Avrupa Bilim ve Teknoloji Dergisi
Sayı 27, S. 792-801, Kasım 2021

Araştırma Makalesi

www.ejosat.com ISSN:2148-2683

European Journal of Science and Technology
No. 27, pp. 792-801, November 2021

Research Article

Kötücül Web Sayfalarının Tespitinde Doc2Vec Modeli ve Makine Öğrenmesi Yaklaşımı

Recep Sinan Arslan[1*]

[1*] Kayseri Üniversitesi, Mühendislik Mimarlık ve Tasarım Fakültesi, Bilgisayar Mühendisliği Bölümü, Kayseri, Türkiye, (ORCID: 0000-0002-3028-0416), sinanarslanemail@gmail.com

(İlk Geliş Tarihi 11 Ağustos 2021 ve Kabul Tarihi 6 Ekim 2021)

(**DOI:** 10.31590/ejosat.981450)

ATIF/REFERENCE: Arslan, R.S. (2021). Kötücül Web Sayfalarının Tespitinde Doc2Vec Model ve Makine Öğrenmesi Yaklaşımı. *Avrupa Bilim ve Teknoloji Dergisi*, (27), 792-801.

Öz

Günümüzde birçok işlem dijital ortama taşınmakta ve verilerimizi bu ortamda korumak zorlaşmaktadır. Birçok cihazın internete bağlı olması web güvenliği için büyük bir sorun olmaktadır. İnternet kaynaklı saldırıları başlatmanın en yaygın yolu da kötü amaçlı URL adreslerini kullanmaktır. Kötücül faaliyette bulunan korsanlar bu amaçla hazırladıkları web sitelerini kullanarak birçok veriyi elde etmektedirler. Bu tür kötü amaçlı URL adreslerini veya web sitelerini tespit etmenin geleneksel yolu kara liste kullanmaktır. Ancak bu yöntem yeni oluşturulan kötü amaçlı URL'lerin tespit edilmesinde başarılı olmamaktadır. Bu çalışmada, kötücül URL adreslerinin tespitinde verimliliği artırmak ve kara liste gibi bir takım veri tabanlarına bağımlılığı önlemek için makine öğrenmesi kullanan bir yaklaşım önerildi. Makine öğreniminde sınıflandırma için farklı algoritmalar denenirken, özellik çıkarımı için Doc2Vec yaklaşımı kullanılmıştır. Sadece URL adreslerinden elde edilen özellikler kullanılarak sınıflandırma yapılmaktadır. ISCX2016URL veri seti ile yapılan testlerin birinci aşamasında URL adresinin kötücül ve iyicil olarak sınıflandırma için Logistic Regresyon algoritması ile %99,2 doğruluk yakalanırken, kesinlik, duyarlılık ve F-skoru değerlerinde sırasıyla %98,9, %99,1 ve %99,2 değerleri yakalanmıştır. Testlerin ikinci aşamasında ise kötücül URL adreslerinin spam, kimlik avı, kötücül amaçlı yazılım dağıtan ve tahrif edilmiş sınıflarına aitlikleri test edilmiştir. Sonuçta SVC sınıflandırıcı ile %88,1 doğruluk ile kötücül URL adresleri sınıflandırılmıştır. Sonuçta ortaya çıkan modeli herhangi bir vekil sunucuda veya bir ağ denetleyici platforma üzerinde uygulamak mümkündür.

Anahtar Kelimeler: Tekdüzen kaynak bulucu(URL), Doc2Vec, Web güvenliği, Makine öğrenmesi, URL filtreleme.

A Detection Method for Malicious Web Pages using Doc2vec Model and Machine Learning

Abstract

Today, many transactions are transferred to the digital environment and it is difficult to protect our data in this environment. Due to the fact that many things are connected to the internet, web security is emerging as a big problem. The most common way to initiate Internet-borne attacks is using malicious URL addresses. Hackers engaged in malicious activity obtain a lot of data with using the websites they have prepared for this purpose. The traditional way to detect such malicious URL addresses or websites is using a blacklist. However, this method does not succeed in detecting newly created malicious URLs. In this study, an approach using machine learning is proposed to increase efficiency in detecting malicious URLs and prevent dependence on some databases such as blacklists. While different machine learning algorithms is used for classification, Doc2Vec approach is used for feature extraction. Classification is made using only the features obtained from URL addresses. In the first stage of the tests conducted with the ISCX2016URL data set, URLs are classified as malicious or benign. With the Logistic Regression algorithm, 99.2% accuracy was achieved, while the precision, recall and F-score values were 98.9%, 99.1% and 99.2%, respectively. In the second stage of the tests, the malicious URLs belonging to the classes spam, phishing, malware and defacement are tested. Malicious URLs are classified with SVC with 88.1% accuracy. It is possible to implement the resulting model on any Proxy server or on a network controller platform.

Keywords: Uniform resource locators (URLs), Doc2Vec, Web security, Machine learning, URL filtering.

[*] Sorumlu Yazar: sinanarslanemail@gmail.com

1. Giriş

Web tabanlı hizmetlerin yaygınlaşması ile birlikte, bulut veya her yerden erişilebilir sistemler için web uygulamaları oluşturulmaktadır ve klasik tipteki masaüstü uygulamalarından web uygulamalarına geçiş oldukça hızlı olmuştur. İnsanların sosyalleşme, bilgi arama, paylaşma, forum, e-ticaret hizmetleri gibi birçok günlük aktivitelerde web uygulamalarına bağımlılıkları artmıştır. Amaç herkes için daha kolay kullanılabilir sistemler yaratmaktır. Kullanıcı isteklerine göre dinamik yanıtlar üretme aşamasına gelindiğinde güvenlik konusunda problemli sayfalar kişiler için bilgi güvenliğinde tehlikeyi beraberinde getirmektedir. Web tasarımcıları sınırlı güvenlik bilgileri ve sınırlı ve suni test ortamları ile uygulamalar geliştirmektedirler ve sonuçta ortaya çıkan web sayfalarında birçok potansiyel tehdit oluşabilmektedir. Kötü amaçlı saldırganlarda bu potansiyeli değerlendirip, güvenlik açıklarından yararlanırlar. Kötü amaçlı web siteleri kullanıcıları kötücül bir yazılımı bilgisayara indirmeleri için yönlendirme, gizli bilgilerini kaybetmesine neden olma, daha büyük saldırılar için bir nokta olarak kullanma gibi farklı durumlara neden olabilirler. Bu sebeplerle web saldırıları yaygındır ve bu web sitelerinin tespit edilmesi ve sınıflandırılması beklenmektedir(Chia-Mei ve ark., 2015).

Web sayfalarının sınıflandırılması, ilgili web sayfasının özelliklerinin analiz edilmesi sonrasında bir veya birden fazla kategoriye otomatik olarak atanmasını ifade etmektedir. Otomatik web sayfası sınıflandırması için karar destek sistemi olarak hizmet veren veya başka süreçler ile entegre çalışabilecek birçok uygulama bulunmaktadır. Bir web sayfasının bir sorguyla ilgili bilgileri içerip içermediğine dair istihbarat sağlamak ile ilgili entegre sanal tarayıcı tasarımları, bir web sitesine ait özniteliklerin çıkarılması için önerilmiş yaklaşımlar, reklam gibi belirli türdeki içeriklerden kaçınmak için web sitesi filtreleri, ebeveyn kontrol sistemleri, yinelenen URL'leri tespit etmek ve kanonikleştirmek, web dizinleri oluşturmak, sürdürmek veya genişletme çalışmaları ile belirli konudaki web sayfalarını bulmak üzere oluşturulmuş tarayıcı tasarlamak gibi olabilmektedir(Imma ve ark., 2016).

Literatürde web sayfalarını sınıflandırmak için çeşitli yöntemler önerilmiştir. Bu yöntemler, terime dayalı araçlar(Jasper ve ark., 2019; Floria ve ark., 2002), yapı temelli araçlar (Gideon ve ark., 2021), görsel temelli araçlar (Daniel ve ark, 2019; Ali ve ark., 2011), bağlantı tabanlı araçlar (Jia ve ark., 2016) ve URL tabanlı araçlar (Rajalakshmi ve ark., 2017; Hidayet ve ark., 2007; Rajalakshmi ve ark., 2020; Özgür ve ark., 2019)'dır. Terime dayalı, yapı temelli ve görsel temelli yaklaşımlar web sayfalarının içeriğine bağlı özelliklere dayanmaktadırlar. Bu yöntemlerin çalıştırılabilmesi için web sayfalarının tamamen indirilmesi ve işlenmesi gerekmektedir. Dünya üzerinde milyonlarca web sayfasının bulunması, her bir analiz için bunlarının tümümün sunucuya kaydedilmesi ve ilgili sayfaların her bir analizde filtrelenmesinin gerekmesi ve tüm bu işlemler için büyük bir bant genişliğine ihtiyaç duyulması, bu yöntemlerin web sayfalarını sınıflandırmak için daha az ilgi çekici olmasına neden olmaktadır. Bağlantı tabanlı araçlar, bir web sitesinin kendi sayfaları arasındaki bağlantıları analiz edilerek belirli bir grafik üretilip web sayfasının sınıflandırılması yaklaşımını kullanmaktadır. Öğrenme aşamasında web sitesinde kapsamlı bir tarama gerçekleştirilmektedir. Bu durum, bağlantı tabanlı araçları gerçek dünyada web sayfalarının analizinde daha yaygın olarak kullanılan bir yöntem haline getirmektedir. Birçok çalışmada URL tabanlı sistem geliştirmeye yönelik araştırmalar yapılmaktadır. URL, tüm çevrimiçi etkinlerin altyapısıdır ve kötü amaçlı URL'leri tespit etmek genellikle kötücül olanların iyicil olanlardan ayırmaya yönelik bir sınıflandırma problemidir (Tie ve ark., 2020). Sürekli veri toplama, özellik çıkarma, veri ön işleme ve sınıflandırma gibi çok daha karmaşık sistematik görevleri içermektedir.

