Cilt: 1, Sayı: 1, 2025, Sayfa: 1-26

Yönetim Bilişim Sistemleri Derneği Dergisi

ISSN: 3108-6454

Journal of Management Information Systems Association

https://www.ybs.org.tr/ybsdd

URL VE HTML ÖZELLİKLERİNE DAYALI YIĞIN MODELİYLE KİMLİK AVI WEB SİTELERİNİN TESPİTİ

Aleyna TORLAK[a], Esra ODABAŞ YILDIRIM[b] ve Üstün ÖZEN[c]

Makale Bilgisi

Makale Türü
Araştırma Makalesi

Gönderim Tarihi:
16/10/2025

Kabul Tarihi:
26/11/2025

Anahtar Kelimeler:
Makine Öğrenimi, Lojistik Regresyon, Kimlik Avına Karşı Koruma, HTML Dizesi Yerleştirme, Yığın Modeli

Özet

Bu çalışmada, URL ve HTML özelliklerinden yararlanarak kimlik avı web sitelerinin tespiti için bir yığın (stacking) modeli önerilmektedir. Geliştirilen model, üçüncü taraf servislerine ihtiyaç duymadan hafif, gerçek zamanlı ve maliyet etkin bir tespit mekanizması sunmaktadır. Yığın modelinin Level-0 katmanında Naive Bayes, Karar Ağacı, Lojistik Regresyon, Destek Vektör Makineleri (SVM) ve Çok Katmanlı Algılayıcı (MLPClassifier) gibi sınıflayıcılar; Level-1 katmanında ise meta öğrenici olarak Lojistik Regresyon kullanılmıştır. Model performansı, 130.127 web sayfasından oluşan "phish.csv" veri seti üzerinde değerlendirilmiştir. Temel sınıflayıcılar arasında SVM %90,60 doğruluk ve %95,19 ROC AUC değeri ile en yüksek performansı göstermiştir. MLPClassifier %91,16 doğruluk, Lojistik Regresyon %90,50 doğruluk, Naive Bayes %88,94 doğruluk ve Karar Ağacı %90,62 doğruluk değerleri elde etmiştir. Yığın modelinin meta öğrenicisi ise tüm modellerden daha iyi performans göstererek %92,15 doğruluk ve %96,47 ROC AUC değerine ulaşmıştır. Bulgular, önerilen yığın modelinin farklı algoritmaların güçlü yönlerini bir araya getirerek kimlik avı tespitinde daha yüksek doğruluk ve güvenilirlik sağladığını ortaya koymaktadır.

[a] Yüksek Lisans Öğrencisi, Atatürk Üniversitesi, Yönetim Bilişim Sistemleri, Erzurum/Türkiye, aleyna.torlak19@ogr.atauni.edu.tr ORCID: 0009-0008-1833-2843
[b] Dr. Öğr. Üyesi, Atatürk Üniversitesi, Yazılım Mühendisliği, Erzurum/Türkiye, esra.odabas@atauni.edu.tr ORCID: 0000-0002-0936-1342
[c] Prof. Dr., Atatürk Üniversitesi, Yönetim Bilişim Sistemleri, Erzurum/Türkiye, uozen@atauni.edu.tr ORCID: 0000-0002-7595-4306 (Sorumlu Yazar)

DETECTION OF PHISHING WEBSITES USING A STACKING MODEL BASED ON URL AND HTML FEATURES

Article Information

Article Type
Research Article

Submission Date:
16/10/2025

Acceptance Date:
26/11/2025

Keywords:
Machine Learning, Logistic Regression, Anti-Phishing, Html String Embedding, Stacking Model

Abstract

This study proposes a stacking model for detecting phishing websites using URL and HTML features. The developed model provides a lightweight, real-time, and cost-effective detection mechanism without relying on third-party services. In the stacking architecture, Naive Bayes, Decision Tree, Logistic Regression, Support Vector Machines (SVM), and Multi-Layer Perceptron (MLPClassifier) are employed as Level-0 base classifiers, while Logistic Regression is used as the Level-1 meta-learner. The model's performance was evaluated on the "phish.csv" dataset, which consists of 130,127 web pages. Among the base classifiers, SVM achieved the highest performance with 90.60% accuracy and a 95.19% ROC AUC score. MLPClassifier reached 91.16% accuracy, Logistic Regression 90.50%, Naive Bayes 88.94%, and Decision Tree 88.94%. The meta-learner of the stacking model outperformed all individual models, achieving 92.15% accuracy and a 96.47% ROC AUC score. The findings demonstrate that the proposed stacking model enhances phishing detection accuracy and reliability by combining the strengths of multiple machine learning algorithms.

1. GİRİŞ

İnternetin hızlı gelişimi ve dijitalleşmenin yaygınlaşması, çevrimiçi ortamları hem bireyler hem de kurumlar için vazgeçilmez hâle getirmiştir. Ancak bu dönüşüm, siber saldırıların çeşitlenmesine ve karmaşıklığının artmasına da zemin hazırlamıştır. Özellikle kimlik avı (phishing) saldırıları, son yıllarda küresel ölçekte en hızlı büyüyen siber tehditlerden biri olarak tanımlanmaktadır. Anti-Phishing Working Group'un (APWG, 2023) raporuna göre kimlik avı girişimleri 2022–2023 yılları arasında tarihsel olarak en yüksek seviyelere ulaşmıştır. Kimlik avı saldırıları, sahte ve meşru görünümlü web siteleri aracılığıyla kullanıcıların banka giriş bilgileri, kredi kartı verileri veya kimlik numaraları gibi hassas bilgilerinin ele geçirilmesini hedeflemekte ve bu nedenle ciddi güvenlik riskleri oluşturmaktadır (Aleroud ve Zhou, 2017).

Saldırganların sürekli olarak tekniklerini güncellemesi ve sahte web sitelerini gerçek sitelere neredeyse birebir benzer şekilde tasarlaması, kullanıcıların bu tehditleri fark etmesini zorlaştırmaktadır (Jain ve Gupta, 2018). Bu noktada, makine öğrenmesi tabanlı yaklaşımlar, kimlik avı tespitinde giderek daha etkili hâle gelmiş ve geleneksel kural tabanlı yöntemlere kıyasla daha yüksek tespit başarısı sunduğu literatürde birçok çalışmada gösterilmiştir (Basnet vd., 2011; Zaki, 2020). Özellikle URL ve HTML tabanlı özelliklerin bir arada değerlendirilmesi, kimlik avı sitelerinin yapısal ve içeriksel özelliklerinin daha iyi anlaşılmasını sağlayarak tespit doğruluğunu artırmaktadır (Aburrous vd., 2010).

Bu çalışmada, kimlik avı web sitelerinin tespiti için URL ve HTML tabanlı özellikleri birlikte değerlendiren bir yığın modeli yaklaşımı önerilmiştir. Yığın modeli, ensemble öğrenme stratejilerinden biri olup farklı sınıflandırıcılardan elde edilen tahminlerin meta bir öğrenici tarafından yeniden

işlenmesine dayanan bir meta-öğrenme yapısıdır (Wolpert, 1992). Çalışmada Level-0 katmanında Naive Bayes, Karar Ağacı, Lojistik Regresyon, SVM ve MLPClassifier temel sınıflandırıcıları kullanılmış; Level-1 katmanında ise bu tahminleri birleştirerek nihai çıktıyı üreten meta öğrenici olarak Lojistik Regresyon tercih edilmiştir. Buradaki "katman" kavramı yapay sinir ağlarındaki parametrik ağ katmanlarıyla ilişkili olmayıp yalnızca yığın modelinin öğrenici seviyelerini ifade etmektedir. Bu yapı sayesinde model, farklı algoritmaların güçlü yönlerini bir araya getirerek tekil modellerin üzerinde bir sınıflandırma performansı elde etmeyi amaçlamaktadır.

Yığın modeli, üçüncü taraf kara liste servislerine ihtiyaç duymadan gerçek zamanlı ve maliyet etkin bir kimlik avı tespit mekanizması sunmaktadır. URL tarafında uzunluk, alt alan adı yapısı, şüpheli karakter kullanımı ve hassas kelimeler gibi yapısal özellikler; HTML tarafında ise bağlantı yapısı, form etiketleri ve içerik yoğunluğu gibi nitelikler giriş değişkenleri olarak değerlendirilmiştir. Bu özniteliklerin makine öğrenmesi algoritmalarıyla bütünleştirilerek işlenmesi sonucunda, kimlik avı amaçlı web sitelerinin yüksek doğrulukla sınıflandırılabildiği görülmüştür.

2. TEORİK ARKAPLAN

Günümüzde makine öğrenmesi temelli kimlik avı tespit çalışmaları, farklı algoritmaların birlikte kullanıldığı yığın öğrenme modellerine dayanmaktadır. Bu kısımda, çalışmada kullanılan temel yöntem ve algoritmaların teorik temelleri özetlenmiştir.

2.1. Yığın Öğrenme (Stacking) Yöntemi

Yığın öğrenme ya da yığın genellemesi, birden fazla sınıflandırma modelini bir araya getirerek tahmin performansını iyileştirmeyi amaçlayan bir topluluk öğrenme tekniğidir. Yığın öğrenme modellerinin temel fikri, çeşitli modellerin güçlü yönlerinden yararlanarak zayıf yönlerini hafifletmektir. Tipik bir çerçevesinde, birden fazla temel model (level-0 modelleri olarak da bilinir) aynı veri seti üzerinde eğitilir ve bu modellerin tahminleri, daha üst düzey bir model (level-1 modeli) için giriş özellikleri olarak kullanılır ve nihai tahmini yapar. Bu yaklaşım, bireysel modellerin kaçırabileceği çeşitli veri desenlerini yakalayarak doğruluğu artırabilir (Zou ve Nan, 2019).

Kimlik avı tespit bağlamında, yığın öğrenme modelleri Destek Vektör Makineleri (SVM), Lojistik Regresyon ve Çok Katmanlı Algılayıcı (MLPClassifier) gibi farklı algoritmaları entegre ederek URL ve HTML özelliklerine dayalı olarak kimlik avı web sitelerini tespit edebilen güçlü bir sistem oluşturabilir. Bu modellerin birleştirilmesiyle, yığın öğrenme yaklaşımı genel tespit oranını iyileştirebilir ve siber güvenlik uygulamaları açısından kritik olan yanlış pozitifleri azaltabilir (Al-Sarem vd., 2021).

