T.C.

İSTANBUL KÜLTÜR ÜNİVERSİTESİ

LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ

İÇERİK TABANLI OLTALAMA SALDIRISI TESPİT SİSTEMİ

YÜKSEK LİSANS TEZİ

Uğur ÖZKER

0801020009

Anabilim Dalı: Bilgisayar Mühendisliği

Programı: Bilgisayar Mühendisliği

Tez Danışmanı: Prof. Dr. Özgür Koray ŞAHİNGÖZ

ŞUBAT 2021

T.C.

İSTANBUL KÜLTÜR ÜNİVERSİTESİ

LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ

İÇERİK TABANLI OLTALAMA SALDIRISI TESPİT SİSTEMİ

YÜKSEK LİSANS TEZİ

Uğur ÖZKER

0801020009

Anabilim Dalı: Bilgisayar Mühendisliği

Programı: Bilgisayar Mühendisliği

Tez Danışmanı: Prof. Dr. Özgür Koray ŞAHİNGÖZ

Jüri Üyeleri: Dr. Öğretim Üyesi Hakan AYDIN

Dr. Öğretim Üyesi Fatma Patlar AKBULUT

ŞUBAT 2021

ÖNSÖZ

"MAKİNE ÖĞRENMESİ YÖNTEMLERİ İLE İÇERİK TABANLI OLTALAMA SALDIRILARININ TESPİTİ" adlı yüksek lisans tez çalışmam süresince bilgi ve deneyimi ile çalışmalarımı yönlendiren ve desteğini esirgemeyen değerli tez danışmanım Prof. Dr. Özgür Koray Şahingöz'e, her durumda şartsız ve koşulsuz desteklerini ve sevgilerini benden esirgemeyen eşime ve aileme, katkıda bulunan tüm hocalarıma ve arkadaşlarıma teşekkürlerimi sunarım.

İÇİNDEKİLER

ŞEKİL LİSTESİ

TABLO LİSTESİ

KISALTMALAR

SVM	: Destek Vektör Makinesi Algoritması
K-NN	: K-En Yakın Komşu Algoritması
RF	: Rastgele Orman Algoritması
NB	: Naif Bayes Algoritması
DT	: Karar Ağacı Algoritması
MLP	: Çok Katmanlı Algılayıcı
CV	: Çapraz Doğrulama
CM	: Karışıklık Matrisi
LR	: Doğrusal Regresyon
XGBoost	: Aşırı Gradyan Tahminleme

Üniversite	**:**	**T.C. İstanbul Kültür Üniversitesi**
Enstitüsü	**:**	**Lisansüstü Eğitim Enstitüsü**
Anabilim Dalı	**:**	**Bilgisayar Mühendisliği**
Program	**:**	**Bilgisayar Mühendisliği**
Tez Danışmanı	**:**	**Prof. Dr. Özgür Koray ŞAHİNGÖZ**
Tez Türü ve Tarihi	**:**	**Yüksek Lisans – Şubat 2021**

ÖZET

İÇERİK TABANLI OLTALAMA SALDIRISI TESPİT SİSTEMİ

Son yıllarda internet teknolojilerinin kaçınılmaz büyümesi nedeniyle gerçek dünyadaki sistemlerin neredeyse tamamı dijital platformlara aktarılıyor. Bu, özellikle ilgili hizmetlere her zaman ve her yerde konsept ile bağlanmamızı sağlayan mobil cihazlarla hayatımızın her alanında siber uzay kullanımını artırıyor. Bununla birlikte, bu kaçınılmaz genişleme, özellikle standart son kullanıcılar için birçok güvenlik ihlali de beraberinde getirir. Kimlik avı, bilgisayar korsanlarının kendilerini kolayca engelleyerek kullandıkları en çok tercih edilen saldırı türlerinden biridir. Bu tür saldırı, başlangıçta basit bir e-posta veya sosyal medya mesajı ile tetiklenir ve bu mesaj, esas olarak kurbanları kötü niyetli bir web sayfasına yönlendirir. Güvenlik yöneticileri için tespit edilmesi gerçekten zor saldırı türleridir. Bu nedenle, bu makalede içerik tabanlı bir kimlik avı tespit mekanizması önerilmektedir. Teklifte, en iyi eğitim modellerini seçmek için altı farklı makine öğrenimi modeli uygulanmaktadır. Deneysel sonuçlar, önerilen yaklaşımın çok sağlam olduğunu ve güvenlik yöneticileri için kabul edilebilir doğruluklar verdiğini göstermektedir.

Anahtar Kelimeler: Makine öğrenimi, Güvenlik İhlalleri, Saldırılar, Kimlik Avı.

University : **T.C. İstanbul Kültür University**
Institute : **Institute of Graduate Studies**
Department : **Computer Engineering**
Program : **Computer Engineering**
Thesis Advisor : **Prof. Prof. Özgür Koray ŞAHİNGÖZ**
Degree Awerded And Date : **MA – February 2021**

ABSTRACT

CLASSIFICATION OF CONTENT BASED PHISHING ATTACKS BY MACHINE LEARNING METHODS

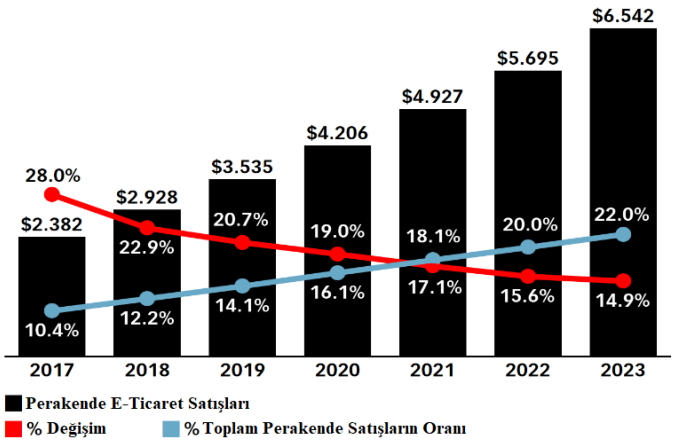
In recent years due to the inevitable growth of Internet technologies, almost all of the real world systems are transferred to digital platforms. This increases the use of cyberspace in every dimention of our lives especially with mobile devices which enable us to connect to related services in anytime and anywhere concept. However, this ineluctable expansion also brings lots of security breaches especially for standard end users. Phishing is one of the mostly preferred attack type that hackers use by easily hindering theirselves. This type attack is initilally triggered with a simple e-mail or social media message which mainly forward the victims to a malicious webpage. For security admins, they are really hard attack types to detect. Therefore in this paper a content based phishing detection mechanism is proposed. In the proposal about six different machine learning models are implemented to select the best training models. Experimental results show that the proposed approach are very robust and give acceptable accuracies for security admins.

Keywords—machine learning, security breaches, attacks, phishing.

1. GİRİŞ

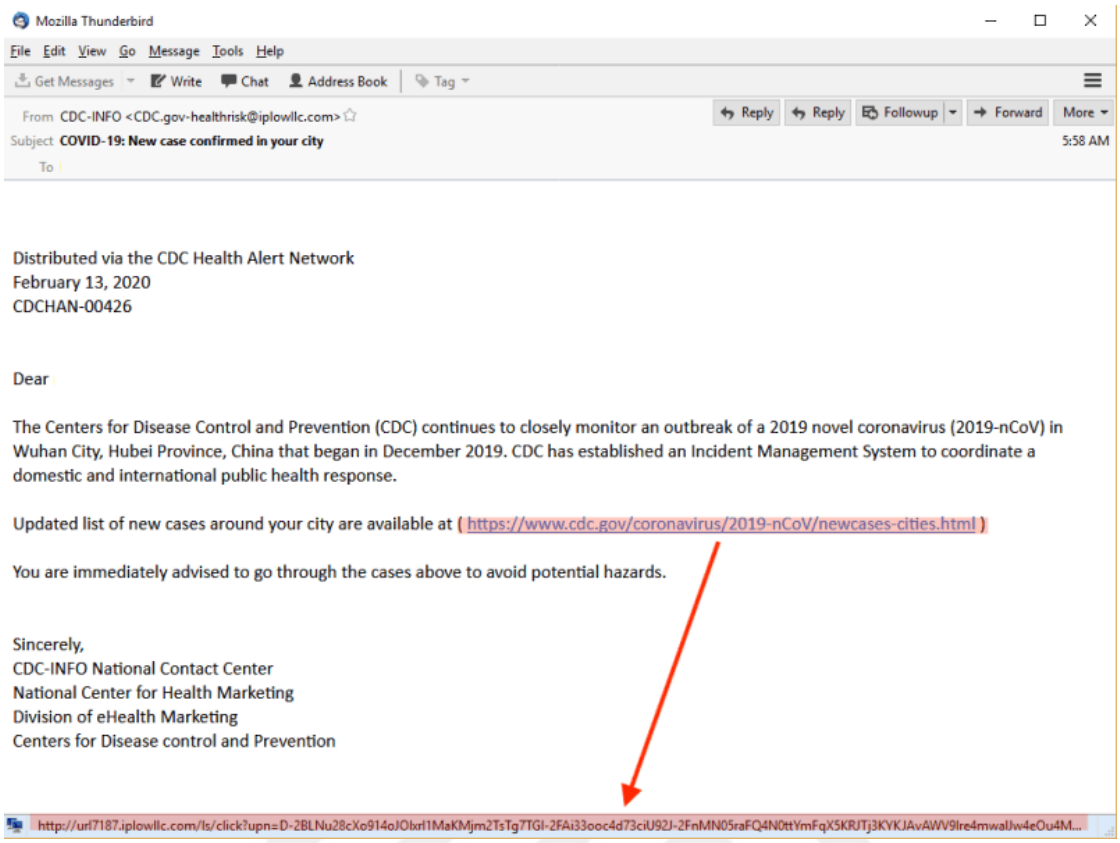
Gelişen teknoloji ile birlikte internet kullanımı her geçen gün artmaya devam ediyor. Son 20 yılda dünyada internet kullanan kişi sayısı % 1167 arttı. 31 Aralık 2019 itibarıyla dünya çapında 4,5 milyardan fazla internet kullanıcısı var [1]. Önümüzdeki yıllarda bu sayının artması bekleniyor. İnsanların internet üzerinden bankacılık, yemek, sağlık, sigorta, eğlence, eğitim başta olmak üzere ihtiyaçlarını karşılaması nedeniyle internet kullanımının hızlı ve yüksek oranda artmasının nedenlerini sıralayabiliriz. 2020 yılı itibarıyla küresel perakende satış hacmi 26 trilyon dolar civarında. Bu rakamın 2023 yılına kadar 29,7 trilyon dolara çıkması bekleniyor. Ancak e-ticaret üzerinden yapılan işlem hacminin 4,2 trilyon dolardan yüzde 19'a, 6,5 trilyon dolara ve yüzde 22'ye çıkması bekleniyor [2]. E-ticaret satış istatistikleriyle ilgili daha detaylı rakamları Şekil 1.1'de görüntüleyebilirsiniz.

İnternet kullanımının artmasıyla birlikte güvenlik sorunları da hızla artacaktır. Özellikle finansal işlemlerin farklı yöntemler kullanılarak yapıldığı web sitelerine her gün birçok saldırı yapılmaktadır. Bu saldırılar siber suçlar, hacker grupları, devlet adına çalışan saldırganlar ve içeriden gelen tehditler tarafından gerçekleştirilmektedir. İçeriden gelen tehditler kötü niyetli, ihmal kaynaklı veya tesadüfi olmak üzere üç alt gruba ayrılır [3]. İlk siber saldırıdan bu yana her alanda Milyonlarca saldırı ve birçok farklı tür günden güne yapılmaktadır. Bu noktada siber saldırılara karşı savunma birimleri ve ilgili strateji açıklamaları oluşturan ülkelere ulaştık. Bu açıklamalara göre enerji, ulaşım ve kritik altyapı hizmetleri kesintiye uğramaz, kişisel bilgiler çalınmaz, ifşa edilmez, elde edilen bilgiler savunulur ve kullanılmaz, bunun sonucunda kurumların ticari sırları ve teknik bilgileri zarar görmez. Elde edilen bilgilerden maddi zararın önlenmesi, kurumlar nezdinde itibar kaybının önlenmesi ve faaliyetlerde kesintinin önlenmesi stratejik planlar kapsamında amaçlanmaktadır [4].



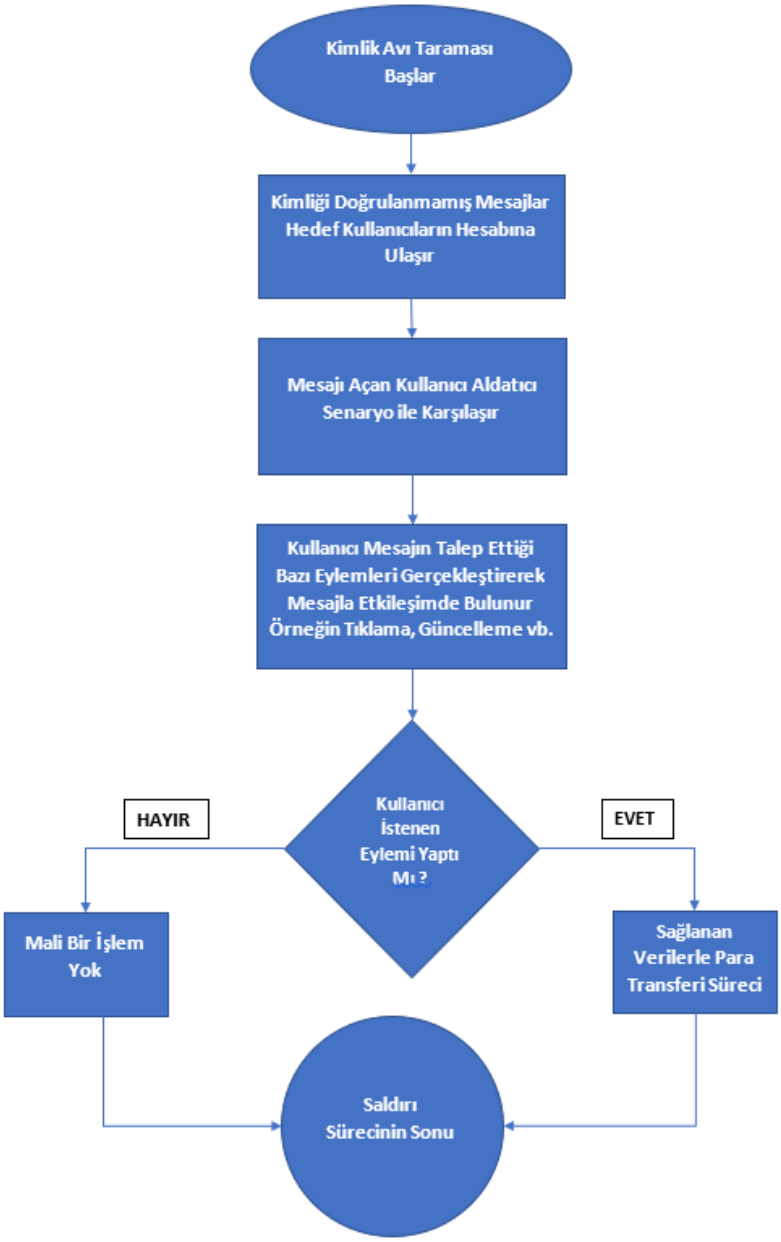
Şekil 1.1- Dünya Çapında Perakende E-Ticaret Satışları 2017-2023

