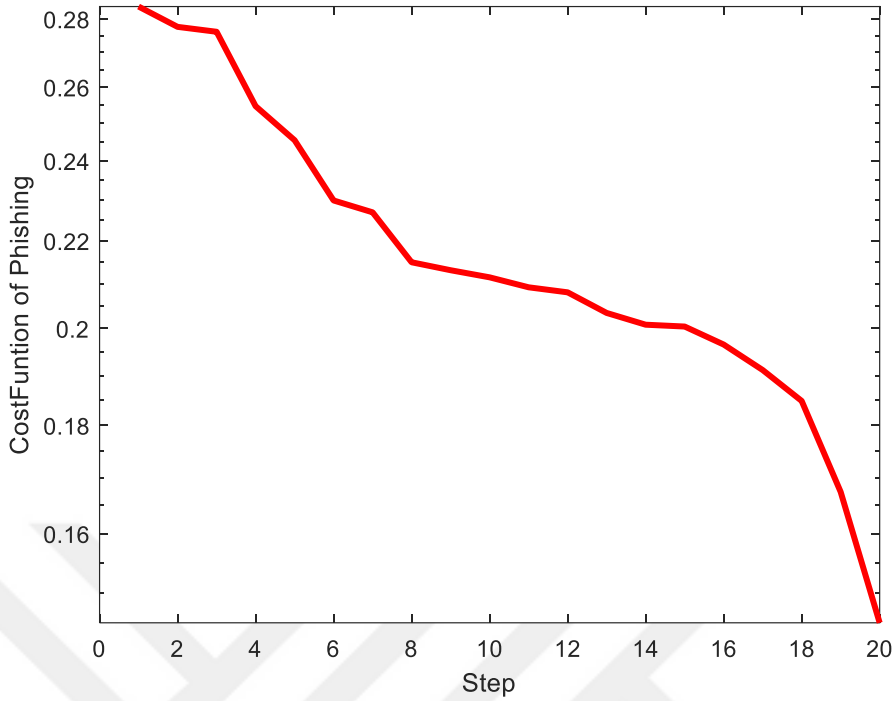
- FP (Yanlış Pozitif): Örnek yasal bir veridir; ancak önerilen yöntem bu örneği hatalı olarak oltalama kategorisine sınıflandırmıştır.
- TN (Doğru Negatif): Örnek bir yasal veridir ve önerilen yöntem bu örneği doğru şekilde yasal kategorisine sınıflandırmıştır.
- FN (Yanlış Negatif): Örnek bir oltalama verisidir; ancak önerilen yöntem bu örneği hatalı olarak yasal kategorisine sınıflandırmıştır.

4.3 Deneylerin Analizi

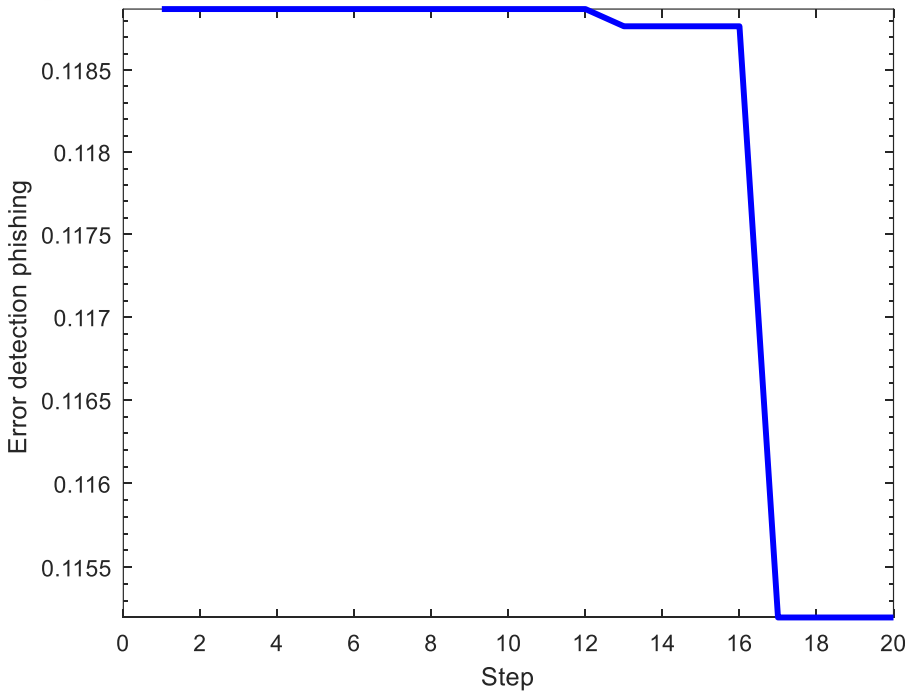
Bu bölümde, gerçekleştirilen testlerden bazıları sunulmaktadır. Önerilen yöntemdeki temel aşamalardan biri özellik seçimidir. Özellik seçimi, problem uzayının boyutunu azaltmakta ve CNN-LSTM eğitimi için temel özelliklerin belirlenmesini sağlamaktadır. Özellik seçimi aşamasında amaç fonksiyonunun değerindeki değişim süreci ile sınıflandırma hatası, sırasıyla (Şekil 4.1) ve (Şekil 4.2)'de gösterilmektedir. Seçim aşamasında gerçekleştirilen testler, önerilen yöntemin karakteristiklerini ortaya koymaktadır.

Deneylerin analizi, önerilen yaklaşımın AVOA algoritmasının yineleme sayısına bağlı olarak özellik seçimi amaç fonksiyonunu azalttığını göstermektedir. Özellik seçimi algoritması her yinelemede tekrarlandıkça, amaç fonksiyonunun değerindeki azalma aşağıdaki sonuçları ortaya koymaktadır:

- Optimal özellik vektörünün optimize edilmesiyle, oltalama saldırılarının tespitine ilişkin ortalama hata oranı azalmaktadır. Bununla birlikte, meta-sezgisel algoritmanın her yinelemesinde saldırı tespit doğruluğu artmaktadır.
- Saldırı tespitine ilişkin hata oranındaki azalma, özellik vektörlerinin optimize edilmesinin doğrudan bir sonucudur. Ayrıca, özellik vektörlerinin optimizasyon yoluyla iyileştirilmesi, önerilen yöntemin saldırı tespit etkinliğini daha da artırmaktadır.