2.2. Destek Vektör Makineleri (SVM)

Destek Vektör Makineleri (SVM), sınıflandırma ve regresyon problemlerinde kullanılan denetimli öğrenme modelleridir. SVM'nin kuramsal temeli Vapnik ve Chervonenkis tarafından tanımlanan İstatistiksel Öğrenme Teorisine dayanmaktadır (Vapnik ve Chervonenkis, 1974). Günümüzde kullanılan

modern SVM formülasyonu ise Vapnik'in 1995 tarihli çalışmasıyla geliştirilmiştir (Vapnik, 1995). SVM'nin temel amacı, yüksek boyutlu özellik uzayında farklı sınıfları en geniş marj ile ayıran optimum hiper düzlemi belirlemektir. Yüksek boyutlu verilerde gösterdiği başarılı genelleme performansı nedeniyle SVM, kimlik avı tespitinde yaygın olarak tercih edilmektedir. Literatürdeki çalışmalar, SVM'nin URL ve HTML temelli özelliklere dayalı olarak web sitelerini meşru veya kimlik avı olarak yüksek doğrulukla sınıflandırabildiğini göstermektedir (Tang ve Mahmoud, 2021; Sahingoz vd., 2019).

2.3. Lojistik Regresyon

Lojistik Regresyon, ikili sınıflandırma problemleri için kullanılan istatistiksel bir yöntemdir. Verilen bir girdinin belirli bir sınıfa ait olma olasılığını lojistik fonksiyonu kullanarak modeller. Lojistik Regresyon modelinin çıktısı, girdinin pozitif sınıfa ait olma olasılığı olarak yorumlanabilen 0 ile 1 arasında bir değerdir (Shaukat, 2023).

Kimlik avı tespit bağlamında, Lojistik Regresyon bir URL veya HTML özellik setinin bir kimlik avı web sitesine ait olma olasılığını değerlendirmek için kullanılabilir. Modelin hesaplama verimliliği ve istatistiksel olarak iyi tanımlanmış yapısı, onu hem temel karşılaştırmalarda hem de yığın öğrenme çerçevelerinde tamamlayıcı bir sınıflayıcı olarak uygun hâle getirmektedir (Al-Sarem vd., 2021).

2.4. Çok Katmanlı Algılayıcı (MLPClassifier)

Çok Katmanlı Algılayıcı olan Multi-Layer Perceptron (MLP), birbirine bağlı düğümlerden (nöronlar) oluşan çok katmanlı bir yapıdır. MLP'ler, verilerdeki karmaşık desenleri öğrenme yeteneğine sahiptir ve hata geri yayılımı (backpropagation) ile model, tahmin hatalarına göre ağırlıklarını ayarlar. MLP'ler, veri içindeki doğrusal olmayan ilişkileri işleyebilir ve bu da onları çeşitli sınıflandırma görevleri için uygun hale getirir (Ariyadasa, 2020). Kimlik avı tespitinde, MLPClassifier, URL ve HTML içeriklerinden türetilen özellikler üzerinde eğitilerek kimlik avı girişimlerini tanımlayabilir. MLP'lerin büyük veri setlerinden karmaşık desenleri öğrenme yeteneği, kimlik avı tespit sistemlerinin yeteneklerini artırabilir ve yığın öğrenme çerçevesindeki diğer modellerle birleştirildiğinde daha da etkili olabilir (Shabudin vd., 2020).

2.5. Naive Bayes

Naive Bayes, sınıflandırma görevlerinde kullanılan, Bayes teoremine dayalı olasılıksal algoritmalar ailesidir. Naive Bayes'in temel prensibi, tahmin ediciler arasında bağımsızlık varsayımında bulunmasıdır; bu da olasılıkların hesaplanmasını basitleştirir. Bu varsayım, modelin girdideki özelliklere dayalı olarak bir sınıfın olasılığını verimli bir şekilde hesaplamasını sağlar (Uppalapati, 2023).

Kimlik avı tespiti bağlamında, Naive Bayes sınıflandırıcıları, yüksek boyutlu verileri işleyebilme yetenekleri ve ilgisiz özelliklere karşı dayanıklılıkları nedeniyle oldukça etkili olabilir. Algoritma, bir kimlik avı web sitesinin URL özellikleri ve HTML içeriği gibi özelliklere dayalı olarak posterior

olasılığını hesaplar. Naive Bayes hem kimlik avı hem de meşru web sitelerinde özelliklerin görülme sıklığından yararlanarak, yeni örnekleri öğrenilen olasılıklara göre sınıflandırabilir (Ali, 2017). Naive Bayes'in en büyük avantajlarından biri, hızı ve verimliliğidir; bu da onu gerçek zamanlı kimlik avı tespit sistemleri için uygun hale getirir. Ayrıca, genellikle etiketlenmiş verilerin sınırlı olduğu kimlik avı tespit senaryolarında, az miktarda eğitim verisiyle bile iyi performans gösterir (Ali, 2017). Bununla birlikte, bağımsızlık varsayımı her zaman gerçek dünyada geçerli olmayabilir ve bu durum, modelin bazı bağlamlarda performansını etkileyebilir.

2.6. Karar Ağaçları

Karar Ağaçları hem sınıflandırma hem de regresyon görevlerinde kullanılan popüler bir makine öğrenimi algoritmasıdır. Karar Ağaçları, veri setini özellik değerlerine göre yinelemeli olarak alt kümelere böler ve her bir iç düğüm bir özelliği, her bir dal bir karar kuralını ve her bir iç düğümün bir özelliği, her bir dalın bir karar kuralını ve her bir yaprak düğümün ise nihai bir sonucu temsil ettiği ağaç benzeri bir yapı oluşturur (Yang vd., 2019). Bölme kriteri değişiklik gösterebilir; yaygın yöntemler arasında Gini kirliliği (Gini impurity) ve bilgi kazancı (information gain) bulunur. Kimlik avı tespitinde, Karar Ağaçları, URL'lerden ve HTML içeriğinden çıkarılan özellikler arasındaki karmaşık ilişkileri etkili bir şekilde modelleyebilir. Bu algoritma, sınıflandırmaların arkasındaki karar verme sürecini anlamayı sağlayan açık ve anlaşılır bir model sunar. Bu açıklanabilirlik, siber güvenlik uygulamalarında özellikle değerlidir; çünkü bir tespitin mantığını anlamak, tespit stratejilerini geliştirmeye ve kullanıcıları kimlik avı tehditleri hakkında eğitmeye yardımcı olabilir (Purwanto vd., 2022). Karar Ağaçları hem sayısal hem de kategorik verileri işleyebilme yeteneğine sahip olup, kimlik avı tespitinde karşılaşılan çeşitli özellik türleri için çok yönlüdür. Ancak, ağaç çok derinleştiğinde aşırı öğrenmeye (overfitting) yatkın hale gelebilir. Bunu hafifletmek için, az önem taşıyan dalları kaldırarak modelin genelleme yeteneklerini artıran budama (pruning) gibi teknikler uygulanabilir (Al-Tamimi ve Shkoukani, 2023).

3. LİTERATÜR TARAMASI

Kimlik avı web sitelerinin tespiti, özellikle kimlik avı saldırıları daha karmaşık ve yaygın hale geldikçe, siber güvenlik araştırmalarında kritik bir alan haline gelmiştir. Literatürde, URL ve HTML özelliklerine dayalı stacking modelleri ile Destek Vektör Makineleri (SVM), Lojistik Regresyon ve Çok Katmanlı Algılayıcı (MLPClassifier) gibi belirli makine öğrenimi algoritmalarının kimlik avı tespitinde kullanımına ilişkin pek çok çalışma bulunmaktadır.

Kimlik avı tespitindeki son gelişmeler, birden fazla sınıflandırıcıyı entegre eden yığın (stacking) öğrenme modellerinin doğruluk oranlarını artırmada etkili olduğunu göstermektedir. Kalabarige vd. (2022), özellikle SVM ve Random Forest algoritmalarını bir araya getirdikleri çok katmanlı bir yığın öğrenme modeli önermiş ve bu yaklaşımın bireysel sınıflayıcılara kıyasla farklı veri setlerinde anlamlı ölçüde daha yüksek performans sunduğunu raporlamıştır. Bu bulgular, Ali (2017) tarafından

vurgulanan, kimlik avı tespitinde makine öğrenimi modellerinin başarımını artırmak için wrapper (sarıcı) tabanlı özellik seçimi tekniklerinin önemini desteklemektedir. Wrapper tabanlı yaklaşımda, özellik seçimi genetik algoritma (GA), parçacık sürü optimizasyonu (PSO) veya karınca kolonisi optimizasyonu (ACO) gibi sezgisel ve meta-sezgisel optimizasyon yöntemleriyle gerçekleştirilmekte olup, model performansına göre en uygun özellik alt kümesinin belirlenmesine olanak sağlamaktadır. Bunun yanında Bhagat (2023), URL ve HTML özniteliklerini birlikte işleyen bir özellik sınıflandırma tekniğini Aşırı Öğrenme Makinesi (Extreme Learning Machine, ELM) algoritmasıyla birleştirerek kimlik avı sitelerinin daha etkili biçimde tanımlanabileceğini ortaya koymuştur. Bu yaklaşım, kimlik avı saldırılarının farklı yapısal ve içeriksel özelliklerini aynı anda ele almanın tespit başarımını artırdığını göstermektedir.

SVM, yüksek boyutlu verileri işleme konusundaki dayanıklılığı nedeniyle kimlik avı tespitinde yaygın olarak kullanılmaktadır. Shaukat (2023), kimlik avı web sitelerinin metin tabanlı sınıflandırmasında SVM kullanarak %88 doğruluk elde ettiklerini bildirmiş ve SVM'nin URL, metin ve görüntü özniteliklerine dayalı olarak kimlik avı ve meşru siteler arasında ayrım yapmadaki etkinliğini göstermiştir. Bu bulgu, SVM'yi özellik ağırlıklandırmada partikül sürü (particle swarm) optimizasyonu ile birleştiren ve geliştirilmiş tespit doğruluğu elde eden Ali ve Malebary'nin (2020) çalışması tarafından da doğrulanmaktadır.

Lojistik Regresyon, güçlü istatistiksel temellere dayanan ve belirli varsayımların sağlanmasını gerektiren bir yöntem olmasına rağmen, yorumlanabilirliği sayesinde kimlik avı tespitinde de sık kullanılan modellerden biridir. Shaukat (2023) tarafından yapılan çalışmada, Lojistik Regresyon %87 doğruluk oranına ulaşmış ve kimlik avı web sitelerini sınıflandırmadaki kullanışlılığını göstermiştir. Bu model, özellikle bireysel özelliklerin tahmin sonuçları üzerindeki etkisini değerlendirme olanağı sunduğu için, sonuçların yorumlanabilir olmasının hedeflendiği durumlarda avantajlıdır.

MLPClassifier, bir tür yapay sinir ağı olarak kimlik avı web sitelerini tespit etme potansiyeli için incelenmiştir. Kalla (2023) tarafından yapılan çalışma, MLP'nin kimlik avı URL'lerini tespit etmek için doğal dil işleme teknikleri ile kullanımının etkinliğini vurgulamıştır. MLP'nin verilerden karmaşık desenleri öğrenme yeteneği, özellikle bir topluluk öğrenme yaklaşımında diğer modellerle birleştirildiğinde kimlik avı tespiti için uygun bir aday haline gelmektedir.

Literatür, çeşitli algoritmaların güçlü yönlerini birleştiren hibrit ve yığın öğrenme modellerine doğru bir eğilim olduğunu göstermektedir. Örneğin, Almalki (2023), kimlik avı tespit yeteneklerini artırmak için birden fazla makine öğrenme tekniğini entegre eden akıllı bir model önermiştir. Bu yaklaşım, kimlik avı tehditlerinin dinamik doğasına yanıt verebilecek uyarlanabilir ve sağlam tespit sistemlerine duyulan ihtiyacın artmakta olduğunu yansıtmaktadır.

Hassan (2015), Naive Bayes, Karar Ağaçları ve Destek Vektör Makineleri (SVM) gibi çeşitli sınıflandırıcıların halka açık veri setlerinde karşılaştırmalı bir analizini yapmıştır. Sonuçlar, Naive

Bayes'in özellikle özellik seçimi teknikleriyle birleştirildiğinde rekabetçi bir performans sergilediğini ve sınıflandırma doğruluğunu artırdığını göstermiştir. Bu, modelin URL ve HTML özelliklerine dayalı olarak kimlik avı girişimlerini tespit etmedeki faydasını ortaya koymaktadır. Ek olarak, Shinde vd., (2015) tarafından yapılan bir çalışmada, kimlik avı web sitelerini tespit etmek için K-Means kümeleme ve Naive Bayes kombinasyonu kullanılmıştır.

Kümeleme yaklaşımı, verileri organize etmeye yardımcı olurken, Naive Bayes sınıflandırma için olasılıksal bir çerçeve sağlamıştır ve bu da modelin kimlik avı tespiti senaryolarındaki çok yönlülüğünü göstermektedir.

Karar Ağaçları ise, kimlik avı tespitinde yorumlanabilirlikleri ve hem kategorik hem de sayısal verileri işleyebilme yetenekleri nedeniyle popüler bir makine öğrenimi algoritmasıdır. Algoritma, veri setini özellik değerlerine göre yinelemeli olarak bölerek, her bir düğümün belirli bir özellikle ilgili bir kararı temsil ettiği ağaç benzeri bir yapı oluşturur (Ogonji, 2023). Kimlik avı tespitinde, Karar Ağaçları, URL'lerden ve HTML içeriklerinden çıkarılan çeşitli özellikler arasındaki ilişkileri etkili bir şekilde modelleyebilir. Örneğin, Uppalapati (2023), kimlik avı tespit modelleri üzerine karşılaştırmalı bir çalışmada Karar Ağaçları da dahil olmak üzere birden fazla algoritmayı incelemiştir. Bulgular, Karar Ağaçlarının net karar yolları sağlayabildiğini ve kullanıcıların sınıflandırmaların arkasındaki mantığı anlamasını kolaylaştırdığını göstermiştir.

Ogonji (2023), kimlik avı saldırılarını tespit etmek için Karar Ağaçlarını içeren hibrit bir model önermiştir. Çalışma, model performansını artırmada özellik seçimi ve ön işlemenin önemini vurgulamış ve Karar Ağaçlarının uygun özellik mühendisliği teknikleriyle birleştirildiğinde kimlik avı web sitelerini etkili bir şekilde tespit edebileceğini göstermiştir. Ayrıca, Al-Tamimi ve Shkoukani (2023) tarafından yapılan çalışma, kimlik avı URL'lerinden çıkarılan özellikler aracılığıyla kimlik avı web sitelerini tespit etmede Karar Ağaçlarının etkililiğini vurgulamıştır. Çalışma, Karar Ağaçlarının kimlik avı girişimlerini doğru bir şekilde sınıflandırabileceğini ortaya koymuş ve bu algoritmaların bu alandaki kullanımını güçlendirmiştir.

Zhu vd. (2019) tarafından tartışıldığı gibi özellik seçimi yöntemlerinin entegrasyonu, kimlik avı tespit modellerinin performansını iyileştirmek için kritik öneme sahiptir. En ilgili özelliklere odaklanarak, araştırmacılar modellerinin verimliliğini ve doğruluğunu artırabilir, böylece yanlış pozitif ve negatif sonuçları azaltabilirler. Tiny-Bert Yığın Modeli tabanlı yaklaşım, URL'leri özellik vektörlerine dönüştürerek ve temel öğrenicilerle bir yığın modeli oluşturarak kimlik avı web sitelerini tespit etmede üstün performans sergilemiştir. Bu yaklaşım, geleneksel yöntemlere kıyasla daha az manuel özellik çıkarma gerektirir ve yüksek doğruluk oranlarıyla kimlik avı tehditlerine karşı etkili bir savunma mekanizması sunar (He vd., 2024). Fazal (2023) ise, çeşitli web sitesiyle ilgili özellikler içeren bir veri setini analiz etmek için Karar Ağaçlarını kullanmıştır. Araştırma, model performansını artırmada özellik

seçimi ve ön işlemenin önemini vurgulamış ve kimlik avı tespitinde Karar Ağaçlarının etkililiğini daha da doğrulamıştır.

Yığın öğrenme yöntemlerinin optimize edilmesiyle elde edilen model, diğer tespit yöntemlerine göre üstün performans göstermiş ve yüksek doğruluk oranlarına sahip olmuştur. Yığın öğrenme yöntemleri, kimlik avı tespiti alanında etkili bir savunma mekanizması oluşturabilir (Al-Sarem vd.,2021). Özellik tabanlı bir makine öğrenme çerçevesi, kimlik avı tehditlerini doğru bir şekilde tespit etmede etkili olmuştur. Rao ve Pais (2019), farklı özellik çıkarma tekniklerini ve makine öğrenme algoritmalarını kullanarak yüksek doğruluk oranları elde etmiş ve kimlik avı web sitelerini meşru sitelerden ayırt edebilmiştir. Zamir ve Khan (2020), farklı makine öğrenme algoritmalarının kimlik avı tespiti üzerindeki etkinliklerini değerlendirmiştir. Rastgele ormanlar ve Destek Vektör Makineleri gibi karmaşık modellerin daha yüksek doğruluk sağladığı gözlemlenmiştir. Doğru sınıflandırma algoritmaları, kimlik avı tespitinde kritik bir rol oynamaktadır. Ayrıca, Igwilo ve Odumuyiwa (2022), Yığın öğrenme ve non-ensemble algoritmalarının kimlik avı URL'lerini tespitindeki performanslarını karşılaştırmıştır. Yığın Modeli öğrenicisinin URL kimlik avı tespitinde üstün yetenek gösterdiği sonucuna ulaşılmıştır.

4. YÖNTEM

4.1. Araştırma Tasarımı

Bu çalışma, kimlik avı web sitelerinin tespitinde farklı makine öğrenimi algoritmalarının performanslarını karşılaştırmak ve bu algoritmaları bir yığın model çerçevesinde bütünleştirmek amacıyla yürütülmüştür. Araştırma, deneysel ve karşılaştırmalı bir tasarım benimsemiştir. Yöntem hem URL hem de HTML temelli özellikleri kullanarak web sayfalarının kimlik avı olup olmadığını sınıflandırmaya odaklanmaktadır. Çalışmanın genel akışı aşağıdaki adımlardan oluşmaktadır:

1. Veri setinin toplanması ve ön işleme,
2. URL ve HTML temelli özelliklerin çıkarımı,
3. Beş temel makine öğrenimi algoritmasının eğitimi (Level-0),
4. Tahmin çıktılarının meta sınıflayıcıya aktarılması (Level-1),
5. Modelin performans metrikleriyle değerlendirilmesi.

Çalışmanın deneysel süreci şu adımlardan oluşmaktadır:

1. Veri setinin yüklenmesi ve ön işleme adımlarının uygulanması,
2. URL ve HTML temelli özellik çıkarımı,
3. Beş temel algoritmanın eğitimi ve doğrulama süreci,
4. Yığın öğrenme modelinin oluşturulması ve meta sınıflayıcının eğitimi,
5. Tüm modellerin test seti üzerindeki performans karşılaştırması,
6. ROC eğrileri ve karışıklık matrislerinin görselleştirilmesi.

Sonuç olarak, yığın öğrenme modeli tekil algoritmalara göre daha yüksek doğruluk (%91,16) ve ROC AUC (%95,19) değerleri elde etmiştir.

Tablo 1. Veritabanında Kullanılan Öznitelikler

Öznitelikler				
domain_is_ip	subdomain_level	embedded_domain	num_tokens_url	sensitive_word
obfuscated_token	len_domain	len_path	len_url	num_dots_url
suspicious_token_url	out_of_position_tld	len_fqdn	len_mld	num_terms_mld
bad_forms	bad_action_fields	non_matching_urls	out_of_position_brand_name	num_terms_in_text
num_terms_in_title	num_input_fields	num_images	num_iframes	num_links
internal_link_ratio	empty_link_frequency	login_form	len_html_style	len_html_script
len_html_link	len_html_comment	len_html_form	alarm_window	redirection
hidden_content	title_domain	internal_resource_ratio	domain_occurence	len_html
len_html_text	dom_depth	dom_node_count	dom_node_type	dom_node_mean
dom_node_std	same_page_ratio	same_folder_ratio	unique_subdomain_type	unique_file_type
directory_depth	directory_path_unique	directory_path_mean	directory_path_std	dist_top_domain
script_percentage	phish			

4.2. Veri Seti

Bu çalışmada kullanılan veri seti, GitHub üzerinden elde edilen phish.csv dosyasıdır. Veri seti, toplam 130.127 web sayfasına ait 57 öznitelik içermekte olup URL yapısı, HTML içerik özellikleri, bağlantı türleri ve form etiketleri gibi web sitelerinin yapısal ve davranışsal bileşenlerini kapsamaktadır. Her gözlem etiketlenmiş bir örnek olup, “1” kimlik avı (phishing), “0” ise yasal (legitimate) web sayfasını temsil etmektedir. Veri setinin sınıf dağılımı dengesiz bir yapı göstermekte olup, 93.241 adet meşru (0) ve 36.886 adet kimlik avı (1) örneği bulunmaktadır. Veri, farklı kaynaklardan derlenmiş ve PhishTank gibi güvenilir referans veri tabanlarıyla doğrulanmıştır. Bu veri setinin tercih edilme nedenleri hem URL hem HTML tabanlı özellikleri aynı anda içermesi, güncel kimlik avı kalıplarını yansıtması ve geniş örnek hacmi sayesinde modellerin genelleme performansının değerlendirilmesine olanak sağlamasıdır.

