
GAZİ ÜNİVERSİTESİ
Gazi
1926

# AÇIK KAYNAKLARDAN SİBER TEHDİT İSTİHBARATI VERİSİ ELDE EDİLMESİ

**Uğur TEKİN**

**YÜKSEK LİSANS**
**BİLGİ GÜVENLİĞİ MÜHENDİSLİĞİ ANA BİLİM DALI**

**GAZİ ÜNİVERSİTESİ**
**FEN BİLİMLERİ ENSTİTÜSÜ**

**KASIM 2021**

## ETİK BEYAN

Gazi Üniversitesi Fen Bilimleri Enstitüsü Tez Yazım Kurallarına uygun olarak hazırladığım bu tez çalışmasında;

- Tez içinde sunduğum verileri, bilgileri ve dokümanları akademik ve etik kurallar çerçevesinde elde ettiğimi,
- Tüm bilgi, belge, değerlendirme ve sonuçları bilimsel etik ve ahlak kurallarına uygun olarak sunduğumu,
- Tez çalışmasında yararlandığım eserlerin tümüne uygun atıfta bulunarak kaynak gösterdiğimi,
- Kullanılan verilerde herhangi bir değişiklik yapmadığımı,
- Bu tezde sunduğum çalışmanın özgün olduğunu,

bildirir, aksi bir durumda aleyhime doğabilecek tüm hak kayıplarını kabullendiğimi beyan ederim.

Uğur TEKİN

08/11/2021

# AÇIK KAYNAKLARDAN SİBER TEHDİT İSTİHBARATI VERİSİ ELDE EDİLMESİ

(Yüksek Lisans Tezi)

Uğur TEKİN

GAZİ ÜNİVERSİTESİ

FEN BİLİMLERİ ENSTİTÜSÜ

Kasım 2021

## ÖZET

Siber saldırıların her geçen gün artmasıyla birlikte siber güvenlik her kurum/kuruluş için giderek artan bir endişe haline gelmiştir. Küçük veya büyük her kurum/kuruluş siber tehditlere karşı önlem almakta ve yatırım yapmaktadır. Siber güvenliğin sağlanması, tehditlerin önlenmesi kapsamında güvenlik duvarları, saldırı tespit/önleme sistemleri, antivirüs, veri kaybı önleme yazımları vb. kullanılmaktadır ve siber tehditlerin tespit edilmesi maksadıyla bu cihaz/uygulamalardan elde edilen kayıtlar SIEM uygulamalarına gönderilmektedir. Siber tehditlerin doğru bir şekilde tespit edilebilmesi için siber tehdit istihbarat verilerinin söz konusu cihaz ve uygulamalarda kullanılması büyük önem arz etmektedir. İstihbarat olmadan savaş kazanmak ne kadar zor ise siber tehdit istihbaratı olmadan etkin bir siber güvenliğin sağlanması oldukça zordur. Siber tehdit istihbaratı ticari yazılımlar vasıtasıyla sağlanabileceği gibi açık kaynak platformlardan da elde edilebilmektedir. Bu çalışmada Twitter'dan elde edilen siber güvenlik verilerini işlemek için derin öğrenme modelleri kullanılmıştır. Özyinelemeli sinir ağları ile veri setinde yer alan tweetlerin siber tehdit istihbaratı ile ilgili olması durumu, ardından söz konusu siber tehdit istihbaratını (Ddos, malware, ransomware, vb.) sınıflandırması yapılmıştır. Çalışma sonucunda siber tehdit istihbaratı ilgili olup olmaması konusunda %88,64 tehdit istihbaratının türü sınıflandırılmasında ise %89,49 başarı elde edilmiştir.

| | | |
|---|---|---|
| Bilim Kodu | : | 92403 |
| Anahtar Kelimeler | : | Siber Tehdit İstihbaratı (CTI), Derin Öğrenme, Twitter, Doğal Dil İşleme, Açık Kaynak İstihbaratı |
| Sayfa Adedi | : | 67 |
| Danışman | : | Prof.Dr. Ercan Nurcan YILMAZ |

OBTAINING CYBER THREAT INTELLIGENCE DATA FROM OPEN SOURCES

(M. Sc. Thesis)

Uğur TEKİN

GAZİ UNIVERSITY

GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

November 2021

## ABSTRACT

With the increasing number of cyber-attacks, cyber security has become an increasing concern for every institution/organization. Every small or large institution/organization takes precautions and invests in cyber threats. Firewalls, intrusion detection/prevention systems, antivirus, data loss prevention software, etc., are used within the scope of ensuring cyber security and preventing threats. In order to detect cyber threats, the records obtained from these devices/applications are sent to SIEM applications. It is of great importance to use cyber threat intelligence data in these devices and applications in order to accurately detect cyber threats. As hard as it is to win a war without intelligence, it is also very difficult to provide effective cyber security without cyber threat intelligence. Cyber threat intelligence can be provided by commercial software as well as from open-source platforms. In this study, deep learning models were used to process the cyber security data obtained from Twitter. With recursive neural networks, the tweets in the data set are related to cyber threat intelligence, then the cyber threat intelligence (DDoS, malware, ransomware, etc.) is classified. As a result of the study, 88.64% success was achieved in terms of whether cyber threat intelligence is relevant or not, and 89.49% success was achieved in classifying the type of threat intelligence.

Science Code : 92403

Key Words : Cyber Threat Intelligence (CTI), Deep Learning, Twitter, Natural Language Processing (NLP), Open-Source Intelligence (OSINT)

Page Number : 67

Supervisor : Prof. Dr. Ercan Nurcan YILMAZ

# TEŞEKKÜR

Tezin hazırlanması aşamasında bana her zaman desteğini veren Prof.Dr. Ercan Nurcan YILMAZ'a, çalışma arkadaşlarıma, yüksek lisans boyunca göstermiş olduğu sevgi, saygı ve anlayışı ile bana her zaman destek olan; yakında aramıza katılacak olan oğlumuz Umut Alp'imizin annesi değerli eşim Fatmanur TEKİN'e sonsuz teşekkür ederim.

# İÇİNDEKİLER

**Sayfa**

**Sayfa**

# ÇİZELGELERİN LİSTESİ

**Çizelge** **Sayfa**

# ŞEKİLLERİN LİSTESİ

**Şekil** **Sayfa**

**Şekil** **Sayfa**

**Şekil** **Sayfa**

**Şekil** **Sayfa**

# SİMGELER VE KISALTMALAR

Bu çalışmada kullanılmış simgeler ve kısaltmalar, açıklamaları ile birlikte aşağıda sunulmuştur.

| Kısaltmalar | Açıklamalar |
|---|---|
| **API** | Application Programming Interface |
| **CNN** | Convolutional Neural Network |
| **CVE** | Common Vulnerabilities and Exposures7 |
| **DDoS** | Denial of Service |
| **DSA** | Derin Sinir Ağları |
| **GPS** | Global Positioning System |
| **GRU** | Gated Recurrent Unit |
| **IoC** | Indicator of Compromise |
| **IoT** | Internet of Things |
| **LSTM** | Long Short Term Memory |
| **NEM** | Named Entity Recognition |
| **NLP** | Natural Language Proccesing |
| **OSINT** | Open Source Intelligence |
| **RDF** | Resource Description Framework |
| **RMS** | Root Mean Square |
| **RNN** | Recurrent Neural Network |
| **SGD** | Stochastic Gradient Descent |
| **SML** | Statical Machine Learning |
| **SMO** | Sequential Minimal Optimization |
| **SOME** | Siber Operasyon Merkezi |
| **STİ** | Siber Tehdit İstihbaratı |
| **TTP** | Teknik, Taktik, Prosedür |
| **YSA** | Yapay Sinir Ağları |

# 1. GİRİŞ

Kaspersky firmasının yayınladığı rapora göre, 2020 yılında her gün ortalama 360 000 zararlı yazılım tespit edilmiş, bir önceki yıl ile karşılaştırıldığında ise %5,2 artış gösterdiği bildirilmiştir [1]. Zararlı yazılımların miktarı son yıllarda istikrarlı bir şekilde artarken, bilgi teknolojilerinde oluşan hasarın etkisi giderek daha belirgin hale gelmeye başlamıştır. 2017 yılının mayıs ayında ortaya çıkan ve başta hastaneler olmak üzere kritik altyapılarda çalışan 200 000'den fazla cihaza enfekte olan WannaCry fidye yazılımı oldukça dikkat çekmiştir [2]. Bu olaydan sadece bir ay sonra, Haziran 2017'de, özellikle Ukrayna'daki şirketleri hedef alan ve diğer ülkelere de yayılan NotPetya adlı ikinci bir fidye yazılımı ortaya çıkmıştır [3]. Her iki olayda da kamu kurumlarının durumla ilgili uyarı yayınlamasından saatler önce, sosyal medya kullanıcıları vakalar ve tehditler hakkında bilgi paylaşılmıştır.

Sosyal ağlar, dünyanın her yerinden kullanıcıların her türlü bilgiyi oluşturması ve paylaşması sebebiyle olay tespiti için oldukça önemli bir veri kaynağı oluşturmaktadır. Bununla birlikte, kullanıcılar tarafından oluşturulan içerikler genellikle belli bir yapısı olmayan, kısa ve güvenilmez verilerdir. Bu sebeple yazılım algoritmalarının istenilen bilgiyi doğrudan elde etmesini neredeyse imkânız hale getirir [4]. Bu problemi çözmek için araştırmacılar, büyük miktarda doğal dil verisindeki kalıpları veya istatistiksel dağılımları tespit etmek için yapay zekâya dayalı yöntemler geliştirmişlerdir. Bu yöntemlerin kullanılması ile sosyal medya aracılığıyla kötü amaçlı yazılımlar, hizmet engelleme saldırıları (DDoS), sıfırıncı gün açıklıkları gibi siber tehdit istihbaratı olarak değerlendirilebilecek verilere kolaylıkla ulaşılabilmesine imkân sağlamıştır.

Günümüzde birçok kuruluş, tehdit ortamı hakkında genel bir bakış sağlamak için siber tehdit istihbarat çözümlerini kullanmaktadır. Birçok ticari tehdit istihbaratı kaynağı, şirkete ait analiz verilerine dayansa da tehdit bilgilerinin kamuya açık olarak paylaşılması giderek daha fazla önem kazanmaktadır.

Siber tehdit analistleri, yeni keşifler ve güvenlik ihlalleri hakkında bilgi alışverişinde bulunmak için topluluklar oluşturmaya başlamıştır. Güvenlik araştırmacıları için ilk referans, genellikle hızlı hareket eden platform Twitter'dır. Twitter'dan elde ettikleri

kullanıcı katkılarını zenginleştirmek ve diğer araştırmacıların yeni bir tehdidi araştırmasını sağlamak için genellikle daha fazla materyali ilişkilendirirler. Bu sebeple, VirusTotal, URLhaus, Hybrid Analysis gibi çevrimiçi analiz platformları, kötü amaçlı yazılımı yalıtılmış bir ortamda çalıştırır ve yakalanan davranışı topluluğa sağlar. Genel olarak, günümüzde tehdit istihbaratı geniş bir alana yayılmış olmasına rağmen dağınık ve düzensiz olarak bulunmaktadır. Sonuç olarak, web üzerindeki bilgi parçalarını kapsamlı bir şekilde toplamak, işlemek ve yapılandırmak için bir sistem gereklidir.

Antivirüs firmaları zararlı yazılımları tespit edip veri tabanlarına imzalarını ekledikçe zararlı yazılımları üreten aktörler yeni yazılımlar piyasaya sürmektedir. Örneğin, ilk olarak 2014 yılında keşfedilen gelişmiş bir bankacılık truva atı olan Emotet, kurbanların makinesinde algılanmamak için tasarımını sürekli olarak değiştirmektedir [5]. Sosyal ağlar ve genel analiz platformları, sık sık yeni kötü amaçlı yazılım türevlerini bildirmekte ve daha fazla analiz için ilgili tehdit göstergelerini (IoC) paylaşmaktadır. Ancak, sosyal medya platformlarında yayınlanan güvenlikle ilgili haber miktarının fazla olması sebebiyle her bir olayı tek başına manuel olarak incelemek oldukça zaman almaktadır. Özellikle, antivirüs şirketleri, virüs tespit oranını artırmak için kötü amaçlı yazılım örneklerinin dosya içerikleri ve sıralamasıyla ilgilenir. Müşterileri korumak için, tespit edilmemiş kötü amaçlı yazılım örneklerinden tehdit göstergesi (IoC) toplayarak ve analiz ederek otomatik bir süreç geliştirirler.

Tehdit istihbaratını toplamak için en değerli kaynaklar, kötü amaçlı yazılım paylaşım platformları ve sosyal ağlardır. Paylaşım platformlarından yapılandırılmış verileri işlemek oldukça önemlidir. Sosyal ağlardan doğal dilde paylaşılan verilerden bilinmeyen istihbaratın çıkarılması da oldukça önemli olmasına rağmen yapılandırılmış verilere göre daha karmaşıktır. Son araştırmalara göre, kullanıcılar tarafından oluşturulan içeriklerden güvenlik açıkları hakkında bilgi çıkarmanın mümkün olduğunu tespit edilmiştir. Bu sebeple, doğal dil işleme yöntemleri kullanılarak doğal dilde yapılan paylaşımlardan bilgi elde edilmesi çalışmaları yapılmaya başlanmıştır. Ancak şu ana kadar, kötü amaçlı yazılım analistlerinin tweetler gibi yapısal olmayan metinlerden türetilen IoC'yi otomatik olarak toplaması, zenginleştirmesi ve önceliklendirmesi için uygun bir çözüm geliştirilmeye çalışmaları devam etse de tam olarak verimli bir uygulama mevcut değildir.

Bu çalışmada oldukça zengin veri içeren Twitter açık kaynak platformdan Siber Tehdit İstihbaratı (STİ) elde edilmesi çalışması yapılmıştır. Twitter yaklaşık 200 milyon kullanıcı ve günlük atılan 500 milyon tweet ile oldukça büyük bir veri kaynağı oluşturmaktadır [6]. Twitter'da yer alan verilerin yapısal olmaması sebebiyle doğal dil işleme ve derin öğrenme yöntemlerine başvurulmuştur. Çalışmanın 2. bölümünde literatürde yer alan benzer çalışmalardan bahsedilmiş, 3. bölümünde ise metodoloji kısaca açıklanmıştır. 4. bölümde deney, sonuç ve gözlemler açıklanarak 5. bölümde sonuç ve elde edilen başarılardan bahsedilmiştir.

Çalışma, açık kaynak bir veri seti üzerinde gerçekleştirilmiştir. Çalışmada iki ayrı sınıflandırma yapılarak gerçekleştirilmiş olup ilk olarak verinin Siber Tehdit İstihbaratı ile ilgili olup olmadığı sınıflandırılmış ve %88,64 başarı elde edilmiştir. Daha sonra elde edilen istihbaratın türüne yönelik sınıflandırma yapılmış olup %89,49 başarı elde edilmiştir.

# 2. LİTERATÜR ARAŞTIRMASI

Açık kaynaklardan siber tehdit istihbaratı elde edilmesi kapsamında literatürde birçok çalışma yapılmıştır. Çalışmalarda veri kaynağı olarak darkweb, hacker formları, siber tehdit istihbaratı paylaşan platformlar kullanıldığı gibi son zamanlarda sosyal medya platformları özelliklede twitter kullanılmaya başlamıştır.

Liew, Sani, Abdullah, Yaakub ve Sharum Twitter'da oltalama amaçlı paylaşılan tweetlerin tespit edilmesine yönelik bir çalışma yapmıştır. Bu çalışmada makine öğrenmesi yöntemlerini kullanan bir model önermişlerdir [7]. Zenebe, Shumba, Carillo ve Cuenca darkweb de yer alan ve hackerlar tarafından paylaşılan verilerden birçok zafiyetin ortaya çıkarabileceğini ve siber tehdit istihbaratı açısından zengin bir kaynak olabileceğini öne sürmüştür. Bu çalışmada makine öğrenmesi yöntemlerini kullanan bir model önerilmiştir [8]. Rodriguez ve Okamura Multi layer keyword filtering ve doc2vec algoritmaları kullanarak daha başarılı bir şekilde açık kaynak istihbaratı elde edebileceğini öne süren bir model öne sürmüştür [9]. Subroto ve Apriyana siber riskleri tahmin etmek için sosyal medyada büyük veri analizi ve istatistiksel makine öğrenimini (SML) kullanan bir model sunmuşlardır [10]. Koloveas, Chantzios, Tryfonopoulos ve Skiadopoulo IoT cihazları ile ilgili açık web, dark web ve sosyal medyandan siber tehdit istihbarat verisi elde edilmesini amaçlayan bir çalışma yapmıştır. Bu çalışmada Crawler oluşturularak elde edilen verilerin değerine göre puanlanmasını öngören bir model önerilmiştir [11]. Erkal, Sezgin ve Gündüz Twitter'da paylaşılan tweetlerin siber tehdit istihbaratı ile ilgili olup olmadığının tespit edilmesini amaçlayan bir çalışma gerçekleştirmiş bu çalışmada makine öğrenmesi yöntemlerini kullanan bir model önerilmiştir [12]. Bose, Behzadan, Aguirre ve Hsu Twitter'dan yeni siber tehdit olaylarının saptanması ve daha önce saptanan bir olayla benzerliklerinin tespit edilmesi amaçlayan bir çalışma gerçekleştirmiş, bu çalışmada elde edilen tweetlerin önemine göre puanlanmasını öngören bir model önermiştir [13]. Ghazi, Anwar, Mumtaz, Saleem ve Tahir yapılsal veya yapısal olmayan kaynaklardan siber tehdit istihbaratı elde edilmesini amaçlayan bir çalışma gerçekleştirmiş, çalışmada makine öğrenmesi ve doğal dil işleme yöntemleri ile birçok farklı kaynaktan elde edilen verilerin işlenerek WEB portal vasıtasıyla sunulmasını öneren bir model sunmuştur [14]. Aslan, Sağlam ve Li siber saldırı bilgilerinin paylaşıldığı twitter tweetlerinin tespit edilmesini amaçlayan bir çalışma gerçekleştirmiş, bu çalışmada makine öğrenmesi yöntemlerini

kullanan bir model önermiştir [15]. Deliu, Leichter ve Franke doğal dil işleme ve makine öğrenmesi algoritmaları kullanılarak hacker forumlarından siber tehdit istihbarat verisi elde edilmesine yönelik bir model sunmuştur [16]. Dionísio, Alves, Ferreira ve Bessani derin sinir ağları ile Twitter'da yer alan verilerden siber tehditlerin tespit edilmesine yönelik bir model önermiştir [17]. Behzadan, Aguirre, Bose ve Hsu derin sinir ağları ile twitterda yer alan verilerden siber tehditlerin tespit edilmesine yönelik bir model sunmuştur [18]. Mittal, Das, Mulwady, Joshi ve Finin çeşitli olası tehditler hakkında istihbarat elde etmek için twitter tweetlerini analiz etmiş ve toplanan istihbarat verilerini kullanarak kullanıma sunan bir model sunmuştur [19]. Wang ve Zhang CNN ve LSTM algoritmaları kullanarak derin öğrenme yöntemleri ile twitter tweetlerinden DDoS saldırılarının tahmin edilmesine yönelik bir model önerilmiştir [20]. Simran, Balakrishna, Vinayakumar ve Soman CNN ve RNN algoritmalarını kullanarak Twitter tweetlerinden STİ elde edilmesine yönelik bir model önermiştir [21]. Singh, Bansal ve Sofat twitter da yer alan zararlı kullanıcıların tespit edilmesine yönelik BayesNet, Naive Bayes, SMO, J48 ve Random Forest algoritmaları kullanan bir model önermiştir [22]. Deliu, Leichter ve Franke hacker forumlarında siber tehdit istihbaratı elde edilmesini amaçlayan bir çalışma gerçekleştir, bu çalışamda; Support Vector Machine ve Convolutional Neural Networks modellerinin performanslarını karşılaştırmıştır [23]. Rao, Kamhoua, Njilla ve Kwiat makine öğrenmesi uygulamaları kullanarak Twitter'dan spam, spammer ve şüheli kullanıcıların tespit edilmesini amaçlayan bir çalışma gerçekleştirmiştir [24]. Le, Wang, Nasim ve Babar Twitter'dan siber tehdit istihbaratı elde edilmesi amacıyla Common Vulnerabilities and Exposures (CVE) içeren tweetlerin makine öğrenmesi algoritmasıyla sınıflandıran bir model önermiştir [25]. Lee ve Shon kritik altyapılar ile ilgili açık kaynaklardan siber tehdit istihbarat veri elde edilmesi amaçlayan bir çalışma gerçekleştirmiştir. Bu çalışmada Shodan, theHarvester gibi açık kaynak platformlarından crawler kullanarak kritik altyapılara yönelik CTI elde edilmesini amaçlayan bir model önermiştir [26].

Açık kaynaklardan siber tehdit istihbaratı elde edilmesini amaçlayan çalışmalara yönelik özet çizelge aşağıda sunulmuştur.

