Oltalama Saldırılarının Makine Öğrenmesi ile Tespitinde Kullanılan Özniteliklerin Analizi

Sibel Kapan

YÜKSEK LİSANS TEZİ

Bilgisayar Mühendisliği Anabilim Dalı

Ocak 2021

Analysis of the Features Used in Detecting Phishing Attacks by Machine Learning

Sibel Kapan

MASTER OF SCIENCE THESIS

Department of Computer Engineering

January 2021

Oltalama Saldırılarının Makine Öğrenmesi ile Tespitinde Kullanılan Özniteliklerin Analizi

Sibel Kapan

Eskişehir Osmangazi Üniversitesi
Fen Bilimleri Enstitüsü
Lisansüstü Yönetmeliği Uyarınca
Bilgisayar Mühendisliği Anabilim Dalı
Bilgisayar Bilimleri Bilim Dalı
YÜKSEK LİSANS TEZİ
Olarak Hazırlanmıştır

Danışman: Dr. Öğr. Üyesi Efnan Şora Günal

Ocak 2021

ETİK BEYAN

Eskişehir Osmangazi Üniversitesi Fen Bilimleri Enstitüsü tez yazım kılavuzuna göre, Dr. Öğr. Üyesi Efnan Şora Günal danışmanlığında hazırlamış olduğum "Oltalama Saldırılarının Makine Öğrenmesi ile Tespitinde Kullanılan Özniteliklerin Analizi" başlıklı tezimin özgün bir çalışma olduğunu; tez çalışmamın tüm aşamalarında bilimsel etik ilke ve kurallara uygun davrandığımı; tezimde verdiğim bilgileri, verileri akademik ve bilimsel etik ilke ve kurallara uygun olarak elde ettiğimi; tez çalışmamda yararlandığım eserlerin tümüne atıf yaptığımı ve kaynak gösterdiğimi ve bilgi, belge ve sonuçları bilimsel etik ilke ve kurallara göre sunduğumu beyan ederim. 20/01/2021

Sibel KAPAN

ÖZET

Oltalama saldırısı teknik altyapısı olan sosyal mühendislik saldırısıdır. Oltalama saldırılarının teknik altyapısını, yasal web sitelerinin kopyalanması ile oluşturulan oltalama web siteleri oluşturur. Oltalama saldırılarının yaygınlaştığı günümüzde, oltalama web sitelerinin tespiti için liste veya sezgisel yöntemler tercih edilmektedir. Liste yöntemleri, sıfır gün saldırılarında etkili olmadığından, sezgisel yöntemler oltalama web sitelerinin tespitinde tercih edilmektedir. Bu çalışmada oltalama web sitelerinin tespitinde kullanmak için yeni bir veri kümesi toplanmıştır. Sunulan veri kümesi 25 öznitelik içermektedir. Bu veri kümesinde *HTTP yanıt geçmişi* yeni bir öznitelik olarak literatüre sunulmuştur. Öznitelikler URL, HTML ve HTTP yanıtları olarak 3 gruba ayrılmıştır. Oltalama web sitelerinin tespitinde farklı gruplara ait özniteliklerin etkileri araştırılmıştır. Deneylerde Destek Vektör Makinesi, Stokastik Gradyan İnişi, Perseptron, Naïve Bayes, Çok Katmanlı Perseptron, k-En Yakın Komşu ve Karar Ağacı makine öğrenmesi sınıflandırıcıları kullanılmıştır. Deney sonuçları incelendiğinde URL ve HTTP yanıtları gruplarına ait özniteliklerin, tüm özniteliklerin kullanılmasına kıyasla zaman ve başarım oranları açısından daha iyi sonuçlar verdiği görülmüştür. Karar Ağacı sınıflandırıcısı 0,99'lık F1-skor oranı ile URL ve HTTP yanıtları özniteliklerini kullanarak en iyi başarım oranına sahiptir.

Anahtar Kelimeler: Oltalama, Makine Öğrenmesi, Veri Madenciliği, Bilgisayar Güvenliği, Bilgi Güvenliği

SUMMARY

Phishing attack is a social engineering with a technical infrastructure. The technical infrastructure of phishing attacks consists of phishing websites created by copying legitimate websites. Nowadays when phishing attacks are widespread, list or heuristic methods are preferred to detect phishing websites. Since list methods are not effective in zero-day attacks, heuristic methods are preferred for detection of phishing websites. In this study, a new data set has been collected to be used in detecting phishing websites. The presented data set contains 25 attributes. In this dataset, *HTTP response history* is presented to the literature as a new feature. Attributes are divided into 3 groups as URL, HTML and HTTP responses. The effects of features belonging to different groups were investigated in the detection of phishing websites. Support Vector Machine, Stochastic Gradient Descent, Perceptron, Naïve Bayes, Multi-layer Perceptron, k-Nearest Neighbors and Desicion Tree machine learning classifiers were used in the experiments. When the results of the experiment were examined, the attributes belonging to the URL and HTTP responses groups gave better results in terms of time and performance rates compared to the use of all attributes. Decision Tree classifier has the best success rate using URL and HTTP responses features with a F1-score rate of 0.99.

Keywords: Phishing, Machine Learning, Data Mining, Computer Security, Data Security

İÇİNDEKİLER

Sayfa

İÇİNDEKİLER (devam)

Sayfa

İÇİNDEKİLER (devam)

Sayfa

ŞEKİLLER DİZİNİ

Şekil **Sayfa**

ÇİZELGELER DİZİNİ

Çizelge **Sayfa**

SİMGELER VE KISALTMALAR DİZİNİ

Kısaltmalar	Açıklama
AME	Arama Motoru Eklentisi
CSS	Basamaklanmış Stil Katmanları (Cascading Style Sheets)
DT	Karar Ağacı (Desicion Tree)
ES	Kapsamlı Arama (Exhaustive Search)
HTML	Hiper Metin İşaret Dili (Hypertext Markup Language)
HTTP	Hiper Metin Transfer Protokolü (Hypertext Transfer Protocol)
IP	İnternet Protokolü (Internet Protocol)
kNN	k-En Yakın Komşu (k-Nearest Neighbors)
MLP	Çok Katmanlı Perseptron (Multi-layer Perceptron)
NB	Naïve Bayes
OWS	Oltalama Web Sitesi
ÖS	Öznitelik Seçimi
RF	Rastgele Orman (Random Forest)
SA	Sınıflandırma Algoritması
SGD	Stokastik Gradyan İnişi (Stochastic Gradient Descent)
SSL	Güvenli Soket Katmanı (Secure Sockets Layer)
SVM	Destek Vektör Makinesi (Support Vector Machine)
TÖS	Toplam Öznitelik Sayısı
URL	İnternet Kaynak Belirteci (Uniform Resource Locator)
WS	Web Sitesi
YWS	Yasal Web Sitesi

1. GİRİŞ VE AMAÇ

İnternetin yaygınlaştığı günümüzde, internet sadece kişisel amaçlar için kullanılmaktan çıkmıştır. Birçok şirket ve organizasyon, günlük hayatta kullandığımız hizmetleri sanallaştırarak, hizmetlere hızlı ve kolay erişim olanağı sağlamaktadır. Hizmetlerin hızlı ve kolay ulaşılması beraberinde veri güvenliği problemini getirmiştir. Bu hizmetlere erişim sağlarken kullanılan kişisel bilgiler, finansal bilgiler ve şifreler veri güvenliği problemini oluşturmaktadır.

Günümüzde kişisel ve finansal verileri çalmak için birçok farklı siber saldırı türü bulunmaktadır. Bu saldırı tiplerinden biri olan oltalama saldırısı; suçluların yasal ve bilinen bir web sitesini kopyalayarak oluşturdukları web sitesi aracılığıyla kullanıcıların kişisel ve finansal bilgilerinin çalınmasıdır. Oltalama saldırıları teknik bir sosyal mühendislik olarak görülür. Sosyal mühendislik, psikolojik manipülasyon yöntemleri kullanarak kurbanlardan kişisel verilerinin toplanmasıdır (Dou vd., 2017). Oltalama saldırılarının teknik aşamasını, yasal bir web sitesinin kopyalanarak ve kurbanlara gönderilmesi oluşturur.

Oltalama saldırıları kişilere ait bilgilerin çalınması nedeni ile sadece kişilere değil, şirket ve organizasyonlara da maddi ve manevi zarar vermektedir. Oltalama saldırıları şirketlerde ve organizasyonlarda güven ve prestij kaybına neden olmaktadır. Oltalama saldırılarının etkilerini araştıran Anti-Phishing Working Group (APWG), birçok şirket tarafından desteklenen oltalama saldırılarının etkilerini araştıran bir organizasyondur (APWG, 2003). APWG raporuna göre oltalama saldırılarından en çok perakende ve finansal sektörler etkilenmiştir (APWG Reports, 2017).

Bu tez çalışması kapsamında, oltalama saldırıları hakkında bilgilendirilme yapılarak, bu saldırıların etkilerinden bahsedilmiştir. Oltalama saldırısı için kullanılan sahte web sitelerinin akıllı bir şekilde tespit etmek için, makine öğrenmesi ve veri madenciliği tekniklerinden yararlanılmıştır. Bu amaca yönelik;

1. Oltalama web sitelerinin tespitinde kullanılmak üzere, yeni bir veri seti oluşturulmuştur.

2. Oltalama veri setleri için kullanılan öznitelikler ve özniteliklerin elde edildiği web siteleri vurgulanmıştır.

3. Makine öğrenmesi tekniklerinde kullanılan algoritmaların oltalama saldırılarının tespitindeki başarım oranları karşılaştırılmıştır.

4. Oluşturulan veri seti, açık kaynak kodlu UCI (Muhammed vd., 2015a) ve Mendeley (Tan, 2018) veri setleri ile karşılaştırılmıştır.

Bu tez çalışmasının Giriş ve Amaç bölümünde, çalışmanın amacı verilmiştir. Literatür Araştırması bölümünde, konu ile ilgili yapılan çalışmalar ile kullandıkları yöntemlerden bahsedilmiştir. Oltalama Web Siteleri bölümünde, oltalama web siteleri ile ilgili genel bilgiler verilmiştir. Materyal ve Yöntem bölümün makine öğrenmesi ve veri setleri ve oltalama web sitelerini tespit etmek için toplanan yeni veri setinin özniteliklerinden bahsedilmiştir. Bulgular ve Tartışma bölümünde deney sonuçları ve sonuçların değerlendirilmesi yapılmıştır. Sonuç ve Öneriler bölümünde, bu tez çalışmasında alınan sonuçlar ve öneriler belirtilmiştir.

2. LİTERATÜR ARAŞTIRMASI

Oltalama saldırılarını inceleyen ve oltalama saldırılarının farklı yönlerini ele alan birçok çalışma yapılmıştır. Kaytan (2016), yaptığı tez çalışmasında, siber saldırıların çeşitlerinden ve tehlikelerinden bahsetmiş ve web tabanlı oltalama saldırılarını tespit etmeye çalışmıştır. Yapay Sinir Ağlarını ve Aşırı öğrenme makinesi kullanarak oltalama saldırı web sitelerini tespit etmeye çalışmıştır. (Muhammed vd., 2015a) kaynak veri seti, çalışmasında açıklanmış ve deneysel çalışmada kullanılmıştır.

Büber (2017), yaptığı tez kapsamında oltalama saldırılarının verdiği zararlardan bahsetmiştir. Oltalama web sitelerini doğal Dil İşleme tekniğini kullanarak tespit etmeye çalışmıştır. Kullanmak istediği yöntem için yasal ve oltalama URL'leri kullanılarak veri kümesi oluşturmuştur.

Bayraktar (2019), yaptığı çalışmada, veri madenciliği ve makine öğrenmesi yöntemlerine değinmiştir. Rastgele Orman ve Aşırı Öğrenme tekniklerinden yararlanılarak oltalama web siteleri tespit edilmeye çalışılmıştır. (Muhammed vd., 2015a) tarafından oluşturulan açık kaynak veri seti kullanılmıştır.

Turhanlar (2019), Türkçe dili ile yapılan oltalama saldırılarını tespit etmeye çalışmıştır. Çalışmasında e-posta, kısa mesaj ve sosyal medya üzerinden yapılan oltalama web siteleri incelenmiştir. Makine öğrenmesi kullanılarak oltalama saldırılarında kullanılan metinler üzerinden, oltalama saldırıları tespit edilmiştir.

Oltalama saldırısı web sitelerini tespit ederken, araştırmacılar çalışmalarını saldırıyı tespit etmek, önlemek ve etkilerini azaltmaya yöneltmiştir. Oltalama saldırılarını tespit etmek için birçok çalışmanın olmasından dolayı yapılan çalışmalar, farklı kategorilere bölünerek incelenmiştir. Saldırıları önlemek ve etki alanını azaltmak için, Beyaz-Kara Liste yöntemi ve Sezgisel yöntemler kullanılmıştır. Bu yöntemler sayesinde, dikkatsiz veya bilinçsizce, oltalama web sitesini ziyaret etmek üzere olan kullanıcının, oltalama web sitesini

ziyaret etmesi engellenir veya ziyaret ettiği web sitesi hakkında uyarılır. Yukarıda bahsedilen, zararlı web sitesine erişimin engellenmesi veya kullanıcının uyarılmasındaki temel amaç, kullanıcılar açısından oltalama web sitelerinin oluşturduğu maddi ve manevi zararı en aza indirmek ve dolaylı olarak, kurum ve kuruluşlara ait yasal gerçek web sitelerinin güvenli bir şekilde hizmet vermesini sağlamaktır.

2.1. Beyaz-Kara Liste Yöntemleri

Beyaz-Kara liste metodu, bir web sitesini yasal veya oltalama web sitesi olarak sınıflandırmak için, yasal ve oltalama olmak üzere iki farklı türde etiketlenerek sınıflandırılmış web sitesi listelerini kullanılır (Mohammad, Thabtah ve McCluskey, 2015a). Beyaz listeler yasal olarak sınıflandırılmış web sitelerinden, kara listeler ise oltalama olarak sınıflandırılmış web sitelerinden oluşmaktadır. Beyaz-Kara liste yöntemleri için en iyi örnekler Google Safe Browsing API ve Netcraft AntiPhishing Tool'dur (Google Safe Browsing API,2007; Netcraft AntiPhishing Toolbar, 1994). Beyaz-Kara liste yöntemi ile listede olan bir web sitesi %100 tespit edilir. Ama bu yöntem sıfır gün saldırılarında hiçbir etkisi yoktur. Sıfır gün saldırıları saldırganlar tarafından daha önce kullanılmamış yeni saldırıdır. Ayrıca, oltalama saldırısı tespit etme oranı listelerin güncelleme hızına bağlıdır.

2.2. Sezgisel Yöntemler

Liste metotlarının bilinmeyen veya yeni oluşturulmuş oltalama web sitelerinin tespitinde başarısız olmaları, araştırmacıları sezgisel yöntemlere yönlendirmiştir (Qabajeh vd., 2018). Sezgisel yöntemler, oltalama saldırısı olarak tanımlanmış web sitelerini inceleyerek daha önce bilinmeyen bir web sitesini yasal ya da oltalama web sitesi olarak sınıflandırmak için kullanılır. Sezgisel tabanlı çalışmalar, içerik tabanlı, kural tabanlı, veri madenciliği ya da makine öğrenmesi tabanlı ve hibrit olmak üzere, alt sınıflara ayrılabilmektedir.

2.2.1. İçerik tabanlı yöntemler

İçerik tabanlı oltalama web sitesi tespit sistemleri web sitesinin metin veya görsel yapısına inceleyerek web sitelerini yasal veya oltalama olarak sınıflandırır. (Y. Zhang vd., 2007), (Wardman vd., 2011) ve (Ramanathan ve Wechsler, 2013) çalışmalarında, web sitesi içerisinde kullanılan metinleri oltalama saldırılarını tespit etmek için kullanmışlardır. (Li vd., 2013) ve (Mao vd., 2017) oltalama web sitelerini görsel yapılarına göre tespit etmişlerdir. Ayrıca, (H. Zhang vd., 2011) metin ve görsel benzerliği birleştirerek oltalama web sitelerini tespit etmiştir.

2.2.1.1. Metin benzerlik tabanlı yöntem

Web sitelerinin metin benzerliğini inceleyen araştırmacılar, web sitesini oluşturan Hiper Metin İşaret Dili (HTML) kaynak kodu ve İnternet Kaynak Belirteci (URL) kısımlarındaki metinleri inceleyerek oltalama web sitelerini sınıflandırmıştır. URL'nin içerdiği kelimeler ve kelime grupları, öznitelik çıkarımında kullanılmaktadır. HTML kaynak kodunun içerdiği paragraflar, tıklanabilir link olan kelimeler ve metinler, düğme metinleri vb. metin içerikleri öznitelik çıkarımında kullanılmıştır.

(Zhang Y. vd., 2007) çalışmasında, içerik tabanlı bir yaklaşım olan CANTINA'da oltalama web sitelerini tespit etmek için Terim Frekansı - Ters Doküman Frekansı (TF-IDF) yöntemini kullanmıştır. TF-IDF bilgi erişim ve metin madenciliği alanlarında sıklıkla kullanılan bir algoritmadır. Bir kelimenin bir arşiv içerisindeki dokümanda ne kadar önemli olduğunu ölçmek için TF-IDF bir ağırlık değeri sağlamaktadır. TF-IDF genellikle belge karşılaştırma, sınıflandırma ve büyük arşivlerden belge getirme işlemlerinde kullanılmaktadır. TF-IDF algoritmasını kullanarak, sözlüksel imzalar oluşturulmuştur. Bir web sitesinin kaynağını etkili bir şekilde ifade eden sözlüksel imzalar, 5 kelimeden oluşmaktadır. CANTINA öncelikle her sayfanın TF-IDF skorunu hesaplar, ardından en yüksek TF-IDF değerine sahip 5 kelimeden oluşan sözlüksel imza arama motorunda aratılır. Web sitesinin alan ismi, arama motorundan gelen ilk N arama sonucu içerisinde ise sayfa yasal web sitesi olarak kabul edilir. Veri kümelerinde 100 adet PhishTank'den toplanmış oltalama web sitesi ve 100 adet 3Sharps'dan toplanmış yasal web sitesi yer almaktadır

(PhishTank, 2006; 3Sharps, 2006). Deneylerinde SpoofGuard ve Netcraft oltalama saldırısı tespit araçları ile CANTINA karşılaştırılmıştır (Stanford Applied Crypto Group, 2006; Netcraft, 1994). Yaptıkları deneylerde N sayısı için 30 eşiğinin daha iyi sonuç verdiği görülmüştür. Çalışmalarında oltalama web sitelerini, geliştirdikleri yöntem ile etkili bir şekilde tespit etmişlerdir.

(Wardman vd., 2011) çalışmasında, oltalama saldırılarını tespit etmek için içerik tabanlı bir araştırma yapılmıştır. Çalışmalarında 5 farklı dosya eşleştirme yöntemi kullanılarak sınıflandırma yapmışlardır. Ana dizin eşleşmesi yöntemi her bir dosyaya eşsiz bir özüt değeri vererek dosya benzerliklerini ölçmektedir. Derin MD5 eşleştirme yöntemi ile web siteleri için benzerlik katsayısı hesaplanır. Derin MD5 eşleştirme yöntemi web sitelerinin gizlenmiş ve dinamik yapıların oluşturduğu problemleri çözmektedir. Oltalama web siteleri gizleme ve dinamik yapı için Basamaklanmış Stil Katmanları (CSS), PHP: Hypertext Preprocessor (PHP) ve JavaScript dosyalarını kullanaraktan gerçek ve yasal web sitesi gibi anlaşılmasını sağlar. Benzerlik oranını hesaplamak için *Jaccard*, *Kulczynski2* ve *Simpson* katsayıları kullanılmıştır. Dosyalar arasındaki farkı anlamak için satırları karşılaştıran *diff* algoritması kullanılmıştır. İçerik tarafından tetiklenen parçalı özüt yöntemi olan *ssdeep* kullanılarak dosyaların birbirine benzerliği ölçülmüştür. *Sözdizimsel Parmak İzi* yöntemi kullanılarak web sitelerinin altyapısal parçaları karşılaştırılarak, iki web sitesinin benzerliği tespit edilmiştir. Çalışmada veri seti oluşturmak için UAB Phishing Data Mine kullanılarak, 49840 oltalama web sitesi URL'si toplanmıştır (UAB, 1994). Veri kümesi 5 farklı yöntem kullanılarak test edilmiştir. Bazı tekniklerin %90 üzerinde başarı sağladığı gözlemlenmiştir.