Siber saldırıların çok uzun bir geçmişi vardır. Teknolojinin gelişmesiyle birlikte yıllar önce ilk siber saldırıdan bu yana çok çeşitli saldırı yöntemleri kullanıldı. Günümüzde en çok kullanılan saldırı türleri; DOS ve DDOS saldırıları, MitM saldırıları, Kimlik avı saldırıları, Parola saldırıları, SQL Enjeksiyon saldırıları, XSS saldırıları, Dinleme saldırıları, Doğum günü saldırıları ve Kötü Amaçlı Yazılım saldırıları. Bu saldırıların her biri farklı teknik altyapı ve yöntemleri takip etmektedir. Saldırılara karşı kullanılan birçok güvenlik yazılımı türü vardır. Bu yazılımlar ile birçok saldırı türüne karşı yüksek başarı oranları elde edilmektedir. Ancak phishing saldırıları için takip edilen birçok farklı yönteme rağmen tam bir savunma mekanizması kurulamamıştır. Bunun temel nedenlerinden biri, kullanıcılara e-posta ile gelen bu saldırıların doğrudan kullanıcı tarafından paylaşılarak karşı tarafa iletilmesidir. Kimlik avı saldırıları temelde kurumsal veya güvenilir web sitelerinin e-posta içeriğini taklit etmeye ve kullanıcıları yakalamaya dayanır. Burada hazırlanan taslak sitelerde mahsur kalmak isteyenlerin doldurması için çeşitli form ve görseller oluşturularak orijinal sitelerle birebir uyumlu tasarımlar hazırlanır. Bu çok düşük maliyetli ve hızlı hazırlanmış bir içeriktir. Phishing saldırıları için hazırlanan taslakta yer alan formlar aracılığıyla kişilerin kişisel verileri, finansal hesap bilgileri ve şifrelerinin ele geçirilmesi hedefleniyor. Saldırının failleri, elde ettikleri bilgileri, kullanıcılar mahsur kaldıklarında gerçek sitelerdeki paralarına ve değerli verilerine el koymak ve formları doldurup farkında olmadan iletmek için kullanırlar. Güvenlik ve finans sektöründe hizmet veren şirketlerin yaklaşık% 32'si, çalışanlarının bir saldırı durumunda davranışlarını ve hazır olma durumlarını değerlendirmek için kimlik avı deneyleri ve eğitimler düzenlemektedir. Bu işletmeler, ortalama olarak her 14 saniyede bir farklı içerikli kimlik avı saldırılarına maruz kalıyor. Çalışanları hedef alan toplam kimlik avı saldırısı sayısı 2019'da% 55 artarak, işletmelere yönelik tüm hedeflenen saldırıların% 71'ini oluşturuyor. Kimlik avı saldırılarındaki bu artış göz önüne alındığında, 2019'daki saldırıların ve ihlallerin% 90'ının kimlik avı öğeleri içerdiği tahmin edilmektedir. Şekil 1.2'de, yakın zamanda yapılmış olan covid-19 başlıklı örnek bir oltalama saldırısı gönderisini görebilirsiniz [5].



Şekil 1.2- Kimlik Avı Saldırı Örneği

Bir oltalama saldırısının temel yaşam döngüsünden bahsetmişken; Öncelikle yukarıda anlattığımız ve içeriği detaylandırılan hazır saldırı postası, internet ağı üzerinden hedef kitleye gönderilir. Daha sonra kimliği doğrulanmayan bu posta, savunmasız kullanıcıların sistemine girer. Mesaj, açık bir şekilde iyi düzenlenmiş aldatıcı bir görünüm içeriyor. Kullanıcı, mesajı gönderen saldırganın istediği şekilde içerikle etkileşime girer ve tıklama, güncelleme, kullanıcı girişi veya ödeme gibi benzer senaryolarla etkileşime girerek bilgilerini paylaşır. Paylaşılan bilgiler doğrudan saldırganın sistemine düşer. Para ve bilgi, elde edilen bilgileri kullanmak için gerçek sistemlerden aktarılır. Bu şekilde, süreç sona erer ve etkileşimde bulunan tüm kullanıcılar Şekil 1.3'deki akıştan geçerek saniyeler içinde basit bir kimlik avı saldırısının kurbanı olur [6].



Şekil 1.3 Oltalama Saldırısı Yaşam Döngüsü

Bu makalenin geri kalanı şu şekilde düzenlenmiştir; Bir sonraki bölüm, kimlik avı sorunlarını çözmek için çeşitli makine öğrenimi ve yapay zeka algoritmalarının kullanımına ilişkin çalışmaları ve açıklamaları içerir. Önerilen makine öğrenimi metodolojisi, algoritmalarla üçüncü bölümde detaylandırılmıştır. Dördüncü bölümde deneysel çalışma ve sonuçlar anlatılmış ve son olarak sonuç ve gelecekteki çalışmalar belirtilmiştir.

1.1. Problem Tanımı

Bir problemin çözümünü üretebilmek için ilk olarak problemin ne olduğunun iyi bir şekilde anlaşılması gerekmektedir. Tarafımıza gelen tüm postalar şüpheli ve tehlike içerebilir. Bu tarz

saldırılar maddi manevi kötü sonuçlar doğurabilir

Oltalama saldırıları tespiti, en kullanışlı ve en düşük maliyetli işlemlerden biri olarak görülmektedir. E-Postaların alıcı ile buluştuktan sonra alıcıların gönderilmiş olan içerik ile etkileşime girmesi kullanıcıların kişisel verileri, banka hesapları ile ilgili detaylar, ve diğer tüm değerli içerikler tehlike altına girebilmektedir. Oltalama saldırıları tespiti, mağdur duruma düşebilecek her türlü vatandaşın korunması açısından hayati ve kritik bir öneme sahiptir. Bunun yanı sıra bu saldırıların tespiti internet ortamının güven içerisinde kullanılabilmesi, iş süreçlerinin yürütülmesi, maddi kayıpların en aza indirilmesi hususunda tatmin edici olup gururlandırıcı bir hal alacaktır. Oltalama saldırı tespiti, e-posta süreçlerinin sıkça kullanıldığı bugünlerde yapılan en basit fakat tespit etmesi en güç saldırı yöntemlerinden birisi olduğu için her gün artarak devam etmektedir. Saldırı içerikleri çeşitli hesap numarası, şifreler, banka bilgileri gibi alanların girilebileceği formları içerir ve bunların doldurulması ile süreç tamamlanır. Bu şekilde veriler çalınmaktadır ve sonrasında kısa süre içerisinde büyük zararlar doğurmaktadır. [7]

1.2. Literatüre Katkıları

Geliştirilen model, günümüzde artık her alanda kullanımı yaygınlaşan oltalama saldırıları ile bu saldırılardan korunmak isteyen vatandaşlar tarafından kullanılabilecektir. Kullanıcı makine öğrenme algoritmalarının sağladığı kolaylık sayesinde hem zamandan kazanacaktır hem de daha yüksek performanslı ve daha güvenilir bir tespit oranı ile saldırılardan korunmayı sağlayacaktır.

Katkı 1. Günümüzde hemen hemen her alanda kullanılan yazılımların hataya açık olma ihtimali vardır. Yazılımcının daha yapılan işi teslim etmeden belirli metrikler yardımıyla yazılımında hata olma ihtimalini test edebilmesi amaçlanmıştır.

Katkı 2. Makine öğrenmesi teknikleri, öngörülü yazılım modelleri oluşturmaya nasıl katkıda bulunur sorusunun cevabı amaçlanmıştır.

Katkı 3. Geliştirilen çalışma sayesinde, önemli bir konu olan saldırılar ile maddi ve manevi kayıplarının en aza indirilmesi amaçlanmaktadır. En az hatalı tespit ile en fazla başarı oranı sağlamak bizim için temel amaçlardan biridir. Her yöntem her veride aynı efektif sonucu vermeyebilir. Bunun için hangi veri kümelerinde hangi metodolojinin yararlı olduğunun karşılaştırmalı analizinin çıkarılması amaçlanmıştır.

Katkı 4. Geliştirilen çalışmada amaçlardan biri de çok fazla miktarda veri işlenirken seçilen algoritmaların nasıl sonuç verdiğidir.

Katkı 5. Yapılan çalışmada diğer önemli bir katkı zaman kavramıdır. Hatayı önceden tespit

etmek hatanın ilerde oluşturacağı sorunları da önceden tespit etmek olacağı için her anlamda maliyeti azaltacağı amaçlanmıştır.

1.3. Tezin Organizasyonu

Bu tez çalışması 5 bölümden oluşmaktadır.

- Birinci bölümde, problem tanımı yapılmıştır. Yapılan çalışma tanıtılmıştır, amacı ve önemi anlatılmıştır ve literatüre katkısından söz edilmiştir.
- İkinci bölümde, tez çalışmasının ana kaynağı olan makine öğrenmesi algoritmalarından ve kullanılan veri setlerinden bahsedilmiştir.
- Üçüncü bölümde, çalışmada kullanılan yöntemler, çalışmada kullanılan teoriler, yaklaşımlardan ve bunların nasıl uygulandığından, amaçlarından bahsedilmiştir.
- Dördüncü bölümde, yapılan çalışma önerilen yöntemle ilgili detaylardan, bileşenlerden, yöntemin akış diyagramından detaylı bir şekilde bahsedilmiştir.
- Beşinci yani son bölümde ise yapılan testler ve sonuçları detaylı bir şekilde belirtilmiştir.

2. ÖN BİLGİLER ve LİTERATÜR ARAŞTIRMASI

Bu bölümde araştırması yapılan çalışmanın temel tanımları ve çalışma için yapılan literatür taraması sunulmuştur. İlk olarak yazılım hatası tanımının ne olduğu ve hata türlerinin neler olduğu anlatılmıştır. Daha sonra makine öğrenmesi tanımı yapılmıştır ve beraberinde tezde kullanılan makine öğrenmesi algoritmaları özetlenmiştir. Bu bölümün sonunda ise konuya ilişkin yapılmış olan akademik çalışmalardan kullanılanlar özetlenmiştir.

2.1. Oltalama Saldırı Tanımı ve Tespit Türleri

2.1.1. Oltalama Saldırısı Tanımı

Oltalama saldırısı genel olarak e-posta içeriğinin kullanıcıyı aldatabilecek bir senaryo ile içeriği doldurarak, ilgili kullanıcılara bunun gönderilmesi sonrasında kullanıcıların gerçeğinden ayırt edemeyeceği kadar inandırıcı olan bu sahtekarlık postalarıyla etkileşime girerek farkında olmadan kişisel tüm bilgi ve içeriklerini mail yoluyla geri iletmesi ile sonuçlanan bir türdür. İçeriğin çok ucuz e basit yöntemlerle hazırlanarak geniş bir kesim ile hızlıca paylaşılabilmesi açısından kullanılan en yaygın yöntemdir ve her gün yeni senaryo ile teknikler kullanıldığı ve etkileşime giren kullanıcıların kendi elleri ile bilgilerini paylaşması sebebiyle tespit edilmeleri gerçekten çok zor ve yönetilmesi güç bir süreçtir. [8] Şuana kadar bu başlık altında korunmak için çeşitli yöntemler kullanılmasına rağmen maalesef yeterli ve güvenilir seviyede bir korunma yöntemi henüz mevcut değildir.

2.1.2. Saldırı Tespit Türleri

2.1.2.1. Liste Tabanlı Saldırı Tespit Sistemleri

Liste tabanlı sistemlerde, URL üzerinden erişilecek adres veya posta içeriği bir kara liste veya beyaz liste aracılığıyla kontrol edilir. Kara liste uygulamasının amacı, önceden kimlik avı saldırıları olarak algılanan kaynaklara erişimi korumak ve engellemektir. Buradaki genel sorun, sistemin URL tabanlı çalışmasından kaynaklanmaktadır. URL adresindeki küçük bir değişiklik bile kontrol mekanizmasını aldatabilir ve bir güvenlik açığı oluşturabilir. Ayrıca bu sistemlerde yeni oluşturulan saldırılara karşı tecrübe edilmediği için koruma sağlanamamaktadır. Sistemin çalışması devam ederken kara listenin gün geçtikçe genişlemesi ile işin performansında ve hızında önemli bir düşüş gözlemlenmektedir.

Beyaz liste bazlı uygulama sistemlerinde ise tam tersi uygulama yöntemi izlenir. Burada, genel amaçlı, korumalı ve intranet tabanlı sistemler için erişilebilen sınırlı sayıda URL adresi önceden tanımlanmıştır ve yalnızca bu adreslere erişime izin verilir. Buradaki en büyük sorun, sistemin getirdiği sıkı kontrol mekanizması nedeniyle sürekli olarak erişim sorunlarıyla karşılaşılmasıdır. Bu, erişim talepleri ve işin engellenmesi gibi sorunlar yaratır.

Kara liste ve beyaz liste ile ilgili önceden yapılmış ve kullanımda olan uygulamalara bakmanız gerekirse Spoofguard. [9] belirli bir sayfanın bir adres sahteciliği saldırısının parçası olma olasılığını değerlendirmek için alan adı, url, bağlantı ve görüntü kontrollerini kullanan sezgisel bir uygulamadır. Netcraft [10], phishing sayfalarını tespit etmek için url sezgisel analiz kullanan başka bir uygulamadır. Netcraft, ortak amacı olmayan karakterler içeren şüpheli url'leri yakalar. Earthlink [11] ve McAfee SiteAdvisor [12], sayfaların kimlik avı olasılığını tahmin etmek için alan adı kaydının sahibi, yaşı ve ülkesi hakkındaki bilgileri kullanır. McAfee SiteAdvisor, kimlik avı sitelerini tespit etmek için meşru sitelere giden bir dizi bağlantıyı sezgisel olarak araştırır.