4.3. Veri Ön İşleme

Veri ön işleme süreci, modelin öğrenme başarısını doğrudan etkileyen kritik bir aşamadır. Bu kapsamda aşağıdaki işlemler uygulanmıştır:

- Veri seti incelenmiş ve herhangi bir eksik değer bulunmadığı için silme veya doldurma işlemine ihtiyaç duyulmamıştır.
- Niteliksel (kategorik) değişkenler sayısal değerlere dönüştürülmüştür.
- Sürekli değişkenler, min–max normalizasyonu ile [0,1] aralığına ölçeklendirilmiştir:

$$X_{norm} = \frac{X - X_{min}}{X_{max} - X_{min}}$$

- Veri seti %80 eğitim ve %20 test oranıyla iki alt kümeye ayrılmıştır.

- Eğitim sürecinde, modelin aşırı öğrenmesini önlemek ve genelleme yeteneğini artırmak amacıyla 5 katlı çapraz doğrulama (5-fold cross-validation) yöntemi uygulanmıştır.

4.4. Özellik Çıkarımı

URL ve HTML yapılarından elde edilen öznitelikler, kimlik avı web sayfalarını meşru sitelerden ayıran en belirgin göstergeleri temsil etmektedir. Bu süreçte, özellik mühendisliği ve Word2Vec tabanlı HTML gömülü teknikleri kullanılmıştır.

4.4.1. URL ile İlgili Özellikler

URL tabanlı özellikler, web adresinin yapısal formundan çıkarılmıştır. Aşağıdaki 8 temel öznitelik, kimlik avı sitelerinde sıklıkla görülen karakteristik davranışlara dayanmaktadır:

1. **IP Adresi Kullanımı:** Alan adının doğrudan IP formatında olup olmadığı (ör. http://62.141.45.54/...) — kimlik avı sitelerinde sık görülür.
2. **Şüpheli Semboller:** '@', '−' ve '~' gibi karakterlerin sayısı. Özellikle '@' sembolü, kullanıcı yönlendirmesi için riskli kabul edilir.
3. **HTTPS Kullanımı:** URL'nin güvenli bağlantı protokolü (https) içerip içermediği.
4. **URL Uzunluğu:** Uzun URL'lerin genellikle kimlik avı sitelerinde bulunması.
5. **Alan Adındaki Nokta Sayısı:** Çoklu alt alan adı (ör. www.bank.login.verify.com) kimlik avı göstergesidir.
6. **Hassas Kelimeler:** "secure", "account", "login", "signin", "banking", "confirm" gibi kelimelerin varlığı.
7. **Üst Düzey Alan Adı (TLD) Özellikleri:** Alan adında birden fazla TLD kullanımı (.com.co.uk gibi).
8. **Marka Benzerliği:** URL dizelerinin hedef markaya benzerlik ölçümü (ör. Levenshtein mesafesi $< 2 \rightarrow$ benzer).

Bu özelliklerin her biri sayısal biçime dönüştürülerek (0 veya 1, ya da frekans değerleriyle) model giriş değişkeni olarak kullanılmıştır.

4.4.2. HTML ile İlgili Özellikler

HTML tabanlı özellikler, web sayfasının kaynak kodundan çıkarılmıştır. Bu özellikler, sitenin davranışsal farklılıklarını yansıtmaktadır:

1. **Dahili/Harici Bağlantılar:** Harici bağlantı oranının yüksek olması kimlik avı eğilimini artırır.
2. **Boş Bağlantılar:** <a href="#"> veya boş link sayısı.
3. **Form Etiketleri:** <form> içinde "password", "login", "signin" gibi giriş alanlarının varlığı.
4. **HTML Uzunluğu:** Kimlik avı sayfalarının genellikle kısa HTML koduna sahip olması.
5. **Alarm Penceresi:** alert() fonksiyonunun bulunup bulunmaması.
6. **Yönlendirmeler:** "redirect" anahtar kelimesinin varlığı.

7. **Gizli İçerikler:** display:none, visibility:hidden veya input type="hidden" etiketleri.
8. **Başlık Tutarlılığı:** <title> etiketiyle URL marka adının uyuşması.
9. **Marka Bağlantı Tutarlılığı:** En sık bağlantı verilen markanın URL markasıyla uyumu.
10. **Kaynak Türleri:** <link>, <script>, <img> etiketlerinin iç/dış kaynak oranı.
11. **Marka Adı Görünme Sayısı:** URL markasının HTML içinde tekrar sayısı.
12. **HTML Dizesi Yerleştirme:** HTML kodundan Word2Vec modeliyle elde edilen gömülü vektörleri.

Bu özellikler modelin öğrenme aşamasında birleştirilmiş, toplam 57 öznitelik oluşturulmuştur.

4.5. Model Konfigürasyonları ve Kullanılan Algoritmalar

Bu çalışmada stacking modelinin Level-0 (temel) katmanında beş farklı sınıflayıcı kullanılmıştır. Tüm modeller Python 3.10 ortamında, scikit-learn kütüphanesi kullanılarak uygulanmıştır. Modellerin eğitiminde kullanılan temel hiperparametre ayarları aşağıda özetlenmiştir:

- **Naive Bayes:** Varsayılan parametrelerle uygulanmıştır.
- **Karar Ağacı:** criterion = "gini" kullanılmıştır.
- **Lojistik Regresyon:** solver = "lbfgs", max_iter = 1000.
- **Destek Vektör Makineleri (SVM):** kernel = "rbf", probability = True.
- **Çok Katmanlı Algılayıcı (MLPClassifier):** hidden_layer_sizes = (10,10), activation = "relu", solver = "adam".

Bu sınıflayıcılar aynı eğitim verisi üzerinde çalıştırılmış ve ürettikleri tahminler Level-1 meta öğrenici olarak kullanılan Lojistik Regresyon modeline aktarılmıştır.

4.6. Yığın (Stacking) Modeli

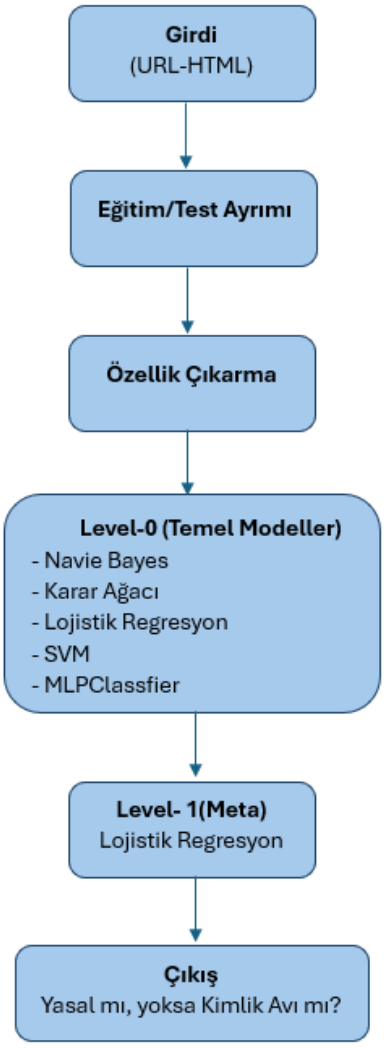
Yığın öğrenme modeli, birden fazla sınıflayıcının tahmin sonuçlarını üst düzey (meta) bir modelle birleştiren yığın öğrenme tekniğidir. Bu çalışmada;

- **Level-0 (Base Learners):** Naive Bayes, Karar Ağacı, Lojistik Regresyon, SVM ve MLPClassifier
- **Level-1 (Meta Classifier):** Lojistik Regresyon

olarak belirlenmiştir. Temel modellerin tahminleri bir ara veri setinde toplanmış ve meta sınıflayıcı, bu tahminleri girdi olarak kullanarak nihai sonucu üretmiştir. Böylece farklı modellerin güçlü yönlerinden yararlanılmış ve genel doğruluk artırılmıştır.

Bu çalışmada önerilen yığın öğrenme yapısının genel işleyişi Şekil 1'de gösterilmektedir. İlk olarak URL ve HTML özellikleri modele girdi olarak alınmakta, ardından eğitim/test ayrımı yapılmaktadır. Özellik çıkarımı sonrasında Level-0 katmanındaki temel sınıflayıcılar (Naive Bayes, Karar Ağacı,

Lojistik Regresyon, SVM ve MLPClassifier) tahminlerini üretmekte; bu tahminler Level-1 meta sınıflayıcı tarafından birleştirilerek nihai sınıflandırma gerçekleştirilmektedir.



Şekil 1.Yığın Modeli Akış Şeması.

4.7. Model Performans Değerlendirme Ölçütleri

Model performansı, aşağıdaki istatistiksel göstergelerle değerlendirilmiştir:

Karışıklık matrisindeki değerler kullanılarak sınıflandırma modelinin performansı ölçülmektedir. Bu çalışmada; doğruluk oranı, özgüllük, hassasiyet, duyarlılık, F1 Skoru ve ROC AUC (eğri altında kalan alan) performans ölçütleri dikkate alınmıştır.

Doğruluk Oranı (Accuracy): Modelin doğru sınıflandırdığı tüm örneklerin oranıdır. Genel olarak modelin ne kadar isabetli olduğunu gösterir.

$$Doğruluk\ Oranı = \frac{\mathrm{DP} + \mathrm{DN}}{\mathrm{DP} + \mathrm{DN} + \mathrm{YP} + \mathrm{YN}}$$

Özgüllük (Specificity-TNR): Gerçek negatif sınıfların doğru tahmin edilme oranıdır. Yanlış pozitiflerin etkisini azaltmak için kullanılır.

$$\text{Özgüllük} = \frac{\text{DN}}{\text{DN} + \text{YP}}$$

Hassasiyet (Precision): Pozitif olarak tahmin edilen örneklerin ne kadarının gerçekten pozitif olduğunu gösterir.

$$Hassasiyet = \frac{\text{DP}}{\text{DP} + \text{YP}}$$

Duyarlılık (Recall): Gerçek pozitiflerin model tarafından ne kadar iyi tahmin edildiğini gösterir.

$$Duyarlılık = \frac{\text{DP}}{DP + YN}$$

F1 Skoru: Precision ve Recall arasındaki dengeyi gösterir. Dengesiz veri kümelerinde özellikle yararlıdır.

$$F1\ Skoru = \frac{\text{Hassasiyet } \times \text{Duyarlılık}}{\text{Hassasiyet } + \text{Duyarlılık}}$$

ROC AUC (Eğri Altında Kalan Alan): Modelin pozitif ve negatif sınıfları ayırt etme becerisini ölçer. AUC değeri, modelin sınıfları ne kadar iyi ayırdığını gösterir.