Çizelge 2.1. Literatürde yer alan benzer çalışmalar

| S.No. | YAYIN ADI | AMAÇ | YÖNTEM | BAŞARI ORANI | ATIF |
|---|---|---|---|---|---|
| **1** | An effective security alert mechanism for real-time phishing tweet detection on Twitter | Twitter üzerinden oltamla amaçlı atılan tweetlerin tespit edilmesi amaçlanmıştır. | Makine öğrenmesi yöntemleri ile otalama amaçlı atılan tweetlerin tespit edilmesine yönelik bir metot önerilmiştir. | %95,49 | [7] |
| **2** | Cyber Threat Discovery from Dark Web | Darkweb de yer alan ve hackerlar tarafından paylaşılan verilerden siber tehdit istihbaratı elde edilmesi amaçlanmıştır. | Makine öğrenmesi yöntemleri kullanarak darkweb de yer alan bilgilerin siber tehdit istihbaratı olarak elde edilmesine yönelik bir model önerilmiştir. | %78,3 | [8] |
| **3** | Cybersecurity Text Data Classification and Optimization for CTI Systems | Makine öğrenmesi kullanan siber tehdit istihbarat modellerin de verinin işlenmesi ve anlamlı çıktı elde edilmesi konusunda daha başarılı olabilecek bir model amaçlanmıştır. | Multi layer keyword filtering ve doc2vec algoritmaları kullanılarak daha verimli olabilecek bir model önerilmiştir. | %92,88 | [9] |
| **4** | Cyber risk prediction through social media big data analytics and statistical machine learning | Siber riskleri tahmin etmek için sosyal medyada büyük veri analizi ve istatistiksel makine öğrenimini (SML) kullanan algoritmik bir model amaçlanmıştır. | Büyük veri analizi ve istatiksel makine öğrenme yöntemleri ile sosyal medya üzerinden siber risklerin öngörülmesine yönelik bir model sunulmuştur. | %96,73 | [10] |
| **5** | A crawler architecture for harvesting the clear, social, and dark web for IoT-related cyber-threat intelligence | IoT cihazları ile ilgili açık web, dark web ve sosyal medyandan siber tehdit istihbarat verisi elde edilmesi amaçlanmıştır. | Crawler oluşturularak verilerin elde edilmesi ve elde edilen verinin değerine göre puan verilmesi önerilmiştir. | %85,63 | [11] |
| **6** | A New Cyber Security Alert System for Twitter | Twitter tweetlerinin siber güvenlik ile ilgili olup olmadığının tespit edilmesi amaçlanmıştır. | Makine öğrenmesi yöntemleri ile atılan tweetlerin siber tehdit ile ilgili olup olmadığının tespit edilmesine yönelik bir model önerilmiştir. | %70,03 | [12] |
| **7** | A Novel Approach for Detection and Ranking of Trendy and Emerging Cyber Threat Events in Twitter Streams | Twitter'da yeni siber tehdit olaylarının saptanması ve daha önce saptanan bir olayla benzerliklerinin tespit edilmesi amaçlanmıştır. | Makine öğrenmesi ile atılan tweetlerin siber güvenlik ile ilgili olup olmadığı tespit edilen ve ilgili tweetlere önemine göre puan verilerek sırlama yapılan bir sistem önerilmiştir. | %93,75 | [13] |
| **8** | A Supervised Machine Learning Based Approach for Automatically Extracting High-Level Threat Intelligence from Unstructured Sources | Yapısal veya yapısal olmayan kaynaklardan siber tehdit istihbaratı elde edilmesi amaçlanmıştır | Makine Öğrenmesi ve Doğal dil işleme yöntemleri ile birçok farklı kayaktan elde edilen verilerin işlenerek web portal vasıtasıyla sunulması önerilmiştir. | %69 | [14] |
| **9** | Automatic Detection of Cyber Security Related Accounts on Online Social Networks: Twitter as an example | Siber saldırı bilgilerinin paylaşıldığı twitter tweetlerinin tespit edilmesi amaçlanmıştır | Makine öğrenmesi yöntemleri kullanarak siber güvenlikle ilgili atılan tweetlerin tespit edilmesine yönelik bir model önerilmiştir. | %95,75 | [15] |
| **10** | Collecting Cyber Threat Intelligence from Hacker Forums via a Two-Stage, Hybrid Process using Support Vector Machines and Latent Dirichlet Allocation | Hacker forumlarından siber tehdit istihbarat verisi elde edilmesi amaçlanmıştır | Doğal dil işleme ve Makine öğrenmesi algoritmaları kullanılarak hacker forumlarından siber tehdit istihbarat verisi elde edilmesine yönelik bir model sunulmuştur. | - | [16] |

Çizelge 2.1. (devam) Literatürde yer alan benzer çalışmalar

| S.No. | YAYIN ADI | AMAÇ | YÖNTEM | BAŞARI ORANI | ATIF |
|---|---|---|---|---|---|
| **11** | Cyberthreat Detection from Twitter using Deep Neural Networks | Twitter verilerinden faydalanarak siber tehdit istihbarat verisi elde edilmesi amaçlanmıştır. | Derin Sinir Ağları ile tweetterde yer alan verilerden siber tehditlerin tespit edilmesine yönelik bir model sunulmuştur. | %92 | [17] |
| **12** | Corpus and Deep Learning Classifier for Collection of Cyber Threat Indicators in Twitter Stream | Twitter verilerinden faydalanarak siber tehdit istihbarat verisi elde edilmesi amaçlanmıştır. | Siber güvenlikle ilişkili tweet'lerin tespiti için ikili bir sınıflandırma ve tweet'lerin birden çok siber tehdit tipine sınıflandırılması için çok sınıflı bir modelden oluşan basamaklı bir Evrişimsel Sinir Ağı (CNN) mimarisini sunulmuştur. | %94,72 | [18] |
| **13** | CyberTwitter: Using Twitter to generate alerts for Cybersecurity Threats and Vulnerabilities | OSINT sağlamak üzere Twitterdan siber güvenlik istihbaratı elde edilmesi amaçlanmıştır. | Çeşitli olası tehditler hakkında istihbarat elde etmek için gerçek zamanlı bilgi güncellemeleri için tweetler analiz edilmiştir. Semantik Web RDF'yi, toplanan istihbarat ve SWRL kurallarını kullanarak kullanıma sunulmuştur. | - | [19] |
| **14** | DDoS Event Forecasting using Twitter Data | Twitter tweetleri üzerinden DDoS saldırılarının tespit edilmesi amaçlanmaktadır. | CNN ve LSTM algoritmaları kullanarak derin öğrenme yöntemleri ile tweetlerden DDoS saldırılarının tahmin edilmesine yönelik bir model önerilmiştir. | - | [20] |
| **15** | Deep Learning Approach for Enhanced Cyber Threat Indicators in Twitter Stream | Twitter tweetlerinden siber tehdit istihbaratı elde edilmesi amaçlanmıştır. | CNN ve RNN algoritmalarını kullanarak STİ elde edilmesine yönelik bir model önerilmiştir. | %89,3 | [21] |
| **16** | Detecting Malicious Users in Twitter using Classifiers | Twitter da yer alan zararlı kullanıcıların tespit edilmesini amaçlanmıştır. | BayesNet, Naive Bayes, SMO, J48 ve Random Forest algoritmaları kullanarak Twitter'da yer alan zararlı kullanıcıların tespit edilmesine yönelik bir model önerilmiştir. | %99,8 | [22] |
| **17** | Extracting Cyber Threat Intelligence From Hacker Forums: Support Vector Machines versus Convolutional Neural Networks | Hacker forumlarından siber tehdit istihbarat verisi elde edilmesi amaçlanmıştır. | Support Vector Machine ve Convolutional Neural Networks modellerinin performanslarını karşılaştırarak hacker forumlarından siber tehdit istihbarat elde edilmesini amaçlayan bir model önerilmiştir. | %98,10 | [23] |
| **18** | Methods to Detect Cyberthreats on Twitter | Twitter'dan spam, spammer ve şüheli kullanıcıların tespit edilmesini amaçlamıştır. | Makine öğrenmesi uygulamaları kullanarak spam, spammer ve şüpheli kullanıcıların tespit edilmesini amaçlayan bir model önerilmiştir. | - | [24] |
| **19** | Gathering Cyber Threat Intelligence from TwitterUsing Novelty Classification | Twitter'dan siber tehdir istihbaratı elde edilmesi amaçlanmıştır. | Common Vulnerabilities and Exposures (CVE) içeren tweetlerin makine öğrenmesi algoritmasıyla sınıflandırılmasını amaçlayan bir model önerilmiştir. | %64,3 | [26] |
| **20** | Open Source Intelligence Base Cyber Threat Inspection Framework for Critical Infrastructures | Kritik altyapılar ile ilgili açık kaynaklardan siber tehdit istihbarat veri elde edilmesi amaçlanmıştır. | Shodan, theHarvester gibi açık kaynak platformlarından crawler kullanarak kritik altyapılara yönelik CTI elde edilmesini amaçlayan bir model önerilmiştir. | - | [26] |

# 3. METOT, YÖNTEM VE TEMEL BİLGİLER

## 3.1. Siber İstihbarat

Tarihsel olarak, savaşlar söz konusu olduğunda, güvenilir ve etkili istihbarat toplama taktikleri, muharebe senaryolarının sonucunu belirleyen başlıca faktörlerdir. Enigma'nın karmaşık kodunun deşifresinden Soğuk Savaş döneminde fark yaratan olaylardan, gizli operasyonlara kadar, istihbarat toplama teknikleri, kriptografik beceriler ve bunları deşifre etmek için kullanılan yöntemler zaman içinde önemli ölçüde değişmiştir. Mevcut düzene ve teknolojik ortamın değişen yapısına uygun olarak, dijital güvenlik tehditlerini izleme, analiz etme ve bunlara karşı koyma araçlarında da büyük ölçüde gelişme olmuştur.

Karmaşık olarak şifrelenmiş veri dosyaları, kodlanmış radyo mesajlarının yerini almış ve günümüzde, karmaşık güvenlik önlemlerini kırmak her zamankinden daha zor hale gelmiştir. Ancak buna rağmen, bu daha gelişmiş teknolojilerde insan hatasına daha az yer olsa da hatalar hâlâ mevcuttur. Potansiyel bir tehdidin ölçeğini ve büyüklüğünü anlamamak, yapılan ilk hatadır; siber suçları ve eylem kalıplarını bir an önce anlayamamak veya yorumlamamak, ardından tüm ortamlarda bulunan verilerin güvenliğini sağlayamamaktır.

Mevcut dijital ortam, ayrıntılar için titiz bir göze sahip olanlara iyi bir istihbarat ortamı sağlarken her şeyi kontrolsüz bırakma eğilimine sahip olanlar için tehlike oluşturmaktadır. Siber ortamda bulunan kurum/kuruluşlar potansiyel olarak siber saldırıların hedefi durumundadır. Siber saldırılar sonucunda elde edilen en değerli varlık ise siber istihbarat verileridir.

Amerika Birleşik Devletleri Savunma Bakanlığı'na göre, Siber istihbarat ve siber istihbarat faaliyetinde bulunan aktörler şu şekilde tanımlanmıştır [27];

- Yabancı ülkeler, düşman veya potansiyel olarak düşman kuvvetlerin operasyon alanlarıyla ilgili mevcut bilgilerin toplanması, işlenmesi, entegrasyonu, değerlendirilmesi, analizi ve yorumlanmasından kaynaklanan ürün.
- Ürünle sonuçlanan faaliyetler.
- Bu tür faaliyetlerde bulunan kuruluşlar.

### 3.2. Siber Tehdit İstihbaratı

Siber tehdit istihbaratı, siber tehdit bilgilerinin toplanıp, kaynağı ve güvenilirliği bağlamında değerlendirilmesi, uzman kişilerce detaylı olarak incelenmesinin ardından elde edilen bilgilerdir. Tüm istihbaratlar gibi, siber tehdit istihbaratı da bilgiye sahip olan için belirsizliği azaltmakta ve tehditleri ve fırsatları belirlemesine yardımcı olan siber tehdit bilgilerine katma değer sağlamaktadır.

Siber Tehdit İstihbaratını gelişimi, uçtan uca bir süreçte geliştirilmekten ziyade, istihbarat döngüsü olarak adlandırılan döngüsel bir süreçte gerçekleşmektedir. Bu döngüde gereksinimler belirlenir; veri toplama planlanır, uygulanır ve değerlendirilir; sonuçlar istihbarat üretmek için analiz edilir ve ortaya çıkan istihbarat, geri bildirimleri bağlamında yayılır ve yeniden değerlendirilir. Döngünün analiz kısmı, istihbaratı bilgi toplama ve yaymadan ayıran şeydir. İstihbarat analizi, önyargıların, düşüncelerin ve belirsizliklerin ortadan kaldırılmasını sağlamayan yapılandırılmış analitik teknikleri kullanan titiz bir düşünme biçimine dayanmaktadır. İstihbarat analistleri, zor sorular hakkında sadece sonuçlara varmak yerine, sonuçlara nasıl ulaştıklarını da düşünürler. Bu ekstra adım, mümkün olduğu ölçüde, analistlerin zihniyetlerinin ve önyargılarının hesaba katılmasını ve en aza indirilmesini veya gerektiğinde dâhil edilmesini sağlar.

Siber tehdit istihbaratında, analiz genellikle taktikleri, teknikleri ve prosedürleri (TTP'ler), motivasyonları ve amaçlanan hedeflere erişimi göz önünde bulundurarak aktörler, niyet ve yetenek üçlüsüne dayanır. Bu üçlüyü inceleyerek, bilgiye dayalı, ileriye dönük stratejik, operasyonel ve taktiksel değerlendirmeler yapmak genellikle mümkündür.

Stratejik siber tehdit istihbaratı

Entegre görünümler oluşturmak için farklı bilgi parçalarını değerlendirir. Karar vericileri ve politika yapıcıları geniş veya uzun vadeli konularda bilgilendirir ve/veya tehditlere karşı zamanında uyarı sağlar. Stratejik siber tehdit istihbaratı, aktörler, araçlar ve TTP'ler dahil olmak üzere kötü niyetli siber tehditlerin niyetinin ve yeteneklerinin genel bir resmini oluşturur.

Operasyonel siber tehdit istihbaratı:

Olaylar, soruşturmalar ve/veya faaliyetlerle ilgili belirli, potansiyel olayları değerlendirir ve müdahale operasyonlarına rehberlik edebilecek ve destekleyebilecek öngörüler sağlar. Operasyonel veya teknik siber tehdit istihbaratı, belirli olaylara müdahaleyi yönlendirmek ve desteklemek için son derece uzmanlaşmış, teknik odaklı istihbarat sağlar; bu tür istihbarat genellikle kampanyalar, kötü amaçlı yazılımlar ve/veya araçlarla ilgilidir ve adli raporlar şeklinde gelebilir.

Taktik siber tehdit istihbaratı

Gerçek zamanlı olayları, araştırmaları ve/veya faaliyetleri değerlendirir ve günlük operasyonel destek sağlar. Taktik siber tehdit istihbaratı, imzaların ve tehdit göstergelerinin (IOC) geliştirilmesi gibi günlük operasyonlar ve olaylar için destek sağlar. Genellikle geleneksel istihbarat analiz tekniklerinin sınırlı uygulamasını içerir.

### 3.3. Açık Kaynak İstihbaratı

Açık kaynak istihbaratı (OSINT), istihbarat bağlamında kullanılmak üzere kamuya açık kaynaklarda erişilebilir veriler hakkında toplama, analiz etme ve karar verme için çok faktörlü (nitel, nicel) bir metodolojidir. İstihbarat camiasında, "açık" terimi ile aleni, kamuya açık kaynaklara (gizli veya gizli kaynakların aksine) atıfta bulunmaktadır. Hızlı iletişimin ve bilgi aktarımının ortaya çıkmasıyla birlikte, artık kamuya açık, sınıflandırılmamış kaynaklardan çok sayıda eyleme geçirilebilir ve tahmine dayalı istihbarat elde edilebilmektedir [28].

OSINT, halka açık kaynaklardan toplanan bilgilerin toplanması ve analizidir [29]. OSINT öncelikle ulusal güvenlik, kanun yaptırımı ve iş zekâsı işlevlerinde kullanılır ve önceki istihbarat disiplinlerinde sınıflandırılmış, sınıflandırılmamış veya tescilli istihbarat gereksinimlerini yanıtlarken hassas olmayan istihbarat kullanan analistler için değerlidir.
OSINT kaynakları altı farklı bilgi akışı kategorisine ayrılabilir [30]:

- Medya, basılı gazeteler, dergiler, radyo ve televizyon.
- İnternet, çevrimiçi yayınlar, bloglar, tartışma grupları, YouTube, Facebook, Twitter, Instagram gibi sosyal medya platformları. Bu kaynak aynı zamanda

güncelliği ve erişim kolaylığı nedeniyle diğer çeşitli kaynaklardan daha zengin veriler içerebilmektedir.

- Kamu verileri, kamu raporları, bütçeler, duruşmalar, telefon rehberleri, basın toplantıları, web siteleri ve konuşmalar. Bu kaynak resmi bir kaynaktan gelmesine rağmen, kamuya açıktır ve açık ve ücretsiz olarak kullanılabilir.
- Profesyonel ve akademik yayınlar, dergilerden, konferanslardan, sempozyumlardan, akademik makalelerden, tezlerden elde edilen bilgiler.
- Ticari veriler, ticari görüntüler, finansal ve endüstriyel değerlendirmeler ve veri tabanları.
- Teknik raporlar, ön baskılar, patentler, çalışma kâğıtları, ticari belgeler, yayınlanmamış çalışmalar ve haber bültenleri.

### 3.4. Doğal Dil İşleme

Doğal dil işleme (NLP), bilgisayar biliminin daha özel olarak yapay zekânın bilgisayarlara metinleri ve konuşulan kelimeleri insanların yapabileceği şekilde anlama yeteneği kazandırılmasıdır.

NLP, hesaplamalı dilbilimini (insan dilinin kural tabanlı modellemesi) istatistiksel, makine öğrenimi ve derin öğrenme modelleriyle birleştirir. Bu teknolojiler bilgisayarların insan dilini metin veya ses verileri biçiminde işlemesine ve konuşmacının veya yazarın niyeti ve duygusuyla birlikte tam anlamını "anlamasına" olanak tanımaktadır [31].

NLP, metni bir dilden diğerine çeviren, sözlü komutlara yanıt veren ve büyük hacimli metni hızlı bir şekilde, hatta gerçek zamanlı olarak özetleyen bilgisayar programlarını çalıştırabilmektedir. NLP ile sesle çalışan GPS sistemleri, dijital yardımcılar, konuşmadan metne çeviri yazılımı, müşteri hizmetleri sohbet robotları tasarlanabilmektedir [32]. Ayrıca NLP, iş operasyonlarını kolaylaştırmaya, çalışan üretkenliğini artırmaya ve kritik iş süreçlerini basitleştirmeye yardımcı olan kurumsal çözümlerde de büyüyen bir rol oynamaktadır.

İnsan dili, metin veya ses verilerinin amaçlanan anlamını doğru bir şekilde belirleyen bir yazılım üretmeyi oldukça zorlaştıran belirsizliklerle doludur. Eş anlamlı sözcükler, eş sesli sözcükler, alaycılık, deyimler, metaforlar, dilbilgisi ve kullanım istisnaları, cümle yapısındaki farklılıklar insanların için öğrenmesi yıllar süren ve doğal dil odaklı uygulamaları öğretmesi gereken insan dilinin düzensizliklerinden sadece birkaçıdır [33].

NLP modelleri, insan metinlerini ve ses verilerini, bilgisayarın anlayacağı şekilde anlamlandırmasına yardımcı olacak şekilde çalışır. Bu modellerden bazıları aşağıda tanımlanmıştır:

Konuşma tanıma (Speech recognition)

Konuşmadan metne olarak da adlandırılan konuşma tanıma, ses verilerini güvenilir bir şekilde metin verilerine dönüştürme işlemidir. Sesli komutları takip eden veya sözlü soruları yanıtlayan tüm uygulamalar için konuşma tanıma gereklidir. Konuşma tanımayı özellikle zorlaştıran şey insanların, hızlı bir şekilde konuşması, kelimeleri farklı vurgulaması ve tonlamaları, farklı aksanlarda ve genellikle yanlış dilbilgisi kullanmalarıdır.

Konuşma bölümü etiketleme (Part of speech tagging)

Gramer etiketlemesi olarak da adlandırılır, belirli bir kelimenin veya metin parçasının kullanım ve bağlamına dayalı olarak konuşmanın bir bölümünü belirleme işlemidir.

Kelime anlamı belirsizleştirme (Word sense disambiguation)

Kelime anlamı belirsizleştirme, belirli bir bağlamda en anlamlı kelimeyi belirleyen bir anlamsal analiz süreci yoluyla birden çok anlamı olan bir kelimenin anlamının seçilmesidir.

Adlandırılmış varlık tanıma (Named entity recognition-NEM)

Sözcükleri veya tümcecikleri yararlı varlıklar olarak tanımlar. NEM, 'Kentucky'yi bir yer adı olarak veya 'Fred'i bir erkek adı olarak tanımlar.

Duygu analizi (Sentiment analysis)

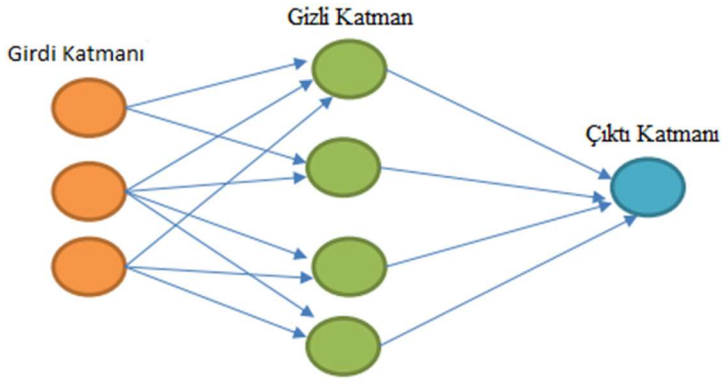
Metinden öznel nitelikleri (tutumlar, duygular, alaycılık, kafa karışıklığı, şüphe) çıkarmayı amaçlayan modellerdir.

Doğal dil üretimi (Natural language generation)

Konuşma tanımanın veya konuşmayı metne dönüştürmenin tersi olarak tanımlanır; yapılandırılmış bilgiyi insan diline yerleştirme amaçlayan modellerdir.

### 3.5. Yapay Sinir Ağları

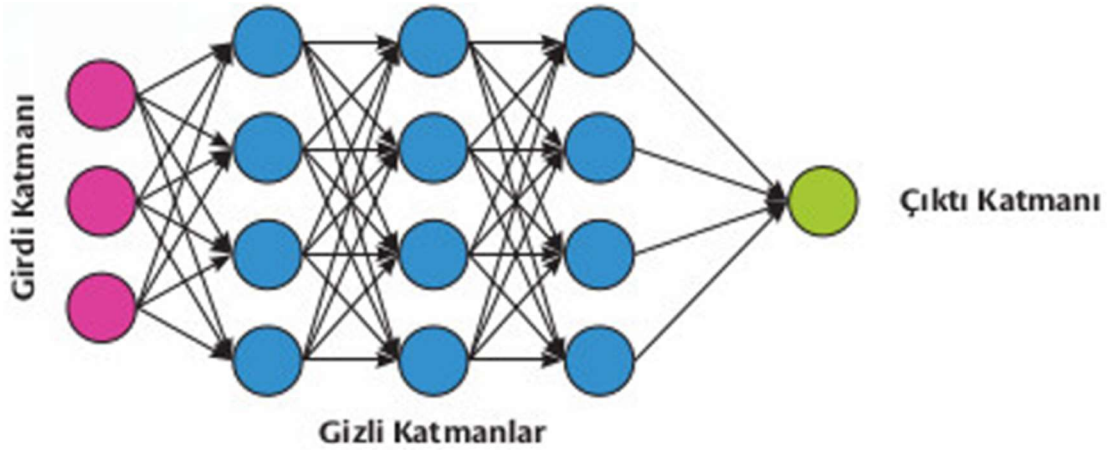
Yapay Sinir Ağları, biyolojik bir sinir ağının işlevselliğinden ilham alınarak tasarlanan bir makine öğrenimi modelleri sınıfıdır. YSA, algılayıcı adı verilen bağlı işlem birimlerinin toplamından oluşan sistemdir. Bu birimler katmanlar halinde birbirine bağlanmış ve katmanlar bir sinir ağı oluşturmak için bir araya getirilmiştir. Girdiyi alan katmana girdi katmanı, çıktıyı üreten katmana da çıktı katmanı denilmektedir. Ara katmanlara ise gizli katmanlar denilmektedir.




Şekil 3.1. Yapay sinir ağının bir örneği [34]

### 3.6. Derin Sinir Ağları

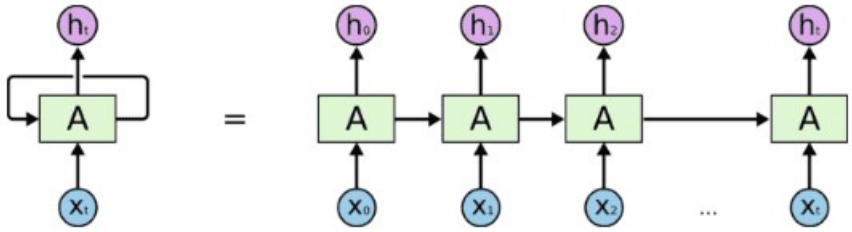
Derin Sinir Ağları, YSA'nın bir alt sınıfıdır. Daha karmaşık işlevlerin modellenmesine yardımcı olan birden fazla gizli katmana sahiptir. DSA, veriler doğrusal olarak ayrılabilir olmadığında daha başarılıdır. DSA, bilgisayarlı görü, konuşma tanıma, ses tanıma, biyoinformatik, doğal dil işleme vb. gibi yapay zekânın birçok alanında makine öğrenimi tekniklerine öncülük etmektedir.




Şekil 3.2. Derin sinir ağının bir örneği [35]

### 3.7. Özyinelemeli Sinir Ağları

Özyinelemeli Sinir Ağlarında (RNN), en az bir hücrenin çıktısı kendisine veya diğer hücrelere girdi olarak verilir ve genellikle geri besleme bir gecikme elemanı üzerinden üretilir. Geri besleme hücreler arasında, katmanlar arasında olabileceği gibi bir katmandaki hücreler arasında da olabilir. Bu nedenle geribildirimin üretildiği yönteme göre çeşitli yapı ve davranışlarda tekrarlayan mimariler elde edilebilir. Şekil 3.3'de verilen temel bir RNN, nöronların hem giriş nöronlarından hem de gizli katmanların çıktılarından değer alabildiğini göstermektedir.