Yapılan başka bir çalışmada doğal dil işleme ve makine öğrenmesi yöntemlerinden yararlanılmıştır (Ramanathan ve Wechsler, 2013). Yaptıkları çalışmada insan isimleri, organizasyonlar ve konum bilgileri Koşullu Rastgele Alanlar (CRF) yöntemi ile toplanmıştır. Gizli içerikleri toplamak için Gizli Dirichlet Ayırımı (LDA) yöntemi kullanılmıştır. LDA sayesinde mesajın içeriği kelime çantası modeli yapısı ile bulunmuş, aynı anlama gelen kelimeler kontrol edilerek yanlış yazılmış olan kelimeler bulunmuştur. Oltalama e-posta veri kümesi için yasal e-postaları SpamsAssassin PublicCorpus'dan, oltalama e-postaları ise PhishingCorpus ve Spam Archive'den toplanmıştır (SpamsAssassin,

2004; PhishingCorpus, 2006; untroubled.org, 1998). Oltalama URL veri seti için PhishTank arşivleri kullanılmıştır. PhishTank'den toplanan URL'ler kullanılarak yapılan web taraması sonucunda 2011 ve 2011-2012 yılları arasında toplanan 2 adet web sitesi veri kümesi oluşturulmuştur. Sınıflandırma için AdaBoost sınıflandırma yöntemi kullanılmıştır. Test veri kümesini ayırmak için 10 katlamalı çapraz doğrulama yöntemi kullanılmıştır. Veri kümeleri karşılaştırılınca Spam Archive'den toplanan e-postaların daha yüksek tespit etme oranına sahip olduğu görülmüştür.

2.2.1.2. Görsel benzerlik tabanlı yöntem

Görsel benzerlik tabanlı sistemler, web sitelerinin CSS kaynak kodlarını ve web sitesinin ekran görüntülerini kullanarak, web sitelerini sınıflandırmaktadır. CSS kaynak kodları bir web sitesinin görsel stilini belirlemek için kullanılır. Web sitesini oluşturan elementlerin görsel görünümünü değiştirir. Bir web sitesinin ekran görüntüsü kullanıcının web sitesini arama motoru ile görüntülediği durumudur.

Yapılan çalışmada yarı kontrollü Transdüktif Destek Vektör Makinesi (TSVM) kullanılarak oltalama web siteleri sınıflandırılmıştır (Li vd., 2013). Çalışmada web sitesinin ekran görüntüsünün resmi kullanılarak öznitelikler toplanmıştır. Web sitesinin ekran görüntüsünün gri seviye histogramı, renk histogramı ve alt grafiklerin merkezlerinin resmin üzerindeki konumu görsel öznitelikleri oluşturmaktadır. Görsel özniteliklerin yanında web sitesinin Belge Nesnesi Modeli (DOM) öznitelikleri de kullanılmıştır. Çalışmalarında Destek Vektör Makinesi (SVM) ve TSVM karşılaştırılmıştır. Deney sonucunda TSVM ve görsel öznitelikler daha iyi performans vermiştir. Veri kümesi için oltalama web siteleri PhishTank, yasal web siteleri ise Google arama motorundan toplanmıştır (Google Inc., 1998).

Yapılan diğer bir çalışmada oltalama saldırılarını tespit etmek için Google Chrome arama motoruna eklenti oluşturulmuştur (Mao vd., 2017; Google Inc., 2008). Oluşturdukları eklenti, web sitelerinin görsel benzerliklerini kullanarak oltalama web sitelerini tespit etmektedir. Çalışmada web sitesinin ekran görüntüsü yerine CSS dosyası üzerinde çalışılmıştır. Oltalama web siteleri tarafından kullanılan CSS dosyalarının çoğunlukla hedef

web sitesinin CSS dosyasının kopyalanması ile oluşturulduğu ortaya çıkmıştır. Çalışmalarında öznitelik olarak CSS dosyasındaki oldukça küçük boyutlu, gizli ve görüntülenmeyen elementler seçilmiştir. Hedef alınan site ve oltalama web sitesi benzerlik skoru sayesinde karşılaştırılmıştır. Veri kümesi için hedef alınan yasal siteler ile ilgili oltalama web siteleri PhishTank'den alınmıştır.

2.2.1.3. Metin ve görsel benzerlik tabanlı yöntem

Web sitelerinin hem metin hem de görsel içeriğini inceleyen çalışmalar literatürde mevcuttur. İki farklı öznitelik yapısının birleştirildiği bu çalışmalarda, özniteliklerin birbirleri ile olan etkileşimleri önemli rol oynar.

Metin ve görsel benzerliğin birlikte kullanıldığı çalışmada, metin ve görsel içeriklerin benzerliklerini karşılaştırarak oltalama web sitelerini tespit etmeye çalışılmıştır (Zhang H. vd., 2011). Metin sınıflandırması için, Naïve Bayes (NB) kullanılarak, web sitesinin oltalama web sitesi olma olasılığı hesaplanmıştır. Görsel sınıflandırma için, Yer Değiştiricinin Mesafesi (Earth Mover Distance) yöntemi kullanılarak görsel benzerlik ölçülmüştür. İki farklı yapıda olan metin ve görsel verileri birleştirmek için ağırlık yaklaşımı kullanılmıştır. Çalışmalarında, yasal siteleri bulmak için, 26 anahtar kelime Google arama motoru vasıtasıyla taranarak, 10272 web sitesi toplanmıştır. Oltalama web siteleri ise PhishTank'den toplanmıştır. Çalışmalarında e-ticaret, online bankacılık gibi kişisel ve finansal bilgilerin kullanıldığı 8 farklı web sitesi baz alınmıştır. Deneysel çalışma sonucunda, metin ve görsel verilerin Bayes yöntemi ile birleştirildiği yöntem daha iyi başarım göstermiştir.

2.2.2. Kural tabanlı yöntemler

Oltalama web sitelerinin sebep olduğu zararlardan bahsedilen çalışmada kural tabanlı yöntem benimsenmiştir (Mohammad, Thabtah ve McCluskey, 2012). Kara listeyi baz alan korunma yollarının, sıfır gün saldırılarında koruma görevini yerine getirememesinden dolayı, sezgisel tabanlı yöntemlerin daha başarılı koruma sağlayacağı belirtilmiştir. Araştırmalarında, veri madenciliği yöntemi kullanarak, oltalama web siteleri tespit edilmeye

çalışılmıştır. Web sitelerinden topladıkları 17 farklı özniteliği 4 farklı kategoriye ayırmışlardır (Mohammad, Thabtah ve McCluskey, 2014a). Veri kümesini oluşturmak için 2500 adet oltalama web sitesi, PhishTank arşivlerinden toplanarak, oltalama saldırıları hakkında bilgi edinmek için Millersmiles kaynağından yararlanılmıştır (Millersmiles, 2003). Yahoo! kullanılarak 450 adet yasal web sitesi toplanmıştır (Yahoo Inc., 1994). Veri kümesini dengeli oluşturmak için deney aşamasında 450 oltalama web sitesi rastgele seçilmiştir. Çalışmalarında oltalama web sitelerini oluşturmak için 'URL isteği' ve 'alan yaşı', ve 'HTTPS ve SSL' öznitelikleri en çok kullanılmıştır. Deney aşamasında, 17 farklı özniteliğe sahip veri kümesi C4.5, RIPPER, PRISM ve İlişkilendirmeye Dayalı Sınıflandırma (CBA) algoritmaları ile test edilmiştir. Deney sonucunda C4.5 algoritması en düşük hata oranına sahiptir. Veri kümesinden oltalama web sitelerinin tespitinde önemsiz etkiye sahip 8 öznitelik Ki-Kare yardımı ile çıkarılmıştır. Veri kümesinde kalan 9 farklı öznitelik ile deney yapıldığında bütün algoritmaların hata oranı düşmüş ve CBA algoritması en düşük hata oranına sahip olduğu gözlemlenmiştir.

Diğer çalışmalarında ise, yapay sinir ağları kullanılmıştır (Mohammad, Thabtah ve Mccluskey, 2014b). Gizli aşama sayısı düşük tutulmaya çalışılmıştır. Gizli aşamalardaki tüm düğümler kullanılarak, performans arttırılmaya çalışılmıştır. Hazırlanan veri kümesi UCI Machine Learning Repository web sitesinde yayınlanmıştır (Mohammad, Thabtah ve McCluskey, 2015b).

Yapılan çalışmada, oltalama web sitesi tespit etmek için kural tabanlı yöntemler kullanılmıştır (Basnet vd., 2011). Çalışmalarında öncelikle, oltalama web sitelerinin çalışma taktikleri incelenerek kurallar çıkarılmıştır. Ayrıca daha önceki çalışmalarda makine öğrenmesi yöntemleri ile çıkarılan öznitelikler kullanılmıştır. Arama motoru, kırmızı bayraklı kelime, gizleme, kara liste, itibar ve içerik tabanlı olmak üzere gruplara ayrılmış 15 öznitelik oluşturulmuştur. Kırmızı bayraklı kelimeler eğitim kümesi içerisinde en çok gözlenen 62 kelimeden oluşur. Veri kümesini oluştururken oltalama web siteleri PhishTank'den, yasal web siteleri Yahoo!' dan toplanmıştır. Veri kümesi test aşaması için 10 katlamalı çapraz doğrulama yöntemi ile ayrılmıştır. Deneysel aşamada C4.5 ve Logistic Regression (LR) algoritmaları kullanılmıştır. Deneyin sonuçları incelenince C4.5 ve LR algoritmalarının verdiği sonuçların arasındaki farkın oldukça önemsiz olduğu görülmüştür.

(Moghimi ve Varjani, 2016) çalışmasında, internet bankacılığına yönelik yapılan oltalama saldırıları engellenmeye çalışılmıştır. Yapılan çalışmada kural tabanlı bir yöntem oluşturulmuştur. Kullanmak istedikleri yöntem için daha önceden yapılan çalışmalarda kullanılan 50'den fazla öznitelik içerisinden 5 tanesi seçilmiştir. Bunlar; URL'nin İnternet Protokolü (IP) adresi içermesi, web sitesinin Güvenli Soket Katmanı (SSL) sertifikasına sahip olması, URL'nin içerdiği nokta sayısı, URL uzunluğu ve kara liste kelimelerini içermesidir. Bu özniteliklere ek olarak, web sitesinin DOM özellikleri kullanılarak güvenli sayfa linki, JavaScript dosyaları, stil sayfası dosyaları ve resimler seçilerek Levenshtein Distance (LD) yöntemine tabi tutularak ziyaret edilerek istenen web sitesi ile hesaplanarak dört adet öznitelik elde edilir. Karar ağacı ile SVM' nin oluşturduğu Destek Vektör Makinesi - Karar Ağacı (SVM-DT) algoritması ile sınıflandırma yapılır. Google Chrome arama motoruna eklenti olarak hazırlanan sistem için yasal siteleri Yahoo! Dizin hizmetinden, oltalama web siteleri ise PhishTank'den alınarak oluşturulan veri kümesi ile test edilmiştir.

2.2.3. Makine öğrenmesi ve veri madenciliği tabanlı yöntemler

(Fette vd., 2007) çalışmasında, oltalama e-postalarını makine öğrenmesi yöntemiyle tespit edilmeye çalışılmıştır. Oltalama saldırılarının internetin yaygınlaşması ile artmasından ve saldırganlar tarafından gerçek web sitelerin kolaylıkla kopyalanmasından bahsedilmiştir. Çalışmalarında PILTER adlı oltalama saldırılarından korunmak için oluşturulmuş Spoofguard ve Netcraft araç çubukları ve e-posta filtreleme aracı olan SpamAssassin'den bahsedilmiştir (Apache, 2003). Yasal ve oltalama e-posta olmak üzere 2 sınıflı bir sınıflandırma yapmışlardır. E-posta sınıflandırması için gerekli 10 öznitelik belirlenmiştir. Çalışmalarında, ayrıca web siteleri içinde 3 tane öznitelikden bahsedilmiştir. Veri kümesi oluştururken yasal e-postalar SpamAssassin'den alınmıştır. Oltalama e-postaları PhishingCorpus'dan alınmıştır (PhishingCorpus, 2006). Öznitelikler ham veri kümesinden çıkarıldıktan sonra eğitim ve test aşaması için 10 katlamalı çapraz doğrulama yöntemi ile ayrılmıştır. PILTER, Rastgele Orman (RF) algoritmasını kullanmaktadır. SVM, kural tabanlı algoritmalar ve Bayes gibi farklı sınıflandırma algoritmaları ile yapılan deneyde sınıflandırma algoritmaları arasında önemli bir fark görülmemiştir. Deneysel aşamada aynı veri kümesi kullanılarak PILTER ve SpamAssassin karşılaştırılmıştır. Çalışmanın

sonucunda PILTER sadece spam filtresi kullanılarak yapılan bir sınıflandırmadan daha başarılı sonuçlar vermiştir.

Bulanık veri madenciliği teknikleri kullanılarak, internet bankacılığını hedef alan oltalama web sitelerini tespit etmeye çalışmışlardır (Aburrous vd., 2009; Aburrous vd., 2010a). Bulanık veri madenciliği tarafından kullanılacak kurallar RIPPER, PART, Prism, C4.5 ve CBA veri madenciliği sınıflandırma teknikleri kullanılarak çıkarılmıştır. Veri kümesini PhishTank'den topladıkları 606 oltalama web sitesi ve APWG arşivlerindeki 1006 oltalama web sitesi ile oluşturulmuştur (APWG, 2003). Veri kümesi 6 gruba ayrılmış 27 öznitelik içermektedir. Test aşamasında veri setini ayırmak için 10 katlamalı çapraz doğrulama yöntemi kullanılmıştır. Bulanık veri madenciliği yöntemi, 3 katmandan oluşmaktadır. Yapılan deney sonucunda oltalama web sitelerinin tespitinde URL ve Alan Bilgisi ile Güvenlik ve Şifreleme gruplarına ait özniteliklerinin önemi ortaya çıkarılmıştır. Sayfa Stili ve İçerik ile Sosyal İnsan Faktörü gruplarına ait özniteliklerin oltalama web sitelerinin tespitinde önemsiz bir etkisi olduğu gözlemlenmiştir. Araştırmalarında, Sosyal İnsan Faktörü özniteliklerinin nasıl toplandığı hakkında herhangi bir bilgi verilmemiştir. Başka bir çalışmalarında ise veri madenciliği ve ilişkisel sınıflandırma algoritmaları kullanarak daha önceki çalışmalarında oluşturdukları veri kümesini incelemişlerdir (Aburrous vd., 2010b). C4.5, PART, RIPPER, Prism, CBA ve Çok Sınıflı Sınıflandırma (MCAR) algoritmalar deney aşamasında kullanılmıştır (Thabtah vd., 2005). Deney sonucunda MCAR algoritmasının daha düşük hata oranına sahip olduğu gözlemlenmiştir.

Yapılan bir diğer çalışmada, öznitelik seçimi için kullanılan sarmal ve filtre ölçüm öznitelik seçim yöntemlerinde kullanılan kesme noktasının güvenilir olmamasından dolayı Birikimli Dağılım Fonksiyon gradyan (CDF-g) algoritmasını sunmuşlardır (Chiew vd., 2019). CDF-g algoritması otomatik olarak kesme derecesini bulmaktadır. Bu yöntem sayesinde etkili ve özlü temel öznitelik kümesi oluşturula bilinmiştir. Temel özniteliklerin kararlılığını arttırmak için, CDF-g algoritması hibrit bir sisteme yerleştirilmiştir. Hibrit Grup Öznitelik Seçimi (HEFS) adını verdikleri sistem, Veri Bozulması (Data Perturbation) ve İşlev Bozulması (Function Perturbation) tekniklerini içermektedir. Veri bozulması yöntemi, aynı öznitelik seçimi yöntemini veri kümesinin birden fazla alt kümesine uygularken, işlev bozulması yöntemi farklı öznitelik seçim yöntemlerini veri kümesinin aynı kümesine

uygular. Veri Bozulması ve İşlev Bozulması yöntemleri kullanılarak hibrit bir sistem oluşturulmuştur. Oltalama saldırılarını tespit etmek için oluşturdukları veri kümesi için web sitesinin URL ve HTML kaynak kodu içerisinden öznitelikler çıkarılmıştır. PhishTank ve OpenPhish arşivlerinden 5000 oltalama web sitesi toplanmıştır (OpenPhish, 2020). Alexa ve Common Crawl arşivlerinden 5000 yasal web sitesi toplanmıştır (Alexa, 1996; The Common Crawl Foundation, 2020). Hazırlanan web sitesi Waikato Environment for Knowledge Analysis (WEKA) uzantılı dosya olarak kullanıma sunulmuştur (University of Waikato, 1993; Tan, 2018). Sınıflandırıcı olarak WEKA da bulunan SVM, NB, C4.5, RF, JRip ve PART kullanılmıştır. Öznitelikleri derecelendirmek için Bilgi Kazancı (IG), Relief-F, Symmetrical Uncertainty, CS, GR ve Pearson Ürün Momenti Korelasyon Katsayısı (PCC) filtreleri kullanılmıştır. Yapılan deneyler sonucunda HEFS yöntemi ile seçilen öznitelikler Dağınık Orman (RF) algoritması ile sınıflandırıldığında diğer algoritmalara göre daha iyi sonuç vermiştir.

Yapılan çalışmada gerçek zamanlı web sitelerini tespit etmek için web sitesinin URL'sini kullanmışlardır (Sahingoz vd., 2019). URL yardımı ile kelime vektörü, doğal dil işleme tabanlı ve hibrit öznitelik kümeleri oluşturulmuştur. Öznitelik kümeleri Karar Ağacı, AdaBoost, K-Yıldız (KS), k-en Yakın Komşu (KNN) (k =3), RF, Sıralı Minimum Optimizasyon (SMO) ve NB makine öğrenimi algoritmaları ile test edilmiştir. Veri kümesini oluşturmak için, PhishTank arşivlerinden 37175 adet oltalama web sitesi URL'si toplanmıştır. Ayrıca, Yandex Search API kullanılarak 36400 adet yasal web sitesi URL'si toplanmıştır (Yandex, 2020). Öznitelik kümelerini oluşturmak için URL'ler ön işleme tabi tutulmuştur. İlk aşamada, marka ismi ve anahtar kelime kontrolü yapılmıştır. İkinci aşamada, URL'in rastgele kelime içerip içermediği Markov Zincir Modeli (MCM) kullanılarak bulunmuştur. Üçüncü aşamada, kelimeler alt kümelerine bölünerek, kelime listesine eklenmektedir. Son aşamada ise, bilerek yazım hatası yapılarak, oluşturulan URL'leri tespit etmek için LD algoritması kullanılmıştır. Doğal Dil İşleme tabanlı öznitelikler, kelime uzunluğu, öznitelik çıkarım aşamasında bulunan kelime sayaçları ve özel karakter içerip içermediğine bakılarak toplanmıştır. Öznitelik kümeleri ayrı ayrı 7 farklı makine öğrenmesi algoritması ile test edilmiştir. Doğal dil işleme yöntemleri ile oluşturulan öznitelik kümesi RF algoritması %97,98 doğruluk oranı ile en iyi sonuca sahiptir.