2.1.2.2. Kural Tabanlı Saldırı Tespit Sistemleri

Kural tabanlı sistemlerde, erişilecek url veya web adresinin basit kurallarla bir phishing saldırısı olup olmadığının anlaşılması amaçlanır. Burada kullanılan yaklaşım genellikle AND, OR veya IF THEN benzeri yapılara sahip Boolean türü dönüş kurallarının çıktılarına göre işlem yapılarak gerçekleştirilir. Kural tabanlı sistemler alt gruplara ayrılır. İlk yaklaşım, arama motoru tabanlı kural mekanizmalarından oluşur. Bu yaklaşıma göre url veya domain adresinin global arama motorlarının indekslerinde yer alıp almadığı if koşulu konularak kontrol edilir. Arama motorlarının indekslerinde herhangi bir domain ve url bilgisine ulaşılamaması durumunda ilgili adres saldırı olarak tespit edilir. İkinci yaklaşım, anahtar kelime tabanlı bir kural mekanizması olarak belirlenir. Bu yaklaşıma göre, araştırmada 62 farklı tür özellik ve kelime grubu işaretlenerek rastgele seçilen internet sitelerinin% 80'i belirlenmiştir. Sistem, işaretli sözcükleri içeren adreslerin oltalama saldırıları olduğu kuralına göre çalışır. [13]

2.1.2.3. Görsel Benzerlik Tabanlı Saldırı Tespit Sistemleri

Görsel benzerlik tabanlı saldırı tespit sistemleri, web sitelerinin sayfalarının görsel benzerliğini karşılaştırarak çalışır. Phishing, bunların dışındaki diğer siteler, sitelerin sunucu tarafı görünümü alınarak saldırı veya saldırı olarak iki ayrı sınıfa ayrılır ve tahmin yapılır. Bu iki veri, görüntü işleme teknikleriyle karşılaştırılır. Saldırı sistemleri genellikle gerçek sitelere

çok yakın tasarlanır. Ancak görsel olarak aralarında ufak farklar var. Görüntü işleme teknikleriyle, kullanıcıların kolayca fark edemediği bu farklılıkları fark etmek daha kolaydır. Elde edilen benzerliğe göre web sitesinde oltalama saldırısı olup olmadığına karar verilebilir.

2.1.2.4. Makine Öğrenmesi Tabanlı Saldırı Tespit Sistemleri

Makine öğrenimi tabanlı saldırı tespit sistemleri, en yeni ve en güncel yaklaşımdır. Genelde sistem, çok sayıda saldırı ve gerçek web sitesi içeriği elde edildikten sonra bir model oluşturarak ve özelliklerini belirleyerek ve sistem yeni bir web sitesi ile karşılaştığında bu modelleri sorgulayarak olasılık tahmin sonuçlarına göre çalışır. Bu anlayışın ön plana çıkmasının sebebi birçok web sitesinden elde edilen özellik ve veriler ile diğer sistemlere göre çok yüksek performans oranıyla sonuç vermesidir. Ancak zamanla performans sorunlarına neden olmama, yeni üretilen bir oltalama saldırı taslağı olsa bile tespit edebilme, onu kullandıkça öğrenebilme, kendini geliştirme ve daha fazlasını verme tercihinde sistemin büyük payı olmuştur doğru sonuçların. Bu çalışmada, literatürde kabul gören denetimli öğrenme algoritmaları kullanılarak öznitelikler çıkarılarak modeller oluşturulmuş ve böyle bir sistem için hangi modelin daha verimli olduğu incelenmiştir. Yapay zeka temelli bu modelin diğer tüm sistemlerden daha başarılı olacağı ve yakın gelecekte tüm sistemlerde mevcut yaklaşımların yerini alacağı tahmin ediliyor. [14]

2.2. Makine Öğrenmesi

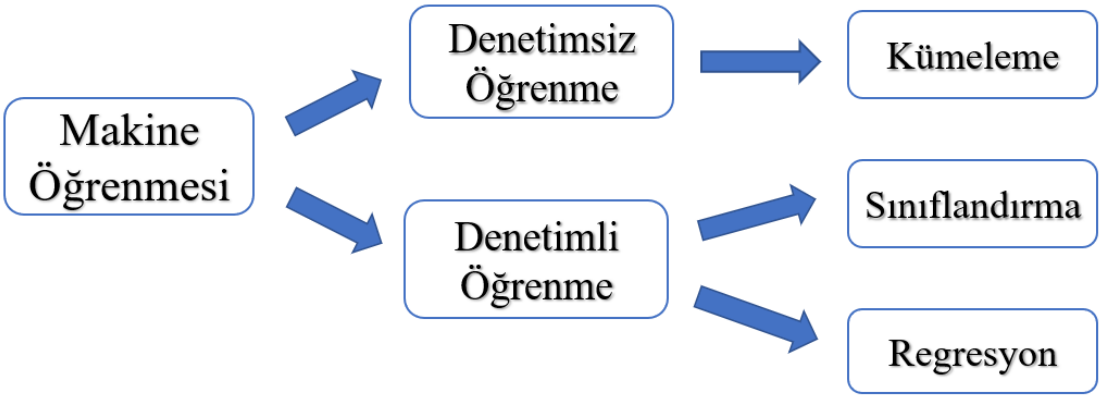
2.2.1. Makine Öğrenmesi Tanımı ve Amacı

Makine öğrenimi, sistemlere açıkça programlanmadan deneyimden otomatik olarak öğrenme ve geliştirme yeteneği sağlayan yapay zekanın bir uygulamasıdır. Yazılım mühendisleri, sistem geliştirme aşamalarını zaman ve maliyet tüketimlerini en aza indirgemek için makineleri kullanmaktadırlar [15]. Makine öğrenimi, verilere erişebilen ve bunları kendileri için kullanabilen bilgisayar programı geliştirilmesine odaklanır.

Öğrenme süresi, verilen örneklere dayanarak verilerdeki kalıpları aramak ve gelecekte daha iyi kararlar vermek için, doğrudan deneyim veya komutlar ile başlar. Makine öğrenmesinin birincil amacı, bilgisayarların insan yardımı olmadan otomatik olarak öğrenme sağlaması ve öğrendiklerini kullanması gereken an gelince eyleme dönüştürebilmesidir.

2.2.2. Makine Öğrenmesi Yöntemleri

Bu bölümde makine öğrenmesi algoritmalarındaki üç farklı öğrenme yöntemi anlatılmıştır. Şekil 2.1'de makine öğrenmeleri kategorize edilip gösterilmiştir.



Şekil 2.1- Makine Öğrenmesi Yöntemleri

Denetimli Öğrenme: Denetimli öğrenmenin ilk adımı iyi tanımlanmış olan geniş bir eğitim verisine sahip olmaktır [16]. Giriş ve çıkış verileri, gelecekteki veri işleme için bir öğrenme temeli sağlamak üzere etiketlenmiştir. Denetimli öğrenme terimi, bu algoritmanın öğretmen olarak düşünülebilecek bir eğitim veri kümesinden öğrenildiği fikrinden gelmektedir. Denetimli öğrenme problemleri; sınıflandırma ve regresyon olarak gruplandırılırlar.

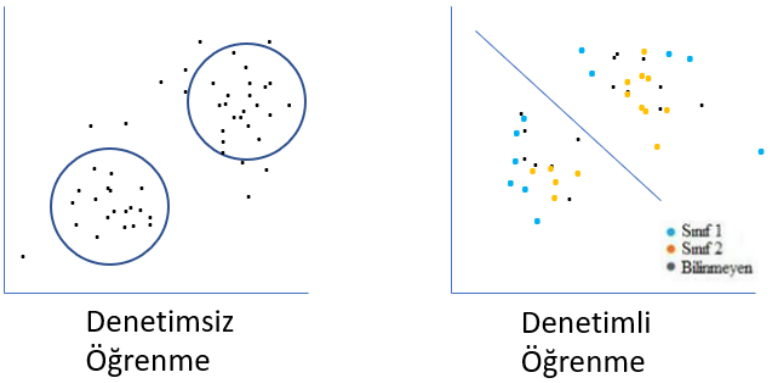
Sınıflandırma problemi, çıktı değişkeninin "kırmızı" veya "mavi", "hastalık var" veya "hastalık yok" gibi kategori olması durumudur.

Regresyon problemi, çıktı değişkeninin "dolar" veya "ağırlık" gibi gerçek bir değer olmasıdır.

En yaygın kullanılan denetimli öğrenme algoritmaları; Destek Vektör Makineleri (SVM), Karar ağaçları (DT), K-En Yakın Komşu Algoritması (KNN), Naif Bayes (NB) ve Regülasyon olarak sıralanabilmektedir.

Denetimsiz Öğrenme: Yalnızca giriş verilerinin olduğu ve buna karşılık gelen çıkış verilerinin olmadığı öğrenmedir. Denetimsiz öğrenmenin amacı, veriler hakkında daha fazla bilgi edinmek için verilerin temelini oluşturan yapıyı veya dağılımı modellemektir. Denetimli öğrenmenin aksine doğru cevapları yoktur ve tabiri caizse öğretmenleri yoktur. Denetimsiz öğrenme teknikleri rekabetçi öğrenme teknikleridir. [17]

Kümeleme problemi, satın alma davranışı yolu ile müşterileri gruplama gibi verilerden doğal gruplamaların keşfedilmek istenildiği problemlerdir. Şekil 2.2'de denetimli ve denetimsiz öğrenme farklı gösterilmiştir.



Şekil 2.2- Denetimli ve Denetimsiz Öğrenme

Yarı Denetimli Öğrenme: Büyük miktarda giriş verisine ve sadece bazı çıkış verilerinin etiketlendiği yöntemlere yarı denetimli öğrenme denir. Bu problemler denetimli ve denetimsiz öğrenme arasındadır. Örnek olarak sınıflandırma ve regresyon verilebilmektedir. Etiketlenmemiş verilerin nasıl modelleneceği hakkında varsayımlar yapan diğer yöntemlerin uzantılarıdırlar.

2.2.3. Yararlanılan Makine Öğrenmesi Algoritmaları

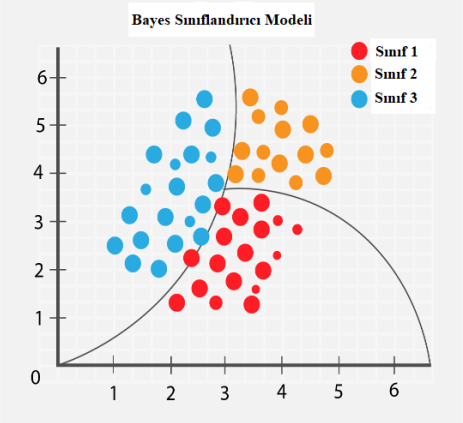
2.2.3.1. Saf Bayes

İlk algoritma Naive Bayes'tir. Naive Bayesian sınıflandırıcı, yordayıcılar arasındaki bağımsızlık varsayımları ile Bayes'in teoremine dayanmaktadır. Naive Bayes modelinin, karmaşık yinelemeli parametre tahmini olmaksızın oluşturulması kolaydır, bu da onu çok büyük veri kümeleri için özellikle yararlı kılar. Basitliğine rağmen, Naive Bayesian sınıflandırıcı genellikle şaşırtıcı derecede iyi performans gösterir ve genellikle daha karmaşık sınıflandırma yöntemlerinden daha iyi performans gösterdiği için yaygın olarak kullanılır. Bayes teoremi Şekil 2.3 üzerinde ifade edildiği üzere temel olarak, P (c), P (x) ve P (x | c) 'den posterior olasılığı, P (c | x) hesaplamanın bir yolunu sağlar. Naive Bayes sınıflandırıcısı, bir yordayıcı (x) değerinin belirli bir sınıf (c) üzerindeki etkisinin diğer yordayıcıların değerlerinden bağımsız olduğunu varsayar. Bu varsayıma sınıf koşullu bağımsızlığı denir. [18]

$$P(A|B) = \frac{P(B|A)\,P(A)}{P(B)}$$

Bayes Olasılık Terminolojisi kullanılarak yukarıdaki denklem aşağıdaki gibi yazılabilir.

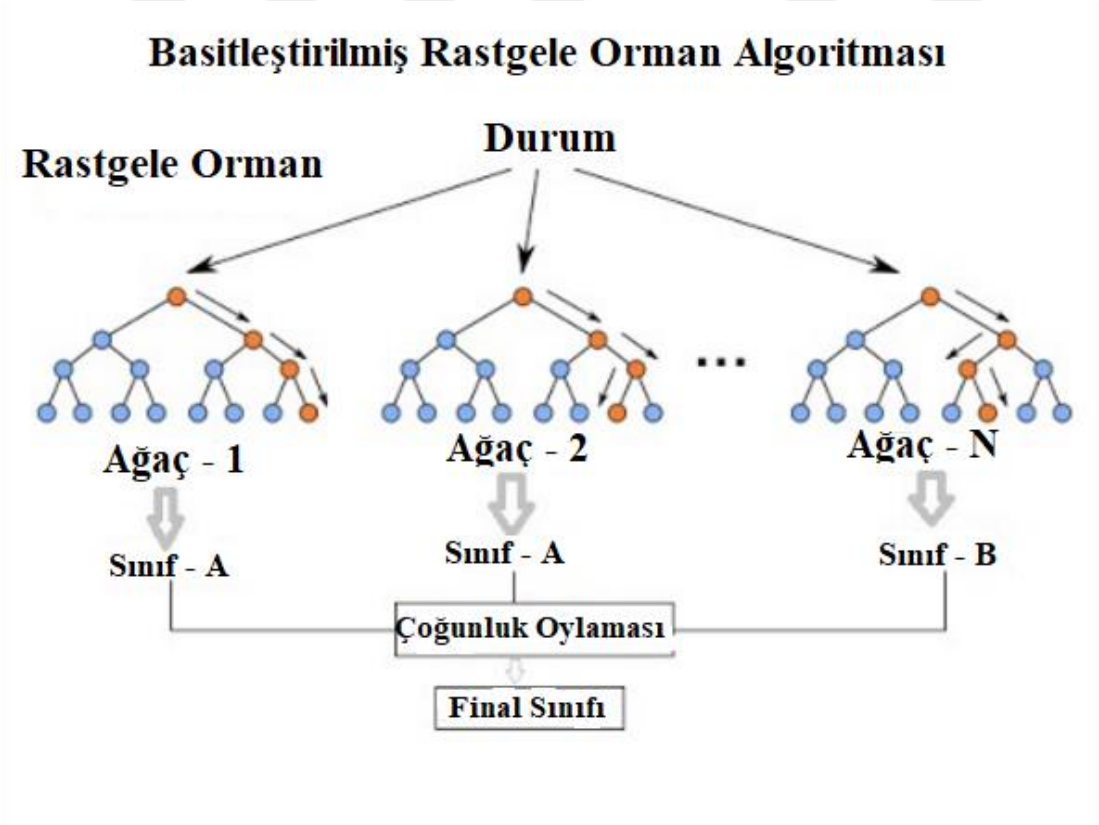
$$\text{SONRA GELEN} = \frac{\text{ÖNCEKİ} \times \text{OLASILIK}}{\text{DEĞER}}$$



