DÜZCE ÜNİVERSİTESİ | LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ

ZARARLI URL'LERİN DERİN ÖĞRENME İLE TESPİTİ

FATİH TİRYAKİ

YÜKSEK LİSANS TEZİ

SİBER GÜVENLİK ANABİLİM DALI

DANIŞMAN

PROF. DR. İBRAHİM YÜCEDAĞ

DÜZCE, 2024

T.C.

DÜZCE ÜNİVERSİTESİ
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ
ZARARLI URL'LERİN DERİN ÖĞRENME İLE TESPİTİ

Fatih TİRYAKİ tarafından hazırlanan tez çalışması aşağıdaki jüri tarafından Düzce Üniversitesi Lisansüstü Eğitim Enstitüsü Siber Güvenlik Anabilim Dalı'nda **YÜKSEK LİSANS TEZİ** olarak kabul edilmiştir.

Tez Danışmanı

Prof. Dr. İbrahim YÜCEDAĞ

Düzce Üniversitesi

Eş Danışman

Dr. Öğr. Üyesi Ümit ŞENTÜRK

Bolu Abant İzzet Baysal Üniversitesi

Jüri Üyeleri

Prof. Dr. İbrahim YÜCEDAĞ

Düzce Üniversitesi _____

Doç. Dr. Abdullah Talha KABAKUŞ

Düzce Üniversitesi _____

Doç. Dr. Nihat DALDAL

Bolu Abant İzzet Baysal Üniversitesi _____

Tez Savunma Tarihi: 22/01/2024

BEYAN

Bu tez çalışmasının kendi çalışmam olduğunu, tezin planlanmasından yazımına kadar bütün aşamalarda etik dışı davranışımın olmadığını, bu tezdeki bütün bilgileri akademik ve etik kurallar içinde elde ettiğimi, bu tez çalışmasıyla elde edilmeyen bütün bilgi ve yorumlara kaynak gösterdiğimi ve bu kaynakları da kaynaklar listesine aldığımı, yine bu tezin çalışılması ve yazımı sırasında patent ve telif haklarını ihlal edici bir davranışımın olmadığını beyan ederim.

22 Ocak 2024

Fatih TİRYAKİ

TEŞEKKÜR

Yüksek Lisans öğrenimimde ve bu tezin hazırlanmasında gösterdiği her türlü destek ve yardımdan dolayı çok değerli hocam Prof. Dr. İbrahim YÜCEDAĞ'a en içten dileklerimle teşekkür ederim.

Tez çalışmam boyunca değerli katkılarını esirgemeyen eş danışmanım Dr. Öğretim Üyesi Ümit ŞENTÜRK'e de şükranlarımı sunarım.

Bu çalışma boyunca yardımlarını ve desteklerini esirgemeyen sevgili eşim Zeliha TİRYAKİ'ye ve çocuklarıma sonsuz teşekkürlerimi sunarım.

22 Ocak 2024 **Fatih TİRYAKİ**

İÇİNDEKİLER

ŞEKİL LİSTESİ

Sayfa No

ÇİZELGE LİSTESİ

KISALTMALAR

AE	Derin Kodlayıcılar
ASP	Etkin Sunucu Sayfası
BRNN	Çift Yönlü Tekrarlayan Sinir Ağları
CNN	Evrişimli Sinir Ağları
DAE	Derin Oto Kodlayıcı
DBN	Derin İnanç Ağları
DNS	Alan Adı Sistemi
DL	Derin Öğrenme
FP	Yanlış Olumlu
FN	Yanlış Olumsuz
FTP	Dosya Transfer Protokolü
FP	Yanlış Olumlu
HTTP	Hiper Metin Transfer Protokolü
HTTPS	Güvenli Hiper Metin Transfer Protokolü
ISP	İnternet Servis Protokolü
IOT	Nesnelerin İnterneti
ML	Makine Öğrenme
K-NN	En Yakın Komşu Algoritması
LSTM	Uzun Kısa Süreli Bellek
MLP	Çok Katmalı Algılayıcı
PHP	Üstün Yazı Ön İşlemci
RBM	Sınırlı Bolizman Makineleri
RCNN	Bölge Tabanlı Evrişimli Sinir Ağları
SGD	Stokastik Gradyan İnişi
SVM	Destek Vektör Makinası

TN	Doğru Olumsuz
TP	Doğru Olumlu
TN	Doğru Olumsuz
URL	Tekdüzen Kaynak Bulucu
YSA	Yapay Sinir Ağları
YZ	Yapay Zeka
WWW	Dünya Geniş Ağ

ÖZET
ZARARLI URL'LERİN DERİN ÖĞRENME İLE TESPİTİ

Fatih TİRYAKİ

Düzce Üniversitesi

Lisansüstü Eğitim Enstitüsü, Siber Güvenlik Anabilim Dalı

Yüksek Lisans Tezi

Danışman: Prof. Dr. İbrahim YÜCEDAĞ

Eş Danışman: Dr. Öğr. Üyesi Ümit ŞENTÜRK

Ocak 2024, 41 sayfa

Günümüzde internetin her geçen yıl kullanımın artmasıyla hayatımızda çok önemli bir hale gelmiş ve yeni iletişim teknolojileri, sosyal ağlar, e-ticaret, kullandığımız teknolojik ev aletlerinden, çevrimiçi bankacılık dâhil olmak üzere birçok uygulamada işlerin teşvik edilmesinde ve büyütülmesinde önemli bir etkiye sahiptir. Bu tez çalışmasının amacı, kullanılan yapay zekâ modeli ile zararlı URL adreslerini tespit edilmesi sağlanmış, bu işlemler için büyük bir veri seti ile çalışılmış ve en iyi sonucu elde etmek hedeflenmiştir. Bu tezde literatürde yapılan çalışmalar incelenmiştir, veri setlerinin büyüklüğü ile doğruluk başarı oranına katkısı görülmüştür. Yapılan çalışmada, veri setindeki URL sayısının fazla olması ve RNN model mimarisinin kullanılması performans değerlerini 2 puan oranında arttırmıştır. Çalışmada oluşturulan, 7 katmanlı RNN modeli kullanılmış, modelde çalıştırmak üzere ulusal ve uluslararası birbirine benzer iki adet veri seti birleştirilmiş, 579.112 adet URL adresinden oluşan yeni veri seti oluşturulmuştur. Daha sonra bu yeni veri seti eğitim ve test setlerine ayrılmıştır. İlk olarak veri setimiz modelde eğitilmiş ve ardından ikinci veri seti test edilmiştir. Bu veri seti modelimizde işlendiğinde %91 doğruluk oranı elde edilmiştir. Bu çalışmamızla, zararlı URL adreslerinin tespiti için daha etkin yöntemlerin geliştirilmesine önemli katkı sağlamak hedeflenmektedir.

Anahtar Sözcükler: Kötücül URL, Siber Güvenlik, Yapay Zekâ, Derin Öğrenme, RNN

ABSTRACT

DETECTION OF MALICIOUS URLS USING DEEP LEARNING

Fatih TİRYAKİ

Düzce University

Graduate School, Department of Cyber Security

Master's Thesis

Supervisor: Prof. Dr. İbrahim YÜCEDAĞ

Co-supervisor: Asst. Dr. Ümit ŞENTÜRK

January 2024, 41 pages

In recent years, with the increasing use of the internet, it has become a crucial aspect of our lives. New communication technologies, social networks, e-commerce, and various applications, including online banking and the use of technological home appliances, play a significant role in promoting and expanding business. The purpose of this thesis is to detect malicious URL addresses using the artificial intelligence model employed, working with a large dataset to achieve the best results. Existing studies in the literature were reviewed in this thesis, emphasizing the contribution of dataset size to accuracy success rates. In the study, the high number of URLs in the data set and the use of RNN model architecture increased the performance values by 2 points. A 7-layered RNN model was created in this work, combining two similar national and international datasets to form new dataset comprising 579.112 URL addresses. This new dataset was then divided into training and test sets. Initially, the model was trained on the dataset and subsequently tested on the second dataset. When this dataset was processed through our model, 91% was achieved, indicating results in detecting malicious URL addresses. This study aims to make a significant contribution to the development of more effective methods for detecting malicious URL addresses.

Keywords: Malicious URL, Cyber Security, Artificial Intelligence, Deep Learning, RNN

1. GİRİŞ

Günümüzde internet her geçen gün kullanımın artmasıyla hayatımızın vazgeçilmezi olmuştur. Özellikle yeni iletişim teknolojilerinin (akıllı telefonlar, adroid televizyonlar, giyilebilir saat ve bileklikler) bireysel kullanımı yaygınlaşmıştır. Sosyal ağların aktif ve resmi iletişim aracı olarak kullanılması, e-ticaretin günlük alış verişte kullanılması, eğitim öğretimin sistemin vaz geçilmez bir parçası haline gelen uzaktan eğitim sistemi, bankacılığın neredeyse çevrimiçi bankacılık haline gelmesi gibi birçok gelişmeler internetin daha yoğun kullanılmasına sebep olmuştur. İnternet, evlerimizdeki beyaz eşyalarımızın günlük standart kullanımından daha çok anlık iletişimde olmamızı ve onları uzaktan yönetmemizi sağlamaktadır. İnternetin hayatımızın bu kadar vazgeçilmez bir parçası olmasıyla önemi daha da fazla artmıştır [1].

Özellikle Aralık 2020 de dünyayı saran Covid-19 salgını, internetin kullanımı ve yaygınlaşmasını sürecini daha da hızlandırmıştır. Salgının ilk günlerinde internettin kullanımı ve internette dolaşan kullanıcı sayısı neredeyse ikiye katlamıştır [2]. Bu hızlı artış gittikçe katlanarak devam etmiş ve 2022 yılı sonunda küresel internet kullanıcısı sayısının yaklaşık olarak 5 milyara ulaştığı, internette ki web site sayısının da 2 milyarı bulduğu tahmin edilmektedir. Salgınla beraber ani alınan kararla uzaktan çalışma hayatına geçilmiş, bankalar, eğitim-öğretim kurumları, üniversiteler, iş dünyası, akademik ve bilimsel çalışmalara kadar her şey internetten kullanıma açılmış ve her türlü veri paylaşımı yapılmıştır. İnternetin hayatımızda bu kadar çok kullanılması ve vazgeçilmez hale gelmesiyle güvenli internet kullanımı, internetteki bilginin güvenilirliğinin sorgulanması ve bilgi güvenliğinin kritikliğini önemli hale getirmiştir.

Ülkemizde de 2020 yılında Cumhurbaşkanlığı İletişim Başkanlığı tarafından yayınlanan Bilgi ve İletişim Güvenliği Rehberi ile kamu kurum ve kuruluşları başta olmak üzere kritik kuruluşların güvenli bilgiye ulaşılması ve bilginin güvenliği amaçlanmış, rehberin çalışmalarında güvenli Tekdüzen Kaynak Bulucu (URL) ile ilgili tedbirlerde ele alınmıştır.

İnternetin hayatımızın her alanında bu kadar yoğun kullanımı, hayatın vaz geçilmez bir parçası haline gelmesi, kullanıcı sayısı ve web sitesi sayısı milyarlar ile ifade edilmesi ve her saniye artmaya devam etmesi bilginin güvenilirliğinin önemini daha da arttırmıştır.

Siber suçlular internet kullanıcılarını kandırabilmek için bu internet sitesi adreslerini yani URL adreslerini kullanmaya başlamışlardır. Bu suçlular, kullanılan bu internet adreslerinin gerçeğine çok benzerlerini veya profesyonel yeni internet siteleri yaparak kullanıcıları tuzaklarına düşürmeye çalışmışlardır. Siber suçluların yaptıkları bu zararlı ve riskli URL adresleri kullanıcılar için büyük risk teşkil etmektedir. Bu nedenle, URL adreslerinin zararlı olup olmadığının öncesinde tespit edilmesi güvenli internet kullanımı için çok önemli ve kritik olmuştur. Zararlı ve zararsız URL adreslerinin detaylı olarak incelenerek ayırt edilmesi ve internet kullanıcıları güvenli internete ulaşmaları sağlanmalıdır [3].

Bu tezde zararlı URL adreslerinin tespit edilmesi ile ilgili bu sorunun çözümüne aşağıdaki katkılar yapılması hedeflenmiştir.

• URL adres bilgilerinden elde edilen bilgi ve öznitelikler kullanılarak gerekli optimizasyon süreçleri kullanılmış, 7 katmanlı bir derin sinir ağı olan RNN modeli tasarlanmıştır.

• Araştırılan literatür taramalarında yer alan çalışmalara göre, doğruluk oranının arttırılması için daha geniş ve evrensel bir veri seti oluşturulmuş, elde edilen bu veri seti eğitim seti ve test veri seti olarak belirli oranlarda bölünmüş, eğitilmiş veri seti ile modelin kullanılabilirliği ve performansı doğrulanmıştır.

• Kullanılan modelin, incelenen diğer modellere göre detaylı analizi yapılmış ve daha yüksek doğruluk oranına sahip olduğu görülmüştür.

Tezimiz ikinci bölümünde genel bilgiler, URL ve yapısı, yapay zekâ ve çeşitleri hakkında bilgiler verilerek detaylı açıklanmıştır. Üçüncü bölümde ise çalışma ortamı ve kullanılan yöntem, çalışmada kullanılan veri seti, model mimarisi ve modelin eğitimi, modelin test edilmesine yer verilmiştir. Dördüncü bölümde değerlendirme metrikleri ve yöntemin karşılaştırılması yapılmıştır. Beşinci bölümde elde edilen sonuçlar değerlendirilmiştir.

1. 1. LİTERATÜR TARAMA

Zararlı URL adreslerinin tespit edilmesi ile ilgili detaylı literatür tarama yapılmış, yapılan araştırmalardaki literatür bilgilerin geniş özetleri aşağıda verilmiş ve sonrasından tablo haline getirilerek karşılaştırılmıştır.

Chen ve arkadaşları, bu makalede zararlı URL'leri tespit etmek için YOLO algoritmasını

temel alan geliştirilmiş çok katmanlı tekrarlayan evrişimli sinir ağı (CNN) modeli önerilmektedir. Yöntemde öncelikle tek karakterli kelime gömme kullanılarak yoğun vektörlere işlenmiş ve URL'nin yapısal özelliklerine göre tüm modelin eğitim sürecine dahil edilmiştir. URL'nin özelliklerini çıkarmak için geliştirilmiş YOLO algoritmasını temel alan CSPDarknet sinir ağı modeli önerilmiştir. Son olarak, çıkarılan özellikler çift yönlü LSTM tekrarlayan sinir ağı algoritması tarafından zararlı URL'yi değerlendirmek için kullanılmıştır. Makalede, "iyi" olarak etiketlenmiş 100.000 zararsız URL ve "zararlı" olarak etiketlenmiş 100.000 zararlı URL dahil olmak üzere 200.000 URL toplanmıştır. Modelin eğitilmesi sürecinde, veri setinin yüzde yirmisi doğrulama veri seti olarak seçilmiştir. Deneysel sonuçlar, yöntemin zararlı URL'leri daha hızlı ve etkili bir şekilde tespit ettiğini ve Text RCNN, BRNN ve diğer modellerle karşılaştırıldığında yüksek doğruluk, yüksek geri çağırma oranı ve yüksek doğruluğa sahip olduğunu göstermektedir ve %90 başarı elde edildiği iddia edilmişlerdir [4].

Ahammad ve arkadaşları, 35.300 URL adresinden oluşan veri setinde zararlı olup olmadığını keşfetmeye yönelik makine öğrenme (ML) modeli kullanmışlardır. Modelde Rastgele Orman, Karar Ağacı, Hafif GBM, Lojistik Regresyon ve Destek Vektörü Makine gibi algoritmalar kullanarak dilsel ve etki alanı tabanlı özelliklerini araştırmışlardır. Oluşturdukları makine öğrenimi modeli ile en iyi sonucu %86 oranı ile Hafif GBM algoritmasında almışlarıdır [5]. Paydey ve arkadaşları, 20.000 URL adreslerinden oluşan veri setlerinden ML Hibrit model ile %85 başarıya ulaştıklarını iddia etmişlerdir. Hibrit topluluk modeli yerine veri setilerine bireysel modeller uyguladıklarında, Çok Katmalı Algılayıcı (MLP) ve Karar Ağacı'nda %85,1 doğruluk üretirken, Destek Vektör Makinası (SVM) %77,3 ve rastgele orman %85,25 doğruluk üretmiştir. Ama Hibrit modeli kullandıklarında bu oranın %85,37'ye çıktığını ve bu sonucun kullandıkları tüm modellerden daha iyi bir oran olduğunu bildirmişlerdir. [6].