2.2.4. Hibrit tabanlı yöntem

Yapılan çalışmada PhiDMA adıyla Google Chrome eklentisi ara yüzüne sahip bir model oluşturularak oltalama web siteleri tespit edilmeye çalışılmıştır (Sonowal ve Kuppusamy, 2020). Oluşturdukları sistem 5 katmandan oluşmaktadır. İlk katmanda, beyaz liste filtre yapısı kullanılarak erişilmeye çalışılan URL adresinin yasal olup olmadığına bakılmıştır. İkinci katmanda, URL'nin sahip olduğu özelliklere bakılmıştır. Bu özellikler; URL'nin IP adresi içermesi, domain yaşı, URL uzunluğu, URL'nin içerdiği nokta sayısı ve şüpheli bir URL olup olmamasıdır ('-' ve '@' sembolleri içermesi). Üçüncü katmanda, URL adresi, sitenin sözlük imzası ile arama motorunda yapılan arama sonucunda herhangi bir link elde edilmesine bakılmıştır. Dördüncü katmanda, üçüncü aşamada bulunan linkler ile ziyaret edilmek istenen web sitesi katar eşleştirme algoritmaları ile test edilmiştir. Son katmanda ise, web sitesinin ulaşılabilirlik puanı hesaplanmıştır. Yaptıkları çalışmada Phishload ve PhishTank arşivlerinden yararlanılmıştır (Maurer, 2012). Çalışma, karşılaştırılan yöntemlerden daha başarılı sonuç göstermiştir.

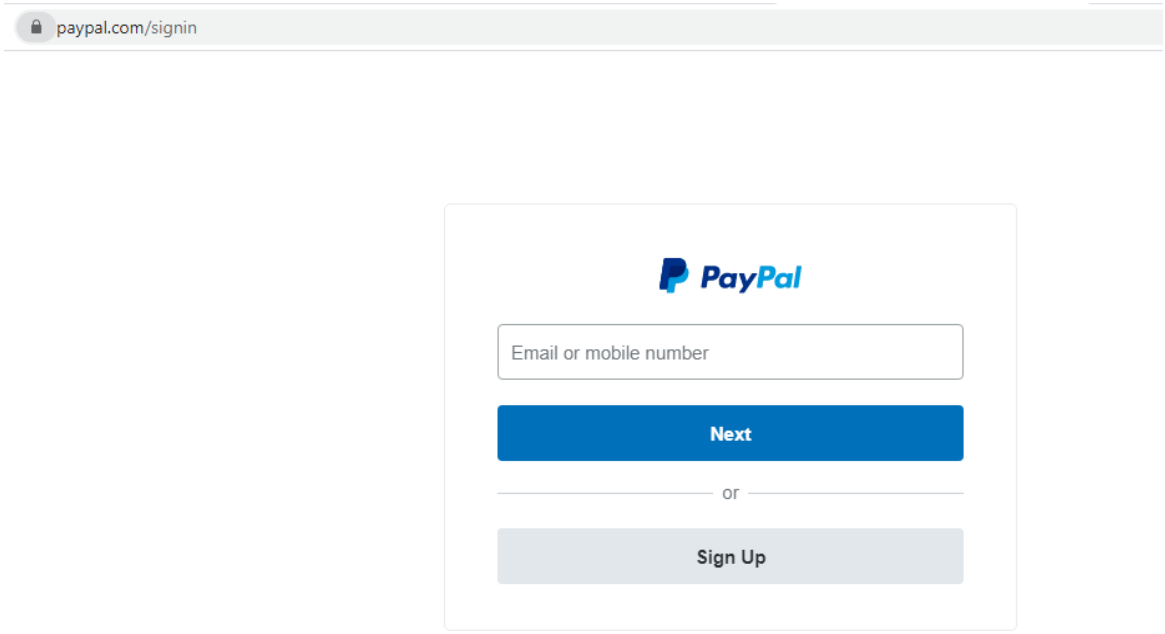
3. OLTALAMA WEBSİTELERİ

Son yıllarda, hizmetlere hızlı bir şekilde ulaşılmak istendiğinden dolayı, şirketler ve kuruluşlar sağladıkları hizmetleri sanallaştırmaya ve online olarak kullanıcılara ulaştırmaya başlamıştır. Sağlanan bu hizmetler için, kullanılan kişisel ve finansal bilgiler günümüzde saldırganların ilgisini çekmektedir. Saldırganlar bu bilgilere erişmek için, oltalama saldırılarını kullanmaktadır. Oltalama saldırıları, teknik ve sosyal mühendislik yöntemlerini kullanarak kurbanların bilgilerini çalmaya yarar. Oltalama saldırıları sonucunda, kurbanlar kişisel ve finansal bilgi kaybı yaşanmaktadır. Oltalama saldırılarından sadece kurbanlar değil, şirketler ve kuruluşlar itibar ve maddi kayıp yaşamaktadır.

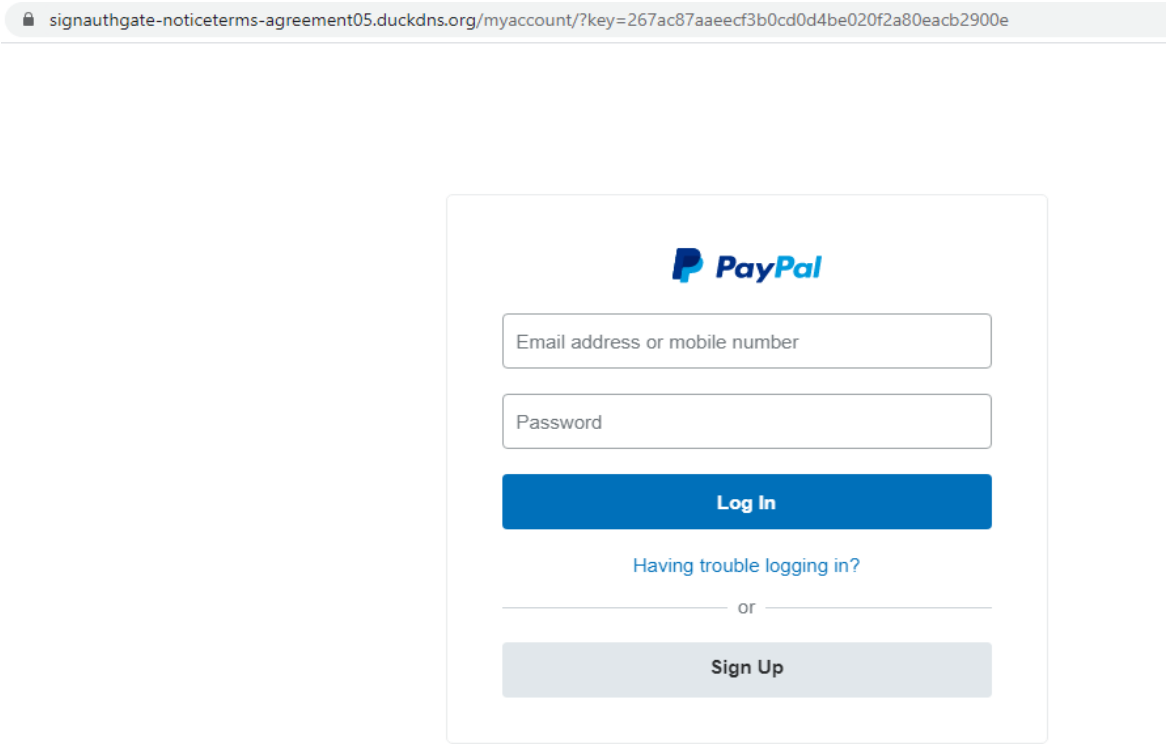
Oltalama saldırılarının sosyal mühendislik kısmında oltalama e-posta ve kısa mesaj yer alır. Oltalama e-posta ve kısa mesajlar, kurbanları kandırmak ve aldatmak için insanları psikolojik olarak çekecek içeriklere sahiptir. İçerikler genellikle bilgi güncelleme, kişiye özel kampanya, çekiliş hakkı vb. türde konular hakkındadır. İnsanların ilgisini çeken bu konular, oltalama saldırıları hakkında bilgili insanları bile kurban konumuna getirebilmektedir. Oltalama e-posta ve kısa mesaj hala popüler olmasına rağmen, kullanıcı kitlesi fazla olan sosyal medya platformları da, oltalama saldırısı için kullanılan mesaj iletiminde önemli rol oynamaktadır. Mesajlar sadece saldırganlar tarafından gönderilmeyip, ayrıca farkında olmayan ya da çok bilinçli olmayan, kurbanların yakın çevresi tarafından iletilebilmekte ve bu nedenle oltalama saldırısı mesajları kitleler arasında hızla yayılabilmektedir.

Oltalama saldırılarının teknik kısmında web siteleri yer almaktadır. Web siteleri yasal web sitelerinin kopyalanarak, benzerinin oluşturulması yöntemi ile yapılmaktadır. Hazırlanan oltalama web siteleri ve yasal web siteleri arasında çok az fark olduğu için kurbanlar girdikleri site hakkında şüphe duymuyorlar. Oltalama web siteleri görsel olarak kopyalandığı yasal web sitesi ile arasındaki fark yok sanacak kadar az olmasına rağmen, ziyaret edilen web sitesinin URL'si incelenerek yasal web sitesi olup olmadığı anlaşılabilmektedir. Yasal web sitesi için örnek verilen web sitesinin ekran görüntüsü Şekil 3.1.'de verilmiştir ve Şekil 3.2.'de yasal web sitesinden kopyalanarak oluşturulan oltalama

web sitesi ekran görüntüsü verilmiştir. Şekil 3.1. ve Şekil 3.2. incelendiğinde 2 web sitesinin görsel olarak benzer olduğu ve farklı URL kullandığı görülmektedir. Oltalama web siteleri ve kopyalandıkları sahte web siteleri arasındaki benzerlik, oltalama saldırılarının tehlikesini göstermektedir.



Şekil 3.1. Yasal web sitesi ekran görüntüsü



Şekil 3.2. Oltalama web sitesi ekran görüntüsü

3.1. Oltalama Saldırıları Nasıl Başlar?

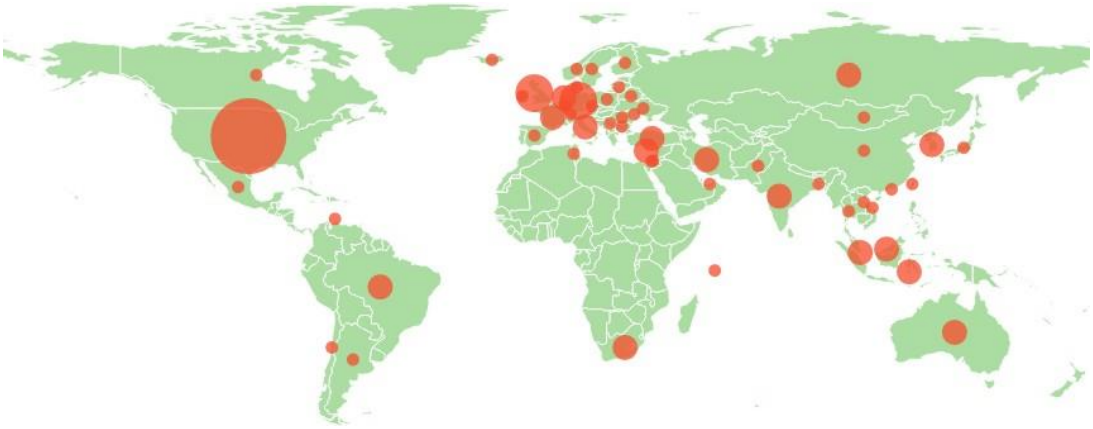
Oltalama saldırıları, saldırganlar tarafından hazırlanan sahte web siteleri üzerinden yapılmaktadır. Ancak, bu web sitelerine, arama motorları tarafından bulunan sonuçlar içinden erişilmez. Arama motorlarında, bir web sitesi arandığında, arama motoru tarafından en alakalı sonuçlar gösterilir. Günümüzde kullandığımız birçok küresel ve bölgesel hizmet sunan web siteleri arama motorları tarafından bilinmekte ve en alakalı sonuç olarak bu yasal siteler kullanıcıya verilmektedir. Arama motorları, en çok kullanılan, kullanıcılara etkili hizmet veren web sitelerini bilmesine rağmen oltalama saldırıları için etkili bir yöntem geliştirilememiştir (Chiew vd., 2018).

Oltalama saldırılarında, kilit noktaya sahip oltalama web siteleri, günümüzde e-posta, kısa mesaj ve sosyal medya aracılığı ile kurbanlara yönlendirilmektedir. Oltalama web sitelerine yönlendiren bu tarz mesajlara oltalama mesajları denir. Oltalama mesajları, oltalama saldırılarının başlangıç noktasıdır. Bu mesajlar kurbanları oltalama web sitesine yönlendirmek için kullanılır. Son yıllarda, kullanım oranının artmasıyla, sosyal medya uygulamaları üzerinden reklam verme veya daha önceden belirlenmiş kurbanlara sosyal medya üzerinden mesaj yollama yöntemleri ile oltalama web siteleri yayılmaktadır (Jakobsson, 2007). Sosyal medya üzerinden yapılan saldırılar, sınırlı sayıda katılımcılı kampanya veya indirim kuponu vb. içerikler ile kurbanların hızlı bir şekilde ilgisini çekebilmeyi hedeflemektedir. Günümüzde kullanılan sosyal medya uygulamaları, genellikle içerisinde mesaj paylaşma ve kısa mesaj alt yapısına sahip olduğundan, oltalama mesajları saldırgan tarafından belirlenen kurbanlar tarafından başka kurbanlara iletilebilir. Kurbanların da yeni kurbanlar oluşturmasından dolayı, oltalama saldırıları kısa süre içinde binlerce insanı etkileyebilmektedir.

3.2. Oltalama Saldırılarının Etkileri

Oltalama saldırıları internet kullanımının artması ile artış göstermektedir. APWG tarafından hazırlanan raporda bu saldırı tipinin azalmadığı ve her zaman kendisine yeni kurbanlar bulduğu gösteriliştir. Yapılan saldırıların çoğunlukla internet kullanımının fazla olduğu ülkelerde olduğu Şekil 3.3.'de görülmektedir. Şekil 3.3.'de 24 saatlik süre içinde

bildirilen oltalama web sitelerinin etkilediği ülkeler görülmektedir. OpenPhish tarafından verilen bu bilgi her 5 dakikada bir güncellenmektedir.



Şekil 3.3. OpenPhish oltalama saldırılarının dünya çapında yoğunluğu (OpenPhish, 2020)

APWG tarafından 2016 yılında hazırlanan raporu incelendiğinde, oltalama saldırılarının en çok perakende, finans, ödeme servisleri ve internet servis sağlayıcısı sektörlerinin etkilendiği görülmektedir (Çizelge 3.1.). APWG'nin 2020 yılında hazırladığı Çizelge 3.2.'deki en çok etkilenen sektörler incelendiğinde, oltalama saldırılarının etkilendiği sektörlerin değiştiği gözlemlenmektedir.

Çizelge 3.1. Oltalama Saldırısı Trend Raporu – 4Q 2016 Raporu En Çok Hedef Alınan Sektörler (APWG Reports, 2017)

Sektörler	**%**
Perakende/Hizmet	41,85
Finans	19,60
ISP	12,58
Ödeme Servisi	11,33
Multimedya	5,15
Sınıflandırılmamış	4,30
Sosyal Ağ	3,32
Devlet	1,31
Müzayede	0,24
Oyun	0,13
Sınıflandırılmış	0,08
Teslimat Hizmeti	0,07
Bayi	0,02
Eğitim	0,01

Çizelge 3.2. Oltalama Saldırısı Trend Raporu – 4Q 2019 Raporu En Çok Hedef Alınan Sektörler (APWG Reports, 2020)

Sektörler	**%**
SaaS/Web posta	30,80
Ödeme	19,80
Finansal Kuruluş	19,40
Sosyal Medya	6,80
e-ticaret/Perakende	5,40
Bulut Depolama/Dosya Barındırma	3,40
Telekom	3,30

Oltalama saldırılarının düzenli olarak değişmesi bu saldırılardan etkilenen sektörleri düzenli olarak değiştirmektedir. Çizelge 3.3.'de 24 saat içinde tespit edilen oltalama web sitelerinin etkilediği sektörler OpenPhish tarafından verilmektedir. Ayrıca Çizelge 3.4.'de 24 saat içinde en çok hedef alınan web siteleri verilmiştir.

Çizelge 3.3. OpenPhish en çok hedef alınan sektörler (OpenPhish, 2020)

Sektörler	**%**
Finans	38,4
Çevrimiçi/Bulut Servisi	14,6
Sosyal Ağ	12,1
Telekomünikasyon	8,9
e-ticaret	8,1
e-posta Sağlayıcısı	6,0
Devlet	5,1
Ödeme Servisi	4,9
Lojistik ve Kuryeler	1,1
Oyun	0,4
Diğer	0,04

Çizelge 3.4. OpenPhish en çok hedef alınan kuruluşlar (OpenPhish, 2020)

Kuruluş	**%**
Halifax Bank of Scotland	7.7
Facebook, INC.	7.0
Lloyds TSB Group	6.0
Bank of America	5.4
Office365	5.2
Amazon.com Inc.	4.8
Three UK	4.3
Chase Personal Banking	4.1
Outlook	4.0
Paypal Inc.	2.7

Son yıllarda, oltalama saldırılarından etkilenen farklı sektörlerin yüzdelerinin artması, bu sektörlerin insanlar tarafından popüler olarak tercih edilmesidir. Sosyal medya ve bulut alt yapılarının popüler olması yapılan saldırıların odak noktasının değişmesine neden olmuştur. Saldırganların sürekli olarak popüler web sitelerine yönelmesi, saldırılan web sitelerinin sürekli değişim içinde olması oltalama saldırılarının etkin bir şekilde önlenmesini engellemektedir.

4. MATERYAL VE YÖNTEM

Yapılan çalışmada makine öğrenmesi algoritmaları kullanılmıştır. Kullanılan algoritmaların çalışma prensipleri açıklanmıştır. Deney için gerekli veri kümesi hazırlamak için deneyde kullanılacak öznitelikler ve öznitelik grupları belirlenmiştir. Deney için gerekli ortam hazırlanmıştır.

4.1. Makine Öğrenmesi

Makine öğrenmesi, bilgisayarlara karmaşık veri yığınları içerisinden anlamlı örüntüleri algılama ve veriye dayalı karar verme becerisi kazandırma çalışmasıdır. Makine öğrenmesi algoritmaları istatistik, olasılık, mantık vb. matematiksel yöntemler kullanarak bilgisayarların insan iletişimi olmadan karar vermesini sağlamaktadır (Bishop, 2006). Makine öğrenmesi yöntemi insan faktörünü ve insandan kaynaklanabilecek hatayı kaldırmaya veya en aza indirmeye çalışır. Makine öğrenmesi birkaç sınıfa ayrılsa da en temel sınıflar denetimli öğrenme ve denetimsiz öğrenmedir (Alpaydin, 2020).

Denetimli öğrenme de verilen eğitim veri kümesi her örnek için çıktı değeri verilmiştir. Veri kümesinde, giriş ve çıkış değerleri arasında ilişki kurularak belli bir örüntünün bilgisayar tarafından tanınması sağlanır. Böylece geçmiş örneklerin giriş-çıkış bilgileri kullanılarak eğitilmiş model yardımıyla, gelecekte bilgisayara verilecek yeni verilerin bilgisayar tarafından doğru bir şekilde tahmin edilmesi veya tanınması sağlanmaktadır. Denetimli öğrenme en çok kullanılan makine öğrenmesi yöntemidir.

Denetimsiz öğrenme de verilen veri kümesinde, veriler hakkında herhangi bir bilgi veya birbirleri ile olan ilişkileri bilinmemektedir. Bu tür öğrenmede veriler arasında örüntü aranmaktadır. Veriler arasındaki ilişki birbirlerine olan uzaklıkları, birbirleriyle olan ilişkiler vb. gibi yöntemler ile bulunur. Veriler ile ilgili bir bilgiye sahip olunmadığı için bulunan örüntüler kesin doğru olarak kabul edilmez.