Şekil 2.3- Saf Bayes Algoritması

2.2.3.2. Rastgele Orman Yöntemi

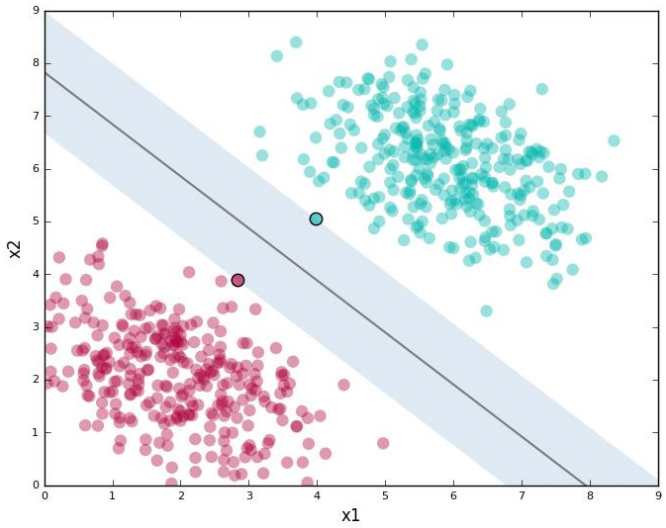
Rastgele Orman, denetimli bir öğrenme algoritmasıdır. Adından da anlaşılacağı gibi, bir orman yaratır ve bir şekilde rastgele yapar. Oluşturduğu "orman", genellikle "torbalama" yöntemiyle eğitilen bir karar ağaçları koleksiyonudur. Torbalama yönteminin genel fikri, öğrenme modellerinin bir kombinasyonunun genel sonucu artırmasıdır. Basit bir deyişle, rastgele orman algoritması Şekil 2.4 örneğinde olduğu gibi birden fazla karar ağacı oluşturur ve daha doğru ve istikrarlı bir tahmin elde etmek için bunları bir araya getirir. [19]



Şekil 2.4- Rastgele Orman Algoritması

2.2.3.3. Destek Vektör Makinesi

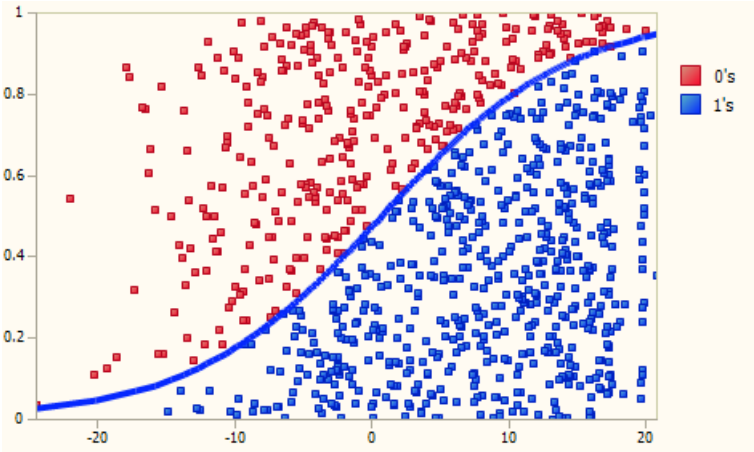
Destek Vektör Makinesi, sınıflandırma veya regresyon problemleri için kullanılabilen kontrollü bir makine öğrenme algoritmasıdır. Ancak çoğunlukla sınıflandırma problemleri için kullanılmaktadır. Bu algoritmada, her veri öğesi, belirli bir koordinatın değeri olan her özelliğin değeriyle birlikte n boyutlu uzayda (burada n sahip olduğunuz özelliklerin sayısıdır) bir nokta olarak çizilir. Daha sonra sınıflandırma, Şekil 2.5 üzerinde X,Y koordinat düzleminde yer aldığı şekilde genel olarak iki sınıftan oldukça iyi ayıran hiper düzlem bulunarak gerçekleştirilir. Destek Vektörleri sadece gözlemin koordinatlarıdır. Destek Vektör Makinesi, iki sınıfı en iyi şekilde ayıran bir sınırdır. [20]



Şekil 2.5- Destek Vektör Makinesi Algoritması

2.2.3.4. Lojistik Regresyon (LR)

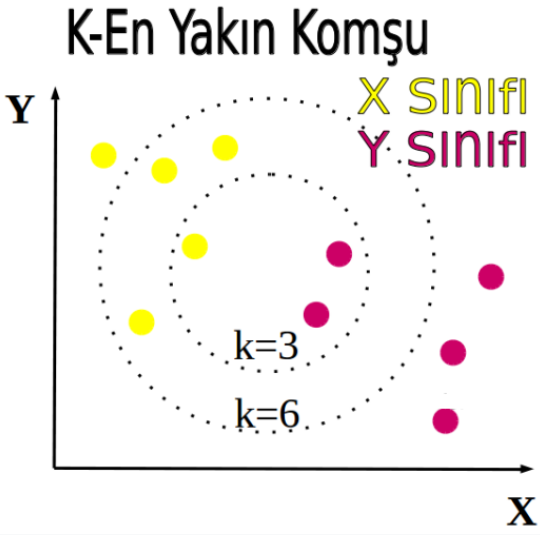
Lojistik Regresyon, sınıflandırma için bir regresyon yöntemidir. Kategorik veya sayısal verileri sınıflandırmak için kullanılır. Yalnızca bağımlı değişken olan sonuç 2 farklı değer alabiliyorsa çalışır. (Evet / Hayır, Erkek / Kadın, Şişman / Zayıf vb.) Doğrusal sınıflandırma problemlerinde yaygın olarak kullanılmaktadır. Bu nedenle Doğrusal Regresyon'a çok benzer. Sınıflandırma metodolojisini Şekil 2.6 üzerinde inceleyebilirsiniz.



Şekil 2.6- Lojistik Regresyon Algoritması

2.2.3.5. K-En Yakın Komşu (KNN)

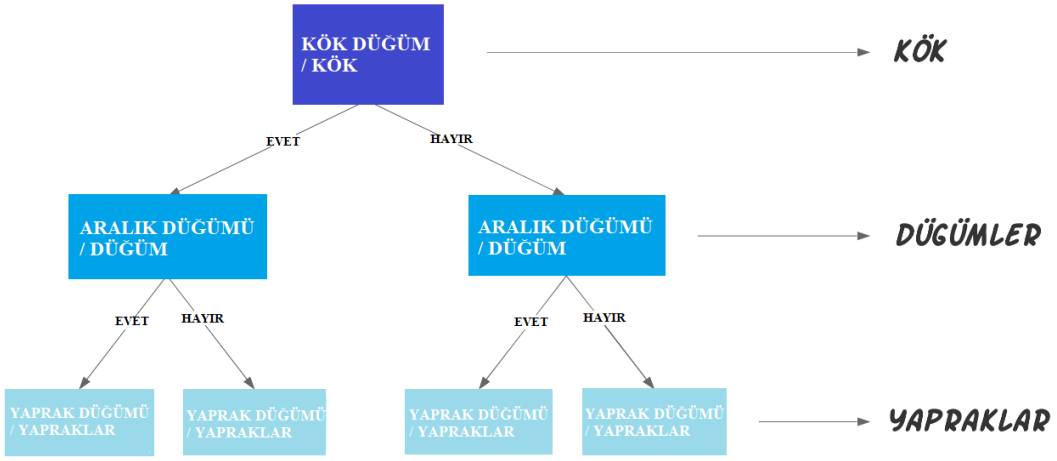
K-NN (K-En Yakın Komşu) algoritması, en basit ve en çok kullanılan sınıflandırma algoritmalarından biridir. K-NN parametrik olmayan (parametrik olmayan), tembel (tembel) bir öğrenme algoritmasıdır. Tembel kavramını anlamaya çalışırsak, istekli öğrenmenin aksine, tembel öğrenmenin bir eğitim aşaması yoktur. Eğitim verilerini öğrenmez, bunun yerine eğitim veri setini "ezberler". Bir tahminde bulunmak istediğimizde, tüm veri setinde en yakın komşuları arar. Algoritmanın işleyişinde bir K değeri belirlenir. Bu K değerinin anlamı, bakılacak eleman sayısıdır. Bir değer geldiğinde, değer arasındaki mesafe, en yakın K sayıda eleman alınarak hesaplanır. Öklid işlevi genellikle mesafe hesaplamasında kullanılır. Manhattan, Minkowski ve Hamming işlevleri de Öklid işlevine alternatif olarak kullanılabilir. Mesafe hesaplandıktan sonra sıralanır ve Şekil 2.7 örneğindeki gibi karşılık gelen değer uygun sınıfa atanır. [21]



Şekil 2.7- K-En Yakın Komşu Algoritması

2.2.3.6. *Karar Ağacı Yöntemi (DT)*

Son algoritma Karar Ağacı yöntemi, hem sınıflandırma hem de regresyon problemlerinde kullanılan en popüler makine öğrenme algoritmalarından biridir. Veri madenciliği alanında da sıklıkla kullanılmaktadır. Karar ağaçları genellikle insan seviyesinde düşünülebilir, bu nedenle verileri anlamak ve biraz iyi yorumlama ve görselleştirme yapmak çok basittir. Karar ağacı, adından da anlaşılacağı gibi, bir ağaç yapısı kullanılan özyinelemeli bir süreçtir. Tek bir düğümle başlar ve yeni sonuçlara dalarak bir ağaç yapısı oluşturur. Algoritma çalıştığında girilen değer düğümlere bakarak belli bir yolda hareket eder ve sonuç verir. Karar ağacı yönteminin yapısını Şekil 2.8 üzerinde daha detaylı olarak görebilirsiniz.



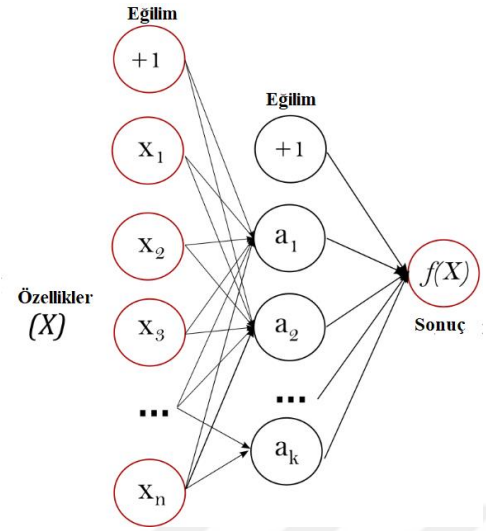
Şekil 2.8- Karar Ağacı Algoritması

2.2.3.7. *Çok Katmanlı Algılayıcı Algoritması (MLP)*

Çok katmanlı algılayıcı algoritması, bir işlevi öğrenme üzerine kurulu denetimli öğrenme algoritmalarından biridir, herhangi bir veri seti üzerinde eğitim alarak, burada girdi boyutlarının sayısı ve çıktı için boyutların sayısına göre model eğitim süreçlerini sınıflandırma yaparak yönetmektedir. Bir dizi özellik ve buna bağlı veri sağlandığında ve bir öğrenme süreci başlatıldığında, sınıflandırma veya regresyon için doğrusal olmayan bir fonksiyon tahmin edicisini öğrenebilir. Lojistik regresyon ile benzer fakat farklıdır, çünkü lojistik regresyondan farklı olarak girdi ve çıktı katmanı arasında gizli katmanlar adı verilen bir veya daha fazla doğrusal olmayan katmanda sürece dahil olabilmektedir. Şekil 2.9, skaler çıktılı tek bir gizli katman MLP'yi göstermektedir.

Çok Katmanlı algılayıcı algoritmasında birden fazla doğrusal katman (nöron kombinasyonları) olabilir. Basit şekilde tarif etmek gerekirse, üç katmanlı ağ, ilk katman giriş katmanı ve sonuncusu çıktı katmanı ve orta katman gizli katman olarak konumlanıyor. Girdi verilerimizi girdi katmanına besleniyor ve çıktıyı çıktı katmanından alabiliyoruz. Modeli

amacımıza göre daha karmaşık ve kompleks hale getirmek için gizli katman sayısını istediğimiz kadar artırıp parametreleri özelleştirebiliriz ve bu sayede ağ daha da fazla dizilim içerebilir.



Şekil 2.9- Çok Katmanlı Algılayıcı Algoritması

2.2.3.8. Aşırı Gradyan Arttırma Algoritması (XGBoost)

Aşırı Gradyan Arttırma Algoritmasının en belirgin ve önemli özellikleri yüksek model tahmini yapabilme gücü elde etmesi, gereğinden fazla öğrenme sürecini önleyebilmesi ve verileri daha iyi performans sağlayabilecek şekilde yönetebilmesidir. Aşırı Gradyan Arttırma Algoritması ilk tahmini yapmakla başlar. Yapılan her yeni tahmin ile modelin hata payı, doğruluk oranı ve performansı çeşitli parametreler göz önünde bulundurularak incelenir. Yapılan bir tahminin ne kadar iyi olduğu modelin hatalı tahminleri ile incelenir. Hatalar, doğru değerden tahmin edilen değerin çıkarılması ve burada elde edilen eşitsizlikler ile bulunabilmektedir. Bundan sonraki adımda Gradyan Arttırma algoritmasının basit versiyonunda olduğu gibi elde ettiğimiz hatalarla karar ağacı oluşturulur. Burada hedef hatalardan öğrenerek doğru tahmin yapabilme yeteneğini kazanabilmektir. Oluşturulan ağacın her bir kolu için benzerlik değeri elde edilir. Benzerlik değeri verilerin kollarda ne kadar iyi gruplandığını belirtir. Benzerlik değeri elde edildikten sonraki kilit nokta ise modelin performansının daha da arttırılıp artırılamayacağıdır. Bunun cevabını elde edebilmek için olabilecek bütün karar ağaçlar kurulur. Hepsi için benzerlik değerleri ve diğer isterler hesaplanır. Hangi karar ağacının en iyi olduğunu belirlemek için kazanç hesaplanır. Benzerlik değeri ile her bir kol değerlendirilirken, kazanç ile bütün ağaç değerlendirilir. Bu şekilde hesaplanan tüm ağaçlar arasından elde edilen en iyi kazanç değeri bize ideal modeli elde etmemizi sağlamaktadır.

Kazanç=Sol Benzerlik Skoru + Sağ Benzerlik Skoru – Önceki Ağacın Benzerlik Skorudur [22].

3. VERİ SETİNİN İŞLENMESİ VE ÖZELLİK ÇIKARIMI

Veri, en kısa tanımı ile işlenmiş bilgilerdir. Veri; ölçüm, deney, gözlem, sayım veya araştırma yolu gibi yöntemler ile elde edilebilir. Çalışmamızda edindiğimiz verilerin tamamı daha önce yapılmış saldırıların web sayfası ön yüz kodları (Javascript ve HTML) ile çeşitli özelliklerinin çıkarımları ile elde edilmiştir. Burada büyük veri literatürünün girdiği yerde ne kadar çok özellik ve veri örneği elde edersek kullandığımız algoritma ile elde edeceğimiz makine öğrenimi tekniği o kadar iyi bir sonuç sağlayacaktır.

3.1 Yararlanılan Veri Setlerindeki Metrikler

Makine öğrenmesi sürecinde kullanılan veri setleri Phistank.com [23] adresinde ki web sayfası önyüz kodlarından sağlanmıştır, oltalama saldırılarının çeşitli şekillerde tespit edildiği ve çok çeşitli veri kaynaklarını içeren bir sitedir. Birçok kurumsal şirket ve siber güvenlik şirketi burada yer alan verileri kullanarak savunma sistemleri tasarlamaktadır. Literatür taramasında, makine öğrenimi yönteminde kullanılan oltalama verilerinin genellikle Phistank.com'dan alındığı görülmüştür. Önceki web sayfası veya e-posta adresleri için gerekli sınıflandırmayı yaptı. Ayrıca, pozitif veya negatif sınıflandırma formları hakkında veri sağladı. Ayrıca var olmayan ilgili web sayfalarının içeriklerini de Python programlama dili ile HTML sağladık.

Tablo 3.1-Veri Seti Bilgisi

	Veri Seti
Oltalama Verisi	8,353
Temiz Veri	5,438
Toplam	13,791

3.2Veri Setinden Sağlanan Metrikler

Kullandığınız algoritma seçimleri ve sahip olduğunuz verileri işleme yöntemleri, makine öğreniminin başarı oranı için hayati önem taşır. Bu nedenle, kritik özellikleri tespit etmek için kapsamlı bir özellik tespit çalışması yürüttük. Web sitelerinin ve e-postaların kaynak kodlarını ve içeriklerini inceleyerek arka planda yapılan daha detaylı çalışmaları analiz etmeye

odaklandık. E-posta ve web sitelerinin özellikleri, javascript ve HTML sayfa kodlamasında ayrıntılı olarak incelenmiştir. Çalışmamızda bu içerikte 58 farklı özellik tespit ettik. Bu özellikler Python programlama dili kullanılarak yazılan betiklerle elde edildi. Böylelikle kaynak kodu ve sayfa şablonları analiz edilerek elde edilen verilerde çok daha yüksek bir başarı oranı hedeflenmiştir. Çalışmada kullanılan özellikler Tablo 3.2'de listelenmiştir.

Tablo 3.2-Veri Seti Metrik Bilgisi

#		İsim	#		İsim	#		İsim
1	MEVCUT	Form	21	MEVCUT	Downloadable Content	41	ADET	Article Element
2		POST Method	22		Cookie	42		Hidden Element
3		Input Element	23		Cache	43		P Element
4		Image	24		Favicon / icon	44		Content Spec Char
5		Button	25		Copyright	45		Content Word
6		Submit	26		Readable HTML	46		Black List Word
7		Non UTF-8 Char	27		Black Listed Word Usage	47		HTTP Link
8		Checkbox	28		Hidden Element	48		HTTPS Link
9		Password	29		Redirect	49		MetaTag
10		Link	30	ADET	Input Element	50		HTML Element
11		BlackListed Link	31		Option Element	51		Link

12		Title	32		Select Element	52		Checkbox
13		Title has spec char	33		TH Element	53		Button
14		E-Mail Input	34		TR Element	54		Image
15		Script Window	35		Table Element	55	UZUNLUK	Title
16		IFrane	36		LI Element	56		Longest Word
17		Date Time	37		UL Element	57		Shortest Word
18		Name or Surname	38		Href Element	58		Content
19		Phone Number	39		Div Element			
20		MetaTag	40		Span Element			

3.3 Veri Seti Metriklerinin Genel Özellikleri

Form Kullanımı Var Mı: Site ziyaretçisinden bazı veriler toplamak istediğinizde HTML Formları gereklidir. Örneğin, kullanıcı kaydı sırasında isim, e-posta adresi, kredi kartı vb. Gibi bilgileri toplamak istersiniz. Bir form site ziyaretçisinden girdi alacak ve daha sonra bunu CGI, ASP Script veya PHP komut dosyası gibi bir arka uç uygulamasına gönderecektir. Arka uç uygulaması, içindeki tanımlanmış iş mantığına göre aktarılan veriler üzerinde gerekli işlemleri gerçekleştirecektir uygulama. Metin alanları, metin alanı alanları, açılır menüler, radyo düğmeleri, onay kutuları vb. Gibi çeşitli form öğeleri mevcuttur.

POST Method Kullanımı Var Mı: Post methodlar AJAX kontrolleri ile genelde dışarıya çeşitli protokoller üzerinden veri aktarımı ve veri alımını sağlar bu tarz işlevler genel olarak e-posta içeriklerinde sıkıntı yaratabilecek tehlikelerin başında gelir. Çünkü saldırganlar yaptıkları saldırılardan elde edecekleri verileri bu yöntem sayesinde kullanıcının habersiz olarak kendisine göndermesini sağlamaktadırlar.

Input Kullanımı Var Mı: Formlarda kullanılan üç tür veri giriş alanı vardır; Tek satırlı metin girişi alanları, bu alan, arama kutuları veya adlar gibi yalnızca bir satır kullanıcı girişi gerektiren öğeler için kullanılır. HTML <input> etiketi kullanılarak oluşturulurlar. Bir diğeri parola giriş kontrolleridir; bu aynı zamanda tek satırlık bir metin girişidir, ancak bir kullanıcı girer girmez karakteri maskeler. Ayrıca HTMl <input> etiketi kullanılarak oluşturulurlar. Son veri giriş tipi ise çok satırlı metin giriş kontrollerinden oluşur. Bu, kullanıcının tek bir cümleden daha uzun olabilecek ayrıntıları vermesi gerektiğinde kullanılır. Çok satırlı giriş kontrolleri, HTML <textarea> etiketi kullanılarak oluşturulur. Saldırganlar genel olarak kredi kartı bilgileri, şifreleri gibi bilgilerin tedarik edilmesi amacıyla pek çok oltalama saldırısı içeriğinde bu kontrollerin hepsini sıkça kullanırlar. Bu nedenle çok güvenilen bir siteden geldiğini düşünseniz bile verilerinizin çeşitli ortamlar tarafından ele geçebileceği ihtimalini unutmayarak özellikle mail gibi ortamlar aracılığıyla bu verileri paylaşmamak en sağlıklı yöntemdir.

Resim Kullanımı Var Mı: Görsellerin güzelleştirilmesi ve birçok karmaşık kavramı e-posta içeriklerinde basit bir şekilde tasvir edilmesi çok önemlidir, çünkü özellikle saldırı içeriklerinde resimlerin ve çeşitli görsellerin kullanılması kullanıcıları gerçekliğe ve kurumsal bir imaj izlenimi verilip kandırılması için oldukça önemlidir.

Düğme Kullanımı Var Mı: Düğmeler özellikle girilen bilgilerin mail gönderiminden farklı olarak post methodlarının tetiklenmesi için ve parametre giriş alanlarında paylaşılan bilgilerin üçüncü taraflara aktarılması için kullanılabilmektedir. Oldukça tehlikelidir çünkü verilen bilgilerin daha hızlı aksiyon alınabilmesi için otomatik işletilmesi için saldırı postalarında kullanılmaktadır. Posta girişlerinde bu sebeple yasal bir gönderenden bile gelse her hangi bir düğme ile etkileşime girilmemelidir.

Submit Methodu Kullanılmış Mı: Düğme gibi alanların arka ucunda girilen bilgilerin iletilmesi için ön yüz kodlarının bulunduğu HTML ve Javascript alanlarında kullanılmaktadır. Bu methodun düğmelere bağlı olarak kullanılmış olması verilerin aktarılması ihtimalini daha da güçlendirmektedir.

UTF-8 Karakter Seti Haricinde Kullanım Var Mı: Bu işlem özellikle yurt dışı kaynaklı saldırılarda yapılmaktadır. Rusya tabanlı saldırılarda özellikle arka yüz kodlamalarında Kiril alfabesi ve karakter setine ait az bile olsa kesinlikle kullanımlar bulunmaktadır. Bu sebeple bu karakter setleri UTF-8 uyumlu olmadığından, buna aykırı durumda olan karakterlerin bulunup bulunmadığına bakarak bu tespit edilmeye çalışılmıştır.

Checkbox Kullanımı Var Mı: Checkbox kontrolünün kullanımı özellikle form gibi alanların olduğu senaryolarda destekleyici özelliği ile senaryoyu daah gerçekçi hale getirmek için kullanılmaktadır. Genel olarak buradan çıkarılan 1-0 değerlerinin herhangi bir tehlikesi olmasa da kullanıcıyı inandırmak için bu tarz kontroller kullanılarak e-posta içeriği zenginleştirilebilmetledir.

Şifre Kullanımı Var Mı: Bu kontrol Input HTML elementi ile birlikte gelmektedir ve maskelenmiş KVKK verisi ve parola gibi yüksek gizlilik seviyesi içeren kritik verilerin maskelenerek alınması için kullanılmaktadır. Oltalama saldırılarında özellikle en sık kullanılan kontroldür. Bu tarz bir kontrolün size geldiği görülen istisnasız her posta içerik ile herhangi bir temasta bulunmadan silinmeli ve spam olarak bildirilmelidir. Ayrıca her ne kadar bu alan için kullanılan veriler ön yüzde maskelenmiş olarak görülse de arka planda iletim sırasında açık olarak tüm veriler izlenebilmektedir.

Link Kullanımı Var Mı: Link kullanımı özellikle bu tarz saldırı işlemlerinde üçüncü parti sitelere yönlendirme, linkte yer alan scriptlere tıklayarak yönlendirilmiş kod çalıştırılıp virüs bulaştırma ve ekrandaki tüm verilerin yine link üzerinde yer alan özellikler ile elde edilebilmesi için kullanılabilmektedir. Maskelenerek gösterilen link her ne olursa olsun arka planda açılan pencereler farklı olabilmektedir. Bu sebeple posta içerisinde gelen link gibi içeriklerle ettileşim kurmak oldukça tehlikeli senaryolardan biridir.

Kara Listede Yer Alan Link Var Mı: Özellikle bu tarz saldırıların gözlemlendiği sistemlerde zarar veren çeşitli linkler bir kara liste oluşturularak önlem amaçlı kayıt altında tutulmaktadır. Bu linkler öncelikli olarak kontrol edilerek yer aldığının tespit edilmesi durumunda doğrudan postanın engellenmesi yada spam olarak etiketlenmesi işlemi günümüzde diğer sistemler tarafındanda yapılmaktadır.

Başlık Kullanımı Var Mı: Başlık kullanımı tüm posta içeriklerinde mevcut olabilir. Burada saldırı olup olmadığı konusunda dikkat edilecek husus başlığın içeriğe ne kadar uygun olduğu içerdiği ifadeler, karakter uzunluğu ve kullanılan karakterler ile gönderen kurum ile tutarlı bir içeriğe sahip olmasıdır.

Başlık Özel Karakter İçeriyor Mu: Başlığın özel karakter içermesi özellikle yurt dışı kaynaklı saldırılarda karşılaşılan en yaygın örneklerden biridir. Başlık alanında içerikle tutarlı olmayan durumlarda kesin olarak postanın saldırı amaçlı olduğu tespitine varılabilir.

E-Posta Kullanımı Var Mı: E-posta input kontrolü çoğunlukla zararsızdır. Fakat senaryo gereği şifre, kullanıcı adı ve hesap numarası gibi bilgilerle birlikte hazırlanan form içerisinde senaryonun inandırıcı olması için kullanılmaktadır. Ayrıca burada kullanıcılar tarafından verilen e-posta adresleri bir sonraki saldırılarda hedef olmak üzere saldırganların atak listelerine eklenmektedir. Bu gibi listelere eklenmek ise sürekli olarak saldırı hedefi haline getirebilmektedir.

Script Window Kullanımı Var Mı: Script.window kullanımı çoğunlukla posta içeriklerinde yer alan kontroller için javascript kodları üzerinden çeşitli işlemler yapılması amacı ile kullanulabikmektedir. Özellikle postalarda amacın sadece kişilerin bilgi ve belge paylaşımı olduğu düşünülürse javascript tarafında script.window gibi özelliklerin kullanımı oldukça amacın dışında ve gereksiz olabilmektedir.

IFrame Kullanımı Var Mı: Iframe kullanımı script.window ile aynı amaçlı ve aynı işlevleri taşımaktadır birbirlerinin yerine kullanılabilmektedir. Özellikle posta içerikleri içerisinde IFrame ve window.script kullanılması oldukça gereksiz ve amaç dışıdır.

Date/Time Kullanımı Var Mı: Posta içeriklerinde tarih, saat ve zaman bilgilerinin tutulması için kullanılabilmektedir. Bu tarz bilgiler normal şartlarda e-postaların başlık bilgileri içerisinde de bulunmaktadır.

İsim/Soyisim Kullanımı Var Mı: Kredi/Banka kartı gibi bilgileri tedarik ederken çok önemlidir. Kullanıcı adı, soyadı ve kimlik numarası gibi bilgiler tedarik edilen KVKK kapsamındaki en önemli kişisel bilgilerdendir. Her türlü ortamda ödeme ve para transferinin gerçekleştirilmesi amacı ile kullanılabilmektedir.

Telefon Numarası Kullanımı Var Mı: Telefon numarasının paylaşıldığı oltalama saldırılarında telefonumuza çok bir zarar sağlamasa da yine e-posta adreslerinin sağlanması gibi daha sonraki süreçte yaşanacak sahtekarlık içeriklerinde telefonumuzun hedef haline gelmesi için kullanılmaktadır.

Metatag Kullanımı Var Mı: Metatag verileri özellikle e-posta içeriğinin güvenilir olup olmadığı konusunda belirli durumlarda bilgi sağlayabilmektedir. Standart HTML metatag etiketleri sadece sayfanın tasarım ve teknoloji özellikleri ile ilgili bilgiler içermektedir, bunlar dışında alışılmadık etiketler üçüncü parti saldırı yazılımları tarafından kullanılabilmektedir.