Şekil 4.1 Özellik Seçimi Aşamasında Amaç Fonksiyonunun, Özellik Seçim Algoritmasının Yineleme Sayısına Göre Azalması

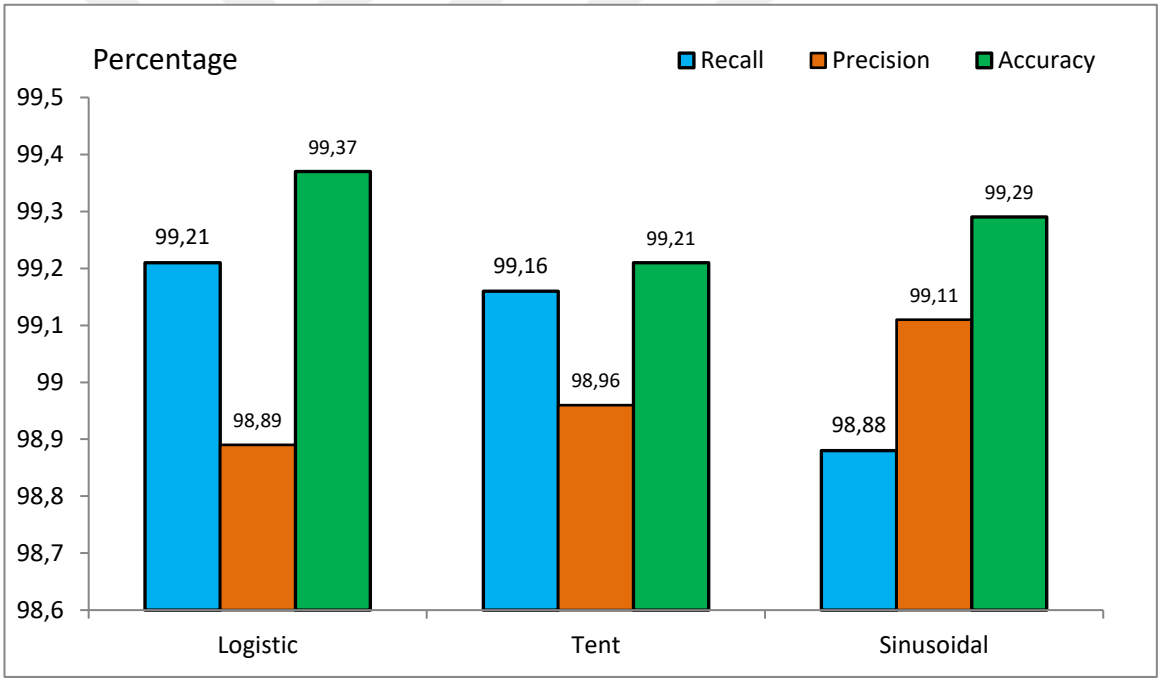


Şekil 4.2 Özellik Seçimi Algoritmasının Tekrarlanmasıyla, Özellik Seçimi Aşamasında Sınıflandırma Hatasının Azalması

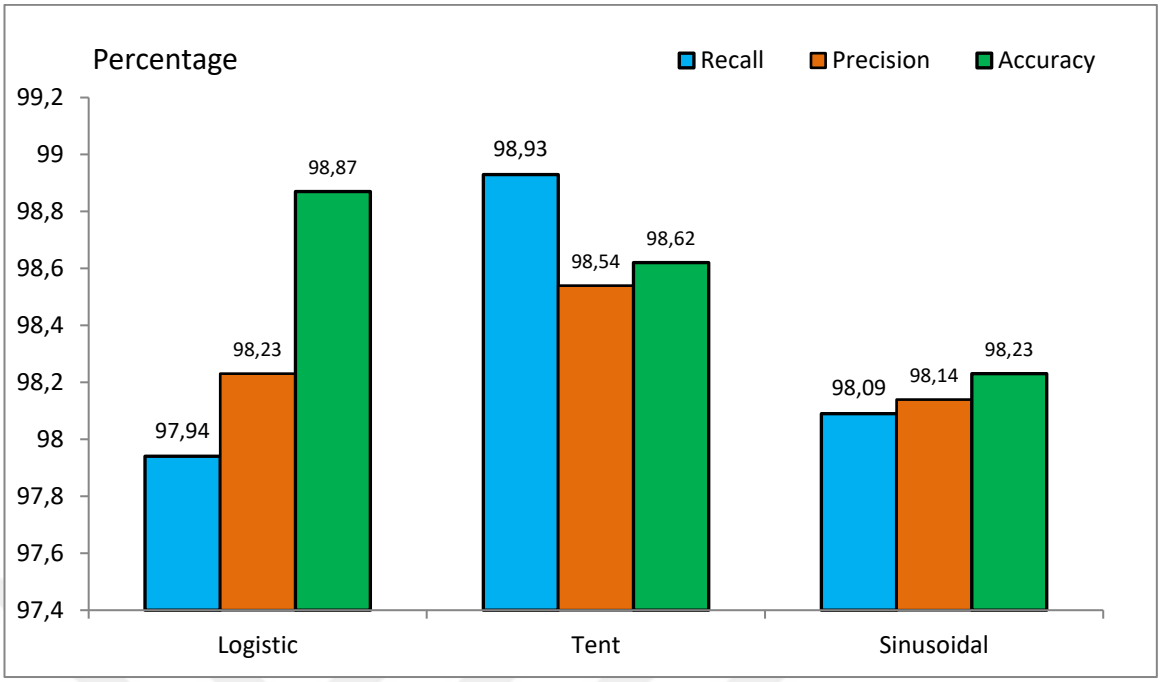
4.4 Değerlendirme ve Sonuçlar

Önerilen yöntem, yani GS-A-CL, oltalama saldırılarını tespit etmek amacıyla AVOA algoritmasının rastgele parametreleri için üç farklı kaotik fonksiyon (Logistic, Tent ve Sinusoidal) kullanmaktadır.

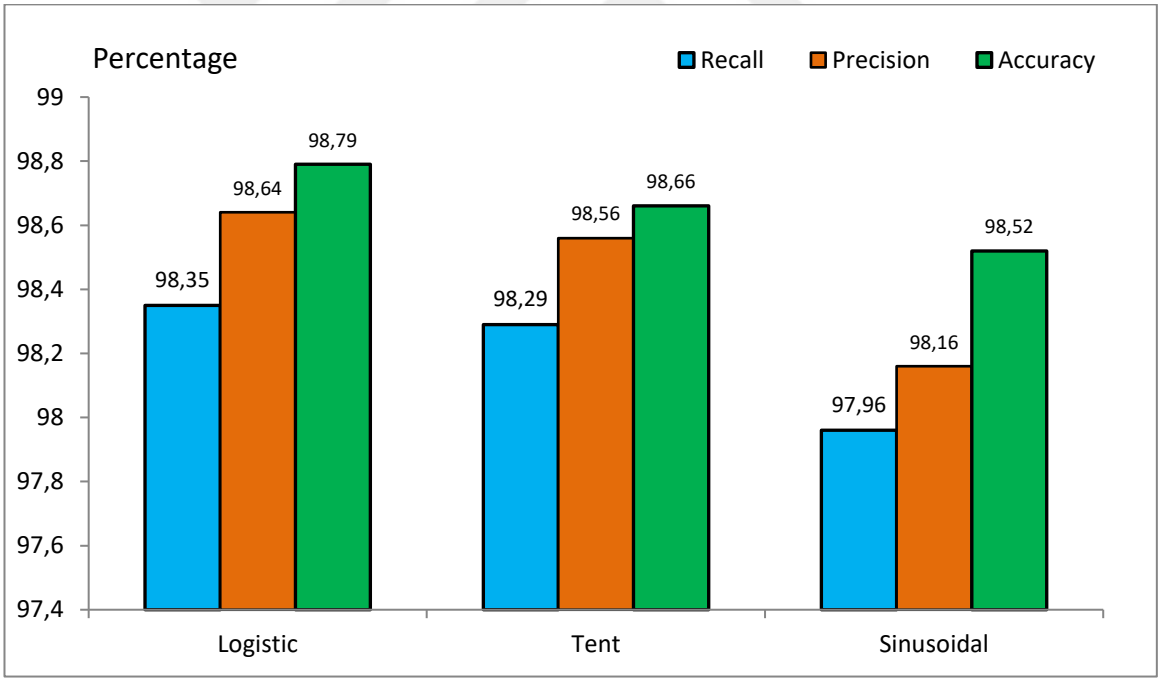
(Şekil 4.3), (4.4) ve (4.5) sırasıyla Phishtank, UCI ve Tan veri kümelerinde önerilen yaklaşımın kesinlik (precision), duyarlılık (sensitivity) ve doğruluk (correctness index) oranlarını göstermektedir.



Şekil 4.3 PhishTank Veri Kümesinde Önerilen Yöntemin Değerlendirme Ölçütlerinin Analizi



Şekil 4.4 UCI Veri Kümesinde Önerilen Yöntemin Değerlendirme Ölçütlerinin Analizi



Şekil 4.5 Tan Veri Kümesinde Önerilen Yöntemin Değerlendirme Ölçütlerinin Analizi

PhishTank veri kümesinde yapılan analiz, önerilen yöntemde Logistic kaotik fonksiyonunun kullanılması durumunda, kesinlik ve duyarlılık değerlerinin Tent ve Sinusoidal kaotik fonksiyonlarına göre daha yüksek olduğunu göstermektedir. Sinusoidal kaotik fonksiyonu kullanıldığında doğruluk oranı %99,11 olmaktadır.

Başka bir deyişle, PhishTank veri kümesinde en iyi durumda elde edilen doğruluk, kesinlik ve duyarlılık değerleri sırasıyla %99,37, %99,11 ve %99,21'dir.

UCI veri kümesinde, Logistic kaotik fonksiyonuna karşılık gelen doğruluk değeri en yüksek değerdir ve %98,87 olarak hesaplanmıştır. Tent kaotik fonksiyonu için duyarlılık ve doğruluk indeksleri maksimum değerlere sahiptir ve sırasıyla %98,93 ve %98,54'tür.

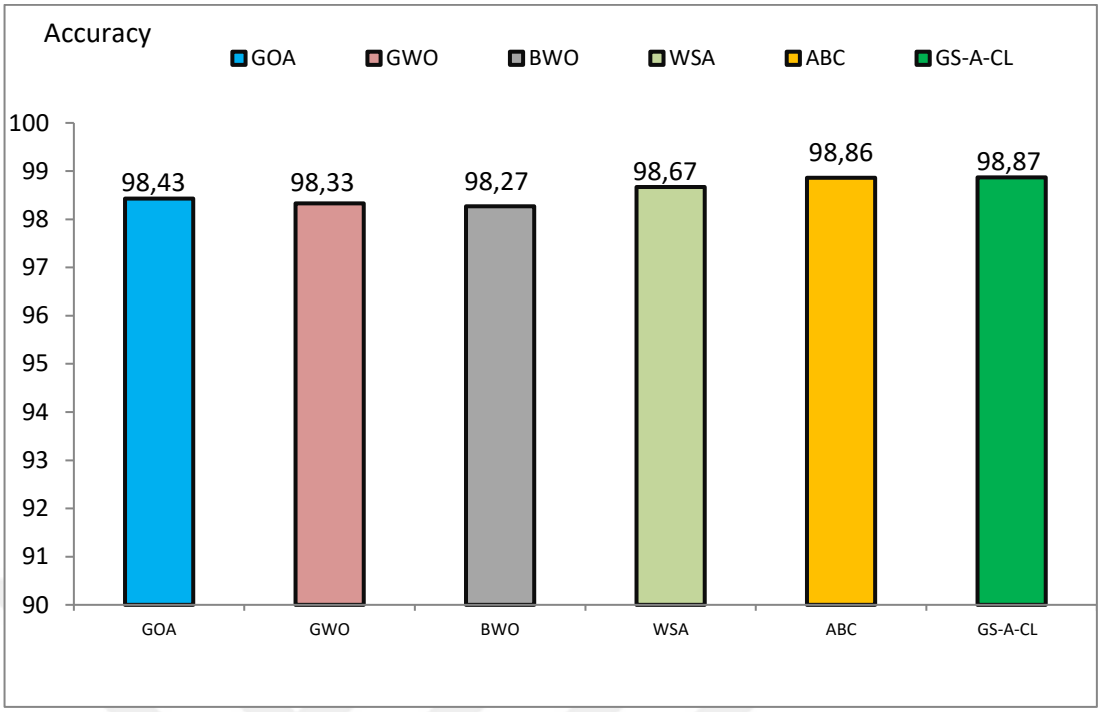
Tan veri kümesinde ise, Logistic kaotik fonksiyonuna göre doğruluk, kesinlik ve duyarlılık indeksleri sırasıyla %98,79, %98,64 ve %98,35'tir. Yapılan deneyler genel olarak, Logistic kaotik fonksiyonunun doğruluk, duyarlılık ve kesinlik ölçütleri açısından oltalama saldırılarının tespitinde diğer kaotik fonksiyonlara göre daha iyi performans gösterdiğini ortaya koymaktadır.