Karışıklık matrisindeki bu değerler kullanılarak sınıflandırma modelinin performansı ölçülmektedir. DP: gerçekte doğru olan değerin doğru tahmin edilmesi, YP: gerçekte yanlış olan değerin doğru tahmin edilmesi, YN: gerçekte doğru olan değerin yanlış tahmin edilmesi, DN: gerçekte yanlış olan değerin yanlış tahmin edilmesini ifade etmektedir. Karışıklık matrisinin genel gösterimi Tablo 2'de verilmiştir.

Tablo 2. Karışıklık Matrisi

Karışıklık (Confusion) Matrisi		**Gerçek Sonuçlar**	
		Pozitif (1)	**Negatif (0)**
Tahmin Edilen Sonuçlar	**Pozitif (1)**	DP [1,1] Doğru Pozitif	YP [1,0] Yanlış Pozitif
	Negatif (0)	YN [0,1] Yanlış Negatif	DN [0,0] Doğru Negatif

5. BULGULAR

Bu kısımda, kimlik avı web sitelerinin tespitinde farklı makine öğrenimi algoritmalarını bir araya getiren yığın modelinin etkinliği değerlendirilmiştir. Çeşitli veri setlerinden elde edilen URL ve HTML tabanlı özellikler kullanılarak Destek Vektör Makineleri (SVM), Naive Bayes, Karar Ağaçları, Lojistik Regresyon ve MLPClassifier algoritmaları hem bireysel olarak hem de yığın modeli içerisinde analiz edilmiştir. Aşağıda sunulan bulgular, doğruluk, kesinlik (precision), duyarlılık (recall/TPR) ve ROC

AUC gibi temel performans ölçütleri üzerinden modellenen sistemin kimlik avı tespitindeki başarımını kapsamlı biçimde ortaya koymaktadır. Tablo 3'te temel sınıflayıcıların genel doğruluk, özgüllük (TNR), hassasiyet, duyarlılık, F1 skoru ve ROC AUC performansları gösterilmekte; Tablo 4 ise bu sınıflayıcılardan elde edilen çıktılarla eğitilen yığın (stacking) meta öğrenicinin nihai performansını sunmaktadır.

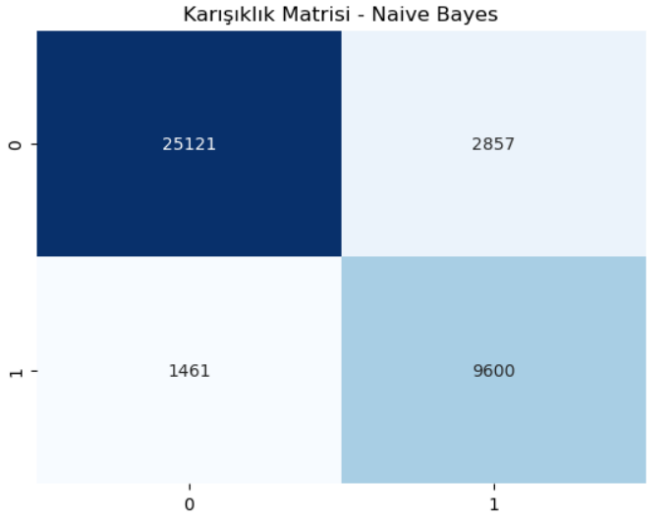
Tablo 3. Level-0 Temel Sınıflayıcılara Ait Performans Sonuçları

Model	**Genel Doğruluk (%)**	**Özgüllük (TNR) (%)**	**Hassasiyet (Precision) (%)**	**Duyarlılık (TPR) (%)**	**F1 Skoru (%)**	**ROC AUC (%)**
Naive Bayes	88.94	89.79	77.07	86.79	81.64	92.77
Karar Ağacı	90.62	92.45	81.83	86.00	83.87	89.23
MLPClassifier	91.16	93.31	83.51	85.72	84.60	89.52
Lojistik Regresyon	90.50	92.53	81.88	85.37	83.59	95.13
Destek Vektör Makinesi (SVM)	90.60	93.13	82.90	84.21	83.55	95.19

Tablo 4. Level-1 Yığın (Meta Öğrenici) Modelinin Performans Sonuçları

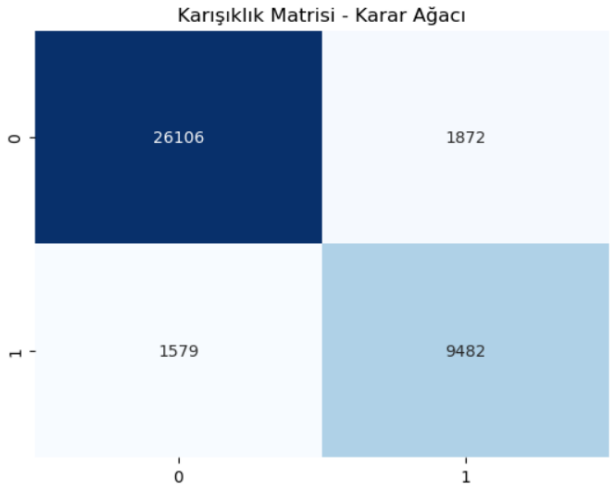
Model	**Genel Doğruluk (%)**	**Özgüllük (TNR) (%)**	**Hassasiyet (Precision) (%)**	**Duyarlılık (TPR) (%)**	**F1 Skoru (%)**	**ROC AUC (%)**
LR Meta Öğrenici	92.15	94.85	86.76	85.34	86.04	96.47

Yapılan analizler sonucu, Naive Bayes modeli, %88,94 genel doğruluk oranı ile diğer modellere kıyasla en düşük doğruluk oranına sahiptir. Ancak, %92,77 ROC AUC değeri, Naive Bayes modelinin iyi bir sınıflandırma yeteneği sunduğunu göstermektedir. Model, %77,07 hassasiyet ve %86,79 duyarlılık değerleriyle, özellikle yanlış pozitif oranını minimize etmeyi hedefleyen durumlarda etkili bir sonuç sunmaktadır. Bu problemde duyarlılık sonucunun daha yüksek olması modelin kimlik avı sitelerini tespit etmede daha başarılı olduğu sonucunu göstermektedir. Naive Bayes modeli %81,64 F1 skoru hassasiyet ve duyarlılık arasında başarılı bir denge sağladığını ortaya koymaktadır. Modelin %86,79 doğru pozitif oranı (TPR) ve %89,79 özgüllük yani doğru negatif oranı (TNR), yasal ve kimlik avı sitelerinin doğru tahmin edilme oranlarını yansıtırken, %92,77 ROC AUC değeri modelin sınıflandırma yeteneğinin yüksek olduğunu ifade etmektedir. Model, gerçek pozitiflerden 25,121'ini ve gerçek negatiflerden 9,600'ünü doğru şekilde sınıflandırmış; yanlış pozitifler 2,857 ve yanlış negatifler ise 1,461 olarak rapor edilmiştir. Naive Bayes modeli karışıklık matrisi Şekil 2'de gösterilmiştir.



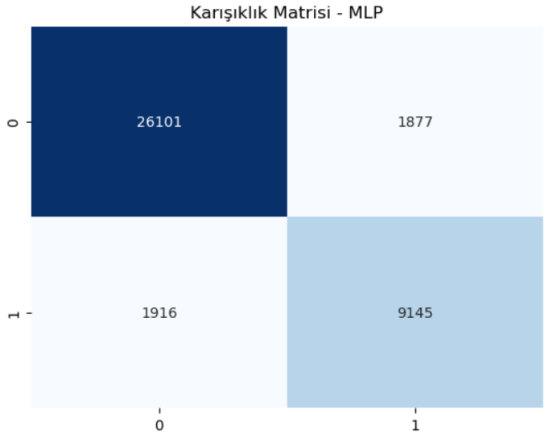
Şekil 2. Naive Bayes Modeli Karışıklık Matrisi

Karar Ağacı (Decision Tree) modeli, %90,62 genel doğruluk oranı ile yüksek bir performans sunmaktadır. %89,23 ROC AUC değeri ile etkili bir sınıflandırma gücüne sahip olup, diğer modellerle kıyaslandığında ROC AUC değeri biraz daha düşüktür. Model %81,83 hassasiyet, %86 duyarlılık sonucu elde etmiştir. Duyarlılık sonucunun daha yüksek olması ile yasal sitelerin aksine phishing sitelerini tespit etmede daha başarılı olduğu sonucu görülmüştür. Karar Ağaçları %83,87 F1 skoru hassasiyet ve duyarlılık arasında başarılı bir denge sağladığını ortaya koymaktadır. Modelin %86 doğru pozitif oranı (TPR) ve %92,45 doğru negatif oranı (TNR), yasal ve kimlik avı sitelerinin doğru tahmin edilme oranlarını yansıtırken, %89,23 ROC AUC değeri modelin sınıflandırma yeteneğinin yüksek olduğunu ifade etmektedir. Model, gerçek pozitiflerden 26106'sını ve gerçek negatiflerden 9482'sini doğru şekilde sınıflandırmış; yanlış pozitifler 1872 ve yanlış negatifler ise 1579 olarak rapor edilmiştir. Karar Ağacı modeli karışıklık matrisi Şekil 3'te gösterilmiştir.