Çoğu ticari anti virüs yazılımı veya açık kaynaklı çözüm (Netcraft, 2018; Navisite, 2021), kötü amaçlı veya kimlik avı web sitelerini tespit etmek için geniş URL veri tabanlarını veya kara listelerini kullanmaktadırlar (Chia-Mei ve ark., 2015). Kara liste, kötü niyetli web sayfalarıyla başa çıkmak için kullanılan basit ve belirli seviyede doğruluk sağlayan tipik bir yaklaşımdır. Bu teknik yalnızca listeler zamanında güncellendiğinde ve kötü amaçlı web sayfalarını bulmak için web siteleri yoğun bir şekilde ziyaret edildiğinde etkilidir. Bu yöntem çevrimiçi kullanıcıların zamanında korunmasını sağlamak için yetersiz kalmaktadır (Momammad ve ark., 2016). Bunun yanında içerisinde ip adresleri ve URL bilgileri barındıran kara listeler, pahalı ve karmaşık filtreleme teknolojileri yardımıyla çıkarıldığından, şirketler tarafından güncellenmiş halleri ücretsiz olarak satılmazlar. Dahası, web sayfalarına uygulanan adresleri gizleme veya URL ve ip adresi değiştirme gibi teknikler kötücül adreslerin kara liste ile kontrol edilmeleri olasılığını düşürmektedir. Ayrıca, güvenilir web sayfaları hem kötü amaçlı hem de yasal web sayfalarını içeren ve sahte URL olarak bilinen gizli URL kullanabilmektedirler. Kötü amaçlı URL tespit araçlarının, gerçek zamanlı olarak çalışabilmeleri, yüksek doğrulukla tespit yapabilmeleri ve kimlik avı gibi özel tipteki saldırı türlerine karşı da etkili olmaları beklenmektedir (Momammad ve ark., 2016).

Bilgisayar sistemlerindeki muazzam miktardaki veri, büyük organize veri kümeleri ve güçlü paralel hesaplama makineleri, gerçek dünya sistemlerindeki makine öğreniminin hızlı bir şekilde benimsenmesine neden olmuştur (Wei ve ark., 2020)). Makine öğrenimi, verilerden model oluşturmaya veya satranç, dama, go gibi oyunlarda olduğu gibi sorunu yöneten bir dizi kuralı kullanmaya olanak tanımaktadır. Modern makine öğrenimi yöntemleri, özellikle elimizde çok fazla veri olduğunda, neredeyse gerçek dünyadaki tüm probleme belirli oranlarda çözüm üretebilmektedir (Arslan ve ark., 2019; Yurttakal ve ark., 2020; Arslan ve ark., 2019). Günümüzde insanlar tarafından okunması ve anlaşılması imkânsız olan büyük verileri analiz etmek büyük bir problemdir. Aynı durum modern suç yöntemleri için de geçerlidir. Hesaplama tekniklerindeki hızlı gelişim, siber saldırıların benzeri görülmemiş ölçekte büyümesine ve yürütülmesine izin vermiştir. Makine öğrenimi teknikleri, kötü amaçlı web sitelerini URL adreslerinden, web içeriğinden veya ağ etkinliğinden çıkarılan özellikler kullanarak sınıflandırma için kullanmaktadırlar. Web içeriğinin analizini benimseyen araçlar, daha fazla hesaplama süresi ve kaynağa ihtiyaç duymaktadırlar. Bu nedenle kötücül web sitesi tespitinde URL tabanlı teknikler tercih edilmektedir. Spam, reklam yazılımı, kimlik avı gibi farklı tür saldırılar için URL'lerin özellikleri farklılık gösterebilmektedir (Momammad ve ark., 2016). Bir web sayfasının sadece URL adresinden elde edilen özelliklere göre sınıflandırılması, web sayfasının tümüyle indirilmeden analiz edilebilmesine imkân tanıdığı için caziptir ve bu performans üzerinde olumlu etkiye sahiptir.

Bu çalışmada, web sayfalarını indirmek zorunda kalmadan sınıflandırmaya izin veren bir yaklaşım önerilmektedir. Önerilen yaklaşımda sınıflandırma modelini oluşturmak için web sayfasında kapsamlı bir tarama yapılması gerekmemektedir. Sadece web sayfalarına ait URL adreslerinden çıkarılan özellikler

kullanılmaktadır. Makine öğrenimi tekniklerini kullanarak kötücül URL'lerin tespiti ve sınıflandırması yapılmıştır. Spam URL'leri, kimlik avı URL'leri, kötü amaçlı yazılım dağıtan web sitelerine ait URL'ler ve tahrif edilmiş URL'ler olmak üzere 4 tür kötücül URL kullanımına bakılmıştır. Herhangi bir sözlük veya kullanıcı girdisine ihtiyaç duyulmamaktadır. Ayrıca önerilen yaklaşım site, dil ve etki alanından bağımsızdır. Bu sebeple hem ölçeklenebilir hem de genel olarak uygulanabilir bir yaklaşımdır. Önerilen model, ISCX-URL2016 veriseti kullanılarak doğrulanmıştır.

Bu çalışmanın URL adreslerinin sınıflandırmasına yönelik olarak katkıları şunlardır:

- Ağ trafiğindeki URL'ler üzerinde metin bazlı segmentasyon ve vektörleştirme yapılmıştır. Web sitelerini, sadece URL adreslerinden sınıflandırmak için Makine öğrenme tekniklerini kullanan yeni bir yaklaşım önerilmiştir.
- Özellik vektörlerini oluşturmak için Doc2Vec ağına göre DM ve DBOW algoritmaları kullanılmıştır.
- Kötü amaçlı URL'lerin tespiti için etkili olan faktörleri değerlendirmek için çok gruplu deneyler gerçekleştirilmiştir.
- Doc2Vec modelleri ile üretilen özellik vektörlerinin başarılı sonuçlar ürettiği doğrulanmıştır.

Bu makalenin geri kalanı şu şekilde düzenlenmiştir:

2. bölümde URL sınıflandırmasında kara liste kullanan, web sitesinin içeriğinin analizine dayalı ve URL adresini kullanarak sınıflandırmaya ilişkin güncel çalışmalardan detaylı olarak bahsedilmiştir. Sonrasında bu çalışmada önerilen modelin metodolojisi; URL ön işlem süreçleri, model tasarımı ve Doc2Vec yapısını içerecek şekilde 3. bölümde verilmiştir. 4. bölümde önerilen modelin test ortamına ilişkin parametreler ve test sonuçları DM, DBOW ve bu iki modelden elde edilen vektörlerin birleştirilerek kullanıldığı hibrit model için ayrı ayrı gösterilmiştir. Çalışmanın tasarımında karşılaşılan kısıtlar ve modelin avantajlı noktaları 5. bölümde anlatılmıştır. Son bölümde, çalışmanın genel bir değerlendirmesi yapıldıktan sonra, model tasarımında karşılaşılan kısıtlar göz önünde bulundurularak gelecekte yapılabilecek çalışmalara ilişkin öneriler açıklanmıştır.

2. Konu ile İlgili Çalışmalar

Kötü amaçlı URL'leri tespit etmek için birçok yaklaşım geliştirilmiştir. Bu yöntemler, kara liste, içerik tabanlı sınıflandırma, URL tabanlı sınıflandırma yaklaşımı olmak üzere 3 farklı türe ayrılabilir.

2.1. Kara Listeler

Kara listeler, genel URL'leri filtrelemek için bilinen kötü amaçlı URL kayıtlarını kullanan yöntemleri ifade etmektedir. Kara liste hizmetleri, sahte web sitelerini tespit etmek için araç çubuklarında, uygulamalarda ve arama motorlarında kapsüllenebilir ve veri tabanları tarafından tutulan manuel raporlama veri tabanlarından alınan URL'lerden oluşan kara listelerden yararlanırlar. Trendmicro Internet Security (Trendmicro, 2021), Norton Safe Web Plugin (Norton, 2021), Google Safe Browsing (Google, 2020), Microsoft Smart Screen (Microsoft, 2021) gibi servislerden bu hizmetler alınabilmektedir. Her ne kadar bu listeler sık sık güncelleniyor olsa da bir takım kötücül siteleri gözden kaçırırlar (Goutam ve ark., 2017).