Önerilen yaklaşım, Tan veri kümesi üzerinde 2022 yılında yapılan (Almousa vd., 2022) numaralı çalışmanın sonuçlarıyla karşılaştırılmıştır (bkz. Tablo 4.1). Tan veri kümesi 49 öz nitelikten oluşmakta; bunlardan 48'i giriş, 49'uncu ise çıkış olarak kullanılmaktadır. Bu veri kümesinde önerilen yöntem %98,79 doğruluk, %98,64 kesinlik ve %98,35 duyarlılık değerleri elde etmiştir. Önerilen yöntem, LSTM-URL, LSTM-content, LSTM-all, FCnet-URL, FCnet-content, FCnet-all, CNN-URL, CNN-content ve CNN-all tekniklerine kıyasla oltalama saldırılarını tespit etmede daha yüksek doğruluk göstermektedir.

UCI veri kümesinde, önerilen yöntemin doğruluk değeri, referans (Tanha & Zarei, 2023)'de belgelenen GOA, WSA, ABC, BWO ve GWO gibi meta-sezgisel algoritmalarla karşılaştırılmıştır. Karşılaştırmalar, GOA, GWO, BWO, WSA ve ABC algoritmalarının doğruluk oranlarının sırasıyla %98,43, %98,33, %98,27, %98,67 ve %98.86 olduğunu ortaya koymuştur bkz. (Şekil 4.6). Önerilen yaklaşım, %98,89 doğruluk oranı ile bu meta-sezgisel yöntemlerin tümünden daha iyi performans göstermektedir. Meta-sezgisel algoritmalar arasında, arı optimizasyon algoritması (bee optimization method) %98,88 doğruluk oranı ile önerilen yaklaşıma çok yakın bir performans sergilemekte ancak önerilen yöntemden biraz daha düşük kalmaktadır.

Tablo 4.1 Tan Veri Kümesinde Önerilen Yöntemin Derin Öğrenme Yöntemleriyle Karşılaştırılması

Yöntem	Doğruluk (Accuracy)	Kesinlik (Precision)	Duyarlılık (Sensitivity)
LSTM-URL	90,40	91,45	89,13
LSTM-content	96,00	94,71	97,93
LSTM-all	97,37	96,71	98,07
FCnet-URL	90,47	89,41	91,80
FCnet-content	96,00	96,52	96,20
FCnet-all	96,77	96,36	97,20
CNN-URL	90,70	87,85	94,47
CNN-content	96,43	96,47	96,40
CNN-all	97,27	96,64	97,93
GS-A-CL	**98,79**	**98,64**	**98,35**

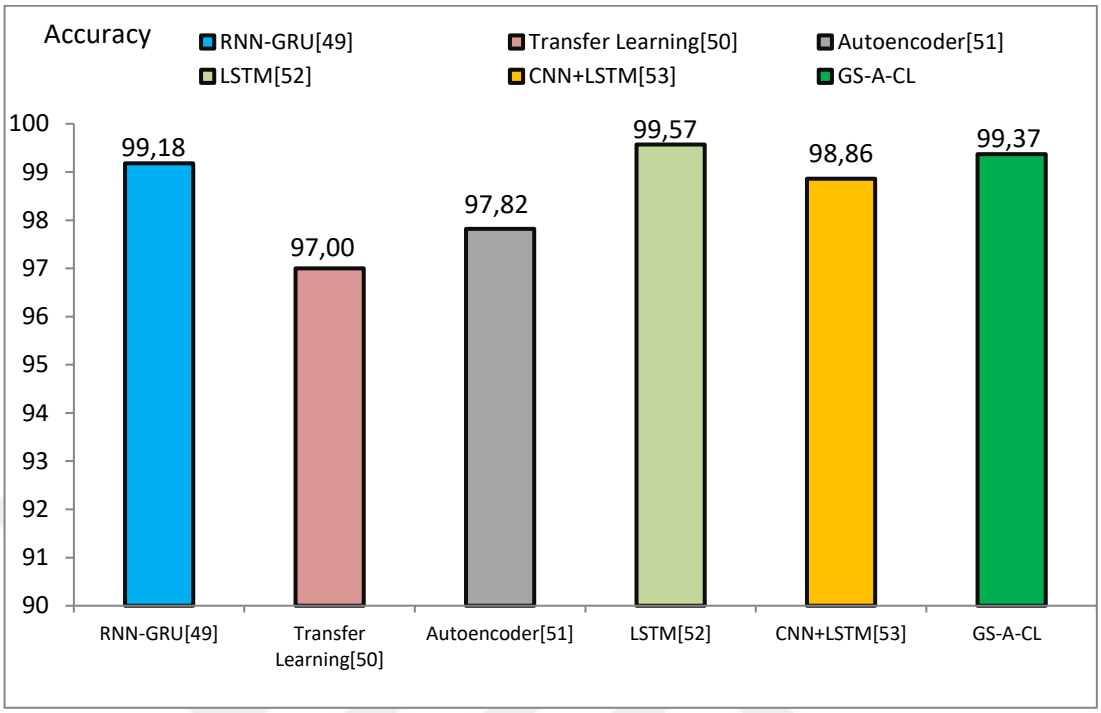


Şekil 4.6 UCI Veri Kümesinde Önerilen Yöntemin Özellik Seçim Yöntemleriyle Doğruluk Açısından Karşılaştırılması

UCI veri kümesi üzerinde yapılan karşılaştırmalı analizler, önerilen GS-A-CL yönteminin, özellikle dengesiz veri kümeleri üzerinde daha yüksek doğruluk sağladığını ortaya koymaktadır. UCI veri kümesi dengesiz yapıda olduğundan, önerilen yöntemin veri dengeleme aşaması, modelin performansını önemli ölçüde artırmıştır.

Buna karşın, Tan veri kümesi dengeli bir yapıya sahip olduğundan, bu veri kümesinde dengeleme aşamasının sonuçlara etkisi görece daha düşüktür. Bu nedenle, önerilen yöntemin başarısında öne çıkan unsur özellik seçimi ve derin öğrenme mimarisinin optimizasyonudur.

PhishTank veri kümesi üzerinde önerilen yöntemle yapılan deney sonuçları ise, çeşitli derin öğrenme tabanlı çalışmalara göre üstün performans gösterdiğini ortaya koymaktadır. Bu karşılaştırmalı sonuçlar (Şekil 4.7)'de görsel olarak sunulmuştur. Grafik, önerilen yöntemin farklı veri kümelerinde sağladığı doğruluk avantajını, özellikle meta-sezgisel özellik seçimi stratejisi ve CNN-LSTM entegrasyonuyla desteklendiğini göstermektedir.



