Makine Öğrenimi ile Zararlı Web İsteklerinin Tespiti

Detection of Malicious Web Queries with Machine Learning

Alper KARACA – Zeynep GÜNEY – Oğuz KORTUN – Oudoum Ali HOUMED
AISECLAB
(Artificial Intelligence Security & Defense Lab)
aiseclab.org

***Özetçe*— İnternetteki web uygulamalarının artışı ile birlikte web uygulamalarına yönelik saldırılar artmıştır. Gelişmiş saldırı yöntemlerinin tespiti, internetin güvenli kullanımı için önemli bir görevdir. Rastgele Orman sınıflandırma görevlerinde başarılı performans sergileyen popüler bir algoritmadır.. Bu çalışmada, zararlı web sorgularının tespiti için Rastgele Orman algoritması kullanılmıştır. Zararlı ve zararsız olarak etiketlenen örneklerden oluşan bir veri setiyle eğitilerek %97,38 test doğruluk başarımı elde edilmiştir. Geliştirilen model, canlı ortamda kullanıcıların hizmetine sunulmuştur.**

Anahtar Kelimeler — web uygulaması güvenlik duvarı, zararlı sorgular, toplu öğrenim modelleri.

***Abstract*— With the increase of web applications on the Internet, attacks agains web applications have increased. Detection of advanced attack methods is an important task fort he secure use of the Internet. Random Forest is a popular algorithm that performs well in classification tasks. In this study, Random Forest algorithm is used for the detection of malicious web queries. A test accuracy of 97.38% was achieved by training with a dataset consisting of samples labelled as malicious and harmless. The developed model was presented to the users in a live environment.**

Keywords — web application firewall, malicious queries, ensemble models.

I. GİRİŞ

Günümüzde teknolojinin hızlı gelişimi, internet tabanlı işlemlerin ve web uygulamalarının yayınlaşmasıyla birlikte siber tehditler artmıştır. İnternet üzerinden hizmet sunan işletmeler, web siteleri ve uygulamaları, çeşitli siber saldırıların hedefi olma riski altındadır. Bu tür saldırılar, veri sızıntıları, kimlik hırsızlıkları, hizmet kesintileri gibi bir dizi olumsuz sonuçlara yola açabilmektedir. Bu tür saldırıların önüne geçmek için Web Uygulama Güvenlik Duvarı adı verilen güvenlik çözümü kullanılmaktadır. Giderek karmaşıklaşan saldırılar karşısında geleneksel web güvenlik duvarları etkili olamamaktadır. Bu nedenle, modern siber güvenlik zorluklarına daha iyi yanıt verebilmek için makine öğrenmesi algoritmaları önemli bir rol oynamaktadır. Makine öğrenmesi algoritmaları, saldırı girişimlerini tespit etmek için gelişmiş yetenekler sunar. Özellikle sıradışı davranışları tespit etme konusunda yetenekli olan bu algoritmalar, saldırıları hızlı bir şekilde belirleyebilir ve karşılık verebilir. Bununla beraber, Tablo 1'de ifade edildiği gibi son yıllarda zararlı web sorgularını tespit etmek için makine öğrenimi algoritmalarının kullanılması giderek popüler hale gelmiştir.

TABLO I. MAKİNE ÖĞRENMESİ TABANLI WEB UYGULAMA GÜVENLİĞİ İLE İLGİLİ LİTERATÜR ÖZETİ

Yazarlar	*Veri seti*	*Modeller*	*Doğruluk*
Bayrak vd. (2022) [1]	CSIC 2010	LSTM	%86,0
Sezer vd. (2022) [2]	CSIC 2010	LSTM	%94,10
İncir vd. (2020) [3]	PhishTank	Rastgele Orman	%96,17
Demir vd. (2023) [4]	Üniversitenin Firewall cihazından toplanmış	SVM	%98,50
Erçin vd. (2022) [5]	Elle oluşturulmuş	CNN	%80,00
Özekes vd. (2019) [6]	CICIDS2017	Rastgele Orman	%99,96
Demir vd. (2021) [7]	ISCX-2012	Karar Ağacı	%100,00
Özer vd. (2019) [8]	Karma	Naive Bayes	%79,00
Şahingöz vd. (2019) [9]	NSL-KDD	Karar Ağacı	%99,34

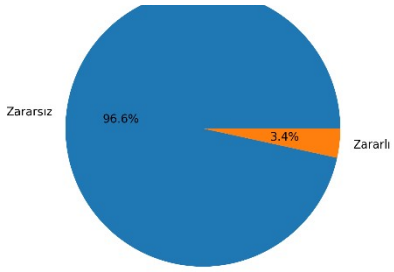
Bu çalışmada, zararlı web sorgularının tespiti için Rastgele Orman Algoritması kullanılmıştır. