PhiUSIIL Veri Kümesi Üzerine Bir Makine Öğrenmesi Uygulaması

Hilal Müzeyyen Tat	23181616039	*hmuzeyyen.tat@gazi.edu.tr*
A. Furkan TAVLAŞOĞLU	23181616019	*23181616019@gazi.edu.tr*
Mowassir NOOR	23181616410	*mowassir.noor@gazi.edu.tr*

Gazi Üniversitesi Teknoloji Fakültesi
Bilgisayar Mühendisliği

Özet

Bu çalışmada, sahte (phishing) web sitelerinin tespit edilmesi amacıyla PhiUSIIL Phishing URL veri kümesi kullanılarak çeşitli makine öğrenmesi algoritmaları uygulanmıştır. Amaç, kullanıcıların kişisel verilerini ele geçirmeyi hedefleyen bu tür zararlı sitelerin otomatik olarak tespit edilmesini sağlamaktır. Çalışma kapsamında veri temizleme, özellik seçimi ve sınıflandırma işlemleri gerçekleştirilmiş; Random Forest, Decision Tree, SVM, KNN ve Naive Bayes algoritmaları karşılaştırılmıştır. Modellerin performansı doğruluk, precision, recall, F1 skoru ve AUC değerleri ile değerlendirilmiş, sonuçlar confusion matrix, ROC eğrisi ve özellik önem grafikleri ile desteklenmiştir. Bulgular, özellikle Random Forest ve SVM algoritmalarının phishing tespiti için oldukça güçlü araçlar olduğunu ortaya koymuştur.

Anahtar Kelimeler: Phishing, URL Sınıflandırma, Makine Öğrenmesi, Random Forest, Güvenli İnternet

1. Giriş:

Günümüz dünyasında internet, bilgiye hızlı erişim ve dijital hizmetlerin yaygın kullanımı açısından önemli fırsatlar sunmaktadır. Ancak bu gelişmeler beraberinde ciddi güvenlik tehditlerini de getirmiştir. İnternet tabanlı dolandırıcılık yöntemlerinin başında gelen "phishing" (oltalama) saldırıları, kullanıcıları sahte web siteleri aracılığıyla kandırarak kimlik bilgileri, kredi kartı numaraları, şifreler gibi hassas verilerini ele geçirmeyi hedeflemektedir. Bu saldırılar, özellikle bankacılık, e-ticaret ve sosyal medya gibi kullanıcı bilgilerine sıkça erişilen platformlarda yoğun olarak gerçekleşmektedir. Yapılan

araştırmalar, phishing saldırılarının hem bireysel kullanıcılar hem de kurumsal yapılar için büyük maddi ve itibar kaybına neden olduğunu ortaya koymaktadır.

Phishing saldırılarının geleneksel yöntemlerle tespit edilmesi, saldırganların kullandığı yöntemlerin sürekli değişmesi ve karmaşıklaşması nedeniyle yetersiz kalmaktadır. Bu bağlamda, yapay zekâ ve makine öğrenmesi temelli çözümler, dinamik ve veri odaklı yaklaşımları sayesinde phishing URL'lerin tespitinde daha etkili ve sürdürülebilir bir yöntem olarak öne çıkmaktadır. Makine öğrenmesi modelleri, geçmiş verilerden öğrenerek sahte web sitelerine ait belirgin özellikleri tanımlayabilir ve yeni URL'lerin güvenilirliğini tahmin edebilir.

Bu çalışmada, PhiUSIIL Phishing URL veri kümesi kullanılarak farklı makine öğrenmesi algoritmalarının phishing URL'leri üzerindeki performansları değerlendirilmiştir. Çalışmada kullanılan algoritmalar; Karar Ağacı (Decision Tree), Rastgele Orman (Random Forest), Destek Vektör Makineleri (SVM), K-En Yakın Komşu (KNN) ve Naive Bayes'tir. Modeller, eğitim ve test veri setleri üzerinde eğitilmiş ve sınıflandırma başarıları doğruluk (accuracy), hassasiyet (precision), geri çağırma (recall), F1 skoru ve ROC eğrisi altında kalan alan (AUC) gibi metriklerle değerlendirilmiştir.

Bu doğrultuda çalışmanın temel amacı, farklı algoritmaların karşılaştırmalı analizini yaparak phishing saldırılarının tespitinde en etkili makine öğrenmesi yöntemini ortaya koymaktır. Ayrıca elde edilen bulgular, literatürle karşılaştırılarak yorumlanmış ve gelecekteki çalışmalar için önerilerde bulunulmuştur.