4.1.1. Destek vektör makinesi (SVM)

SVM örüntü tanıma ve tahmin yapmakta oldukça başarılı bir makine öğrenmesi sınıflandırma algoritmasıdır (Theodoridis ve Koutroumbas, 2009). İki sınıflı veri kümelerini ayırmak için kullanılan bir sınıflandırıcıdır. SVM yüksek boyutlu uzaylarda kullanılabilmesinden dolayı avantajlı bir algoritmadır. Hiper düzlem sınıflara ait veri noktaları arasındaki ayrım uzaklığının en fazla olmasına dikkat etmektedir. Hiper düzleme yakın olan veri noktalarına destek vektörü denir. Destek vektörleri hiper düzlemin pozisyonunu etkiler. SVM algoritması doğrusal ve doğrusal olmayan veri kümelerini ikiye ayrılabilir.

Doğrusal SVM sınıflandırıcısı iki veri grubunu ayırmak için optimum düzlemi bulmaya çalışır.

$$g(x) = \omega^T x + \omega_0 = 0 \tag{4.1}$$

Bir noktanın hiper düzleme uzaklığı olan *w* değeri hesaplanır.

$$z = \frac{|g(x)|}{\|w\|} \tag{4.2}$$

Düzlemin iki grubu ayırmak için iki grubun en yakın noktaları olan w_1 ve w_2 arasındaki uzaklığı hesaplanır.

$$\frac{1}{\|w\|} + \frac{1}{\|w\|} = \frac{2}{\|w\|} \tag{4.3}$$

g(x) doğrusu için w_1 1 değerine ve w_2 ise -1 değerine eşittir.

$$w^T x + \omega_0 \geq 1, \forall x \in w_1$$

$$w^T x + \omega_0 \leq -1, \forall x \in w_2$$

Her bir x_i noktası için y_i (w_1 için 1 ve w_2 için -1) noktası hesaplanır. Denklem 4.4 minimalime edilerek Denklem 4.5 tatmin edilir.

$$J(w, w_0) \equiv \frac{1}{2}\|w\|^2 \tag{4.4}$$

$$y_i(w^T x_i + w_0) \geq 1, \qquad i = 1,2, \dots, N \tag{4.5}$$

Denklem 4.4' ü minimalime etmek için Lagrange çarpanı kullanılır.

$$\frac{\partial}{\partial w}\mathcal{L}(w, w_0, \lambda) = 0 \tag{4.6}$$

$$\frac{\partial}{\partial w_0}\mathcal{L}(w, w_0, \lambda) = 0 \tag{4.7}$$

$$\lambda_i \geq 0, i = 1,2, \dots, N \tag{4.8}$$

$$\lambda_i[y_i(w^T x_i + w_0) - 1] = 0, i = 1,2, \dots, N \tag{4.9}$$

Lagrange çarpanı Denklem 4.10' da verilmiştir. Denklem 4.10'daki Lagrange çarpanını Denklem 4.6 ve Denklem 4.7 ile birleştirilince hiper düzlem için en iyi uzaklık Denklem 4.11 elde edilir.

$$\mathcal{L}(w, w_0, \lambda) = \frac{1}{2}w^T w - \sum_{i=1}^{N} \lambda_i[y_i(w^T x_i + w_0) - 1] \tag{4.10}$$

$$w = \sum_{i=1}^{N} \lambda_i y_i x_i \tag{4.11}$$

İdeal hiper düzlem sınıflandırıcı olan destek vektör makinesi hesaplanmış olur.

4.1.2. k-en yakın komşu (kNN)

kNN denetimli makine öğrenmesi algoritmasıdır (Al Banna vd., 2020). Veriyi sınıflandırmak için bu veriye yakın olan diğer verilerin sınıflarını kullanır. Sınıflandırma için k değeri sınıflandırıcının hızını etkilememek için çok büyük bir değer seçilmez. Sınıflandırılacak veri ile yakın komşuları olan veriler için, tek tek Öklid uzaklığı hesaplanır. İki nokta olan *a* ve *b* arasındaki Öklid uzaklığı Denklem 4.12'de verilmiştir. İki nokta olan *a* ve *b* noktaları *x* ve *y* koordinatları ile belirtilmiştir. Uzayda *a* noktası x_1, y_1 olarak *b* noktası ise x_2, y_2 olarak ifade edilmiştir. Veri Öklid uzunlukları en küçük olan k adet veri içinde en çok rastlanan sınıfa dahil edilir. *k* değeri eşitlik durumu ile karşılaşılmaması için tek sayı olarak seçilir.

$$\|a - b\| = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \tag{4.12}$$

4.1.3. Karar ağacı (DT)

Karar Ağaçları çok sınıflı doğrusal olmayan bir sınıflandırıcıdır (Al Banna vd., 2020). Çok sınıflı veri kümeleri için idealdir. Ayrıca iki kümeli veri kümelerinde de kullanılır. Veriyi ait olduğu sınıfı bulana kadar birden fazla karar verme aşamasından geçirir. Bunun sonucunda öznitelik alanı ilgili sınıflar için farklı alanlara bölünür. Oluşan ağaç yapısı yapılan kararlar tarafından oluşturulan düğümlerden oluşur. Ağaç yapıları çok sınıflı yapılarda avantaj sağlamaktadır. Anlaması kolay olan bir yapıya sahip olsa da ezbere öğrenme problemi yaşanır.

İki sınıflı bir karar ağacı oluşturmak için her bir düğüm t, eğitim kümesi *X* 'in belli bir X_t alt kümesi ile ilişkilidir. Bir düğümü ayırmak, X_t altkümesini, X_{tN} ve X_{tY} olarak ikiye ayırmaktır.

$$X_{tY} \cap X_{tN} = \emptyset$$

$$X_{tY} \cup X_{tN} = X_t$$

Bir ayırma sonucunda w_1 ve w_2 sınıfları X_{tY} altkümesine ait sınıfları, w_3 ve w_4 ise X_{tN} alt kümesine ait sınıfları oluşturur. Düğümleri en iyi şekilde ayırmak için, bir ayırma kriteri belirlenir. Düğüm olması ve dallanma sonucu oluşan düğümlerin saf olmama oranının en düşük olmasına çalışılır. $P(wi|t)$, X_t alt kümesindeki bir vektörün, t düğümündeki, ait olduğu w_i, $i = 1, 2, ..., M$ sınıflarına ait olasılığıdır. Düğümün saf olmaması $I(t)$ olarak Denklem 4.13' de verilmiştir.

$$I(t) = -\sum_{i=1}^{M} P(w_i|t) \log_2 P(w_i|t) \tag{4.13}$$

Saf olmama oranındaki düşüşü bulmak için, X_{tY} düğümündeki olasılık N_{tY}/N_t ve X_{tN} düğümündeki olasılık N_{tN}/N_t olarak alınır. Saf olmamadaki düşüş Denklem 4.14' de verilmiştir.

$$\Delta I(t) = I(t) - \frac{N_{tY}}{N_t} I(t_Y) - \frac{N_{tN}}{N_t} I(t_N) \tag{4.14}$$

Bir düğüm oluşturulduktan sonra, düğümün hangi sınıfa ait olduğu Denklem 4.15' deki çoğunluk kuralı ile belirlenir.

$$j = arg\max_i P(w_i|t) \tag{4.15}$$

4.1.4. Naïve Bayes (NB)

Naïve Bayes olasılık hesaplaması kullanan, Bayes teorisini temel alan bir sınıflandırma yöntemidir (Ren vd., 2009). Bu algoritma her veri için tüm sınıf olasılıkları hesaplayarak, en yüksek olasılığa göre sınıflandırır. Bir verinin ait olduğu sınıfı bulmak için olasılık hesabı kullandığından, dengesiz veri hesaplamasında oldukça başarılı sonuçlar verir. Test veri kümesinde, eğitim veri kümesinde görülmeyen bir veri varsa eksik olan veriler sıfırla doldurulur. Eksik veri durumu sonucunda eksik verilerin sıfırla doldurulması olasılık hesaplaması sonuçlarını olumsuz etkiler.

Bayes teoremi 2 rasgele olayın koşullu ve marjinal olasılığı ile ilişkilendirilir. Sınıfı belli olmayan $x = (x_1, x_2, ..., x_d)$ d boyutlu veri örneğinin hangi sınıfa ait olduğunu bulmak için Bayes teoreminden yararlanılır. Sınıfların $C = \{C_1, C_2, ..., C_K\}$ veri kümesinde $P(C_k)$, C_k $(k=1, 2, ..., K)$ için öncü olasılıktır. Koşullu olasılık olan $P(x|C_k)$, x verisinin C_k sınıfında olduğunu kabul eder. Bayes teoremi Denklem 4.16' de verilmiştir.

$$P(C_k|x) = \frac{P(x|C_k)P(C_k)}{\sum_{k'} P(xk_{k'})P(C_{k'})} \tag{4.16}$$

Naïve Bayes algoritması bir sınıfa ait olan özniteliğin değerinin başka bir öznitelikten farklı kabul etmesinden dolayı koşullu olasılık Denklem 4.17' de verilmiştir.

$$P(x|C_k) = \prod_{j=1}^{d} p\left(x^j \middle| C_k\right) \tag{4.17}$$

4.1.5. Stokastik gradyan inişi (SGD)

SGD algoritması, makine öğrenmesi ile kullanılan, sinir ağlarının basit yapısını oluşturan bir sınıflandırıcıdır (Bottou, 2012). SGD sinir ağlarını oluştururken geri yayılım yöntemini kullanır. SGD algoritması eğitim veri kümesinin geniş olduğu durumlarda iyi bir öğrenme algoritmasıdır.

Kayıp fonksiyonu $l(\hat{y}, y)$ olan z örnek verisi (x, y)' den oluşur. Ağırlık vektörü w, $f_w(x)$ fonksiyonudur. Kayıp fonksiyonunu en aza indiren kayıp $Q(z, w) = l(f_w(x), y)$ örnekleri üzerinde ortalanmıştır. Ortalama hesaplanırken $dP(z)$ bilinmeyen bir dağılım kullanılmak istenir. Eğitim kümesine göre ortalama hesaplanması gerektiği için $z_1 ... z_n$' e kadar olan örnekler seçilir. Denklem 4.18' de beklenen risk $E(f)$ genel performans ölçümü verilmiştir. Denklem 4.19' de deneysel risk $E_n(f)$ eğitim kümesi için verilmiştir.

$$E(f) = \int l(f(x), y)\, dP(z) \tag{4.18}$$

$$E_n(f) = \frac{1}{n}\sum_{i=1}^{n} l(f(x_i), y_i) \tag{4.19}$$

Deneysel riski en aza indirmek için gradyan inişi kullanılır ve $E_n(f_w)$ ağırlık w ile güncellenir. Denklem 4.20' de seçilen öğrenme oranı γ ile gösterilmiştir.

$$w_{t+1} = w_t - \gamma \frac{1}{n}\sum_{i=1}^{n} \nabla_w Q(z_i, w_t) \tag{4.20}$$

SGD deneysel risk $E_n(f_w)$ için gradyanı hesaplarken her bir tekrarlama için gradyanı rastgele seçilen bir z_t örnek verisi için hesaplar. SGD tarafından kullanılan deneysel risk hesaplaması Denklem 4.21' de verilmiştir.

$$\omega_{t+1} = w_t - \gamma_t \nabla_w Q(z_t, w_t) \tag{4.21}$$

SGD her bir tekrarlamada ziyaret ettiği noktayı hatırlamak zorunda olmadığı için anında işlem yapabilmektedir. Bu durumda, SGD sınıflandırıcısı beklenen riski seçilen örneklerin rastgele seçilmesiyle doğrudan optimize eder.

4.1.6. Perseptron

Perseptron algoritması, ikili sınıflandırma yapan bir denetimli makine öğrenmesi algoritmasıdır (Theodoridis ve Koutroumbas, 2009). Giriş ve çıkış olmak üzere iki katmana sahip basit bir sinir ağı yapısıdır. Giriş katmanında giriş verisi olan vektörün, ağırlık vektörünü hesaplamak için Denklem 4.22' da verilen maliyet *J(w)* kullanılır.

$$J(w) = \sum_{x \in y} (\delta_x \omega^T x) \tag{4.22}$$

Maliyet fonksiyonunu en düşük yapmak için Denklem 4.23' deki gradyan inişi yöntemi kullanılmıştır.

$$\omega(t+1) = \omega(t) - p_t \frac{\partial J(w)}{\partial \omega}\Big|_{w=\omega(t)} \tag{4.23}$$

Maliyet fonksiyonu ve gradyan inişi birleştirilince Denklem 4.24' deki Perseptron algoritması elde edilmiştir.

$$w(t+1) = w(t) - p_t \sum_{x \in Y} \delta_x x \tag{4.24}$$

4.1.7. Çok katmanlı perseptron (MLP)

MLP algoritması, makine öğrenmesi için sıklıkla kullanılan bir algoritmadır. Bu algoritma Perseptron algoritmasındaki giriş ve çıkış katmanlarından farklı olarak giriş ve çıkış katmanları arasında gizli katmana sahiptir. MLP algoritması birden fazla gizli katmana sahip olabilir. Bazı sınıflandırma problemlerinde iki gizli katmanın, tek gizli katmana göre daha hızlı olduğu görülmüştür (Ruck vd., 1990). Doğrusal olmayan verilerde kullanılabilir ve gerçek zamanlı olarak modelden öğrenme yapabilmektedir. Kayıp fonksiyonu, birden fazla yerel minimal değerine sahip olabildiği için doğruluk oranı değişebilmektedir.

Verilen eğitim setinde $(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)$, $x_i \in R^n$ verileri ve $y_i \in \{0,1\}$ sınıfları temsil eder (Scikit-Learn Classification, 2013). Bir gizli katmanlı MLP algoritması model parametreleri $W_1 \in R^m$ ve $W_2, b_1, b_2 \in R$ olan $f(x) = W_2 g(W_1^T x + b_1) + b_2$ fonksiyonunu öğrenmeye çalışır. W_1 ve W_2 giriş ve gizli katmanın ağırlığını, b_1 ve b_2 gizli ve çıkış katmanlarına eklenen yanlılığı temsil eder. Hiperbolik tanjant fonksiyonu *g()*, Denklem 4.25' de verilmiştir.

$$g(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}} \tag{4.25}$$

İkili sınıflı durumda f(x) fonksiyonu logaritmik fonksiyon olan $g(z) = 1/(1 + e^{-z})$ ile sonuç değeri sıfır ve bir değerleri arasında elde edilir. Kesim değerine göre sınıflar belirlenir. MLP kayıp fonksiyonunu Denklem 4.26' yı kullanarak hesaplar. Kayıp fonksiyonunu en aza indirgemek için ağırlık oranını sürekli olarak günceller.

$$Loss(\hat{y}, y, W) = -y \ln \hat{y} - (1 - y) \ln(1 - \hat{y}) + \alpha \left|\left|W\right|\right|_2^2 \qquad (4.26)$$

4.2. Veri Setleri

Oltalama saldırı tespiti için, birçok çalışma yapılmış olup, çoğunluğunda kendi veri setlerini oluşturmuşlardır. Oluşturdukları veri setleri oltalama saldırıları ile ilgili farklı yapıda veri türlerini içermektedir. Bu tez çalışmasında oltalama web siteleri incelendiği için, oltalama web sitelerini tespit etmek için kullanılan veri setleri incelendi. Açık kaynak kodlu UCI (Muhammed vd., 2015a) ve Mendeley (Tan, 2018) veri setleri, oltalama web sitelerini tespit etmek için yapılan birçok çalışmada kullanılmıştır.

UCI (Muhammed vd., 2015a), oltalama web sitelerinin tespiti için hazırlanmış WEKA'ya uyumlu bir veri setidir. Toplam 11055 web sitesinden veri toplanmıştır, veri kümesinde yasal ve oltalama web siteleri eşit bir dağılıma sahip değildir. Veri seti 30 öznitelik içermektedir. Yasal web siteleri Yahoo!'dan toplanmıştır (Yahoo Inc., 1994). Oltalama web siteleri PhishTank arşivinden toplanmıştır (PhishTank, 2006).

Mendeley (Tan, 2018), oltalama web sitelerinin tespitinde, güncel hazırlanmış bir veri setinin, yeni yapılan oltalama saldırılarının tespitinde daha iyi başarım göstereceği düşüncesiyle hazırlanmıştır. WEKA'ya uyumlu hazırlanan veri seti, 48 adet öznitelikden oluşur. Toplam 10000 adet web sitesi içerir. Yasal ve oltalama web sitelerinin dağılımı eşittir. Yasal web siteleri Alexa ve Common Crowl arşivlerinden toplanmıştır (Alexa, 1996; The Common Crawl Foundation, 2011). Oltalama web siteleri PhishTank ve OpenPhish arşivinden toplanmıştır (PhishTank, 2006; OpenPhish, 2020).

Bu tez çalışmasında kullanılmak üzere, oltalama web sitelerini sınıflandırmak için yeni bir veri kümesi hazırlanmıştır. Veri kümesi için, yasal web sitelerinin URL'leri

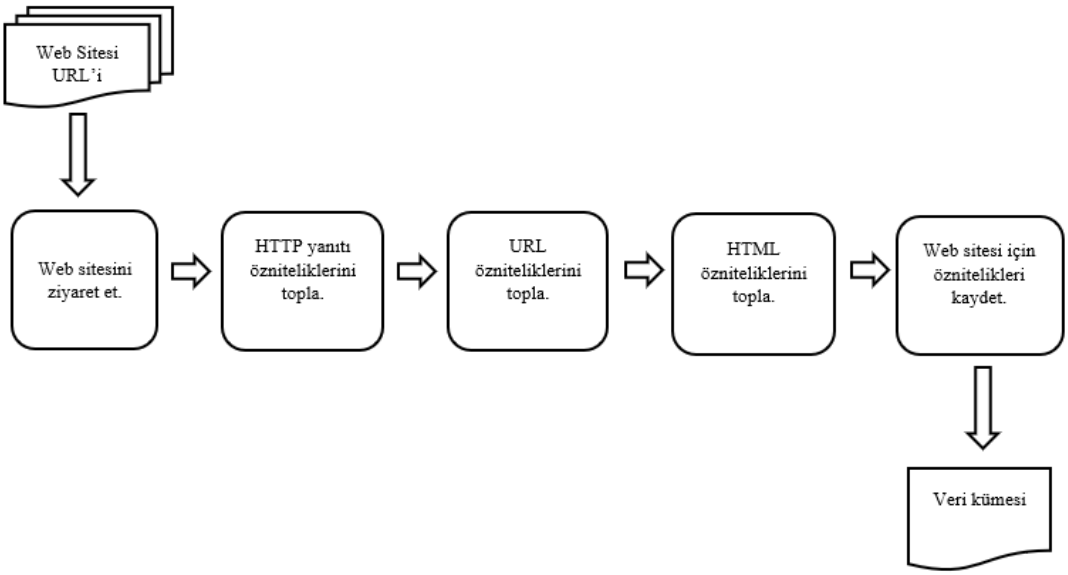
Alexa'dan ve oltalama web sitelerinin URL'leri PhishTank'den toplanmıştır. Toplanan yasal web siteleri, sadece en çok ziyaret edilen web siteler olmaması için "update, finance, log in, shopping" kelimeleri aranarak elde edilen web siteleri de eklenmiştir. Bu sayede çok az ziyaret edilen ve oltalama web siteleri ile karıştırılabilecek web siteleri URL'leri de toplanmıştır.

Toplanan URL'ler Selenium kütüphanesi kullanılarak Firefox arama motoru üzerinden ziyaret edilerek 25 öznitelik toplanmıştır (Selenium, 2020; Mozilla Foundation, 2002). Kayıp değerleri olan web siteleri veri kümesine dahil edilmemiştir. Veri kümesi için 500 oltalama ve 500 yasal eşsiz web sitesinden oluşmaktadır. Öznitelikler 3 ayrı gruba ayrılmıştır; URL, HTML ve HTTP yanıtı.

Öznitelikler çıkarılırken, öncellikle URL'nin ulaşılır olup olmadığı kontrol edilmiştir. Toplanan web sitesi URL'ine ulaşılmadıysa, bu web sitesi veri kümesine dahil edilmez. Ulaşılan web sitesinin öncelikle HTTP yanıtı öznitelikleri çıkarılmıştır. HTTP yanıtı öznitelikleri web sitesinin tutulduğu sunucudan gelen cevap olduğu için bu web sitesinin ulaşılabilirliği açısından önemli bir veridir.