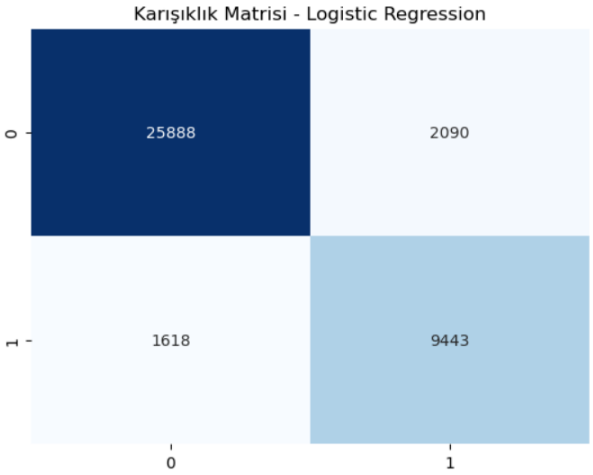
Şekil 3. Karar Ağacı Modeli Karışıklık Matrisi

Çok Katmanlı Algılayıcı (MLPClassifier) modeli, %91,16 genel doğruluk oranı ile en yüksek performansı sergileyen modeldir. %89,52 ROC AUC değeri ile güçlü bir sınıflandırma performansı sunsa da SVM ve Lojistik Regresyon modellerinin gerisinde kalmaktadır. Model %83,51 hassasiyet ve %85,72 duyarlılık sonucu elde etmiştir. Bu problemde duyarlılık sonucunun daha yüksek olması, modelin phishing sitelerini tespit etmede daha başarılı olduğu sonucunu göstermektedir. MLPClassifier modeli %84,60 F1 skoru ile hassasiyet ve duyarlılık arasında başarılı bir denge sağladığını ortaya koymaktadır. Modelin %85,72 doğru pozitif oranı (TPR) ve %93,31 özgüllük yani doğru negatif oranı (TNR), yasal ve kimlik avı sitelerinin doğru tahmin edilme oranlarını yansıtırken, %89,52 ROC AUC değeri modelin sınıflandırma yeteneğinin yüksek olduğunu ifade etmektedir. Model, gerçek pozitiflerden 26,101'ini ve gerçek negatiflerden 9,145'ini doğru şekilde tanımlamış; yanlış pozitifler 1,877 ve yanlış negatifler ise 1,916 olarak rapor edilmiştir. MLPClassifier modeli karışıklık matrisi Şekil 4'te gösterilmiştir.



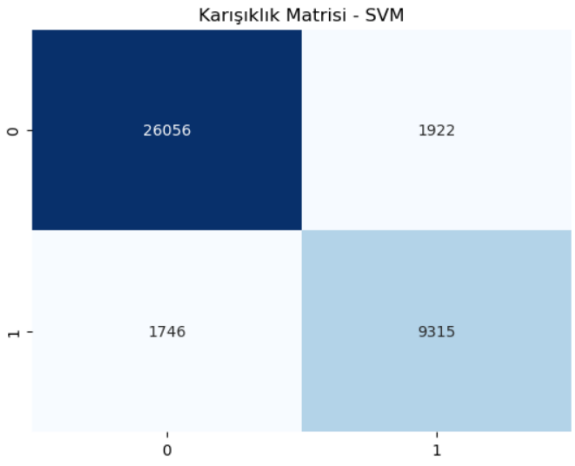
Şekil 4. MLPClassifier Modeli Karışıklık Matrisi

Lojistik Regresyon modeli, %90,50 genel doğruluk oranı ile etkileyici bir performans sergilemektedir. %95,13 ROC AUC değeri ile sınıflandırma yeteneği açısından güçlü bir performans sunmaktadır, bu da onu yüksek doğruluklu bir model yapmaktadır. Model %81,88 hassasiyet, %85,37 duyarlılık sonucu elde etmiştir. Duyarlılık sonucunun daha yüksek olması ile yasal sitelerin aksine phishing sitelerini tespit etmede daha başarılı olduğu sonucu görülmüştür. Lojistik Regresyon %83,59 F1 skoru hassasiyet ve duyarlılık arasında başarılı bir denge sağladığını ortaya koymaktadır. Modelin %85,37 doğru pozitif oranı (TPR) ve %92,53 doğru negatif oranı (TNR), yasal ve kimlik avı sitelerinin doğru tahmin edilme oranlarını yansıtırken, %95,13 ROC AUC değeri modelin sınıflandırma yeteneğinin yüksek olduğunu ifade etmektedir. Model, gerçek pozitiflerden 25,888'ini ve gerçek negatiflerden 9,443'ünü doğru şekilde tanımlamış; yanlış pozitifler 2,090 ve yanlış negatifler ise 1,618 olarak rapor edilmiştir. Lojistik Regresyon modeli karışıklık matrisi Şekil 5'te gösterilmiştir.



Şekil 5. Lojistik Regresyon Modeli Karışıklık Matrisi

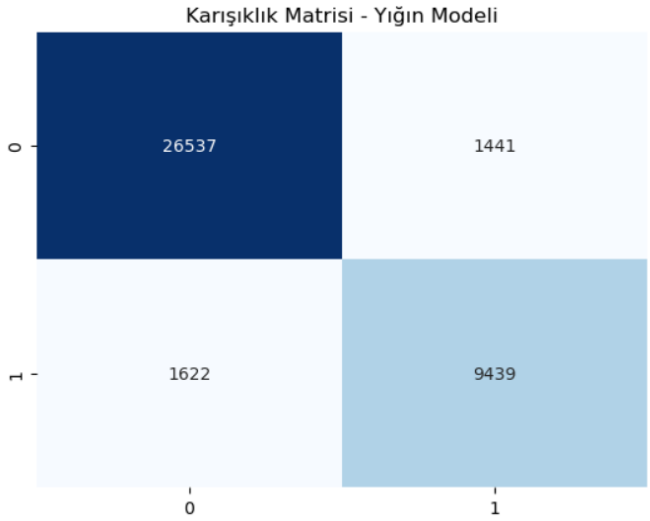
Destek Vektör Makinesi (SVM) modeli, %90,60 genel doğruluk oranı ile güçlü bir performans sergileyen SVM modeli, %95,19 ROC AUC değeri ile sınıflandırma yeteneği açısından en yüksek başarıyı elde etmektedir. Bu yüksek ROC AUC değeri, SVM'nin diğer modellere kıyasla üstün bir sınıflandırma gücüne sahip olduğunu açıkça ortaya koymaktadır. Model %82,90 hassasiyet, %84,21 duyarlılık sonucu elde etmiştir. Duyarlılık sonucunun daha yüksek olması ile yasal sitelerin aksine kimlik avı sitelerini tespit etmede daha başarılı olduğu sonucu görülmüştür. SVM %83,55 F1 skoru hassasiyet ve duyarlılık arasında başarılı bir denge sağladığını ortaya koymaktadır. Modelin %84,21 doğru pozitif oranı (TPR) ve %93,13 doğru negatif oranı (TNR), yasal ve kimlik avı sitelerinin doğru tahmin edilme oranlarını yansıtırken, %95,19 ROC AUC değeri modelin sınıflandırma yeteneğinin yüksek olduğunu ifade etmektedir. Model, gerçek pozitiflerden 26,056'sını ve gerçek negatiflerden 9,315'ini doğru şekilde tanımlamış; yanlış pozitifler 1,922 ve yanlış negatifler ise 1,746 olarak rapor edilmiştir. SVM modeli karışıklık matrisi Şekil 6'da gösterilmiştir.



Şekil 6. SVM Modeli Karışıklık Matrisi

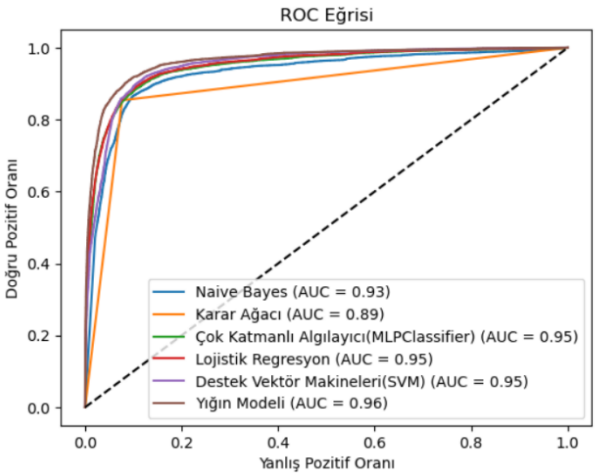
Yığın modeli, %92,15 genel doğruluk oranı ile temel sınıflayıcıların üzerinde bir performans sergilemiştir. Modelin ROC AUC değerinin %96,47 olması, sınıflar arasındaki ayrım gücünün oldukça

yüksek olduğunu ve meta öğrenicinin temel modellerin çıktılarından etkili biçimde yararlandığını göstermektedir. Yığın modeli %86,76 hassasiyet ve %85,34 duyarlılık (TPR) değerleriyle kimlik avı sitelerini tespit etmede dengeli ve güçlü bir performans ortaya koymuştur. Duyarlılık değerinin yüksek olması, modelin kimlik avı örneklerini doğru tanımlama kapasitesinin güçlü olduğunu göstermektedir. Ayrıca %86,04 F1 skoru, hassasiyet ve duyarlılık arasında başarılı bir denge kurulduğunu göstermektedir. Modelin %94,85 doğru negatif oranı (TNR), yasal siteleri doğru ayırt etme konusunda yüksek başarı sağladığını göstermektedir. Karışıklık matrisi incelendiğinde, modelin 26.537 yasal siteyi ve 9.439 kimlik avı sitesini doğru sınıflandırdığı; yanlış pozitif sayısının 1.441, yanlış negatif sayısının ise 1.622 olduğu görülmektedir. Bu sonuçlar, yığın modelinin temel sınıflayıcıların performanslarını aşarak kimlik avı tespitinde en yüksek ayrıştırma gücünü sunduğunu doğrulamaktadır. Yığın modeline ait karışıklık matrisi Şekil 7'de gösterilmiştir.



Şekil 7. Level-1 Meta Öğrenici (Yığın Modeli) Karışıklık Matrisi

ROC AUC sonuçları incelendiğinde, kimlik avı tespiti için en yüksek ayırt ediciliğe sahip modellerin SVM, Lojistik Regresyon ve MLPClassifier olduğu görülmektedir. Bu modellerin her biri, 0.95 düzeyindeki ROC AUC değerleriyle pozitif ve negatif sınıfları ayırt etme bakımından güçlü bir performans ortaya koymaktadır. Naive Bayes modeli 0.93'lük ROC AUC değeri ile başarılı olmakla birlikte, en yüksek performans gösteren modellerin bir miktar gerisinde kalmaktadır. Karar Ağacı modeli ise 0.89'luk ROC AUC değeri ile değerlendirme yapılan modeller arasında en düşük ayrım gücüne sahip sınıflandırıcıdır. Bununla birlikte, temel sınıflayıcıların çıktılarından yararlanarak karar üreten yığın modeli, 0.96'lık ROC AUC değeriyle tüm modelleri geride bırakmış ve en yüksek genel performansı sergilemiştir. Yığın yapısının farklı algoritmaların tamamlayıcı özelliklerini bir araya getirerek daha tutarlı ve genellenebilir bir sınıflandırma mekanizması sunduğu bu sonuçlarla doğrulanmaktadır. Böylece yığın modeli, tekil sınıflayıcılara kıyasla kimlik avı tespitinde daha üstün bir ayırt edicilik sağlamıştır. Modellere ait ROC eğrileri Şekil 8'de birlikte sunulmaktadır.