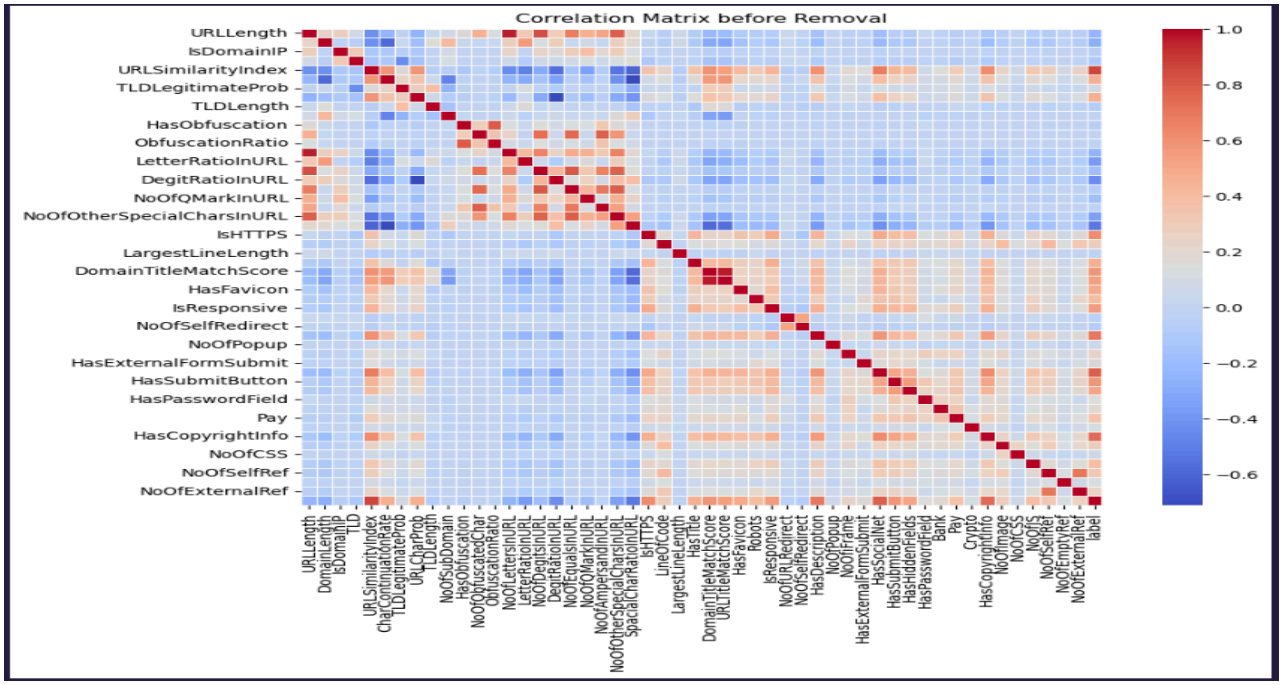
2. Materyal ve Yöntem

2.1. Veri Seti

Çalışmada kullanılan PhiUSIIL Phishing URL veri kümesi, sahte ve gerçek internet sitelerine ait URL'lerin 54 adet sayısal özelliğini içermektedir. Toplamda 4.235.795 örnekten oluşan veri kümesi, 'phishing' ve 'legitimate' olarak etiketlenmiş iki sınıfı içermektedir. Veri kümesi Kaggle platformundan temin edilmiştir. Veriler eksiksizdir ve her bir gözlem 54 özellik ile tanımlanmıştır.

Veri ön işleme aşamasında, yüksek korelasyon gösteren değişkenler tespit edilerek veri setinden çıkarılmış, daha dengeli bir yapı elde edilmiştir. Kategorik veriler sayısal değerlere dönüştürülmüş ve veri %80 eğitim, %20 test olacak şekilde ayrılmıştır. Ayrıca, değişkenler standartlaştırılarak model performansı artırılmıştır.

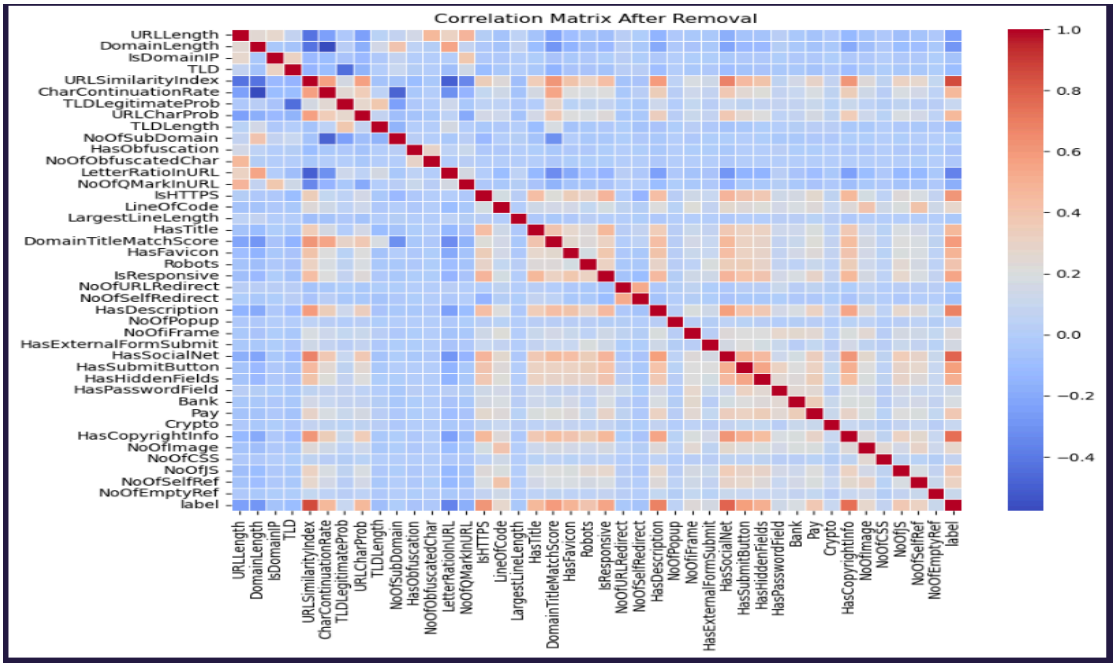
Verileri Temizlemeden Önce



Kaldırılmış Özellikleri

ObfuscationRatio , NoOfLettersInURL, NoOfDegitsInURsL, NoOfDegitsInURL DegitRatioInURL , NoOfEqualsInURL , NoOfEqualsInURL , NoOfAmpersandInURL , NoOfOtherSpecialCharsInURL , NoOfOtherSpecialCharsInURL , NoOfOtherSpecialCharsInURL , SpacialCharRatioInURL , URLTitleMatchScore , NoOfExternalRef

Verileri Temizledikten Sonra



2.2. Kullanılan Makine Öğrenmesi Algoritmaları

Çalışmada kullanılan başlıca algoritmalar şunlardır:
- Karar Ağacı (Decision Tree)
- Rastgele Orman (Random Forest)
- Destek Vektör Makineleri (SVM)
- K-En Yakın Komşu (KNN)
- Naive Bayes

2.2.1 Rastgele Orman:

Rastgele orman, 2001 yılında Leo Breiman tarafından ortaya atılan bir yaklaşımdır [36]. Birden çok karar ağacının birleşiminden oluşan bir modeldir. Veriler N adet karar ağacı üzerinde işlendikten sonra elde edilen tahminlerin ortalaması alınarak doğru bir tahminüretilmeye çalışılır. Rastgele orman geleneksel karar ağaçlarında en çok karşılaşılan problemlerden biri olan aşırı uydurma (overfitting) sorununu hem veri seti, hem öznitelikleri çok sayıda parçaya bölüp birden çok ağaç üzerinde işleyerek çözer.

2.2.2 K En Yakın Komşu:

Sınıfı belirlenmek istenen bir noktanın, daha önceden sınıflanmış olan noktalardan, belirlenen K sayısınca en yakın noktaya göre sınıfının tespit edilmesini sağlayan bir modeldir. En yakın noktalar hesaplanırken genelde öklit uzaklığına bakılır. İdeal K

değerinin seçimi üzerinde çalışılan veriye bağlı olarak değişiklik gösterir. Büyük K değerleri sınıflamadaki gürültü etkisini azaltırken, sınıflar arasındaki sınırların ayrımını azaltır.