Şekil 3.3. Özyinelemeli sinir ağı (RNN) [34]

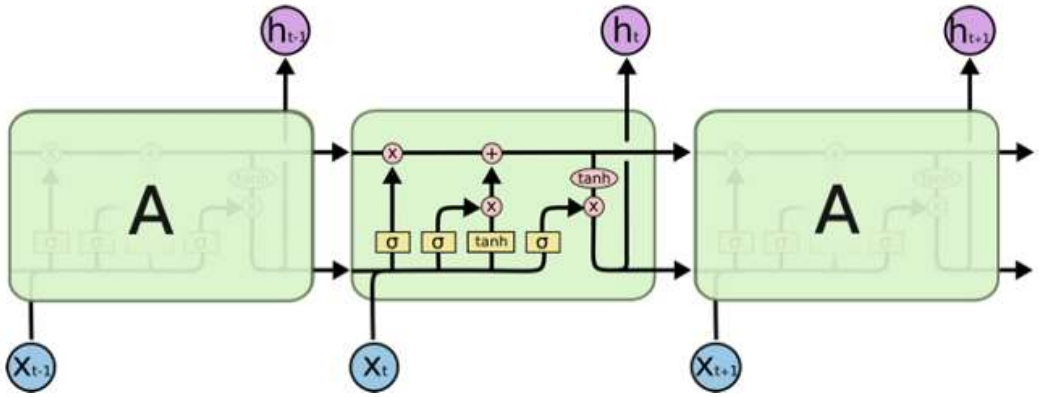
#### 3.7.1. Uzun kısa vadeli bellek (Long short term memory-LSTM)

Özyineleme (RNN) mimarilerinde önceki bilgi kullanımına dayalı bir yaklaşım bulunmaktadır. Örneğin ağaçların toprakta büyüdüğünü belirten cümledeki toprak kelimesini tahmin etmek kolaydır. Ancak bağlamlar arasındaki boşluk arttığında, RNN'nin geçmişten gelen bilgileri kullanma zorluğu artmaktadır. Örneğin, “I have been lived in the Turkey for 10 years ……., I love Turkish people.” Bu metindeki “Turkish” sözcüğü

tahmin edilirken cümlenin bağlamına göre bir dil adı olacağı tahmin edilebilir, ancak cümlenin metnin başında yer alması gerektiğini tahmin etmek için cümleyi metnin başında tutmak gereklidir. Teoride mümkün olan uzun vadeli bağımlılıkların pratikte büyük sorunlara yol açtığı gözlemlenmiştir. Bu sorunu çözmek için uzun süreli bağımlılıkları öğrenebilen özel bir RNN türü olarak Uzun Kısa Süreli Bellek Ağları (LSTM) tasarlanmıştır. Hochreiter ve Schmidhuber (1997) tarafından tanıtılan ağ, çok çeşitli problemler üzerinde oldukça başarılı çalışmaktadır [36].

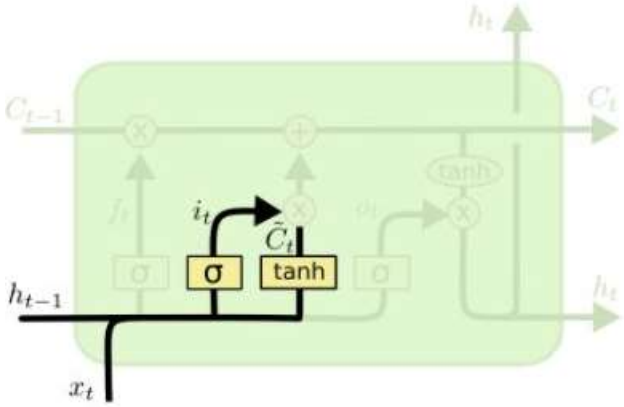
Özyinelemeli yapay sinir ağlarının yöntemlerinden biri olan LSTM ağları; cümle tamamlama, doğal dil işleme, sensör verileri, yani zamana bağlı, sıralı sensör verileri, anlama, sınıflandırma ve tahmin amaçları için kullanılmaktadır.

LSTM mimarisinde kapılar ve hücre durumları olmak üzere iki önemli parametre mevcuttur. Kapı, talep üzerine bilgi iletilen yoldur. LSTM, hücre durumuna bilgi ekleyebilen veya çıkarabilen kapılar adı verilen yapılarla düzenlenmiştir (Şekil 3.4).




Şekil 3.4. LSTM mimarisi [37]

LSTM algoritmasındaki ilk adım, hücre durumundan hangi bilgilerin atılacağına karar verilmesidir. Şekil 3.5'de görüldüğü gibi bu karar “forget gate layer” adı verilen sigmoid katmanı tarafından verilmektedir.

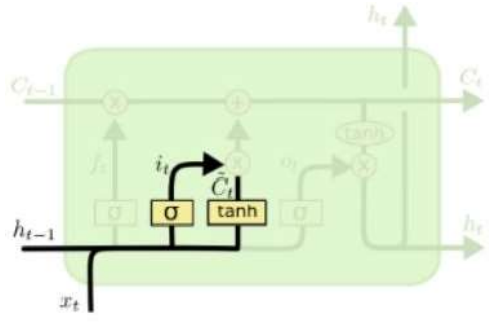



Şekil 3.5. Unutma kapısı (Forget gate) [38]

Unutma kapısı $h_{t-1}$ ve $x_t$'ye bakar ve $c_{t-1}$ hücre durumuna çıktı oluşturur. Çıktı 0 ile 1 arasında bir sayıdır, burada 1 "bu bilgiyi sakla", 0 "bilgiyi saklama" anlamına gelir, "W" ağırlıktır ve b denklemdeki önyargı terimidir (Eş. 3.1).

$$f_t = \sigma(W_f.[h_t-1, x_t] + b_f) \tag{3.1}$$

Bir sonraki adımda, hücrede hangi yeni bilgilerin depolanacağına karar verilmektedir. İlk olarak, sigmoid katmanı içeren giriş kapısı katmanı, güncellenecek yeni değerlere belirlenir (Eş. 3.2). Daha sonra, bir tanh katmanı, aşağıdaki denklemde gösterildiği gibi hücre durumuna eklenecek olan bir başvuru vektörü $\hat{C}_t$ oluşturulur (Eş. 3.3). Şekil 3.6'da gösterildiği gibi, son aşamada sigmoid ve tanh katmanlarını birleştirilerek hücre durumu güncellenmektedir.

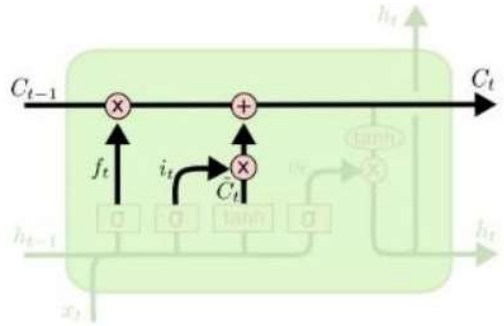



Şekil 3.6. Hücrede saklanacak yeni bilgilerin seçimi için karar bloğu [38]

$$i_t = \sigma(W_i.[h_{i-1}, x_t] + b_i) \tag{3.2}$$

$$\hat{C}_t = tanh(W_c\ .[h_{t-1}, x_t\ ] + b_c) \tag{3.3}$$

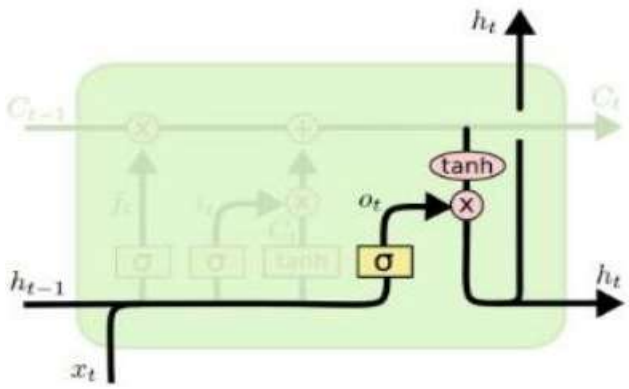
Daha sonra eski hücre durumu $C_{t-1}$, yeni hücre durumu $C_t$'ye güncellenir. Şekil 3.7'de gösterildiği gibi, eski $C_{t-1}$durumu $f_t$ ile çarpılır ve daha sonra denkleme $i_t * \hat{C}_t$ eklenir (Eş. 3.4). Bu, her durum değerinin ne kadar güncellenmesi gerektiğine göre hesaplanan yeni başvuru değeridir. Önceki dil örneğinde, eski durumla ilgili bilgilerin unutulduğu ve yeni bilgilerin eklendiği yer burasıdır.




Şekil 3.7. Eski bilgilerin unutulması ve yeni bilgilerin eklenmesi [38]

$$C_t = f_t * C_{t-1} + i_t * \hat{C}_t \tag{3.4}$$

Son olarak, filtrelenmiş hücre durumuna dayalı karar aşamasına (Şekil 3.8) gelinir. İlk olarak, hücre durumu bileşenlerinin çıkarılacağına karar veren bir sigmoid katman oluşturulur (Eş. 3.5). Daha sonra, hücre durumu tanh yoluyla beslenir ve sigmoid geçidin çıktısı ile çarpılır ve çıktı üretilir (Eş. 3.6).



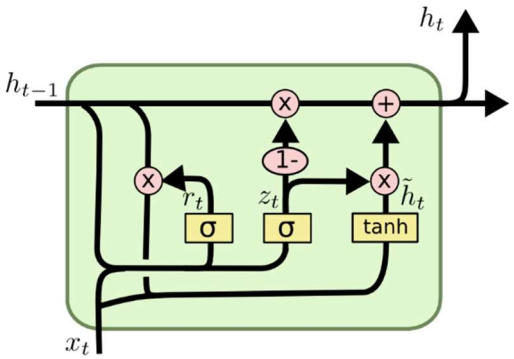

Şekil 3.8. Çıktı bilgisinin oluşturulması [38]

$$o_t = \sigma(W_o\,[h_{t-1}, x_t\,] + b_o) \tag{3.5}$$

$$h_t = o_t * tanh(C_t) \tag{3.6}$$

### 3.7.2. Geçitli tekrarlayan birim (Gated recurrent unit-GRU)

Geçitli Tekrarlayan Birim (GRU) 2014 yılında tanıtılmıştır [38]. GRU LSTM'e benzer bir yapıdadır ancak daha az parametreye sahiptir. Ayrıca, LSTM'de yer alan birim içindeki bilgi akışını kontrol eden ayrı bellek hücreleri yoktur. LSTM'den farklı olarak, GRU'nun çıkış kapısı yoktur. GRU çalışma şeması ve formülasyonu aşağıdaki denklemlerle gösterildiği gibidir.




Şekil 3.9. GRU mimarisi ve formülasyonu [38]

GRU formülasyonu, $r_t$, $z_t$, $x_t$, $h_t$'nin sırasıyla sıfırlama kapısı, güncelleme kapısı, giriş vektörü ve çıkış vektörü olduğu yukarıda gösterilen denklemlerle verilebilir. LSTM'e benzer şekilde, W ağırlık matrislerini, b önyargıları, sigm sigmoid aktivasyonu ve tanh hiperbolik tanjant aktivasyon fonksiyonunu ifade etmektedir.

Hem LSTM hem de GRU, uzun vadeli bağımlılıkları ele alma konusunda eşit derecede yeteneklidir ve makine çevirisi görevleriyle denenmiş ve karşılaştırılmıştır ve karşılaştırmalı olarak verimli oldukları kanıtlanmıştır [38].

### 3.8. Hiper Parametre Optimizasyonu

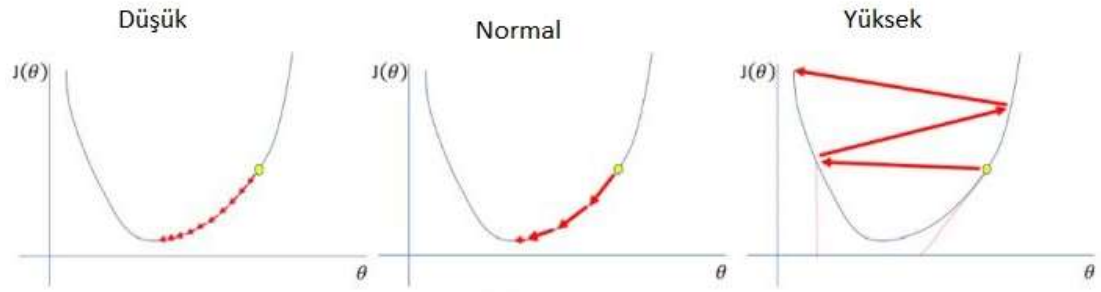
Yapay sinir ağının başarısını belirleyen parametreler hiper parametreler olarak adlandırılır ve bu parametreler eğitimin başarı oranı belirlemektedir. Bunlar, küme boyutu (batch size), öğrenme oranı (learning rate), eğitim süresi (epoch), optimizasyon algoritması, aktivasyon fonksiyonu, katman sayısı ve nöron sayısı gibi değişkenlerdir. Başarı oranını artırmak için hem hiper parametrelerde hem de aşağıdaki bölümlerde detaylı bilgiler verilen veri serilerinde değişiklikler yapılmıştır [39].

### 3.8.1. Küme boyutu (Batch size)

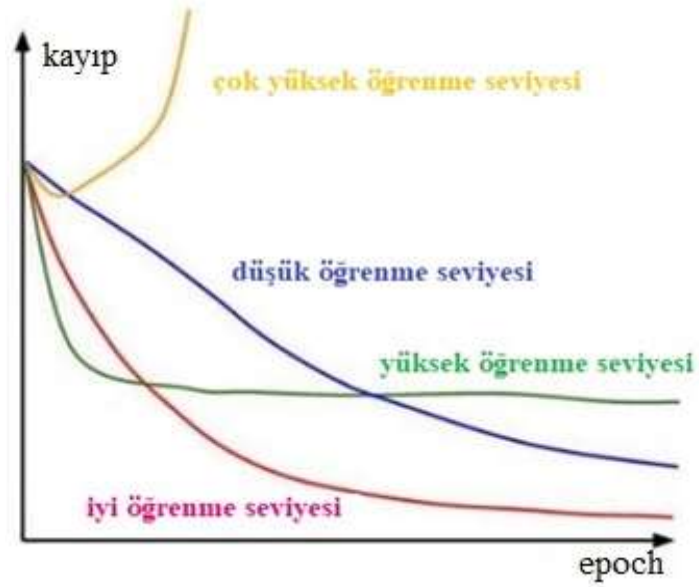
Derin öğrenme modellerinde veri setinin büyük olması öğrenme sürecinin başarısını arttırmaktadır. Aynı zamanda veri setinin büyüklüğü eğitim için harcanan zamanı ve öğrenme sonunda elde edilen modelin boyutunu da artırmaktadır. Yapay sinir ağının öğrenme aşamasında veri setindeki tüm verilerin aynı anda öğrenilmesi öğrenme süresi ve hafıza açısından önemlidir. Her öğrenme yinelemesinde, ağırlıkları güncellemek için geri yayılım işlemiyle ağda gradyan inişi hesaplanır. Bu hesaplamadaki veri sayısı ne kadar yüksek olursa, işlemin tamamlanması o kadar uzun sürer. Bu problemi çözmek için; veri seti küçük gruplara bölünür ve eğitim bu küçük gruplarla yapılır. Küme boyutu (batch size) genellikle 64 ile 512 arasında 2'nin katı olan değerlerden belirlenmektedir [39].

### 3.8.2. Öğrenme oranı (Learning rate)

Derin öğrenmede ağırlıklar geri yayılım süreci ile güncellenmektedir. Bu işlemde, öğrenme oranı ile çarpılan türev bulunarak ve önceki ağırlık parametrelerinden çıkarılarak ağırlıkların güncellemesi hesaplanır. Daha yüksek öğrenme oranları bir dalgalanmaya neden olurken, küçük bir öğrenme oranı ile güncellenen ağırlıklar öğrenme süresini uzatır.




Şekil 3.10. Gradient descent üzerinde öğrenme hızının etkisi [40]




Şekil 3.11. Kaybın öğrenme oranına etkisi [41]

Öğrenme hızının etkisi Şekil 3.11'de gösterilmekte olup, burada kayıp hatayı, epoch ise periyodun tekrar sayısını ifade etmektedir. Eğitim başlatıldığında, öğrenme hızına genellikle 0,01'lik bir başlangıç değeri atanır ve daha sonra 0,001 ile 0,0001 veya daha düşük değerler atanır, doğruluk üzerindeki etkisi test edilerek en uygun değer belirlenebilir [42].

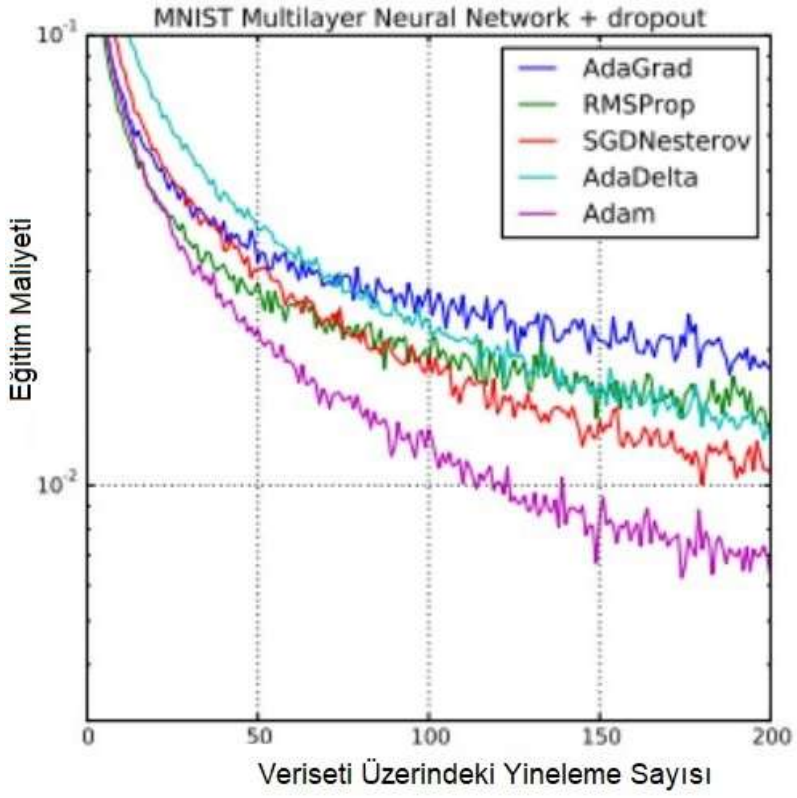
### 3.8.3. Eğitim süresi (Traning epoch)

Derin öğrenme çalışmalarında model eğitilirken, tüm veriler aynı anda eğitime dahil edilmemektedir. Yapay sinir ağı verileri parçalar halinde alır, ilk kısım eğitilir, modelin performansı test edilir ve geriye yayılım kullanılarak ağırlıklar güncellenir. Daha sonra model yeni veri seti ile yeniden eğitilir ve ağırlıklar tekrar güncellenir. Model için en uygun ağırlık değerlerini hesaplamak için bu işlem her eğitim adımında tekrarlanır. Bu eğitim adımlarının her birine bir çağ (epoch) denir [42].

Epoch sayısı arttıkça modelin performansı önemli ölçüde artmaktadır. Belli bir epoch değerinden model ezberlemeye başlayacağından aşırı öğrenme durumu ortaya çıkmaktadır. Bu sebeple optimum epoch sayısı belirlenerek eğitim sonlandırılmalıdır.

### 3.8.4. Optimizasyon algoritması seçimi

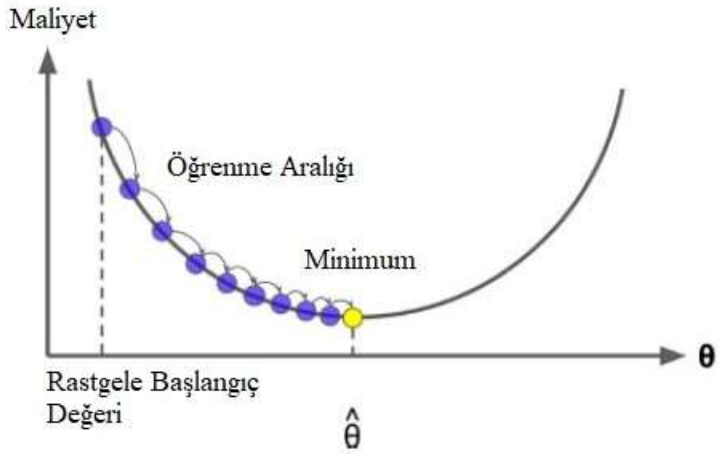
Derin öğrenme çalışmalarında meydana gelebilecek hataları minimum seviyeye indirmek için optimizasyon algoritmaları kullanılmaktadır. Bu algoritmalar gradyan iniş (Gradient Descent) algoritmaları olarak da tanımlanmaktadır. Optimizasyon adım adım işletilen bir işlemdir. Bu süreçte işletilen, adım sayısına öğrenme katsayısı denilmektedir. Öğrenme katsayısı belirlenirken en uygun değerin seçilmesi gerekmektedir. Öğrenme katsayısının küçük seçilmesi durumunda modelin çalışma süreci uzamaktadır, büyük seçilmesi durumunda ise minimum noktanın gözden kaçırılmasına sebep olmaktadır. Stokastik gradyan inişi (SGD), RMSProp, Adamax, Adagrad, Adam ve Adadelta algoritmaları derin öğrenme uygulamalarında yaygın olarak kullanılan optimizasyon algoritmalarıdır. Bu algoritmalar arasında performans ve hız açısından farklılıklar bulunmakta olup, örnek olarak MNIST veri seti için farklı algoritmaların performansı Şekil 3.12'de gösterilmiştir.




Şekil 3.12. MNIST veri seti için farklı optimizasyon algoritmalarının performansı [43]

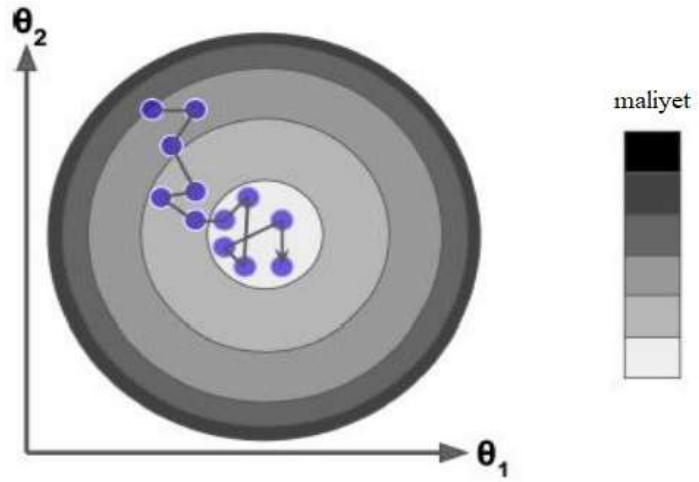
Gradient Descent, çok çeşitli problemler için en uygun çözümleri bulmak üzere tasarlanmış genel bir optimizasyon algoritmasıdır. Gradient Descent, maliyet fonksiyonunu en aza indirmek için parametreleri tekrar tekrar ayarlamaktır (Şekil 3.13). Öğrenme hızı

Gradient Descent için önemli bir hiper parametredir. Batch Gradient Descent, eğitim seti çok büyük olduğunda yinelemeyi yavaşlatır ve her adımda inişleri hesaplamak için tüm eğitim setini kullanır.




Şekil 3.13. Gradient descent [44]

Stokastik Gradyan Decent'de, eğitim setindeki her adımda rastgele bir örnek seçilir ve yalnızca bir örneğe dayalı olarak gradyanlar hesaplanır. Bu, algoritmayı daha hızlı hale getirir çünkü her yinelemede çok daha az veri kullanılmaktadır. Maliyet fonksiyonu, kademeli olarak minimum seviyeye indirilmek yerine, bir ortalama etrafında yukarı ve aşağı atlatılır (Şekil 3.14). Zamanla minimum seviyeye yaklaşır ve minimuma vardığında bir son noktaya yerleşmeden salınım yapılmaya devam eder.




Şekil 3.14. Stochastic gradient descent [44]

Adagrad, gradyan tabanlı bir algoritmadır. Öğrenme hızını parametrelere uyarlar, seyrek

parametreler için büyük güncellemeler yaparken sık parametreler için daha küçük güncellemeler yapar. Doğal dil ve bilgisayarla görme sorunları için kullanılır.

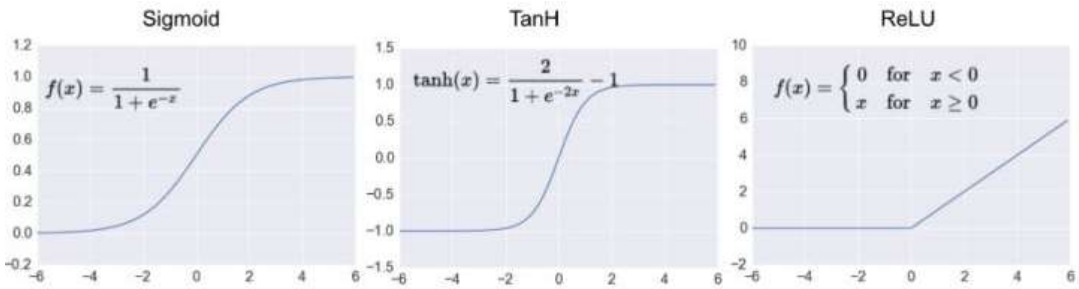
Adadelta, agresif ve sıkı bir şekilde azaltılmış öğrenme oranını düşürmeye çalışan Adagrad'ın bir uzantısıdır. Adadelta, önceki toplam kare inişlerini toplamak yerine, birikmiş önceki iniş penceresini sabit bir boyutla sınırlar.