2.2. İçeriğe Dayalı Sınıflandırma

Jim ve ark. tarafından Javascript bağlantılı saldırıları önlemek için tarayıcı tarafında gömülü politikalar isimli bir mekanizma önerilmiştir. Her bir web sayfası için bir güvenlik politikası tanımlanmıştır. Bu nedenle tarayıcı, komut dosyasını belirtilen politikaya göre ziyaret edilen web sayfasında yürütür. Mekanizma tüm web siteleri için uygulanabilir yapıda olup, politikanın iyi belirlenmiş olması durumunda iyi sonuç vermektedir (Trevor ve ark., 2007).

Xiang ve ark. CANTINA'yı (Yue ve ark., 2007) genişleterek CANTINA+'ı (Guang ve ark., 2011) önerdi. CANTINA, metin tabanı bir kimlik avı algılama tekniği olup, belge içindeki kullanım frekanslarına göre anahtar kelimeleri çıkarmaktadır. Daha sonra anahtar kelimeler Google arama motorunda aranmaktadır. Web sitesi arama sonuçlarına dahil edilmişse iyicil olarak sınıflandırılmıştır. Çalışmanın sadece İngilizce diline duyarlı olması başarısını sınırlandırmaktadır. CANTINA+'da ise 15 adet html tabanlı özellik kullanılmıştır. Sistem %92 doğruluk ile sınıflandırma başarısına ulaşmıştır ancak FP oranı oldukça yüksektir.

URL ve HTML özelliklerini kullanarak kimlik avına ilişkin olarak hazırlamış web sayfalarını tespit etmek için bir modeL önerilmiştir (Yukun ve ark., 2019). GradientBoosting, XGBoost ve LightGBM sınıflandırıcıları, çok katmanlı olarak birleştirilmiş ve kimlik avı web sayfalarını algılamada daha yüksek performansa sahip bir yığın modeli tasarlanmıştır. 49497 ve 53103 web sayfasından oluşan iki veri seti ile yapılan çalışmalarda %97.30 ile doğru sınıflandırma yapılırken, yanlış pozitif değeri %4.46'da kalmıştır.

2.3. Url Tabanlı Sınıflandırma

URL içindeki simgeleri ve tüm n-gramları (n=4 ten 8) dikkate alarak çeşitli özellikler çıkarıp, URL sınıflandırılmasının ayrıntılı bir analizinin gerçekleştirildiği çalışmada (Baykan ve ark., 2011), herhangi bir özellik seçme yöntemi uygulamadan n-gram özelliklerinin önemli ve yeterli olduğu gösterilmiştir. Eğitim boyutunda büyümenin n-gram üretmede sorun oluşturması sebebiyle büyük ölçekli verileri için uygun olmadığı belirtilmiştir.

Rajalakshmi ve ark. tarafından gerçekleştirilen çalışmada (Rajalaskshmi ve ark., 2018), URL özelliklerinin otomatik öğrenilmesine yönelik bir yaklaşım önerilmiştir. Bu yaklaşımda, sadece 4-gram ile özellik çıkarmak yerine, web sayfasının kategorilerini belirlemek için n=3,4,5,6,7,8 gibi tüm n-gramları çıkarılarak her bir token için özellikler üretilmiştir. Veri setinden bağımsız olarak çalışmakta olup, sınıflandırma için Naive Bayes kullanılmıştır.

Zouina ve ark. (Zouina ve ark., 2017) tarafından tamamen URL'ye dayalı olarak kimlik avı saldırılarının algılanmasına yönelik bir yaklaşım önerilmiştir. 1000 kötücül ve 1000 iyicil URL'den oluşan bir veri seti ile çalışılmış olup SVM ile sınıflandırma yapılmıştır. URL boyutu, kısa çizgi sayısı, nokta sayısı, sayısal karakter sayısı ve benzerlik indeksi olmak üzere 6 özellik çıkarılarak kullanılmıştır. Sistem %95.80 tanıma oranı ile çalışmaktadır.

Görsel olarak yakında olan iki öğenin muhtemelen aynı sınıfa ait olduğu ve benzer şekilde benzer URL'lerinde muhtemelen hedef olarak benzer sayfalara sahip olduğu fikrine dayalı olarak önerilen çalışmada (Lawrence ve ark., 2004), denetimsiz bir sınıflandırma modeli önerilmiştir. Model, URL adreslerinin "/",

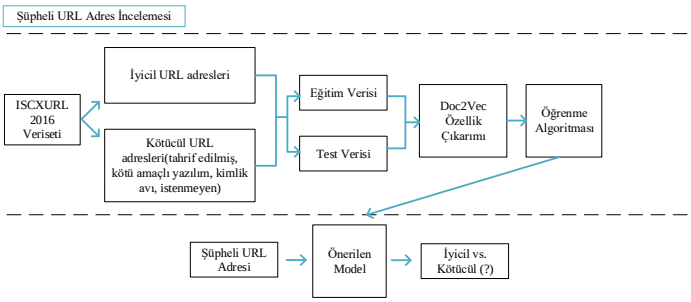
"&" ve "?" karakterlerine göre ayrıştırmakta ve elde edilen her bir tokenı ağaç yapısına yerleştirmektedir. "http" adresi kök olursa, devamındaki tokenlar alt düğümlere eklenir. Ağaçtaki her bir yaprak eğitilmiş bir URL kümesine sahip sınıfı temsil eder ve her bir yeni URL için tüm yapraklara göre olasılıklar hesaplanır. Her bir şüpheli URL en yüksek olasılığa sahip düğüme dahil edilir.

URL adresine ilişkin olarak adres uzunluğu, adres içerisindeki çizgi sayısı, URL ve alt alan adlarındaki noktaların sayısı ve konumu gibi URL tabanlı özelliklerin kullanıldığı ve taşıma katmanı güvenliğine ilişkin olarak yapılan çalışmada (Carolin ve ark., 2016), apriori algoritması kullanılarak kurallar oluşturulmuştur. Deneysel sonuçlarda, kimlik avı URL'lerini %93 oranı ile sınıflandırabildiği gösterilmiştir.

Sungjim Kim ve ark. tarafından yapılan çalışmada (Sungjin ve ark., 2018), URL tabanlı olarak kötücül niyetli URL'lere ait davranış izlerini çıkararak sınıflandırma yapılan model önerilmiştir. 1529433 kötü amaçlı URL içeren bir veri seti ile çalışılmıştır. Saldırganların URL'lerle ilgili taktik davranışlarını analiz etmekte ve ortak özelliklerini çıkarmaktadır. Buna göre 3 seviyeli bir güvenlik ihlali düzeyi belirlenmektedir. Önerilen yaklaşım ile %70 ve üzerinde bir doğrulukla sınıflandırma yapılabilmiştir. URL'in kötü niyetli olup olmadığını tahmin etmek için bir tür web filtresi ve risk bazlı ölçekleyici tasarlanmıştır.

3. Metodoloji

Bu çalışmada web sayfalarının URL adresleri analiz edilerek isteklerin kötücül olup olmadığı tespit edilmektedir. Bu tespiti yapabilmek için Şekil-1'de gösterilen akış şemasına göre süreç yürütülmektedir.



Şekil 1. Şüpheli URL Adreslerinin Analizi Yaklaşımı

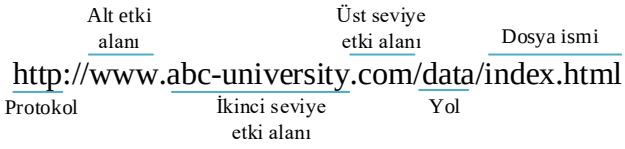
Şüpheli URL adreslerinin analiz edilerek kötücül veya iyicil olup olmadıklarını tespit etmek amacıyla önerilmiş model incelendiğinde ilk aşamada model eğitiminde kullanılmak üzere bir veri setine ihtiyaç duyulmaktadır. Önerilen modelin başarısındaki objektifliği sağlayabilmek ve benzer çalışmalar ile karşılaştırabilmek için 2016 yılında yayınlanmış ve bugüne kadar birçok URL filtreleme çalışmasında kullanılmış ISCX2016URL veri seti kullanılmıştır. Veri seti incelendiğinde içerisinde hem iyicil hem de Spam URL'leri, kimlik avı URL'leri, kötü amaçlı yazılım dağıtan web sitelerine ait URL'ler ve tahrif edilmiş URL'ler bulunmaktadır. Birinci gruptaki testler için sadece iyicil ve kötücül ayrımı yapıldığı için 4 farklı kötücül türündeki veriler tek bir grup altında toplanmıştır. İkinci test grubunda ise sadece 4 farklı kötücül grup için çok sınıflı bir test gerçekleştirilmiştir.

Veri setinin toplanması ve bir takım ön işlem süreçlerinden geçirilmesinden sonra, elde edilen iki sınıfa ait veri seti eğitim ve test aşamalarında kullanılmak üzere rastgele olarak %70-%30 olacak şekilde ayrılmıştır. Bu ayrım sonrasında bu çalışmada önerilen modelin en özgün yanı olan URL adreslerinin özelliklerinin çıkarılması için Doc2Vec modelinden yararlanılmıştır. Hem DM hem de DBOW modeli ile eğitim gerçekleştirilmiştir. Eğitim aşamasında kullanılacak tokenları elde etmek için URL adresleri "/" karakterine göre ayrıştırılmıştır. Tokenların eğitimi sonrasında sınıflandırma algoritmalarında kullanılmak üzere URL adreslerini temsil eden vektörler elde edilmiştir. Böylece oldukça pratik ve hızlı bir şekilde URL adreslerinin özellikleri çıkarılmıştır.