Web sitesi ulaşılabiliyorsa 2. aşamada URL öznitelikleri toplanmıştır. URL öznitelikleri, URL üzerinde veri toplanarak yapılmıştır. URL özniteliklerini elde etmek için, web sitesini ziyaret etmeye gerek olmadığı düşünülmektedir. Ancak, arama motoru tarafından gönderilen anlık URL, aranan URL'den farklılık gösterebilmektedir. Bu nedenle, web sitesine ulaşmak bu öznitelik grubu içinde önemlidir.

Veri toplamanın 3. Aşamasında, web sitesinin HTML öznitelikleri toplanmıştır. Bir web sitesine ulaşılsa bile HTML öznitelikleri toplanırken, web sitesinin yapısından dolayı HTML verisine ulaşılamayan durumlar olmaktadır. HTML verisine ulaşılamadığı zaman web sitesi veri kümesine dahil edilmez. Bunun nedeni veri kümesinde kayıp değer istenilmemesidir.



Şekil 4.1 Veri toplama aşaması

Bir web sitesinin URL, HTML ve HTTP yanıtı özniteliklerinin tamamı toplanabilirse, bu web sitesi veri kümesine dahil edilmiştir. Veri kümesine dahil edilen 500 yasal web sitesi ve oltalama web sitesinin öznitelikleri bu kurallara uyarak kayıp değer olmayacak şekilde toplanmıştır. Veri toplama aşamasının akış şeması Şekil 4.1' de verilmiştir.

4.3. Öznitelikler

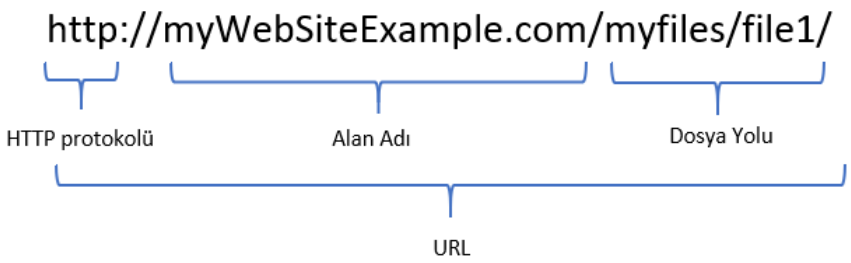
Oltalama ve normal web sitelerine ait 25 öznitelik toplanmıştır. Toplanan öznitelikler URL, HTML ve HTTP yanıtı olarak 3 farklı gruba ayrılmıştır. Veri kümesi 18 adet URL özniteliği, 5 adet HTML ve 2 adet HTTP yanıtı özniteliği içerir. Toplanan özniteliklerin öznitelik grubu, ne tür veri içerdiği ve alabileceği değerler Çizelge 4.1.'de açıklanmıştır.

Çizelge 4.1. Bu tez kapsamında hazırlanan veri setinin öznitelikleri ve öznitelik grupları

Öznitelik Grubu	Öznitelik	Tanımı	Değerler
URL	Alan adı benzerliği	Sayısal	
	URL uzunluğu	Sayısal	
	HTTP protokolü	http veya https	http, https
	Toplam nokta sayısı	Sayısal	
	Toplam eğik çizgi sayısı	Sayısal	
	Toplam çift eğik çizgi sayısı	Sayısal	
	Toplam kısa çizgi sayısı	Sayısal	
	Toplam alt tire sayısı	Sayısal	
	Toplam eşittir işareti sayısı	Sayısal	
	Toplam parantez işareti sayısı	Sayısal	
	Toplam küme ayracı sayısı	Sayısal	
	Toplam köşeli ayraç sayısı	Sayısal	
	Toplam küçüktür ve büyüktür işareti sayısı	Sayısal	
	Toplam tilda işareti sayısı	Sayısal	
	Toplam yıldız işareti sayısı	Sayısal	
	Toplam artı işareti sayısı	Sayısal	
	URL'nin '@' işareti içermesi	Boolean	true, false
	URL'nin IP içermesi	Boolean	true, false
HTML	Toplam 'a' etiketi sayısı	Sayısal	
	Toplam giriş etiketi sayısı	Sayısal	
	Toplam düğme etiketi sayısı	Sayısal	
	Toplam 'link' etiketi sayısı	Sayısal	
	Toplam iFrame sayısı	Sayısal	
HTTP Yanıtı	HTTP yanıt geçmişi	Dizi	Response 3XX
	Yeniden yönlendirme	Boolean	true, false

4.3.1. URL öznitelikleri

URL, internetteki bir kaynağın veya bilgisayarda tutulan bir dosyanın adresidir. Sunucu üzerinde tutulan kaynak bir web sitesi, dosya, resim veya uygulama olabilir. URL adresi 3 ana yapıdan oluşur: dosyaya veya web sitesine ulaşmak için kullanılan protokol tipi, dosyanın bulunduğu sunucunun alan adı veya İnternet İletişim Kuralı adresi ve dosyanın yol ismini içerir. URL örneği 'http://myWebSiteExample.com/myfiles/file1/' HTTP protokolünü kullanır. Sunucunun alan adı 'myWebSiteExample.com' olup, '/myFiles/file1' dosyasının yol ismidir. Şekil 4.2.'de bir web sitesi URL'si örneğini verilmiştir.



Şekil 4.2. Bir web sitesi URL'si örneği ve yapısı

4.3.1.1. Alan adı benzerliği

Alan adı bir web sitesinin internet üzerinde kullandığı isimdir. Bir web sitesine ulaşmak için IP adresinin bilinmesi gerekmektedir. IP adresinin akılda kalması zor olduğu için IP adresi için bir alan adı kullanılır. Arşivden toplanan web sitesini URL'nin alan adı ve son olarak yönlendirilen web sitesi URL'nin alan adı karşılaştırılır. Karşılaştırma işleminde, Python *difflib* kütüphanesi kullanılmıştır (Python Documentation, 2001). *difflib* kütüphanesinde kullanılan Dizi Eşleştiricisi (Sequence Matcher) algoritması, (Ratcliff ve Metzener, 1988) tarafından geliştirilen algoritma temel alınarak oluşturulmuştur. Dosyaların içerikleri arasındaki farklar hesaplanarak, dosyaların benzerlik oranını bulunur. Genellikle aynı dosyanın farklı sürümleri arasındaki değişikliği göstermek için kullanılır.

4.3.1.2. URL uzunluğu

URL adresinin içerdiği karakter sayısını veri olarak tutulur. Veri tipi tam sayıdır. URL adresi 'https://www.XWebSitesi.com/docs/files/' olan web sitesinin URL uzunluğu 37'dir.

4.3.1.3. HTTP protokolü

Web sitesinin Hypertext Transfer Protocol (HTTP) veya Hypertext Transfer Protocol Secure (HTTPS) protokollerinden hangisini kullandığına bakılmıştır. Katar veri tipinde 'https' ya 'http' verisini saklar. HTTP web üzerinden paket alım ve gönderimi için Transmission Control Protocol (TCP) kullanır (Jackson, 2016). HTTP genellikle port 80'i kullanır. HTTPS web üzerinden paket alım ve gönderimi için TCP kullanır. HTTP'den farklı

olarak HTTPS port 443 üzerinden Transport Layer Security (TLS) tarafından şifrelenmiş bir bağlantı ile web'e bağlanır.

4.3.1.4. Toplam nokta sayısı

URL'nin içerdiği toplam nokta sayısından 3 eksiltilerek veri olarak tutulur. Bir web sitesini URL'si 'http://www.XWebSitesi.com/' şeklinde veya ülke kodu *tr* olacak 'http://www.XWebSitesi.com.tr/' şekilde tanımlanabilir. Ülke kodlu URL'den dolayı toplam nokta sayısından 3 azaltılmıştır.

4.3.1.5. Toplam eğik çizgi sayısı

URL'nin içerdiği toplam eğik çizgi '/' sayısından 3 eksiltilerek veri olarak tutulur. Bir web sitesinin URL'nin en basit hali ile 'http://www.XWebSitesi.com/' olarak tanımlanır. URL'nin en temel hali içerisinde 3 adet eğik çizgi karakteri içermektedir. Bu nedenle toplam eğik çizgi sayısından 3 eksiltilmiştir.

4.3.1.6. Toplam çift eğik çizgi sayısı

URL'nin içerdiği toplam çift eğik çizgi '//' sayısından 1 eksiltilerek veri olarak tutulur. Bir web sitesinin URL'nin en basit hali ile 'http://www.XWebSitesi.com/' olarak tanımlanır. URL'in en temel hali içerisinde 1 adet çift eğik çizgi karakteri içermektedir. Bu nedenle toplam eğik çizgi sayısından 1 eksiltilmiştir.

4.3.1.7. Toplam kısa çizgi sayısı

Bir web sitesi URL'si oluşturulurken kısa çizgi '-' karakteri ile kelimeler ayrılmaz. Bundan dolayı toplam kısa çizgi sayısı öznitelik olarak tutulmuştur. Toplam kısa çizgi sayısı 'http://www.XWebSitesi.com/sites-web-abc/login/my-page/my-files/' URL'si için 4'tür.

4.3.1.8. Toplam alt tire sayısı

Bir web sitesi URL'si oluşturulurken alt tire '_' karakteri ile kelimeler ayrılmaz. Bundan dolayı toplam alt tire sayısı öznitelik olarak tutulmuştur.

4.3.1.9. Toplam eşittir işareti sayısı

Bir web sitesi URL'si oluşturulurken eşittir '=' karakteri kullanılmaz. Bundan dolayı toplam eşittir işareti sayısı öznitelik olarak tutulmuştur.

4.3.1.10. Toplam parantez işareti sayısı

Bir web sitesi URL'si oluşturulurken parantez '(,)' karakterleri kullanılmaz. Bundan dolayı toplam parantez işareti sayısı öznitelik olarak tutulmuştur.

4.3.1.11. Toplam küme ayracı sayısı

Bir web sitesi URL'si oluşturulurken küme ayracı ',' karakterleri kullanılmaz. Bundan dolayı toplam küme ayracı sayısı öznitelik olarak tutulmuştur.

4.3.1.12. Toplam köşeli ayraç sayısı

Bir web sitesi URL'si oluşturulurken köşeli ayraç '[,]' karakterleri kullanılmaz. Bundan dolayı toplam köşeli ayraç sayısı öznitelik olarak tutulmuştur.

4.3.1.13. Toplam küçüktür ve büyüktür işareti sayısı

Bir web sitesi URL'si oluşturulurken büyüktür ve küçüktür '<, >' karakterleri kullanılmaz. Bundan dolayı toplam büyüktür ve küçüktür sayısı öznitelik olarak tutulmuştur.

4.3.1.14. Toplam tilda işareti sayısı

Bir web sitesi URL'si oluşturulurken tilda '~' karakteri kullanılmaz. Bundan dolayı toplam tilda sayısı öznitelik olarak tutulmuştur.

4.3.1.15. Toplam yıldız işareti sayısı

Bir web sitesi URL'si oluşturulurken yıldız '*' karakteri kullanılmaz. Bundan dolayı toplam yıldız sayısı öznitelik olarak tutulmuştur.

4.3.1.16. Toplam artı işareti sayısı

Bir web sitesi URL'si oluşturulurken artı '+' karakteri kullanılmaz. Bundan dolayı toplam artı sayısı öznitelik olarak tutulmuştur.

4.3.1.17. URL'nin '@' işareti içermesi

Bir web sitesi URL'si oluşturulurken '@' karakteri kullanılmaz. Bu karakter eposta adreslerinde kullanılır. Bundan dolayı URL'nin '@' işareti içerip içermediği öznitelik olarak tutulmuştur.

4.3.1.18. URL'nin IP içermesi

IP bir web sitesinin adresini temsil eder. IP adreslerinin hatırlanması zor olduğu için alan adı kullanılır. IP adresleri genellikle IPv4 standardını kullanır, 32 bitlik bir adres alanı vardır. IPv4 standardını kullanan bir web sitesinin IP adresi '64.233.160.150' şeklindedir. URL içerisinde IPv4 standardında IP içerip içermediği öznitelik olarak tutulmuştur.

4.3.2. HTML öznitelikleri

HTML web sitelerini oluşturmak için kullanılan standart bir metin dilidir. Arama motorları veya bu dili yorumlayan programlar aracılığı ile çalışır. Metin, resim ve video verilerini ve web sitesi sayfalarını birbirine bağlamak için kullanılır. Web sitesinin nasıl görünmesi gerektiği ile ilgili kuralları belirler. HTML dili yorumlayan arama motoruna göre farklı sonuçlar verebilir.

4.3.2.1. Toplam ‘a’ etiketi sayısı

HTML etiketi olan ‘a href’ web sitesi içerisinde farklı roller için kullanılan bir yapıya sahiptir (W3Schools, 1999). Bağlantı hedef noktasını göstermesi ile bilinir ve farklı amaçlarla kullanılabilir. Bu etiket farklı bir web sitesi URL’sinin (href="http://www.example.com/default.htm"), aynı web sitesi içerisindeki farklı bir dosyayı (href="default.htm") veya aynı web sitesi içerisindeki özel isimlendirilmiş bir elementi (href="#section2") hedef gösterebilir. Ayrıca en önemli özelliği komut dosyası kodu (href="javascript:alert('Hello');") içerebilmesidir.

4.3.2.2. Toplam giriş etiketi sayısı

Bir web sitesi kullanıcı tarafından veri girişi almak için HTML dilinin özelliği olan form etiketi kullanır. Bu sayede kullanıcıdan veri alınır. Bu veri web sitesi içerisinde kullanılabileceği gibi farklı web sitelerine yönlendirme işleminde kullanılabilir. Ayrıca kullanıcı tarafından girilen bilgiler saklanabilir. Toplam veri giriş sayısı veri olarak tutulmuştur.

4.3.2.3. Toplam düğme etiketi sayısı

HTML etiketi olan düğme etiketi, onaylama, kaydetme veya güncelleme düğmesi olarak web sitesinde kullanılır. Düğmeye tıklayarak form etiketinden gelen veriler

kaydedilebilir. Ayrıca düğme etiketinin tıklanma özelliği kullanılarak farklı web sitelerine yönlendirilebilir. Toplam düğme sayısı veri olarak tutulmuştur.

4.3.2.4. Toplam 'link' etiketi sayısı

HTML etiketi olan link etiketi CSS dosyasına erişmek için kullanılır. CSS dosyası bir web sitesinin görünüşünü etkilemektedir. Web siteleri kendine ait CSS dosyalarına sahip olmasına rağmen, CSS dosyasına erişmek için bir web sitesi de kullanılabilir. CSS dosyasını web sitesi URL'si kullanan toplam link etiketi sayısı veri olarak tutulmuştur.

4.3.2.5. Toplam iFrame sayısı

HTML elementi olan iFrame, Inline Frame'in kısaltılmışıdır. HTML dokümanı içerisine farklı dokümanlar, video ve interaktif medya elementi eklemeye yarar. Ayrıca bir web sitesinin içine farklı bir web sitesi bu element kullanılarak eklenebilir. Bundan dolayı toplam iFrame sayısı veri olarak tutulmuştur.

4.3.3. HTTP yanıtı öznitelikleri

HTTP yanıtı, server tarafından istek gönderen arama motoruna gönderilen yanıt cevabıdır. HTTP yanıtları 5 farklı kod grubuna ayrılmıştır. Bilgilendirme, Başarı, Yeniden Yönlendirme, Kullanıcı Hataları ve Server Hataları kodlarından oluşur. Bu çalışmada veri toplarken ulaşabildiğimiz web sitelerinden yararlanıldığı için, Kullanıcı ve Server hataları dikkate alınmamıştır. Toplanan tüm sitelere başarılı bir şekilde ulaşıldığı için Başarı kodu öznitelik olarak anlamsız olmaktadır. Bilgilendirme kodları bu çalışmada dikkate alınmamıştır. Yeniden yönlendirme kodları 3XX olarak kodlanmıştır ve web sitesi yönlendirmesi hakkında bilgi vermektedir.

4.3.3.1. HTTP yanıt geçmişi

Bu öznitelikte Server tarafından gönderilen tüm HTTP yanıtları katar veri tipinde tutulmuştur. Bu öznitelik Yeniden Yönlendirme kodlarını içermektedir. Yeniden Yönlendirme kodları bir web sitesinin farklı bir web sitesine yönlendirildiği hakkında bilgi verir.

4.3.3.2. Yeniden yönlendirme

Bu öznitelik HTTP yanıt Geçmişi özniteliği kullanılarak oluşturulmuştur. Yeniden yönlendirme olup olmadığı hakkında bilgi verir. Oltalama saldırılarında kullanılan web sitelerinin toplandığı PhishTank arşivlerinden toplanan URL'lerde web sitesinin http veya *https* protokolü bilindiği için tüm HTTP Yanıtı Geçmişi kullanılmıştır. HTTP Yanıt Geçmişi olan oltalama web siteleri yeniden yönlendirme özniteliği olduğu bilgisi veri kümesinde tutulmuştur. Alexa arşivlerinden toplanan yasal web sitelerinin http veya *https* protokolü bilinmediği için arama motoruna http protokolü ile gönderilmiştir. Bundan dolayı yasal web sitesinin *https* protokolüne ve ülke koduna sahip olabileceği düşünülerek ikiden fazla HTTP yanıtına sahipse yeniden yönlendirme olduğu kabul edilmiştir.

5. BULGULAR VE TARTIŞMA

Bu bölümde, oltalama web sitelerini tespit etmek için gerçekleştirilen deneysel çalışmalarda elde edilen sonuçlara değinilmiştir. Deneysel çalışmalar sonucunda elde edilen değerlerin literatürdeki sonuçlarla karşılaştırılması yapılmıştır.

5.1. Deneysel Çalışma

Yapılan deneyler, Intel(R) Core(TM) i5-2430M CPU @2.40GHz işlemcili, 8,00 GB RAM donanımında, Python programlama dili kullanılarak gerçekleştirilmiştir. Makine öğrenmesi sınıflandırıcılarını kullanmak için Scikit-learn kütüphanesinden yararlanılmıştır (Scikit-Learn, 2013). Oltalama web sitelerinin tespiti amaçlı yapılan tez kapsamındaki deneysel çalışmalarda, yasal/oltalama web sayfalarından elde edilen farklı türdeki öznitelik gruplarının, sınıflandırma modeline olan katkıları kapsamlı arama yöntemleri ile incelenmiştir. Hazırlanan veri setinde, üç farklı kategoriye ait, URL, HTML ve HTTP cevabı şeklinde gruplanmış öznitelikler, kapsamlı tarama yöntemi ile, 2^3 - 1 farklı deney grubuna ayrılmıştır. Hazırlanan veri kümesi sayesinde öznitelik gruplarının sınıflandırma algoritmalarındaki başarım oranları ve birbirleri ile olan ilişkileri ve ayırt edicilikleri ortaya çıkarılmış olup, tez kapsamında veri kümelerinin her bir algoritma için Süre, Doğruluk, Kesinlik, Duyarlılık, F1-skor, TN, FP, FN ve TP değerleri karşılaştırmalı olarak sunulmuştur.

Süre değeri bir deney sürecinin başlangıcı ve sonucu arasında geçen süre olarak bulunmuştur. Zaman değeri hesaplamasında deney ortamının hazırlandığı süreç katılmamıştır, sadece öğrenme ve test aşaması dikkate alınmıştır.

$$Süre = t_{bitiş} - t_{başlangıç}$$

Yapılan çalışmada, oltalama web siteleri tespit edilmeye çalışılmaktadır. Çalışmada oluşturulan veri kümesinde, oltalama web siteleri *Oltalama,* yasal web siteleri ise *Yasal*

olarak ile ifade edilmiştir. True-Negatif (TN) değeri, doğru tahmin edilen yasal web sitelerini ifade etmektedir. True-Pozitif (TP) değeri, doğru tahmin edilen oltalama web sitelerini ifade etmektedir. False-Pozitif (FP) değeri, yanlış tahmin edilen oltalama web sitesini ifade eder, ayrıca FP değeri yanlış alarm olarak da adlandırılır. False-Negatif (FN) değeri, yanlış tahmin edilen yasal web sitelerini ifade eder. TN, TP, FP ve FN değerlerini içeren Karışıklık Matrisi (Confusion Matrix) Çizelge 5.1.'de verilmiştir (Alpaydin, 2020).