RMSprop, Geoffrey Hinton tarafından tasarlanmıştır [45]. RMS desteği, öğrenme oranını ayarlama ihtiyacını ortadan kaldırır ve bunu otomatik olarak yapar. Ayrıca, RMSProp her parametre için farklı bir öğrenme oranı seçer. RMSProp, parametre güncellemelerini yeniden ölçeklendirilmiş gradyan üzerinde bir momentum kullanarak oluşturur.

Adam optimizer, Adagrad ve RMSprop'un birleşimidir. Uyarlanabilir bir öğrenme oranı yöntemidir. Adam, parametre öğrenme oranlarını RMSProp'taki ortalama başlangıç taban çizgisine ayarlamak yerine, gradyanların birinci ve ikinci anlarının ortalamasını kullanarak güncellemeler yapar. Adam, hızlı ve iyi sonuçlar veren ve derin öğrenme çalışmalarında sıklıkla kullanılan bir optimizasyon algoritmasıdır.

### 3.8.5. Aktivasyon fonksiyonu

Çok katmanlı yapay sinir ağlarında doğrusal olmayan dönüşüm işlemleri için aktivasyon fonksiyonları kullanılmaktadır. Gizli katmanların çıktısı, gizli katmanlardaki gradyanı hesaplamak için bazı aktivasyon fonksiyonları tarafından normalleştirilir (Şekil 3.15).

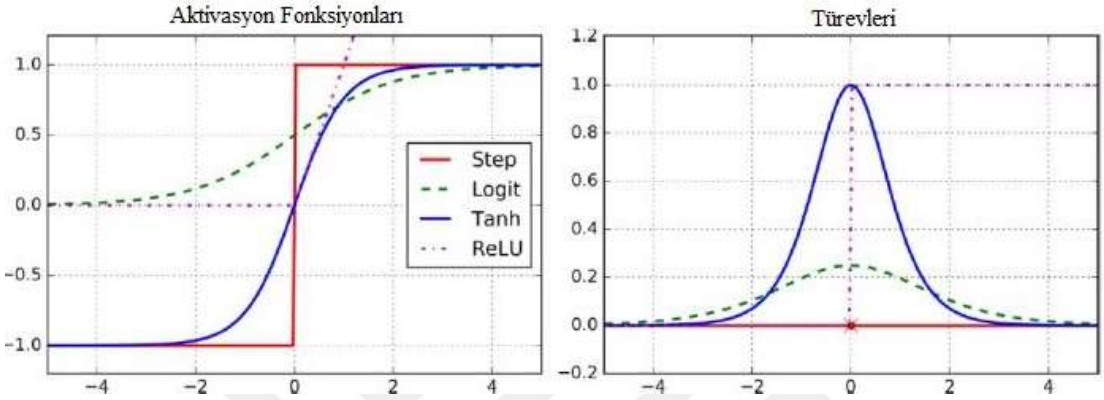



Şekil 3.15. Sıklıkla kullanılan aktivasyon fonksiyonları [46]

Sigmoid fonksiyonu 0 ile 1 arasında, tanh fonksiyonu -1 ile 1 arasında, Rektifiye Lineer Birim (ReLU) fonksiyonu 0 ile sonsuz arasında değer almaktadır.

ReLU, derin öğrenme çalışmalarında özellikle de Evrişimli Sinir Ağları (CNN) çalışmalarında en çok kullanılan aktivasyon fonksiyonlarından biridir.

Aktivasyon fonksiyonları, ağırlıkları güncellemek için gradyan iniş ile birlikte kullanılır. Buradaki amaç Şekil 3.16'da gösterilen kolay türevleri elde etmektir.

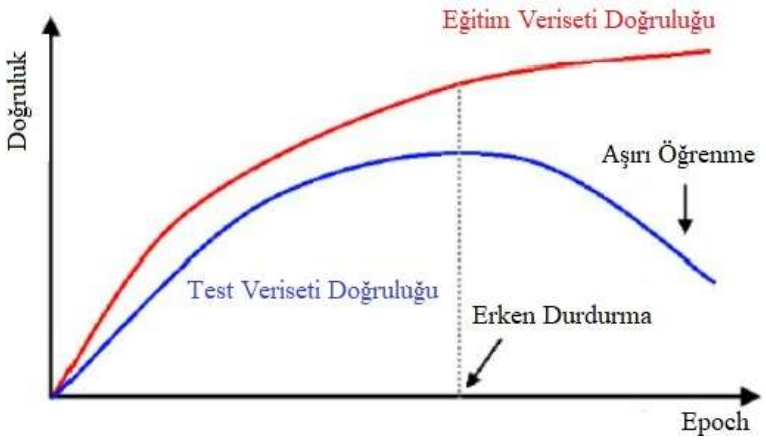



Şekil 3.16. Aktivasyon fonksiyonları ve türevleri [44]

### 3.8.6. Aşırı öğrenme, az öğrenme

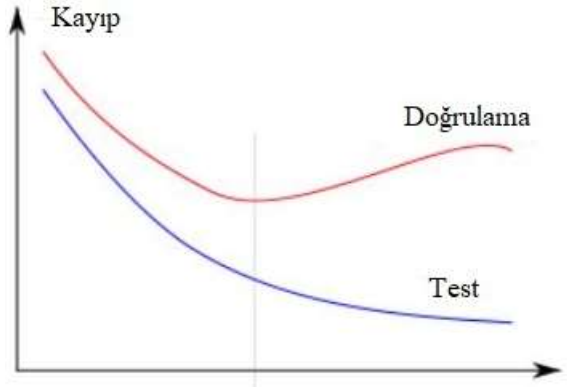
Derin öğrenmede sinir ağları optimum sayıda epoch ile eğitilmelidir. Az sayıda epoch ile eğitilirse, model iyi öğrenemez, eğitim ve test süreçlerinin başarısını azaltır. Model, çok fazla epoch ile eğitilirse, fazla uydurmaya yol açan eğitim setlerini ezberler. Bu gerçekleştiğinde, eğitim seti daha yüksek doğruluk sağlar (neredeyse 1,0 veya 1,0'a çok yakındır), test seti daha düşük bir doğruluğa yerleşirken doğrulama kaybı artmaya başlar. Ağın doğruluğu, doğru sınıflandırılmış vakaların tüm örneklere oranı ile belirlenir.

Eğitim ve test setlerinin doğruluk ve kayıp grafikleri, genellikle aşırı veya az öğrenme problemlerini anlamak için birlikte çizilir. Eğitim doğruluğu ile test doğruluğu arasındaki fark artıyorsa (Şekil 3.17) model fazla uydurmaya başlar ve eğitim sonlandırılır.




Şekil 3.17. Eğitim ve test doğruluk grafiği [47]

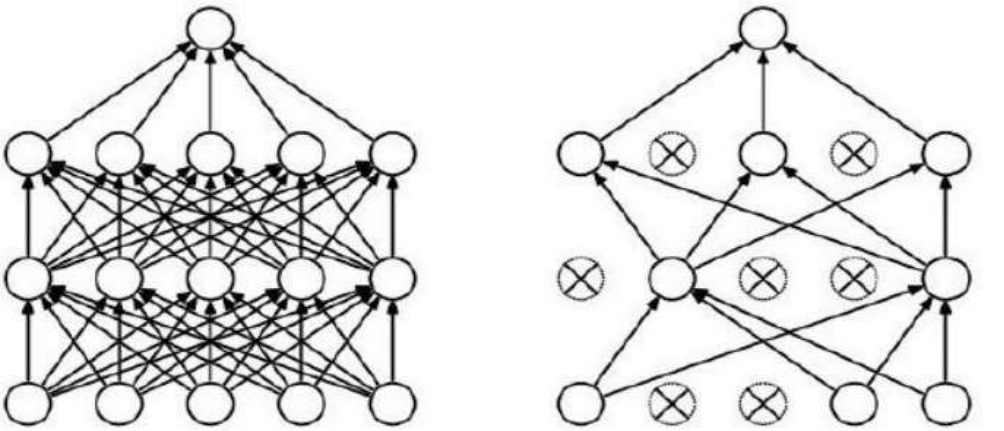
Benzer şekilde, doğrulama kaybının değerindeki artış nedeniyle aşırı öğrenme meydana gelir (Şekil 3.18).




Şekil 3.18. Doğrulama kayıp grafiği [48]

Eğitim sürecinde aşırı öğrenmeyi önlemek için model kapasitesi azaltılabilir veya seyreltme (dropout) tekniği kullanılabilir [49].

Srivastava ve arkadaşları tarafından önerilen seyreltme (dropout) tekniği, rastgele seçilen nöronların eğitim sırasında göz ardı edilmesi şeklinde çalışır [50]. (Şekil 3.19). Seyreltme tekniğinde, katmanların hangi çıktılarının çıkarılacağı veya katmanların hangi çıktılarının modelde tutulacağı olasılığını belirlemek için p hiper parametresi ($0 \leq p \leq 1$) belirlenir. 1,0 bırakma olmadığını gösterirken, 0,0 katmandan çıktı olmadığı anlamına gelir. Seyreltme, ağı daha az hassas hale getirir. Bu şekilde ağ, aşırı öğrenme sorununu azaltarak verilerdeki gürültü gibi çok ince ayrıntılara karşı duyarsız hale gelir.




Şekil 3.19. Seyreltme (Dropout) uygulaması [50]

Şekil 3.20-22'te, Python ile yazılmış ve TensorFlow [51] üzerinde çalışabilen üst düzey bir sinir ağı API'si olan Keras çerçevesini kullanan farklı büyüklükteki modeller için parametre sayısını gösterilmektedir.

```
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
dense (Dense)                (None, 16)                160016
_________________________________________________________________
dense_1 (Dense)              (None, 16)                272
_________________________________________________________________
dense_2 (Dense)              (None, 1)                 17
=================================================================
Total params: 160,305
Trainable params: 160,305
Non-trainable params: 0
```

Şekil 3.20. Temel derin öğrenme modeli [50]

```
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
dense_3 (Dense)              (None, 4)                 40004
_________________________________________________________________
dense_4 (Dense)              (None, 4)                 20
_________________________________________________________________
dense_5 (Dense)              (None, 1)                 5
=================================================================
Total params: 40,029
Trainable params: 40,029
Non-trainable params: 0
_________________________________________________________________
```

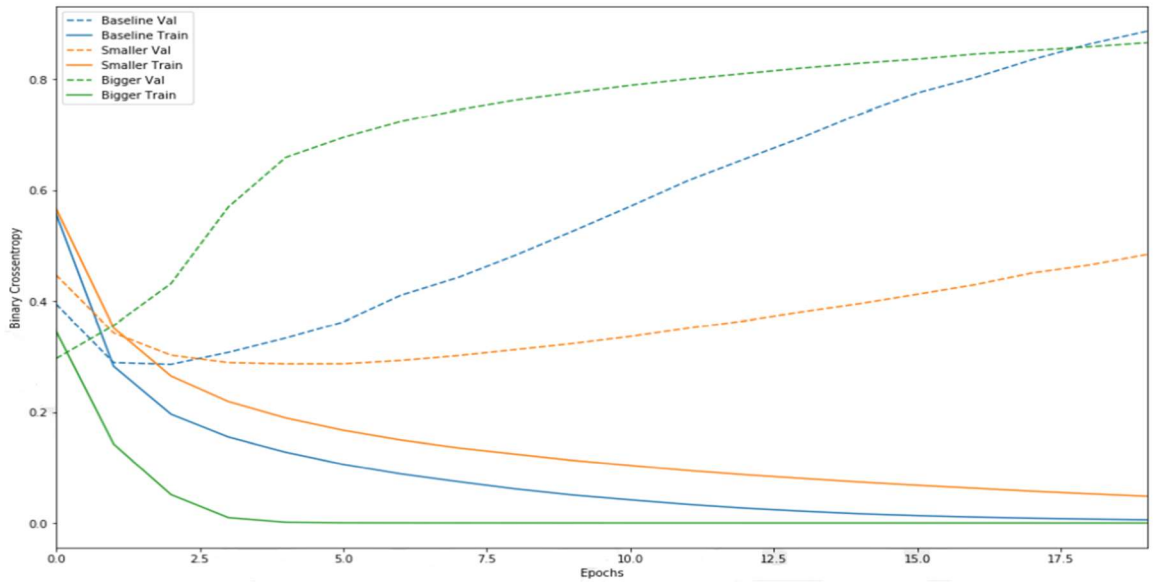
Şekil 3.21. Daha küçük derin öğrenme modeli [50]

| Layer (type) | Output Shape | Param # |
|---|---|---|
| dense_6 (Dense) | (None, 512) | 5120512 |
| dense_7 (Dense) | (None, 512) | 262656 |
| dense_8 (Dense) | (None, 1) | 513 |

Total params: 5,383,681
Trainable params: 5,383,681
Non-trainable params: 0

Şekil 3.22. Daha büyük derin öğrenme modeli [50]

Şekil 3.23'te gösterildiği gibi, mavi noktalı çizgi temel ağ doğrulama kaybını, mavi düz çizgi eğitim kaybını, sarı noktalı çizgi daha küçük ağ doğrulama kaybını, sarı düz çizgi eğitim kaybını, yeşil noktalı çizgi daha büyük ağ doğrulama kaybını, yeşil düz çizgi daha büyük ağ doğrulama kaybını temsil etmektedir. Şekil 3.23'te gösterildiği gibi, daha büyük ağ, sadece epoch'dan sonra neredeyse aşırı öğrenmeye başlamıştır. Daha büyük kapasite, daha az eğitim kaybı, aynı zamanda test seti için daha az doğruluk anlamına gelir [52].




Şekil 3.23. Farklı büyüklükteki ağlar için kayıp grafiği [50]

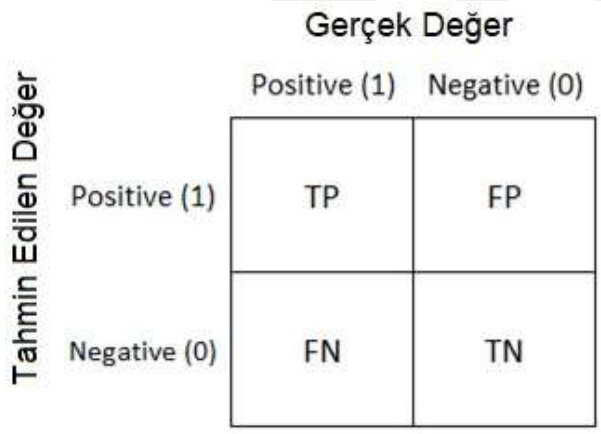
### 3.9. Değerlendirme ve Ölçüm Metrikleri

Veri setinin hazırlanması ve modellerinin eğitimi, derin öğrenme uygulamalarında önemli bir adım olsa da bu eğitilmiş modelin performansını ölçmek de oldukça önemlidir. Modelin görünmeyen veriler üzerinde ne kadar iyi genelleme yaptığı, uyarlanabilir ve

uyarlanabilir olmayan derin öğrenme modellerini tanımlayan şeydir [51]. Performans değerlendirmesi için farklı metrikler kullanarak, tasarlanan modelin genel tahmin gücünü iyileştirilebilmelidir. Farklı metrikler kullanarak derin öğrenme modelinin başarısını daha doğru ölçmek mümkündür. Doğruluk (Accuracy), Kesinlik (Precision), Hassasiyet (Recall), F-1 Skor başlıca değerlendirme metriklerindedir [52].

### 3.9.1. Karmaşıklık matrisi (Confisuon matrix)

Derin öğrenme uygulamalarında sınıflandırma problemi için bir performans ölçümüdür. Tahmin edilen ve gerçek değerlerin 4 farklı kombinasyonunu içeren tablodur. Karmaşıklık matrisinde, bilinmesi gereken bazı terimler vardır. Eğer tahmin edilecek problemi hasta olan bir kişi olarak tanımlarsak karmaşıklık matrisine göre aşağıdaki şekilde tanımlanabilir [53].




Şekil 3.24. Karmaşıklık matrisi

TP (True positive — Doğru Pozitif): Hasta olan birinin hasta olarak tanımlanması.
FP (False positive — Yanlış Pozitif): Hasta olmayan birinin hasta olarak tanımlanması.
TN (True negative — Doğru Negatif): Hasta olmayan birinin hasta değil olarak tanımlanması.
FN (False negative — Yanlış Negatif): Hasta olan birinin hasta değil olarak tanımlanması.

### 3.9.2. Doğruluk (Accuracy)

Doğru olarak sınıflandırılan örneklerin yüzdesidir. Veri setinde yer alan sınıfların yaklaşık olarak eşit büyüklükte olduğunda, doğru sınıflandırılmış değerler verecek olan doğruluk

(accuracy) kullanabilir. Doğruluk, sınıflandırma problemleri için yaygın bir değerlendirme metriğidir. Yapılan tüm tahminlerin oranı olarak yapılan doğru tahminlerin sayısıdır (Eş. 3.7).

$$\text{Accuracy} = (TP+TN) / (TP+FP+FN+TN) \quad (3.7)$$

### 3.9.3. Kesinlik (Precision)

Veri setinde yer alan sınıfların dengesiz olduğu durumda doğruluk (accuracy), performansımızı ölçmek için güvenilmez bir ölçü haline gelebilir. Böyle bir durumda kesinlik (precision) değerlendirme metriği daha doğru sonuç vermektedir (Eş. 3.8).

$$\text{Precision} = (TP) / (TP+FP) \quad (3.8)$$

### 3.9.4. Duyarlılık (Recall)

Duyarlılık (Recall), pozitif olarak tahmin edilmesi gereken işlemlerin ne kadarının pozitif olarak tahmin edildiğini gösteren bir metriktir. Biçimsel olarak, doğru pozitif cevap sayısı (true pozitif) ile doğru pozitif cevap (true pozitif) ve yanlış negatif olarak etiketlenmiş cevap (false negatif) arasındaki orandır (Eş. 3.9).

$$\text{Recall} = (TP) / (TP+FN) \quad (3.9)$$

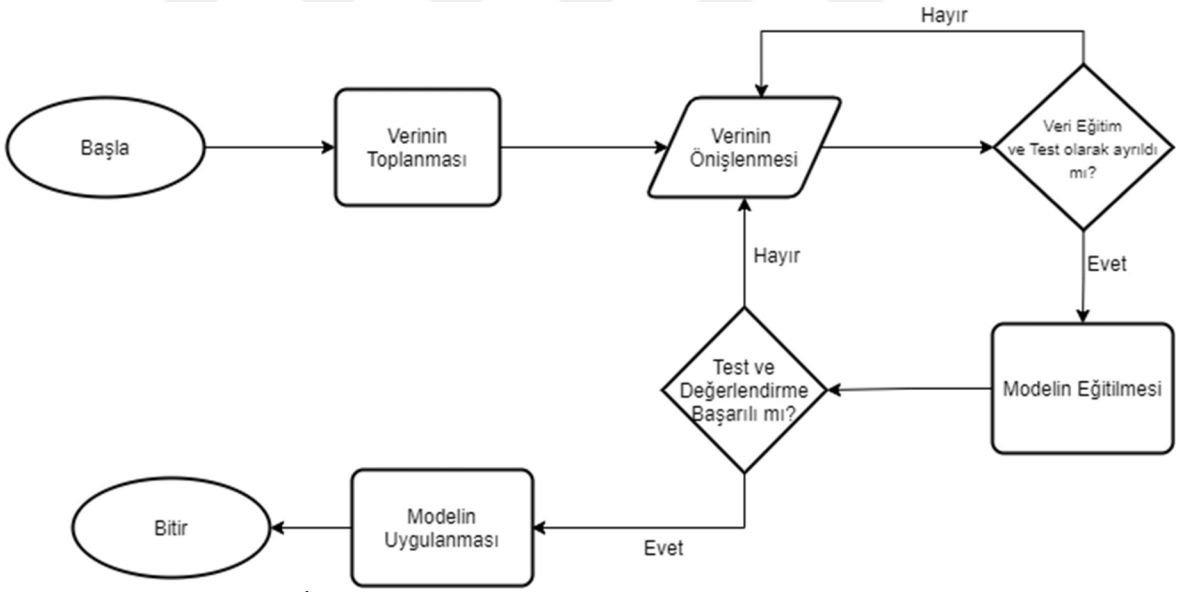
### 3.9.5. F1 skor

F1 skoru, bir testin doğruluğunun ölçüsüdür. Kesinlik (Accuracy) ve duyarlılığın (Recall) harmonik ortalamasıdır. Maksimum 1 (mükemmel kesinlik ve duyarlılık) ve minimum 0'a değerine sahiptir. Genel olarak, modelin kesinlik ve sağlamlık ölçüsünün göstergesidir. Düşük kesinlik ve yüksek duyarlılık değerlerine sahip iki modeli karşılaştırmak zordur. Bu sebeple karşılaştırılabilir hale getirmek için F1 Skoru kullanılmaktadır (Eş. 3.10).

$$\text{F1 Skor} = (2*\text{Precision}*\text{Recall}) / (\text{Precision} + \text{Recall}) \quad (3.10)$$

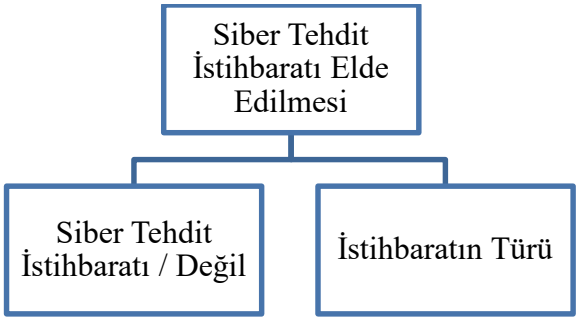
# 4. ARAŞTIRMA, BULGU VE GÖZLEMLER

Açık kaynaklardan siber tehdit istihbaratı elde edilmesi çalışmasında açık kaynak olarak Twitter'dan faydalanılmıştır. Bu çalışmada oldukça zengin veri içeren Twitter açık kaynak platformudan Siber Tehdit İstihbaratı (STİ) elde edilmesi çalışması yapılmıştır. Twitter yaklaşık 200 milyon kullanıcı ve günlük atılan 500 milyon tweet ile oldukça büyük bir veri kaynağı oluşturmaktadır [6]. Twitter'da yer alan verilerin yapısal olmaması sebebiyle doğal dil işleme ve derin öğrenme yöntemlerine başvurulmuştur. Çalışma siber tehdit istihbaratı ile ilgili oluşturan verilerin çok büyük bir kısmın İngilizce dilinde olması sebebiyle İngilizce dilinde oluşturulan bir veri seti üzerinde yapılmıştır. Çalışma python yazılım dilinde Şekil 4.1. de gösterilen akış diyagramına uygun olarak gerçekleştirilmiştir.




Şekil 4.1. Siber Tehdit İstihbaratı Elde Edilmesi Akış Diyagramı

Bu bölümde araştırma, bulgu ve gözlemlerden bahsedilecektir. İlk olarak çalışmada kullanılan veri seti hakkında bilgiler verilecek, ikinci bölümde veri setinde yer alan verilerin derin öğrenme öncesi ön işlenmesi, üçüncü bölümde bu çalışmada kullanılan modellerden bahsedilmiştir. Çalışma iki ayrı aşamada gerçekleştirilmiştir. İlk olarak veri setinde yer alan verilerin Siber Tehdit İstihbaratı ile ilgili olup olamaması sınıflandırması yapılmış, ikinci aşamada ise siber tehdit istihbaratının türüne yönelik sınıflandırma yapılmıştır.




Şekil 4.2. Siber tehdit istihbaratı elde edilmesi çalışma safhaları

## 4.1. Veri Seti

Bu çalışmada Behzadan ve arkadaşları tarafından sağlanan açık kaynak veri setinden yararlanılmıştır [13]. Yazarlar, Twitter'dan 21 368 tweet'lik bir veri seti oluşturmuştur. Tweetlerin canlı akışını dinlemek için Tweepy [54] ile yapılmış özel bir akış dinleyicisi geliştirmişlerdir. Akış dinleyici sonuçlarını önceden filtrelemek ve daraltmak için bir anahtar kelime listesi oluşturmuşlardır. " vulnerability" ve "0day" gibi genel kelimeler, STİ ile genel alakaları nedeniyle seçilirken, bunların yanı sıra daha hedefli veri üretmek için "DDoS", "SQLinjection", " buffer overflow" gibi belirli tehdit türleriyle ilgili kelimeler de seçilmiştir. Tweet veri seti, dört günlük bir süre boyunca sürekli olarak dinlenen bir akışın sonucunda oluşturulmuştur. Oluşturulan veri setinde STİ ile ilgili veriler ile birlikte ilgili olmayan veriler de bulunmaktadır. Ayrıca veri setinde farlı tehdit türlerinde veriler de yer almaktadır. Veri setinde yer alan verilerin siber tehdit istihbaratı ile ilgili olup olamama durumu Çizelge 4.1'de verilerin tehdit türü durumu Çizelge 4.2'de sunulmuştur.

Çizelge 4.1. Veri setinde yer alan tweetlerin STİ ile ilgili olup olmama durumu

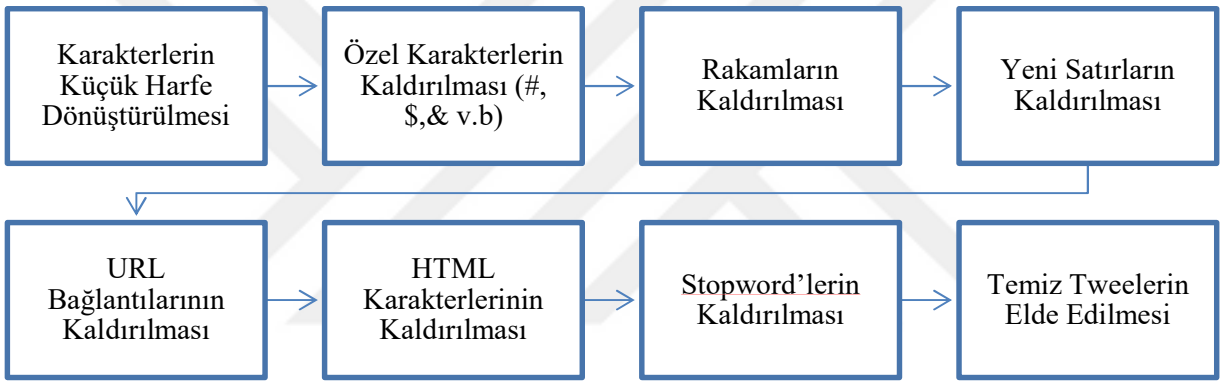
| Veri Tipi | Veri Adedi |
|---|---|
| Relevant | 14.770 |
| Irrelevant | 6.598 |

Çizelge 4.2. Veri setinde yer alan tweetlerin STİ türüne göre dağılımı

| Veri Tipi | Veri Adedi |
|---|---|
| Vulnerability | 7.354 |
| Ransomware | 3.203 |
| DDoS | 2.245 |
| General | 6.998 |
| ZeroDay | 730 |
| Botnet | 702 |
| Leak | 136 |

## 4.2. Veri Ön İşleme

Derin öğrenme işlemine başlamadan önce öğrenme sürecinde kullanılacak olan veri setinde yer alan gürültülü ve işlem sırasında ihtiyaç duyulmayan verilerin veri setinden çıkarılması gerekmektedir. Veri ön işleme kapsamında sırası ile veri setinde yer alan tüm veriler küçük harfe dönüştürülmüş, özel karakterler, rakamlar, metin içerisinde yer alan yeni satırlar, url bağlantıları, html karakterleri ve stopwordler kaldırılmıştır. Bu işlemin sonucunda veri seti derin öğrenme sürecine hazır hale getirilmiş temiz tweetler elde edilmiştir. Ayrıca leak etiketine sahip satırlar derin öğrenme için yeterli sayıda olması sebebiyle veri setinden çıkarılmıştır. Söz konusu işlem süreçleri Şekil 4.3'te gösterilmiştir.




Şekil 4.3. Veri önişleme aşamaları

## 4.3. Önerilen Modeller

Bu çalışmada, verinin ön işlenmesi aşamasından sonra ilk olarak siber tehdit istihbaratı verisi olup olmadığı ardından istihbaratın türü sınıflandırması yapılmıştır. Çalışmada özyinemeli sinir ağlarından faydalanılmış, LSTM ve GRU algoritmalarından kullanılarak her iki sınıflandırma için 6'şar adet model sunulmuştur.

### 4.3.1. Siber istihbarat verisi/değil sınıflandırması

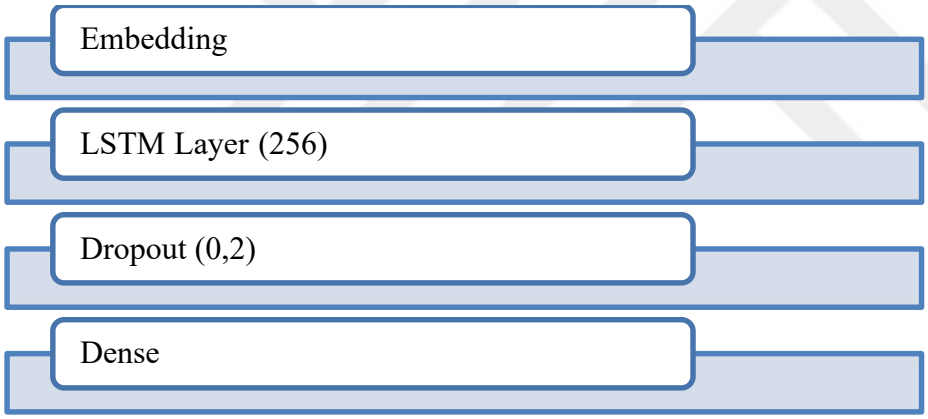
Verinin ön işlenmesinin ardından ilk olarak siber tehdit istihbarat verisi olup olmadığı sınıflandırması yapılmıştır. Bu çalışmada 6 adet model üzerinde çalışılmıştır. Çalışma esnasında farklı hiper parametreler uygulanmış en iyi sonuç alınan parametreler Çizelge 4.3'te sunulmuştur.

Çizelge 4.3 Siber istihbarat verisi/değil sınıflandırması kullanılan hiper parametreler

| Parametre | Değer |
|---|---|
| Max Feature | 250 |
| Output Dimension | 10 |
| Bach Size | 256 |
| Aktivasyon Fonksiyonu | sigmoid |
| Optimizer | adam |
| Epoc | 15 |

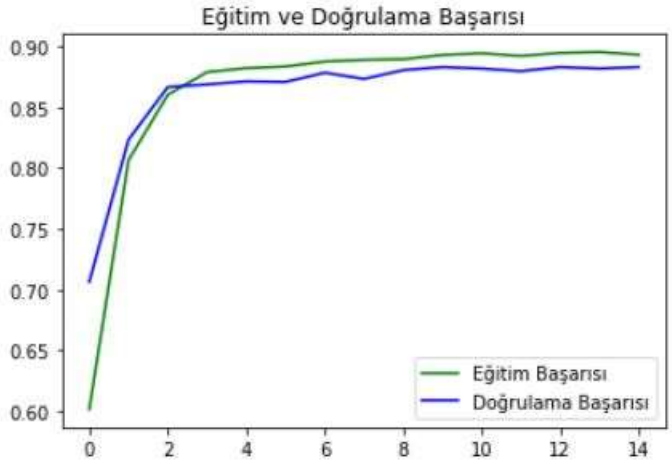
Model 1

İlk modelde Embedding katmanının oluşturulmasının ardından 256 katmanlı LSTM algoritmasından geçirilmiş, 0,2'lik seyreltme (dropout) uygulanmış ve dense katmanı oluşturulmuştur. Bu işlem sonunda 276 422 adet eğitim parametresi elde edilmiştir. Oluşturulan model Şekil 4.4'te sunulmuştur.

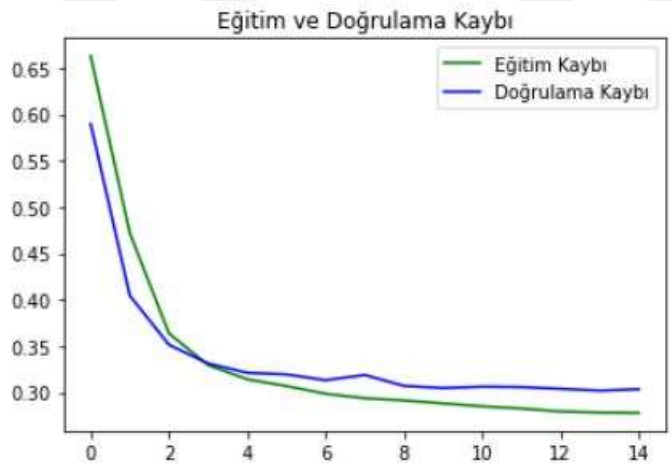



Şekil 4.4. Siber istihbarat verisi/değil sınıflandırması model 1

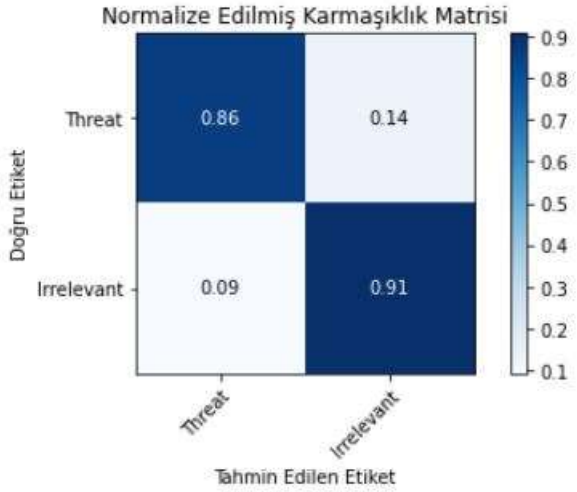
Bu model ile yapılan eğitim sonucunda %88,14 başarı elde edilmiştir. Modelin eğitimi sonucunda elde edilen başarı grafiği Şekil 4.5'te kayıp grafiği Şekil 4.6'da, Karmaşıklık Matrisi Şekil 4.7'de sunulmuştur.




Şekil 4.5. Siber istihbarat verisi/değil sınıflandırması model 1 başarı grafiği



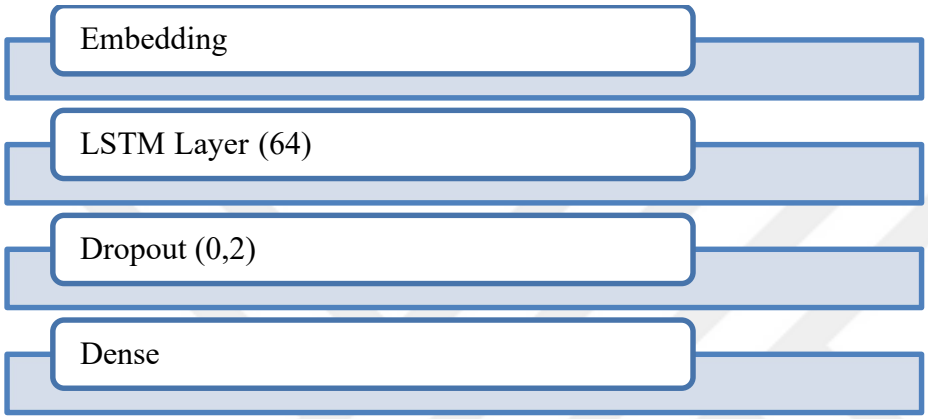

Şekil 4.6. Siber istihbarat verisi/değil sınıflandırması model 1 kayıp grafiği




Şekil 4.7. Siber istihbarat verisi/değil sınıflandırması model 1 karmaşıklık matrisi

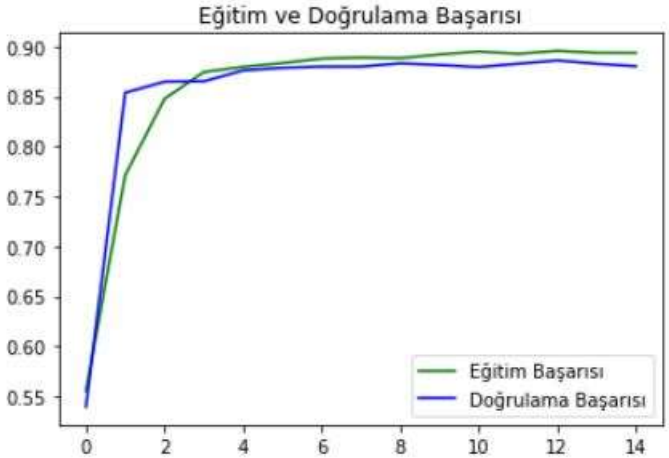
Model 2

İkinci modelde Embedding katmanının oluşturulmasının ardından LSTM katmanı azaltılarak 64'e düşürülmüş, 0,2'lik seyreltme (dropout) uygulanmış ve dense katmanı oluşturulmuştur. Bu işlem sonunda 21 830 eğitim parametresi elde edilmiştir. Oluşturulan model Şekil 4.8'de sunulmuştur.

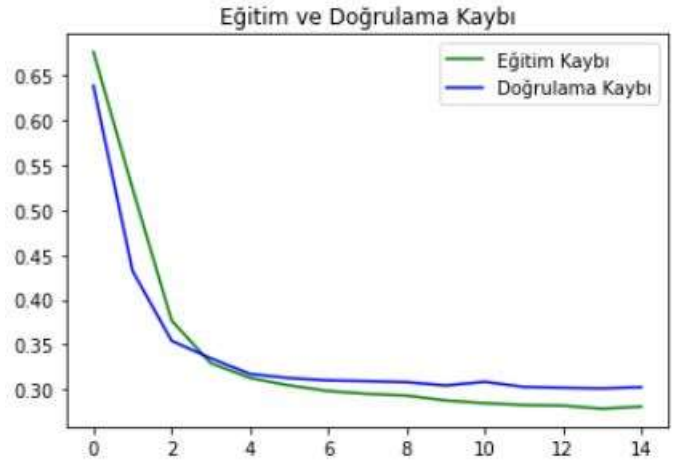



Şekil 4.8. Siber istihbarat verisi/değil sınıflandırması model 2

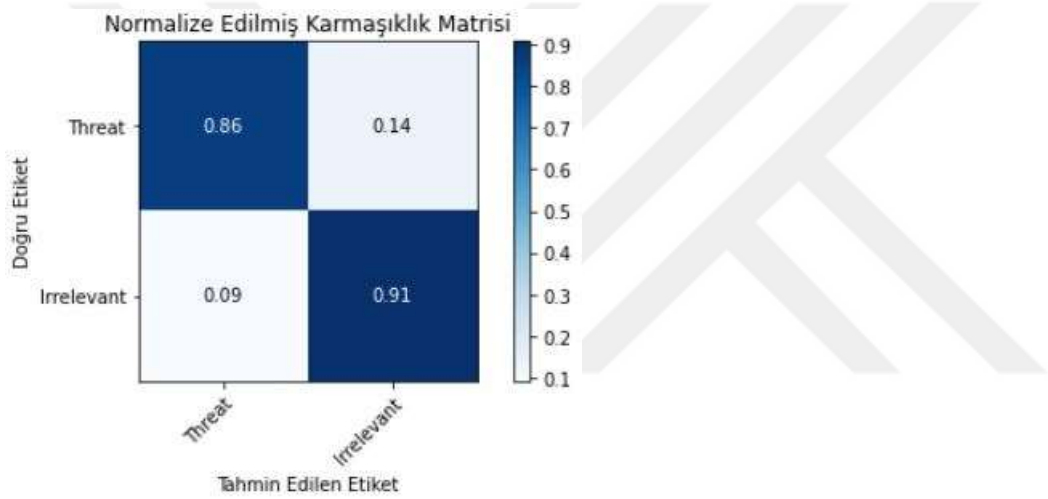
Bu model ile yapılan eğitim sonucunda %87,80 başarı elde edilmiştir. Modelin eğitimi sonucunda elde edilen başarı grafiği Şekil 4.9'da kayıp grafiği Şekil 4.10'da Karmaşıklık Matrisi Şekil 4.11'de sunulmuştur.




Şekil 4.9. Siber istihbarat verisi/değil sınıflandırması model 2 başarı grafiği



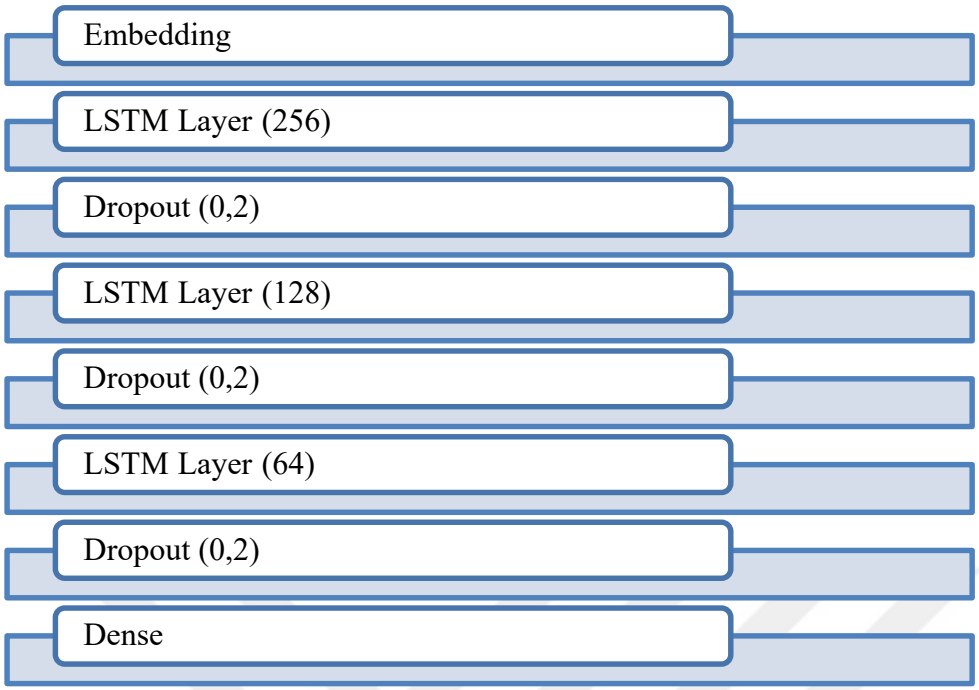

Şekil 4.10. Siber istihbarat verisi/değil sınıflandırması model 2 kayıp grafiği




Şekil 4.11. Siber istihbarat verisi/değil sınıflandırması model 2 karmaşıklık matrisi

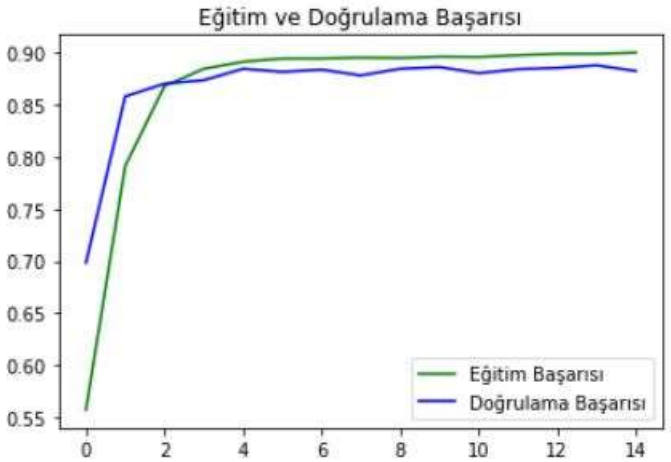
Model 3

Üçüncü modelde Embedding katmanının oluşturulmasının ardından üç adet LSTM katmanı yığın (stack) olarak oluşturulmuştur. Birinci LSTM 256, ikinci 128, üçüncü 64 katmandan oluşmaktadır. Her LSTM katmanından sonra 0,2'lik seyreltme (dropout) uygulanmış ardından dense katmanı oluşturulmuştur. Bu işlem sonunda 522 566 adet eğitim parametresi elde edilmiştir. Oluşturulan model Şekil 4.12'de sunulmuştur.

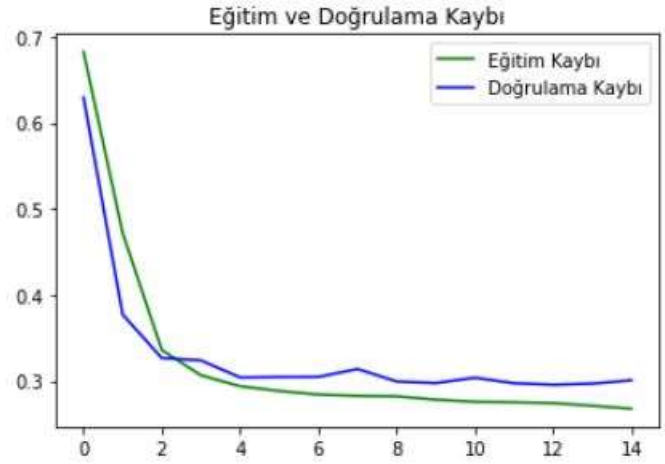



Şekil 4.12. Siber istihbarat verisi/değil sınıflandırması model 3

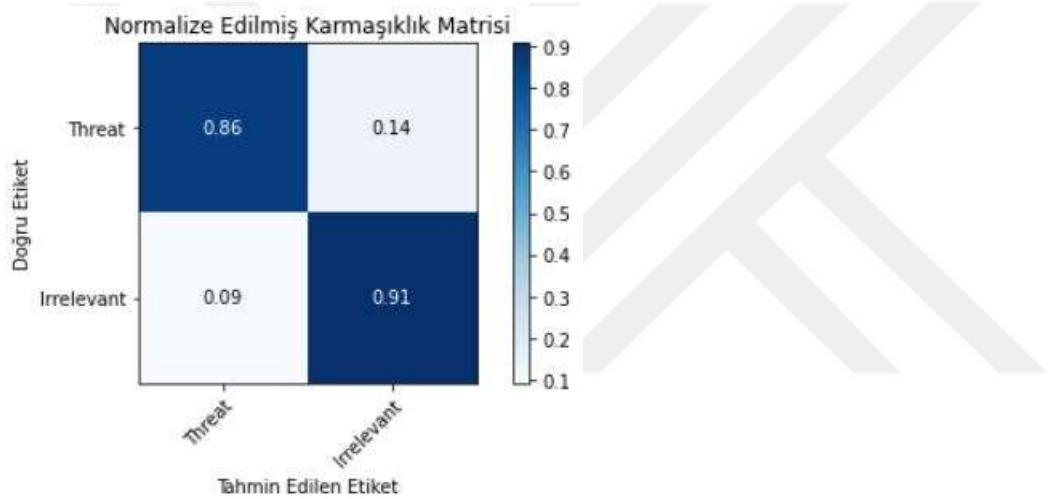
Bu model ile yapılan eğitim sonucunda %88,34 başarı elde edilmiştir. Modelin eğitimi sonucunda elde edilen başarı grafiği Şekil 4.13'te kayıp grafiği Şekil 4.14'de, Karmaşıklık Matrisi Şekil 4.15'te sunulmuştur.