Elde edilen özellik vektörleri farklı makine öğrenmesi algoritmaları için giriş değeri olarak değerlendirilmiştir. Sonuçta ortaya sadece URL adreslerinden çıkarılan özellikleri kullanan bir model ortaya çıkmıştır. Modelin eğitim süreci tamamlandıktan sonra, kötücül ve iyicil kategorideki URL adresleri test edilmekte ve sonuçta ikili bir sınıflandırma yapılmaktadır. Sınıflandırma modeli çoklu sınıfta çalışacak şekilde güncellendikten sonra aynı metodoloji kullanılarak kötücül gruptaki URL adreslerinin de kendi arasında sınıflandırılması ikinci grup testlerde gerçekleştirilmiştir.

3.1. URL Ön işlem

URL, bir kaynağı ve erişim protokolünü tanımlayan bir karakter dizisidir. URL sözdizimi IETF tarafından RFC 3986 de tanımlanmıştır. Bu tanıma göre, bir URL farklı segment türlerinden oluşmakta olup Şekil-2'de gösterilmiştir. İlk olarak bir protokol (ör. html, ftp) sonrasında bir otorite veya alan adı (ör. https://dergipark.org.tr) daha sonra eğik çizgi karakterleriyle ayrılmış bir dizi web adres yolu(ör. /tr/pub) ve son olarak iki isteğe bağlı bölüm olarak bir soru işareti ile devamında bir sorgu dizisi veya # işareti bulunur. Sorgu dizesi, web sunucusuna gönderilen parametrelerin adları ve değerleri hakkında bilgi sağlayan bir yapıdır (ör. ? arastirmaciid=303076&alan=fen). Her bir segment bir sayfayı işaret ederken, her bir parametre o sayfaya ait özellikleri tanımlar. Bir URL adresi "/, ?, #, &, = ve :" gibi ayırıcılara sahip olup, istenilen sayıda karakter barındıran bir dizidir (Imma ve ark., 2016).



Şekil 2. URL Bölümleri

Bu çalışmada sadece http trafiğindeki URL'lerle ilgilenildi. Bir uygulama, kötü amaçlı bir URL'i ziyaret ettiğinde, enfekte olabilir. Bunun yanında, çoğu kötü amaçlı yazılım, kötü amaçlı davranışlarını uygulamak için komut almaları gerektiğinde, URL'de bulunan parametreleri kullanır. Bu nedenle URL'lere dayalı kötücül yazılım araçları etkilidir (Shanshan ve ark., 2020).

3.1. URL Vektör Temsili

Kelimelerin veya kelime gruplarının vektörlere dönüşümleri, günümüzde birçok doğal dil algoritmasında kullanılan bir yaklaşımdır. İlk girişimler, kelimeleri yüksek boyutlu vektörler olarak temsil ederek, kelimelerin anlam ve sonlarına göre kümeleyerek bu sorunu çözmeye çalışmıştır (Petros ve ark., 2018). Son yıllarda sinir ağlarının dil modellemesin kullanılmasında kelimelerden üretilen bu vektörlerin kullanılması fikri önerilmiştir. Kelimelerin bu şekilde temsil edilmesine yönelik yaklaşım Word to Vector (word2vec) olarak bilinmektedir. Elde edilen böyle bir vektör sayesinde boyut

indirgemesi, kümeleme, sınıflandırma, benzerlik arama gibi farklı veri manipülasyon yaklaşımları kullanılabildiğinden metinler ile çalışmayı kolaylaştırmaktadır (Petros ve ark., 2018).

Kelimelerin bu kadar yoğun bir şekilde temsil edilebilmeleri sağlamak için Continuous Bag of Words (CBOW) (Tomas ve ark. 2013)) ve Skip-Gram (Tomas ve ark., 2013) modelleri önerilmiştir. Bir kelime dizisinin [w1, w2, w3, …, wn] olarak temsil edildiği varsayıldığında, CBOW modeli ilk olarak her bir kelime vektörünü rastgele olarak başlatır ve ardından sonucu tahmin edilen kelimenin vektörü olan tek katmanlı sinir ağı kullanarak orijinal tahminler ile modeli optimize eder. Bunun yanında Skip-gram modelinde ise, tam tersi olarak bağlam kelimelerini tahmin etmek için "w" kelimesini kullanır. Tahmin görevi aşağıda gösterilmiş olan eşitliğe göre ayrıştırılabilir. Word2vec modelinin amacı ortalama log olasılığını maksimize etmektir (Tomas ve ark., 2013).

$$\frac{1}{T}\sum_{t=1}^{T-1}\log p(w_t \mid w_{t-1},\dots,w_{t+1}) \quad (1)$$

w_t kelimesi Softmax gibi bir çoklu sınıflama aracı kullanılarak kolayca tahmin edilebilir.

$$p(w_t \mid w_{t-1},\dots,w_{t+1}) = \frac{e^{y_{w_t}}}{\sum_i e^{y_{w_t}}} \quad (2)$$

y_{w_t} terimlerinin her biri, her bir çıkış w_i sözcüğü için normalize edilmemiş log olasılığını gösterir ve şu şekilde hesaplanır.

$$y = b + Uh(w_{t-1},\dots,w_{t+1};W) \quad (3)$$

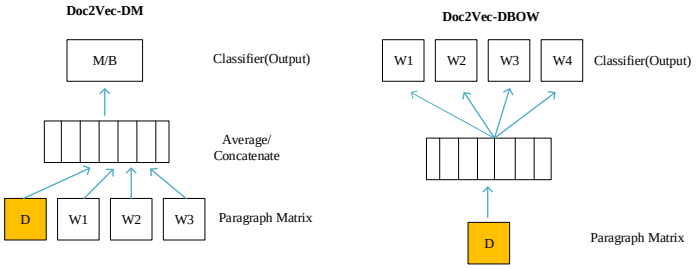
b, gizli ve çıkış katmaları arasındaki bias değerini, U, gizli ve çıkış katmanları arasındaki ağırlık matrisini, h bağlam sözcükleri için birleşim veya ortalama değerini ve W ise sözcük temsil matrisini ifade eder.

Word2vec modelinin sonrasında paragraph2vec ve son olarak doc2vec modelleri ortaya atılmıştır. Doc2vec kelime vektör modeli üzerine geliştirilmiştir. Kelimelerin ve kelime sırasının dikkate alınmaması ve aynı temsile sahip farklı cümlelerin ortaya çıkması ile sonuçlanmaktadır. Doc2vec'te, D, eğitim veri setindeki tüm belgeler için bir vektörü temsil eder. Her belge D matrisindeki bir sütunla temsil edilen benzersiz bir vektör ile eşlenir ve her bir kelime W matrisindeki bir sütunla temsil edilen benzersiz bir vektörü ifade eder. Bu nedenle ağ formülasyonunda D eklenerek Word2Vec, Doc2Vec haline dönüştürülür.

$$y = b + Uh(w_{t-1},\dots,w_{t+1};W,D) \quad (4)$$

Doc2Vec paragraf vektör olarak ifade edilen her bir belgenin temsilini 2 farklı algoritma kullanarak yapar. Dağıtılmış bellek (DM) ve DBOW. DM, CBOW'un bir uzantısıdır ve bu modeldeki tek değişiklik, yeni bir belge ID'si eklemektir. Şekil 3'te turuncu renk ile gösterilmiştir. DBOW ise Skip-gram'ın bir uzantısıdır ve mevcut sözcük, geçerli belge ID'si ile değiştirilir. Şekil1'de DM ve DBOW yapıları gösterilmiştir.

Ağ yeterince eğitildikten sonra, her bir belgenin temsili sağlanmış olur. Bu da daha sonra kümeleme veya sınıflandırma gibi metin madenciliği görevlerinde kullanılabilir. Bu çalışmada kelime kümesi, iyicil veya kötücül web adreslerinin "/" karakterine göre ayrılmış her bir bölümünde oluşmaktadır.



Şekil 3. Doc2Vec DM ve DBOW Model Yapısı

4. Deneysel Testler ve Sonuçlar

Bu bölümde önerilen modelin sınıflandırma algoritmalarına ilişkin deneysel ayrıntıları verilmiştir. DM ve DBOW modeli ile özellik çıkarma yapısı, veri seti ayrıntısı ve özellikleri kullanan algoritmaların sonuçları karşılaştırmaları olarak gösterilmiştir.

4.1. Deneysel Ortam

DM ve DBOW modelleri için kullanılan hiper-parametreler Tablo-1'de gösterilmiştir. İki model arasındaki parametreler incelendiğinde en temel fark dm değerinin DBOW modelinde 0 olarak belirlenmesidir. Diğer parametreler probleme bağlı olarak değişkenlik gösterebilmektedir. Bu çalışmada tabloda gösterildiği gibi belirlenmiştir.