İndirilebilir İçerik Kullanımı Var Mı: Özellikle Link ve Href gibi elementler aracılığı ile posta içerisinde indirilebilir içerik sunulması, içeriği indirdikten sonra Truva atı ve solucan saldırısı gibi çok daha tehlikeli virüs ve ajanların bilgisayarımıza kurulmasına ve bulunduğumuz ağ sistemine sızmasına neden olabilmektedir. Bu sebeple indirilebilir içeriklerden uzak durulmalıdır. Bu tarz içerikler postalar içerisinde geldiğinde güvenilir bir kaynaktan alındığına, sağlayıcısına ve anti-virüs taramalarının yapıldığından emin olunmalıdır.

İkon Kullanımı Var Mı: İkon kullanımı özellikle banka ve resmi kurumların içerikleri taklit edilmek istenildiğinde posta içeriğine eklenmektedir. Burada önemli olan nokta şüpheli olan durumlarda copyright içeriğinin kontrol edilerek teyit edilmesi veya ikon ile ilgili bilgilerin HTML tarafında kontrol edilmesidir. Özellikle içeriğin zengin gösterilmesi ve mağdurlara daha inandırıcı senaryolar sağlanması için ikon kullanımı kritiktir.

Copyright Kullanımı Var Mı: Copyright resmi kurumlardan gelen posta içeriklerinin arka yüz kodlamalarında önemli etikiye sahiptir. Çünkü bu kurumların sahip olduğu kodlar sağlayıcılar tarafından lisans altına alınarak sunulmaktadır ve ilgili haklar bu gibi belirteçler kullanılarak posta içeriklerine dahil edilmektedir. Copyright bilgilerinde yer alan herhangi bir tutarsızlık sahtecilik tespiti konusunda önemli rol oynayabilmektedir.

HTML içeriği Okunabiliyor Mu: Antivirüs ve diğer koruma sistemleri bazen posta içeriğinin okunmasını engelleyebiliyor. Bu tarz durumlarda olası tehlikelere karşı koruma prosedürü uygulanıyor ve içerik okunamaz şekilde etiketleniyor. Okunamayan içerikler az sayıda karşılaşılan fakat kesinlikle saldırı içeriğine sahip postalardır.

Kara Listeden Kelime Kullanılmış Mı: Kara liste altına alınan risk grubu kelimeleri oluşturulmuştur. Bu listede bitcoin, blockchain, paypal, nakit, cash, havale, EFT, POS, Kredi, Öde, ethereum kelimeleri bulunmaktadır. Saldırganlar parasal transfer işlemlerini çoğunlukla kripto para satın alma işlemi üzerinden gerçekleştirdikleri için en yaygın hırsızlık yöntemi olarak gözükmektedir ve paranın transfer edilmesi bölümünün yer aldığı javascript kodlarında yada POST işlemlerinde bu tarz kelimelere rastlanabilmektedir.

Gizli Element Kullanımı Var Mı: Gizli element kullanımları çoğunlukla sayfanın arka planında veri tutmak ve işlem yapmak amacı ile yapılır. Bu tarz işlemler her zaman saldırı amaçlı olarak kullanılmasa da saldırı içeriklerinde sık olarak karşımıza çıkmaktadır. Hidden özelliğine sahip elementler bunun dışında animasyon yada cache işlemi için kullanılabilmektedir.

Yönlendirme Kullanımı Var Mı: Redirect element kullanımı link ve href özelinde yönlendirme amacı ile sıkça kullanılmaktadır. Posta içeriklerinin arka planında yapılan bu tarz üçüncü parti kaynak yönlendirmeleri saldırı amaçlı olabilmektedir. Bu sebeple bu tarz redirect etiketine sahip içeriklerle karşılaşıldığında detaylı olarak incelenmelidir.

Input Element Sayısı: Formlarda kullanılan input element sayıları postanın saldırı olup olmadığını anlayabilmemiz açısından bize fikir sağlayabilir. Özellikle saldırı amaçlı posta içerikleri çok sayıda farklı parametre ve veri içeriğine ihtiyaç duyduğundan bu bilgileri çok sayıda input kontrolünün kullanıcıya uygun bir senaryo aracılı ile sunularak girişi sağlanması üzerine kurulmaktadır. Normal posta içeriklerinde pek çok zaman hiçbir input element bulunmamaktadır.

Option Element Sayısı: HTML option elementini ifade etmektedir, modelleme için posta içeriklerinde kaç tane option elementinin bulunduğu da bu kapsamda incelenmektedir.

TH Element Sayısı: TH elementi formlar içerisinde yer alan tablolarda bulunan satırların hücre sayılarını ifade eder. Bu elementin kullanılması inputların konumlandırılması için zorunludur bu sebeple burada sayısal olarak formların ne kadar detay içerdiğini detaylandırmak önemlidir.

TR Element Sayısı: Tablolarda yer alan satır sayısını belirlemektedir, bu şekilde bir postada kullanıcıdan kaç satır ve kaç hücrede veri girişi istenildiği bilgisi elde edilebilmektedir.

Table Element Sayısı: Posta içeriklerinde kaç tane veri girişi amaçlı tablo kullanımı olduğu bilgisini sağlamaktadır. Çoğunlukla bir tablo ve 3 adet veri giriş alanı hırsızlık için yeterli olmaktadır.

Href Element Sayısı: Href elementi içerikte yer alan link ve URL için kullanılmaktadır. Normal posta içeriklerinde referans göstermek yada kurumların medya ortamlarının iletişimlerini paylaşmak amacıyla kullanımları mevcuttur fakat çok sayıda kullanım olduğu zaman istenmeyen durumların tespiti için fikir sağlayabilmektedir.

Div Element Sayısı: HTML elementleri arasında en yaygın kullanımı olandır, elementler arası bağlantıların kurulması ve sayfa içerisinde konumlandırılması için kullanılmaktadır, bu sebeple div sayısı içeriğin ne kadar fazla sayıda element ve karmaşık bir içeriğe sahip olduğu konusunda bize fikir sağlamaktadır. Karmaşık ve çok sayıda element içeren posta içerikleri saldırı postalarında kullanıcıları yanılgıya düşürmek için sık kullanılan bir yöntemdir.

Span Element Sayısı: HTML içeriğinde yer alan span element sayısını vermektedir. Buradan sağlanan ortalama element sayıları ile HTML sayfa içeriğinde normalden farklı seyreden bir durum olup olmadığı konusunda bilgi sağlanmaktadır.

Article Element Sayısı: Posta HTML içeriğinde yer alan Article element sayısını vermektedir. Farklılık içeren elementlerin çok sayıda kullanımı gönderilen postanın ne kadar geniş kapsamlı ve karmaşık olduğu konusunda fikir sağlamaktadır.

Hidden Element Sayısı: Gizli element kullanım sayısını vermektedir. Diğer elementlerde olduğu gibi ne kadar sayıda görünmez element ve veri kullanımı olduğu bilgisini sağlamaktadır.

P Element Sayısı: P element sayılarının çok fazla olması sayfa içerisinde açıklama metinlerinin kaç farklı kontrolde yer verildiğini göstermektedir. P sayısının çokluğu aynı zamanda metinin görsel mi yoksa metin ağırlıklı mı olduğu bilgisini sağlamaktadır.

İçerik Özel Karakter Sayısı: İçerikte yer alan özel karakterler ne kadar çok olursa güvenilirlik o kadar azalır. Çünkü UTF-8 dışında kalan ve alfa-numerik dışında kalan tüm özel karakterler çok az sayıda kullanılmaktadır. Kullanımları ne kadar artarsa alıcıdan istenen veri sayısı çoğalmaya ve posta içeriğinin karmaşıklık oranı artmaya başlamaktadır.

İçerikte Yer Alan Kelime Sayısı: İçerikte yer alan kelime sayısı postanın ne kadar uzun ve kaç kelimeden oluştuğu bilgisini içermektedir. Kişiler arası konuşmaların genelde kısa ve daha öz terimler içerdiği herkes tarafından bilinmektedir. Veri tedarik amaçlı hazırlanan saldırı senaryoları karmaşık bir içeriğe sahip olduğu için çok sayıda kelime içermektedir.

Kara Liste Kelime Sayısı: Yukarıda belirtilen kara liste kelimelerinden kaç tanesinin posta içeriğinde yada HTML ile javascript kodlarında yer aldığını ifade etmektedir. Kara listede yer alan kelimelerin çok sayıda oluşu postanın amacının parasal bir içeriğe sahip olduğunu ve muhtemel bir para transferi ile devam edebilecek bir süreç içerisinde bulunduğu konusunda fikir sağlamaktadır.

Http Link Sayısı: http protokolü ile erişilen SSL sertifikasına sahip olmayan kaç adet web sayfası linki yer aldığı belirtilmektedir. Güvensiz siteler genel anlamda saldırıya daha açık olunduğu konusunda fikir vermektedir.

Https Link Sayısı: Https protokolü ile erişilen SSL sertifikasına sahip kaç adet web sayfası linki yer aldığı belirtilmektedir. Güvenilir siteler saldırıya karşı daha fazla korumaya sahiptir ve SSL sertifikası kullanıcılar için sayfalara bağlanırken güvenilir bir protokol sunmaktadır.

Metatag Sayısı: HTML sayfalarının bulunduğu içeriklerde kaç tane metatag verisi bulunduğu belirtilmektedir çoğunlukla normal bir posta yada web sayfasında az sayıda metatag etiketi bulunmaktadır. Az rastlanılan ve çok sayıda metatag etiketinin kullanılması posta yada web sayfasının güvenilirliğini azaltmaktadır ve risk oranını arttırmaktadır.

HTML Element Sayısı: Tüm posta içerisinde toplamda kaç adet element kullanıldığını belirtmektedir. Saldırı postalarında veri toplama ve kullanıcıdan bilgi temin etmek gibi amaçlar olduğu için HTML elementlerinin sayısı çoğalmaktadır. HTML element sayısı ne kadar az sayıda olursa kullanıcıdan veri beklentisi ve karmaşıklık seviyesi o kadar düşmektedir.

Link Sayısı: Posta içerisinde kaç adet farklı web sitesine ait link bulunduğunu belirtmektedir. Farklı sitelere yönlendirmeler saldırganlar tarafından çoğunlukla virüs ve Truva atı saldırıları için kullanılmaktadır.

Checkbox Sayısı: Posta içeriğinde kullanılan form da yer alan checbox adetini belirtmekedir. Checkbox elementleri anket dışında normal posta içeriklerinde kullanılmazlar, bu sebeple saldırganların kurguladıkları karmaşık senaryoları inandırıcı hale getirmek için kullanılmaktadır.

Button Element Sayısı: Düğme elementleri posta içeriklerinde yer aldığı zaman POST ve submit gibi veri aktarma işlemlerini tetiklemek için javascript yönlendiricisi olarak kullanılırlar. Bu element dış ortamlarla postaların adresler arası gönderimi dışında işlevsellik için kullanılırlar.

Image Element Sayısı: E-posta içerisinde yer alan görselleri belirtmektedir. Saldırı postalarında tasarlanan senaryoya uygun olacak şekilde desteklenmesi için kullanılmaktadır.

Title Uzunluğu: Başlık uzunluğunun çok uzun ve karmaşık olması kişiler tarafında anlaşılması güç durumlara ve karmaşıklığa yol açabilmektedir. Bu sebeple kullanıcıların şüphesini çekmemek için genelde gerçeğine yakın, yalın ve sade başlık seçimleri yapılmaktadır.

En Uzun Kelime Uzunluğu: İçerikte yer alan kelimelerden en uzun harf sayısına sahip olanı temsil etmektedir. Kişilerin yazdığı postalarda uzun kelimeler kullanıcılar sebebiyle daha sık karşılaşılmaktadır.

En Kısa Kelime Uzunluğu: En kısa kelime uzunluğunu ifade etmektedir. Türkçe gibi sondan eklemeli diller içinde oldukça faydalı olmak ile beraber İngilizce gibi farklı dil ailesine dahil ekleme içermeyen dil grupları için katkı sağlamayabilir.

İçerik Uzunluğu: İçeriğin kaç kelime yada karakterden oluştuğu gibi uzunluk bilgileri çoğu zaman ciddi katkılar sağlamaktadır. Posta içeriklerini yazan kişilerin kullandığı kelimeler çoğunlukla belli bir standart ve ortalama uzunluğa sahiptir, Görsel yada input içeren saldırı postalarında ise kompleks senaryolar oluşturularak kullanıcılara sunulduğu için bu verilerin sayısı daha yüksek olabilmektedir.

Tablo 3.3-Veri Seti Min-Maks-Ort Değerleri

ÖZELLİK	MİN.	MAKS.	ORT.	PHISH ORT.	TEMİZ ORT.
Form Kullanımı Var mı	0	1	0,8	0,97	0,32
POST Method Kullanımı Var mı	0	1	0,67	0,70	0,48
Input Element Kullanımı Var mı	0	1	0,84	0,91	0,28
Image Kullanımı Var mı	0	1	0,89	0,92	0,81
Button Kullanımı Var mı	0	1	0,45	0,76	0,18
Submit Methodu Kullanılmış mı	0	1	0,57	0,62	0,44
Non UTF-8 Char Kullanımı Var mı	0	1	0,72	0,91	0,60
Checkbox Kullanımı Var mı	0	1	0,28	0,30	0,09
Password Kullanımı Var mı	0	1	0,37	0,92	0,09
Link Kullanımı Var mı	0	1	0,52	0,98	0,36
Kara Listede Yer Alan Link Var mı	0	1	0,11	0,02	0,13
Title Kullanımı Var mı	0	1	0,45	0,48	0,43
Title Özel Karakter Kullanımı Var mı	0	1	0,75	0,90	0,68
E-Mail Element Kullanımı Var mı	0	1	0,33	0,31	0,42
Script Window Kullanımı Var mı	0	1	0,39	0,42	0,34
Iframe Kullanımı Var mı	0	1	0,65	0,69	0,61
Date/Time Kullanımı Var Mı	0	1	0,59	0,55	0,63
Name/Surname Var mı	0	1	0,28	0,47	0,19
Phone Var mı	0	1	0,88	0,92	0,31
Metatag Kullanımı Var mı	0	1	0,76	0,78	0,74
İndirilebilir İçerik Kullanımı Var mı	0	1	0,69	0,39	0,87
Cookie Kullanımı Var mı	0	1	0,58	0,70	0,29
Cache Kullanımı Var mı	0	1	0,75	0,78	0,70
Icon Kullanımı Var mı	0	1	0,81	0,67	0,95
Copyright Kullanımı Var mı	0	1	0,62	0,54	0,73
HTML İçeriği Okunabilir mi	0	1	0,67	0,49	0,89
Kara Listeden Kelime Kullanılmış mı	0	1	0,74	0,88	0,17
Hidden Element Kullanımı Var mı	0	1	0,47	0,71	0,38
Redirect Element Kullanımı Var mı	0	1	0,49	0,51	0,33
Input Element Sayısı	1	13	5,75	7,66	3,93
Option Element Sayısı	0	7	6,48	5,33	6,09
Select Element Sayısı	0	4	2,28	2,76	2,11
TH Element Sayısı	1	8	5,34	5,78	4,81
TR Element Sayısı	2	12	8,66	8,79	7,94