Kumar ve arkadaşları, 35.000 zararsız ve zararlı mobil URL adresinden oluşan veri seti kullanmışlar ve statik analiz özelliklere dayalı Kayo modeli tasarlamışlardır. KAYO, mobil web sayfalarının Hiper Metin İşaretleme Dili (HTML) ve Java Script (JS) içeriklerinden, URL'lerinden ve mobil cihazlara özgü gelişmiş yeteneklerinden türetilen statik özelliklerini kullanmış ve bu model ile %90 başarı elde ettiklerini iddia etmişlerdir [7]. Zahao ve arkadaşları, DGA alan listesindeki 200.000 zararlı URL ve Alexa'daki 100.000 zararsız URL olmak üzere toplam 300.000 URL den oluşan veri seti kullanılmıştır. Bu veri seti ile N-Gram frekansı, seviye sayısı, uzunluğu ve sayıdan harfe

frekansının 11 boyutlu istatistiksel özelliklerini oluşturdukları algılama algoritmaları ile işleme almışlar ve bu algoritmada %90 başarı sağlamışlardır [8].

Bharadwaj ve arkadaşları, çalışmalarını 80.128 adet zararsız URL ve 147.781 adet zararlı URL içeren 227.909 adet URL'den oluşan veri tabanı üzerinde gerçekleştirilmiştir. Çalışmalarında GloVe aracılığıyla elde edilen sözcük vektörü temsilini içeren özelliklere sahip Yapay Sinir Ağı (YSA) modeli kullanmışlar ve %89 doğruluk elde etmişlerdir [9]. Vecile ve arkadaşları, çalışmalarında 35.377 zararsız URL, 12.000 spam URL, 11.566 zararlı URL ve 9.965 kimlik avı URL kullanmışlardır. Zararsız, Spam, zararlı ve kimlik avı URL'lerini bir araya getirilerek toplam 68.908 URL adresinden oluşan veri setini oluşturmuşlar ve ML'de Uzun Kısa Süreli Bellek (LSTM) modellerinden %79 doğruluk oranına sahip olduklarını iddia etmişlerdir [10].

Rupa Chiramdasu ve arkadaşları, çalışmalarında ikili sınıflandırmada kullanılan lojistik regresyon (AI) adı verilen ML algoritmasını kullanarak URL'leri otomatik olarak sınıflandırmasıdır. Bunun ML algoritmasına yani AI'ye verilmesinin ön aşamasında özellik çıkarımı ve tokenizasyon kullanmışlardır. AI algoritması, verileri kötü amaçlı veya orijinal olarak sınıflandırır. Ayrıca orijinal olanlar iyi, kötü niyetli ise kötü olarak etiketlenmektedir. Çalışmalarında %90 başarı oranını bulduklarını bildirmişlerdir [11]. Chaitanya R. Vyawhare ve arkadaşları, ML yöntemlerinin kullanılarak zararlı URL adreslerinin tespit edilmesine anlatmışlar ama belirli bir deneme çalışması yapmamışlardır [12]. Mansi Mehndiratta ve arkadaşları, çalışmalarında zararlı URL tespiti için K-En Yakın Komşu (K-NN) ve AI gibi denetimli öğrenme algoritmaları kullanılmıştır. Yaptıkları çalışmalarında, K-NN'nin bu bağlamda AI'dan daha yüksek öğrenme doğruluğuna ulaştığını göstermişlerdir. Çalışmalarında K-NN'nin elde ettiği doğruluk %90,61'dir [13].

Yapılan literatür taramasında model, doğruluk oranı, kullanılan veri seti büyüklüğüne göre daha kolay anlaşılması için aşağıdaki gibi özet bir tablo hazırlanmıştır.

Çizelge 1.1. Literatür Özeti

Ref. No	**Veri Seti**	**Model**	**Doğruluk**
4	200.000 URL (100.000 zararsız, 100.000 zararlı)	YSA CNN Yolo algoritması	%90

5	35.300 URL	ML Hafif GBM algoritması	%86
6	20.000 URL	ML Hibrit model	%85
7	350.000 URL	Statik analiz özelliklere dayalı Kayo modeli	%90
8	200.000 zararlı URL ve Alexa'daki 100.000 zararsız URL olmak üzere toplam 300.000 URL	N-Gram frekansı, seviye sayısı, uzunluğu ve sayıdan harfe frekansının 11 boyutlu istatistiksel özelliklerini oluşturdukları algılama algoritmaları	%90
9	80.128 adet zararsız URL ve 147.781 adet zararlı URL içeren 227.909 URL	YSA GloVe aracılığıyla elde edilen sözcük vektörü	%89
10	35.377 zararsız URL, 33.531 zararlı URL olmak üzere toplam 68.908 URL	ML Uzun Kısa Süreli Bellek (LSTM)	%79
11	32.000 URL	Lojistik regresyona dayalı makine öğrenimi	%90
12	-	ML yöntemleri denenmiş	-
13	50.000 URL	K-En Yakın Komşu Algoritması	%90

Literatür çalışmasında incelenen diğer makalelerde 35.000 ile 300.000 arasında veri seti kullanılmış, makine öğrenme modeli (Karar ağacı, Gbm, Lojistik regresyon vb.) ve yapay sinir ağı modelleri (Yolo, LSTM, Glove Tabanlı vb.) kullanılarak %79 - %90 arasında başarı elde edildiği iddia edilmiştir.

Bu tezde incelenen çalışmalara göre daha fazla URL adreslerinden oluşturulmuş veri seti (579 112 adet) kullanılmış ve oluşturulan 7 katmalı RNN modeli kullanılmış ve %91 üzerinde başarı sağlanmıştır.

2. GENEL BİLGİLER

2.1. İNTERNETİN TARİHÇESİ VE YAPISI

İnternet, birçok bilgisayar sistemini TCP/IP protokolü ile birbirine bağlayan, dünya çapında yaygın olarak kullanılan ve sürekli gelişip büyüyen bir iletişim ağıdır denilebilir. Yâda genel bir tanım yapmak gerekirse internet, bilgiye hızlı, daha kolay, uygun ve daha güvenli ulaşmanın ve bilgiyi paylaşmanın en geçerli yoludur denilebilir. Bu sebepten dolayı internet hayatımızın olmazsa olmazı olmuş, önemli gittikçe artmıştır [14].

İnternetin bu kadar yaygınlaşmasıyla kamu, eğitim ve iş sektöründe daha fazla kullanılmasına sebep olmuştur. Özellikle internetin bu faydalı özelliklerini kullanmak isteyen iş dünyası müşterilerine daha fazla hizmeti daha kısa sürede ulaştırmak ve teknolojik imkânları müşterilerine sunmaya çalışmışlardır. Firmalar artık sadece internet sitesine sahip olmanın değil internet üzerinden gerekli tüm hizmetin verilmesi gerekliliğinin önemini kavramaya başlamışlardır [15].

İnternetin yıllara göre genel tarihi aşağıda detaylı olarak verilmiştir.

Soğuk Savaş Dönemi ve ARPANET (1950'ler-1960'lar): İnternetin kökenleri, Amerika Birleşik Devletleri'nin Soğuk Savaş döneminde ortaya çıkan ARPANET projesine kadar dayanmaktadır. 1950'lerin sonlarına doğru, Amerika Savunma Bakanlığı Sovyetler Birliği ile rekabete girmek ve onlara araştırma projelerinde üstünlük sağlamak için bir ağ kurma kararı almıştır. Bu amaçla da, 1960'larda ARPANET projesi başlatıldı. ARPANET, ilk olarak üniversiteler ve araştırma kurumları arasında bilgi paylaşımını kolaylaştırmak ve bilgisayarlar arasında iletişimi sağlamak amacıyla bir ağ oluşturmayı hedeflemiştir. [16].

TCP/IP Protokolü ve İnternetin Temelleri (1970'ler): ARPANET üzerindeki iletişim protokolleri, günümüzde de temel altyapıyı oluşturan TCP/IP protokolüyle yapılmıştır [17]. Bu yıllarda, TCP/IP protokolünün benimsenmesi sayesinde farklı bilgisayar sistemleri arasında standart bir iletişim sağlanarak internetin temelleri atılmıştır.

Alan Adı Sistemi (DNS) (1980'ler): DNS yapısı 1983'te geliştirildi. DNS internet üzerindeki bilgisayarları kullanıcılar ile daha iyi iletişim sağlamak için kullanılmaya başlanmıştır. DNS, IP adresleri yerine alan adlarını kullanmamıza olanak tanıdı ve bu sayede internet üzerindeki gezinmeyi daha da kolay hale getirdi [18].

World Wide Web ve İnternetin Halka Açılması (1990'lar): 1990'ların başlarında, Tim Berners-Lee tarafından geliştirilen World Wide Web (WWW) internet üzerinde bilgi paylaşımını ve internet sayfaları arasında gezinmeyi devrimleştirdi [19]. Bu dönemde, internet bireysel kullanıma açılmaya başlandı. Ayrıca internet servis sağlayıcıları (ISP) ortaya çıkmış ve bu sayede ev kullanıcıları internete erişmeye başlamıştır. Devlet kurumları, iş hayatı üniversiteler, organizasyonlar da bu gelişmeye hızlı bir şekilde ayak uydurmuşlardır. Bağlantı noktalarına isim verilmeye başlanmış, bu kurumlar kendi isimlerinde internet siteleri açmaya başlamışlardır. İlk siber banka internet üzerinde 1994'te kurulmuştur ve ilk olarak Pizza Hut firması internet üzerinden sipariş almaya başlamıştır. Zamanla iletişim firmalarının hemen hepsi internete yatırım yapmaya başlamışlardır [20].

Mobil İnternet ve Yüksek Hızlı Bağlantılar (2000'ler): 2000'lerle birlikte mobil cihazların internet kullanmaya başlaması, yüksek hızlı internet bağlantılarının yaygınlaşması gibi gelişmeler internet kullanımını daha da artırmıştır. Bu gelişmeler, geniş bant bağlantılarının ve fiber optik altyapının gelişimi, daha hızlı ve güvenilir internet erişimini sağladı.

İnternetin ülkemizdeki gelişimi ise, 1990'lı yılların başına dayanmaktadır. Türkiye, ilk internete Nisan 1993'te bağlanıldı ve bu ilk bağlantı Ortadoğu Teknik Üniversitesi'nde (ODTÜ) gerçekleştirilmiştir. Bu internet bağlantı hattı 64kbit/sn hızında ve uzun bir süre de kullanılmıştır. İnternet tüm Türkiye'de öncelikle akademik ortamlarda yaygınlaşmaya başlamıştır. Ardından sırasıyla Ege Üniversitesi, Bilkent Üniversitesi, Boğaziçi Üniversitesi ve İstanbul Teknik Üniversitesi (İTÜ) bağlantıları gerçekleşmiştir. 1996 yılı Ağustos ayında Turnet çalışmaya başlamıştır [21]. 1997 yılında, akademik kuruluşların internet bağlantısını sağlayan ULAKNET çalışmaya başlamış ve üniversiteler nispeten hızlı bir omurga yapısıyla birbirine bağlanmış ve internet kullanılır hale gelmiştir. Özellikle 1999 yılından itibaren internet kullanıcılarının sayısındaki hızlı artış, Türkiye'deki işletmeleri de internet ortamına girmeye hızlandırmıştır. Bu gelişmelere paralel olarak işletmeler, rekabette geri kalmamak için gerekli çalışmalara başlamışlardır. İnternetin bu gelişimine en hızlı uyum sağlayan bankacılık sektörü olmuş internet bankacığı diye yeni kavram ortaya çıkmıştır. İnternet bankacılığı, zaman ve mekân sınırı olmaksızın bir internet bağlantısı ve bu bağlantıyı kullanabilecek bilgisayar veya cep telefonu gibi bir iletişim aracı ile bankacılık hizmetlerinin alınabilmesini sağlayan alternatif imkân sağlayan yeni bir hizmet kanalı olmuştur [22]. Böylelikle zaman ve

mekândan bağımsız, internet erişimine sahip bütün müşterilerin istediği hizmeti alabilmesini imkân sağlamaktır [23]. 1999 yılında, ticari ağ yapısında büyük değişiklikler olmuş ve TURNET'in yerini TTnet adında yeni bir oluşum olmuştur. 2000'lerin başında; ticari kullanıcılar TTnet omurgası üzerinden, akademik kuruluşlar ve ilgili birimler ile Ulaknet omurgası üzerinden internet erişimine sahip olmaya başlamışlardır. Ayrıca bu iki omurga arasında yüksek hızlı bağlantı kurulmuştur [24].

2.1.1. İnternet Temel Kavramlar

İnternet ile ilgili daha doğru bilgi sahibi olmak için temel kavramların bilinmesi gerekmektedir. Bu temel kavramlar aşağıda detaylı olarak açıklanmıştır.

www (World Wide Web): Bu kavramının anlamı dünya çapındaki geniş ağdır. Bu kavaram sayesinde internet popüler ve kullanışlı yapıya kavuşmuştur. Bu yapı ortaya çıkmadan önce internette sadece metin (text) dosyaları bulunuyordu. WWW kavramı ile internet sunucularında sadece metin dosyaları değil resim, müzik, video, animasyon, tasarım ve görsel yapılı dosyalarda sunulmaya başlamıştır [25].

İnternet Tarayıcısı (Web Browser): Kullanıcılar ile internetteki web siteleri arasındaki bağlantıyı sağlayan yazılımlardır [26]. Bu yazılımlar, istenilen web sunucuları ile iletişime geçerek istenilen internet sayfalarını bilgisayarlarımıza ulaştırırlar. Günümüzde Microsoft Edge, Mozilla Firefox, Google Chrome, Apple Safari, Opera, Microsoft İnternet Explorer en çok kullanılan internet tarayıcılarıdır.

Web Sayfası: Bir internet sayfasının içerisinde metin, resim, ses yâda görsel olarak zenginleştirilmiş yapılardan oluşan sayfalardır. HTML yâda diğer web programlama dilleri ile oluşturulmuş olabilirler [27]. Sabit içerikli web sayfası: Kullanıcılar web tarayıcısında ilgili sayfayı ne zaman açsa hep aynı içerik sayfalarını görüntülerler. İçerikte herhangi değişiklik olmaz, genellikler metin yazıları kullanılır [28]. Değişken içerikli web sayfası: Kullanıcıların istekleri veya sistemin özelliklerine göre sayfalar dinamik olarak değişebilir [29]. Bu internet sayfalar oluşturulurken HTML ile birlikte PHP, ASP, ASP.NET, Java İnternet web programlama dilleri kullanılır ve bu internet sayfalarındaki verilerin işlenmesi için veri tabanından faydalanılır.

Bağlantı: İnternet sayfalar arasında geçişleri sağlamak içi kullanılan köprülerdir. Sayfalar arası bu bağlantı sağlamak için metin veya resimlerden faydalanılabilir [25]. Köprü: Bağlantı ile aynı işlevi görürler ve bir birleri yerine de kullanılabilen ifadelerdir [30]. Ana Sayfa (Başlangıç Sayfası): İnternetteki bir web sayfasının ilk açıldığında karşımıza gelen

sayfaya ana sayfa veya başlangıç sayfası denilmektedir [31]. İnternet Sunucu: İnternete bağlantısı olan ve binlerce internet kullanıcısına internet hizmet sunabilen, internet sitelerinin dosyalarını, verilerini depolama ve yayınlama hizmeti sunan bilgisayarlara internet sunucu denilir [32]. Barındırma İşlemi: Web sunucuların internet sayfalarının verilerini saklama ve yayınlama hizmetine barındırma işlemi denir. Bu barındırma ile internet web sitelerinde yayınlanmak istenen bütün bilgilerin internet kullanıcıları tarafından internet tarayıcıyla (Egde, Mozilla Firefox, Internet Explorer, Safari, Google Chrome vb.) erişebileceği bir sunucuda bulundurma hizmetidir. Bulut bilişim hizmeti sunan yazılım işletmeleri bu yazılım ürünlerinin kullanımı ver erişimi için barındırma hizmeti sunarlar [33]. Güvenlik Sunucusu: Sistemi koruyan güvenlik yazılımlarını bünyesinde barındıran sunuculardır. [34].