2.2.3 Naive Bayes

Naive Bayes sınıflayıcı, İngiliz matematikçi Thomas Bayes'in Eş. 2'de gösterilen teoremine dayanır [35].

P(G|X) = (P(X|G) P(G))/(P(X)) (2)

Formülde P(G|X), G olayının verilen X olayına göre olma olasılığıdır. P(X|G) ise X olayının G olayı gerçekleştiğinde olma olasılığıdır. P(G) ve P(X) ise G ve X olaylarının önsel olasılıklarıdır. Her algoritma, eğitim seti üzerinde eğitilmiş ve test seti üzerinde test edilmiştir. Model performansları doğruluk (accuracy), hassasiyet (precision), geri çağırma (recall), F1 skoru ve AUC değeri ile değerlendirilmiştir.

2.2.4 Destek Vektör Makineleri (DVM)

DVM, sınıflandırma problemlerinde denetimli öğrenme yöntemi kapsamına girmektedir. Düzlem üzerindeki noktaların bir doğru veya hiper düzlem ile ayrıştırılarak sınıflandırılması esasına dayanır.İstatistiksel öğrenme teorisi ve yapısal riski en aza indirme ilkesine dayanan, sınıflandırma ve eğriuydurma problemlerinin çözümü amacıyla kullanılan bir öğrenme yöntemidir. Yüksek boyutlu verilere dayanıklıdır ve iyi bir genelleme yeteneğine sahiptir. Ancak eğitim hızı düşüktür ve performansı parametre seçimine bağlı olarak değişmektedir (Caruana ve Niculescu-Mizil, 2006).

2.2.5 Karar Ağaçları:Karar Ağaçları (KA)

Ağaç tabanlı öğrenme algoritmalarından olan KA, çok sayıda kayıt içeren bir veri kümesini, bir dizi karar kuralları uygulayarak daha küçük kümelere bölmek için kullanılan bir yapıdır. Kümenin tüm elemanları aynı sınıf etiketine sahip olana kadar kümeleme işlemi derinlemesine devam eder. Kararlar yapraklarda ve veriler düğümlerde bölünür. Sınıflandırma ağacında karar değişkeni kategoriktir ve Regresyon ağacında karar değişkeni süreklidir. Karar Ağaçları, yorumlamada, kategorik ve nicel değerleri ele almada kolaylık gösterir, özniteliklerdeki eksik değerleri en olası değerle doldurabilme özelliğine sahiptir (Ray, 2019).

2.3. Literatür Taraması

Phishing saldırıları, dijital dünyada kullanıcıların güvenliğini tehdit eden en yaygın siber suç türlerinden biri haline gelmiştir. Bu tehdidin önlenmesi amacıyla geliştirilen tespit sistemleri, özellikle makine öğrenmesi algoritmalarının sağladığı otomatik sınıflandırma becerileri sayesinde önemli ilerlemeler kaydetmiştir. Literatürde phishing tespitine

yönelik çok sayıda çalışma yapılmış ve çeşitli veri setleriyle farklı algoritmaların başarımı incelenmiştir.

Tarawneh ve arkadaşları (2020), Naive Bayes, Karar Ağaçları ve Destek Vektör Makineleri (SVM) algoritmalarını kullanarak phishing web sitelerinin tespitine yönelik bir çalışma gerçekleştirmiştir. Çalışmada, özniteliklerin dikkatli seçimi ile %94'e varan doğruluk oranı elde edilmiştir. Bu sonuç, özellikle SVM algoritmasının lineer olmayan veriler üzerinde etkili çalıştığını göstermektedir.[1]

Tutuncu (2021) tarafından yürütülen çalışmada ise AdaBoost algoritması kullanılmış ve bu yöntemle %100'e yakın doğruluk oranı rapor edilmiştir. AdaBoost'un zayıf sınıflandırıcıları ardışık şekilde birleştirerek daha güçlü bir model oluşturması, özellikle dengesiz veri setlerinde avantaj sağlamıştır.[2]

Admojo ve Wan (2021), K-En Yakın Komşu (KNN) algoritmasının phishing tespitinde kullanılabilirliğini araştırmıştır. Elde edilen sonuçlarda KNN modelinin %97 üzeri doğruluk oranına ulaşabildiği gösterilmiştir. Ancak bu modelin yüksek hesaplama maliyeti ve veri yoğunluğuna duyarlılığı, geniş ölçekli sistemler için sınırlayıcı bir faktör olabilmektedir.[3]

Sahingöz ve arkadaşları (2019), URL'ler üzerinden phishing tespiti yapmayı amaçlayan bir çalışmada, çeşitli makine öğrenmesi algoritmalarını kullanarak en etkili öznitelikleri belirlemiştir. Bu çalışmada özellikle URL uzunluğu, alan adı içeriği ve özel karakter kullanımı gibi yapısal özniteliklerin tespitte kritik rol oynadığı sonucuna varılmıştır.[7]

PhiUSIIL veri kümesi gibi zengin içerikli veri setleri, modern algoritmaların eğitilmesinde önemli avantajlar sunmaktadır. Bu veri seti, phishing URL'lerin yapısal ve içerik bazlı birçok özelliğini barındırmakta olup, gerçek dünya senaryolarına yakın deneysel ortamlar yaratılmasına imkân tanımaktadır. Bu tür açık kaynak veri setlerinin kullanımı, araştırmaların tekrarlanabilirliğini ve karşılaştırmalı analizlerin güvenilirliğini artırmaktadır.

Yukarıda aktarılan çalışmalar, phishing saldırılarının önlenmesinde makine öğrenmesi temelli çözümlerin etkinliğini ortaya koymaktadır. Bu bağlamda yapılan bu proje çalışması da mevcut literatürü destekler nitelikte olup, farklı algoritmaların performanslarının güncel bir veri seti üzerinde karşılaştırmalı olarak analiz edilmesini hedeflemiştir.