Şekil 4.7 PhishTank Veri Kümesinde Önerilen Yöntemin Derin Öğrenme Yöntemleriyle Doğruluk Açısından Karşılaştırılması

(Şekil 4.7)'deki diyagram, önerilen yöntemi RNN-GRU, Transfer Learning, Autoencoder, LSTM ve CNN-LSTM yöntemleriyle karşılaştırmaktadır. RNN-GRU, Transfer Learning, Autoencoder, LSTM ve CNN-LSTM yöntemlerinin doğruluk oranları sırasıyla %99,18, %97, %97,82, %99,57 ve %98,86'dır. Önerilen yöntem, (Adebowale vd., 2023) numaralı çalışmada sunulan araştırmaya kıyasla daha düşük bir doğruluk oranı sergilemektedir; ancak diğer derin öğrenme yöntemleriyle karşılaştırıldığında saldırıları tespit etmede üstün bir performans göstermektedir. Önerilen yöntem, oltalama URL'lerini tespit etmede (Somesha vd., 2020) numaralı araştırmaya göre biraz daha zayıf olmakla birlikte, UCI ve Tan veri kümelerinde LSTM yönteminden daha yüksek doğruluk sağlamaktadır. (Tablo 4.2), önerilen yöntemin oltalama saldırılarını tespit etme performansını çeşitli derin öğrenme ve optimizasyon yöntemleriyle karşılaştırmaktadır (Adebowale vd., 2023).

Tablo 4.2 Önerilen Yöntemin Derin Öğrenme ve Meta-Sezgisel Yöntemlerle Karşılaştırılması

Yöntem (Methods)	Doğruluk (Accuracy)	Kesinlik (Precision)	Duyarlılık (Recall)
ODAE-WPDC	99,28	99,29	99,24
DL-SGD	94,64	94,97	95,17
DL-RMSProp	92,84	93,77	95,34
DL-Adam	94,69	94,87	95,93
SI-BBA	94,93	94,59	94,84
PDGAN	94,12	94,96	94,02
NIOSELM	93,40	94,65	94,66
MLP-SL	87,80	88,75	87,41
SVM-SL	83,37	87,22	88,54
GS-A-CL	99,37	99,11	99,21

Önerilen yaklaşım, oltalama URL'lerinin tespitinde %99,37 doğruluk, %99,11 kesinlik ve %99,21 duyarlılık elde etmektedir. Deneysel sonuçlar, önerilen yöntemin doğruluğunun ODAE-WPDC, DL-SGD, DL-RMSProp, DL-Adam, SI-BBA, PDGAN, NIOSELM, MLP-SL ve SVM-SL gibi yöntemlerden daha yüksek olduğunu göstermektedir.

Veri kümesinin dengelenmesi sayesinde, önerilen yöntem oltalama örneklerini tespit etme ve sınıflandırma konusunda yüksek başarı sağlamaktadır. Ayrıca, önerilen yaklaşım ODAE-WPDC yöntemine göre daha hızlıdır; çünkü ODAE-WPDC'nin aksine, parametreleri optimize etmek için meta-sezgisel algoritmalar kullanılmamaktadır.

Bunun yerine, önerilen yöntem CNN-LSTM sinir ağının giriş boyutunu ve veri boyutunu azaltmak amacıyla meta-sezgisel algoritmalar kullanmakta ve bu sayede öğrenme hızını artırmaktadır. Bu strateji, hem yüksek doğruluk hem de daha hızlı eğitim süresi elde edilmesine katkı sağlamaktadır.

5. SONUÇLAR VE GELECEK ÇALIŞMALAR

5.1 Sonuç

Oltalama (phishing), internet kullanıcılarına karşı aldatma ve sosyal mühendislik yoluyla gerçekleştirilen bir siber saldırıdır. Bu saldırılarda, sahte bağlantılar e-posta veya diğer iletişim araçları aracılığıyla kullanıcılara gönderilir ve kullanıcılar sahte web sitelerine yönlendirilir. Oltalama saldırıları oldukça zararlıdır ve yapılan tahminler bu saldırıların milyonlarca dolarlık kullanıcı kaybına yol açtığını göstermektedir. Bu tür saldırıların tespiti için pratik bir yaklaşım, makine öğrenmesi ve derin öğrenme yöntemlerinin kullanılmasıdır. CNN sinir ağı, örüntü analizi ve tanıma için etkili bir derin öğrenme aracıdır. Genellikle görüntü işleme için kullanılan CNN sinir ağı, bazı çalışmalarda metin ve URL dizilerini sınıflandırmak için de kullanılmıştır.

Önerilen oltalama tespit yönteminde, örnekler öncelikle dengelenmiştir. Örnek dengelemesi için olasılıksal GAN versiyonu, SMOTE yöntemiyle birleştirilmiş ve böylece dengelenmiş veri kümesi üzerinde derin öğrenme gerçekleştirilerek öğrenme hatası azaltılmıştır. Özellik çıkarımı için N-Gram yöntemi kullanılmış; temel özellikler ise kaos teorisiyle geliştirilmiş AVOA algoritmasıyla seçilmiştir.

Seçilen özellikler veri kümesine uygulanmış ve seçilen örnekler CNN için giriş görüntülerine dönüştürülmüştür. Son aşamada, oltalama sayfalarının ve bağlantılarının tespiti için CNN ve LSTM'nin birleşik ve optimal bir yaklaşımı sunulmuştur. Deneyler PhishTank, UCI ve Tan veri kümeleri üzerinde gerçekleştirilmiştir. Önerilen yöntemin bu veri kümeleri üzerindeki doğruluk oranları sırasıyla: UCI için %98,87, PhishTank için %99,37 ve Tan için %98,79'dur. Deneyler, AVOA'nın arama yeteneğini artırmak için Logistic kaotik fonksiyonunun daha iyi performans gösterdiğini ortaya koymaktadır. Önerilen yaklaşım, özellikle özellik seçimi aşamasında WSA, BWO ve ABC algoritmalarına göre daha yüksek doğruluk sergilemektedir. Ayrıca önerilen yöntem, örnek sınıflandırmasında CNN, RNN-GRU ve Autoencoder yöntemlerini geride bırakmaktadır.

Önerilen yöntemin avantajları şunlardır: veri kümesini iki farklı yöntemle dengeleyerek daha dengeli bir veri kümesi oluşturulması; koşullu olasılıklara dayalı oyun teorisinin GAN yönteminden daha doğru sonuçlar üretmesi; özellik seçimi aşamasında kaos teorisiyle geliştirilmiş Afrika akbabası algoritmasının kullanılması; ve CNN ile LSTM mimarilerinin entegrasyonu sayesinde oltalama saldırılarının tespitinde sınıflandırma doğruluğunun artırılmasıdır.