Çizelge 5.1. Karışıklık Matrisi

		Tahmin	
		Yasal	Oltalama
Asıl	Yasal	TN	FP
	Oltalama	FN	TP

Doğruluk değeri, ikili sınıflandırmalarda testin ne kadar doğru çalıştığını hesaplamak için kullanılır. Doğruluk tüm tahminler içerisinde doğru yapılan tahminlerin oranıdır. TP ve TN değerlerinin tüm tahminlere oranı karşılaştırılır.

$$Doğruluk = \frac{TN + TP}{TN + TP + FN + FP}$$

Kesinlik değeri, doğru tahmin edilen değerlerden ne kadarının aslında doğru değer olduğunu bulmak için kullanılır. Tahminleri TP ve FP değerleri oluşturur. Kesinlik FP değerinin sonuca yüksek etkisi olduğu durumlarda iyi bir ölçüm yöntemidir.

$$Kesinlik = \frac{TP}{TP + FP}$$

Duyarlılık değeri doğru değerlerden ne kadarının doğru tahmin edildiğini bulmak için kullanılır. Doğru değerleri FN ve TP oluşturmaktadır. Duyarlılık değerini yanlış tahmin edilen sayısı FN belirler. Eğer yanlış tahminlerin sonuca yüksek etkisi varsa duyarlılık iyi bir ölçüm yöntemidir.

$$Duyarlılık = \frac{TP}{TP\ +\ FN}$$

F1-skor değeri Kesinlik ve Duyarlılık değerlerinin harmonik ortalamasıdır. Kesinlik ve Duyarlılık değerleri arasında denge sağlanmaya çalışılmıştır. Dengeli olmayan veri setlerinde kullanılmaktadır.

$$F1 - Skor = 2\,.\frac{Kesinlik\,.\,Duyarlılık}{Kesinlik\ +Duyarlılık} = \frac{2TP}{2TP + FP + FN}$$

Çizelge 5.2.'de, kullanılan sınıflandırma algoritmaları ve hangi öznitelik grubunun seçildiği belirtilmiştir. Eğitim ve test veri setleri, her deney için aynı olup, sadece kullanılan öznitelikler, öznitelik grubuna göre değişmektedir. Veri kümesi üç adet öznitelik grubu içermektedir. Öznitelik grupları bazında kapsamlı arama yapılarak, en iyi öznitelik grubu seçilmeye çalışılmıştır. Her bir sınıflandırıcı için, 7 farklı deney gerçekleştirilmiştir. URL öznitelikleri için *URL* kısaltması, HTML öznitelikleri için *HTML* ve HTTP Yanıtı Öznitelikleri için *HTTP* kelimesi kullanılmıştır. Ayrıca deneylerin sonuçları süre, doğruluk (Acc), kesinlik (Pre), duyarlılık (Rec), F1-skor (F1-S), TN, TP, FN ve FP ölçümleri olarak verilmiştir.

SVM algoritması kullanılarak yapılan deneylerde, tüm özniteliklerin kullanıldığı *URL + HTML + HTTP* deney grubu %98 oranı ile en iyi F1-skor değerine sahiptir. *URL + HTTP* deney grubu incelendiğinde %97 oranında F1-skor değerine sahiptir. İki deney grubu arasında, 5 öznitelik sayısı ve 0,02 saniye sınıflandırma süresi farkları bulunmaktadır. Ayrıca, duyarlılık değerleri incelendiğinde, her iki deney grubu içinde %99 değerini vermektedir. SVM algoritması için, daha az öznitelikle birlikte, daha hızlı sınıflandırılmakta olup, tüm özniteliklerin kullanıldığı deneye yakın sonuç elde edildiği için, *URL + HTTP* öznitelik grubu tercih edilebilir bir öznitelik kombinasyonu olarak kabul edilebilir.

SGD algoritması kullanılarak yapılan deneylerde, bütün özniteliklerin kullanıldığı *URL + HTML + HTTP* deney grubu %98 F1-skor değerine ve %97'lik duyarlılık oranına sahiptir. Bunun yanında *URL + HTTP* deney grubu %97 F1-skor değerine ve %97'lik duyarlılık oranına sahiptir. *URL + HTTP* deney grubu daha az özniteliğe sahip olmasına

rağmen birbirine çok yakın değerlere sahiptir. Ayrıca, daha kısa sürede sınıflandırma yapılması nedeniyle daha iyi bir öznitelik grubu olarak kabul edilebilir.

Perseptron algoritması ile yapılan deneyler incelendiğinde, *URL + HTML + HTTP* ve *URL + HTTP* deney grupları için süre ve F1-skor değerlerinin aynı elde edildiği gözlemlenmektedir. F1-skor değerleri aynı çıkmasına rağmen, *URL + HTTP* deney grubu daha az özniteliğe sahip olduğu ve diğer deney grubu ile aynı sürede sınıflandırma yaptığı için, *URL + HTTP* deney grubunun, Perseptron sınıflandırma algoritması için başarılı bir grup olduğu anlaşılmıştır.

MLP algoritması ile yapılan deneylerde *URL + HTTP* deney grubu, %98 F1-skor değeri ile en başarılı öznitelik grubu olmuştur. Ayrıca duyarlılık değerlerine bakılınca *URL + HTTP* ve *HTTP* deney grupları %99 oranı ile en iyi sonuçlara sahiptir. Bu çalışmada FN değeri, aslında oltalama web sitesi olan web sitelerinin yasal web sitesi olarak sınıflandırıldığını gösterir. MLP algoritması ile alınan sonuçlara bakılınca *HTTP* özniteliklerinin FN değerini düşürdüğü bunun sonucunda oltalama web sitelerini tespit etme oranının arttığı gözlemlenmiştir. Tüm özniteliklerin kullanıldığı *URL + HTML + HTTP* deney grubu en iyi F1-skor sonucuna sahip olmamasına rağmen, en kısa tanımlama süresine sahiptir.

kNN algoritması için yapılan deneylerde, ideal *k* değeri 5 olarak belirlenmiştir (Tyagi vd., 2018). kNN algoritması ile yapılan deney sonuçları incelendiğinde *URL + HTML + HTTP* ve *URL + HTTP* deney grupları %97 F1-skor oranlarına sahip olup, sınıflandırma süreleri de aynıdır. Ancak, FN ve duyarlılık değeri incelendiğinde *URL + HTTP* deney grubunun daha başarılı olduğu gözlemlenmiştir. Ayrıca, *URL + HTTP* deney grubu daha az öznitelik içermektedir.

DT algoritması kullanılarak yapılan deney sonuçları incelendiğinde, *URL + HTTP* deney grubunun %99 F1-skor oranı ile en başarılı öznitelik grubu olduğu gözlemlenmiştir. Ayrıca, FN ve duyarlılık değerleri incelendiğinde *URL + HTTP* ve *HTTP* gruplarının düşük FN değere ve yüksek duyarlılık değerine sahip olduğu gözlemlenmektedir.

Çizelge 5.2. Hazırlanan veri kümesinin test sonuçları

SA	Öznitelik Grupları	TÖS	Süre	Acc	Pre	Rec	F1-S	TN	FP	FN	TP
	URL	18	0,03	0,91	0,91	0,91	0,91	137	13	14	136
	URL + HTML	23	0,03	0,94	0,93	0,95	0,94	140	10	8	142
	URL + HTTP	20	**0,03**	0,97	0,96	0,99	0,97	144	6	2	148
SVM	*HTML*	5	0,03	0,85	0,81	0,91	0,86	119	31	14	136
	HTML + HTTP	7	0,03	0,92	0,88	0,97	0,93	131	19	4	146
	HTTP	2	0,02	0,93	0,88	0,99	0,93	130	20	2	148
	URL + HTML + HTTP	25	0,05	0,98	0,97	0,99	**0,98**	145	5	1	149
	URL	18	0,03	0,84	0,86	0,81	0,84	130	20	28	122
	URL + HTML	23	0,03	0,92	0,91	0,93	0,92	137	13	11	139
	URL + HTTP	20	**0,02**	0,97	0,97	0,97	0,97	146	6	6	146
SGD	*HTML*	5	0,02	0,84	0,84	0,85	0,84	125	25	22	128
	HTML + HTTP	7	0,03	0,91	0,88	0,95	0,92	131	19	7	143
	HTTP	2	0,03	0,92	0,87	0,99	0,93	128	22	1	149
	URL + HTML + HTTP	25	0,03	0,98	0,98	0,97	**0,98**	147	3	4	146
	URL	18	0,03	0,92	0,91	0,92	0,92	137	13	12	138
	URL + HTML	23	0,03	0,94	0,92	0,96	0,94	138	12	6	144
	URL + HTTP	**20**	**0,02**	0,97	**0,98**	0,96	**0,97**	147	3	6	144
Perseptron	*HTML*	5	0,02	0,79	0,72	0,95	0,82	96	54	8	142
	HTML + HTTP	7	0,02	0,91	0,89	0,94	0,91	132	18	9	141
	HTTP	2	0,03	0,88	0,88	0,87	0,88	132	18	19	131
	URL + HTML + HTTP	25	0,02	0,97	0,96	**0,97**	0,97	144	6	4	146
	URL	18	0,0	0,67	0,93	0,37	0,53	146	4	95	55
	URL + HTML	23	0,03	0,68	0,95	0,37	0,54	147	3	94	56
	URL + HTTP	20	0,02	0,67	0,93	0,37	0,53	146	4	94	56
NB	*HTML*	5	0,02	0,75	0,68	0,94	0,79	84	66	9	141
	HTML + HTTP	7	**0,02**	0,81	0,75	0,95	0,84	102	48	8	142
	HTTP	2	0,03	0,92	0,87	0,99	**0,93**	128	22	1	149
	URL + HTML + HTTP	25	0,03	0,68	0,95	0,38	0,54	147	3	93	57
	URL	18	0,61	0,95	0,92	0,97	0,95	138	12	4	146
	URL + HTML	23	0,08	0,93	0,90	0,96	0,93	134	16	6	144
	URL + HTTP	20	0,12	0,98	0,97	0,99	**0,98**	146	4	2	148
MLP	*HTML*	5	0,17	0,84	0,83	0,87	0,85	123	27	20	130
	HTML + HTTP	7	1,0	0,91	0,88	0,95	0,92	131	19	7	143
	HTTP	2	0,08	0,92	0,87	0,99	0,93	128	22	1	149
	URL + HTML + HTTP	25	**0,06**	0,96	0,97	0,95	0,96	145	5	8	142
	URL	18	0,05	0,93	0,89	0,98	0,93	132	18	3	147
	URL + HTML	23	0,05	0,92	0,91	0,93	0,92	137	13	11	139
	URL + HTTP	20	0,05	0,97	0,94	**0,99**	**0,97**	141	9	1	149
kNN	*HTML*	5	0,05	0,85	0,85	0,85	0,85	128	22	23	127
	HTML + HTTP	7	0,05	0,93	0,91	0,96	0,94	136	14	6	144
	HTTP	2	0,05	0,92	0,87	0,98	0,92	128	22	3	147
	URL + HTML + HTTP	25	0,05	0,97	0,95	0,98	0,97	143	7	3	147
	URL	18	**0,02**	0,92	0,93	0,92	0,92	139	11	12	138
	URL + HTML	23	0,02	0,91	0,90	0,92	0,91	134	16	12	138
	URL + HTTP	20	0,03	0,99	0,99	0,98	**0,99**	149	1	3	147
DT	*HTML*	5	0,02	0,81	0,80	0,83	0,82	119	31	25	125
	HTML + HTTP	7	0,02	0,91	0,90	0,92	0,91	134	16	12	138
	HTTP	2	0,03	0,93	0,89	0,99	0,94	131	19	1	149
	URL + HTML + HTTP	25	0,03	0,96	0,98	0,94	0,96	147	3	9	141

NB algoritması kullanılarak yapılan deneylerde ise, diğer algoritmalardan farklı olarak, sadece *HTTP* deney grubu belirgin bir fark yaratarak, %93 F1-skor oranına sahiptir. Sonuçlar dikkatli bir şekilde incelendiğinde, URL özniteliklerini içeren *URL*, *URL + HTML*, *URL + HTTP* ve *URL + HTML + HTTP* öznitelik grupları için, sırasıyla %53, %54, %53 ve %54 F1-skor oranlarına sahiplerdir. Diğer sınıflandırıcı algoritmalarının performans karakteristiklerinden farklı olarak, *URL* özniteliklerini içeren ve içermeyen deney grupları incelendiğinde, *URL* öznitelik grubunun sonuçlara negatif etkisi olduğu görülmektedir.

URL + HTTP öznitelik grubu, diğer sınıflandırıcı algoritmalarında yüksek başarım göstermesine rağmen, NB algoritması ile düşük başarım göstermektedir. Bu nedenle, NB algoritması ile birlikte, *URL + HTTP ((18+2) Adet)* öznitelik grubuna, kapsamlı arama yapılarak, URL grubunun negatif etkisi araştırılmıştır. Bu deneysel çalışmaların sonucunda, en yüksek doğruluk oranı, dört adet öznitelik (Alan Adı Benzerliği, URL Uzunluğu, HTTP Yanıt Geçmişi, Yeniden Yönlendirme) ile %97 olarak elde edilmiştir. Ayrıntılı Arama sonucunda, en iyi öznitelik kümesi Çizelge 5.3.' de verilmiştir. Bu özniteliklerden ikisi *URL* öznitelik grubundan, diğer ikisi ise *HTTP* öznitelik grubundandır. Yapılan kapsamlı arama, elenen diğer *URL* özniteliklerinin ayırt ediciliklerinin düşük olduğunu ve bundan dolayı NB algoritması ile elde edilen modelde, oltalama web sitelerini tespit etmeyi güçleştirdiklerini ortaya çıkarmıştır.

Çizelge 5.3. NB ile Kapsamlı Arama sonucu *URL + HTTP* grubu ayırt edici öznitelikler

Öznitelik	**Öznitelik Grubu**
Alan Adı Benzerliği	*URL*
URL Uzunluğu	*URL*
HTTP Yanıt Geçmişi	*HTTP*
Yeniden Yönlendirme	*HTTP*

NB algoritmasının kullanıldığı *HTTP* grubu deneyinde FP ve FN sonuçları incelendiğinde FP değerinin 22 ve FN değerinin 1 olduğu gözlemlenir. FP değeri, aslında yasal web sitesi olan web sitelerin oltalama web sitesi olarak sınıflandırıldığını ve FN değeri ise aslında oltalama web sitesi olan web sitelerinin yasal web sitesi olarak sınıflandırıldığını gösterir. Bu çalışmada, amaç oltalama web sitelerini tespit etmek olduğu için FN değerinin düşük olması çalışma için önem arz etmektedir. Bu deneysel sonuç, *HTTP* grubu özniteliklerinin oltalama saldırılarının tespitinde oynadığı önemi göstermiştir.

NB algoritması kullanılarak, *URL + HTTP* deney grubunda yapılan kapsamlı arama sonucunda, yalnızca dört adet öznitelik ile yüksek başarımlı bir sonuç elde edilmiştir. Sadece dört adet özniteliğin yüksek başarım için yeterli olması, oltalama web sitelerinin tespiti için özniteliklerin sınıflandırıcı ile olan etkisini ortaya çıkarmıştır. Teorik olarak, NB algoritmasının sınıflandırıcı modelinde yüksek başarımlı olabilmesi için, kullanılan özniteliklerin birbirleri ile olan korelasyonunun düşük olduğu kabullendiğinden dolayı, tüm öznitelikler için, Pearson korelasyon katsayıları hesaplanarak bir matris tablosu oluşturulmuştur. Pearson korelasyonu katsayısı, iki öznitelik arasında doğrusal bir ilişki olup olmadığını belirler (Benesty vd., 2009). İki farklı öznitelik arasında ilişki varsa, ilişkinin yönünü ve şiddetini belirler. Korelasyon katsayısı -1 ile 1 değerleri arasında değişir. İki öznitelik arasındaki ilişki negatif ise, özniteliklerden birinin değeri artarken, diğerinin değerinin azaldığını gösterir. Bir özniteliğin kendisi ile korelasyonu sayısal olarak 1.0' dır. Birbiri ile ilişkisi olmayan iki özniteliğin korelasyon katsayısı 0(sıfır)dır.

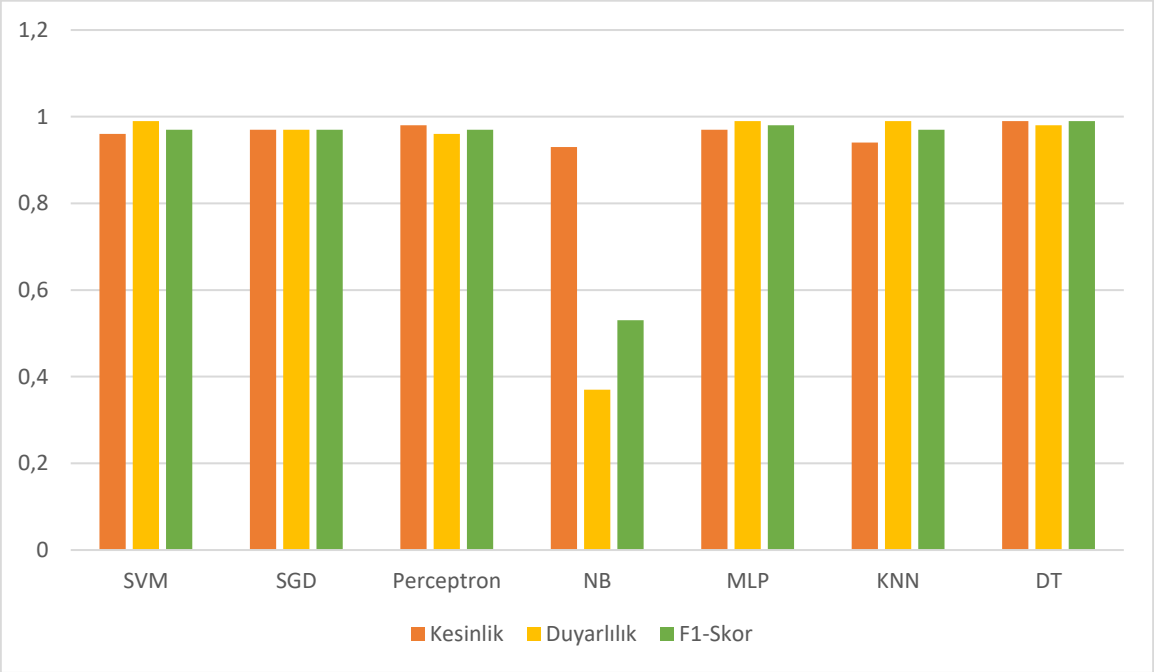
Veri setindeki özniteliklerin birbiri ile olan ilişkilerini anlamak için, eğitim veri setinin Pearson korelasyon matrisi hesaplanmıştır. Korelasyon hesaplamasını yapmak için Pandas kütüphanesinden (Pandas, 2008) yararlanılarak, işlemler sonucunda elde edilen Pearson korelasyon matrisi Çizelge 5.4.' de verilmiştir. Korelasyon matrisi incelendiğinde, içerisinde veri olmayan altı URL özniteliğinin, korelasyon katsayısı olmadığı görülmektedir. Veri setinde bu özniteliklerin gereksiz olduğu korelasyon matrisi yardımı ile görülmüştür.