Tablo 1. DM ve DBOW Modelleri için Belirlenen Hiper Parametreler

DM		DBOW	
Parametre	Değer	Parametre	Değer
Size	180	Dm	0
Window	10	Vector_size	300
Min_count	2	Negative	5
Sample	0	hs	0
Negative	5	Min_count	0
Workers	All cores	Sample	0
Dm	1	alpha	0.065
Dm_mean	1	Min_alpha	0
alpha	0.065	Workers	All cores
Min_alpha	0.0	Window	10
hs	0		

Elde edilen URL listelerinde ön işlem aşamalarını yürütebilmek için Beatifulsoup, lxml, tqdm, nltk gibi üçüncü parti Python kütüphanesi kullanılmıştır. Deneyler 2.0 Ghz Intel Core i7 işlemciye ve 8GB 1867 Mhz DDR3 RAM'e sahip bir Windows dizüstü bilgisayar üzerinde gerçekleştirilmiştir. Önerilen modelin test edilebilmesi için Python programlama dilinde Gensim Kütüphanesi kullanılarak yazılım geliştirilmiştir. DBOW ve DM modelleri kullanılarak eğitim ve test aşamaları gerçekleştirilmiştir.

Her bir test seti farklı makine öğrenme algoritması ile koşulmuştur. Karışıklık matrisindeki değerler kullanılarak, algoritmanın başarısı ve verimliliğini ölçmek için kesinlik, duyarlılık, f-ölçütü ve doğruluk olmak üzere 4 farklı metriğe göre hesaplama yapılmıştır.

4.2. Veri Seti

Bu çalışmada önerilen modelin test edilebilmesi için ISCX-URL2016 veri seti kullanılmıştır (Momammad ve ark., 2016). Bu

veri setinde toplanmış olan URL'ler toplam 5 farklı sınıfa ayrılmıştır. Her bir gruba ait URL sayısı ve elde edilme metotları Tablo-2'de gösterilmiştir.

Tablo 2. ISXCURL2016 Verisetinde URL Tipine Göre Örnek Sayısı Dağılımı

URL Tipi	Elde Edilme Yöntemi	URL Sayısı
İyicil URL	İyicil URL içeren web sitesi adresleri Alexa'nın en iyi web sitelerinden seçilmiştir. Domainlere ait web URL'lerinin çıkarılmasında Heritrix web crawler'ı kullanılmıştır. Taranan her bir URL içerisinden iyicil olanları filtrelemek için Virüstotal sitesi kullanılmıştır.	35380
Spam URL	WEBSPAM-UK2007 verisetinden alınmıştır.	12001
Kimlik avı URL	OpenPhish aktif kimlik avı sitesi deposundan alınmıştır.	9967
Kötü amaçlı yazılım dağıtan URL	Kötü amaçlı yazılım siteleri listesini tutan DNS-BH üzerinden alınmıştır.	11567
Tahrif edilmiş URL	Sahte veya gizli URL barındırıp kötücül sınıfta yer alan URL'ler Alexa tarafından sıralanan güvenilir sitelerden alınmıştır.	96457

Tablo 2'de görüleceği üzere veri seti farklı kaynaklardan alınan, farklı tür ve sayıda toplam 165372 adet URL adresi barındırmaktadır. Her ne kadar homojen bir dağılım olmasa da sınıflandırma algoritmalarında kullanmak için uygundur. Aşağıda örnek birkaç URL adresi gösterilmiştir.

http://www.sind3usc3ongoias.com.cr/index.html

http://www.tehobsledovanie.ru/zakazat

https://www.blogs.miis.edu/trade/2011/01/27/top-10-consulting-firms/

https://www.blogs.scripps.com/albq/staley/

Bu çalışmada URL tipleri öncelikle iyicil ve kötücül (spam, phishing, malware, defacement) olarak ikili bir değerlendirmeye tabi tutulmuştur. Bunun için tüm URL'ler tek bir dosyada birleştirilip sınıf adı tanımlanmıştır. Çalışmanın ikinci aşamasında ise, kötücül URL tiplerini kendi arasında sınıflandırmaya yönelik olarak çoklu sınıf barındıran bir model tasarlanmıştır.

4.3. Göstergelerin Değerlendirilmesi

Literatürde çeşitli değerlendirme göstergeleri bulunmak ile birlikte en yaygın olanları kesinlik, duyarlılık, F-skoru ve doğruluk hesaplama formülleri Eş-5 – 8'te gösterilmiştir.

$$Kesinlik = \frac{TP}{TP + FP} \quad (5)$$

$$Duyarlılık = \frac{TP}{TP + FN} \quad (6)$$

$$F\ skoru = 2\ x\ \frac{Kesinlik\ x\ Duyarlılık}{Kesinlik + Duyarlılık} \quad (7)$$

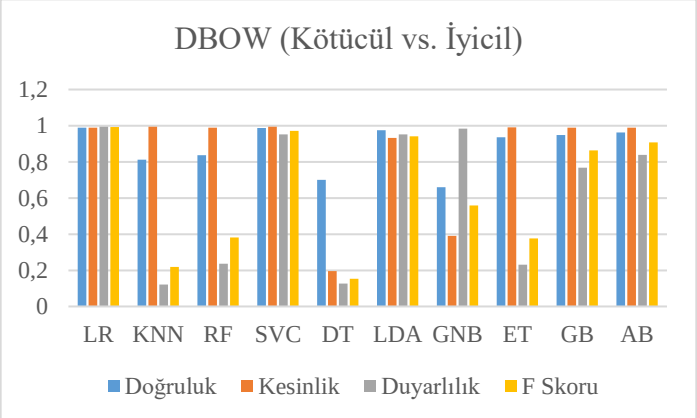
$$Doğruluk = \frac{TP + TN}{TP + TN + FN + FP} \quad (8)$$

Eşitliklere göre, Gerçek Pozitif (TP) iyicil web sayfalarının doğru bir şekilde sınıflandırıldığı sayıyı, Yanlış Pozitif (FP) kötücül web sayfalarının iyicil olarak sınıflandırılma sayısını, Gerçek Negatif (TN), kötücül web sayfalarının kötücül olarak sınıflandırılma sayısını ve Yanlış Negatif (FN) ise iyicil web sayfalarının kötücül olarak sınıflandırıldığı sayıyı göstermektedir. Kesinlik metriği, kötücül olarak sınıflandırılan bir URL adresinin gerçekten % kaçının kötücül bir adres olduğunu göstermektedir. Yanlış pozitifin maliyetinin yüksek olduğu problemler için önemli bir ölçüm değeridir. Duyarlılık, sadece kötücül adreslerle ilgili bir değerdir. Kötücül olan URL adreslerinin kaç tanesinin tespit edildiğini gösterir. Yanlış negatifin maliyetinin yüksek olduğu problemler için önemli bir ölçüm değerini ifade eder. F score ise kesinlik ve duyarlılık değerlerinin harmonik ortalamasıdır. Doğruluk değeri ise en temel performans ölçüm metriği olup, doğru olarak tahmin ettiğimiz URL adreslerinin toplam URL adresi sayısına oranını ifade etmektedir. Tek başına yeterli bir ölçüm olmayıp model hakkında genel bir fikir vermektedir.

4.4. İkili Sınıflandırma (İyicil ve Kötücül)

Bu çalışmanın ilk aşamasında önerilen modelin performans değerlendirmesi bir URL adresinin iyicil mi kötücül mü olduğuna ilişkindir. Bu testler için kullanılan veri tabanındaki kötücül uygulamalar tek bir sınıf altına toplanarak ikili bir sınıflandırmaya tabi tutulmuştur. Sınıflandırma için ihtiyaç duyulan özellik vektörleri Doc2Vec algoritmasına göre çalışan DBOW ve DM modelleri kullanılarak ayrı ayrı elde edilmiştir. Sonrasında da bu iki model birleştirilerek hibrit bir yapı da testler tekrarlanmıştır.

Sınıflandırma için lojistik regresyon (LR), K-en yakın komşu (KNN), rassal orman (RF), destek vektör makinesi (SVC), karar ağacı (DT), lineer discriminant analizi (LDA), gaussion naïve bayes (GNB), ektra ağaç (ET), gradyan artırma (GB) ve adaboost (AB) algoritmaları kullanılmıştır. 10 farklı makine öğrenmesi tekniği ile gerçekleştirilen ve özelliklerin DBOW modeli ile çıkarıldığı model için testlerde elde edilen sonuçlar Şekil-4'te grafik olarak gösterilmiştir.