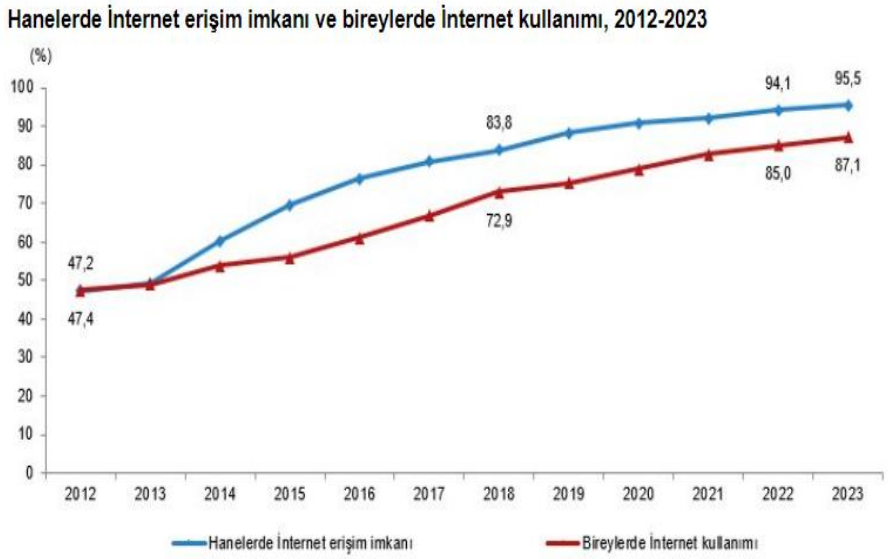
Dosya Transfer Protokolü (FTP) Sunucusu: FTP kelimesinin anlamı İngilizcesi File Transfer Protokol'den gelmektedir [35]. FTP, internet üzerinden dosya sunucu ile dosyaların paylaşımını ifade etmektedir. Proxy Sunucusu: Kendi kullandığımız local yani yerel ağ dünyamız ile dış dünya arasında yer alır. Bilginin bu iki dünya arasında bilginin güvenli olarak temin edilmesi ve veri paketlerin depolanmasını sağlar [36].

TCP/IP protokolü: TCP/IP iki katmanlı bir programdır. TCP/IP protokol paketinin daha üst katmanı, İnternet üzerinden veri iletiminin yönetilmesinde çok önemli bir rol oynayan İletim Kontrol Protokolüdür [37]. Transfer Kontrol Protokolü (TCP), mesajları veya dosyaları ağ üzerinden verimli bir şekilde iletilebilecek daha küçük paketlere böler. Alıcı ise, başka bir TCP katmanı, paketleri orijinal biçimlerine yeniden birleştirerek iletilen verilerin bütünlüğünü ve eksiksizliğini sağlar [38]. İnternet Protokolü (IP) ise, TCP/IP protokol paketinin alt katmanıdır, doğru hedefe teslimatı sağlamak için paket adreslemeyi yönetir [37]. IP Adresi: TCP/IP protokolünde kullanılırlar ve bilgisayarların birbirleriyle sağlık iletişim kurmalarını sağladıkları numaralardır. Bant Genişliği: Bant genişliği kavramı kullanılan veri kablosu üzerinden belli bir süre içinde ne kadar veri transferi yapılabileceğimizi belirtir. Kullanılan toplam kapasitenin (bit/sn) saniyede bit sayısına dönüştürülerek, toplam internet kullanıcısı sayısına bölünerek bulunur [39].

2.1.2. Türkiye'de İnternetin Kullanımı

Ülkemizde internete erişim gittikçe artmaktadır. Tuik'in yaptığı araştırmaya göre şekil 2.1de hane halkı bilişim teknolojileri kullanımı; 2023 yılında evden internete erişim imkânı olanların oranı 2022 yılına göre 1,4 puan artarak %95,5 olmuştur. Ayrıca bu

araştırma sonucunda; internet kullanım oranı, 16-74 yaş grubundaki bireylerde 2022 yılında %85,0 iken 2023 yılında %87,1 olduğu görülmektedir. Bu oran 2023 yılında internet kullanım oranı; erkeklerde %90,9, kadınlarda %83,3 olarak gözlendi [40].

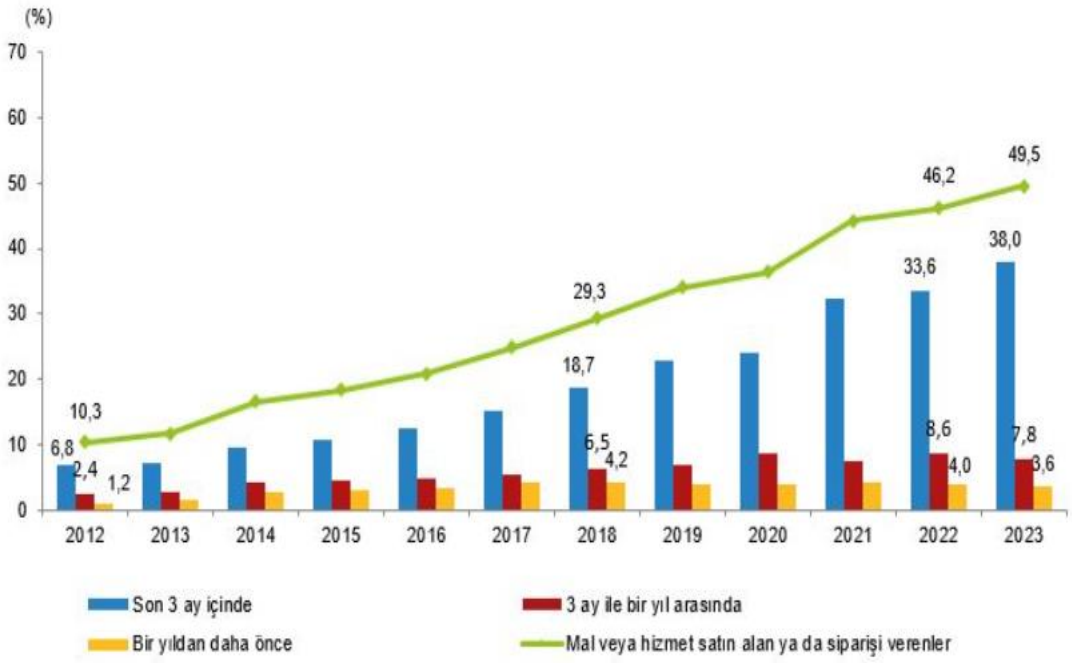


Şekil 2.1. Hanelerde internet erişim imkânı ve bireylerde internet kullan 2012-2023 [40]

E-devlet hizmetlerini kullanan bireylerin oranı %73,9 olarak gerçekleştir. Türkiye İstatistik Kurumunun yaptığı araştırmada son 12 ay içinde vatandaşların kamu kurumlarının internet sayfalarını ve uygulamaları ile kamu hizmetlerinden yararlanma oranı %73,9 görülmüştür [41]. Kamu hizmetlerinin internete taşınmasıyla hizmete ulaşılabilirliği arttırmış ayrıca bu oran kamu hizmetlerinin internette sunduğu hizmete olan güveni de göstermektedir. Bu güven e-Devlet uygulamalarının kullanımı gittikçe arttırmaktadır.

Şekil 2.2. Yıllara göre kullanıcı sayısı [41]

E-devlet hizmetlerinin kullanım amaçlarından %69,6 ile resmi makamlardan kendi kişisel verilerine ulaşma ilk sırada olurken, bunu %51,3 ilke kamu kurumlarından randevu alma, %48,2 ile diğer kamu kuruluşlarının internet sayfalarından bilgi edinme hizmeti takip etmiştir [40]. İnternet üzerinden mal veya hizmet satın alma ya da sipariş verme oranı %49,5'e yükselmiştir. İnternet kullanımın son 12 ayına bakıldığından özel kullanım amacıyla mal ve hizmet satın alma veya internetten mal ve hizmet sipariş verme oranı, 2022 yılında %46,2 iken 2023 yılında %3,3 artmış ve %49,5 olmuştur. Ayrıca internet kullanımın erkeklerin oranı %52,4 ve kadınların oranı %46,6 olmuştur [40]. Erkekler internet kullanımının kadınlardan daha fazla olduğu görülmüştür.

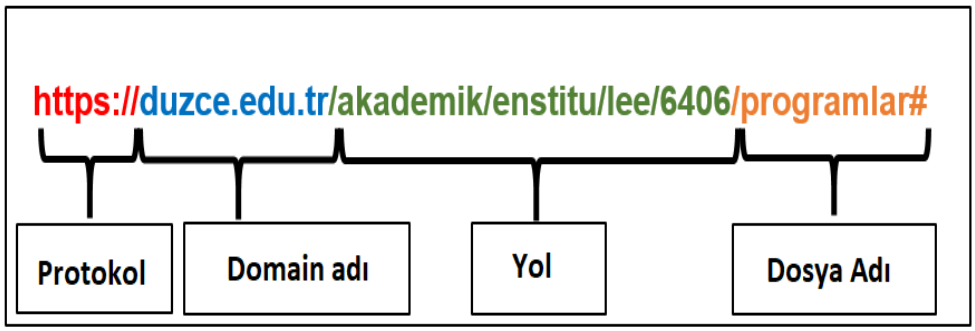


Şekil 2.3. İnternetten en son satın alma veya sipariş oranı, 2012-2023 [40]

Ayrıca yapılan araştırmada 2012-2023 yılları arasında genel karşılaştırılması yapılmış internetten sipariş verme oranın %10,3 den %49,5 arttığı, yani başka bir ifadeyle internet kullanımın çok büyük bir hızla arttığı görülmüştür.

2.2. URL YAPISI

URL tanımı için dosya veya web sayfası gibi herhangi bir veri parçasının internet 'adresi' denilebilir. Yani verinin kaynağını belirten adrestir. Bu internet adreslerinin yapısı kendisi ile ilgili farklı bilgiler vermektedir [42]. Şekil 2.4 de örnek bir URL adresi ve yapısı verilmiştir.



Şekil -2.4. URL yapısı [43]

URL'nin temele bileşenleri detaylı olarak açıklanmıştır. Protokol, internet sayfasındaki kaynağa erişmek için kullanılan protokolü belirtmektedir. Örneğin, bu protokoller HTTP (HyperText Transfer Protocol) veya HTTPS (HTTP Secure)'dir [44]. Alan Adı, internetteki bilgisayarın ve hizmetlerin adresini IP ile gösteririz. Bu internet üzerindeki bilgisayarın veya hizmetin daha kolay isimlendirmesi için kullanılan kelimeler olarak da bilinir [45]. Alan adına www.duzce.edu.tr örnek verilebilir. Port Numarası, sunucuyla bağlantı kurmak için kullanılan belirli bir bağlantı noktasındır [46]. Eğer bu bağlantı numarası belirtilmezse, kullanılan tarayıcılar varsayılan olarak HTTP için 80, HTTPS için ise 443 portunu kullanırlar. Yol, internet sunucusundaki belirli bir dosyanın veya bu dosyanın kaynağının konumunu belirtir. Şekil 2.4 URL yapısındaki "akademik/enstitü/lee/6406/ yolu göstermektedir. Sorgu Parametreleri, istenen isteğe bağlı olarak, sunucuya iletilen ek bilgileri belirtir [47]. İnternet sayfa yönlendirici, internet sunucusundaki sayfanın belirli bir bölümüne yönlendirmek yapmak için kullanılır [48].

2.3. YAPAY ZEKA

Yapay zekâ her alanda olmaya başladı. Yapay zekânın kullanıldığı veya etkilendiği alanlarda eğitim, öğretim, iş, planlama, akıl yürütme oyunları, satranç, yazılımlar, matematiksel problem ispatlar, robotlar, otonom araçlar, sağlık hizmetlerindeki hastalık teşhis metotları, çevrimiçi teknoloji yer almaktadır. Yapay zekânın bu kadar geniş bir alanda kullanımı bu teknolojinin evrensel olduğu göstermektedir. Teknolojik makinaların insanlar gibi düşünmesi hem bizleri korkutmakta hem de yapılabilecekleri düşündükçe heyecanlandırmaktadır. Makinalarında insanlara olan düşünüp karar vermeleri ile ilgili çalışmalar artarak günümüze kadar gelmiştir [49]. Bu bilgilere göre kısaca yapay zekâyı tanımlarsak; Kendisine verilen karmaşık bilgileri alır, çevrelerindeki diğer etkenleri

algılar, veriler arasında ilişkiler kurur, toplanan verileri yapılandırır, belirli bir yorumlama katarak hareket eden ve insanları tasarladığı sistemlerdir [44].

2.3.1. Makine Öğrenme

Makine öğrenmesi (ML), bilgisayar sistemlerinin verilerden öğrenme yeteneği kazandığı bir yapay zekâ alt dalıdır. Bu sistemler, belirli bir görevi veya problemi çözmek için algoritmalar kullanarak veri setlerinden öğrenir ve deneyim kazanır. Makine öğrenimi, programlamaya gerek kalmadan bir bilgisayarın veri analizi yapabilmesine ve belirli bir görevi gerçekleştirebilmesine imkân tanır [50].

Makine öğrenmesi, modellerden maksimum performansı elde etmek üzere üç grup öğrenme türü vardır. Denetimli Öğrenme: Bu yöntemde algoritma, öğrenme sürecinde etiketlenmiş veri setleri üzerinden öğrenir. Her giriş verisi ile bir çıkış etiketi ilişkilidir ve algoritma bu ilişkiyi öğrenmeye çalışır [51]. Öğrendikten sonra, yeni verilerle tahminler yapabilir. Bir resmin içerdiği nesneleri tanımlama görevi denetimli öğrenme örneklerinden biridir. Denetimsiz Öğrenme: Etiketlenmemiş veri setleri üzerinden öğrenme prensibi üzerine kuruludur. Algoritma, veri setindeki yapıları keşfetmeye çalışır. Kümeleme ve boyut azaltma gibi teknikler, bu kategoride kullanılan yöntemlere örnektir. Yarı Denetimli Öğrenme: Bu öğrenme şeklinde ise girdilerin çevreleriyle etkileştirilir ve ödül denilen geri bildirimlerini en yüksek seviyeye çıkarılır, en uygun hareket tarzını bulmayı amaçlanır. Bu yönüyle diğer öğrenim türlerinden farklıdır [52].

2.3.2. Derin Öğrenme

Derin öğrenme, yapay sinir ağları adı verilen çok katmanlı mimarilerin kullanıldığı bir makine öğrenimi alt dalıdır. Genellikle görüntü üzerinde sınıflandırma problemlerinde kullanılır [53]. Derin öğrenme, Evrişimli Sinir Ağları (CNN), Tekrarlayan Sinir Ağları (RNN), Derin Oto-Kodlayıcılar (AE), Uzun-Kısa Vadeli Hafıza Ağları (LSTM), Derin İnanç Ağları (DBN), Sınırlı Boltzmann Makineleri (RBM) gibi mimarilere sahiptir.

2.3.2.1 Evrişimli Sinir Ağları

Evrişimli Sinir Ağları (CNN), özellikle görüntü tanıma görevlerinde başarıyla kullanılan bir tür derin öğrenme modelidir [54]. CNN'ler, özellikle veri setlerindeki örüntülerin konumsal yapıları üzerinde çok etkili bir şekilde çalışabilen bir mimariye sahiptir. CNN'lerin temel katman özellikleri detaylı açıklanmıştır. Evrişim Katmanları, giriş verisi üzerinde belirli filtre boyutlarıyla kaydırma işlemleri yapar. Bu sayede görüntülerdeki

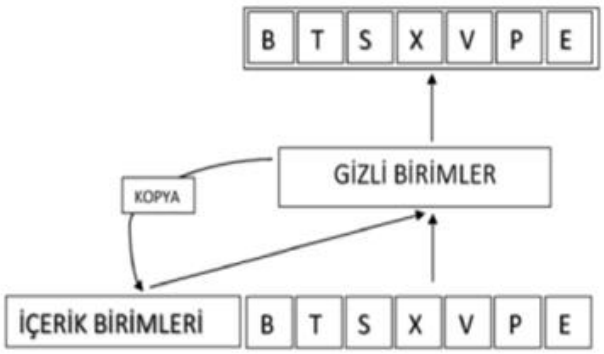
kenarlar, özellikler, renk geçişleri gibi yerel desenler tespit edilebilir. Her evrişim katmanı, genellikle ardışık evrişim, aktivasyon ve pooling katmanlarından oluşur [55]. Aktivasyon Fonksiyonları, CNN'lerde yaygın olarak kullanılan aktivasyon fonksiyonu genellikle düzeltilmiş doğrusal birimdir. ReLU, negatif girişleri sıfır olarak döndürerek non-lineerlik ekler. Pooling Katmanları, evrişim sonrası elde edilen özellik haritasının boyutunu azaltmak ve hesaplama maliyetini düşürmek için kullanılır.