Şekil 8. Level-0 Temel Modeller ile Level-1 Meta Öğrenicinin ROC Eğrisi Karşılaştırması

6. SONUÇ VE TARTIŞMA

Kimlik avı saldırıları, çevrimiçi güvenliği tehdit eden en yaygın ve tehlikeli siber saldırı türlerinden biri hâline gelmiştir. Sahte web sitelerinin gerçek sitelere benzer şekilde tasarlanması, kullanıcıların kişisel ve finansal bilgilerini ele geçirme riskini artırmaktadır. Bu nedenle hem kullanıcılar hem de kurumlar açısından etkili ve güvenilir kimlik avı tespit sistemlerinin geliştirilmesi kritik önem taşımaktadır. URL ve HTML tabanlı özellikler, kimlik avı sitelerinin yapısal ve davranışsal farklılıklarını yansıtması bakımından makine öğrenimi modelleri için güçlü bir temel sunmaktadır.

Bu çalışma kapsamında, Naive Bayes, Karar Ağaçları, Destek Vektör Makineleri (SVM), Lojistik Regresyon ve MLPClassifier gibi temel sınıflandırıcıları (base learners) bir araya getiren bir yığın modeli geliştirilmiştir. Yığın modelinde temel sınıflayıcıların tahmin çıktıları meta-öğrenici olarak kullanılan Lojistik Regresyon tarafından yeniden değerlendirilmiş ve böylece her bir algoritmanın güçlü yönlerinden yararlanan bütünleşik bir yapı oluşturulmuştur. Elde edilen bulgular, yığın modelinin, bireysel sınıflayıcılara kıyasla daha yüksek genel doğruluk ve daha güçlü bir ayırt edicilik performansı (ROC AUC) sunduğunu göstermektedir.

Elde edilen sonuçlar, temel sınıflayıcılar arasında SVM, Lojistik Regresyon ve MLPClassifier modellerinin en başarılı sonuçları verdiğini ortaya koymaktadır. Bu üç model, %95'in üzerinde ROC AUC değerleri ile pozitif ve negatif sınıfları ayırt etmede üstün performans sergilemiştir. ROC AUC değerinin yüksek olması, modelin sınıfları her olasılık eşiğinde başarılı şekilde ayırabildiğini gösterdiğinden, bu sonuçlar kimlik avı tespitinde bu modellerin güçlü adaylar olduğunu doğrulamaktadır.

MLPClassifier modeli, %91,16 doğruluk oranı ve %89,52 ROC AUC değeri ile yüksek performanslı bir yapı sunmuştur. Doğrusal olmayan ilişkileri öğrenebilme kapasitesi ve geniş veri setlerinde gösterdiği esneklik, MLPClassifier'ın başarısının temel nedenlerindendir. %85,72 duyarlılık ve %83,51 hassasiyet değerleri, bu modelin hem kimlik avı hem de meşru siteleri dengeli şekilde ayırt edebildiğini

göstermektedir. Literatürde de (örn. Kalla, 2023), MLPClassifier'ın karmaşık veri setlerinde etkili sonuçlar verdiği vurgulanmaktadır.

Lojistik Regresyon modeli, %90,50 doğruluk ve %95,13 ROC AUC değeri ile yorumlanabilirlik ve performansı başarılı şekilde dengeleyen bir model olarak öne çıkmaktadır. %85,37 duyarlılık değeri, bu modelin kimlik avı sitelerini tespit etmekte etkili olduğunu göstermektedir. Shaukat (2023) tarafından yapılan çalışmalarda da Lojistik Regresyon'un kimlik avı tespitinde güvenilir bir seçenek olduğu belirtilmiştir.

SVM modeli, %90,60 doğruluk ve %95,19 ROC AUC değeri ile sınıflandırma yeteneği açısından en güçlü algoritmalardan biri olmuştur. SVM'nin yüksek boyutlu verilerdeki başarısı literatürde geniş ölçüde kabul görmektedir (Vapnik, 1995; Wang vd., 2019). %82,90 hassasiyet ve %84,21 duyarlılık sonuçları, modelin kimlik avı tespitinde etkili olduğunu ortaya koymaktadır. Ali ve Malebary (2020) tarafından yapılan çalışmalar, SVM'nin kimlik avı tespitinde güçlü bir tercih olduğunu doğrulamaktadır.

Bununla birlikte, Naive Bayes modeli, %88,94 doğruluk ve %92,77 ROC AUC değeri ile rekabetçi bir performans sunmuştur. Basit ve hızlı olması nedeniyle gerçek zamanlı tespit sistemleri için uygun bir adaydır. Yüksek boyutlu verilerdeki verimliliği, Uppalapati (2023) tarafından da vurgulanmaktadır. Ancak bağımsızlık varsayımı bazı durumlarda sınırlayıcı olabilmektedir.

Karar Ağaçları modeli, %88,94 doğruluk ve %92,77 ROC AUC değeri ile diğer modellere kıyasla daha düşük performans göstermiştir. Aşırı öğrenmeye (overfitting) eğilimli yapısı nedeniyle dikkatle budama gerektirmektedir. Bu bulgu, Al-Tamimi ve Shkoukani (2023) çalışmalarında ifade edilen değerlendirmelerle uyumludur.

Tüm bu modellerin bireysel performanslarına karşın yığın modeli, meta-öğrenme yaklaşımı sayesinde temel sınıflayıcıların güçlü yönlerini bir araya getirerek daha üstün bir genel performans elde etmiştir. %92,15 doğruluk oranı ve %96,47 ROC AUC değeri, yığın modelinin kimlik avı tespitinde diğer tüm modellere kıyasla daha yüksek sınıflandırma başarısı sunduğunu göstermektedir. Bu sonuç, literatürde Bhagat (2023) ve Kalabarige vd. (2022) gibi araştırmacıların yığın öğrenme ve çoklu model entegrasyonu üzerine yaptığı çalışmalarla da uyumludur.

Bu çalışmada elde edilen bulgular, önerilen yığın modelinin yalnızca teknik açıdan değil, Yönetim Bilişim Sistemleri (YBS) perspektifinden de önemli katkılar sunduğunu göstermektedir. Özellikle URL ve HTML tabanlı özelliklerle çalışan bu model, üçüncü taraf kara liste servislerine duyulan bağımlılığı ortadan kaldırarak kurumlar için daha güvenilir, ölçeklenebilir ve maliyet etkin bir kimlik avı tespit mekanizması sağlamaktadır. Modelin gerçek zamanlı tehdit algılama kapasitesi, kurumların bilgi güvenliği süreçlerinde erken uyarı sistemlerinin güçlendirilmesine; operasyonel risklerin azaltılmasına ve saldırı vektörlerinin daha hızlı tanımlanmasına olanak tanımaktadır. Ayrıca yığın modelinin ürettiği yorumlanabilir çıktılar, karar destek sistemlerinde yöneticilere daha doğru ve veri temelli değerlendirmeler sunmakta; bu yönüyle risk yönetimi ve kurumsal güvenlik politikalarının

geliştirilmesine katkı sağlamaktadır. Bu çerçevede, çalışma hem teknik hem de yönetsel açıdan YBS literatürüne anlamlı bir katkı sunmaktadır.

Sonuç olarak, bu çalışma hem temel makine öğrenimi algoritmalarının hem de yığın öğrenme yaklaşımının kimlik avı tespitinde etkili sonuçlar sunduğunu göstermektedir. Özellikle URL ve HTML tabanlı özelliklerin birlikte kullanılması, model başarımına önemli katkı sağlamıştır. Gelecekte, daha büyük ve güncel veri setlerinin kullanılması, derin öğrenme tabanlı yöntemlerin entegrasyonu ve gerçek zamanlı tespit mekanizmalarının geliştirilmesi, kimlik avı tespit sistemlerinin güvenilirliğini ve doğruluğunu daha da artırabilir. Marchal vd. (2014) tarafından geliştirilen PhishStorm gibi gerçek zamanlı tespit sistemleri, bu alandaki gelişmelere öncülük etmektedir ve gelecekte yapılacak çalışmalar için yol gösterici niteliktedir.

KAYNAKÇA

Aburrous, M., Hossain, M. A., Dahal, K., & Thabtah, F. (2010). Intelligent phishing detection system for e-banking using fuzzy data mining. Expert Systems with Applications, 37(12), 7913–7921. https://doi.org/10.1016/j.eswa.2010.04.044

Alazaidah, R., Al-Shaikh, A., Al-Mousa, M. R., Khafajah, H., Samara, G., Alzyoud, M., Al-Shanableh, N., & Almatarneh, S. (2024). Website phishing detection using machine learning techniques. Journal of Statistics Applications & Probability, 13(1), 119–129. https://doi.org/10.18576/jsap/130108

Aleroud, A., & Zhou, L. (2017). Phishing environments, techniques, and countermeasures: A survey. Computers & Security, 68, 160–196. https://doi.org/10.1016/j.cose.2017.04.006

Ali, W. (2017). Phishing website detection based on supervised machine learning with wrapper features selection. International Journal of Advanced Computer Science and Applications, 8(9). https://doi.org/10.14569/ijacsa.2017.080910

Ali, W., & Malebary, S. (2020). Particle swarm optimization-based feature weighting for improving intelligent phishing website detection. IEEE Access, 8, 116766–116780. https://doi.org/10.1109/access.2020.3003569

Almalki, S. (2023). A new intelligent model for phishing web sites detection. International Journal of Intelligent Digital Computing (IJIDC), 1(1), 1–14.

Al-Sarem, M., Saeed, F., Al-Mekhlafi, Z. G., Mohammed, B. A., Al-Hadhrami, T., Alshammari, M. T., Alreshidi, A., & Alshammari, T. S. (2021). An optimized stacking ensemble model for phishing websites detection. Electronics, 10(11), 1285. https://doi.org/10.3390/electronics10111285

Al-Tamimi, Y., & Shkoukani, M. (2023). Employing cluster-based class decomposition approach to detect phishing websites using machine learning classifiers. International Journal of Data and Network Science, 7(1), 313–328. https://doi.org/10.5267/j.ijdns.2022.10.002

Anti-Phishing Working Group. (2023). Phishing activity trends report: 2023. https://apwg.org/trendsreports/

Ariyadasa, S. (2020). Detecting phishing attacks using a combined model of LSTM and CNN. International Journal of Advanced and Applied Sciences, 7(7), 56–67. https://doi.org/10.21833/ijaas.2020.07.007

Basnet, R., Sung, A., & Liu, Q. (2011). Rule-based phishing attack detection. Proceedings of the International Conference on Security and Management (SAM'11), 1–8.