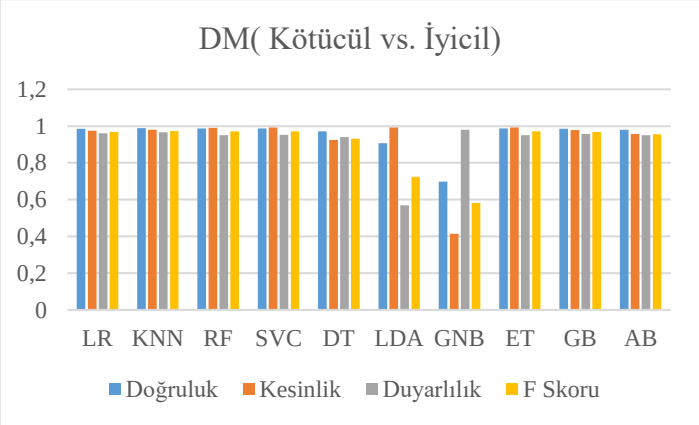


Şekil 4. DBOW Modeli Kullanılarak Elde Edilen Sonuçlar

Grafik incelendiğinde, en yüksek doğruluk değeri Logistic Regresyon ile elde edildi ve %99.2 oranı ile sınıflandırma yapılmıştır. Bunun yanında kesinlik, duyarlılık ve f skoru değerleri de sırasıyla %98.9, %99.1 ve %99.2 olarak elde edilmiştir. Böylece elde edilen doğruluk değerinin başarısı kanıtlanmıştır. DBOW modeli kullanılarak elde edilen özellik vektörü ile yapılan sınıflandırma da sadece URL adresi kullanarak

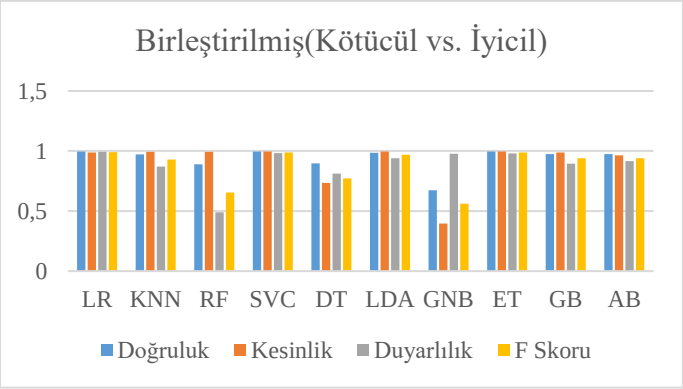
elde edilen özellikler ile oldukça yüksek bir başarı değeri yakalanabilmiştir.

Benzer şekilde Doc2Vec yapısındaki bir diğer model olan DM ile çıkarılan özellikler kullanılarak yapılan sınıflandırmada en yüksek başarı oranı RF, KNN, Extra Tree ve SVC ile elde edilmiş olup sırasıyla %98.8, %98.9, %98.8 ve %98.8 doğruuk değerleri elde edilmiştir. Her bir sınıflandırıcı için elde edilen sonuçlar Şekil-5'te verilmiştir. Başarı oranının genel olarak tüm sınıflandırıcılarda yüksek olduğu gösterilmiştir. DBOW modelinde olduğu gibi, sadece URL tabanlı olarak DM modeli ile üretilen özellik vektörlerinin kötücül ve iyicil sınıflandırmasında başarılı olduğu gösterilmiştir.



Şekil 5. DM Modeli Kullanılarak Elde Edilen Sonuçlar

URL adreslerinin iyicil ve kötücül olarak sınıflandırmasında gerçekleştirilen son test ise DBOW ve DM modellerinden gelen özellik vektörlerinin birleştirilmiş(concatenate) hali ile yapılmıştır. İki vektörün birleştirilerek kullanıldığı son testte DBOW veya DM modellerinde düşük performans gösteren sınıflandırıcıların performansında bir artış yakalandığı gözlemlenmiştir. Böylece önerilen modelin genel olarak tüm sınıflandırıcılar ile etkin bir sonuç verebilmesi mümkün olmuştur. Hibrit modelin katkısı düşük performans gösteren algoritmaları açısından faydalı olmuştur. Bunun yanında en yüksek performansa sahip algoritmalar açısından da küçükte olsa düşüşe neden olmuştur. Hibrit vektör kullanılarak yapılan testlere ilişkin sonuçlar Şekil-6'da gösterilmiştir.



Şekil 6. Birleştirilmiş Özellik Vektörü Kullanarak Elde Edilen Sonuçlar

4.5. Çoklu Sınıflandırma

Aynı veri setinde kötücül URL adresleri Spam, Malware, Phishing ve Defacement olarak ayrılmıştır. Doc2Vec modeli ile üretilen özellik vektörleri ile kötücül URL adreslerinin kendi içinde sınıflandırılmalarına ilişkin olarak ikinci bir test yapılmıştır. Doc2Vec yapısının URL adres sınıflandırmasında çoklu sınıfa sahip modeller için başarısı ölçülmeye çalışılmıştır. Buna göre DBOW, DM ve Hibrit modele göre en iyi sonuçların alındığı algoritmaların performans değerleri Tablo-3'te gösterilmiştir.

Tablo 3. DBOW, DM ve Birleştirilmiş Hibrit Model için En İyi Sonuçlar

	Algoritma	Doğruluk	Kesinlik	Duyarlılık	F Skoru
DBOW (Multiclass)	LR	0,878	0,888	0,898	0,877
	SVC	0,841	0,840	0,842	0,841
	LDA	0,841	0,840	0,843	0,840
DM (Multiclass)	LR	0,755	0,754	0,756	0,759
	KNN	0,747	0,748	0,747	0,749
	RF	0,783	0,783	0,784	0,782
	ETREE	0,758	0,754	0,759	0,760
CONCAT (Multiclass)	LR	0,881	0,880	0,882	0,881
	SVC	0,883	0,882	0,881	0,884
	LDA	0,850	0,851	0,852	0,849

Tablo-3'te sonuçlara göre çoklu sınıflandırmada en iyi sınıflandırma sonucu %88,3 olarak elde edilmiştir. Bu sonuç DBOW ve DM modellerinden elde edilen özellik vektörlerinin birleştirilmiş halini kullanan birleştirilmiş hibrit model ile yakalanmıştır. Çoklu sınıflandırmada sonuçlarında ikili sınıflandırmaya göre daha düşük sonuç elde edilmesinin sebebinin 4 kümeye ait örnek sayısındaki ciddi dengesizlik olduğu düşünülmektedir.

4.6. Önerilen Modelin Avantajlı Noktaları

İçeriğe bağlı web site analiz araçlarının çoğunda, sistemin yürütülmesi için dil oldukça önemlidir. Bu çalışmada önerilen model yalnızca URL adreslerini segmentlere ayırarak oluşturulmuş öznitelikleri barındıran vektörler kullanmaktadır. Bu sebeple, sistemin başarısı oluşturulan kelime vektörüne bağlı olup dilden bağımsız bir şekilde çalışmaktadır.

Günümüzde web sitelerini oluşturmak hem kolay hem de ucuz bir iştir. Bu sebeple kötücül faaliyet göstermek üzere saldırganlar tarafından oluşturulan web sayfaları hızlı bir şekilde aktif olmakta ve kısa sürelerde çalıştırılmaktadırlar. Bu nedenle web sitelerinde kötücül tespiti yapabilmek için mümkünse gerçek zamanlı çalışan araçlar geliştirilmesi beklenmektedir. Bu çalışmada önerilen sistemde sadece URL adresleri kullanıldığı için oldukça hızlı bir analiz yapılmakta ve web siteleri ihmal edilebilir sürelerde sınıflandırılmaktadır.

Sadece URL adresinden üretilen kelime vektörleri kullanılarak yapılan sınıflandırma sayesinde daha önce etiketlenmemiş ve oldukça tehlikeli saldırı türlerinden biri olan sıfırıncı gün saldırılarına karşıda etkindir. Kara liste/beyaz liste, sıralama sayfaları, ağ trafik ölçüleri, etki alanı algılama gibi literatürde önerilmiş birçok model bulunmaktadır. Bu sistemler saldırı ve tespit sistemlerinin etkinliği artırmakta oldukça önemlidirler. Ancak bunlardan gerçek zamanlı durumlar için yararlı olamazlar.

4.7. Önerilen Modelin Diğer Çalışmalar ile Karşılaştırılması

Bu çalışmada önerilen modelin test edilebilmesi için ISCXURL2016 veriseti kullanılmıştır. 2016 yılından beri açık kaynak olarak paylaşılmasından dolayı bu veri seti kullanılarak URL adreslerinin sınıflandırılmasına yönelik olarak birçok çalışma yapılmış olup, detayları Tablo-4'te gösterilmiştir.