Şekil 4.13. Siber istihbarat verisi/değil sınıflandırması model 3 başarı grafiği



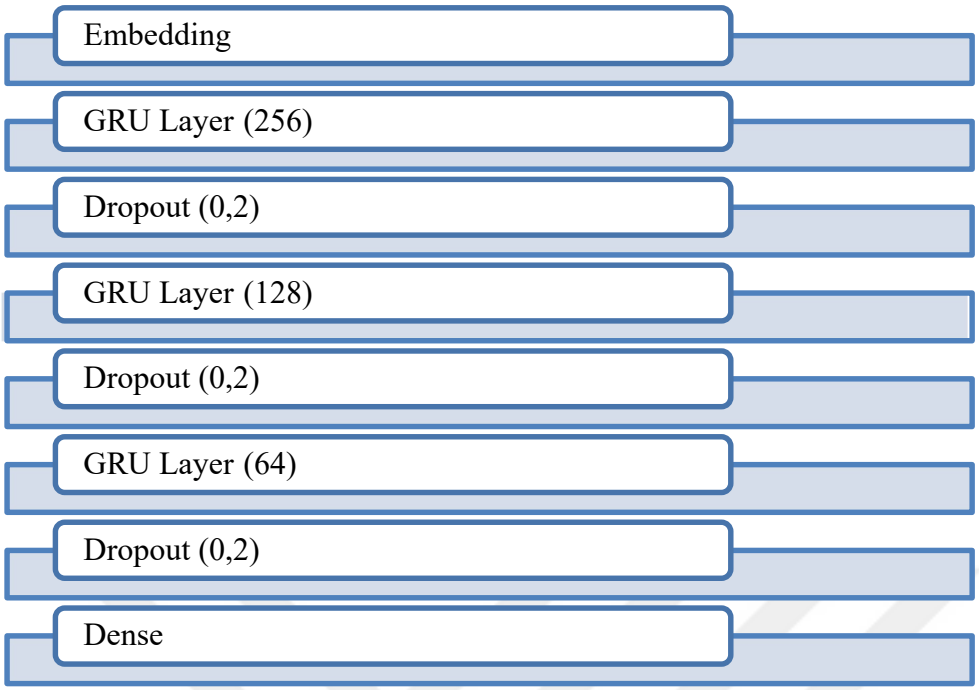

Şekil 4.14. Siber istihbarat verisi/değil sınıflandırması model 3 kayıp grafiği




Şekil 4.15. Siber istihbarat verisi/değil sınıflandırması model 3 karmaşıklık matrisi

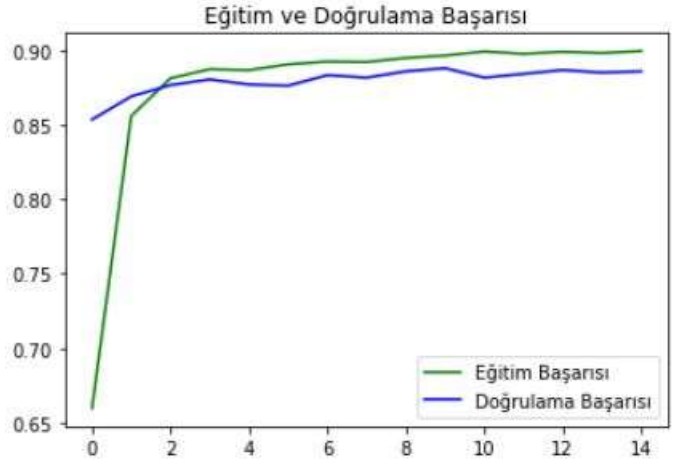
Model 4

Dördüncü modelde Embedding katmanının oluşturulmasının ardından üç adet GRU katmanı yığın (stack) olarak oluşturulmuştur. Birinci GRU 256, ikinci 128, üçüncü 64 katmandan oluşmaktadır. Her GRU katmanından sonra 0,2'lik seyreltme (dropout) uygulanmış ardından dense katmanı oluşturulmuştur. Bu işlem sonunda 522 566 adet eğitim parametresi elde edilmiştir. Oluşturulan model Şekil 4.16'da sunulmuştur.

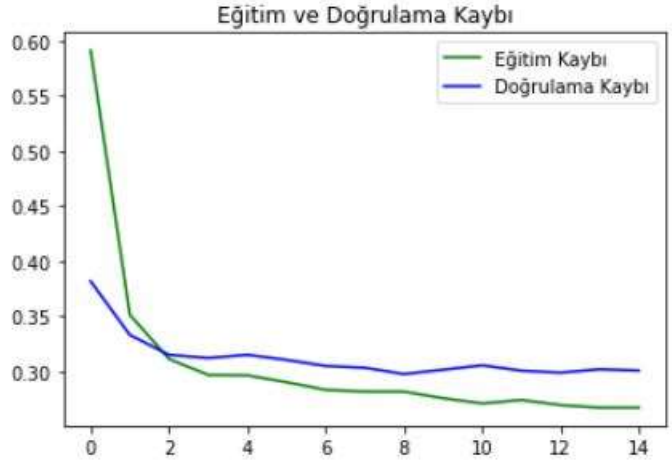



Şekil 4.16. Siber istihbarat verisi/değil sınıflandırması model 4

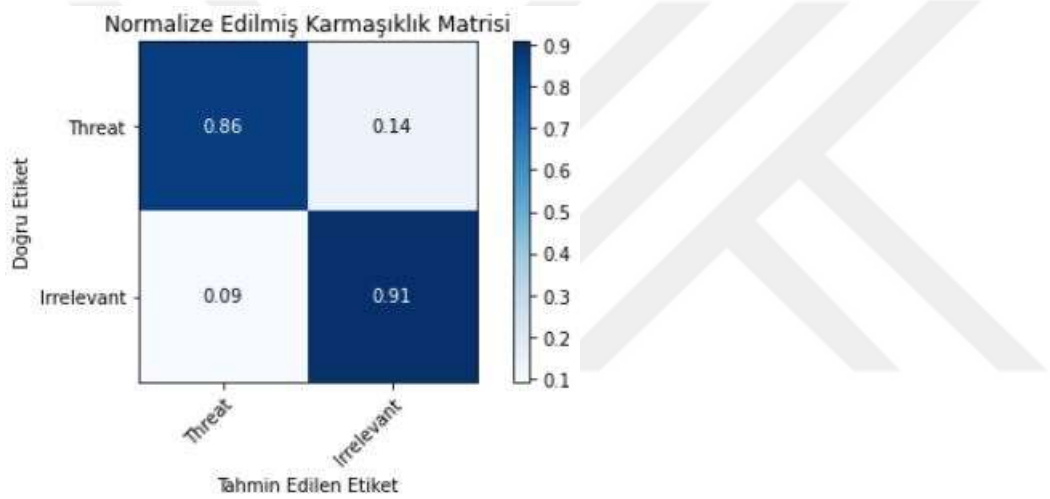
Bu model ile yapılan eğitim sonucunda %88,51 başarı elde edilmiştir. Modelin eğitimi sonucunda elde edilen başarı grafiği Şekil 4.17'de kayıp grafiği Şekil 4.18'de, Karmaşıklık Matrisi Şekil 4.19'da sunulmuştur.




Şekil 4.17. Siber istihbarat verisi/değil sınıflandırması model 4 başarı grafiği



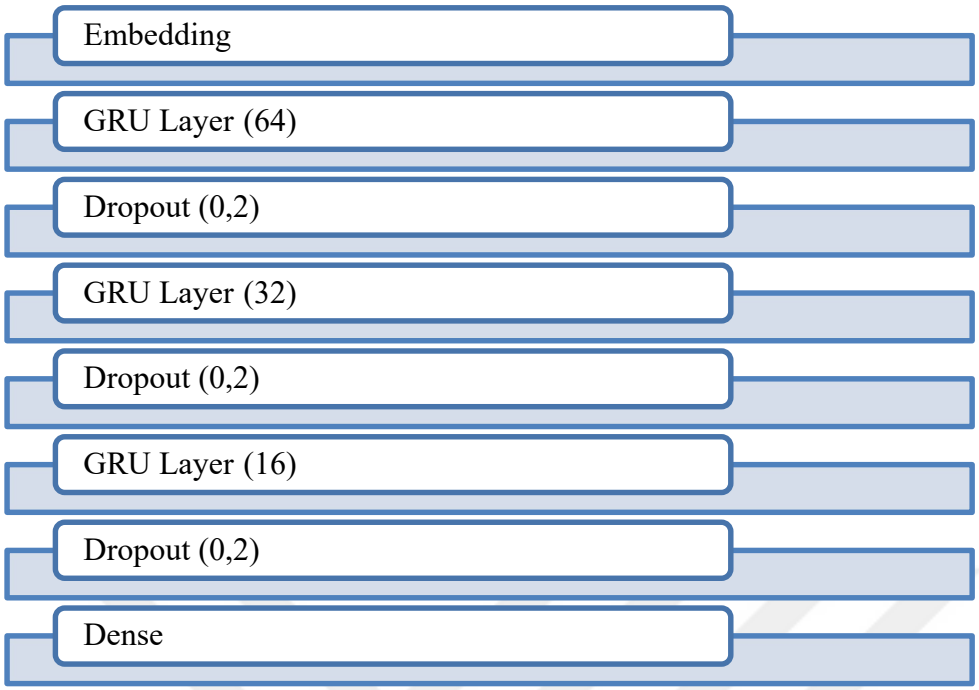

Şekil 4.18. Siber istihbarat verisi/değil sınıflandırması model 4 kayıp grafiği




Şekil 4.19. Siber istihbarat verisi/değil sınıflandırması model 4 karmaşıklık matrisi

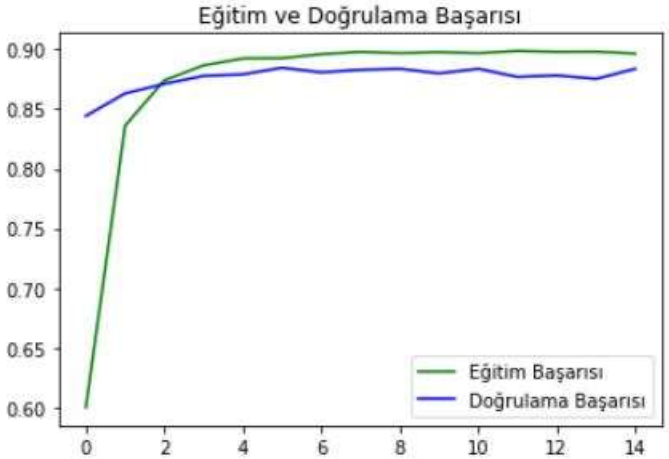
Model 5

Beşinci modelde Embedding katmanının oluşturulmasının ardından GRU yığınında yer alan katman sayısı azaltılmıştır. Birinci GRU 64, ikinci 32, üçüncü 16 katmandan oluşmaktadır. Her GRU katmanından sonra 0,2'lik seyreltme (dropout) uygulanmış ardından dense katmanı oluşturulmuştur. Bu işlem sonunda 28 934 adet eğitim parametresi elde edilmiştir. Oluşturulan model Şekil 4.20'de sunulmuştur.

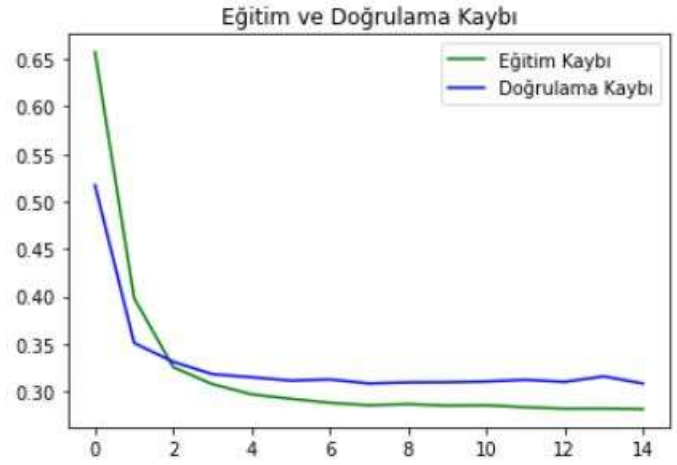



Şekil 4.20. Siber istihbarat verisi/değil sınıflandırması model 5

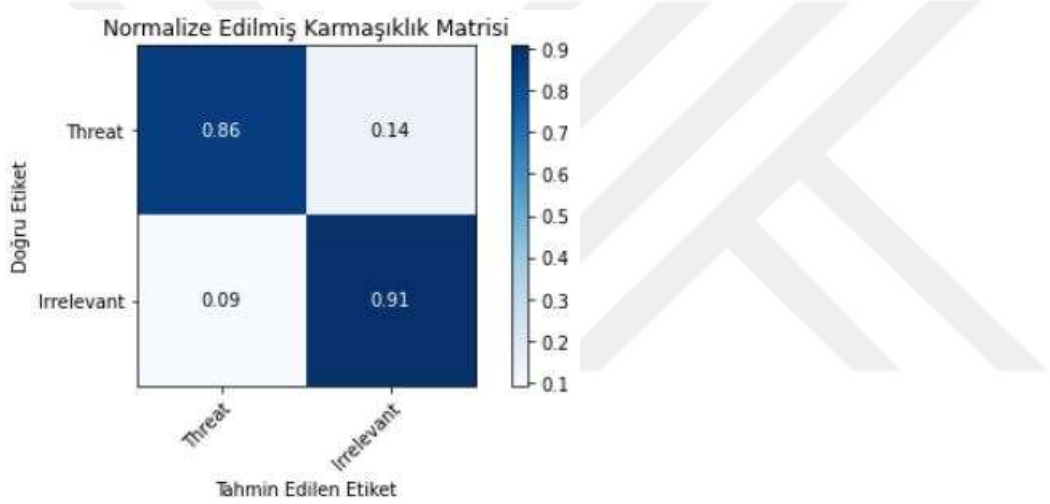
Bu model ile yapılan eğitim sonucunda %88,24 başarı elde edilmiştir. Modelin eğitimi sonucunda elde edilen başarı grafiği Şekil 4.21'de kayıp grafiği Şekil 4.22'de Karmaşıklık Matrisi Şekil 4.23'te sunulmuştur.




Şekil 4.21. Siber istihbarat verisi/değil sınıflandırması model 5 başarı grafiği



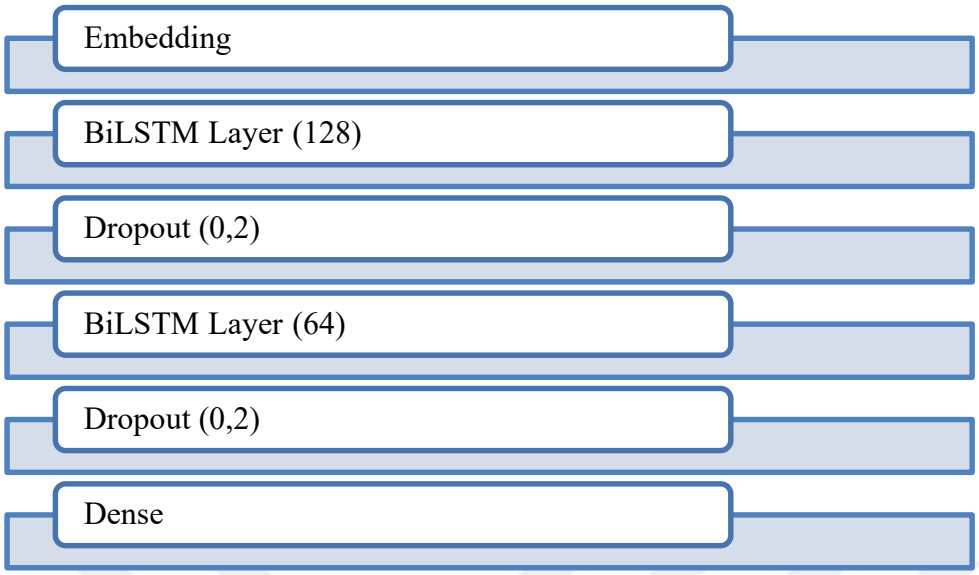

Şekil 4.22. Siber istihbarat verisi/değil sınıflandırması model 5 kayıp grafiği




Şekil 4.23. Siber istihbarat verisi/değil sınıflandırması model 5 karmaşıklık matrisi

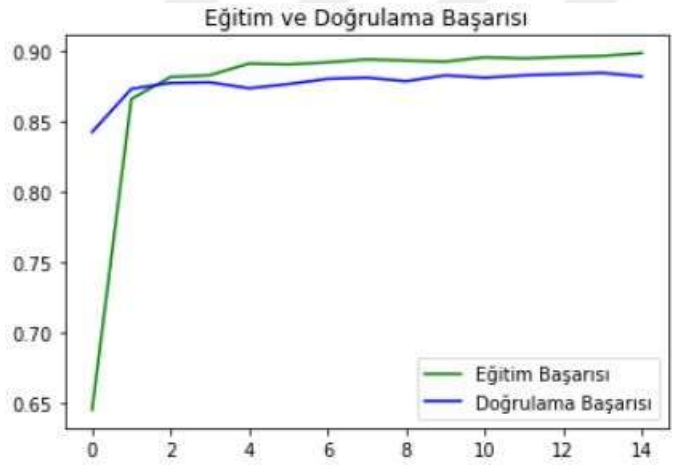
Model 6

Altınca modelde Embedding katmanının oluşturulmasının ardından BiLSTM yığını oluşturulmuştur. Birinci BiLSTM 128, ikinci 64 katmandan oluşmaktadır. Her BiLSTM katmanından sonra 0,2'lik seyreltme (dropout) uygulanmış ardından dense katmanı oluşturulmuştur. Bu işlem sonunda 309 446 adet eğitim parametresi elde edilmiştir. Oluşturulan model Şekil 4.24'te sunulmuştur.

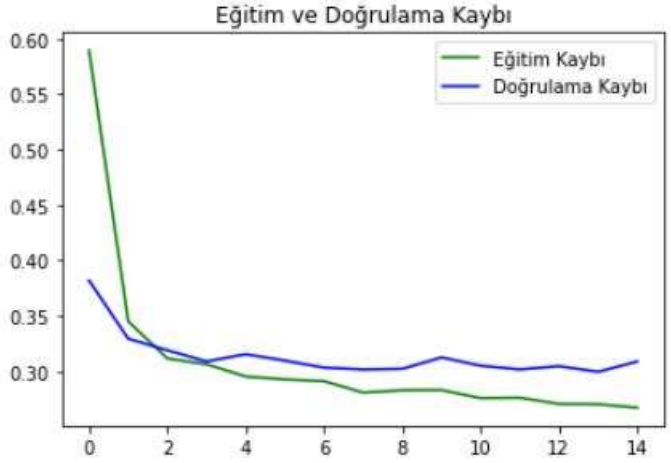



Şekil 4.24. Siber istihbarat verisi/değil sınıflandırması model 6

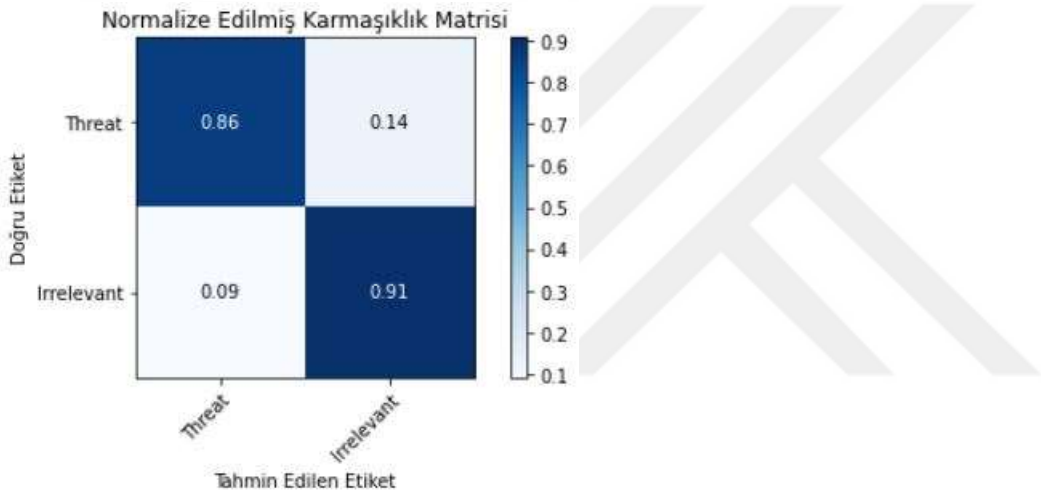
Bu model ile yapılan eğitim sonucunda %88,64 başarı elde edilmiştir. Modelin eğitimi sonucunda elde edilen başarı grafiği Şekil 4.25'te kayıp grafiği Şekil 4.26'da, Karmaşıklık Matrisi Şekil 4.27'de sunulmuştur.




Şekil 4.25. Siber istihbarat verisi/değil sınıflandırması model 6 başarı grafiği




Şekil 4.26. Siber istihbarat verisi/değil sınıflandırması model 6 kayıp grafiği




Şekil 4.27. Siber istihbarat verisi/değil sınıflandırması model 6 karmaşıklık matrisi

Çalışmada kullanılan veri setinde yer alan verilerin Siber Tehdit İstihbaratı ile ilgili olup olamama durumu sınıflandırmasında, eğitimi yapılan altı model arasında altıncı modelin en başarılı model olduğu görülmüştür. Eğitilen modellere ait başarı tablosu Çizelge 4.4'te sunulmuştur.

Çizelge 4.4. Siber tehdit istihbaratı/değil sınıflandırması başarı tablosu

| Model | Başarı Oranı |
|---|---|
| Model 1 | 88,14 |
| Model 2 | 87,80 |
| Model 3 | 88,34 |
| Model 4 | 88,51 |
| Model 5 | 88,24 |
| Model 6 | 88,64 |

### 4.3.2. Siber istihbarat türü sınıflandırması

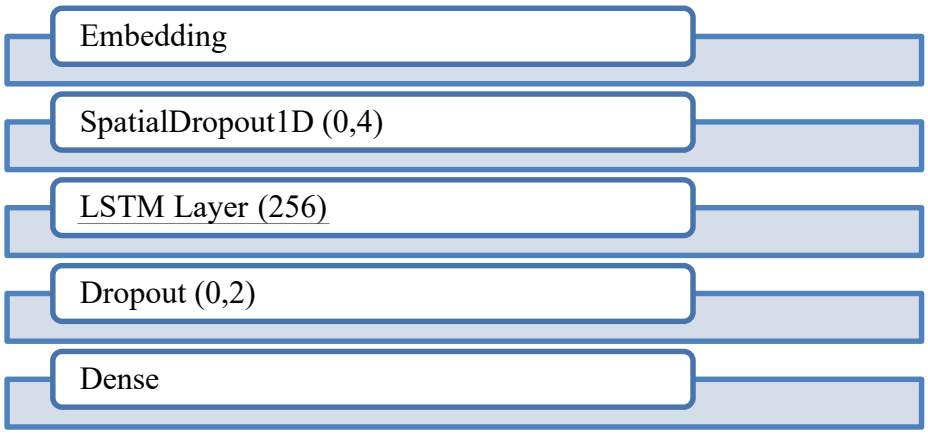
İkinci aşamada siber istihbarat türü sınıflandırması yapılmıştır. Bu çalışmada da diğer çalışmada olduğu gibi 6 adet model üzerinde çalışılmıştır. Çalışma esnasında farklı hiper parametreler uygulanmış en iyi sonuç alınan parametreler Çizelge 4.5'te sunulmuştur.

Çizelge 4.5 Siber istihbarat türü sınıflandırması kullanılan hiper parametreler

| Parametre | Değer |
|---|---|
| Max Feature | 750 |
| Output Dimension | 50 |
| Bach Size | 256 |
| Aktivasyon Fonksiyonu | softmax |
| Optimizer | adam |
| Epoc | 25 |

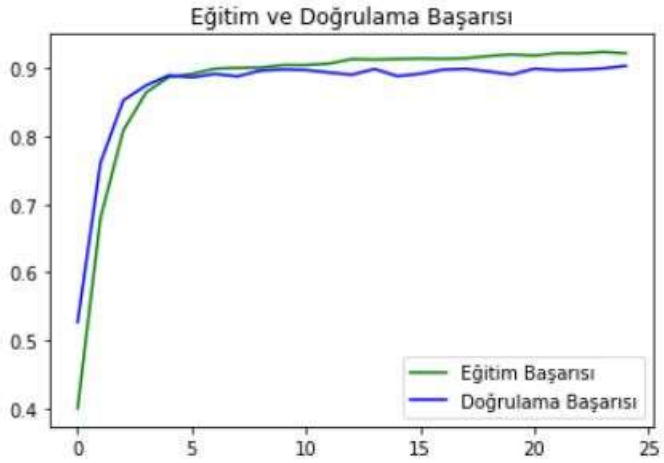
Model 1

İlk modelde Embedding katmanının oluşturulmasının ardından 1 boyutlu SpatialDropout katmanı oluşturulmuş ve 0,4'lük seyreltme uygulanmıştır. Ardından 256 katmanlı LSTM algoritmasından geçirilmiş, 0,2'lik seyreltme (dropout) uygulanmış ve dense katmanı oluşturulmuştur. Bu işlem sonunda 353 410 adet eğitim parametresi elde edilmiştir. Oluşturulan model Şekil 4.28'de sunulmuştur.