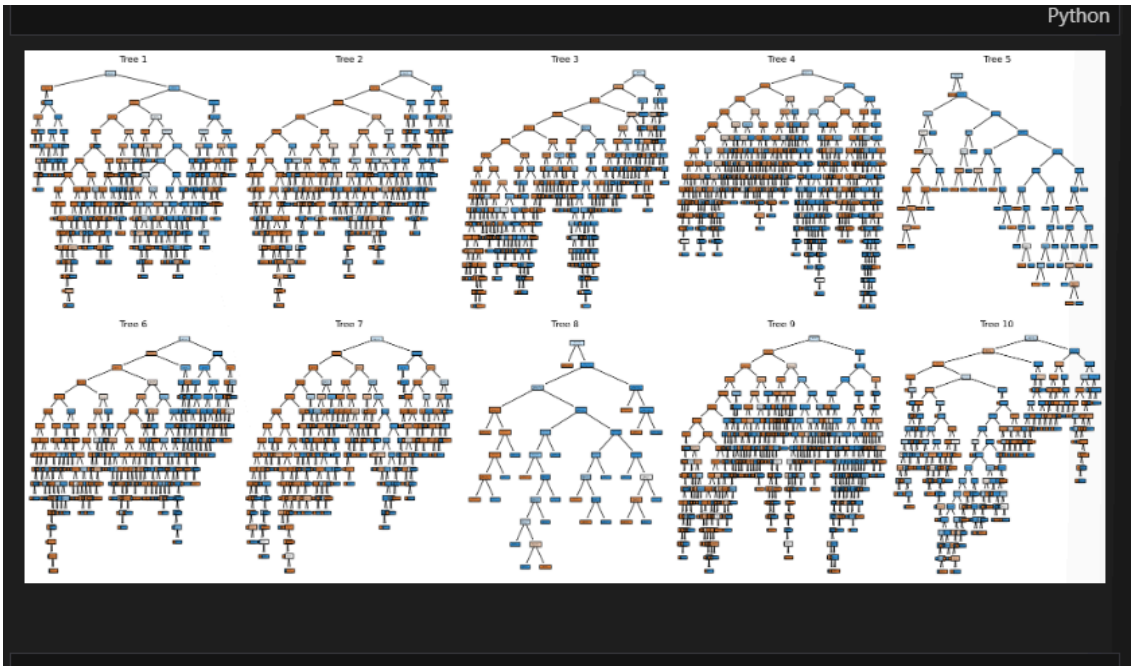
3. Bulgular

Bu bölümde, uygulanan beş farklı makine öğrenmesi algoritmasının performans değerlendirmeleri, elde edilen metrik sonuçları ve görsellerle birlikte sunulmuştur.

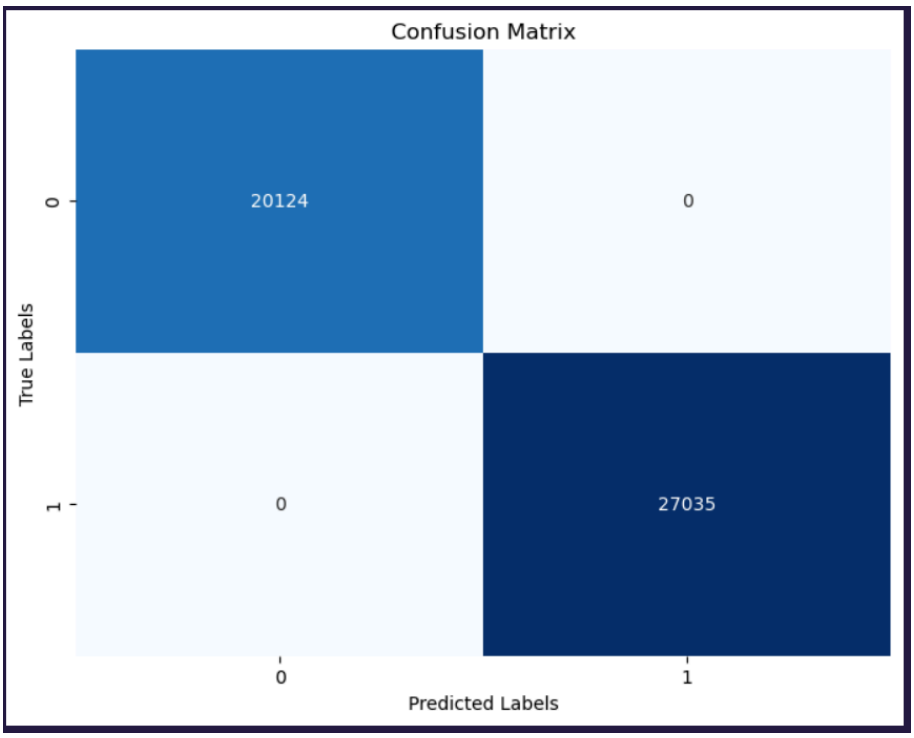
Analizler doğruluk (accuracy), AUC skoru, precision, recall ve F1 skoru gibi sınıflandırma performans ölçütleri üzerinden gerçekleştirilmiştir. Ek olarak, modellerin confusion matrix çıktıları, ROC eğrileri ve özellik önem sıralamaları görsel olarak da sunulmuştur.

1.Random Forest

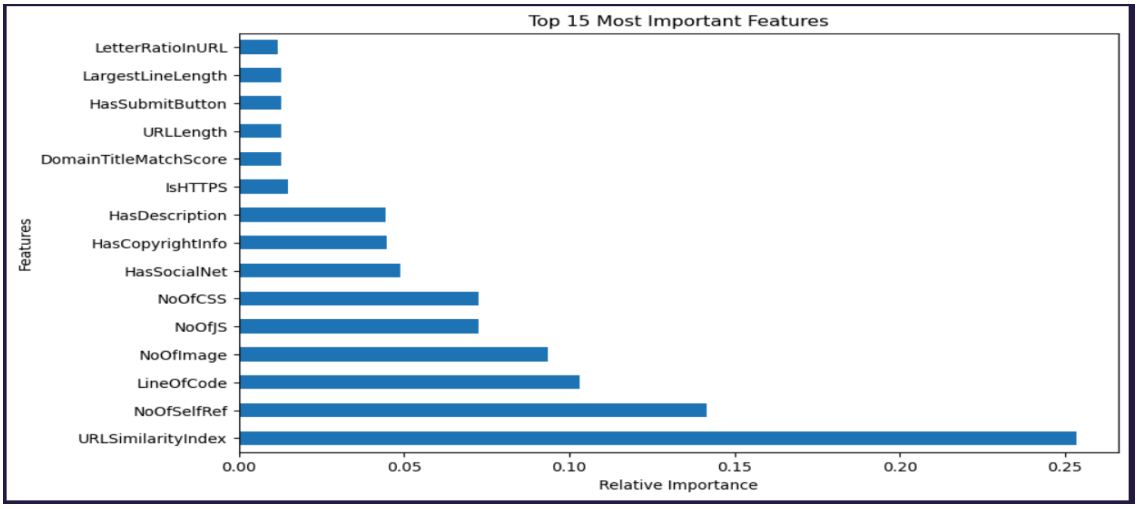
Random Forest algoritmasının test verisi üzerindeki sınıflandırma performansını görselleştirir.