Tam Bağlantılı Katmanlar, CNN'ler genellikle evrişim ve pooling katmanlarından sonra bir veya daha fazla tam bağlantılı katman içerir. Bu katmanlar, özellik haritalarından öğrenilen özellikleri birleştirip, nihai sınıflandırma yapacak kadar geniş bir öğrenme kapasitesi sağlar. Kayıp Fonksiyonları ve Optimizasyon, CNN'ler genellikle sınıflandırma görevlerinde kullanıldığı için "softmax" aktivasyon fonksiyonu ile sona ererler. Kayıp fonksiyonu olarak genellikle "cross-entropy loss" tercih edilir. Optimizasyon algoritmaları arasında SGD veya daha gelişmiş optimizasyon algoritmaları kullanılabilir. CNN'ler, özellikle görüntü tanıma, nesne tespiti ve benzeri görevlerde büyük başarı elde etmiştir. Bu modeller, özellikle önceki katmanlarda öğrenilen düşük seviye özellikleri kullanarak karmaşık yapıları tanıma yeteneği ile bilinirler [56].

2.3.2.2 Tekrarlayan Sinir Ağı

Elman tarafından tasarlanan basit tekrarlayan sinir ağları dil bilimciler ve psikanaliz için çığır açan bir yaklaşım olmuştu. Tekrarlayan Sinir Ağı (RNN), sıralı verilerle çalışmak üzere tasarlanmış bir tür yapay sinir ağıdır. RNN'ler, önceki adımlardaki bilgileri hatırlayabilen ve bu bilgileri bir sonraki adıma iletebilen yapılarıyla bilinirler [57]. Bu özellikleri, özellikle doğal dil işleme gibi zamanla değişen veri türleri için uygundur.

RNN'lerin temel avantajlarından biri, bir zaman serisi içindeki bağımlılıkları öğrenebilme yetenekleridir. Ancak, uzun vadeli bağımlılıkları başarılı bir şekilde öğrenme konusunda zorluklar yaşayabilirler. Bu nedenle, daha gelişmiş türleri olan LSTM veya GRU, uzun vadeli bağımlılıkları daha iyi ele alabilmek için kullanılır [58]. RNN'ler, dil modelleme, metin oluşturma, zaman serisi analizi gibi birçok uygulamada kullanılabilir. Ancak, bazen kaybolan gradyan problemi gibi zorluklarla karşılaşabilirler. Bu nedenle, daha gelişmiş RNN türleri veya başka yaklaşımlar kullanılarak bu sorunlar giderilmeye çalışılır.



Şekil 2.5. Basit tekrarlayan ağ mimarisi

RNN'lerin temel özellikleri detaylı olarak açıklanmıştır. Tekrarlayan bağlantı: RNN'ler, her bir zaman adımında önceki zaman adımından gelen bilgileri içselleştirmek için tekrarlayan bağlantıları kullanır. Bu, RNN'nin önceki girdi ve durumları hatırlama yeteneğini sağlar [59]. Hücre (Cell) yapısı, bir RNN'nin temel yapısı olarak adlandırılır. Her bir hücre, mevcut girişle birlikte önceki zaman adımındaki durumu içerir. Bu durum, ağın geçmiş bilgileri hatırlamasını sağlar. RNN'lerde yaygın olarak kullanılan aktivasyon fonksiyonları arasında tanh ve sigmoid bulunur. Bu fonksiyonlar, hücrelerin çeşitli durumlarını ve çıkışlarını sınırlamak için kullanılır.

Eğitim ve optimizasyon, RNN'ler genellikle zaman serileri veya sıralı veri üzerindeki desenleri öğrenmek için eğitilir. Kayıp fonksiyonları ve optimizasyon algoritmaları, ağı belirli bir görev için eğitmek için kullanılır. RNN'ler, sıralı veri üzerinde çalışırken karşılaşılan zorlukları ele alabilir, ancak bazı problemlerle başa çıkma zorluğu vardır, örneğin, uzun vadeli bağımlılıkları yakalamak için zayıf yetenekleri ve eğitim sırasında karşılaşılan gradyan kaybını içerir. Gelişmiş RNN modelleri arasında LSTM ve GRU gibi türler bulunur. Bu modeller, RNN'lerin temel yapılarını geliştirerek daha uzun vadeli bağımlılıkları ele alabilirler.

2.3.2.3 Uzun Kısa-Süreli Hafıza

Uzun Kısa-Süreli Hafıza (LSTM), RNN'lerin bir türüdür ve özellikle uzun vadeli bağımlılıkları ele alabilen özel bir hücre yapısına sahiptir [60]. LSTM'ler, geleneksel RNN'lerin karşılaştığı uzun vadeli bağımlılık problemlerine çözüm sunarak sıralı veriler üzerinde daha etkili bir şekilde çalışabilirler.

LSTM hücresi, tipik olarak üç ana kapıya (giriş, çıkış ve unutma kapıları) sahiptir. Her bir kapı, hücre içindeki bilgileri kontrol ederek ve düzenleyerek uzun vadeli bağımlılıkları

korur. İşte LSTM'nin temel bileşenler açıklaması verilmiştir. Giriş Kapısı, giriş kapısı, hücreye yeni bilgiler eklemeye karar verir. Bu kapı, mevcut girişle önceki hücre durumu arasında hangi bilgilerin saklanacağına karar verir [61]. Unutma Kapısı, hücre içindeki bilgilerin ne kadarının unutulacağına karar verir. Bu, geçmiş zaman adımlarındaki bilgilerin uzun vadeli hafızada tutulmasını kontrol eder. Çıkış kapısı, hücrenin içindeki bilgileri kullanarak yeni hücre durumunu belirler. Bu kapı, hücrenin yeni durumu ve çıkışını kontrol eder. LSTM hücresi, bu kapılar sayesinde mevcut giriş ve önceki hücre durumu arasındaki bilgileri düzenler ve geçmiş zaman adımlarındaki önemli bilgileri uzun vadeli hafızada korur [62]. LSTM'ler, özellikle uzun vadeli bağımlılıkların önemli olduğu görevlerde, örneğin doğal dil işleme ve zaman serileri analizi gibi alanlarda başarılı bir şekilde kullanılmaktadır.

2.3.2.4 Kısıtlı Boltzmann Makinesi

Kısıtlı Boltzmann Makinesi (RBM), enerji tabanlı bir olasılık modelidir ve özellikle temel bileşen analizi, belirli tipte öğrenme ve özellik çıkarımı gibi görevlerde kullanılır. RBM, öğrenme sürecinde genellikle kullanıcı tarafından belirlenen sınırlı bir bağlantı yapısına sahiptir, bu nedenle "kısıtlı" olarak adlandırılır [63]. RBM'nin temel özellikleri verilmiştir. Gizli ve görünür katmanlar, RBM iki tip katmana sahiptir. Bu katmanlar gizli ve görünür katmanlar. Görünür katman, modelin gözlemlediği veriyi temsil ederken, gizli katman, verinin içsel özelliklerini öğrenmeye çalışır. Bağlantılar ve enerji fonksiyonu, RBM'nin gizli ve görünür katmanları arasındaki bağlantılar öğrenilebilen ağırlıklarla temsil edilir. Enerji fonksiyonu, görünür ve gizli katmanların durumlarına dayanarak bir enerji değeri üretir. Boltzmann dağılımı ve olasılık, RBM enerji fonksiyonu üzerinden Boltzmann dağılımını kullanır. Bu dağılım, görünür ve gizli katmanların belirli durumlarının olasılıkla ilişkilendirilmesini sağlar [64]. Eğitim, RBM eğitim sürecinde belirli bir veri setine uyum sağlar. Gibbs örnekleme ve kontrastif divergence gibi teknikler, RBM'yi eğitmek için kullanılır. Bu süreçte, ağırlıkların ve bağlantıların güncellenmesiyle model, veri setindeki desenleri öğrenir. Özellik çıkarımı ve temel bileşen analizi, RBM özellik çıkarımı ve temel bileşen analizi gibi görevlerde kullanılarak verideki gizli yapıları ortaya çıkarabilir. RBM'ler, derin öğrenme modellerinin temelini oluşturan bir tür olasılık modelidir. Ancak, daha sonraki gelişmelerle birlikte, özellikle derin ince ağlar ve derin evrişimli sinir ağları gibi modellerin popülerliği arttıkça, RBM'nin kullanımı biraz azalmıştır. Ancak temel öğrenme prensiplerini anlamak için hala önemlidir.

2.3.2.5 Derin İnanç Ağı

Derin İnanç Ağı (DBN), kısıtlı RBM adı verilen özel bir türleştirmeye dayanan bir tür derin öğrenme modelidir. DBN'ler, katmanlı bir yapıya sahip olup, birbirini takip eden bir dizi kısıtlı Boltzmann makinesinden oluşur. DBN'ler, özellik çıkarımı, boyut azaltma ve veri temsili gibi görevlerde kullanılarak özellikle sıralı veri setlerinde etkilidir [65]. DBN'nin temel katman özellikleri detaylı verilmiştir. Katmanlı yapı, DBN, genellikle üç katmanlı bir yapıya sahiptir. İlk katman görünür veriyi temsil eder, orta katman gizli katman olarak adlandırılır ve en üstte ise genellikle bir sınıflandırıcı olarak kullanılan bir doğru katman bulunur. Birbirini takip eden RBM'ler, DBN'nin her bir katmanı, birbirini takip eden kısıtlı Bolzmann makinelerinden oluşur. Her bir RBM, bir öncekine dayanarak eğitilir. Greedy Layer-Wise eğitim, DBN genellikle katman katman eğitim prensibiyle eğitilir. Bu, her bir RBM'nin sırasıyla eğitilmesini içerir. Bu sürecin amacı, her katmanın önceki katmandan gelen temel özellikleri öğrenmesini sağlamaktır. Feeding-Forward ve Fine-Tuning, RBM'lerin eğitimi tamamlandıktan sonra, DBN genellikle bir yukarıdan aşağıya süreç ve fine-tuning adı verilen bir süreçle tamamlanır. Bu süreçte genel modelin eğitimi gerçekleştirilir [66]. DBN'ler, özellikle sıralı veri setleri ve tabakalı temsil öğrenimi görevlerinde başarı elde edebilir. Ancak, daha yeni gelişmelerle birlikte, özellikle evrişimli ve kaybolan gradyan birleştirildiği modeller gibi diğer derin öğrenme yapıları da popülerlik kazanmıştır.

2.3.2.6 Derin Oto-Kodlayıcılar

Derin oto-kodlayıcılar (DAE), öğrenme esnasında gizli bir temsili öğrenmek için tasarlanmış sinir ağı modelleridir. Bu modeller, giriş verisini temsil eden daha düşük boyutlu bir kodlamaya dönüştürmeyi ve ardından bu kodlamayı kullanarak giriş verisini yeniden üretmeyi amaçlar. DAE'ler, veri içindeki önemli özellikleri öğrenerek verinin daha kompakt bir temsilini oluşturabilirler [67]. DAE, genellikle iki temel bileşenden oluşur. Kodlayıcı, bu kısım, giriş verisini daha düşük boyutlu bir temsile dönüştürür. Bu işlem genellikle boyut azaltma amacı taşır. Çözücü, bu kısım, kodlayıcı tarafından üretilen temsili kullanarak giriş verisini yeniden oluşturur. Yani, orijinal giriş verisine benzeyen bir çıkış üretir. DAE, birçok katman içerebilen ve bu sayede daha karmaşık yapıları öğrenmeye yeten bir tür olan "derin" oto-kodlayıcılar olarak da adlandırılabilir. Bu derin yapının öğrenme süreci genellikle denetimsiz öğrenme olarak geçer, çünkü eğitim verisi içerisinde etiket bilgisi kullanılmaz. DAE kullanım alanları verilmiştir.

Özellik çıkarımı, DAE, giriş verisinden önemli özellikleri çıkarmak için kullanılabilir. Özellikle görüntü işleme ve ses işleme gibi alanlarda başarılı olabilirler [68]. Gürültü temizleme, model, gürültülü veriyi temiz bir şekilde çıkışa dönüştürmeyi öğrenerek gürültü temizleme görevlerinde kullanılabilir. Veri görselleştirmesi, DAE'ler yüksek boyutlu veriyi daha düşük boyutlu bir temsile dönüştürerek veriyi görselleştirmek için kullanılabilir. Jeneratif modelleme, gelişmiş oto derin modelleri, veri setine benzer veriler üretebilen jeneratif modeller olarak da kullanılabilir. DAE'ler, genellikle sinir ağı tabanlı olmaları nedeniyle, büyük miktarda veri ve hesaplama gücü gerektirebilir. Ancak, başarılı bir şekilde eğitildiklerinde, veri setindeki desenleri öğrenme ve özellik çıkarımı gibi birçok görevde etkili olabilirler.

2.3.3. Yapay Sinir Ağları

Yapay Sinir Ağları (YSA), biyolojik sinir ağlarının işleyişinden ilham alarak tasarlanan, öğrenme, tanıma, karar verme gibi görevleri gerçekleştirmek üzere kullanılan matematiksel model ve algoritmaları ifade eder. YSA, bilgisayar sistemlerinde karmaşık işlevleri yerine getirmek üzere tasarlanmış bir dizi bağlantılı birimden oluşur [69].

Yapay sinir ağlarının temel özellikleri detaylı verilmiştir. Nöronlar ve katmanlar, yapay sinir ağları, nöronlar adı verilen hesaplama birimlerinden oluşur. Bu nöronlar genellikle katmanlar halinde düzenlenir. Giriş katmanı ve çıkış katmanı olmak üzere iki temel katmanı vardır. Arada kalan katmanlar ise genellikle gizli katmanlar olarak adlandırılır. Ağırlıklar ve bağlantılar, nöronlar arasındaki bağlantılar, ağırlıklar tarafından temsil edilir. Bu ağırlıklar, nöronlar arasındaki bağlantıların gücünü ifade eder ve öğrenme süreci boyunca uyarlanabilir. Aktivasyon fonksiyonları, her nöron girişini alır, ağırlıklarla çarpar, aktivasyon fonksiyonu tarafından işlenir ve bir çıkış üretir. Bu aktivasyon fonksiyonları, nöronun çıkışını belirleyen non-lineer işlevlerdir. Sigmoid, tanh ve ReLU gibi fonksiyonlar yaygın olarak kullanılır. YSA'lar, veri setlerinden öğrenmek için çeşitli öğrenme algoritmaları kullanır. [70]. Derin öğrenme (DL), çok katmanlı yapılara sahip derin sinir ağlarını içerir. DL'ler, karmaşık özellikleri öğrenme yeteneği ile büyük veri setlerinde başarı elde edebilir. YSA, çeşitli uygulama alanlarında kullanılır, örneğin görüntü tanıma, doğal dil işleme, oyun stratejileri, ses işleme, tıp ve finans gibi birçok alanda. YSA, DL modelleri gibi gelişmiş formları da içerir. Bu modeller, büyük ölçüde karmaşık problemleri çözmek ve desenleri öğrenmek için kullanılır [71].

3. MATERYAL VE YÖNTEMLER

Bu bölümde, araştırmanın temelini oluşturan materyal ve yöntemlere odaklanılmıştır. Zararlı URL adreslerinin tespit edilmesi için model geliştirerek katkı sağlamak amacıyla kullanılan materyal ve yöntemler detaylı bir şekilde açıklanmıştır. Ayrıca çalışmada kullanılan veri setleri, geliştirme ortamı, önerilen yöntem, model oluşturma ve mimari yapı süreçlerine dair bilgiler verilmiştir.

3.1. VERİ SETİ

Veri setinin performansı, büyüklük ölçüde veri setinin büyüklüğüne ve daha geniş çaplı ulusal veri olmasına bağlıdır. Gerçek veri seti oluşturulurken büyüklüğüne dikkat edilmiştir.