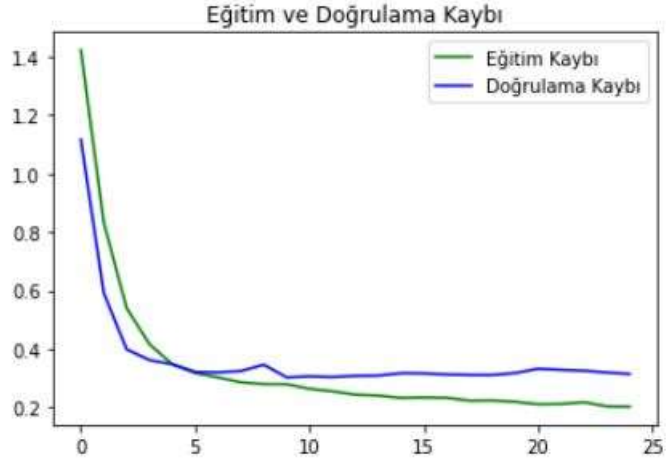



Şekil 4.28. Siber tehdit istihbaratı türü sınıflandırması model 1

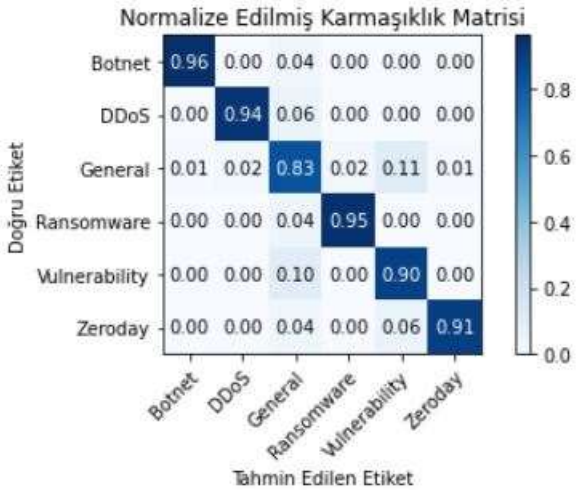
Bu model ile yapılan eğitim sonucunda %89,49 başarı elde edilmiştir. Modelin eğitimi sonucunda elde edilen başarı grafiği Şekil 4.29'da kayıp grafiği Şekil 4.30'da Karmaşıklık Matrisi Şekil 4.31'de sunulmuştur.




Şekil 4.29. Siber tehdit istihbaratı türü sınıflandırması model 1 başarı grafiği



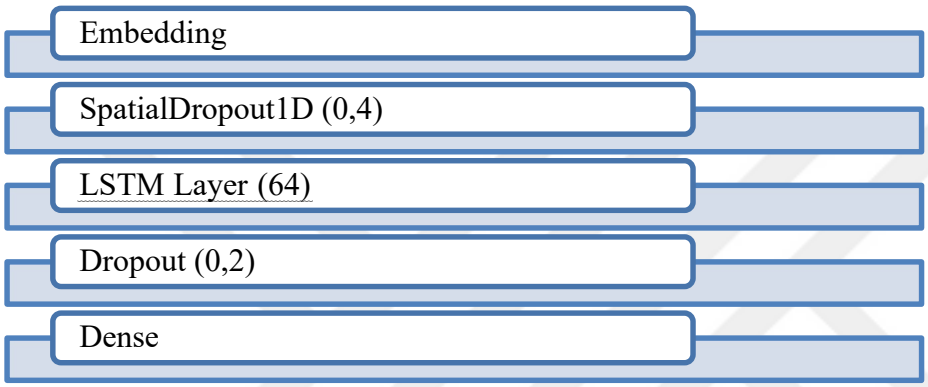

Şekil 4.30. Siber tehdit istihbaratı türü sınıflandırması model 1 kayıp grafiği




Şekil 4.31. Siber tehdit istihbaratı türü sınıflandırması model 1 karmaşıklık matrisi

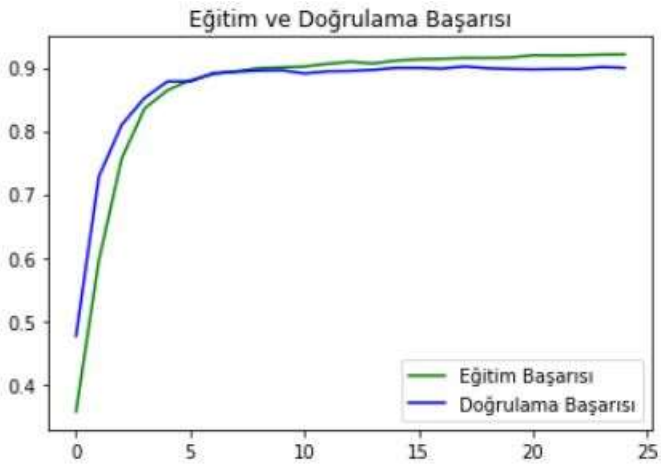
Model 2

İkinci modelde Embedding katmanının oluşturulmasının ardından 1 boyutlu SpatialDropout katmanı oluşturulmuş ve 0,4'lük seyreltme uygulanmıştır. LSTM katmanı azaltılarak 256'ya düşürülmüş, 0,2'lik seyreltme (dropout) uygulanmış ve dense katmanı oluşturulmuştur. Bu işlem sonunda 67 330 adet eğitim parametresi elde edilmiştir. Oluşturulan model Şekil 4.32'de sunulmuştur.

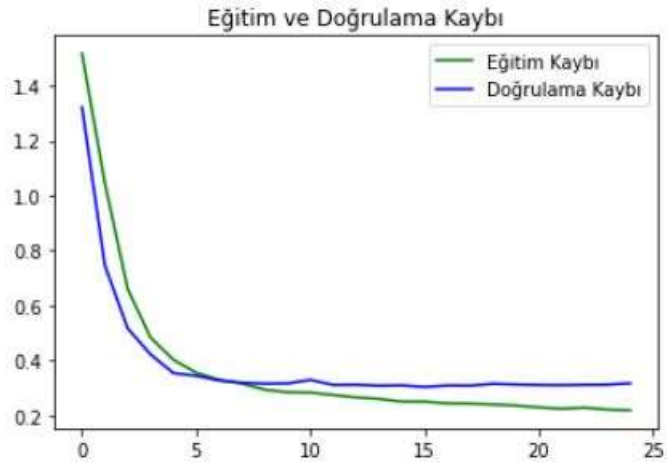



Şekil 4.32. Siber tehdit istihbaratı türü sınıflandırması model 2

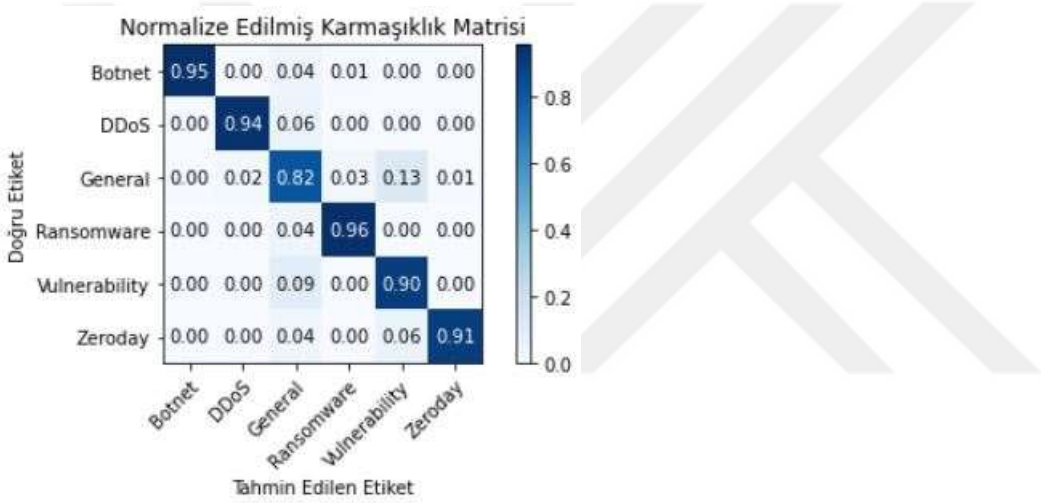
Bu model ile yapılan eğitim sonucunda %88,86 başarı elde edilmiştir. Modelin eğitimi sonucunda elde edilen başarı grafiği Şekil 4.33'te kayıp grafiği Şekil 4.34'te, Karmaşıklık Matrisi Şekil 4.35'te sunulmuştur.




Şekil 4.33. Siber tehdit istihbaratı türü sınıflandırması model 2 başarı grafiği



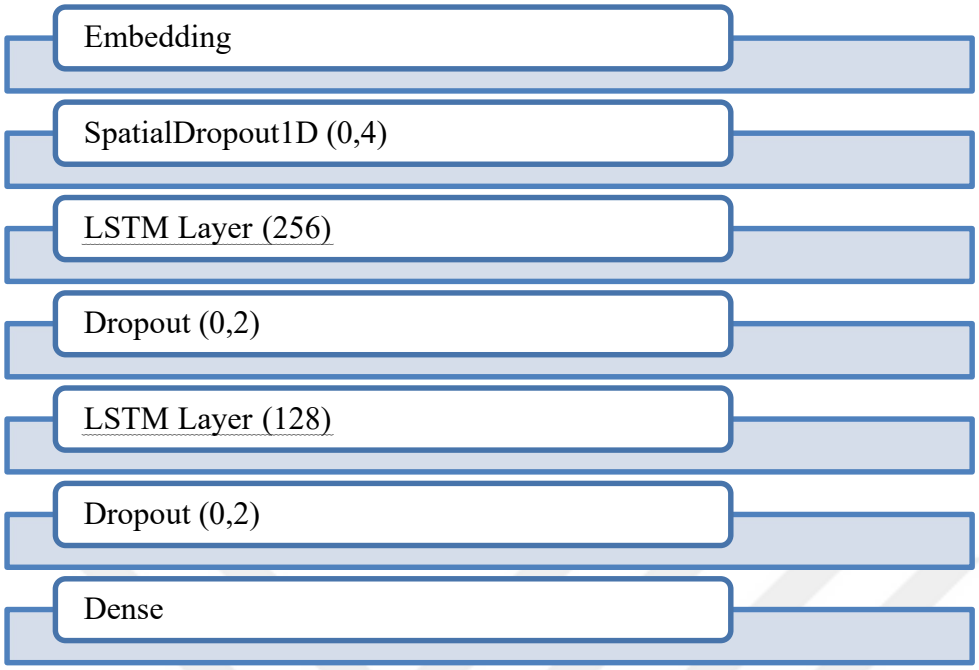

Şekil 4.34. Siber tehdit istihbaratı türü sınıflandırması model 2 kayıp grafiği




Şekil 4.35. Siber tehdit istihbaratı türü sınıflandırması model 2 karmaşıklık matrisi

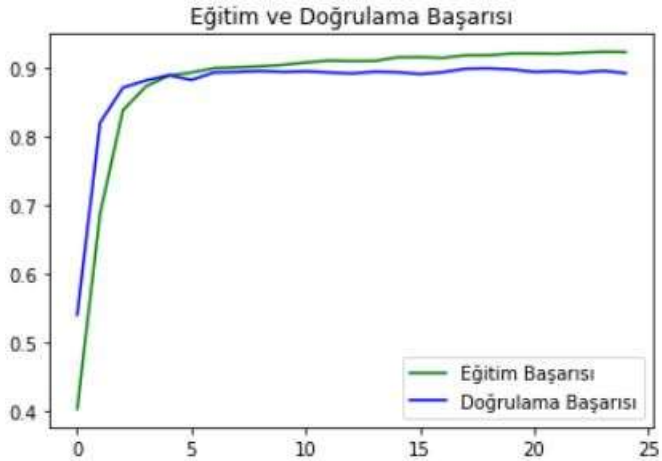
Model 3

Üçüncü modelde Embedding katmanının oluşturulmasının arından 1 boyutlu SpatialDropout katmanı oluşturulmuş ve 0,4'lük seyreltme uygulanmıştır. Daha sonra LSTM katmanı yığın (stack) olarak oluşturulmuştur. Birinci LSTM 256, ikinci 128 katmandan oluşmaktadır. Her LSTM katmanından sonra 0,2'lik seyreltme (dropout) uygulanmış ardından dense katmanı oluşturulmuştur. Bu işlem sonunda 522 566 adet eğitim parametresi elde edilmiştir. Oluşturulan model Şekil 4.36'da sunulmuştur.

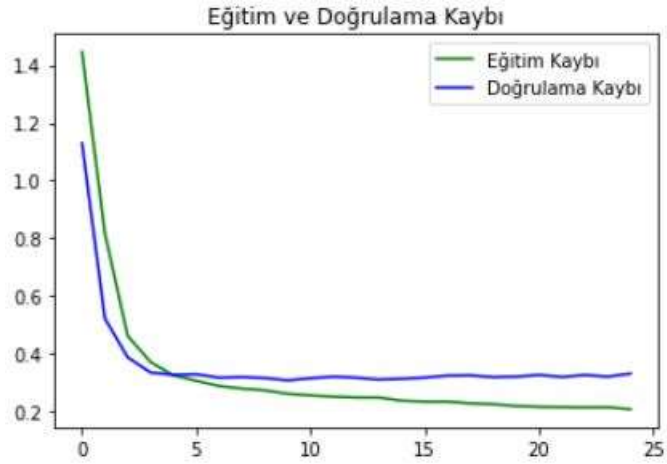



Şekil 4.36. Siber tehdit istihbaratı türü sınıflandırması model 3

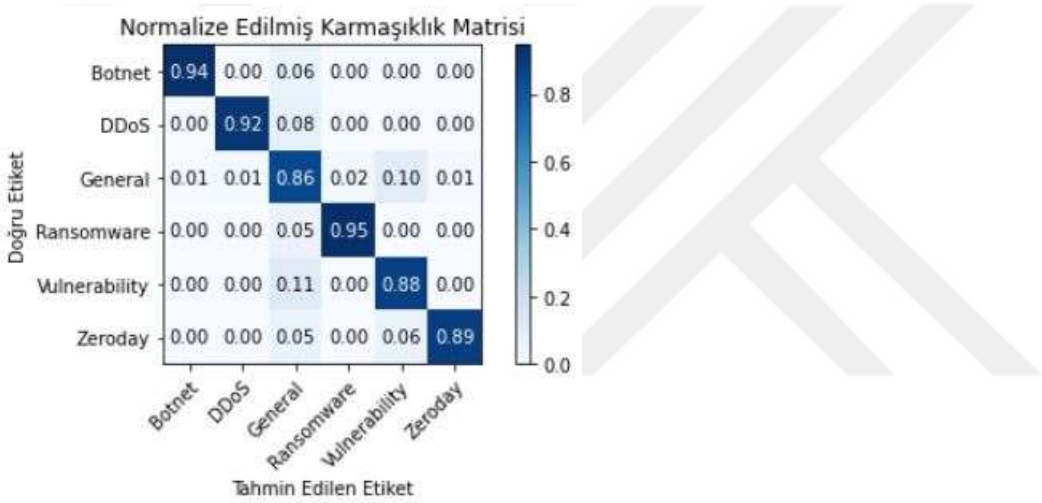
Bu model ile yapılan eğitim sonucunda %88,95 başarı elde edilmiştir. Modelin eğitimi sonucunda elde edilen başarı grafiği Şekil 4.37'de kayıp grafiği Şekil 4.38'de, Karmaşıklık Matrisi Şekil 4.39'da sunulmuştur.




Şekil 4.37. Siber tehdit istihbaratı türü sınıflandırması model 3 başarı grafiği



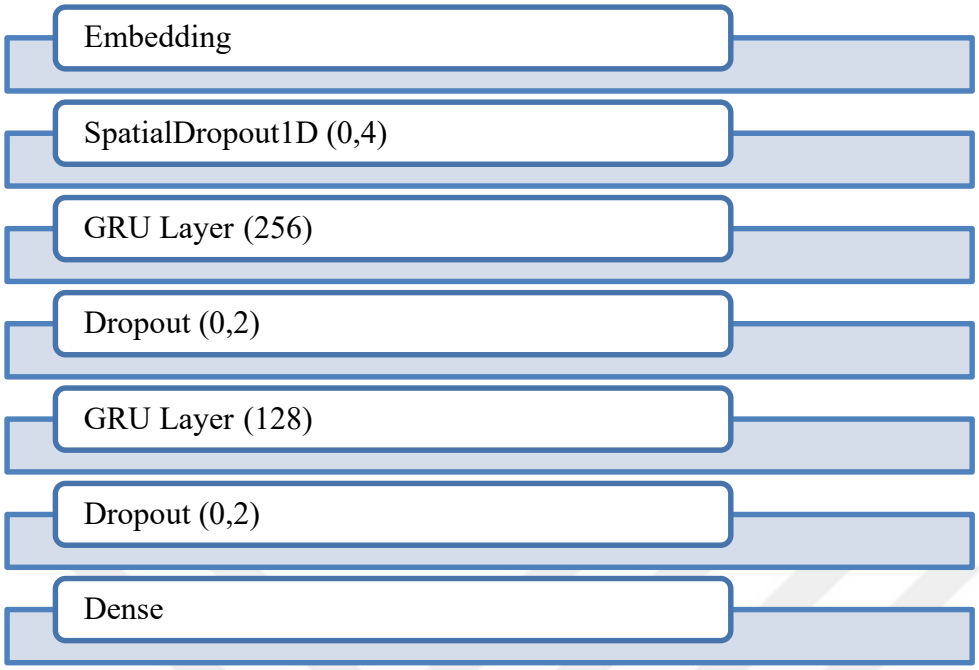

Şekil 4.38. Siber tehdit istihbaratı türü sınıflandırması model 3 kayıp grafiği




Şekil 4.39. Siber tehdit istihbaratı türü sınıflandırması model 3 karmaşıklık matrisi

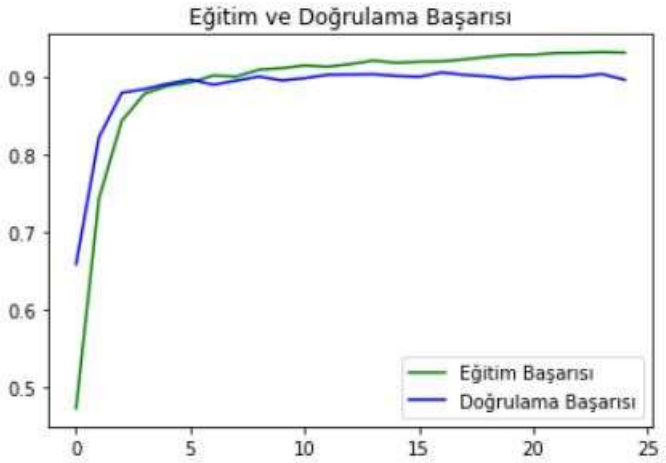
Model 4

Dördüncü modelde Embedding katmanının oluşturulmasının arından 1 boyutlu SpatialDropout katmanı oluşturulmuş ve 0,4'lük seyreltme uygulanmıştır. Daha sonra GRU katmanı yığın (stack) olarak oluşturulmuştur. Birinci GRU 256, ikinci 128, katmandan oluşmaktadır. Her GRU katmanından sonra 0,2'lik seyreltme (dropout) uygulanmış ardından dense katmanı oluşturulmuştur. Bu işlem sonunda 423 042 adet eğitim parametresi elde edilmiştir. Oluşturulan model Şekil 4.40'da sunulmuştur.

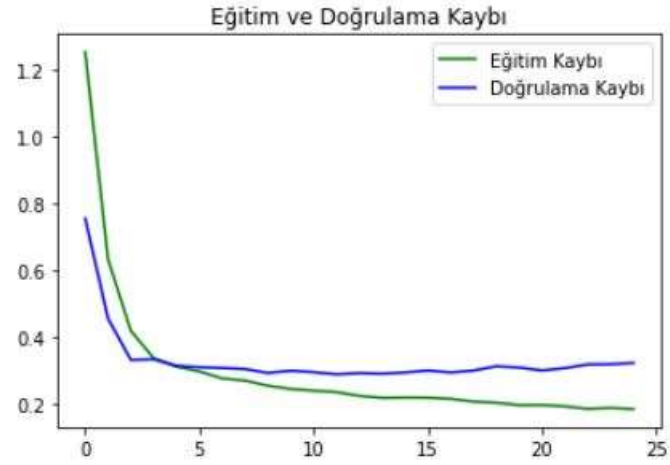



Şekil 4.40. Siber tehdit istihbaratı türü sınıflandırması model 4

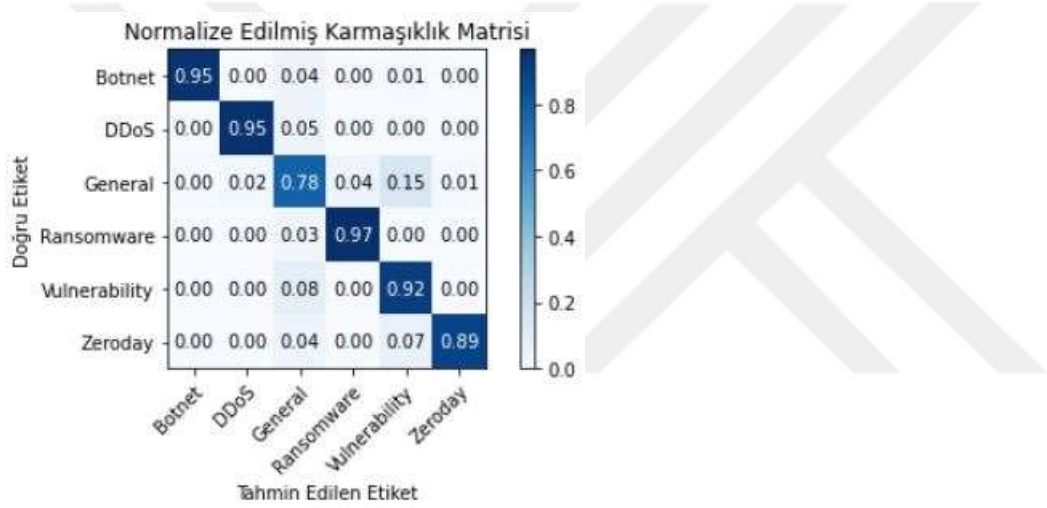
Bu model ile yapılan eğitim sonucunda %88,08 başarı elde edilmiştir. Modelin eğitimi sonucunda elde edilen başarı grafiği Şekil 4.41'de kayıp grafiği Şekil 4.42'de, Karmaşıklık Matrisi Şekil 4.43'te sunulmuştur.




Şekil 4.41. Siber tehdit istihbaratı türü sınıflandırması model 4 başarı grafiği



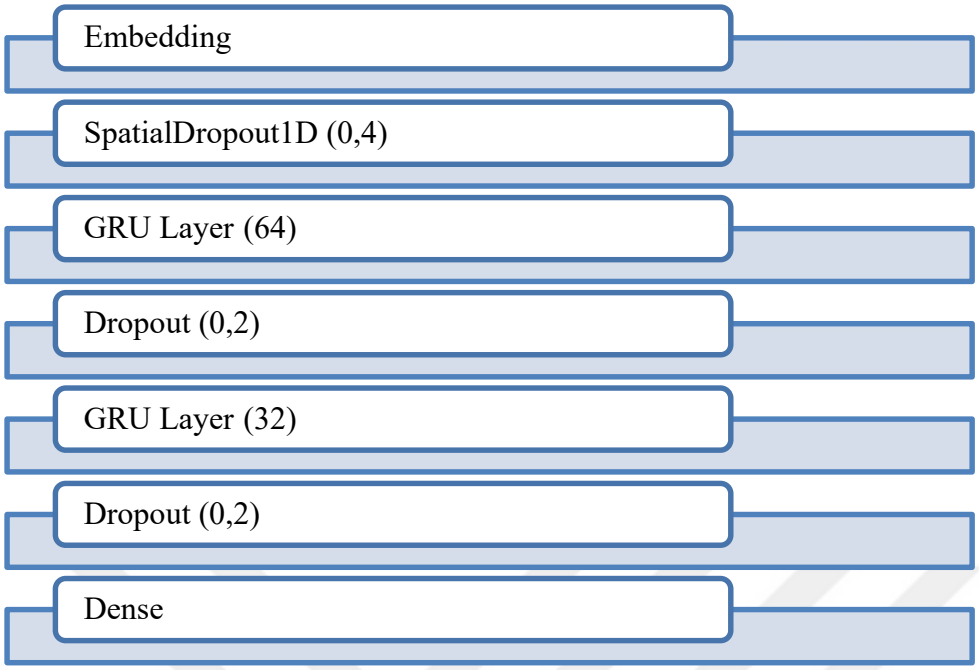

Şekil 4.42. Siber tehdit istihbaratı türü sınıflandırması model 4 kayıp grafiği




Şekil 4.43. Siber tehdit istihbaratı türü sınıflandırması model 4 karmaşıklık matrisi

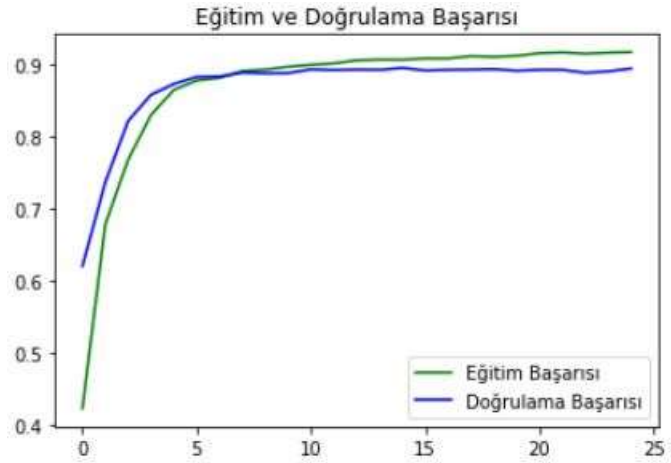
Model 5

Beşinci modelde Embedding katmanının oluşturulmasının arından 1 boyutlu SpatialDropout katmanı oluşturulmuş, 0,4'lük seyreltme uygulanmış ve GRU yığınında yer alan katman sayısı azaltılmıştır. Birinci GRU 64, ikinci 32 katmandan oluşmaktadır. Her GRU katmanından sonra 0,2'lik seyreltme (dropout) uygulanmış ardından dense katmanı oluşturulmuştur. Bu işlem sonunda 69 378 adet eğitim parametresi elde edilmiştir. Oluşturulan model Şekil 4.44'te sunulmuştur.