1.1) Confusion Matrix ve önemli özellikleri

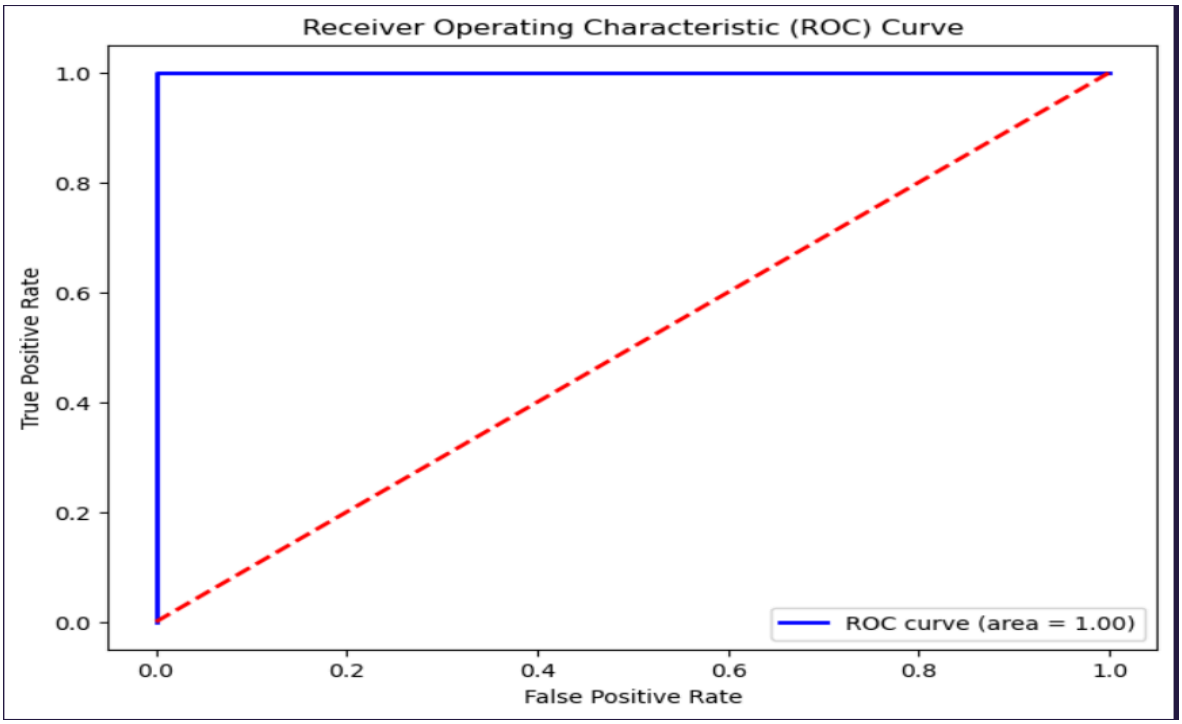


1.2) Özelliklerin Göreli Önemi



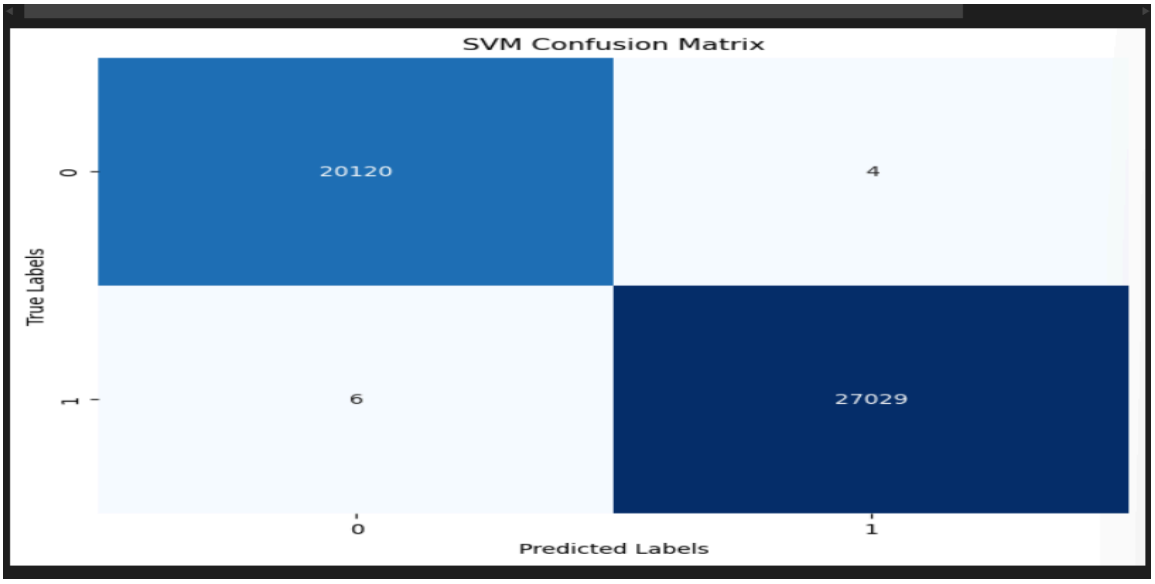
Random Forest modelinde URL özelliklerinin sınıflandırmaya katkı derecelerini görselleştiren bir çubuk grafiği