Çizelge – 3.2 URL Veri Seti Sayısı

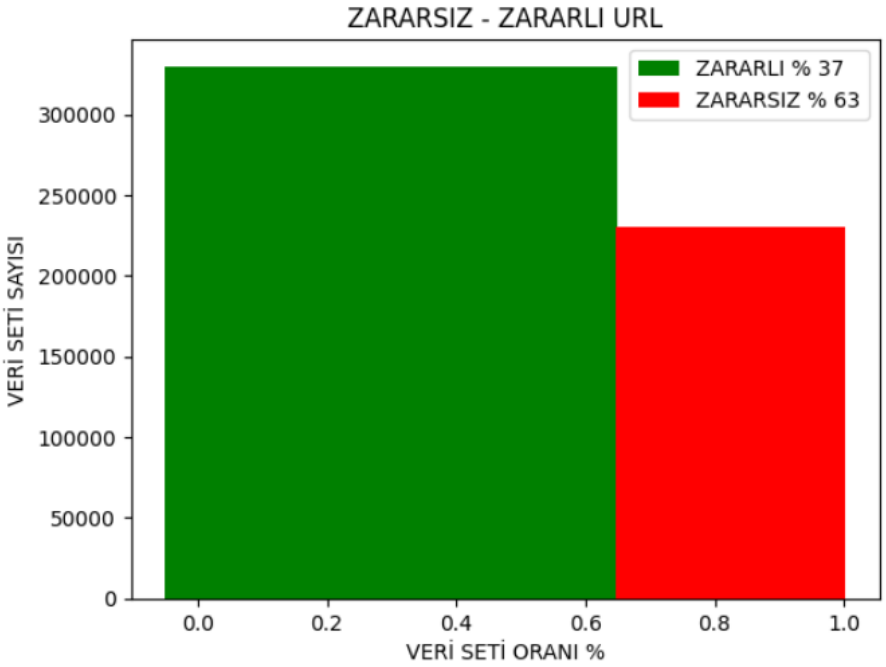
URL Veri Seti	**Kategori**	**URL Sayısı**
CC0: Public Domain Lisanslı	Karma	411.247
Ulusal Siber Olaylara Müdahale Merkezi (SOME)	Zararlı	167.865
Toplam		**579.112**

Çalışmada kullanılmak üzere yeni veri seti oluşturulmak için Kaggle tarafından dağıtımı yapılan CC0 Public Domain Lisanslı 411.247 adet karma URL adresi [72] ve Ulusal Siber Olaylara Müdahale Merkezi tarafından 12.09.2022 tarihinde zararlı linkler bölümünde dağıtımı yapılan 167.865 adet zararlı URL [73] adresi birleştirilmiştir. Böylelikle ulusal ve uluslararası ve daha geniş veri setine ulaşılmıştır.

```
116661  memorialwebsites.legacy.com/sallyphipps/Subpage.aspx?mod=7,zararsiz
116662  memoriesswingteam.com/,zararsiz
116663  dijitaldovizyenidenizbank.com,zararli
116664  denizonlinekredibasvurusistemim.gq,zararli
116665  istkartkampanya.net,zararli
116666  melvinmt.com/,zararsiz
116667  bimcelltrhizmet.com,zararli
116668  medlibrary.org/medwiki/FIFA_U-17_World_Cup,zararsiz
116669  a101sanalmarket.com,zararli
116670  menardcad.org/,zararsiz
116671  memory-beta.wikia.com/wiki/16th_century,zararsiz
```

Şekil 3.1.Veri seti örneği

Birleştirilen iki veri setiyle toplam 579.112 adet URL adresinden oluşan yeni veri setinin ismi urldata olarak belirlenmiştir. Şekil 3.1'de görüldüğü gibi veriler zararlı ve zararsız ismiyle etiketlenmiştir.



Şekil 3.2. Zararlı olmayan ve Zararlı URL veri seti oranı

Bu oluşturulan yeni veri setinin % 37'si zararlı URL adresleri % 63'ü zararsız URL adresleri kullanılarak daha geniş ve şeffaf veri seti oluşturulmak istenmiştir.

3.2. GELİŞTİRME ORTAMI

Google Colaboratory

Yapay zeka, makine öğrenme ve derin öğrenme problemleri için çözümler üretebilen modelleri oluşturma imkânı sunmaktadır [74]. Oluşturduğumuz bu modellin çalıştırılması için donanımsal olarak çok güçlü bilgisayarlara ihtiyaç duyulmaktadır. Donanımsal olarak iyi bilgisayarlarımız olsa da eğitim ve test süreleri saatler almaktadır. Colaboratory GUP desteği ile bu işlemleri rahatlıkla yapabilmektedir.

Python programlama dili ile geliştirilen bu internet tabanlı platform sayesinde, bu uygulamaların kolayca geliştirilmesine fırsat verilmiştir. Tez çalışmasında yapılan uygulamaların hepsi Google Colaboratory üzerinde gerçekleştirilmiştir. Modelleri gerçekleştirmek için GPU geliştiricilere performanslı kullanım imkânı vermektedir. Çalışmalarda CPU yerine GPU kullanımı tez çalışmasının süresini kısaltmıştır.

```
from google.colab import files

uploaded = files.upload()
```

Şekil 3.3. Colab dosya yükleme kod bloğu

Çalışmada öncelikler veri setinin yükleme işleme yapılmıştır. Şekil 3.2'de from google.colab import files komutuyla Colab ortamında dosya yükleme işlevselliğini sağlamak üzere Google Colab kütüphanesinden gerekli modülleri içe aktarma işlevini yapmıştır. İkinci sıradaki uploaded = files.upload() komutuyla, kullanıcıdan dosya seçmesini sağlayan bir iletişim penceresi açılır ve seçilen dosyaları Colab ortamına yüklenir. Yüklenen dosyalar uploaded adlı bir sözlükte depolanır, böylece daha sonra bu dosyalara tekrar erişebiliriz.

Kullanılan Yazılım Kütüphaneleri

Bu kullanılan modeldeki kod blokları Python programlama dili ile Google Colaboratory platformu üzerinde geliştirilmiştir. Bu tez çalışmasına ait verilerin kullanılması, ön işlemden geçirilmesi, derin öğrenme ağ modelinin tasarlanması, ağın eğitimi, ağın test edilmesi ve modelin tüm değerlendirme adımları için birçok kütüphane kullanılmıştır. Bu derin öğrenme modellerinin uygulamaları için Keras, Tensorflow, NumPy, Pandas, Matplotlib kütüphaneleri kullanılmıştır. Numpy kütüphanesi, matematiksel işlemlerin gerçekleştirilmesini sağlayan açık kaynak kodlu bir kütüphanedir. Matris ve dizi işlemleri bu kütüphane yardımıyla gerçekleştirilebilmektedir [75]. Pandas kütüphanesi, farklı dosya tipine sahip olan verilerin analiz edilmesini, okunmasını, yazılmasını vb. işlemlerin kolaylıkla yapılmasını sağlayan bir kütüphanedir. Bu tez çalışmasında kullanılan verileri düzenleme işlemleri bu kütüphanelerde yapılmıştır [52].

TensorFlow kütüphanesi açık kaynak kodlu bir kütüphanedir. Bu kütüphanede hızlı ve kolay sayısal hesaplamalar yapılabilmektedir. Makine öğrenmesi modellerini uygulamak oldukça güçtür. TensorFlow ile bu modellerin eğitimi ve tahmin işlemleri kolayca gerçekleştirilebilir. TensorFlow, derin öğrenme ve makine öğrenmesi modellerini ortak bir şekilde kullanımını sağlayabilir. Uygulamalara kullanışlı bir ön uç API' si sağlamak için Python programlama dilini kullanmaktadır. TensorFlow, görüntü sınıflandırma, el yazısı sınıflandırma, doğal dil işleme, sıralı modeller ve kısmi diferansiyel tabanlı simülasyonlar için derin sinir ağı oluşturabilmektedir. TensorFlow kütüphanesi grafik işlemlerine de izin vermektedir. Ayrıca bu kütüphanede Python dili ile uygulama, model geliştirmek kolay yapılmaktadır. Bu kütüphanenin kullanılan nesneler ve uygulamaların

da her biri bir Python uygulamasıdır [76]. Keras kütüphanesi, derin öğrenme uygulamaları için yazılmış bir kütüphanedir. Derin öğrenme modellerinde kullanılan TensorFlow kütüphaneleri üzerinde çalışabilir ve bazı sembolik işlemlerde de kullanılabilir. CPU veya GPU da çalışmasını tensorFlow kütüphenesi üzerinden sağlayabilmektedir. TensorFlow ve Theano kütüphanelerinden daha gelişmiş olduğu için uygulama geliştirilebilir. Sık kullanılan Makine Öğrenmesi katmanları Keras'ın içinde tanımlanmış durumdadır. Keras'ın kurulumu pip ile gerçekleştirilebilir [77].

Bu tez çalışmasının uygulamasında Pandas, Seaborn, Matplotlib ve Sklearn kütüphaneleri kullanılmıştır. Bu kod bloğunda, Python dilinde veri analizi ve görselleştirmesi yapmak için altı popüler kütüphane kullanılmıştır. Çizelge 3.3 de kullanılan kütüphanelerin versiyon, yazar ve lisans bilgileri tablo yapılmış ve kullanım amaçları aşağıda verilmiştir.

Çizelge 3.4. Kullanılan Kütüphaneler

Kütüphane Adı	Versiyon	Yazar	Lisans
Pandas	1.5.3	The Pandas Development Team	BSD-3-Clause
Seaborn	0.13.1	Michael Waskom	-
Matplotlib	3.7.1	John D. Hunter, Michael Droettboom	PSF
Sklearn	1.2.2	John D. Hunter, Michael Droettboom	PSF
TensorFlow	2.15.0	Google Inc.	Apache 2.0
Keras	2.15.0	Keras team	Apache 2.0

Çalışmada kullanılan kütüphanelerin çalıştırılan kod bloğundaki yaptığı işlevler hakkında şu şekilde bilgi verilmiştir. Pandas (import pandas as pd): Veri analizlerini ve veri manipülasyonlarını yapmak için kullanılan bir kütüphanedir. Pandas, veri çerçeveleri (dataframes) üzerinde işlem yapmayı sağlar. Seaborn (import seaborn as sns): Matplotlib üzerine inşa edilmiş ve bilgi verici istatistiksel grafikleri sağlayan görselleştirme kütüphanesidir [78]. Matplotlib.pyplot (import matplotlib.pyplot as plt): Matplotlib, grafikler, görseller ve çizimler oluşturmak için kullanılan bir kütüphanedir. Pyplot modülü ise Matplotlib'in MATLAB tarzında bir arayüz verir. Sklearn.preprocessing (from sklearn import preprocessing): Veri ön işleme de kullanılan bir kütüphanedir. Yani

veriyi standartlaştırmak, normalize etmek ve kodlama gibi işlemler için çeşitli araçlar sunar. Bu kütüphaneler, veri setleri üzerinde çalışmak, veriyi görselleştirmek ve makine öğrenimi modelleri geliştirmek gibi çeşitli veri bilimi görevlerinde sıklıkla kullanılır.

Kullanılan Donanımsal Araç

Uygulamanın daha performanslı çalışabilmesi için donanımsal özellikleri yüksek bilgisayar kullanılmıştır. Kullanılan bilgisayarın donanımsal özellikleri aşağıda tablo halinde verilmiştir.

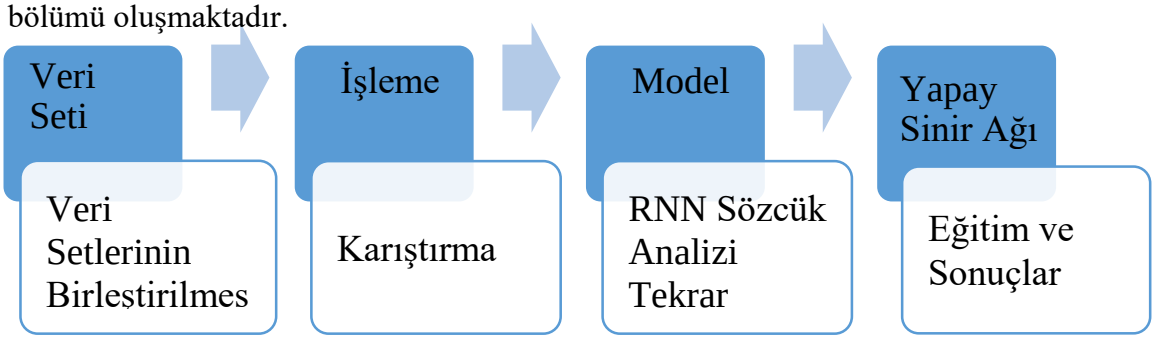
Çizelge – 3.4 Kullanılan Bilgisayar Özellikleri

İşlemci	i7
İşlemci Hızı	2.4 Ghz 4 Çekirdek
Bellek	32 GB
Disk	SSD
İşletim Sistemi	Windows 10 Pro
İşletim Sistem Versiyonu	21 H2

Çalışmada ortamında kablolu 24 Mbps hızında internete kullanılmıştır. Uygulama için programlama dili olarak Python ve internet tabanlı Google Colab kullanılmıştır.

3.3. ÖNERİLEN YÖNTEM

Yapılan çalışmamın işlemlerini gösteren URL İşlem Diyagramı aşağıda gösterilmektedir. Diyagramda veri setlerinin birleştirildiği veri seti bölümü, verilerin karıştırıldığı işleme bölümü, RNN modelinin oluşturulduğu, Sözcük analiz (tokenizer) ve tekrar (epoch) yapıldığı model bölümü, eğitimlerin yapıldığı ve sonuçların alındığı yapay sinir ağı bölümü oluşmaktadır.



ŞEKİL 3.5 RNN Model [43]

3.4. MODELİN OLUŞTURULMASI

Yapılan çalışmada 7 katmanlı RNN modeli oluşturulmuştur. Bu katmaların her birinden yapılması istenen işlemler tanımlanmıştır. Oluşturulan model aşağıdaki şekilde verilmiş olup, her katmanın görevi açıklanmıştır.

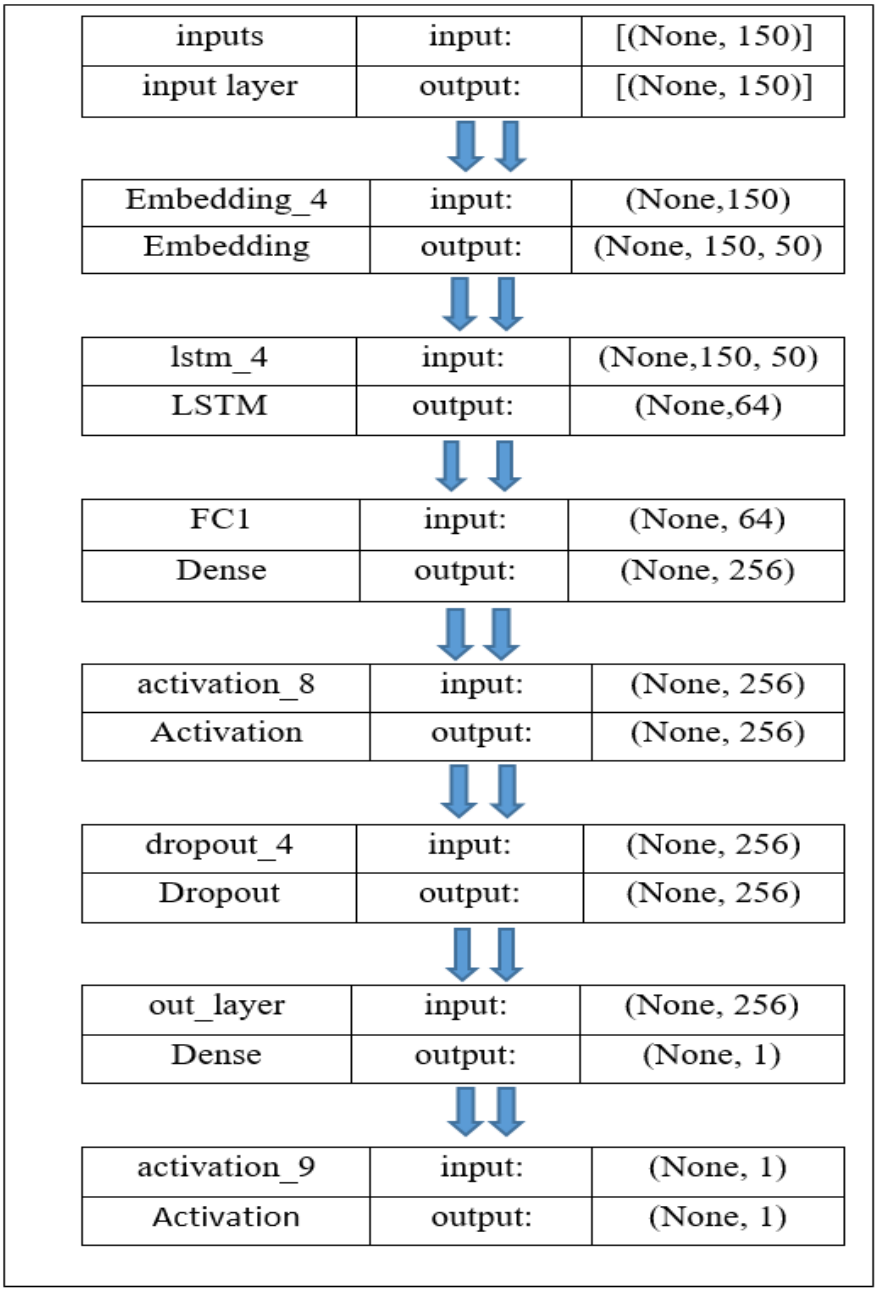
```
# model
def RNN():
 inputs = Input(name='inputs',shape=[max_len])
 layer = Embedding(max_words,50,input_length=max_len)(inputs)
 layer = LSTM(64)(layer)
 layer = Dense(256,name='FC1')(layer)
 layer = Activation('relu')(layer)
 layer = Dropout(0.5)(layer)
 layer = Dense(1,name='out_layer')(layer)
 layer = Activation('sigmoid')(layer)
 modelim = tf.keras.Model(inputs=inputs,outputs=layer)
 return modelim
```

ŞEKİL 3.6. Oluşturulan model katmaları

Giriş Katmanı, giriş katmanı, modelin giriş verilerini alır. Giriş verileri max_len uzunluğunda bir dizi olarak kabul edilir. Gömme Katmanı, kelime haznesini belirli bir boyuttaki vektöre kadar temsil eder. Bu satırda her kelime 50 boyutlu bir vektörle temsil edilmektedir. LSTM Katmanı, uzun vadeli bağımlılıkları ele alabilen bir sinir ağı katmanıdır. Bu modelde 64 hücreye sahip bir LSTM katmanı kullanılmıştır. Yoğun Katman, genişletilmiş özellik işlemleri gerçekleştirir. Bu katmanda 256 adet nöron bulunmaktadır. ReLu aktivasyonunda pozitif giriş değerlerini doğrudan geçirir, negatif değerleri sıfır yapar. Sönümleme katmanında, eğitim sırasında belirli bir olasılıkla rasgele nöronları devre dışı bırakarak aşırı uyumu önlemeye yardımcı olur. Yoğun Katman ve Sigmoid Aktivasyonu, modelin çıkışını üreten son katmanlardır.