Ancak önerilen yöntemin bir zorluğu, CNN-LSTM yöntemiyle dengeleme ve öğrenme aşamalarında yüksek karmaşıklık ve zaman maliyetidir; ayrıca LSTM ve CNN parametrelerinin optimize edilmesi gerekmektedir. Gelecekteki çalışmalarda, CNN-LSTM parametreleri meta-sezgisel algoritmalarla optimize edilecektir. Bir diğer gelecek çalışma ise, oltalama saldırılarını tespit etmek için Google Chrome gibi tarayıcılar için bir eklenti geliştirilmesidir.

5.2 Gelecek Çalışmalar

Her ne kadar önerilen GS-A-CL çerçevesi, IoT ortamlarında oltalama saldırılarını tespit etmede yüksek doğruluk ve sağlamlık göstermiş olsa da, gelecekte keşfedilebilecek birçok umut vadeden yön bulunmaktadır:

1. Gerçek Zamanlı Sistemlere Genişletme:

Gelecekteki çalışmalar, önerilen modelin canlı IoT ortamları için gerçek zamanlı saldırı tespit sistemlerine (IDS) uygulanmasına odaklanacaktır. Bu, modelin çıkarım süresinin optimize edilmesini ve gömülü cihazlara uygun hafif çerçevelerle entegrasyonunu içerecektir.

2. Diğer Saldırı Türlerine Genişletme:

Mevcut model, oltalama ve DDoS saldırılarını tespit etmeye yöneliktir. Gelecekte, modelin botnet, kötü amaçlı yazılım yayılımı ve içeriden gelen saldırılar gibi diğer siber tehdit türlerini tespit edecek şekilde genişletilmesi planlanmaktadır.

3. Yorumlanabilirliğin Artırılması:

Derin öğrenme tabanlı hibrit model iyi performans sergilese de yorumlanabilirliği düşüktür. Gelecekteki çalışmalar, tespit sonuçlarını ağ yöneticileri ve siber güvenlik

uzmanları için daha anlaşılır hale getirmek amacıyla açıklanabilir yapay zekâ (XAI) tekniklerini araştıracaktır.

4. Federated ve Transfer Learning Uygulamaları:

Federated learning'in uygulanması, ham veriyi paylaşmadan çoklu IoT cihazlarında dağıtık eğitim yapılmasına olanak tanıyarak gizliliği artıracaktır. Transfer learning ise modelin yeni ortamlara veya saldırı türlerine minimum eğitimle uyarlanmasını sağlayabilir.

5. Dinamik Özellik Uyarlaması:

Gelecekteki araştırmalar, gerçek zamanlı veri akışlarına dayalı olarak zaman içinde evrilen adaptif özellik seçimi yöntemlerini keşfedebilir. Bu, sistemin yeni veya mutasyona uğramış tehditlere karşı etkili kalmasını sağlayacaktır.

6. Çok Katmanlı Ansambl Mimariler:

Modelin, CNN-LSTM ile GRU veya Transformer tabanlı ağların yığılması gibi derin öğrenme ansambl teknikleriyle güçlendirilmesi, farklı veri kümeleri arasında daha yüksek dayanıklılık ve genelleme kabiliyeti sağlayabilir.

7. Blokzincir Tabanlı Günlükleme ile Entegrasyon:

Tespit günlüklerinin bütünlüğünü ve inkâr edilemezliğini sağlamak amacıyla, tespit sisteminin blokzincir teknolojisiyle entegre edilmesi gelecekteki sürümlerde araştırılabilir.

8. Enerji ve Kaynak Optimizasyonu:

IoT ortamları sınırlı kaynaklara sahip olduğundan, gelecekteki çalışmalar GS-A-CL modelinin doğruluk performansından ödün vermeden enerji tüketimini azaltmaya ve kaynak kullanımını optimize etmeye odaklanacaktır.

.

KAYNAKLAR

Ab-BelKhair, A., Rahebi, J., & Abdulhamed Mohamed Nureddin, A. (2020). A Study of Deep Neural Network Controller-Based Power Quality Improvement of Hybrid PV/Wind Systems by Using Smart Inverter. *International Journal of Photoenergy*, *2020*.

Abdollahzadeh, B., Gharehchopogh, F. S., & Mirjalili, S. (2021). African vultures optimization algorithm: A new nature-inspired metaheuristic algorithm for global optimization problems. *Computers & Industrial Engineering*, *158*, 107408.

Abdulghani Ali Ahmed, A. A. A., Ali Al-Bayatti, A. A.-B., Mubarak Saif, M. S., Jabbar, W. A., Jabbar, W. A., Rassem, T. H., & Rassem, T. H. (2023). A Honeybee-Inspired Framework for a Smart City Free of Internet Scams. *Sensors*, *23*(4284), 1–14.

Abdullah, A. S., Özok, Y. E., & Rahebi, J. (2018). A novel method for retinal optic disc detection using bat meta-heuristic algorithm. *Medical & Biological Engineering & Computing*, *56*(11), 2015–2024.

Abdullah, A. S., Rahebi, J., Özok, Y. E., & Aljanabi, M. (2020). A new and effective method for human retina optic disc segmentation with fuzzy clustering method based on active contour model. *Medical & Biological Engineering & Computing*, *58*(1), 25–37.

Abdulrahman, L. M., Ahmed, S. H., Rashid, Z. N., Jghef, Y. S., Ghazi, T. M., & Jader, U. H. (2023). Web Phishing Detection Using Web Crawling, Cloud Infrastructure and Deep Learning Framework. *Journal of Applied Science and Technology Trends*, *4*(01), 54–71.

Abed, A. H., Rahebi, J., & Farzamnia, A. (2017). Improvement for power quality by using dynamic voltage restorer in electrical distribution networks. In *2017 IEEE 2nd International Conference on Automatic Control and Intelligent Systems (I2CACIS)* (pp. 122–127). IEEE.

Adebowale, M. A., Lwin, K. T., & Hossain, M. A. (2023). Intelligent phishing detection scheme using deep learning algorithms. *Journal of Enterprise Information Management*, *36*(3), 747–766.

Al Shalchi, N. F. A., & Rahebi, J. (2022). Human retinal optic disc detection with grasshopper optimization algorithm. *Multimedia Tools and Applications*, 1–19.

Alabandi, G. A. (2017). Combining Deep Learning with Traditional Machine Learning to Improve Classification Accuracy on Small Datasets.

Aljanabi, M., Özok, Y. E., Rahebi, J., & Abdullah, A. S. (2018). Skin lesion segmentation method for dermoscopy images using artificial bee colony algorithm. *Symmetry*, *10*(8), 347.

Almousa, M., Zhang, T., Sarrafzadeh, A., & Anwar, M. (2022). Phishing website detection: How effective are deep learning-based models and hyperparameter optimization? *Security and Privacy*, *5*(6), e256.