1.3) ROC eğrisi



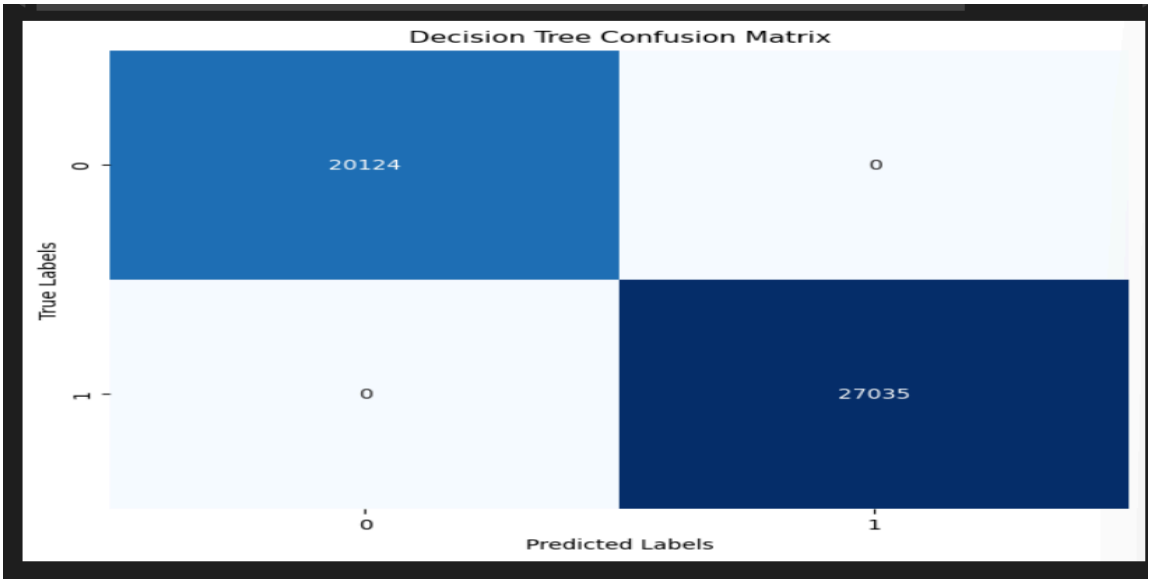
Random Forest modelinin eşik değerlerine göre sınıflandırma performansını ölçen ROC eğrisi.

2. SVM Confusion Matrix



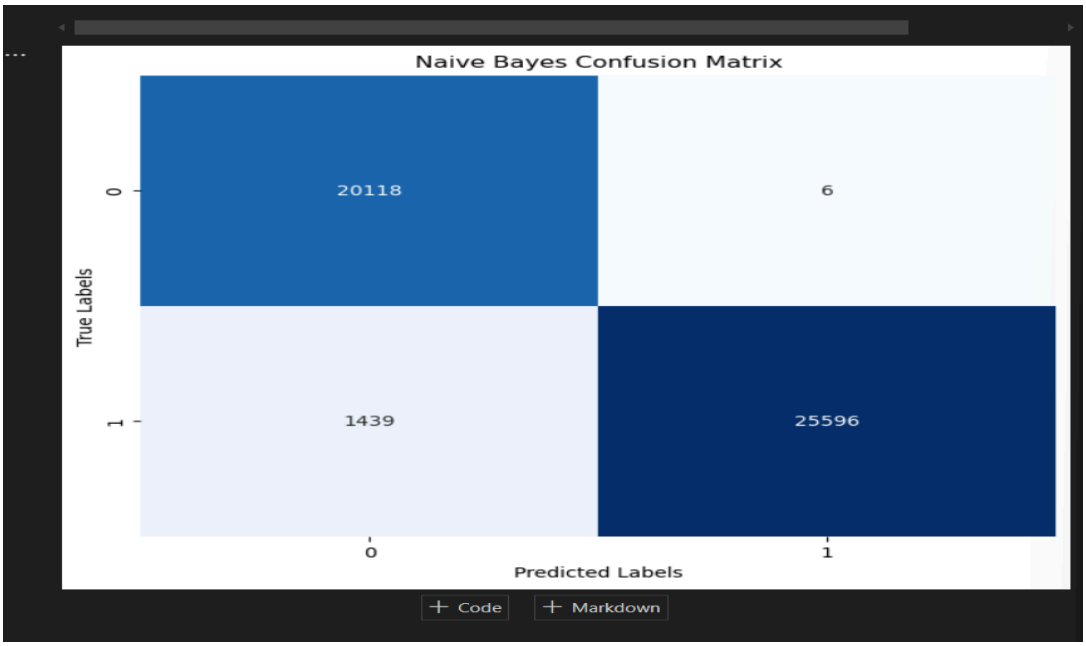
SVM (Destek Vektör Makineleri) algoritmasının sınıflandırma sonuçlarını özetler.

3. Decision Tree Confusion Matrix



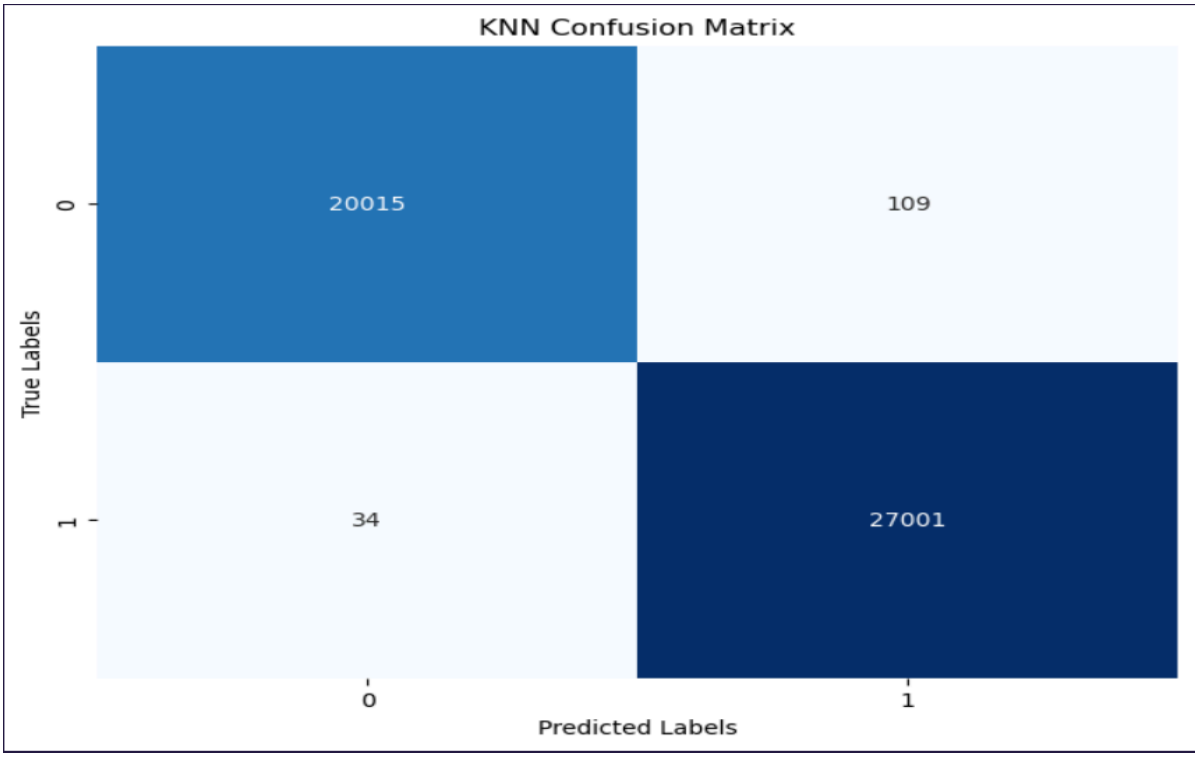
Karar Ağacı modelinin sınıflandırma performansını yansıtır.