NB sınıflandırıcısı ile HTTP özniteliklerinin F1-skor değerlendirmesine göre %93 başarım oranına sahiptir. HTTP öznitelikleri olan "yanıt geçmişi" ve "yeniden yönlendirme" özniteliklerinin, birbirleri ile olan Pearson korelasyon katsayıları 0,02' dir. Aynı grup içerisinde yer almalarına rağmen, korelasyon katsayılarının sıfıra yakın olması nedeni ile, NB algoritmasında, bu iki özniteliğin, oltalama web sitelerinin tespitinde oldukça başarılı olduğu görülmektedir. Çizelge 5.4. incelendiğinde HTML ve HTTP grubu özniteliklerinin korelasyon katsayılarının yüksek olduğu ve bu nedenle NB sınıflandırma sonucunu olumsuz etkilediği görülmüştür. URL özniteliklerinden altı adetinin veri içermemesi ve bazı özniteliklerinin korelasyonunun yüksek olması, URL özniteliklerinin başarımı olumsuz etkilemesine neden olmuştur.

Çizelge 5.4. Pearson korelasyon matrisi

		iFrame sayısı	"link" etiketi sayısı	düğme etiketi sayısı	giriş etiketi sayısı	"a" etiketi sayısı	yeniden yönlendirme	yanıt geçmişi	IP içermesi	"@" işareti sayısı	artı işareti sayısı	yıldız işareti sayısı	tilda işareti sayısı	küçüktür ve büyüktür işareti sayısı	köşeli ayraç sayısı	küme ayracı sayısı	parantez işareti sayısı	eşittir işareti sayısı	alt tire sayısı	kısa çizgi sayısı	çift eğik çizgi sayısı	eğik çizgi sayısı	nokta sayısı	http protokolü	url uzunluğu	alan adı benzerliği
URL	alan adı benzerliği	0,50	0,53	0,43	0,38	0,54	-0,63	-0,48	-0,39	-0,32			-0,05					-0,64	-0,42	-0,28	-0,24	-0,86	-0,32	0,54	-0,64	1,00
	url uzunluğu	-0,69	-0,72	-0,68	-0,61	-0,78	0,42	0,74	0,17	0,12			-0,02					0,56	0,26	0,88	0,14	0,69	0,79	-0,50	1,00	-0,64
	http protokolü	0,46	0,58	0,51	0,36	0,54	-0,28	-0,69	-0,42	-0,28			-0,05					-0,27	-0,32	-0,26	-0,14	-0,59	-0,30	1,00	-0,50	0,54
	nokta sayısı	-0,45	-0,46	-0,45	-0,37	-0,49	0,22	0,48	0,23	0,20			-0,09					0,28	0,04	0,75	0,04	0,34	1,00	-0,30	0,79	-0,32
	eğik çizgi sayısı	-0,53	-0,59	-0,49	-0,42	-0,60	0,44	0,56	0,33	0,11			0,08					0,39	0,22	0,35	0,32	1,00	0,34	0,69	-0,59	-0,86
	çift eğik çizgi sayısı	-0,21	-0,24	-0,21	-0,20	-0,25	-0,02	0,23	-0,03	-0,06			-0,04					0,10	-0,03	0,02	1,00	0,32	0,04	-0,14	-0,14	-0,24
	kısa çizgi sayısı	-0,59	-0,59	-0,58	-0,55	-0,67	0,24	0,65	-0,03	-0,03			-0,03					0,36	0,04	1,00	0,02	0,35	0,75	-0,26	0,88	-0,28
	alt tire sayısı	-0,19	-0,18	-0,22	-0,17	-0,11	0,21	0,15	0,09	0,02			-0,08					0,51	1,00	0,04	-0,03	0,22	0,04	-0,32	0,26	-0,42
	eşittir işareti sayısı	-0,40	-0,40	-0,37	-0,38	-0,40	0,54	0,24	0,18	0,36			-0,10					1,00	0,51	0,36	0,10	0,39	0,28	-0,27	0,56	-0,64
	parantez işareti sayısı																									
	küme ayracı sayısı																									
	köşeli ayraç sayısı																									
	küçüktür ve büyüktür işareti sayısı																									
	tilda işareti sayısı	-0,10	-0,12	-0,10	-0,09	-0,12	-0,05	0,09	-0,06	-0,08			1,00					-0,10	-0,08	-0,03	-0,04	0,08	-0,09	-0,05	-0,02	-0,05
	yıldız işareti sayısı																									
	artı işareti sayısı																									
	"@" işareti sayısı	-0,21	-0,22	-0,21	-0,13	-0,23	0,15	0,15	0,66	1,00			-0,08					0,36	0,02	-0,03	-0,06	0,11	0,20	-0,28	0,12	-0,32
	IP içermesi	-0,24	-0,27	-0,24	-0,14	-0,26	0,11	0,23	1,00	0,66			-0,06					0,18	0,09	-0,03	-0,03	0,33	0,23	-0,42	0,17	-0,39
HTTP Yanıtı	yanıt geçmişi	-0,69	-0,74	-0,71	-0,55	-0,80	0,02	1,00	0,23	0,15			0,09					0,24	0,15	0,65	0,23	0,56	0,48	-0,69	0,74	-0,48
	yeniden yönlendirme	-0,36	-0,31	-0,27	-0,30	-0,34	1,00	0,02	0,11	0,15			-0,05					0,54	0,21	0,24	-0,02	0,44	0,22	-0,28	0,42	-0,63
HTML	"a" etiketi sayısı	0,79	0,69	0,71	0,66	1,00	-0,34	-0,80	-0,26	-0,23			-0,12					-0,40	-0,11	-0,67	-0,25	-0,60	-0,49	0,54	-0,78	0,54
	giriş etiketi sayısı	0,48	0,45	0,54	1,00	0,66	-0,30	-0,55	-0,14	-0,13			-0,09					-0,38	-0,17	-0,55	-0,20	-0,42	-0,37	0,36	-0,61	0,38
	düğme etiketi sayısı	0,58	0,69	1,00	0,54	0,71	-0,27	-0,71	-0,24	-0,21			-0,10					-0,37	-0,22	-0,58	-0,21	-0,49	-0,45	0,51	-0,68	0,43
	"link" etiketi sayısı	0,67	1,00	0,69	0,45	0,69	-0,31	-0,74	-0,27	-0,22			-0,12					-0,40	-0,18	-0,59	-0,24	-0,59	-0,46	0,58	-0,72	0,53
	iFrame sayısı	1,00	0,67	0,58	0,48	0,79	-0,36	-0,69	-0,24	-0,21			-0,10					-0,40	-0,19	-0,59	-0,21	-0,53	-0,45	0,46	-0,69	0,50

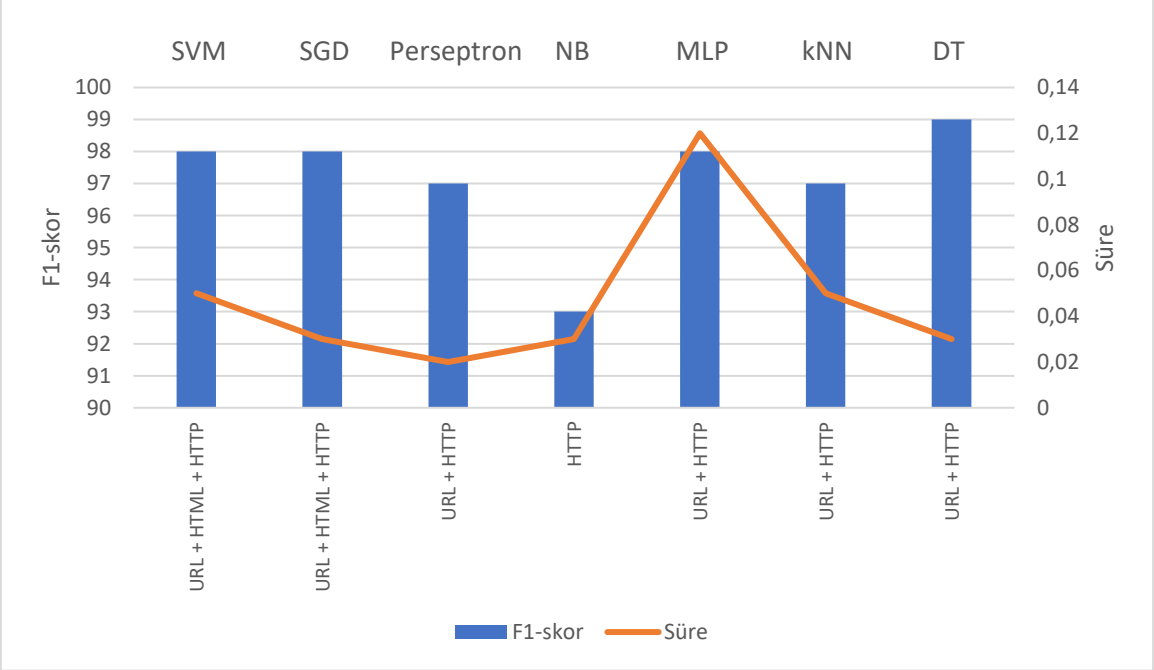
Yukarıdaki deneylerdeki elde edilen sonuçlara göre, *URL + HTTP* öznitelik grubunun sonuçlarını daha iyi analiz edebilmek için, kesinlik, duyarlılık ve F1-skor oranları çizilerek, Şekil 5.1.' de gösterilmiştir. Şekil 5.1. incelendiğinde, NB algoritması sonuçlarında büyük düşüş görülmektedir. Bu durumun nedeni, URL özniteliklerinin, HTTP öznitelikleri ile olan korelasyon katsayılarının büyük olmasından dolayı sonuçları olumsuz olarak etkilemesidir. Yapılan çalışmada, öznitelik seçimi grup bazında yapıldığından dolayı NB algoritmasında düşük değer alınmıştır. Diğer tüm sınıflandırma algoritmaları ile yapılan deneyler incelendiğinde, *URL + HTTP* öznitelik grubunun oltalama web sitelerini tespit etmek için iyi bir öznitelik grubu olduğu anlaşılmıştır. Deney sonuçları incelendiğinde en iyi başarım DT algoritması ile elde edilmiştir.



Şekil 5.1. *URL + HTTP* öznitelik grubu kesinlik, duyarlılık ve F1-skor değerleri

F1-skor, duyarlılık ve kesinlik ölçümlerinin harmonik ortalamasından oluşmaktadır. FN ve FP değerlerinin sınıflandırma sonuçlarını etkilediği durumlarda kullanılır. Şekil 5.2.' de yer alan değerler Çizelge 5.2.' den alınmıştır. Şekil 5.2.' de her bir algoritma için en yüksek F1-skorunu veren öznitelik grupları ve sınıflandırma süreleri verilmiştir. SVM ve SGD algoritmalarında, %98'lik F1-skor değerleri, tüm özniteliklerin kullanıldığı deney grubu *URL + HTML + HTTP* ile elde edilmiştir. SVM ve SGD algoritmaları ile alınan sonuçlar incelendiğinde, SGD algoritması daha kısa bir sürede sınıflandırma yapmaktadır.

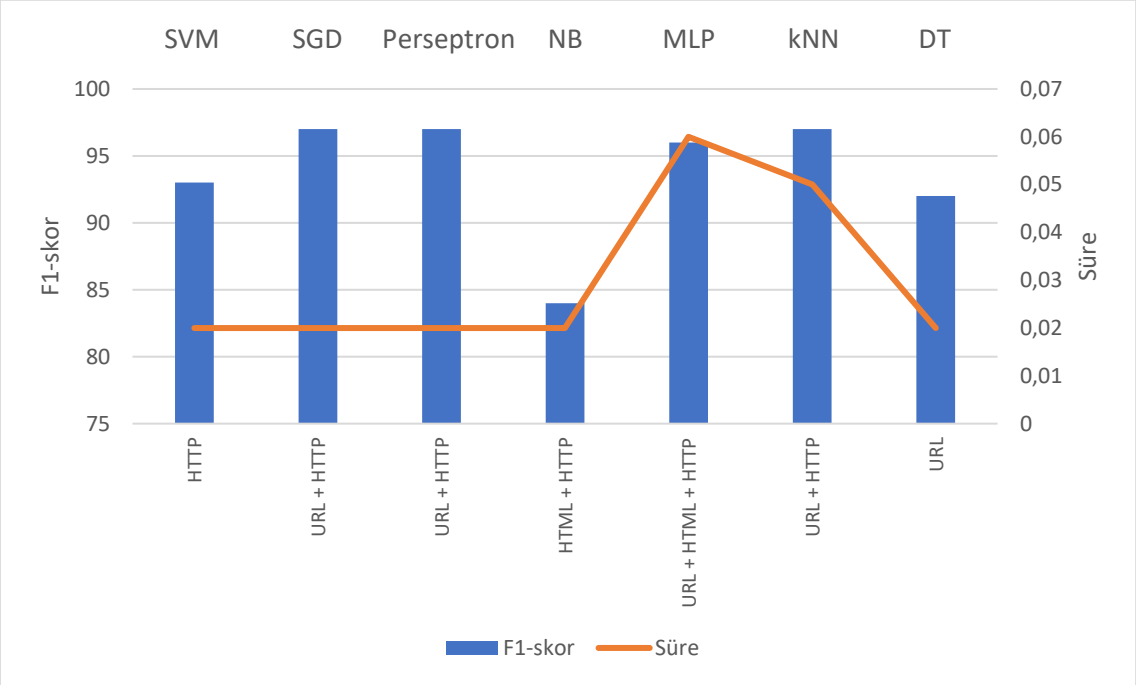
Perseptron algoritması *URL + HTML + HTTP* ve *URL + HTTP* gruplarında aynı F1-skor ve sınıflandırma süresi sonuçları almıştır. Şekil 5.2. incelendiğinde Perseptron algoritmasının daha hızlı bir sınıflandırıcı olduğu görülmektedir. NB algoritması en düşük F1-skoruna sahip olsa da, NB algoritması düşük korelasyona sahip özniteliklerin bir arada kullanılmasıyla daha güçlü hale gelerek, sadece *HTTP* grubundaki öznitelikleri kullanması, *HTTP* grubu özniteliklerinin oltalama saldırılarının tespitinde önemli rol oynadığını göstermektedir. Perseptron, MLP, kNN ve DT algoritmaları en iyi F1-skorunu *URL + HTTP* öznitelik grubunda elde etmektedir. Perseptron, MLP, kNN ve DT algoritmaları sonuçları Şekil 5.2.' de incelendiğinde, DT algoritmasının sınıflandırma hızının en kısa olması ve F1-skorunun yüksek olması nedeni ile Perseptron, kNN ve MLP algoritmalarından daha başarılı olduğu görülmektedir. Süre grafiği incelendiğinde, MLP algoritmasının yavaşlığı belirgin bir şekilde görülmektedir.



Şekil 5.2. En yüksek F1-skor değerlerine karşılık gelen sınıflandırma süreleri

Sınıflandırma süresinin kısa olması, oltalama saldırılarında önemli olduğu için, en kısa sürede alınan en iyi F1-skor değerleri Çizelge 5.2.' den alınarak, Şekil 5.3. oluşturulmuştur. Şekil 5.3.' de her bir algoritmanın en kısa sınıflandırma süresinde hangi öznitelik deney grubunda en iyi F1-skor aldığı değeri yer almaktadır. Sınıflandırma süreleri

incelendiğinde MLP ve kNN algoritmalarının yüksek F1-skorlarına rağmen oldukça yavaş çalıştığı gözlemlenmiştir. Aynı sınıflandırma süresine sahip SVM, SGD, Perseptron, NB ve DT algoritmalarının F1-skor başarım oranları incelendiğinde, düşük sınıflandırma sürelerine, bir diğer ifadeyle hızlı çalışmalarına rağmen SGD ve Perseptron algoritmaları daha yüksek başarım değerlerine sahip olmuşlardır.



Şekil 5.3. En kısa sınıflandırma sürelerine karşılık gelen F1-skor değeri

5.2. Deney Sonuçları ve Literatürle Karşılaştırılması

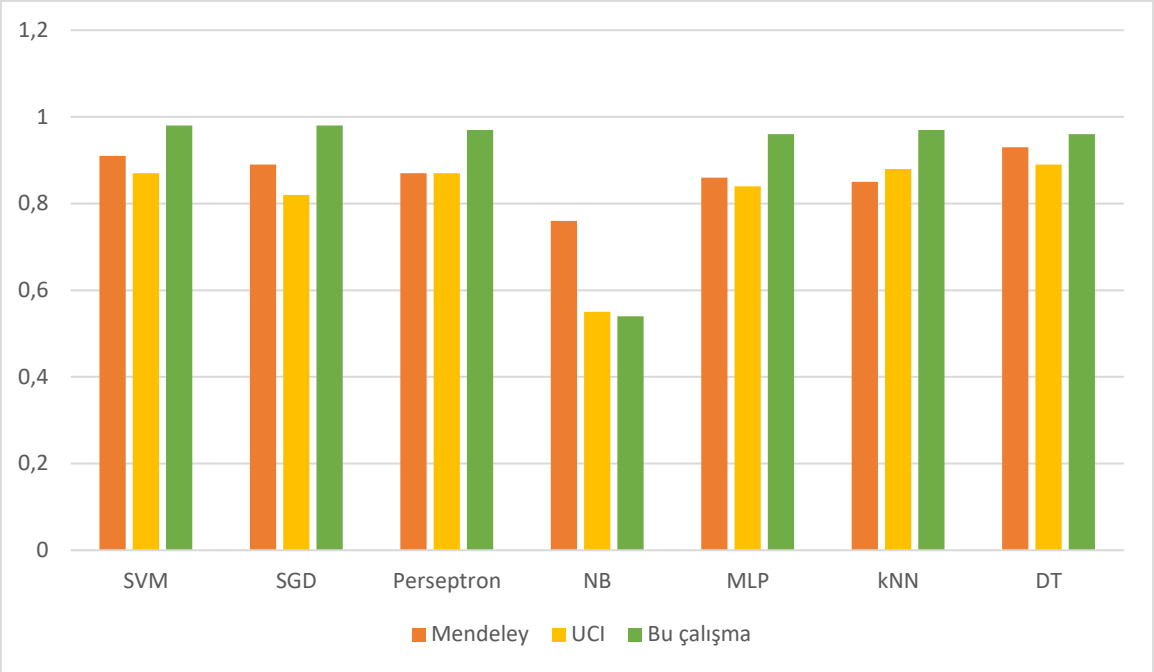
Hazırlanan veri setinin oltalama saldırında kullanılan web sitelerini tespit etmekteki başarısını ölçmek için literatürde sıklıkla kullanılan UCI (Muhammed vd., 2015a) ve güncel Mendeley (Tan, 2018) veri setleri ile karşılaştırılmıştır. Veri setlerinin farklı miktarda örnek içermesi nedeni ile, veri setlerinin örnek miktarı eşitlenmiştir. Her bir veri setinden 500 oltalama ve 500 yasal web sitesi örneği rastgele seçilmiştir. Deney veri setlerinde, öznitelik seçimi yapılmadan uygulanmıştır. Veri seti, %70 eğitim ve %30 test veri seti olarak ayrılmıştır. Eğitim ve test veri setlerinde sınıflar eşit miktardadır. Deney sonuçları Çizelge 5.5.' de ayrıntılı olarak verilmiştir. Deney sonuçları incelendiğinde, hazırlanan veri seti NB

algoritması dışında diğer veri setlerinden daha iyi başarım oranına sahip olduğu gözlemlenmiştir.