Tablo 4. ISCXURL2016 Veri Seti Kullanılan Yapılan Çalışmalar

Çalışmanın Adı ve Yılı	Kullanılan Özellikler	Sınıflandırma Yöntemi	Performans Sonucu
Uçar and Uçar (Uçar ve ark., 2019)	Verisetinde sunulan ve URL adresine ilişkin 80 özellik kullanılmıştır.	LSTM, CNN(3 Max-pool, 1 Hidden, 1 Dropout) İkili sınıflandırma	Doğruluk: LSTM: 91,13 CNN: 95,37
Kapil et al. (Kapil ve ark, 2019)	Verisetinde sunulan ve URL adresine ilişkin 80 özellik içerisinden 47 adet seçilerek kullanılmıştır.	J48, Random Forest, Bayes-Net, Lazy İkili sınıflandırma	Doğruluk: J48:94.4 RF:96.1 BayesNet:92.1 Lazy: 95.4
Deebanchakkara warthi et.al (Deebanckhakka rawarthi ve ark., 2019)-owner of dataset	Verisetinde sunulan ve URL adresine ilişkin 80 özellik kullanılmıştır.	Random Forest	Doğruluk: RF: 97.0
Raju et al. (Raja ve ark, 2020)	Verisetinde sunulan ve URL adresine ilişkin 80 özellik kullanılmıştır.	RF,ExtraTree, Adaboost	Doğruluk: RF:94.1 ExtraTree:94.9 Adaboost: 93.2
Dawn and Tavares (Dawn ve ark., 2019)	Lexical ve Host tabanlı 32 özellik çıkarılmıştır.	ExtraTree, Adaboost	Doğruluk: ExtraTree:91.0 Adaboost: 90.0
Bu çalışmada önerilen model (2021)	Doc2Vec yapısında, DBOW ve DM modelleri ile URL adreslerinin segmentlere ayrılmış hali kullanılarak özellik vektörleri üretilmiştir.	DBOW : (LR,SVC,LDA, AdaBoost) DM: (KNN,RF,SVC,ExtraTree)	Doğruluk (DBOW): LR:98.9 SVC:98.8 LDA:97.5 AdaBoost:96.3 Doğruluk(DM): KNN:98.9 RF: 98.8 SVC: 98.8 ExtraTree:98.8

Tablo4'te ISCX2016 URL veriseti ile kullanılarak yapılan ve URL adreslerinin iyicil veya kötücül olup olmadığına dair yapılan test sonuçları gösterilmiştir. Çalışmalar incelendiğinde iki tanesi haricindekilerin veri seti ile birlikte sunulan sorgu uzunluğu, etki alanı token sayısı, URL adresi token sayısı, ortalama etki alanı token sayısı, en uzun etki alanı token uzunluğu gibi 80 farklı özelliğini sınıflandırma için kullanıldığı anlaşılmaktadır. Bunun ile birlikte bir çalışmada 80 özellik içerisinden özellik seçimi yapılarak daha az sayıda özellik ile testler yapılmış olup, bir çalışmada da bizim çalışmamıza benzer şekilde lexical ve host tabanlı 32 özellik çıkarılarak sınıflandırma yapılmıştır. Sınıflandırma için Random Forest, Extra Tree, Adaboost, LSTM, CNN gibi algoritmalar ve ağ yapıları kullanılmıştır. Bu çalışmalar sonucunda %90 ile %97 arasında değişen oranlarda doğruluk değerleri yakalanmıştır. Bizim çalışmamızda ise özellik çıkarımı için benzer çalışmalardan tamamen farklı olarak Doc2Vec algoritmasındaki DBOW ve DM modelleri kullanılmıştır. Sadece URL adresleri kullanılarak kelimeler öğrenilebilir vektörlere dönüştürüldü ve sınıflandırma için kullanıldı. Bunun sonucunda en yüksek sınıflandırma performansında %98.9 sınıflandırma oranına ulaşılmıştır. Bu performans değerinin güvenilirliği kesinlik, duyarlılık ve f skoru metrikleri ile birlikte ölçülerek kanıtlanmıştır. Sonuçta aynı veri setini kullanan ve tümü 2019 yılı ve sonrasında yapılan güncel çalışmalardan daha iyi sonuç üretebilen bir model önerisi getirilmiştir.

5. Kısıtlar ve Tartışma

Bu çalışmada önerilen yöntem yalnızca HTTP trafiğindeki URL'lere odaklanmaktadır. Bu nedenle http olmayan protokoller ve http şifrelemeleri kullanan kötü amaçlı web sitelerinin tanımlanması/sınıflandırılması gerçekleştirilememektedir. Kötü amaçlı URL tanımlama süreci, eğitim veri kümesi için etiketler gerektirir. Ağın tamamında belirli etiketlere sahip örneklerin bulunması da zor bir süreçtir. Ek olarak, bu çalışmada önerilen model, diğer araçlar kullanılarak oldukça zahmetli bir şekilde gözlemlenebilen veya hiç gözlemlenemeyen kötü niyetli web sitelerini ortaya çıkarmada hızlı ve yardımcı bir araçtır. Önerilen teknik, özellik olarak yalnızca URL'lerin mevcut olduğu senaryolar için uygun bir yaklaşımdır. Örneğin, bazı web sunucuları, ağ trafiğindeki diğer tüm bilgileri açık tutarken, URL verilerini özellikle korumayı tercih edebilmektedir. Bunun yanında kötü amaçlı web siteleri, tespit araçlarını atlatabilmek için şifrelenmiş trafik kullanarak ağlar ile iletişime geçmeyi tercih etmektedir.

Modelimizin bir diğer sınırlaması, özellikle sosyal medya bağlamında son zamanlarda çok kullanılan URL'lerin kısa versiyonları olan URI'ler ile iyi sonuç üretememesidir. Bu URL'lerin kısa uzunlukta olması ve önemli bilgiler içeren terimleri içerisinde barındırmaması sebebiyle, web sitelerine ait özellik vektörlerinin çıkarılması oldukça zor olmaktadır. Örnek olarak https://atif.sobiad.com/index.jsp?modul=kullanici-ayrinti&username=RECEP+S%C4%B0NAN+ARSLAN&arastirmaciid=303076&alan=fen gibi bir URL düşünelim. TinyURL gibi bir web adresi kısaltma servisini kullanarak ilgili URL'i kısalttığımızda hem URL yapısı ve hem de bilgilerinin kaybedilmiş olduğu "tinyurl.com/e7kyacxs" gibi bir adres ortaya çıkmaktadır. Bununla birlikte, kısaltılmış URL'ler yalnızca daha uzun ve yapılandırılmış bir URL'in çevirisi olduğundan ve önerilen model uzun hali ile iyi çalıştığından bu sorunun üstesinden gelinebilir.

6. Sonuçlar ve Gelecek Çalışma Önerileri

Geniş bir kullanım alanına sahip olmaları nedeniyle web sayfalarının sınıflandırılmaları yoğun olarak araştırılan ilginç alanlardan birisidir. Web sayfalarını otomatik olarak sınıflandırmak için geliştirilmiş mevcut araçların birçoğu farklı eksiklikleri sebebiyle gerçek dünyada kullanımları yeterince yaygın değildir. İçerik tabanlı olarak sınıflandırma yapan araçlarda web sitesinin tamamen indirilmesinin gerekmesi ve bunun büyük web siteleri için uygun olmamasıdır. Tüm web sitesinin indirilmesi bu araçların verimliliklerini engeller ve çoğu halde siteye, kullanılan dile ve etki alanına da bağımlıdırlar. Bu sebeple genel olarak kullanılabilir değildirler.

Bu çalışmada önerilen model, URL tabanlı olarak web sayfalarının otomatik sınıflandırılmaları için geliştirilmiştir. Önerilen yaklaşım, URL adreslerinin çeşitli karakterlere göre ayrılmış segmentlerini alırken, her bir sınıfa ait sayfaların URL'lerini temsil eden bir model çıkarır. Bir URL adresinin hangi

bölümlerinin önemli ölçüde sınıfı temsile daha uygun olduğuna veya hangi bölümlerin soyutlanabileceğine karar verme aşaması tamamen model bırakılmıştır. Önerilen aracın en güçlü özellikleri, önceden kapsamlı bir tarama gerektirmemesi, bir web sitesinin indirmeden sınıflandırabilmesi ve sitenin türünden, dilinden ve etki alanından bağımsız olmasıdır. Aracımızı "unb.ca" üzerinde dağıtımı yapılan ve içerisinde 5 farklı gruptan 165 bin URL barındıran veri seti ile doğruladık.

Önerilen çalışma mevcut çalışmalardan daha verimli ve gelişmiş doğruluk seviyesine sahip bir modeldir. Farklı makine öğrenmesi algoritmaları ile testler yapılarak en başarılı sonuç Logistic Regresyon ve KNN sınıflandırıcıları ile elde edilmiştir. Buna göre URL özelliklerinin çıkarılması ve sınıflandırılmasında %98.9 doğruluk oranı yakalanmıştır. Önerilen sistemin temel avantajı kara liste bağımlılığını kaldırması, tahmin için veritabanı gerektirmemesidir. Yönetimi kolaydır ve eğitim verilerine dayanarak kötü amaçlı URL'leri otomatik olarak tahmin eder. URL karmaşıklaştırma(obfuscating) veya atlatma(bypassing) tekniklerine karşı daha fazla güvenlik sağlar. URL kısaltma ve etki alanı oluşturma algoritmalarının ortaya çıkması ile birlikte her geçen gün daha güçlü ve hızlı güvenlik altyapılarına ihtiyaç duyulmaktadır. Gelecekte önerilen model bir vekil sunucusunda veya siber güvenlik uygulamalarında kullanılan bir ağ trafik denetleyicisinde uygulanabilir.