Bhagat, P. (2023). Feature classification and extreme learning machine-based detection of phishing websites. International Journal on Recent and Innovation Trends in Computing and Communication, 11(8s), 132–136. https://doi.org/10.17762/ijritcc.v11i8s.7182

Hassan, D. (2015). On determining the most effective subset of features for detecting phishing websites. International Journal of Computer Applications, 122(20), 1–7. https://doi.org/10.5120/21813-5191

He, D., Lv, X., Zhu, S., Chan, S., & Choo, K. K. R. (2024). A method for detecting phishing websites based on Tiny-BERT stacking. IEEE Internet of Things Journal, 11(2), 2236–2243. https://doi.org/10.1109/JIOT.2023.3292171

Igwilo, C. M., & Odumuyiwa, V. (2022). Comparative analysis of ensemble learning and non-ensemble machine learning algorithms for phishing URL detection. FUOYE Journal of Engineering and Technology (FUOYEJET), 7(3), 305–312. https://doi.org/10.46792/fuoyejet.v7i3.807

Kalabarige, L. R., Rao, R., Abraham, A., & Gabralla, L. (2022). Multilayer stacked ensemble learning model to detect phishing websites. IEEE Access, 10, 79543–79552. https://doi.org/10.1109/ACCESS.2022.3194672

Kalla, D. (2023). Phishing website URLs detection using NLP and machine learning techniques. Journal on Artificial Intelligence, 5(0), 145–162. https://doi.org/10.32604/jai.2023.043366

Marchal, S., François, J., State, R., & Engel, T. (2014). PhishStorm: Detecting phishing with streaming analytics. IEEE Transactions on Network and Service Management, 11(4), 458–471. https://doi.org/10.1109/TNSM.2014.2377295

Ogonji, D. (2023). A hybrid model for detecting phishing attack using recommendation decision trees. ITM Web of Conferences, 57, 01018. https://doi.org/10.1051/itmconf/20235701018

Purwanto, R., Pal, A., Blair, A., & Jha, S. (2022). PhishSim: Aiding phishing website detection with a feature-free tool. IEEE Transactions on Information Forensics and Security, 17, 1497–1512. https://doi.org/10.1109/tifs.2022.3164212

Rao, R. S., & Pais, A. R. (2019). Detection of phishing websites using an efficient feature-based machine learning framework. Neural Computing and Applications, 31, 3851–3873. https://doi.org/10.1007/s00521-017-3305-0

Sahingoz, O. K., Buber, E., Demir, O., & Diri, B. (2019). Machine learning based phishing detection from URLs. Expert Systems with Applications, 117, 345–357. https://doi.org/10.1016/j.eswa.2018.09.029

Shabudin, S., Samsiah, N., Akram, K., & Aliff, M. (2020). Feature selection for phishing website classification. International Journal of Advanced Computer Science and Applications, 11(4). https://doi.org/10.14569/ijacsa.2020.0110477

Shaukat, M. (2023). A hybrid approach for alluring ads phishing attack detection using machine learning. Sensors, 23(19), 8070. https://doi.org/10.3390/s23198070

Shinde, A., Pandey, A., Pawar, R., & Gangule, V. (2015). Clustering and Bayesian approach-based model for detection of phishing. International Journal of Computer Applications, 118(24), 30–33. https://doi.org/10.5120/20958-3385

Tang, L., & Mahmoud, Q. H. (2021). A survey of machine learning-based solutions for phishing website detection. Machine Learning and Knowledge Extraction, 3(3), 672–694. https://doi.org/10.3390/make3030034

Uppalapati, P. (2023). A machine learning approach to identifying phishing websites: A comparative study of classification models and ensemble learning techniques. EAI Endorsed Transactions on Scalable Information Systems. https://doi.org/10.4108/eetsis.vi.3300

Vapnik, V. (1995). The nature of statistical learning theory. Springer. https://doi.org/10.1007/978-1-4757-2440-0

Vapnik, V. N., & Chervonenkis, A. Y. (1974). Theory of pattern recognition. Nauka.

Wang, W., Feng, Z., Luo, X., & Zhang, S. (2019). PDRCNN: Precise phishing detection with recurrent convolutional neural networks. Security and Communication Networks, 2019, 1–15. https://doi.org/10.1155/2019/2595794

Wolpert, D. H. (1992). Stacked generalization. Neural Networks, 5, 241–259. https://doi.org/10.1016/S0893-6080(05)80023-1

Yang, P., Zhao, G., & Zeng, P. (2019). Phishing website detection based on multidimensional features driven by deep learning. IEEE Access, 7, 15196–15209. https://doi.org/10.1109/access.2019.2892066

Zamir, A., Khan, H., Iqbal, T., Yousaf, N., Aslam, F., Anjum, A., & Hamdani, M. (2020). Phishing website detection using diverse machine learning algorithms. The Electronic Library, 38(3), 545–563. https://doi.org/10.1108/EL-05-2019-0118

Zhu, E., Chen, Y., Ye, C., Li, X., & Feng, L. (2019). OFS-NN: An effective phishing websites detection model based on optimal feature selection and neural network. IEEE Access, 7, 73271–73284. https://doi.org/10.1109/access.2019.2920655

Zou, L., & Nan, T. (2019). A phishing webpage detection method based on stacked autoencoder and correlation coefficients. Journal of Computing and Information Technology, 27(2), 41–54. https://doi.org/10.20532/cit.2019.1004702

EXTENDED ABSTRACT

DETECTION OF PHISHING WEBSITES USING A STACKING MODEL BASED ON URL AND HTML FEATURES

Phishing attacks have become one of the most common and dangerous cyber threats targeting both individuals and organizations in recent years. Attackers design fake pages that closely resemble real websites, aiming to steal users' personal and financial information, making these attacks extremely difficult to detect. The rise in internet use, the proliferation of online services, and the ubiquity of digital transactions in daily life have further exacerbated the impact of phishing attacks. In particular, the constant development of new techniques by attackers limits the effectiveness of traditional defense methods, highlighting the need for dynamic, learning-capable, and real-time models. This study aims to develop a stack model capable of detecting phishing sites with high accuracy by combining URL- and HTML-based features.

The proposed model is based on a multi-layered ensemble learning approach that aims to achieve stronger classification performance by combining the complementary aspects of different machine learning algorithms. The Level-0 (base learner) layer of the model includes Naive Bayes, Decision Tree, Logistic Regression, Support Vector Machines (SVM), and Multilayer Perceptron (MLPClassifier). These base models provide various prediction signals to the meta-learner thanks to their uniquely capturing structures of different patterns in the dataset. Logistic Regression, used in the Level-1 (meta learner) layer, optimizes these signals and generates the final classification decision. This results in a more stable and reliable system that overcomes the limitations of a single model, and increases the overall accuracy of the model.

The phish.csv dataset used in the study contains 57 attributes from 130,127 web pages. These attributes include indicators such as URL length, token count, IP address usage, suspicious symbols, subdomain level, and sensitive keyword content; and statistics on form tags, redirects, link rates, hidden content,

and DOM structure obtained from the HTML source code. Of the dataset, 93,241 are legitimate and 36,886 are phishing samples, and this distribution requires caution regarding class imbalance. Because there were no missing values, no cleaning was required; all variables were scaled using the min-max method, and the training-test separation was performed at 80%-20%. Additionally, five-fold cross-validation was applied to strengthen the generalizability of the models. This process reduced the model's data dependency and contributed to a more consistent performance.

When comparing the performance of the baseline (Level-0) classifiers, it was observed that SVM provided the highest discrimination with a ROC AUC of 95.19%. MLPClassifier achieved the best accuracy with 91.16% accuracy, while Logistic Regression demonstrated a strong and consistent discrimination ability with a ROC AUC of 95.13%. Naive Bayes, with 88.94% accuracy and 92.77% ROC AUC, demonstrates that it is an effective alternative, especially in scenarios requiring low-cost and fast prediction. The Decision Tree model, despite producing 90.62% accuracy, performed lower than the other models in terms of discrimination power. These findings demonstrate that while the base classifiers are strong in certain respects, they cannot provide optimal performance on their own.

When the performance of the stack model is evaluated, it is seen that the meta-learner outperformed all the base classifiers. The model demonstrated the highest performance in the study with 92.15% accuracy and 96.47% ROC AUC. Furthermore, 86.76% sensitivity, 85.34% sensitivity, and 86.04% F1 score confirm the model's balanced and stable classification ability. The stack model reduces false positive and false negative rates, improving user safety and minimizing false alarms in cybersecurity systems. ROC curves also demonstrate that the stack model offers higher discrimination power compared to single classifiers.

This study offers significant contributions not only in terms of technical performance but also in the context of Management Information Systems (MIS). The model allows organizations to develop a more sustainable, reliable, and cost-effective security architecture by reducing their reliance on third-party blacklisting services. Furthermore, the model's interpretable structure provides decision-makers with clearer insights into risk assessment processes, particularly contributing to operational security units' more accurate threat assessment. Its lightweight, real-time operation allows for easy integration into both individual user applications and corporate security infrastructures. Furthermore, the stack model's modular structure allows for easy future expansion with different data sources or additional feature sets, increasing the model's long-term potential for use.

In conclusion, the stack learning approach, which combines URL and HTML-based features, offers an effective method for phishing detection, providing high accuracy and strong discrimination capabilities. The results demonstrate that ensemble learning models deliver superior and more reliable performance compared to single machine learning algorithms. Furthermore, the model's real-time, low-cost, and extensible structure offers significant advantages for both academic and corporate applications. Future

improvements include the use of larger, more up-to-date datasets, the integration of deep learning architectures (CNN, LSTM, BERT, etc.), the addition of visual similarity analyses, and the development of real-time detection systems. Furthermore, the model's flexible structure allows for the integration of different feature sets, behavioral data, or visual similarity metrics, providing significant scope for future research. When implemented at an enterprise scale, it can form a strong foundation for real-time threat detection and early warning systems. In this respect, the developed model not only provides an academic contribution but also has high potential for applicability in practical security solutions.