3.5. MİMARI YAPI

Veri setimizi 7 katmandan oluşan RNN modeline işleme alınmış ve eğitimlerini tamamlanmıştır. Daha sonra Eğitilmiş RNN modelimize test verilerimizi işleme alarak gerçek doğruluk değerine ulaşmaya çalıştık. Bu işlemde kullandığımız model şeması aşağıda verilmiştir.



Şekil 3.6. Model Akışı [43]

Oluşturulan 7 katmanlı RNN modelin akışını ile ilgili bilgiler verilmiştir. Modeli oluştururken maksimum 150 karakter uzunluğunda ve en fazla 50 kelime türetebilen bir gömme katmanı, 1x64 boyutunda bir özellik vektörü çıkaran bir LSTM katmanı ve 256 düğümlü bir çıkış katmanı kullanılmıştır [79]. Çıkış katmanımda çok sınıflı bir veri kullandığım için aktivasyon fonksiyonu olarak ReLu fonksiyonu kullanılmıştır. Daha sonra loss fonksiyonu olarak binary_crossentropy ve optimizasyon algoritması olarak Adam algoritmasını metrik olarak da doğruluk kullanılmıştır. Modelin eğitimde10 tekrarı yapılmış, test verisinin değerlendirme başarısı 0.91 olarak ölçülmüştür [43].

Layer (type)	Output Shape	Param #
inputs (InputLayer)	[(None, 150)]	0
embedding (Embedding)	(None, 150, 50)	50000
lstm (LSTM)	(None, 64)	29440
FC1 (Dense)	(None, 256)	16640
activation (Activation)	(None, 256)	0
dropout (Dropout)	(None, 256)	0
out_layer (Dense)	(None, 1)	257
activation_1 (Activation)	(None, 1)	0

Total params: 96,337
Trainable params: 96,337
Non-trainable params: 0

Şekil 3.7. Giriş çıkış katmalar

```
model.fit(sequences_matrix,Y_train,batch_size=128,epochs=10,validation_split=0.2)
```

Şekil 3.8. Model test kodlarının oluşturması

Bu kod satırı ile veriye modelinizi (fit) ile yapar ve eğitimi gerçekleştirmeyi ifade eder. Kullanılan kod satırını aşağıda detaylı açıklanmıştır.

Sequences_matrix: Eğitim verilerinin sayısal olarak temsil edildiği matristir [80]. Modelinizin girişine bu matris ile eğitim gerçekleştirilmektedir. Y_train: Eğitim verilerinin bu etiket ile yapacağız. Model, belirlediğimiz bu etikete göre eğitilecek ve doğruluk değerini öğrenecek. Batch_size: Her eğitim adımında kullanılacak örnek sayısını ifade eder. Büyük bir veri seti üzerinde eğitim yaparken, işlem yükünü azaltmak için genellikle küçük örnek grupları kullanılır. Bu satırda 128 kullanılmıştır. Epochs: Modelin kaç kez eğitim verisini tekrarlayacağını ifade eder. Her tekrar, modelin tüm eğitim verisini bir kez geçmesini temsil eder. Bu satırda 10 kez kullanıldı. Validation_split: Eğitim sırasında modelin performansını izlemek ve aşırı uyumu kontrol etmek için ayrılan doğrulama veri setinin oranıdır [81]. Örneğin, 0.2 değeri, eğitim verisinin %80'ini kullanırken geriye kalan %20'sini doğrulama için kullanacağınız anlamına gelir.

3.5.1. Modelin Eğitilmesi

İki farklı yerden alınan veri setleri birleştirilmiştir. Bu çalışmada Python da yapay zekâ metodu RNN modeli uygulanmıştır. Yeni hazırlanan veri seti sisteme tanımlanmış, URL adresleri karıştırılmış, eğitim ve test seti olarak ikiye ayrılmıştır.

Çizelge – 3.5 Eğitim test veri seti dağılımı

URL Veri Seti Dağılımı Eğitim / Test	**Kategori**	**URL Sayısı**
Veri Seti	Eğitim	463.290
	Test	115.822
Toplam		**579 112**

Bu URL adresleri %80'ini (463.290) eğitim setine ayırmış geri kalan %20'ini (115.822) test setine ayrılmıştır. Böylelikle veri setleri daha doğru eğitilmiş ve en doğru sonuç alınmaya çalışılmıştır.

```
X_train,X_test,Y_train,Y_test = train_test_split(X,Y,test_size=0.2, random_state=42)
```

Şekil 3.9. Eğitim ve test verilerinin bölünmesi

Bu kod satırı ile veri setini eğitim ve test setlerine bölme işlemini gerçekleştirilmiş ve aşağıda detaylı açıklanmıştır.

X: Bağımsız değişkenlerin (özelliklerin) bulunduğu veri setidir. Y: Bağımlı değişkenlerin (etiketlerin) bulunduğu veri setidir. Test_size: Veri setinin ne kadarının test seti olarak ayrılacağını belirleyen orandır. Bu durumda, veri setinin %20'si test seti olarak ayrılır (test_size=0.2). Random_state: Veri setini bölme işlemi sırasında kullanılacak rastgele bir oran değeridir [82]. Böylelikle aynı veri seti üzerinde yapılan bölme işlemleri sonuçları tekrarlanabilir. Eğitim ve test setleri, ayrı ayrı X_train, X_test, Y_train, ve Y_test değişkenlerine atanırlar. Bölünen bu setler daha sonrada modelin eğitilmesinde ve performans değerlendirilmesinde kullanılır. Bu tür bir veri bölme işlemi genellikle modelin eğitim verileri üzerinde öğrenmesini ve sonra test verileri üzerinde genelleme yapma yeteneğini değerlendirmesi için kullanılır. Train_test_split fonksiyonu, veriyi rastgele iki alt küme arasında böler. Random_state parametresi sayesinde bu bölme işleminin tekrarlanabilir.

3.5.2. Modelin Test Edilmesi

Eğitilen modelin 115.822URL'den oluşan test seti ile test edilmiş ve tahminler yapılmıştır. Doğruluk oranı hesaplandığından %91 olduğu görülmüştür. Bu oran kullanılan modelin yüksek hassasiyet ve doğruluğa sahip olduğunun en büyük kanıtı olmuştur.

```
test_sequences = tok.texts_to_sequences(X_test)
test_sequences_matrix = tf.keras.preprocessing.sequence.pad_sequences(test_sequences,maxlen=max_len)
accr = model.evaluate(test_sequences_matrix,Y_test)
```

Şekil 3.10 Matrix oluşumu

Bu kod bloğu modelin test verileri üzerinde performansını değerlendirmede kullanılmıştır ve detaylı açıklanmıştır. tok.texts_to_sequences (X_test) koduyla test verilerini tokenizer (tok) nesnesi kullanan texts_to_sequences fonksiyonu ile sayısal dizilere dönüştürür. Bu, metin verilerini önceden belirlenmiş bir kelime dizisi içindeki indislerle temsil edilen sayısal dizilere dönüştürür. tf.keras.preprocessing.sequence.pad_sequences (test_sequences,maxlen=max_len) bloğunda dönüştürülmüş sayısal dizileri belirli bir maksimum uzunluktaki diziye dönüştürür. Bu durum ise modelin giriş boyutlarına uygun hale getirilir. Eğer bir dizi maksimum uzunluktan daha kısa ise, eksik olan bölümler sıfırlarla doldurulur. model.evaluate(test_sequences_matrix, Y_test) bloğunda modelin test seti üzerinde performansını değerlendirir. test_sequences_matrix, test verilerini modelin anlayabileceği sayısal bir forma getiren matrisi, Y_test ise gerçek etiketleri içerir. evaluate fonksiyonu, modelin belirtilen veri üzerindeki performansını döndürür. Bu performans değerleri, modelin doğruluk ölçümünde ne kadar başarılı olduğunu gösterir.

Sonuçlar, genellikle bir liste şeklinde döner. Örneğin, [loss, accuracy] gibi. Eğer etiketli verilerle çalışılıyorsa, doğruluk (accuracy) değeri önemli bir performans ölçütü olabilir.

3.5.3. Modelin Performansı

```
print('Test seti\n Loss: {:0.3f}\n Accuracy: {:0.3f}'.format(accr[0],accr[1]))
```

Şekil 3.11 Doğruluk sonucunun yazdırılması

Bu kod ile modelin test seti üzerinde elde ettiği performansı ekrana yazdırmak için kullanılmış ve detaylı açıklanmıştır. print('Test seti\n Loss: {:0.3f}\n Accuracy

{:0.3f}'.format(accr[0], accr[1])) bloğunda formatlı bir şekilde ekrana yazdırmak için kullanılır. {} içine yerleştirilen değerler, süslü parantez içindeki sırayla format fonksiyonuna verilen argümanlardan gelir**.** '{:0.3f}' ile ondalık sayıları üç ondalık basamağa kadar yazdırmak için kullanılır. Bu, genellikle kayıp (loss) değeri için kullanılır [83]. '{:0.3f}' koduyla ondalık sayıları üç ondalık basamağa kadar yazdırmak için kullanılır. Bu, genellikle doğruluk değeri için kullanılır. accr[0] ve accr[1] sırasıyla kayıp ve doğruluk değerlerini içeren bir listenin elemanlarıdır. Bu değerler, modelin test seti üzerinde elde ettiği kayıp ve doğruluk değerlerini temsil eder. Bu çıktı, genellikle modelin performansını değerlendirmek için kullanılır.

4. BULGULAR VE TARTIŞMA

Bu bölümde, çalışmanın değerlendirme metrikleri ile ilgili bilgi verilmiş, doğruluk formülünden hesaplaması yapılmış, eğitim setinin tekrar işlemindeki doğruluk grafiği çıkarılmıştır. Yapılan işlemlerin sonunda test setinden doğruluk değerinin hesaplaması yapılmış ve değerlendirmesi yapılmıştır.

4.1. DEĞERLENDİRME METRİKLERİ

Modelin performansını değerlendirmek için kesinlik, duyarlılık, doğruluk ve f skoru metrikleri kullanılmıştır.

	Zararlı URL	**Zararsız URL**
Tahmin Edilen Zararlı URL	**Doğru Pozitif (TP)**	**Yanlış Pozitif (FP)**
Tahmin Edilen Zararsız URL	**Yanlış Negatif (FN)**	**Doğru Negatif (TN)**

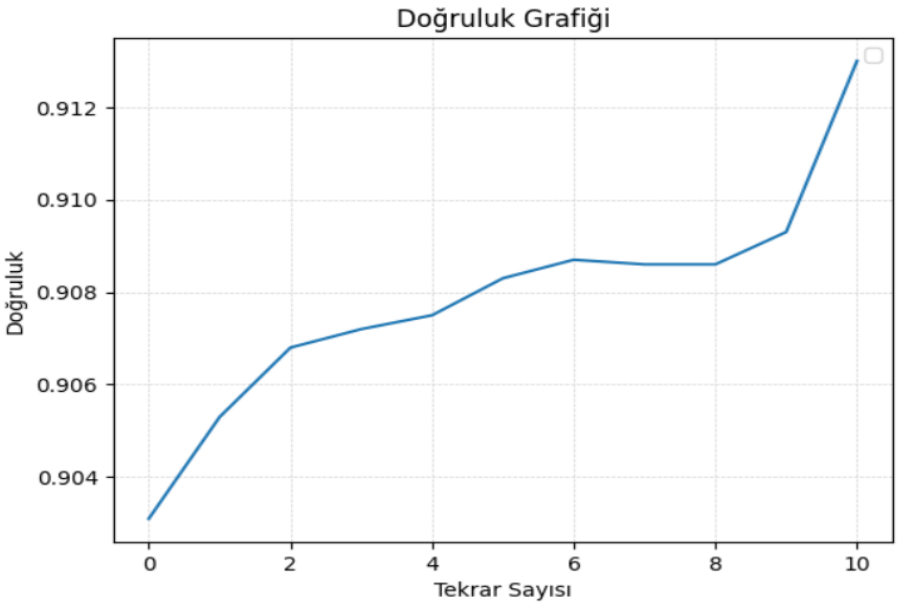
Şekil 4.1 Doğruluk Tablosu [43]

Doğru Pozitif (TP) ve Doğru Negatif (TN) modelin doğru olarak tahmin edildiği, Yanlış Pozitif (FP) ve Yanlış Negatif (FN) ise modelin yanlış olarak tahmin edildiği alanları göstermektedir. Bu alanların anlamları aşağıda verilmiştir.

- Doğru Pozitif (TP) :Doğru tahmin edilen zararlı URL,
- Doğru Negatif (TN): Doğru tahmin edilen zararsız normal URL,
- Yanlış Pozitif (FP): Yanlış tahmin edilen zararsız normal URL,
- Yanlış Negatif (FN): Yanlış tahmin edilen zararlı URL.

Doğruluk: Doğru olarak tahmin edilen zararlı URL sayısının, toplam URL sayısını oranı ifade eder. Yani, doğru sınıflandırılmış tüm URL'lerin toplam URL'ler içindeki oranını gösterir. Doğruluk, 0 ila 1 arasında değer alır. Eğer bu oran yüksek ise sonucun doğruluğunun yüksek olduğunu ifade eder. Hesaplama formülü aşağıdaki gibidir [84].

$$Doğruluk = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{FP} + \mathrm{TN} + \mathrm{FN}} \quad (4.1)$$



Şekil 4.2. Eğitim Seti Doğruluk Başarı Oranı

Yapılan çalışmada şekil 4.2. de görüldüğü gibi eğitim setinin tekrar sayısına artıkça doğruluk başarı oranı artış görülmüştür. Veri setinin eğitimi tamamlandıktan sonra test veri setinin doğruluk başarı oranı %91 çıktığı görülmüştür.

Kesinlik: Doğru olarak tahmin edilen kötücül URL sayısının, tüm kötücül URL sayısına oranını ifade eder. Hesaplama formülü aşağıdaki gibidir [85].

$$\text{Keskinlik} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}} \quad (4.2)$$

Duyarlılık: Doğru olarak tahmin edilen zararlı URL sayısının, yanlış olarak tahmin edilen zararlı olmayan URL sayısına oranı olarak ifade edilir [86].