Çizelge 5.5. Veri setlerinin karşılaştırılması

SA	Veri Seti	TÖS	Acc	Pre	Rec	F1-S	TN	FP	FN	TP
SVM	*Mendeley*	48	0,91	0,90	0,91	0,91	135	15	13	137
	UCI	30	0,87	0,87	0,87	0,87	131	19	19	131
	Bu çalışma	25	0,98	0,97	0,99	**0,98**	145	5	1	149
SGD	*Mendeley*	48	0,89	0,90	0,88	0,89	136	14	18	132
	UCI	30	0,84	0,90	0,75	0,82	138	12	37	113
	Bu çalışma	25	0,98	0,98	0,97	**0,98**	147	3	4	146
Perseptron	*Mendeley*	48	0,88	0,90	0,85	0,87	135	15	22	128
	UCI	30	0,87	0,87	0,87	0,87	130	20	19	131
	Bu çalışma	25	0,97	0,96	0,97	**0,97**	144	6	4	146
NB	*Mendeley*	48	0,81	0,99	0,62	**0,76**	149	1	57	93
	UCI	30	0,68	0,97	0,38	0,55	148	2	93	57
	Bu çalışma	25	0,68	0,99	0,38	0,54	147	3	93	57
MLP	*Mendeley*	48	0,86	0,86	0,86	0,86	129	21	21	129
	UCI	30	0,84	0,84	0,83	0,84	126	24	25	125
	Bu çalışma	25	0,96	0,97	0,95	**0,96**	145	5	8	142
kNN	*Mendeley*	48	0,85	0,85	0,86	0,85	127	23	21	129
	UCI	30	0,88	0,88	0,89	0,88	131	19	17	133
	Bu çalışma	25	0,97	0,99	0,98	**0,97**	143	7	3	147
DT	*Mendeley*	48	0,93	0,93	0,93	0,93	140	10	10	140
	UCI	30	0,9	0,95	0,84	0,89	143	7	24	126
	Bu çalışma	25	0,96	0,98	0,94	**0,96**	147	3	9	141

Şekil 5.4.' de veri setlerinin F1-skor değerleri karşılaştırılmıştır. Hazırlanan veri seti, diğer veri setleri ile karşılaştırıldığında yüksek F1-skor değerine sahiptir. NB algoritmasının diğer veri setlerinde de başarım oranının düşük olması, NB sınıflandırıcısının oltalama saldırılarında kullanılan web sitelerini tespit etmede başarısız olduğu gözlemlenmiştir. Ayrıca, DT algoritmasının üç veri setinde de yüksek başarım oranına sahip olduğu gözlemlenmiştir. Bu deney sonucunda, hazırlanan veri setinin, oltalama web sitelerini tespit etmekteki başarısı ölçülmüştür. Ayrıca, DT algoritmasının oltalama web sitelerinin tespitindeki başarısı gösterilmiştir.



Şekil 5.4. Veri setlerinin F1-skor değerleri

Çizelge 5.6. Literatür karşılaştırması

Literatür	**ÖS**	**SA**	**Acc**	**Pre**	**Rec**	**F1-S**	**TP**	**TN**	**FP**	**TPR**	**TNR**	**FPR**	**FNR**	**ER**	**CC**	**IC**
(Zhang Y. vd., 2007)	-	TF-IDF	-	-	-	-	0,94	-	-	-	-	-	-	-	-	-
(Wardman vd., 2011)	-	-	-	-	-	-	0,95	-	0,14	-	-	-	-	-	-	-
(Ramanathan ve Wechsler, 2013)	-	AdaBoost	-	0,96	0,96	0,96	-	-	-	0,96	-	0,04	-	-	-	-
(Li vd., 2013)	-	TSVM	0,96	0.96	0,91	-	-	-	-	-	-	-	-	-	-	-
(Mao vd., 2017)	-	-	-	1	0,98	0,99	-	-	-	-	-	-	-	-	-	-
(Zhang H. vd., 2011)	-	-	-	-	-	0,99	-	-	-	-	-	-	-	-	-	-
(Mohammad, Thabtah ve McCluskey, 2014)	-	-	-	-	-	-	-	-	-	-	-	-	-	0,05	-	-
(Basnet vd., 2011)	-	-	-	-	-	-	-	-	-	0,92	0,98	0,02	0,08	-	-	-
(Moghimi ve Varjani, 2016)	-	-	0,98	-	-	0,99	0,99	0,97	0,02	-	-	-	-	-	-	-
(Fette vd., 2007)	-	-	0,99	-	-	-	-	-	-	0,96	-	0,001	0,03	-	-	-
(Aburrous vd., 2009)	-	PART	-	-	-	-	-	-	-	-	-	-	-	-	0,86	0,14
(Chiew vd., 2019)	**HEFS**	**RF**	**0,94**	-	-	-	-	-	-	-	-	-	-	-	-	-
(Sahingoz vd., 2019)	-	RF	0,98	-	0,97	-	0,98	-	-	-	-	-	-	-	-	-
(Sonowal ve Kuppusamy, 2020)	-	-	0,93	-	-	-	-	-	-	0,91	0,94	006	0,09	-	-	-
Bu çalışma	**ES**	**DT**	**0,99**	**0,99**	**0.98**	**0,99**	-	-	-	-	-	-	-	-	-	-

Bu tez çalışmasını literatürle karşılaştırma amaçlı, Çizelge 5.6. oluşturulmuştur. Son yıllarda yapılan Chiew vd. (2019) çalışması, Mendeley veri setini kullanmaktadır ve RF

algoritmasının oltalama web sitelerinin tespitinde başarılı sonuçlar verdiğini göstermiştir. Ayrıca, yapılan deneylerde DT algoritmasının gösterdiği başarı, ağaç yapısını kullanan algoritmaların oltalama web sitelerinin tespitinde önemli bir başarıya sahip olduğu görülmektedir. Chiew vd. (2019) tarafından yapılan çalışmanın sonuçları, bu çalışmada yapılan deney sonuçları ile karşılaştırıldığında, bu tez çalışmasında daha yüksek bir başarım oranının elde edildiğini göstermiştir. Bu tez çalışmasında hazırlanan veri seti ve Mendeley veri setinin URL ve HTML özniteliklerini içermesi, veri kümesinin karşılaştırılmasını sağlamıştır. Bu çalışmada hazırlanan veri kümesi URL ve HTML özniteliklerinin dışında HTTP yanıtını özniteliklerine de sahiptir.

6. SONUÇ VE ÖNERİLER

Oltalama saldırılarının günümüzde oldukça etkili ve fazla olması bu saldırılardan korunmanın yollarının aranmasına neden olmuştur. Oltalama saldırıları için kullanılan oltalama web sitelerinin tespit edilmesi için farklı yöntemler kullanılıyor olsa da, etkili bir yöntem bulunamamıştır. Oltalama web sitelerinin sürekli olarak değişmesi, sıfır gün saldırılarında da etkili olan sezgisel yöntemler kullanılarak yapılan tespit yöntemlerini öne çıkarmıştır. Bu tez çalışmasında, oltalama web sitelerini tespit etmek için yeni bir veri kümesi hazırlanmıştır. Veri kümesi PhishTank ve Alexa arşivleri kullanılarak hazırlanmıştır. URL, HTML ve HTTP yanıtı özniteliği olarak ayrılan toplam 25 öznitelik vardır. Bu öznitelik grupları kullanılarak "kapsamlı arama yöntemi" ile en iyi ayırt edici öznitelik grubu aranmıştır. URL ve HTTP yanıtı özniteliklerinin birlikte kullanılması ile daha iyi bir sınıflandırma yapılmıştır. Veri kümesi ile yapılan çalışmada URL ve HTTP yanıtı öznitelikleri DT algoritması ile test edildiğinde %99 F1-skor oranı ile en yüksek başarım oranına sahiptir.

Bu tez kapsamında hazırlanan veri setinde, kullanılan öznitelik gruplarının sınıflandırma algoritmalarındaki ayırt edici etkileri incelenmiş olup, NB algoritması kullanıldığında, farklı öznitelik gruplarının oluşturduğu olumsuz etkilerin nedenleri, özniteliklerin birbirleri ile olan benzerlik ya da ayrılık açısından korelasyon matrisi ile araştırılmıştır. Elde edilen karşılaştırmalı korelasyon matrisi tablosu ile de, yüksek korelasyona sahip öznitelik çiftlerinin bulunduğu kombinasyonların, NB algoritmasının çalışma prensibinden dolayı sınıflandırıcıyı olumsuz etkilediği kanıtlanmıştır. Hazırlanan veri setindeki başarım oranları, literatürde sıklıkla kullanılan açık kaynak olarak erişilebilen veri setleri ile karşılaştırmalı olarak gösterilmiştir.

İlerleyen çalışmalarda, HTTP yanıtı özniteliklerin çeşitliliği arttırarak literatürdeki önemi vurgulanacaktır. Ayrıca URL, HTML ve HTTP yanıtı özniteliklerinin web sitelerinden toplanma süresi tespit edilerek, özniteliklerin toplanma hızının etkileri ile sınıflandırma maliyetleri birlikte ele alınarak maliyet analizi incelenebilir.

KAYNAKLAR DİZİNİ

3Sharps, 2006, 3Sharps, http://www.3sharp.com/projects/antiphishing/, erişim tarihi:-

Aburrous, M., Hossain, M., Dahal, K. ve Thabtah, F., 2009, Modelling Intelligent Phishing Detection System for E-banking Using Fuzzy Data Mining, p. 265–272.

Aburrous, M., Hossain, M., Dahal, K. ve Thabtah, F., 2010a, Intelligent phishing detection system for e-banking using fuzzy data mining, Expert Systems with Applications, p. 7913–7921.

Aburrous, M., Hossain, M., Dahal, K. ve Thabtah, F., 2010b, Predicting Phishing Websites Using Classification Mining Techniques with Experimental Case Studies., p. 176–181.

Al Banna, M. H., Taher, K. A., Kaiser, M. S., Mahmud, M., Rahman, M. S., Hosen, A. S., ve Cho, G. H., 2020, Application of artificial intelligence in predicting earthquakes: state-of-the-art and future challenges, IEEE Access, p. 192880-192923.

Alexa, 1996, https://www.alexa.com/, erişim tarihi:19.11.2020.

Alpaydin, E., 2020, Introduction to machine learning, MIT press.

Apache, 2003, SpamAssassin, https://spamassassin.apache.org/, erişim tarihi:19.11.2020.

APWG, 2003, Anti-Phishing Working Group, https://apwg.org/, erişim tarihi:19.11.2020.

APWG Reports, 2017, Phishing Attack Trends Report – 4Q 2016, https://docs.apwg.org//reports/apwg_trends_report_q4_2016.pdf, erişim tarihi:19.11.2020.

APWG Reports, 2020, Phishing Attack Trends Report – 4Q 2019, https://docs.apwg.org/reports/apwg_trends_report_q4_2019.pdf, erişim tarihi:24.11.2020.

Basnet, R., Sung, A. ve Liu, Q., 2011, Rule-Based Phishing Attack Detection, Proceedings of the International Conference on Security and Management (SAM).

Bayraktar, B., 2019, Rastgele Ormanlar ve Aşırı Öğrenme Makineleri Teknikleri ile Oltalama Saldırılarının Tespiti, Yüksek Lisans tezi, İstanbul Üniversitesi-Cerrahpaşa Lisansüstü Eğitim Enstitüsü, 102 s.

Benesty, J., Chen, J., Huang, Y., ve Cohen, I., 2009, Pearson correlation coefficient. In Noise reduction in speech processing, p.1-4.

Bishop, C. M., 2006, Pattern recognition and machine learning, Springer.

Bottou, L., 2012, Stochastic gradient descent tricks, Neural networks: Tricks of the trade, Springer, p. 421-436.

KAYNAKLAR DİZİNİ (devam)

Büber, E., 2017, Oltalama Saldırılarında Kullanılan Url'lerin Makine Öğrenmesi Teknikleri ile Tespit Edilmesi, Yüksel Lisans tezi, Marmara Üniversitesi Fen Bilimleri Enstitüsü, 79 s.

Chiew, K. L., Tan, C. L., Wong, K., Yong, K. S., ve Tiong, W. K., 2019, A new hybrid ensemble feature selection framework for machine learning-based phishing detection system, Information Sciences, p.153-166.

Chiew, K. L., Yong, K. S. C., ve Tan, C. L., 2018, A survey of phishing attacks: Their types, vectors and technical approaches, Expert Systems with Applications, p. 1-20.

Dou, Z., Khalil, I., Khreishah, A., Al-Fuqaha, A. ve Guizani, M., 2017, Systematization of Knowledge (SoK): A Systematic Review of Software-Based Web Phishing Detection, IEEE Communications Surveys & Tutorials, p. 2797-2819.

Fette, I., Sadeh, N. ve Tomasic, A., 2007, Learning to Detect Phishing Emails, Proceedings of the 16th International Conference on World Wide Web. Association for Computing Machinery, p. 649–656.

Google Inc., 1998, Google, https://www.google.com/, erişim tarihi:19.11.2020.

Google Inc., 2008, Chrome, https://www.google.com/chrome/, erişim tarihi:19.11.2020.

Google Inc., 2007, Google Safe Browsing API, https://safebrowsing.google.com/, erişim tarihi: 18.01.2021.

Jackson B., 2016, What Is the Difference Between HTTP and HTTPS?, https://www.keycdn.com/blog/difference-between-http-and-https, erişim tarihi:19.11.2020.

Jakobsson, M., 2007, The human factor in phishing. Privacy & Security of Consumer Information, p. 1-19.

Kaytan, M., 2016, Web Tabanlı Oltalama Saldırılarının Makine Öğrenmesi Yöntemleri ile Tespiti, Yüksek Lisans tezi, İnönü Üniversitesi Fen Bilimleri Enstitüsü, 100 s.

Li, Y., Xiao, R., Feng, J. ve Zhao, L., 2013, A semi-supervised learning approach for detection of phishing webpages, Optik, p. 6027–6033.

Mao, J., Tian, W., Li, P., Wei, T. ve Liang, Z., 2017, Phishing-Alarm: Robust and Efficient Phishing Detection via Page Component Similarity, IEEE Access, p. 17020–17030.

Maurer, M.E., 2012, Phishload, http://www.medien.ifi.lmu.de/team/max.maurer/files/phishload/, erişim tarihi:19.11.2020.

Millersmiles, 2003, http://www.millersmiles.co.uk/, erişim tarihi:19.11.2020.

KAYNAKLAR DİZİNİ (devam)

Moghimi, M. ve Varjani, A.Y., 2016, New rule-based phishing detection method, Expert Systems with Applications, p. 231–242.

Mohammad, R.M., Thabtah, F. ve McCluskey, L., 2012, An assessment of features related to phishing websites using an automated technique, p. 492–497.

Mohammad, R.M., Thabtah, F. ve McCluskey, L., 2014, Intelligent rule-based phishing websites classification, IET Information Security, p. 153–160.

Mohammad, R.M., Thabtah, F. ve Mccluskey, L., 2014, Predicting Phishing Websites Based on Self-Structuring Neural Network, Neural Computing and Applications, p. 443–458.

Mohammad, R.M., Thabtah, F. ve McCluskey, L., 2015, Phishing Websites Data Set, https://archive.ics.uci.edu/ml/datasets/phishing+websites, erişim tarihi:19.11.2020.

Mohammad, R.M., Thabtah, F. ve McCluskey, ,2015, L., Tutorial and critical analysis of phishing websites methods, Computer Science Review, p. 1-24.

Mozilla Foundation, 2002, Firefox, https://www.mozilla.org/tr/firefox/new/, erişim tarihi:19.11.2020.

Netcraft, 1994, Netcraft Anti-Phishing Toolbar, https://www.netcraft.com/apps/, erişim tarihi:19.11.2020.

OpenPhish, 2020. https://openphish.com/, erişim tarihi:19.11.2020.

OpenPhish, 2020, The Global Phishing Activity, https://openphish.com/phishing_activity.html, erişim tarihi:24.11.2020.

PhishingCorpus, 2006, PhishingCorpus, http://monkey.org/~jose/wiki/doku.php?id =PhishingCorpus, erişim tarihi: -.04.2006.

PhishTank, 2006, PhishTank, https://www.phishtank.com/, erişim tarihi:19.11.2020.

Python Documentation, 2001, difflib – Helper for computing deltas, Python Standart Library, https://docs.python.org/3/library/difflib.html, erişim tarihi: 18.01.2021.

Pandas, 2008, Pandas DataFrame Corr, https://pandas.pydata.org/pandas-docs/stable/reference/api/pandas.DataFrame.corr.html, erişim tarihi: 18.02.2021

Qabajeh, I., Thabtah, F., ve Chiclana, F.,2018, A recent review of conventional vs. automated cybersecurity anti-phishing techniques, Computer Science Review, p. 44-55.

Ramanathan, V. ve Wechsler, H., 2013, Phishing Detection and Impersonated Entity Discovery Using Conditional Random Field and Latent Dirichlet Allocation, Computers Security, p. 123–139.

KAYNAKLAR DİZİNİ (devam)

Ratcliff, J. W. ve Metzener, D. E., 1988, Pattern-matching-the gestalt approach, Dr Dobbs Journal, p. 46.

Ren, J., Lee, S.D., Chen, X., Kao, B., Cheng, R. ve Cheung, D., 2009, Naive Bayes Classification of Uncertain Data, 2009 Ninth IEEE International Conference on Data Mining, p. 944-949.

Ruck, D.W., Rogers, S.K ve Kabrisky, M., 1990, Feature selection using a multilayer perceptron, Journal of Neural Network Computing, p. 40-48.

Sahingoz, O. K., Buber, E., Demir, O. ve Diri, B., 2019, Machine learning based phishing detection from URLs, Expert Systems with Applications, p. 345–357.

Scikit-Learn, 2013, Scikit-Learn, https://scikit-learn.org/stable/index.html, erişim tarihi:19.11.2020.

Scikit-Learn Classification, 2013, Multi-layer Perceptron, https://scikitlearn.org/stable/modules/neural_networks_supervised.html, erişim tarihi: 17.02.2021

Selenium, 2020, Selenium Web Driver, https://www.selenium.dev/, erişim tarihi:19.11.2020.

Sonowal, G. ve Kuppusamy, K.S., 2020, PhiDMA – A phishing detection model with multi-filter approach, Journal of King Saud University - Computer and Information Sciences, p. 99–112.

SpamsAssassin, 2004, SpamsAssassin Public Mail Corpus, https://spamassassin.apache.org/old/publiccorpus/, erişim tarihi:19.11.2020.

Stanford Applied Crypto Group, 2006, SpoofGuard, https://crypto.stanford.edu/SpoofGuard/, erişim tarihi:19.11.2020.

Tan, Choon Lin, 2018, Phishing Dataset for Machine Learning: Feature Evaluation, http://dx.doi.org/10.17632/h3cgnj8hft.1, erişim tarihi:19.11.2020.

Thabtah, F., Cowling, P. ve Peng, Y., 2005, MCAR: Multi-class Classification based on Association Rule, p. 33.

The Common Crawl Foundation, 2011, Common Crawl, http://commoncrawl.org/, erişim tarihi:19.11.2020.

Theodoridis S., Koutroumbas K., 2009, Pattern Recognition (Fourth Edition), Academic Press, p. 91-150.

Turhanlar, M., 2019, Detecting Turkish Phishing Attacks With Machine Learning Classifiers, MSc. Thesis, The Graduate School of Informatıcs of The Middle East Technical University, 96 p.

KAYNAKLAR DİZİNİ (devam)

Tyagi, I., Shad, J., Sharma, S., Gaur, S., ve Kaur, G., 2018, A novel machine learning approach to detect phishing websites, 2018 5th international conference on signal processing and integrated networks (SPIN), p. 425-430.

UAB, 1994, UAB Phishing Data Mine, https://www.uab.edu/it/home/security/awareness#phish, erişim tarihi:19.11.2020.

University of Waikato, 1993, Waikato Environment for Knowledge Analysis, https://www.cs.waikato.ac.nz/ml/weka/, erişim tarihi:19.11.2020.

untroubled.org, 1998, SPAM Archive, http://untroubled.org/spam/, erişim tarihi:19.11.2020.

W3Schools, 1999, HTML Elements, https://www.w3schools.com/tags/, erişim tarihi:19.11.2020.

Wardman, B., Stallings, T., Warner, G. ve Skjellum, A., 2011, High-performance content based phishing attack detection, eCrime Researchers Summit, eCrime, p. 1–9.

Yahoo Inc., 1994, Yahoo!, https://www.yahoo.com/, erişim tarihi:19.11.2020.

Yandex, 2000, Yandex Search API, https://yandex.com.tr/dev/xml/, erişim tarihi:19.11.2020.

Zhang, H., Liu, G., Chow, T. ve Wenyin, L., 2011, Textual and Visual Content-Based Anti-Phishing: A Bayesian Approach, IEEE transactions on neural networks / a publication of the IEEE Neural Networks Council, p. 1532–46.

Zhang, Y., Hong, J. ve Cranor, L., 2007, CANTINA: A content based approach to detecting phishing web sites, p. 639–648.