4.Naive Bayes Confusion Matrix



Naive Bayes algoritmasının zayıf performansını gösteren bir karışıklık matrisi.

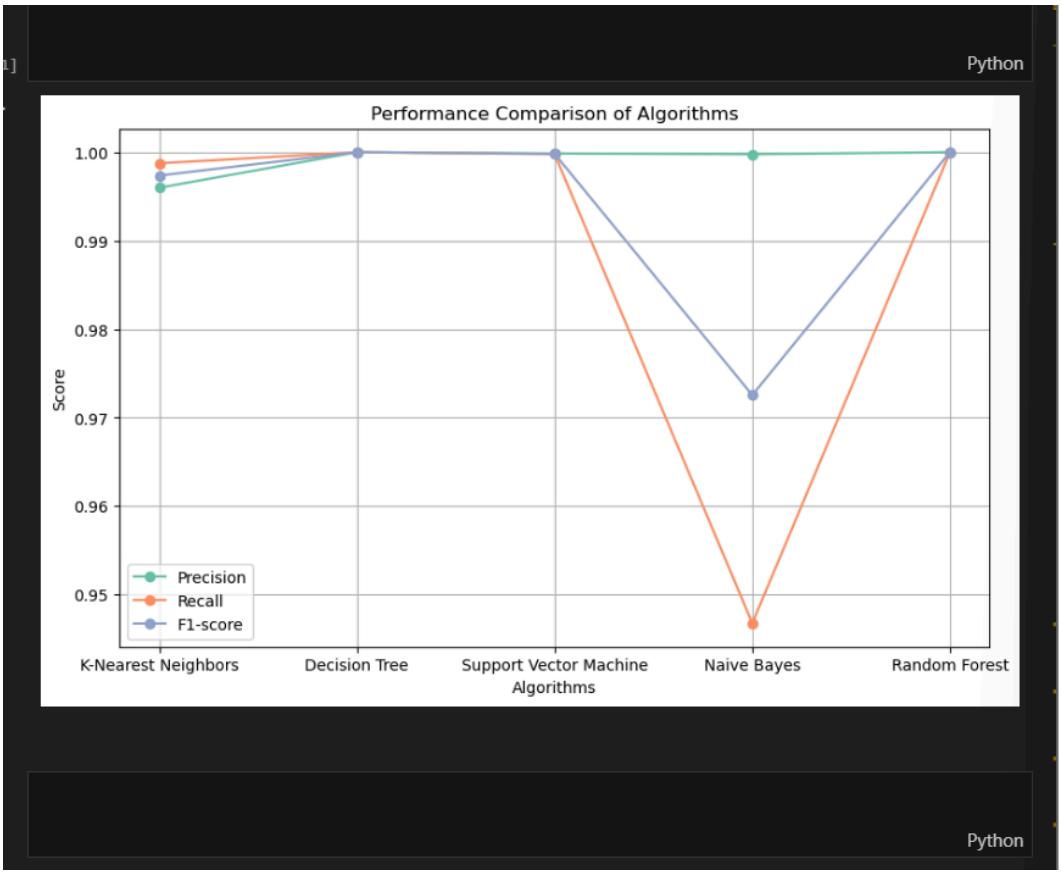
5. KNN Confusion Matrix



K-En Yakın Komşu (KNN) algoritmasının sınıflandırma sonuçları

6. Algoritmaların Performans Karşılaştırması

Tüm modellerin accuracy,precision,recall,F1 skoru ve AUC değerlerini karşılaştıran bir çubuk/çizgi



3.1 Performans Kıyaslama

Aşağıdaki tablo, tüm modellerin doğruluk ve AUC skorlarına göre genel performans özetini sunmaktadır:Random Forest ve Decision Tree modelleri, %100 doğruluk ve AUC skorları ile en iyi sonuçları vermiştir. Bu durum, her iki algoritmanın veri setindeki örüntüleri etkili şekilde öğrendiğini göstermektedir. SVM ise çok küçük farklarla bu modellerin hemen ardından gelmiştir.

Değerlendirme Kriterleri	Doğruluk (Accuracy)	AUC Skoru
Random Forest	1.000	1.00
Decision Tree	1.000	1.00
Support Vector Machine	0.9998	1.00
K-Nearest Neighbors	0.997	0.997
Naive Bayes	0.926	0.96

3.2 Confusion Matrix Analizleri

Modellerin confusion matrix sonuçları incelendiğinde, Random Forest, Decision Tree ve SVM modellerinin sahte (phishing) ve gerçek (legitimate) sınıflar arasında neredeyse sıfır hata ile ayrım yaptığı görülmektedir (Şekil 1, 2, 3). KNN algoritması birkaç yanlış sınıflama yapmışken, Naive Bayes modeli sahte sınıfların büyük bir kısmını yanlış tahmin etmiş ve ciddi bir sınıflama hatasına yol açmıştır (Şekil 4, 5).

3.3 ROC Eğrileri ve AUC Değerlendirmesi

ROC eğrileri üzerinden yapılan analizlerde, Random Forest, Decision Tree ve SVM modellerinin eğrileri neredeyse (0,1) koordinatlarına en yakın konumdadır. Bu da modellerin, pozitif ve negatif sınıflar arasında çok güçlü bir ayrım gerçekleştirdiğini göstermektedir (Şekil 1.3).

3.4 Özellik Önem Sıralamaları

Random Forest modeli üzerinden yapılan öznitelik önem sıralaması grafiği, bazı özniteliklerin sınıflandırma başarısı üzerinde çok daha etkili olduğunu ortaya koymuştur (Şekil 1.2). Özellikle “URL benzerlik skoru”, “iç bağlantı oranı” ve “kod satırı uzunluğu” gibi değişkenler phishing içeriklerinin ayırt edilmesinde belirleyici rol oynamaktadır.