Aynı zamanda "duyarlılık" veya "hassasiyet" olarak da adlandırılır. Duyarlılık değeri her zaman 0 ila 1 arasında bir değer alır. Duyarlılık değeri yüksek ise modelin pozitif sınıfı kaçırma olasılığının düşük olduğunu gösterir. Hesaplama formülü aşağıdaki gibidir.[87]

$$\text{Duyarlılık} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}} \quad (4.3)$$

F Skoru: Kesinlik ve duyarlılık değerlerini birlikte değerlendirmek için kullanılan değerdir. İki değerin harmonik ortalamasıdır. Hesaplama formülü aşağıdaki gibidir [88].

$$\text{F Skoru} = \frac{(2 \text{ x Kesinlik x Duyarlılık})}{(\text{Kesinlik+Duyarlılık})} \tag{4.4}$$

Yüksek doğruluk değeri modelin genel olarak tüm URL tipleri için başarılı bir sınıflandırma yeteneğine sahip olduğunu gösterir. Yüksek duyarlılık değeri yüksek olursa, ilgili tür için o kadar başarılı tanıma yaptığını ifade eder. Kesinlik değeri yüksek olursa, ilgili tür için o kadar yüksek doğruluk değerinin elde edildiğini garanti eder. F skoru değeri yüksek olursa da sistemini tüm türler için sınıflandırma sonuçlarındaki kararlılığının o kadar yüksek olduğunu gösterir [89].

5. SONUÇLAR VE ÖNERİLER

İnternet kullanımı her geçen yıl giderek artmış ve hayatımızın her alanından vazgeçilmezi olmuştur. Özellikler son yıllarda gelişen teknolojik akıllı cihazlar ve salgınında internet kullanımı artmasına sebep olmuştur. Artık internet sadece kullandığımız akıllı cep telefonlarımızda değil, evlerimizde kullandığımız akıllı televizyonlar, giyilebilir cihazlar, akıllı beyaz eşyalara kadar kullanılmakta, iş hayatımızdan eğitim isteminize e-ticaretten bankacılığa kadar vazgeçilmemiz olmuştur.

İnternetten kullanılan internet sayfalarının milyarlar ile ifade edilmesi, internetin güvenliğinin tartışılmasına sebep olmuştur. Özellikler internetin bu kadar kullanılması ve yaşamın en önemli vazgeçilmesi olması siber suçluların internet sitesi adreslerini bireysel kullanıcıların aldatabilmek için bu URL adreslerini saldırı hedefi olarak kullanmasını sebep olmuştur. Bu nedenle güvenli internet kullanımı için gerekli tedbirlerin alınması, kullanıcıların doğru ve güvenli internet için bilinçlendirilmesi ve zararlı URL adreslerinin erken tespit edilerek siber suçlara karşı önlem alınması zorunlu hale gelmiştir. Bununla ilgili çeşitli çalışmalar yapılmaktadır.

Bu çalışmada, zararlı URL adreslerinin tespit edilmesi ile ilgili çalışmalar yapılmış, güvenli internet adreslerine ulaşılabilmesi katkı sağlamak hedeflenmiştir. Çalışmada öncelikle daha geniş ve evrensel olması için Kaggle'da bulunan CC0: Public Domain Lisanslı 411.247 URL adresinden oluşan karma veri seti alınmıştır. Daha sonra bu veri setine Ulusal veri seti de eklemek için Some'de bulunan 168.867 URL adresinden oluşan zararlı veri seti ile birleştirilmiştir. Aynı özelliklere sahip iki ayrı veri seti birleştirilerek 579.112 adet URL'den oluşan büyük bir yeni veri seti oluşturulmuştur. Oluşturulan veri setinin %37'i zararlı URL adresleri ve %63'ü zararlı olmayan URL adresleridir. Elde edilen bu veri setlerinin çalışması için 7 katmanlı RNN model oluşturulmuştur. Bu modelde katmalarda doğruluk oranının artırmak için gerekli tanımlamalar yapılmış, oluşturulan veri seti modelde karıştırılmıştır. Oluşturulan veri seti %20 eğitim seti ve %80 ise test verisi olarak kullanılmıştır.

Modellerin eğitim ve test işlemleri Google Colaboratory platformu üzerinde gerçekleştirilmiştir. GPU desteği ile bu modellerin eğitimi gerçekleştirilmiştir. Bu platformda modellerin eğitimleri oldukça hızlı bir şekilde gerçekleşmiştir. Modeli oluştururken maksimum 150 karakter uzunluğunda ve en fazla 50 kelime türetebilen bir

gömme katmanı, 1x64 boyutunda bir özellik vektörü çıkaran bir LSTM katmanı ve 256 düğümlü bir çıkış katmanı kullanılmıştır. Modelde tekrarlama sayısı 10 seçilerek veri setinin 10 kez tekrarlanması sağlanarak başarı yüzdesinin yüksek olması hedeflenmiştir.

Modellerin ürettiği çıktı performans değerleri oluşturulan 7 katmanlı RNN modeli, %91 doğruluk oranı veren başarılı derin öğrenme modeli olmuştur. Ayrıca eğitim yaparken geçen süre de dikkate alındığı zaman RNN modelinin hızlı eğitimini tamamladığı görülmektedir.

Gelecekteki planlanan çalışmalarda, Türkiye'deki kayıtlı URL adreslerinin tamamından oluşacak yeni veri seti ile SOME'den alınacak zararlı URL adresleri veri seti birleştirilerek daha büyük veri seti hazırlanabilir ve modelde eğitilebilir. Veri seti dışındaki URL adreslerinin modelde sorgulanması yapılarak test edilebilir. Sonuç olarak, bu çalışma ile zararlı URL adreslerinin tespit edilmesine ve gelecekte yapılacak olan benzer çalışmalara ışık tutacaktır.

6. KAYNAKLAR

[1] N. A. Alfouzan ve C. Narmatha, "A Systematic Approach for Malware URL Recognition", *Proc. 2022 2nd Int. Conf. Comput. Inf. Technol. ICCIT 2022*, ss. 325–329, 2022, doi: 10.1109/ICCIT52419.2022.9711614.

[2] F. A. Ghaleb, M. Alsaedi, F. Saeed, J. Ahmad, ve M. Alasli, "Cyber Threat Intelligence-Based Malicious URL Detection Model Using Ensemble Learning", *Sensors*, c. 22, sayı 9, ss. 1–19, 2022, doi: 10.3390/s22093373.

[3] S. J. Bu ve H. J. Kim, "Optimized URL Feature Selection Based on Genetic-Algorithm-Embedded Deep Learning for Phishing Website Detection", *Electron.*, c. 11, sayı 7, 2022, doi: 10.3390/electronics11071090.

[4] Z. Chen, Y. Liu, C. Chen, M. Lu, ve X. Zhang, "Malicious URL Detection Based on Improved Multilayer Recurrent Convolutional Neural Network Model", *Secur. Commun. Networks*, c. 2021, 2021, doi: 10.1155/2021/9994127.

[5] S. H. Ahammad *vd.*, "Phishing URL detection using machine learning methods", *Adv. Eng. Softw.*, c. 173, sayı September, s. 103288, 2022, doi: 10.1016/j.advengsoft.2022.103288.

[6] A. Pandey ve J. Chadawar, "Phishing URL Detection using Hybrid Ensemble Model", *Artic. Int. J. Eng. Tech. Res.*, c. 11, sayı 04, ss. 479–482, 2022, [Çevrimiçi]. Available at: https://www.researchgate.net/publication/360412387

[7] C. Amrutkar, Y. S. Kim, ve P. Traynor, "Detecting Mobile Malicious Webpages in Real Time", *IEEE Trans. Mob. Comput.*, c. 16, sayı 8, ss. 2184–2197, 2017, doi: 10.1109/TMC.2016.2575828.

[8] H. Zhao, Z. Chen, ve R. Yan, "Malicious Domain Names Detection Algorithm Based on Statistical Features of URLs", *2022 IEEE 25th Int. Conf. Comput. Support. Coop. Work Des. CSCWD 2022*, ss. 11–16, 2022, doi: 10.1109/CSCWD54268.2022.9776264.

[9] R. Bharadwaj, A. Bhatia, L. D. Chhibbar, K. Tiwari, ve A. Agrawal, "Is this URL Safe: Detection of Malicious URLs Using Global Vector for Word Representation", *Int. Conf. Inf. Netw.*, c. 2022-Janua, ss. 486–491, 2022, doi: 10.1109/ICOIN53446.2022.9687204.

[10] S. Vecile, K. Lacroix, K. Grolinger, ve J. Samarabandu, "Malicious and Benign URL Dataset Generation Using Character-Level LSTM Models", *5th IEEE Conf. Dependable Secur. Comput. DSC 2022 SECSOC 2022 Work. PASS4IoT 2022 Work. SICSA Int. Pap. Compet. Cybersecurity*, ss. 1–8, 2022, doi: 10.1109/DSC54232.2022.9888835.

[11] R. Chiramdasu, G. Srivastava, S. Bhattacharya, P. K. Reddy, ve T. Reddy Gadekallu, "Malicious url detection using logistic regression", *2021 IEEE Int. Conf. Omni-Layer Intell. Syst. COINS 2021*, ss. 1–6, 2021, doi: 10.1109/COINS51742.2021.9524269.

[12] C. R. Vyawhare, R. Y. Totare, P. S. Sonawane, ve P. B. Deshmukh, "Machine Learning System for Malicious Website Detection: A Literature Review", *Int. J. Res. Appl. Sci. Eng. Technol.*, c. 10, sayı 5, ss. 56–61, 2022, doi: 10.22214/ijraset.2022.42050.

[13] M. Mehndiratta, N. Jain, A. Malhotra, I. Gupta, ve R. Narula, “Malicious URL: Analysis and Detection using Machine Learning”, *Proc. 17th INDIACom; 2023 10th Int. Conf. Comput. Sustain. Glob. Dev. INDIACom 2023*, ss. 1461–1465, 2023.

[14] A. PARLAK, “İnternet ve Türkiye’de İnterneti Gelişimi”, *Firat Üniversitesi Mühendisliik Fakültesi*, c. 1, sayı 1, ss. 1–87, 2005, [Çevrimiçi]. Available at: http://www.hasanbalik.com/projeler/bitirme/39.pdf

[15] A. M. İ, “Türkiye İnternet Raporu 2007”, *XII. “Türkiye’de İnternet” Konf. 8-10 Kasım 2007, Ankara TÜRKİYE*, ss. 175–183, 2007.

[16] G. Andersson, “Critical Rationalism and the Principle”, *J. Philos. Investig.*, c. 17, ss. 21–30, 2023.

[17] T. İçten ve G. Bal, “TCP/IP ve OSI modeli öğretimi için etkileşimli artırılmış gerçeklik ortamı geliştirme”, *Pamukkale Univ. J. Eng. Sci.*, c. 29, sayı 2, ss. 194–208, 2023, doi: 10.5505/pajes.2022.36605.

[18] N. Kostopoulos, D. Kalogeras, D. Pantazatos, M. Grammatikou, ve V. Maglaris, “SHAP Interpretations of Tree and Neural Network DNS Classifiers for Analyzing DGA Family Characteristics”, *IEEE Access*, c. 11, sayı June, ss. 61144–61160, 2023, doi: 10.1109/ACCESS.2023.3286313.

[19] B. ER ve L. KUŞAK, “RİS Takibinde Web Tabanli Haritalarin Kullanilmasi: Korona Virüs Web Haritasi Örneği”, *Mühendislik Bilim. ve Tasarım Derg.*, c. 11, sayı 3, ss. 886–903, 2023, doi: 10.21923/jesd.1245273.

[20] Ö. Arısoy, “İnternet Bağımlılığı ve Tedavisi”, *Psikiyatr. Güncel Yaklaşımlar*, c. 1, sayı 1, ss. 55–67, 2009, [Çevrimiçi]. Available at: http://dergipark.ulakbim.gov.tr/pskguncel/article/view/5000076432

[21] K. MERDAN, “Factors Affecting E-Commerce Transaction Volume Based on a Multiple Regression Model (Case of Turkey)”, *Anadolu Üniversitesi İktisadi ve İdari Bilim. Fakültesi Derg.*, c. 24, sayı 3, ss. 231–260, 2023, doi: 10.53443/anadoluibfd.1176486.

[22] Y. KOCATÜRK, “Bankacılıkta Dijitalleşmenin Etkileri ve Türkiye’deki Analizi”, *Ekon. ve Finans. Araştırmalar Derg.*, c. 5, sayı 1, ss. 38–50, 2023, doi: 10.56668/jefr.1310735.

[23] F. AKBAS, “Bankacılıkta Dijital Dönüşüm ve FinTech”, *Uluslararası Ekon. Araştırmalar Derg.*, c. 4, sayı 3, ss. 23–41, 2023, [Çevrimiçi]. Available at: https://dergipark.org.tr/tr/download/article-file/792932

[24] Ş. Uzundağ, “Türkiye’de İnternet Bankacılığının Gelişimi ve İnternet Bankacılığına İlişkin Tüketici Davranışları Analizi: Aydın İli Merkezinde Görev Yapan Öğretmenler Üzerine Bir Araştırma”, *, Yüksek Lisans Tezi, Adnan Menderes Üniversitesi Sos. Bilim. Enstitüsü*, c. Aydın, Tür, 2013.

[25] S. D. İbrahim Halil SUGÖZÜ, *“İnternet Teknolojisi ve Elektronik Ticaret”*, sayı Temmuz. Nobel Akademik Yayıncılık Eğitim Danışmalığı Tic. Ltd. Şti., 2013.

[26] N. Atalay, “Web Tarayıcılar Aracılığı ile Erişim Sağlanan Canlı Yayın Platformlarında İşlenen Suçların Adli Analizi”, *Int. J. Innov. Eng. Appl. vol. 6, issue 2 Int.*, c. 6, sayı 2, 2022.

[27] F. T. Ngo, C. Marcum, ve S. Belshaw, “The Dark Web: What Is It, How to Access It, and Why We Need to Study It”, *J. Contemp. Crim. Justice*, c. 39, sayı 2, ss. 160–166, 2023, doi: 10.1177/10439862231159774.

[28] N. ÖZDENER, “Web Tasarımında Editör Kullanımı ve HTML Öğretimi”, *Dergipark*, ss. 107–118, 2008.

[29] T. F. Koloğlu, “Web tasarımında işlem basamakları ve renk seçimlerinde bilinmesi gerekenler”, *Küresel Mühendislik Çalışmaları Derg.*, c. 2, sayı 2, ss. 51–61, 2015.

[30] B. O. AYDIN ve S. Gürbüz, “Türkiye’de Kadınların Karşılaştığı Sorunlar ve Köprü Ağ Analizi Yöntemiyle İncelenmesi”, *Anemon Muş Alparslan Üniversitesi Sos. Bilim. Derg.*, c. 6, sayı 4, ss. 579–586, 2018, doi: 10.18506/anemon.381770.

[31] H. Çakır, “Çocuklarda Vergi Bilinci Oluşturmaya Yönelik İnternet Sayfa Tasarımı”, *Ticaret ve Tur. Eğitim Fakültesi Derg.*, ss. 18–34, 2010.

[32] E. Uzun, Y. Kiliçaslan, ve E. Uçar, “Html, XML ve Web Servislerinin İnternet Sunucuları Üzerindeki Etkisinin İncelenmesi”, *Trak. Univ J Sci, 8(2) 81-85, 2008 ISSN 1305–6468 DIC 225EUET820712070108 Araştırma*, c. 8, sayı 2, ss. 81–85, 2008.

[33] D. Ö. Ü. B. ERDEM, “Bulut bilişim uygulama maliyetlerinin, müşteri işletmeler tarafından muhasebeleştirilmesi”, *Muhasebe ve Denetime BAKIŞ*, sayı 59, ss. 233–252, 2020.

[34] O. A. ERDEM ve R. KOCAOĞLU, “Yeni Bir Ağ Güvenliği Yaklaşimi Dinamik Zeki Güvenlik Duvarı Mimarisi”, *Gazi Üniversitesi Mühendislik-Mimarlık Fakültesi Derg.*, c. 29, sayı 4, ss. 707–715, 2014, doi: 10.17341/gummfd.18495.

[35] K. D. Atalay, M. Bilgisi, ve A. Info, “Web sitesi tasarım aşamasındaki kriterlerin önem derecelerinin GB- FUCOM ile belirlenmesi”, *Ankara Yıldırım Beyazıt Ünv. JTOM(6)1, 1010, 1027, 2022*, ss. 1010–1027, 2022.

[36] C. Çakir ve H. Kaptan, “VoIP Teknolojilerinde Opnet Tabanlı Güvenlik Uygulaması Opnet Based Security Application in VoIP Technologies”, *Bilişim Teknol. Derg.*, ss. 1–7, 2009.