Elde edilen bu sonuçlar, aracımızın gerçek dünya web sayfası sınıflandırması için yeterince umut verici göründüğünü, uygulamada verimli olduğunu ve oluşturduğu modellerin web sayfalarını doğru bir şekilde sınıflandırabildiğini göstermektedir. Model tespit oranı açısından kabul edilebilir olmak ile birlikte sistemin verimliliğini daha da artırmak için derin öğrenme gibi güncel öğrenme teknolojileri kullanılabilir. Ayrıca kısa URL adreslerinden özellik üretmede kısır bir yaklaşım olması sebebiyle yeni bir özellik üretim yaklaşımı önerilebilir. Bunun yanında elde edilebilmesi mümkün olduğu takdirde, web sitelerinin kullanım istatistikleri gibi bazı aktif ve güncel veriler özellik vektörüne eklenebilir.

Kaynakça

Ali A., Mehran F. & Mahmoud K. (2011). Intelligent Classification of web pages using contextual and visual features. Applied Soft Computing, 11(2), 1638-1647.

Arslan R.S. & Barışçı N. (2019). Development of Output Correction Methodology for Long Short Term Memory-based Speech Recognition, Sustainability, 11(15), 4250-4266.

Arslan R.S., Doğru İ.A. & Barışçı N.(2019). Permission-based malware detection system for android using machine learning techniques. International Journal of Software Engineering and Knowledge Engineering, 29(1), 43-61.

Baykan E., Henzinger M., Ludmila M. & Ingmar W. (2011). A comprehensive study of features and algorithms for URL-based topic classification. ACM Transactions on the Web, 5(3), 1-29.

Carolin J. & Elijah B. R.(2016). Intelligent phishing URL detection using association rule mining. Humancentric Computing and Information Sciences, 6(1), 1-19.

Chia-Mei C., Jhe-Jhun H. & Ya-Hui O. (2015). Efficient suspicious URL filtering based on reputation. Journal of Information Security and Applications, 20, 26-36.

Daniel L.S., Angelica G. A. & Juan M. C. (2019). Visual Content-based Web Page Categorization with Deep Transfer Learning and Metric Learning. Neurocomputing, 338, 418-431.

Divya K., Anupriya A.B., Nidi M. & Aditya J. (2019). Machine Learning Based Malicious URL Detection. International Journal of Engineering and Advanced Technology, 8(4), 1-5.

Deebanchakkarawarthi G., Parthan AS, Sachin L. & Surya A. (2019). Classification of URL into Malicious or Benign using Machine Learning Approach. International Journal of Advanced Research in Computer and Communication Engineering, 8(2) 1-4.

Dwan R.A.Jr. & Tavares A.M. (2019). Predictive Analysis: Machine Learning Model for URL Classification (Yüksek Lisans Tezi). Worcester Polytechnic Institute, Worchester.

Florian B., Martin E. & Xiaowei X. (2002, Temmuz). Frequent term-based text clustering. International Conference on Knowledge Discovery and Data Mining(pp. 436-442).

Gideon M. B. W., Thomas D., Eleri A., Herbert T.K., Edwin A. V. & Lambert S. (2021). Structure-Tags Improve Text Classification for Scholarly Document Quality Prediction, arXiv:2004.03705v3.

Google, Google Safe Browsing. (2020, Ocak, 1). Erişim Adresi https://safebrowsing.google.com/

Goutam C. & Tsai T. L. (2017, Aralık). A Url address aware classification of malicious websites for online security during web-surfing. International conference on Advanced Networks and Telecommunications Systems (ANTS)(pp. 1-6).

Guang X., Jason H., Carolyn P. R. & Lorrie C. (2011). CANTINA+: A feature-rich machine learning framework for detecting phishing web sites. ACM Transaction Information System Security, 14(2), 1-28.

Hidayet T., Turker A. & İbrahim S.(2007). A Text Based Anomaly Detection for Web Attacks. Journal of the Faculty of Engineering and Architecture of Gazi University, 22(2), 247-253.

Imma H., Carlos R. R., David R. & Rafael C. (2016). CALA: CIAssifying Links Automatically based on their URL. The Journal of Systems and Software, 115, 130-143.

Jasper P., Shantanu M., Kalliopi Z. & Yingqian Z. (2019, Eylül). Term Based Semantic Clusters for Very Short Text Classification. 12th International Conference on Recent Advances in Natural Language Processing(pp. 878-887).

Jia Z., Qing X., Shoou Y. & Wai H. W.(2016). Exploting link structure for web page genre identication. Data Mining and Knowledge Discovery, 30, 550-575.

Lawrence K. S. & David R. K.(2004, Mayıs). Using URLs and Table Layout for Web Classification Tasks. 13th International Conference on WWW (pp. 193-202).

Microsoft, Microsoft Smart Screen. (2021, Nisan, 6). Erişim Adresi https://support.microsoft.com/en-us/topic/what-is-smartscreen-and-how-can-it-help-protect-me-1c9a874a-6826-be5e-45b1-67fa445a74c8

Mohammed M., Muhammed A. R., Arash H. L. & Natalia S. (Eylül, 2016). Detecting Malicious URLs Using Lexical Analysis. International Conference on Network and System Security(pp. 1-17).

Mohammad S.I.M., Mohammad A.R., Arash H.L., Natalia S. & Ali A. G. (2016). Detecting Malicious URLs Using Lexical Analysis. Network and System Security, 467-482.

Mouad Z. & Benaceur O. (2017). A novel lightweight URL phishing detection system using SVM and similarity index. Human-Centric Computing and Information Science, 7(1), 1-17.

Netcraft, Active Cyber Defence. (2018, 1, Ocak). Erişim Adresi https://www.netcraft.com/

Navisite, Navisite Services. (2021, 5, Nisan). Erişim Adresi https://www.navisite.com/services/.

Norton, Norton Safe Web Plugin. (2021, Nisan, 6). Erişim Adresi https://us.norton.com/feature/safe-web

Özgür K. Ş., Ebubekir B., Onder D. & Banu D. (2019). Machine learning based phishing detection from URLs. Expert Systems with Applications, 117, 345-357.

Petros K., Dimitris G., George G. & Chrysostomos S.(2018, Temmuz). Topic recommendation using Doc2Vec. International Joint Conference on Neural Networks (pp. 1-6).

Rajalakshmi R. & Sanju X. (2017). Experimental Study of Feature Weighting Techniques for URL Based Webpage Classification. Procedia Computer Science, 115, 218-225.

Rajalakshmi R., Hans T., Jay P., Ankit K. & Karthik R. (2020). Design of Kids-specific URL Classifier using Recurrrent Convolutional Neural Network. Procedia Computer Science, 167, 2124-2131.

Rajalakshmi R. & Chandrabose A. (2018). Naive Bayes Approach for URL Classification with Supervised Feature Selection and Rejection Framework. Computational Intelligence, 34(2), 363-396.

Raju B.P.R., Lakshmi B.V. & Narayana C.V. L. (2020). Detection of Multi-class Website URLs Using Machine Learning Algorithms. International Journal of Advanced Trends in Computer Science and Engineering, 9(2), 1-9.

Sungjin K., Jinkook K. & Brent B. K.(2018). Malicious URL protection based on attackers habitual behavioral analysis. Computer and Security, 77, 790-806.

Shanshan W., Zhenxiang C., Qiben Y., Ke J., Lizhi P. & Bo Y., Mauro C.(2020). Deep and broad URL feature mining for android malware detection. Information Sciences, 513, 600-613.

Tie L., Gang K. & Yi P. (2020). Improving malicious URLs detection via feature engineering: Linear and nonlinear space transformation methods. Information Systems, 91, 1-18.

Tomas M., Corrado G.S., Kai C. & Jeffren D. (2013, Mayıs). Efficient estimation of word representations in vector space. International Conference on Learning Representations, Scottsdale-Arizona(pp. 1-12).

Tomas M., Ilya S., Kai C. & Corrado G.S. (2013). Distributed representations of words and phrases and their compositionality. Advanced in Neural Information Systems, 26, 3111-3119.

Trevor J., Nikhil S. & Michale H.(2007, Mayıs). Defeating script injection attacks with browser-enforced embedded policies. International Conference on World Wide Web (pp. 601-611).

Trendmicro, Trendmicro sitesafety.(2021, 6 Nisan). Erişim Adresi https://global.sitesafety.trendmicro.com/

Uçar E. & Uçar M. (2019, Ekim). A Deep Learning Approach for Detection of Malicious URLS. 6. International Management Information Systems Conference Connectedness and Cybersecurity (pp.2-10).

Wei W., Qiao K., Jakub N., Marcin K., Rafal S. & Marcin W.(2020). Accurate and fast URL phishing detector: A convolutional neural network approach. Computer Networks, 178, 1-9.

Yue Z., Jason H. & Lorrie C.(2007, Mayıs). Cantina: a content-based approach to detecting phishing web sites. International Conference on World Wide Web(pp. 639-648).

Yukun L., Zhenguo Y., Xu C., Huaping Y. & Wenyin L. (2019). A stacking model using URL and HTML features for phishing webpage detection. Future Generation Computer Systems, 94, 27-39.

Yurttakal A.H., Erbay H. & Arslan R.S. (2020). Grading Brain Histopathological Images Using Deep Residual Networks and Support Vector Machine. Electronic Letters on Science and Engineering, 16(2), 77-83.