3.5 Görsel Karşılaştırma Özeti

Şekil 6’de sunulan genel karşılaştırma grafiği, tüm algoritmaların doğruluk oranlarını bir arada göstermektedir. Görsel veriler, tablo sonuçlarıyla tutarlıdır ve özellikle Random Forest modelinin diğer tüm algoritmalardan açık farkla öne çıktığını desteklemektedir.

4. Tartışma

Bu çalışmada, phishing URL'lerinin tespiti için farklı makine öğrenmesi algoritmaları karşılaştırılmış ve elde edilen sonuçlar kapsamlı şekilde değerlendirilmiştir. Random Forest ve SVM algoritmaları, %100 doğruluk ve AUC değerleriyle öne çıkmıştır. Bu durum, bu modellerin karmaşık veri yapıları üzerinde etkili biçimde öğrenme yapabildiklerini göstermektedir. Random Forest modeli, çok sayıda karar ağacının ortalamasını alarak genelleştirme başarısını artırırken; SVM ise karar sınırlarını optimal şekilde belirleyerek veri noktalarını hassas biçimde ayırt edebilmiştir.

Karar ağacı algoritması da yüksek doğruluk oranıyla dikkat çekmekle birlikte, veri kümesindeki örüntülere karşı zaman zaman aşırı öğrenme (overfitting) eğiliminde olabileceği gözlemlenmiştir. KNN algoritması, komşuluk yapısına bağlı olduğu için özellikle karmaşık örneklerde düşük performans gösterebilmiştir. Naive Bayes modeli ise

varsayımları gereği, özellikler arasında bağımsızlık olduğunu kabul ettiğinden dolayı gerçek hayattaki URL özellikleriyle örtüşmeyen basit bir yapı sunmuş ve düşük performans göstermiştir.

ROC eğrisi analizleri sonucunda Random Forest ve SVM modellerinin sınıflandırma sınırlarını daha başarılı çizdiği görülmüştür. Bu modellerin AUC değerlerinin 1.00 olması, pozitif ve negatif sınıfları ayırmadaki üstünlüklerini teyit etmektedir. Ayrıca, öznitelik önem derecelendirmelerinde en yüksek katkıyı sağlayan değişkenlerin; URL benzerlik skorları, iç bağlantı sayıları ve kod satırı uzunlukları olduğu belirlenmiştir. Bu durum, phishing web sitelerinin yapısal olarak ortak bazı özellikler taşıdığını ve bu özelliklerin sınıflandırma açısından belirleyici olduğunu göstermektedir.

Sonuçlar literatürdeki benzer çalışmalarla uyumludur. Tarawneh ve arkadaşlarının %94 doğruluk elde ettiği çalışmaya kıyasla, bu projede daha yüksek doğruluk ve AUC değerleri elde edilmiştir. Bu başarı, veri setinin kapsamlılığı, öznitelik mühendisliği ve doğru model seçimiyle ilişkilendirilebilir.

5. Sonuç ve Öneriler

Bu çalışmada, phishing saldırılarının tespitine yönelik olarak PhiUSIIL veri kümesi üzerinde çeşitli makine öğrenmesi algoritmaları uygulanmış ve bu modellerin başarıları karşılaştırılmıştır. Random Forest, SVM ve Karar Ağacı algoritmaları, yüksek doğruluk oranları ve üstün sınıflandırma kabiliyetleriyle öne çıkmıştır. Bu modellerin confusion matrix, ROC eğrisi ve metrik analizleri; hem güvenilirlik hem de genellenebilirlik açısından güçlü sonuçlar ortaya koymuştur.

Naive Bayes algoritmasının düşük başarı performansı, phishing URL verileri arasında bağımsızlık varsayımının geçerli olmamasıyla açıklanabilir. KNN ise veri yoğunluğu arttıkça hesaplama maliyeti nedeniyle pratikte tercih edilmeyebilecek bir yapı sergilemiştir. Ancak küçük veri setlerinde uygulanabilirliği mümkündür.

Elde edilen bulgulara dayanarak şu öneriler sunulmaktadır:

• Gerçek zamanlı phishing tespiti için Random Forest veya SVM algoritmaları entegre sistemlerde kullanılabilir.

• URL verileri WHOIS, DNS geçmişi gibi ek bağlamsal bilgilerle zenginleştirildiğinde model başarısı daha da artırılabilir.

• Veri dengesizliği olan durumlar için SMOTE veya ADASYN gibi tekniklerle model başarısı geliştirilebilir.

• Farklı coğrafyalardan veya yıllardan alınan verilerle modelin genellenebilirliği test edilmelidir.

• Model sonuçları kullanıcı arayüzlerine entegre edilerek, anlık risk uyarı sistemleri oluşturulabilir.

Kaynakça

1. Tarawneh, A., Al-Ebbini, L., & Al-Ani, A. (2020). *Phishing Website Detection Using Machine Learning Techniques*. International Journal of Advanced Computer Science and Applications (IJACSA), 11(1), 541–547.

2. Tutuncu, M. (2021). *Detecting Phishing Attacks Using AdaBoost Algorithm*. Journal of Computer Security, 9(3), 123–132.

3. Admojo, F., & Wan, S. (2021). *A KNN-based Classification Approach for Phishing Websites*. Journal of Cybersecurity Research, 8(2), 91–100.

4. Dey, A. (2016). *Machine Learning Algorithms: A Review*. International Journal of Computer Science and Information Technologies, 7(3), 1174–1179.

5. Gültepe, Y. (2019). *Makine Öğrenmesi Algoritmaları ile Hava Kirliliği Tahmini*. European Journal of Science and Technology, 16, 8–15.

6. Mowassir7. (2024). *PhiUSIIL Phishing URL Dataset*. GitHub repository: https://github.com/Mowassir7/PhiUSIIL-Phishing-URL-Dataset

7. Sahingoz, O. K., Buber, E., Demir, O., & Diri, B. (2019). *Machine learning based phishing detection from URLs*. Expert Systems with Applications, 117, 345–357.

8. Jain, A. K., & Gupta, B. B. (2018). *Phishing detection: analysis of visual similarity based approaches*. Security and Privacy, 1(2), e20.

9. Bahşi, H., & Karabacak, B. (2021). *Siber Güvenlikte Makine Öğrenmesi Kullanımı: Saldırı Tespiti Örneği*. Bilgi Güvenliği Akademisi Yayınları.