[37] N. B. Sayyed, “Tcp/Ip Protocol Suite for Network Communication”, *Int. Res. J. Mod. Eng. Technol. Sci.*, sayı 04, ss. 2911–2915, 2023, doi: 10.56726/irjmets36200.

[38] N. Gozuacik, “Implementation of Link-local Registration Protocol for IEEE 802.1 Time-Sensitive Networking”, *31st IEEE Conf. Signal Process. Commun. Appl. SIU 2023*, ss. 1–4, 2023, doi: 10.1109/SIU59756.2023.10223823.

[39] D. Daniel, “Bilgi İletişim Teknolojileri ile Ekonomik Büyüme Arasındaki İlişki: Türkiye için bir Uygulama”, *Uluslararası Ticaret ve Ekon. Araştırmaları Derg.*, c. 151, ss. 10–17, 2015.

[40] TUIK, “Hanehalkı Bilişim Teknolojileri (BT) Kullanım Araştırması, 2023”, *Tuik*, [Çevrimiçi]. Available at: https://data.tuik.gov.tr/bulten/hanehalki

[41] E-devlet, “e-Devlet Kapısı Kullanıcı İstatistikleri”, *turkiye.gov.tr*, ss. 30–31, 2023, [Çevrimiçi]. Available at: https://www.turkiye.gov.tr/edevlet-istatistikleri

[42] N. P. and D. S. David Bell, Brian D. Loadera, *Cyberculture*. 1967.

[43] F. TİRYAKİ, Ü. ŞENTÜRK, ve İ. YÜCEDAĞ, "Developing and Evaluating an Artificial Intelligence Model for Malicious URL Detection", *Eur. J. Sci. Technol.*, sayı 47, ss. 13–17, 2023, doi: 10.31590/ejosat.1234556.

[44] A. KOÇYİĞİT ve A. B. DARI, "Yapay Zeka İletişiminde Chatgpt:İnsanlaşan Dijitalleşmenin Geleceği", *SSAD Strat. ve Sos. Araştırmalar Derg.*, c. 7, sayı 2, 2023, doi: 10.30692/sisad.1311336.

[45] GÖLBAŞI B.T., "E-Ticaret Projelerinde Domain (Alan Adı) Yenilikleri Üzerine Bir İnceleme", *Open Access Ref. E-Journal Ref. Index. http//www.ssdjournal.org / journalssd@gmail.com Artic.*, ss. 149–163, 2022.

[46] M. E. D. ÖZBAY, Z.N., "nDPI Derin Paket İnceleme Aracı Üzerinde Bir Çalışma", *Bilgi. Bilim. ve Mühendisliği Derg.*, ss. 137–146, 2023.

[47] K. Goel, J. Shankar Prasad, ve S. Hilal, "Removing Duplicate URLs based on URL Normalization and Query Parameter", *Int. J. Eng. Technol.*, c. 7, sayı 3.12, s. 361, 2018, doi: 10.14419/ijet.v7i3.12.16107.

[48] M. Liao, X. Liu, ve T. Jia, "Characterizing temporally fragmented human activity networks in cyber space using uniform resource locator (URL) data", *Int. J. Digit. Earth*, c. 17, sayı 1, 2024, doi: 10.1080/17538947.2023.2295986.

[49] Y. H. Sunumu, "The Efect of Artificial Intelligence on Local Service", *Tokat Gaziosmanpaşa Ünv. Sos. Bilim. Araştırmaları Derg.*, c. 2, sayı June, 2023.

[50] Z. Yao *vd.*, "Machine learning for a sustainable energy future", *Nat. Rev. Mater.*, c. 8, sayı 3, ss. 202–215, 2023, doi: 10.1038/s41578-022-00490-5.

[51] J. Yu, H. Yin, X. Xia, T. Chen, J. Li, ve Z. Huang, "Self-Supervised Learning for Recommender Systems: A Survey", *IEEE Trans. Knowl. Data Eng.*, c. 36, sayı 1, ss. 335–355, 2024, doi: 10.1109/TKDE.2023.3282907.

[52] S. ARSLANKAYA ve Ş. TOPRAK, "Using Machine Learning and Deep Learning Algorithms for Stock Price Prediction", *Uluslararası Muhendis. Arastirma ve Gelistirme Derg.*, c. 13, sayı 1, ss. 178–192, 2021, doi: 10.29137/umagd.771671.

[53] Serkan KIRCA, "Derin Öğrenme Yöntemi ile Araç ve Plaka Tanıma", *Kocaeli Ünv. Fen Bilim. Enst. Bilişim Sist. Müh. Anabilim Dalı*, 2021.

[54] M. Krichen, "Convolutional Neural Networks: A Survey", *Computers*, c. 12, sayı 8, ss. 1–41, 2023, doi: 10.3390/computers12080151.

[55] Q. Zhang, J. Xiao, C. Tian, J. Chun-Wei Lin, ve S. Zhang, "A robust deformed convolutional neural network (CNN) for image denoising", *CAAI Trans. Intell. Technol.*, c. 8, sayı 2, ss. 331–342, 2023, doi: 10.1049/cit2.12110.

[56] F. DOĞAN ve İ. TÜRKOĞLU, "Derin Öğrenme Modelleri ve Uygulama Alanlarına İlişkin Bir Derleme", *DÜMF Mühendislik Derg.*, c. 10, sayı 2, ss. 409–445, 2019, doi: 10.24012/dumf.411130.

[57] M. Hibat-allah, R. G. Melko, ve J. Carrasquilla, "Investigating Topological Order using Recurrent Neural Networks", *arXiv:2301.11207v3*, ss. 1–15, 2023, [Çevrimiçi]. Available at: https://link.springer.com/10.1007/978-1-0716-3195-9

[58] Ç. Çoban ve E. Hayat, "Hisse Senedi Piyasası Analizinde Farklı Derin Sinir Ağı Modellerinin Karşılaştırılması", *Aydın Adnan Menderes Üniversitesi, Sos. Bilim. Enstitüsü Dergisi, Yıl 2023*, c. 2, sayı 0009, ss. 120–139, 2023.

[59] E. Brissman, J. Johnander, M. Danelljan, ve M. Felsberg, "Recurrent Graph Neural Networks for Video Instance Segmentation", *Int. J. Comput. Vis.*, c. 131, sayı 2, ss. 471–495, 2023, doi: 10.1007/s11263-022-01703-8.

[60] B. Alfason *vd.*, "Forecasting of Covid-19 positive cases in Indonesia using long Forecasting of Covid-19 positive cases in Indonesia using long memory (LSTM) short-term memory (LSTM)", *Procedia Comput. Sci.*, c. 216, sayı 2022, ss. 177–185, 2023, doi: 10.1016/j.procs.2022.12.125.

[61] C. Özden, "Forecasting Agricultural input Price Index Using Statistical and Deep Learning Methods İstatistiksel ve Derin Öğrenme Yöntemlerini Kullanarak Tarımsal Girdi Fiyat Endeksi ' nin Tahmin Edilmesi", *Turkish J. Agric. - Food Sci. Technol.*, c. 11, sayı 9, ss. 1751–1755, 2023.

[62] T. Uçaklar, Ö. Tabanlı, H. Hızı, ve T. Sistemi, "Deep Learning-Based Airspeed Estimation System for a Commercial Aircraft", *J. Aeronaut. Sp. Technol.*, c. 16, sayı 2, ss. 20–35, 2023.

[63] R. S. Örnek ve Z. H. Tuğcu, "Akıllı Şebeke Uygulamalarında Derin Öğrenme Tekniklerinin Kullanımına İlişkin Kısa Bir İnceleme A Brief Review on the Use of Deep Learning Techniques in Smart Grid Applications", *EMO Bilim. Dergi*, c. 13, sayı 1, ss. 41–61, 2023.

[64] E. AYDINGÖZ ve M. BAL, "Tomosentez Görüntüleri ile Yapılan Derin Öğrenme Çalışmalarında Kullanılan Görüntü Ön İşleme Yöntemleri Üzerine Bir Literatür Araştırması", *Eur. J. Sci. Technol.*, sayı 51, ss. 352–367, 2023, doi: 10.31590/ejosat.1312965.

[65] Z. ping Song, Y. Cheng, Z. kun Zhang, ve T. tian Yang, "Tunnelling performance prediction of cantilever boring machine in sedimentary hard-rock tunnel using deep belief network", *J. Mt. Sci.*, c. 20, sayı 7, ss. 2029–2040, 2023, doi: 10.1007/s11629-023-7931-y.

[66] S. Paheding ve A. A. Reyes-Angulo, "Forward-Forward Algorithm for Hyperspectral Image Classification: A Preliminary Study", *arXiv:2307.00231v1*, ss. 1–10, 2023, [Çevrimiçi]. Available at: http://arxiv.org/abs/2307.00231

[67] M. Er, Burak, Doğan, "Pekiştirmeli Öğrenme Kontrolde Otokodlayıcılar ile örnek verimliliğin arttırma", *Istanbul Tek. Üniversitesi*, 2023.

[68] D. Küçük ve N. Arici, "Doğal Dil İşlemede Derin Öğrenme Uygulamaları Üzerine Bir Literatür Çalışması", *Int. J. Manag. Inf. Syst. Comput. Sci.*, c. 2, sayı 2, ss. 76–86, 2018.

[69] I. Czinege ve D. Harangozó, "Application of artificial neural networks for characterisation of formability properties of sheet metals", *Int. J. Light. Mater. Manuf.*, c. 7, sayı 1, ss. 37–44, 2024, doi: 10.1016/j.ijlmm.2023.08.003.

[70] Y. Tian, Y. Zhang, ve H. Zhang, "Recent Advances in Stochastic Gradient Descent in Deep Learning", *Mathematics*, c. 11, sayı 3, ss. 1–23, 2023, doi: 10.3390/math11030682.

[71] F. YETİZ, M. TERZİOĞLU, ve M. KAYAKUŞ, "Makina Öğrenmesi Yöntemleri ile Türk Mevduat Bankalarının Müşteri Tahminine Yönelik Bir Uygulama", *Sosyoekonomi*, c. 29, sayı 50, ss. 413–432, 2021, doi: 10.17233/sosyoekonomi.2021.04.19.

[72] CC0: Public Domain, “Url Dataset”, *Kaggle*, 2017, [Çevrimiçi]. Available at: https://www.kaggle.com/datasets/teseract/urldataset

[73] U. D. SET, “Url Dataset”, *some*, 2022, [Çevrimiçi]. Available at: https://www.usom.gov.tr/adres./12.09.2022

[74] H. Kocaman, Z. Garip, ve A. Ozden, “2nd International Conference on Innovative Academic Studies Derin Öğrenmeye Dayalı Nesne Tanıma Yöntemi ile Hava Fotoğrafı Üzerinden Karşılaştırmalı Bina Tespiti”, *ICIAS 2nd Int. Conf. Innov. Acad. Stud.*, sayı January, 2023.

[75] A. Ö. Türkçetin, T. Koç, ve Ş. Çilekar, “Akciğer Hastalıklarının Ses ile Tespitinde YSA Kullanımı: KOAH, Astım, Pnömoni Örneği”, *31st IEEE Conf. Signal Process. Commun. Appl. SIU 2023*, ss. 19–22, 2023, doi: 10.1109/SIU59756.2023.10223781.

[76] N. ÇEKİÇ, “Çocuklukta Zatürre Hastalığının Gögüs Röntgen Görüntülerinden Derin Öğrenme ile Tespiti",Yüksek Lisans Tezi, Düzce Üniversitesi Lisansüstü Ens.”, 2023.

[77] F. Tambon, A. Nikanjam, L. An, F. Khomh, ve G. Antoniol, *Silent bugs in deep learning frameworks: an empirical study of Keras and TensorFlow*, c. 29, sayı 1. 2024. doi: 10.1007/s10664-023-10389-6.

[78] M. Waskom, “Seaborn: Statistical Data Visualization”, *JOSS J. Open Source Softw.*, c. 6, sayı 60, s. 3021, 2021, doi: 10.21105/joss.03021.

[79] U. Senturk, I. Yucedag, ve K. Polat, “Repetitive neural network (RNN) based blood pressure estimation using PPG and ECG signals”, *ISMSIT 2018 - 2nd Int. Symp. Multidiscip. Stud. Innov. Technol. Proc.*, ss. 1–4, 2018, doi: 10.1109/ISMSIT.2018.8567071.

[80] G. Keren ve B. Schuller, “Convolutional RNN: An enhanced model for extracting features from sequential data”, *Proc. Int. Jt. Conf. Neural Networks*, c. 2016-Octob, ss. 3412–3419, 2016, doi: 10.1109/IJCNN.2016.7727636.

[81] Y. Bai *vd.*, “How Important is the Train-Validation Split in Meta-Learning?”, *Proc. Mach. Learn. Res.*, c. 139, ss. 543–553, 2021.

[82] A. Boudjella, M. Y. Boudjella, B. Bellebna, ve D. El-Kebir, “Machine Learning-Based Classification of Chest X-Ray MRI Images into Covid-19 Graphic User Interface”, *2022 7th Int. Conf. Image Signal Process. their Appl. ISPA 2022 - Proc.*, ss. 1–7, 2022, doi: 10.1109/ISPA54004.2022.9786349.

[83] E. J. Barnett, D. G. Onete, A. Salekin, ve S. V Faraone, “Genomic Machine Learning Meta-regression: Insights on Associations of Study Features with Reported Model Performance”, *IEEE/ACM Trans. Comput. Biol. Bioinforma.*, c. PP, ss. 1–18, 2023, doi: 10.1109/tcbb.2023.3343808.

[84] R. S. ARSLAN, “Kötücül URL Filtreleme için Derin Öğrenme Modeli Tasarımı”, *Eur. J. Sci. Technol.*, sayı 29, ss. 122–128, 2021, doi: 10.31590/ejosat.1011961.

[85] H. Karamollaoğlu, İ. Yücedağ, ve İ. A. Doğru, “Customer Churn Prediction Using Machine Learning Methods: A Comparative Analysis”, *Proc. - 6th Int. Conf. Comput. Sci. Eng. UBMK 2021*, ss. 139–144, 2021, doi: 10.1109/UBMK52708.2021.9558876.

[86] D. P. M Önder, Ü Şentürk, K Polat, “Diagnosis of alzheimer disease using classification algorithms”, *2023 Int. Conf. Res. Methodol. Knowl. Manag. Artif. Intell. Telecommun. Eng.*, ss. 725–734, 2023, doi: 10.1109/ICISS49785.2020.9316028.

[87] A. ALAN ve M. KARABATAK, “Veri Seti - Sınıflandırma İlişkisinde Performansa Etki Eden Faktörlerin Değerlendirilmesi”, *Fırat Üniversitesi Mühendislik Bilim. Derg.*, c. 32, sayı 2, ss. 531–540, 2020, doi: 10.35234/fumbd.738007.

[88] H. M ve S. M.N, “A Review on Evaluation Metrics for Data Classification Evaluations”, *Int. J. Data Min. Knowl. Manag. Process*, c. 5, sayı 2, ss. 01–11, 2015, doi: 10.5121/ijdkp.2015.5201.

[89] B. Yılmaz, “Makine Öğrenme Teknikleri Kullanılarak Basınç Ülserlerinin Siniflandirilmasi”, *Yüksek Lisans Tezi, Düce Üniversitesi Lisansüstü Enstitüsü Düzce, Türkiye*, 2021.

ÖZGEÇMİŞ

KİŞİSEL BİLGİLER

Adı Soyadı **:** Fatih TİRYAKİ

Yabancı Dili **:** İngilizce

ÖĞRENİM DURUMU

Derece	**Alan**	**Okul/Üniversite**	**Mezuniyet Yılı**
Y. Lisans	Siber Güvenlik	Düzce Üniversitesi	2024
Y. Lisans	Bilgisayar Müh.	Hoca Ahmet Yesevi Uluslararası Türk-Kazak Üniversitesi	2015
Lisans	Bilgisayar ve Öğretim Teknolojileri Öğretmenliği	Girne Amerikan Üniversitesi	2009
Ön Lisans	Bilgisayar Programcılığı	İstanbul Üniversitesi	2003
Lise	Bilgi-İşlem	Çorum Anadolu Tic. Meslek Lisesi	2001

YAYINLAR

A. Uluslararası Hakemli Dergide Basılan Makaleler:

F Tiryaki, Ü Şentürk, İ Yücedağ, “Kötü Amaçlı URL Tespiti İçin Yapay Zekâ Modelinin Geliştirilmesi ve Değerlendirilmesi” Avrupa Bilim ve Teknoloji Dergisi, sayı 47, sayfa 13–17, Ocak 2023.