Table Element Sayısı	0	3	1,87	1,68	2,37
LI Element Sayısı	0	6	3,38	3,76	3,32
UL Element Sayısı	0	6	4,11	4,08	4,43
Href Element Sayısı	1	3	2,21	2,74	1,27
Div Element Sayısı	3	10	5,43	5,22	6,02
Span Element Sayısı	0	3	1,02	1,43	0,88
Article Element Sayısı	0	2	0,81	1,08	0,72
Hidden Element Sayısı	1	6	3,41	4,26	3,22
P Element Sayısı	0	8	3,13	3,08	3,19
Content Özel Karakter Sayısı	7	29	9,64	15,17	7,46
Content Kelime Sayısı	12	49	18,54	26,02	17,14
Kara Liste Kelime Sayısı	0	9	2,41	3,38	0,32
HTTP Link Sayısı	0	7	2,21	1,55	2,32
HTTPS Link Sayısı	0	3	0,55	0,38	0,72
MetaTag Sayısı	2	11	4,68	5,66	4,12
HTML Element Sayısı	10	25	16,72	22,66	14,53
Link Sayısı	0	5	2,36	4,21	1,12
Checkbox Sayısı	0	4	1,34	2,16	0,32
Button Sayısı	1	3	1,12	2,01	0,33
Image Sayısı	0	3	1,58	2,33	1,42
Title Uzunluğu	3	39	23,22	21,37	26,42
En Uzun Kelime Uzunluğu	9	18	14,79	14,98	14,58
En Kısa Kelime Uzunluğu	1	3	1,16	1,14	1,18
Content Uzunluğu	57	112	79,61	87,45	69,95

4. YÖNTEM

Yapılan çalışma birçok basamaktan oluşmaktadır. İlk adımda veriler web sayfasına benzer e-posta içeriklerinin ön yüz kodlarının .html ya da .js dosyalarını elde ederek başlamaktadır. Biz burada verileri hazır halde elde edebileceğimiz https://phishtank.com/ sitesinden tedarik ettik. Veri toplanırken verinin tutarlı olması, makine öğrenmesi yöntemleri ile saldırı yada temiz içerik diye sınıflandırabilme yeteneklerine uygun hizmet edebilecek olması göz önünde bulundurulmuştur.

İkinci adım olarak toplanan bu veriler bir araya getirildikten sonra standart normalizasyon işlemine tabii tutulmuştur. Normalizasyon aşamasında verilerin tamamı python üzerinde işlenmiştir ve feature extraction diye tanımlanan özellik çıkarımı teknikleri kullanılmıştır. Burada çıkarılan her bir özellik aslında veri setinin bir sütununa denk gelmektedir. Her e-posta içeriği ise bir satıra denk gelmektedir.

Normalize edilmiş olan veriler Çapraz Doğrulama yöntemi kullanılarak makine öğrenmesi algoritmaları tarafından sırasıyla eğitilmiştir. Daha önceki bölümlerde bahsedildiği gibi 8 adet makine öğrenmesi algoritması kullanılmıştır. Çıkan sonuçlar daha sonra K-Fold tekniğine göre rastgele test için çalışma anında tahsis edilen veriler ile karşılaştırılarak Karışıklık Matrisine yansıtılmıştır.

Tüm bu hesaplamalar bittikten sonra Karışıklık Matrisinde “Doğru” değerlerinin yüksek ve tutarlı olması beklenmektedir. Bu bize veri setindeki sınıfların dağılımının uygun sayılarda olduğunu ve yapılan eğitim ile testler sonucunda başarım oranına göre kullanılabilir modeller elde edilip edilemediğini gösterir. Aynı zamanda eşik değerleri kullanarak modellerin güvenilirliğini de belli limitlerle yönetebiliriz.

4.1 Normalizasyon İşlemi

Normalizasyon yapılmasının amacı; veri setlerinin yorumlanabilir ve yapay zeka algoritmaları ile eğitilebilir tutarlı bir veritabanı sağlanmasıdır. Burada yapılan işlem sayesinde satır ve sütunlardan oluşan tamamen nümerik ve Standart veritabanı sorgulama ve veri bütünlüğü niteliklerine uygun, mükerrer durumları ve gereksiz veri kullanımını önleyerek veri performansı artırmak yani algoritmalarla eğitimi sağlanabilecek en kullanılabilir veri kümesini elde etmektir. Bu çalışmada kullanılan normalizasyon işlemi Python üzerinde varolan kütüphanelerle ihtiyaç özelinde fonksiyonlar kodlanarak tam anlamıyla sağlanmıştır. Normalizasyon süreçlerinde, X_{yeni} yeni hesaplanacak değerdir. Yani normalize edilmiş değer de denilebilir. X_{min} değeri bu parametre başlığı altındaki en küçük değerdir. X_{max} değeri ise bu parametre başlığı altındaki en büyük değerdir. X’in mevcut değerinin minimum değer ile

farkının, maksimum değer ile minimum değer arasındaki farka oranı yeni X değerini vermektedir. [24]

4.2 Çapraz Doğrulama (CV)

CV, makine öğrenimi modellerinin başarılarını değerlendirmek için kullanılan bir yöntemdir. Python yazılım dilinde hazır olarak K-Fold kütüphanesi altında uygulama örnekleri ve standartları mevcuttur. Bu yöntemde, veri seti eğitim ve test setine ayrılmıştır ve bu işlem için seçilen yöntem modelin başarısını önemli ölçüde etkilemektedir [16]. Ayrıca yöntemi diğer doğrulama yöntemlerinden ayırarak öne çıkaran nokta ise, doğrulama veri setlerinin rastgele küme olarak seçilerek, test sonuçlarının gerçeğe en yakın ve tutarlı sonuçlar getirmesini sağlamasıdır. Bu çalışmada CV kullanımının nedeni; Pek çok farklı HTML içeriğine sahip olan düzensiz bir veriden düzenli veriye geçilmesi ve statik test verisinin belirlenmesi durumunda sadece belli bir bölüm veri üzerinde doğru sonuçlar çıkarabileceği fakat test edilemeyen pek çok veri için daha yanlış ve düşük başarım oranlarının getirmesini önlemektir. Bu sayede ise verilerin daha iyi kullanılmasına yardımcı olur ve algoritma performansı hakkında daha fazla bilgi verir. Bu yöntem kullanılmadan önce bir K-Fold değeri seçilir. Bu çalışmada, k değeri 10 seçilmiştir. Bunun anlamı veri kümesi 10 eşit parçaya bölünmüştür; test verileri her seferinde test edilen verinin test içinde Şekil 4.1'de gösterildiği gibi kullanıldığından emin olmak için bir adım kaydırır. Sonuçta her işlemden ayrı olarak üretilir. Bu sonuçların aritmetik ortalaması alınarak genel bir hata veya başarı metriği sonucu elde edilmiştir. Nihai sonuç olarak çıkan son ortalama son kullanıcıya sunulur ve kullanılacak modelin en kararlı başarı oranı elde edilmiş olur.

	1. Parça	2. Parça	3. Parça	4. Parça	5. Parça	6. Parça	7. Parça	8. Parça	9. Parça	10. Parça
1. Adım	Test	Eğitim	Eğitim	Eğitim	Eğitim	Eğitim	Eğitim	Eğitim	Eğitim	Eğitim
2. Adım	Eğitim	Test	Eğitim	Eğitim	Eğitim	Eğitim	Eğitim	Eğitim	Eğitim	Eğitim
3. Adım	Eğitim	Eğitim	Test	Eğitim	Eğitim	Eğitim	Eğitim	Eğitim	Eğitim	Eğitim
4. Adım	Eğitim	Eğitim	Eğitim	Test	Eğitim	Eğitim	Eğitim	Eğitim	Eğitim	Eğitim
5. Adım	Eğitim	Eğitim	Eğitim	Eğitim	Test	Eğitim	Eğitim	Eğitim	Eğitim	Eğitim
6. Adım	Eğitim	Eğitim	Eğitim	Eğitim	Eğitim	Test	Eğitim	Eğitim	Eğitim	Eğitim
7. Adım	Eğitim	Eğitim	Eğitim	Eğitim	Eğitim	Eğitim	Test	Eğitim	Eğitim	Eğitim
8. Adım	Eğitim	Eğitim	Eğitim	Eğitim	Eğitim	Eğitim	Eğitim	Test	Eğitim	Eğitim
9. Adım	Eğitim	Eğitim	Eğitim	Eğitim	Eğitim	Eğitim	Eğitim	Eğitim	Test	Eğitim
10. Adım	Eğitim	Eğitim	Eğitim	Eğitim	Eğitim	Eğitim	Eğitim	Eğitim	Eğitim	Test

Şekil 4.1- Kullanılan CV Modeli

4.3 Karışıklık Matrisi (CM)

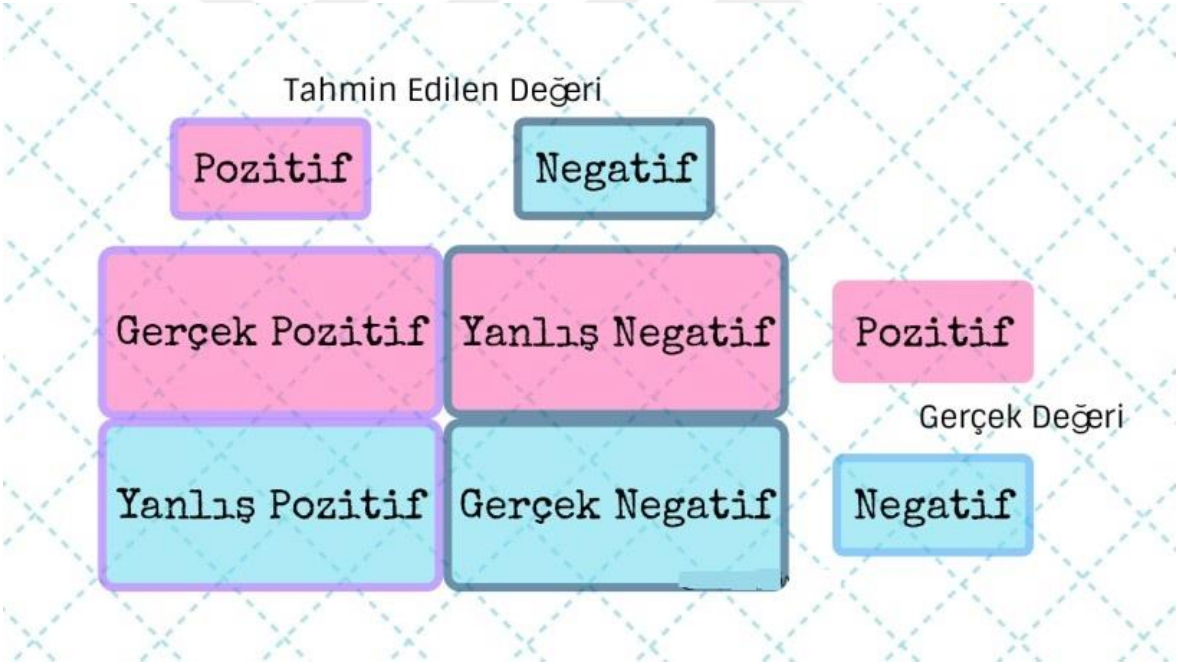
Karışıklık matrisi, makine öğreniminde önceden belirlenmiş hedef veri kümelerinden elde edilen modellerin performansını değerlendirmek için en yaygın kullanılan yöntemdir [5]. Matriste pozitif ve negatif veriler değerlendirme sonuçlarına göre gösterilir. Gelinen aşamada matris ile gösterim alınan sonuçları analiz etme açısından yarar sağlamıştır. Karışıklık matris modeli Şekil 4.2'de gösterilmiştir.

• **Gerçek Pozitif (TP):** Test verilerindeki değer, model değerlerinin sınıfıyla aynıdır. Doğru sınıflandırma yapılmıştır.

• **Yanlış Negatif (FN):** Test verilerindeki değer, model tarafından üretilen sınıftan farklıdır. Yanlış sınıflandırma yapılmıştır.

• **Yanlış Pozitif (FP):** Gerçek değer negatiftir ancak pozitif olarak sınıflandırılmıştır. Yanlış sınıflandırma yapılmıştır.

• **Gerçek Negatif (TN):** Gerçek değer negatiftir ve negatif olarak sınıflandırılmıştır. Doğru sınıflandırma yapılmıştır.



Şekil 4.2- Karışıklık Matrisi Modeli

Karışıklık matrisi ile mevcut veri kümesinin; doğruluk (accuracy), kesinlik (precision) ve duyarlılık (recall) değerleri hesaplanmıştır. Doğruluk değeri denklemi, kesinlik değeri denklemi, duyarlılık değeri denklemi formülleri ile hesaplanmıştır. [25]

$$Doğruluk\ (Accuracy) = (TP + TN) / (TP + TN + FP + FN)$$

$$Kesinlik\ (Precision) = TP / (TP + FP)$$

$$Duyarlılık\ (Recall) = TP / (TP + FN)$$

4.4 Modelde Kullanılan Teknoloji

Tez çalışması yazılım olarak Python dili 3.8 versiyonu ile birlikte kullanılmıştır. IDE olarak ise Visual Studio Code ve Python araçları ve eklentileri kullanılmıştır. Python'un tercih edilmesinin sebebi kolaylıkla verileri analiz edebilme yeteneğidir ve ayrıca kullanışlı arayüzü, tercih edilmesinin nedenleri arasındadır. Python, sahip olduğu geniş kütüphanelerle yazılım sürecini normale göre daha hızlanmasını sağlar. Scipy ve Numpy gibi bilimsel hesaplamalardan scikit-learn gibi makine öğrenimini doğrudan sağlayan ve algoritmaları bir standart olarak kütüphaneler aracılığı ile sağlayan araçlara kadar gerekli yazılım desteği sağlanmaktadır.

Visual Studio Code platformu veri analizi için gerekli kaynak kütüphanelerini içinde barındırdığından ve tüm diller ile birlikte esnek kullanım kolaylığı sağlamasından dolayı süreci hızlandırmıştır. Kullanılan kodlama ve mimari daha detaylı derin öğrenme süreçleri sağlayan Google ürünleri ile de tamamen uyumludur.

Kullanılan kütüphaneler:

- Makine öğrenmesi algoritmaları için sckit-learn,
- Dizi işlemleri için numpy,
- Veriyi görselleştirmek için mathplotlib,
- Veriyle işlemler yapmak ve ayrıştırmak için pandas kütüphaneleri kullanılmıştır.