Alsenani, T. R., Ayon, S. I., Yousuf, S. M., Anik, F. B. K., & Chowdhury, M. E. S. (2023). Intelligent feature selection model based on particle swarm optimization to detect phishing websites. *Multimedia Tools and Applications*, 1–33.

Atlam, H. F., & Oluwatimilehin, O. (2022). Business Email Compromise Phishing Detection Based on Machine Learning: A Systematic Literature Review. *Electronics*, *12*(1), 42.

Bozkir, A. S., Dalgic, F. C., & Aydos, M. (2023). GramBeddings: a new neural network for URL based identification of phishing web pages through n-gram embeddings. *Computers & Security*, *124*, 102964.

Bu, S.-J., & Cho, S.-B. (2021). Integrating deep learning with first-order logic programmed constraints for zero-day phishing attack detection. In *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (pp. 2685–2689). IEEE.

Buono, P., Desolda, G., Greco, F., & Piccinno, A. (2023). Let warnings interrupt the interaction and explain: designing and evaluating phishing email warnings. In *Extended Abstracts of the 2023 CHI Conference on Human Factors in Computing Systems* (pp. 1–6).

Burgess, J., O'Kane, P., Sezer, S., & Carlin, D. (2021). LSTM RNN: detecting exploit kits using redirection chain sequences. *Cybersecurity*, *4*(1), 1–15.

da Silva, C. M. R., Feitosa, E. L., & Garcia, V. C. (2020). Heuristic-based strategy for Phishing prediction: A survey of URL-based approach. *Computers & Security*, *88*, 101613.

Darwassh Hanawy Hussein, T., Frikha, M., Ahmed, S., & Rahebi, J. (2022). BA-CNN: Bat Algorithm-Based Convolutional Neural Network Algorithm for Ambulance Vehicle Routing in Smart Cities. *Mobile Information Systems, 2022*.

Divakaran, D. M., & Oest, A. (2022). Phishing detection leveraging machine learning and deep learning: A review. *ArXiv Preprint ArXiv:2205.07411*.

Elshara, R., Hançerlioğullari, A., Rahebi, J., & Lopez-Guede, J. M. (2024). PV Cells and Modules Parameter Estimation Using Coati Optimization Algorithm. *Energies*, *17*(7), 1716.

Farida, F., & Mustopa, A. (2023). Comparison of Logistic Regression and Random Forest using Correlation-based Feature Selection for Phishing Website Detection. *Sistemasi: Jurnal Sistem Informasi*, *12*(1), 13–20.

Feng, S., Keung, J., Zhang, P., Xiao, Y., & Zhang, M. (2022). The impact of the distance metric and measure on SMOTE-based techniques in software defect prediction. *Information and Software Technology*, *142*, 106742.

Greco, F., Desolda, G., & Esposito, A. (2023). Explaining phishing attacks: An XAI approach to enhance user awareness and trust. In *Proc. of the Italian Conference on CyberSecurity (ITASEC '23)*.

Guo, Y. (2023). A review of Machine Learning-based zero-day attack detection: Challenges and future directions. *Computer Communications, 198*, 175–185.

Gupta, A., Choudhary, G., Shandilya, S. K., & Sihag, V. (2023). A contemporary anti-phishing framework based on visual cryptography and steganography. *International Journal of Internet Technology and Secured Transactions*, *13*(2), 139–158.

Gupta, B. B., Tewari, A., Jain, A. K., & Agrawal, D. P. (2017). Fighting against phishing attacks: state of the art and future challenges. *Neural Computing and Applications*, *28*, 3629–3654.

Gupta, S., & Bansal, H. (2023). Trust evaluation of health websites by eliminating phishing websites and using similarity techniques. *Concurrency and Computation: Practice and Experience*, e7695.

Hameed, O. H. H., Kutbay, U., Rahebi, J., & Hardalaç, F. (2023). Fault Classification for Protection in MMC-HVDC Using Machine Learning Algorithms. In *2023 IEEE 3rd Mysore Sub Section International Conference (MysuruCon)* (pp. 1–4). IEEE.

Hazim Hameed Hameed, O., Kutbay, U., Rahebi, J., Hardalaç, F., & Mahariq, I. (2024). Enhancing Fault Detection and Classification in MMC-HVDC Systems: Integrating Harris Hawks Optimization Algorithm with Machine Learning Methods. *International Transactions on Electrical Energy Systems, 2024*.

He, S., Ruan, J., Long, Y., Wang, J., Wu, C., Ye, G., … Zhang, Y. (2018). Combining deep learning with traditional features for classification and segmentation of pathological images of breast cancer. In *2018 11th International Symposium on Computational Intelligence and Design (ISCID)* (Vol. 1, pp. 3–6). IEEE.

Hindy, H., Atkinson, R., Tachtatzis, C., Colin, J.-N., Bayne, E., & Bellekens, X. (2020). Utilising deep learning techniques for effective zero-day attack detection. *Electronics*, *9*(10), 1684.

Hussein, T. D. H., Frikha, M., Ahmed, S., & Rahebi, J. (2022). Ambulance Vehicle Routing in Smart Cities Using Artificial Neural Network. In *2022 6th International Conference on Advanced Technologies for Signal and Image Processing (ATSIP)* (pp. 1–6). IEEE.

Jain, A. K., & Gupta, B. B. (2016). A novel approach to protect against phishing attacks at client side using auto-updated white-list. *EURASIP Journal on Information Security, 2016*, 1–11.

Jha, B., Atre, M., & Rao, A. (2022). Detecting Cloud-Based Phishing Attacks by Combining Deep Learning Models. In *2022 IEEE 4th International Conference on Trust, Privacy and Security in Intelligent Systems, and Applications (TPS-ISA)*

(pp. 130–139). IEEE.

Kalabarige, L. R., Rao, R. S., Abraham, A., & Gabralla, L. A. (2022). Multilayer stacked ensemble learning model to detect phishing websites. *IEEE Access*, *10*, 79543–79552.

Kaushik, K., Singh, S., Garg, S., Singhal, S., & Pandey, S. (2021). Exploring the mechanisms of phishing. *Computer Fraud & Security*, *2021*(11), 14–19.

Kim, J., Lee, S., Kim, Y., Ahn, S., & Cho, S. (2023). Graph Learning-Based Blockchain Phishing Account Detection with a Heterogeneous Transaction Graph. *Sensors*, *23*(1), 463.

Kingma, D. P., & Ba, J. (2014). Adam: A method for stochastic optimization. *ArXiv Preprint ArXiv:1412.6980*.

Lee, Y. Y., Gan, C. L., & Liew, T. W. (2023). Susceptibility to instant messaging phishing attacks: does systematic information processing differ between genders? *Crime Prevention and Community Safety*, *25*(2), 179–203.