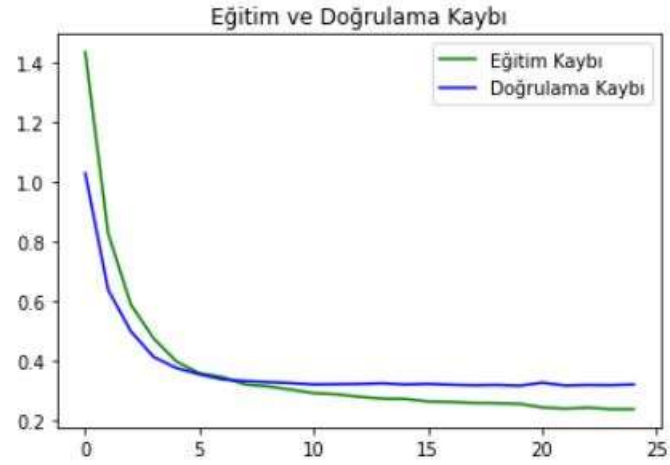



Şekil 4.44. Siber tehdit istihbaratı türü sınıflandırması model 5

Bu model ile yapılan eğitim sonucunda %88,74 başarı elde edilmiştir. Modelin eğitimi sonucunda elde edilen başarı grafiği Şekil 4.45'te kayıp grafiği Şekil 4.46'da, Karmaşıklık Matrisi Şekil 4.47'de sunulmuştur.




Şekil 4.45. Siber tehdit istihbaratı türü sınıflandırması model 5 başarı grafiği

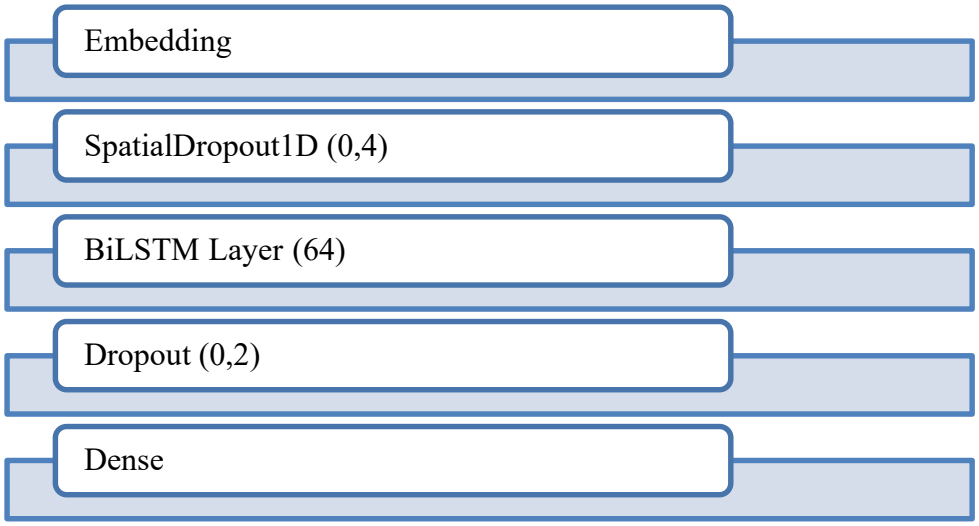



Şekil 4.46. Siber tehdit istihbaratı türü sınıflandırması model 5 kayıp grafiği


Şekil 4.47. Siber tehdit istihbaratı türü sınıflandırması model 5 karmaşıklık matrisi

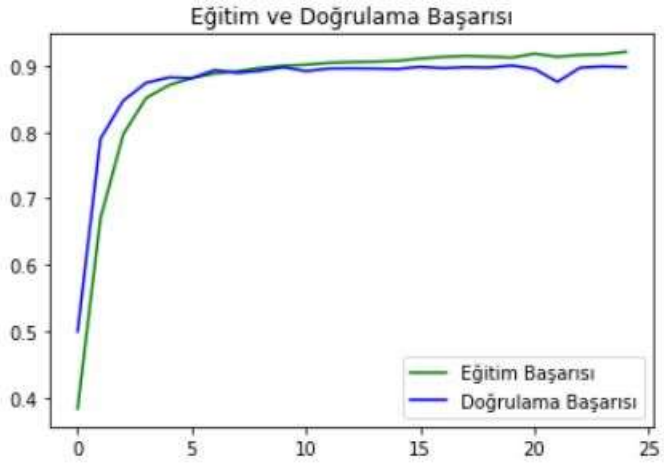
Model 6

Altınca modelde Embedding katmanı oluşturulmuş, 1 boyutlu SpatialDropout katmanı oluşturulmasının ardından 0,4'lük seyretme uygulanmış, 64 katmanlı BiLSTM katmanı oluşturulmuş ve 0,2'lik seyreltme (dropout) uygulanmış ardından dense katmanı oluşturulmuştur. Bu işlem sonunda 91 154 adet eğitim parametresi elde edilmiştir. Oluşturulan model Şekil 4.48'de sunulmuştur.

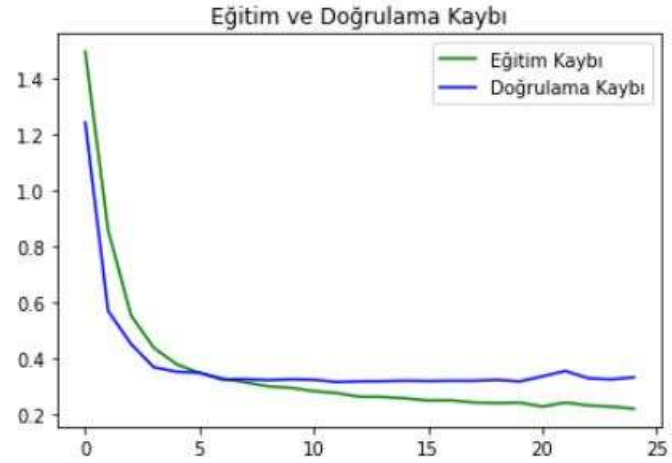



Şekil 4.48. Siber tehdit istihbaratı türü sınıflandırması model 6

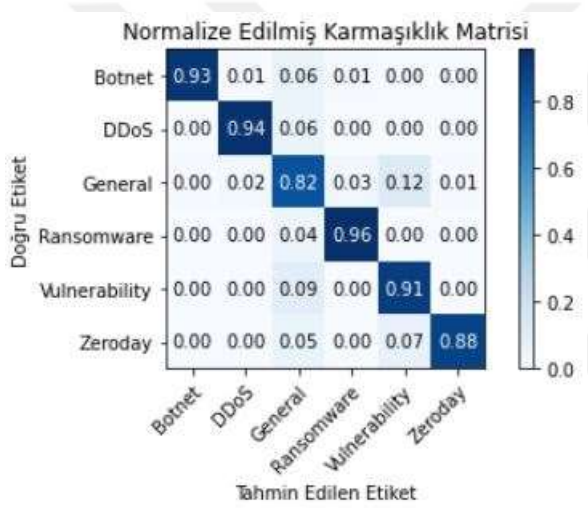
Bu model ile yapılan eğitim sonucunda %88,90 başarı elde edilmiştir. Modelin eğitimi sonucunda elde edilen başarı grafiği Şekil 4.49'da kayıp grafiği Şekil 4.50'de, Karmaşıklık Matrisi Şekil 4.51'de sunulmuştur.




Şekil 4.49. Siber tehdit istihbaratı türü sınıflandırması model 6 başarı grafiği




Şekil 4.50. Siber tehdit istihbaratı türü sınıflandırması model 6 kayıp grafiği




Şekil 4.51. Siber tehdit istihbaratı türü sınıflandırması model 6 karmaşıklık matrisi

Çalışmada kullanılan veri setinde yer alan verilerin Siber Tehdit İstihbaratı türü sınıflandırmasında eğitimi yapılan altı model arasında birinci modelin en başarılı model olduğu görülmüştür. Eğitilen modellere ait başarı tablosu Çizelge 4.5'te sunulmuştur.

Çizelge 4.5 Siber istihbarat türü sınıflandırma başarı tablosu

| Model | Başarı Oranı |
|---|---|
| Model 1 | 89,49 |
| Model 2 | 88,86 |
| Model 3 | 88,95 |
| Model 4 | 88,08 |
| Model 5 | 88,74 |
| Model 6 | 88,90 |

# 5. SONUÇ

Bu çalışmada açık kaynaklardan siber tehdit istihbaratı elde edilmesini amaçlayan bir çalışma yapılmıştır. Çalışmada açık kaynak olarak Twitter'dan faydalanılmıştır. 500 milyon kullanıcı sayısı günlük 2 milyona yakın tweet atılması hem siber tehdit aktörlerini hem de ülkelerin SOME kuruluşları ve siber güvenlik firmaları gibi siber savunma aktörlerinin aynı platformda buluşturması sebebiyle Twitter oldukça zengin bir veri kaynağı durumundadır.

Çalışmada açık kaynak bir veri setinden yararlanılmıştır. Doğal dil işleme ve derin öğrenme metotları kullanılarak yapısal olmayan twitter verisi sınıflandırılmıştır. Çalışma iki safhada gerçekleştirilmiş, ilk olarak veri setinde yer alan tweetlerin siber tehdit istihbaratı ile ilgili olup olmadığı sınıflandırmak üzere 6 model ayrı ayrı eğitilmiş ve en başarılı model %88,64 başarı elde etmiştir. İkinci aşamada ise siber tehdit istihbaratının türü sınıflandırması yapılmış bu çalışmada yine 6 ayrı model eğitilmiş ve en başarılı model %89,49 başarı elde etmiştir.

Aynı veri seti ile yapılan diğer çalışmalar ile karşılaştırıldığında siber tehdit istihbaratı türü sınıflandırılmasında diğerlerine oranla daha başarılı olduğu görülmektedir. Siber tehdit istihbaratı verisi/değil sınıflandırmasında ise veri setini paylaşan yazarın makalesinde belirtmiş olduğu yöntemler uygulansa da aynı başarı oranı elde edilememiştir.

Çizelge 5.1 Aynı veri seti ile yapılan çalışmalar

| S.No. | Çalışma Adı | Başarı Oranı (Yüzde) | |
|---|---|---|---|
| | | Siber Tehdit İstihbaratı Verisi/Değil | Siber Tehdit İstihbaratı Türü |
| 1 | Corpus and Deep Learning Classifier for Collection of Cyber Threat Indicators in Twitter Stream | 94,72 | 87,56 |
| 2 | Deep Learning Approach for Enhanced Cyber Threat Indicators in Twitter Stream | 85,8 | 89,3 |
| 3 | Açık Kaynaklardan Siber Tehdit İstihbarat Verisi Elde Edilmesi | 88,64 | 89,49 |

Çalışma sınırlılıkları;

- Çalışmada faydalanılan veri setinde 21 368 adet veri bulunmakta olup derin öğrenme için az sayılabilecek miktardadır.
- Veri setine yeni veri eklenmesi amacıyla "twitter developer" hesabı başvurusu yapılmış olsa da Twitter'ın güncellenen politikaları sebebiyle başvuru onaylanmamış bu sebeple veri setine daha fazla veri girişi yapılamamıştır.
- Veri setinde bulunan "Leak" sınıfında yalnızca 136 adet veri bulunması sebebiyle başarılı sınıflandırma yapılamamış diğer sınıflarında başarı oranını olumsuz etkilemesi sebebiyle "Leak" sınıfı veri setinden çıkarılmıştır.

Sonuç olarak bu çalışmada kamu kurumları, şirketler ve bireyler tarafından kullanılabilecek ve siber tehdit istihbaratını kolaylıkla elde edilebilecek bir sistem önerilmiştir. Böylece siber güvenliği tesis edilmesinde önemli bir role sahip olan siber tehdit istihbaratının açık kaynaklardan özellikle de Twitter'dan derin öğrenme yöntemleri ile elde edilebileceği, elde edilen istihbaratının türünün de tespit edilebileceği bu çalışmada ortaya koyulmuştur.

Twitter, Facebook gibi sosyal medya platformlarının kullanıcı sayısı ve bu platformlarda üretilen içerik giderek artmaktadır. Bu sebeple gelecekte yapılacak çalışmalarda bugün yapılan çalışmalardan daha başarılı ve daha zengin veriler elde edilebilecektir.

# KAYNAKLAR


1. İnternet: Kaspersky (2020). The number of new malicious files detected every day increases by 5.2% to 360,000 in 2020. URL: https://www.kaspersky.com/about/press-releases/2020_the-number-of-new-malicious-files-detected-every-day-increases-by-52-to-360000-in-2020, Son Erişim Tarihi: 01.07.2021.

2. İnternet: Sophos (2021). WannaCry aftershock. URL: https://www.sophos.com/enus/medialibrary/PDFs/technical-papers/WannaCry-aftershock.pdf, Son Erişim Tarihi: 01.07.2021.

3. Sapienza, A., Ernala, S.K., Bessi, A., Lerman, K. ve Ferrara, E. (2018). *DISCOVER: Mining online chatter for emerging cyber threats*. Companion Proceedings of the The Web Conference 2018*,* Lyon, France, 983-990.

4. Edouard, A. (2017). *Event detection and analysis on short text messages.* Doktora Tezi. Université Côte d'Azur, Information and Communication Sciences and Technologies Institute, France, 92-99.

5. İnternet: Malwarebytes (2021). Let's talk emotet malware. URL: https://www.malwarebytes.com/emotet/, Son Erişim Tarihi: 01.08.2021.

6. İnternet: Brandwatch (2021). 60 Incredible and interesting twitter stats and statistics. URL: https://www.brandwatch.com/blog/twitter-statsand-statistics/, Son Erişim Tarihi: 18.08.2021.

7. Liew, S.W., Sani, N.F.M., Abdullah, M.F., Yaakob, R. ve Sharum, M.Y. (2019). An effective security alert mechanism for real-time phishing tweet detection on Twitter. *Computers & security*, 83, 201–207.

8. Zenebe, A., Shumba, M, Carillo, A. ve Cuenca, S. (2019). Cyber threat discovery from dark web*. EPiC Series in Computing*, 64, 174-183.

9. Rodriguez, A. ve Okamura, K. (2020). Cybersecurity text data classification and optimization for CTI systems. *Advances in Intelligent Systems and Computing,* 1150, 410-419.

10. Subroto, A. ve Apriyana, A. (2019). Cyber risk prediction through social media big data analytics and statistical machine learning. *Journal of Big Data*, 6 (50), 100-119.

11. Koloveas, P., Chantzios, T., Tryfonopoulos, C. ve Skiadopoulo S. (2019). *A crawler architecture for harvesting the clear, social, and dark web for IoT-related cyber-threat intelligence*. 2019 IEEE World Congress on Services, Milan, Italy, 3-8.

12. Erkal, Y., Sezgin, M. ve Gündüz, S. (2015). *A new cyber security alert system for twitter*. 2015 IEEE 14th International Conference on Machine Learning and Applications (ICMLA), Miami, FL, USA, 766-770.

13. Bose, A., Behzadan, V., Aguirre, C. ve Hsu, W.H. (2019). *A novel approach for detection and ranking of trendy and emerging cyber threat events in twitter streams*. ASONAM '19: Proceedings of the 2019 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining, New York, USA, 871–878.

14. Ghazi, Y., Anwar, Z., Mumtaz, R., Saleem, S. ve Tahir, A. (2018). *A supervised machine learning based approach for automatically extracting high-level threat intelligence from unstructured sources.* 2018 International Conference on Frontiers of Information Technology (FIT), Islamabad, Pakistan, 129-134.

15. Aslan, Ç.B., Sağlam, R.B. ve Li, S. (2018). *Automatic detection of cyber security related accounts on online social networks: twitter as an example*. SMSociety '18: Proceedings of the 9th International Conference on Social Media and Society, Copenhagen, Denmark, 236-240.

16. Deliu, I., Leichter, C. ve Franke, K. (2018*). Collecting cyber threat intelligence from hacker forums via a two-stage, hybrid process using support vector machines and latent dirichlet allocation*. 2018 IEEE International Conference on Big Data (Big Data), Seattle, WA, USA, 5008-5013.

17. Dionísio, N., Alves, F., Ferreira, P.M. ve Bessani, A. (2019). *Cyberthreat detection from twitter using deep neural networks.* IJCNN 2019. International Joint Conference on Neural Networks, Budapest, Hungary, 1-8.

18. Behzadan, V., Aguirre, C., Bose, A. ve Hsu, W. (2018). *Corpus and deep learning classifier for collection of cyber threat indicators in twitter stream*. 2018 IEEE International Conference on Big Data (Big Data), Seattle, WA, USA, 5002-5007.

19. Mittal, S., Das, P.K., Mulwady, V., Joshi, A. ve Finin, T. (2016). *CyberTwitter: Using twitter to generate alerts for cybersecurity threats and vulnerabilities*. 2016 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM), San Francisco, CA, USA, 860-867.

20. Wang, Z. ve Zhang, Y. (2017). *DDoS Event forecasting using twitter data.* Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence (IJCAI-17), Melbourne, Australia, 4151-4157.

21. Simran, K., Balakrishna, P., Vinayakumar, R. ve Soman, K.P. (2019). *Deep learning approach for enhanced cyber threat indicators in twitter stream*. The 7th International Symposium on Security in Computing and Communications (SSCC'19), Trivandrum, India, 512-523.

22. Singh, M., Bansal, D. ve Sofat, S. (2014). *Detecting malicious users in twitter using classifiers.* SIN '14: Proceedings of the 7th International Conference on Security of Information and Networks, Glasgow, Scotland, 247–253.

23. Deliu, I., Leichter, C. ve Franke, K. (2017). *Extracting cyber threat intelligence from hacker forums: support vector machines versus convolutional neural networks.* 2017 IEEE International Conference on Big Data (BIGDATA), Boston, MA, USA, 3648-3656.

24. Rao, P., Kamhoua, C., Njilla, L. ve Kwiat, K. (2018). Methods to detect cyberthreats on twitter. *Surveillance in Action*, 1, 333-350.

25. Le, B.D., Wang, G., Nasim, M. ve Babar, M.A. (2019). *Gathering cyber threat intelligence from twitter using novelty classification.* 2019 International Conference on Cyberworlds (CW), Kyoto, Japan, 316-323.

26. Lee, S. ve Shon, T. (2016). *Open source intelligence base cyber threat inspection framework for critical infrastructures*. 2016 Future Technologies Conference (FTC),| San Francisco, CA, USA, 1030-1033.

27. İntermet: Defense of Department (2021). DOD leaders share their intelligence threat assessments. URL: https://www.defense.gov/Explore/News/Article/Article/2655574/dod-leaders-share-their-intelligence-threat-assessments/, Son Erişim Tarihi: 12.07.2021.

28. İnternet: Wikipedia (2021). Open-source intelligence. URL: https://en.wikipedia.org/wiki/Open-source_intelligence, Son Erişim Tarihi: 12.07.2021.

29. van de Schoot, R., de Bruin, J., ve Schram, R. (2021). An open source machine learning framework for efficient and transparent systematic reviews. *Nat Mach Intell 3*, 125–133.

30. Glassman, M. ve Kang, M.J. (2012). Intelligence in the internet age: The emergence and evolution of open source intelligence (OSINT). *Computers in Human Behavior.* 28(2), 763-682.

31. Hayes, D.R. ve Cappa, F. (2018). Open-source intelligence for risk assessment. *Business Horizons,* 61(5), 689-697.

32. Koops, B.J., Hoepman, J.H. ve Leenes R., (2013). Open-source intelligence and privacy by design. *Computer Law & Security Review,* 29(6), 676-688.

33. Best, C. (2011). *Challenges in open source intelligence*. 2011 European Intelligence and Security Informatics Conference, Washington, USA, 58-62.

34. Mulder, W.D., Bethard, S. ve Moens, M.F. (2014). A survey on the application of recurrent neural networks to statistical language modeling. *Computer Speech & Language*, 30(1), 17-18.

35. Lee, S. ve Shon, T. (2016). *Open source intelligence base cyber threat inspection framework for critical infrastructures.* 2016 Future Technologies Conference (FTC), San Francisco, USA, 1030-1033.

36. Hochreiter, S. ve Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, 9(8), 1735-80.

37. Landi, F., Baraldi, L., Cucchiara, R., ve Cornia, M. (2021). Working memory connections for LSTM. *Neural Networks,* 144, 334-341.

38. İnternet: Colah (2015). Understanding LSTM networks. URL: https://colah.github.io/posts/2015-08-Understanding-LSTMs/, Son Erişim Tarihi: 15.07.2021.

39. Cura, A. (2019). *Driver profiling using long short term memory (LSTM) and convolutional neural network (CNN) methods*. Yüksek Lisans Tezi, Marmara Üniversitesi Fen Bilimleri Enstitüsü, İstanbul, 16-40.

40. İnternet: Jordan, J. (2018). Setting the learning rate of your neural network. URL: https://www.jeremyjordan.me/nn-learning-rate/, Son Erişim Tarihi: 17.07.2021.

41. İnternet: Karpathy, A. (2021). CS231n: Convolutional neural networks for visual recognition. URL: http://cs231n.github.io/, Son Erişim Tarihi: 22.07.2021.

42. İnternet: McLaughlin, M. (2012). Using open source intelligence for cybersecurity intelligence. URL: https://www.computerweekly.com/tip/Using-open-source-intelligence-software-for-cybersecurity-intelligence, Son Erişim Tarihi: 12.07.2021.

43. Kingma, D.P. ve Ba, J. (2015). *Adam: A method for stochastic optimization*. 3rd International Conference for Learning Representations, San Diego, USA, 13.

44. Geron, A. (2017). Hands-On machine learning with scikit-learn and tensorflow (First Edition). California: O'Reilly, 92-105.

45. İnternet: Tieleman, T. (2021). Neural networks for machine learning. URL: http://www.cs.toronto.edu/~tijmen/csc321/, Son Erişim Tarihi: 17.07.2021.

46. İnternet: Moujahid, A. (2016). A practical introduction to deep learning with caffe and python. URL: http://adilmoujahid.com/posts/2016/06/introduction-deep-learning-python-caffe/, Son Erişim Tarihi: 17.07.2021.

47. İnternet: Avendi, M. (2021). Another look into overfitting. URL: https://medium.com/randomai/another-look-into-over-fitting-33e15b044a5e, Son Erişim Tarihi: 17.07.2021.

48. Bejani, M.M. ve Ghatee, M. A. (2021). Systematic review on overfitting control in shallow and deep neural networks. *Artificial Intelligence Review*, 54, 6391–6438.

49. İnternet: Tensorflow (2021). Explore overfitting and underfitting. URL: https://www.tensorflow.org/tutorials/keras/overfit_and_underfit, Son Erişim Tarihi: 18.07.2021.

50. Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I. ve Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research,* 15, 1929-1958.

51. İnternet: Keras (2021). Keras: the python deep learning library. URL: https://keras.io/, Son Erişim Tarihi: 18.07.2021.

52. Zhang, H., Zhang, L. ve Jiang, Y. (2019). *Overfitting and underfitting analysis for deep learning based end-to-end communication systems.* 11th International Conference on Wireless Communications and Signal Processing (WCSP), Xi'an, China, 1-6.

53. Lierman, V. ve Li, S. (2021). Methods of machine learning. *The Digital Journey of Banking and Insurance*, 3, 225-238.

54. İnternet: Twepy (2021). Tweepy documentation. URL: https://docs.tweepy.org/en/stable/, Son Erişim Tarihi: 18.07.2021.

*GAZİ GELECEKTİR...*