4.4 Modelde Kullanılan Donanımlar

Tablo 4.1- Kullanılan Bilgisayar Özellikleri

Bilgisayar	Lenovo–YOGA 730 15IWL
İşletim Sistemi	Microsoft Windows 10 Pro
CPU	Intel® Core™ i7-8565U CPU @ 1.80GHz 2.00GHz, 2301 Mhz, 8 Cores
RAM	16,00 GB
IDE	PyCharm 2019.3.3 Community Edition

Model eğitim ve testleri için kullanılan bilgisayar özellikleri Tablo 4.1 üzerinde belirtildiği şekilde Lenovo-YOGA serisi 730 modelidir. İşletim sistemi olarak Microsoft Windows 10 Pro tercih edilmiştir. Bilgisayarın sahip olduğu donanımlara bakmak gerekirse Intel Core i7

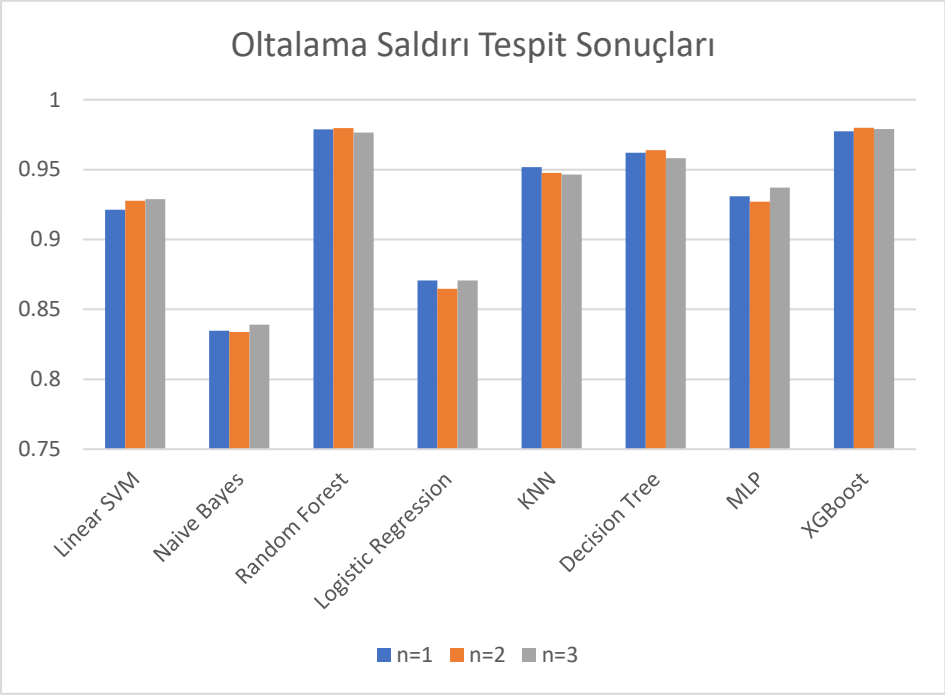
çekirdek setine sahip işlemci 2.00GHz işletim sistemi gücüne ve 8 çekirdek kapasitesine sahip. 16GB Ram üzerinde çalışılmıştır ve Özellik çıkarımı ile diğer tüm test ve performans ölçümü işlemleri için Python için en yaygın IDE konumunda olan PyCharm 2019.3.3 ücretsiz versiyonu tercih edilmiştir.

5 TEST SONUÇLARI VE DEĞERLENDİRMELER

Bu bölümde tüm veri kümelerinin önerilen yaklaşım tarafından yapılan testlerin tüm sonuçları açıklanmakta ve grafiklerde gösterilmektedir. Bu çalışmada, 58 olan özellik sayısına 8 farklı algoritma tarafından yaklaşılmıştır. Birbirine yakın olmakla beraber performans ve başarım oranı açısından farklı sonuçlar gözlemlenebilmektedir.

Sonuç:

Araştırmada kullanılan algoritmalar; veri kümelerindeki ortalama değerler, Şekil 5.1'de gösterilmiştir. Örneğin, değerler hesaplanırken PC1 veri kümesinin doğruluk değeri, tüm algoritmalarda doğruluk sonuçlarının toplamlarının aritmetik ortalaması olarak alınmıştır. Oluşan grafiğe göre sonuçlar;



Şekil 5.1- Algoritma Bazında Veri Seti Metrikleri

Tablo 5.1- Algoritma Bazında Karışıklık Matrisi Sonuçları

	SVM	***NB***	***RF***	***LR***	***KNN***	***DT***	***MLP***	***XGBoost***
Doğruluk	0.927	0.834	0.979	0.865	0.943	0.963	0.935	0.978
TP	5006	3547	5276	4852	5000	5091	5003	5263
FP	432	1891	162	586	438	347	435	175
FN	589	373	140	1261	328	181	825	118
TN	7764	7980	8213	7092	8025	8172	7528	8235
Hassasiyet	0.95	0.81	0.98	0.92	0.95	0.96	0.95	0.98
Geri Çağırma	0.93	0.96	0.98	0.85	0.96	0.98	0.90	0.99
F1-Skoru	0.94	0.88	0.98	0.88	0.95	0.97	0.92	0.98

Tablo 5.2- Algoritma Bazında Başarı Oranı ve Eğitim Süreleri

Algoritma	**Başarı Oranı**	**Süre**
Saf Bayes	0.834	12.7
Rastgele Orman	0.979	520.1
Destek Vektör Makinesi	0.927	7252.3
Lojistik Regresyon	0.865	65.4
K-En Yakın Komşu	0.943	193.1
Karar Ağacı	0.963	22.3
Çok Katmanlı Algılayıcı	0.935	38.6
XGBoost	0.978	78.2

1- Değerlendirilen algoritmalar içerisinde en başarılı model, Rastgele Orman algoritması ile elde edilmiştir.

2- Aşırı Gradyan Tahminleme algoritması da Rastgele Orman algoritmasına çok yakın bir sonuç elde etmiştir. Bunun yanında bu algoritma eğitim süreleri bakımından çok daha iyi bir performansa sahiptir.
3- Eğitim süreleri ve başarı oranı parametrelerini bir arada değerlendirdiğimiz zaman en ideal algoritma Aşırı Gradyan Tahminleme algoritması olarak gözükmektedir.
4- NB ve SVM tüm algoritmalar içerisinde en düşük başarı oranına sahip algoritmalar olarak belirlenmiştir.
5- Birbirine benzer olan bu algoritmalardan Destek Vektör Makinesi algoritmasının hesaplama süreleri bakımından performansı da oldukça yavaş ve düşüktür.
6- NB tüm algoritmalar arasında en hızlı eğitim sürecine sahip olsa da, sağlamış olduğu başarı oranı ile maalesef güven vermemektedir.
7- Hesaplama süreleri ve başarı oranları bir arada değerlendirildiğinde bu model için 3 algoritma kullanılabilir ideal modeller olarak öne çıkmaktadır. (Aşırı Gradyan Tahminleme, MLP, Karar Ağacı Algoritması)

6 SONUÇLAR

Son yıllarda, bilgisayar kullanımı son derece artmaktadır. Hayatımızın her yönünde bilgi teknolojileri yer almaktadır. Bu nedenle neredeyse tüm gerçek dünyalar işlemler siber dünyaya aktarılır. Bu gelişmeler hayatımızı oluştursa da birçok alanda kolay, aynı zamanda problemlerin kaybolmasını da beraberinde getirir, özellikle güvenlik sorunları nedeniyle anonim yapı İnternetin siber saldırı yapmak çok da karmaşık değildir. Acemi bir kullanıcı bile bazı basit saldırılar yapabilir. Oltalama saldırısı yöntemi en çok tercih edilen güvenlik ihlallerinden biridir, çünkü doğrudan zayıflığı ve kullanıcı hatasını kullanarak hedeflerine ulaşıyor saldırganlar. Geleneksel güvenlik mekanizmaları bu tür saldırılardan bizi maalesef koruyamaz. Bu nedenle bazı ek güvenlik mekanizmalar geliştirilmelidir. Bu projede içerik tabanlı bir kimlik avı uyguladık, tespit sistemimiz Web sayfasının metni ve ek özelliklerini analiz eden ve web sayfasının dolandırıcı bir web sayfasıdır ya da değildir diye 0 ile 1 arasında sonuç döndüren bir makine öğrenmesi modeli oluşturduk. Bu yaklaşımda 8 farklı yapay zeka algoritması problemi anlamak ve yapay zeka algoritmalarının verimlerini karşılaştırmak için kullanılmıştır. Deney sonuçları gösterdi ki önerilen model, kabul edilebilir ve standart son kullanıcı için verimli bir güvenlik sağlayabilir seviyededir. Önümüzdeki süreçte daha fazla çalışma ve yaklaşım bir arada kullanılarak modelin verimini arttırmaya yönelik çalışmalar gerçekleştirilebilir, karmaşık durumlar ve tespit verimliliğini arttırmak için bazı hibrit modeller kullanmaya çalışmak sistemin verimliliği açısından oldukça faydalı olacaktır. Ayrıca derin öğrenme gibi yeni modellerin kullanılması hedeflenmektedir. Derin öğrenmesi tabanlı hibrit modeller ile daha yüksek verimli ve daha güvenilir modeller elde edilmesi beklenmektedir. Ayrıca elde edeceğimiz yeni veri setleri ile de daha verimli ve öğrenimi yüksek seviyede tutan modeller üretebiliriz.

KAYNAKÇA

[1] M. M. Group, «World Internet Usage & Population Statistics,» Internet World Stats - https://www.internetworldstats.com/stats.htm, Rishitas, India, 2020.

[2] www.emarketer.com, «Global E-Commerce,» 3 5 2019. [Çevrimiçi]. Available: https://www.emarketer.com/content/global-ecommerce-2019. [Erişildi: 18 01 2021].

[3] J. Point, "Types of Cyber Attackers," 1 1 2018. [Online]. Available: https://www.javatpoint.com/types-of-cyber-attackers. [Accessed 1 10 2020].

[4] Ministry of Transport Maritime Affairs and Communications,, "National Cyber Security Strategy," Republic of Turkey, Ankara, 2016-2019.

[5] purplesec.us, "The Ultimate List of Cyber Security Statistics for 2019," 31 12 2019. [Online]. Available: https://purplesec.us/resources/cyber-security-statistics/. [Accessed 15 9 2020].

[6] A. Orunsolu, "A Middleware Based Anti-Phishing Architecture," Cyber Security Trends, Abeokuta, 2016.

[7] M. Chawla ve S. Chouhan, «A Survey of Phishing Attack Techniques,» *International Journal of Computer Applications,* no. 93, 2014.

[8] F. Josh, «www.csoonline.com,» csoonline, 4 9 2020. [Çevrimiçi]. Available: https://www.csoonline.com/article/2117843/what-is-phishing-how-this-cyber-attack-works-and-how-to-prevent-it.html. [Erişildi: 18 1 2021].

[9] N. Chou, R. Ledesma, Y. Teraguchi ve J. Mitchell, «Client Side Defense Against Web-Based Identity Theft,» %1 içinde *The ISOC symposium on Network and Distributed System Security*, San Diego, February 2004.

[10] C. Cludmark, «Anti-Phishing Working Group,» 28 06 2006. [Çevrimiçi]. Available: www.cloudmark.com/releases/docs/wp_unique_approach_10550406.pdf. [Erişildi: 20 8 2020].

[11] E. Inc, «EarthLink Toolbar,» 14 12 2006. [Çevrimiçi]. Available: http://www.earthlink.net/software/free/toolbar/.

[12] M. Inc., "McAfee SiteAdvisor," 14 12 2006. [Online]. Available: http://www.siteadvisor.com/. [Accessed 20 11 2020].

[13] R. Basnet, A. Sung ve Q. Liu, «Rule Based Phishing Attack Detection,» Sam Houston State University, Texas, 2012.

[14] D. Miyamoto, H. Hazeyama and Y. Kadobayashi, “An Evaluation of Machine Learning-based Methods for Detection of Phishing Sites,” in *Advances in Neuro-Information Processing, 15th International Conferance*, Auckland, New Zelland, 2008.

[15] F. Birihanu and E. Siraj, “A Literature Review Study of Software Defect Prediction using Machine Learning Techniques,” *International Journal of Emerging Research in Management & Technology*, vol. 6, no. 6, pp. 300-306, 2017.

[16] A. Karimi, J. Fada, J. Lui, J. Braid, M. Koyuturk and R. French, “Feature Extraction, Supervised and Unsupervised Machine Learning Classification of PV Cell Electroluminescence Images,” in *IEEE Conference*, Waikoloa Village, HI, USA, USA, 2018.

[17] M. Ferreira, L. Vismari, P. Cugnasca, J. Almeida ve J. Camargo, «A Comperative Analysis of Unsupervised Learning Techniques for Anomaly Detection in Railway Systems,» %1 içinde *IEEE*, Boca Raton, FL, USA, 2019.

[18] S. Sayad, «https://www.saedsayad.com,» 2017. [Çevrimiçi]. Available: https://www.saedsayad.com/naive_bayesian.htm. [Erişildi: 30 12 2020].

[19] M. Siddharth ve L. Hao, «Machine Learning for Subsurface Characterization,» 1 1 2020. [Çevrimiçi]. Available: https://www.sciencedirect.com/topics/engineering/random-forest. [Erişildi: 18 1 2021].

[20] S. University, «https://nlp.stanford.edu/,» Stanford University, 1 1 2008. [Çevrimiçi]. Available: https://nlp.stanford.edu/IR-book/html/htmledition/support-vector-machines-the-linearly-separable-case-1.html. [Erişildi: 18 1 2021].

[21] J. Walters-Williams ve Y. Li, «Comparative Study of Distance Functions for Nearest Neighbors,» %1 içinde *Advanced Techniques in Computing Sciences and Software Engineering. Springer, Dordrecht.*, Dordrecht, 2009.

[22] T. D. Science, «https://towardsdatascience.com,» Towards Data Science, 2020. [Çevrimiçi]. Available: https://towardsdatascience.com/https-medium-com-vishalmorde-xgboost-algorithm-long-she-may-rein-edd9f99be63d. [Erişildi: 18 1 2021].

[23] Phistank.com, «www.phistank.com,» 1 1 2020. [Çevrimiçi]. Available: www.phistank.com. [Erişildi: 18 1 2021].

[24] M. A. Jammal ve H. F. Rezhna, «Data Normalization and Standardization: A Technical Report,» The Machine Learning Lab. at Koya University, Erbil, Iraq, 2014.

[25] S. Visa, B. Ramsay, A. Ralescu ve E. Knaap, «Confusion Matrix-based Feature Selection,» %1 içinde *Proceedings of The 22nd Midwest Artificial Intelligence and Cognitive Science*, Ohio, USA, 2011.