Lin, S.-C., Wl, P.-C., Chen, H.-Y., Morikawa, T., Takahashi, T., & Lin, T.-N. (2022). Senseinput: An image-based sensitive input detection scheme for phishing website detection. In *ICC 2022-IEEE International Conference on Communications* (pp. 4180–4186). IEEE.

Liu, D.-J., Geng, G.-G., & Zhang, X.-C. (2022). Multi-scale semantic deep fusion models for phishing website detection. *Expert Systems with Applications*, *209*, 118305.

Marin, I. A., Burda, P., Zannone, N., & Allodi, L. (2023). The Influence of Human Factors on the Intention to Report Phishing Emails. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems* (pp. 1–18).

Memiş, E., Akarkamçı, H., Yeniad, M., Rahebi, J., & Lopez-Guede, J. M. (2024). Comparative Study for Sentiment Analysis of Financial Tweets with Deep Learning Methods. *Applied Sciences*, *14*(2), 588.

Mohanty, S., & Acharya, A. A. (2023). MFBFST: Building a stable ensemble learning model using multivariate filter-based feature selection technique for detection of suspicious URL. *Procedia Computer Science*, *218*, 1668–1681.

Nureddin, A. A. M., Rahebi, J., & Ab-BelKhair, A. (2020). Power Management Controller for Microgrid Integration of Hybrid PV/Fuel Cell System Based on Artificial Deep Neural Network. *International Journal of Photoenergy*, *2020*.

Pandey, M. K., Singh, M. K., Pal, S., & Tiwari, B. B. (2023). Prediction of phishing websites using machine learning. *Spatial Information Research*, *31*(2), 157–166.

Pham, T. D., Pham, T. T. T., Hoang, S. T., & Ta, V. C. (2021). Exploring efficiency of GAN-based generated URLs for phishing URL detection. In *2021 International Conference on Multimedia Analysis and Pattern Recognition*

(MAPR) (pp. 1–6). IEEE.

Rahebi, J. (2011). Biomedical Image Edge Detection using an Ant Colony Optimization Based on Artificial Neural Networks. *International Journal of Engineering Science*. Retrieved from http://www.ijest.info/docs/IJEST11-03-12-193.pdf%5Cnpapers2://publication/uuid/78D2C20B-3EC5-4019-815F-8DD37FB73A15

Rahebi, Javad, & Hardalaç, F. (2014). Retinal blood vessel segmentation with neural network by using gray-level co-occurrence matrix-based features. *Journal of Medical Systems*, *38*(8), 85. https://doi.org/10.1007/s10916-014-0085-2

Rahebi, Javad, & Hardalaç, F. (2016). A new approach to optic disc detection in human retinal images using the firefly algorithm. *Medical and Biological Engineering and Computing*, *54*(2–3), 453–461. https://doi.org/10.1007/s11517-015-1330-7

Rajeswary, C., & Thirumaran, M. (2023). A Comprehensive Survey of Automated Website Phishing Detection Techniques: A Perspective of Artificial Intelligence and Human Behaviors. In *2023 International Conference on Sustainable Computing and Data Communication Systems (ICSCDS)* (pp. 420–427). IEEE.

Remmide, M. A., Boumahdi, F., Boustia, N., Feknous, C. L., & Della, R. (2022). Detection of phishing URLs using temporal convolutional network. *Procedia Computer Science*, *212*, 74–82.

Shaik, C. M., Penumaka, N. M., Abbireddy, S. K., Kumar, V., & Aravinth, S. S. (2023). Bi-LSTM and Conventional Classifiers for Email Spam Filtering. In *2023 Third International Conference on Artificial Intelligence and Smart Energy (ICAIS)* (pp. 1350–1355). IEEE.

Sharma, A., Singh, P. K., & Chandra, R. (2022). SMOTified-GAN for class imbalanced pattern classification problems. *Ieee Access*, *10*, 30655–30665.

Sharma, S. R., Singh, B., & Kaur, M. (2022). Improving the classification of phishing websites using a hybrid algorithm. *Computational Intelligence*, *38*(2), 667–689.

Soltani, M., Ousat, B., Siavoshani, M. J., & Jahangir, A. H. (2023). An adaptable deep learning-based Intrusion Detection System to zero-day attacks. *Journal of Information Security and Applications*, *76*, 103516.

Somesha, M., Pais, A. R., Rao, R. S., & Rathour, V. S. (2020). Efficient deep learning techniques for the detection of phishing websites. *Sādhanā*, *45*, 1–18.

Sun, Youngghyu, Lee, J., Kim, S., Seon, J., Lee, S., Kyeong, C., & Kim, J. (2023). Energy Theft Detection Model Based on VAE-GAN for Imbalanced Dataset. *Energies*, *16*(3), 1109.

Sun, Yuwei, Chong, N., & Ochiai, H. (2022). Federated Phish Bowl: LSTM-Based Decentralized Phishing Email Detection. In *2022 IEEE International Conference on Systems, Man, and Cybernetics (SMC)* (pp. 20–25). IEEE.

Syafitri, W., Shukur, Z., Asma'Mokhtar, U., Sulaiman, R., & Ibrahim, M. A. (2022). Social engineering attacks prevention: A systematic literature review. *IEEE Access*, *10*, 39325–39343.

Tanha, J., & Zarei, Z. (2023). The Bombus-terrestris bee optimization algorithm for feature selection. *Applied Intelligence*, *53*(1), 470–490.

Thirumaran, M., Karthikeyan, R. P., & Rathaamani, V. (2023). Phishing Website Detection Using Natural Language Processing and Deep Learning Algorithm. *Advances in Science and Technology*, *124*, 712–718.

Trinh, N. B., Phan, T. D., & Pham, V.-H. (2022). Leveraging Deep Learning Image Classifiers for Visual Similarity-based Phishing Website Detection. In *Proceedings of the 11th International Symposium on Information and Communication Technology* (pp. 134–141).

Xie, J., Jiang, H., Song, W., & Yang, J. (2023). A novel quality control method of time-series ocean wave observation data combining deep-learning prediction and statistical analysis. *Journal of Sea Research*, *195*, 102439.

Xu, T., Singh, K., & Rajivan, P. (2023). Personalized persuasion: Quantifying susceptibility to information exploitation in spear-phishing attacks. *Applied Ergonomics*, *108*, 103908.

Yoo, J., & Cho, Y. (2022). ICSA: Intelligent chatbot security assistant using Text-CNN and multi-phase real-time defense against SNS phishing attacks. *Expert Systems with Applications*, *207*, 117893.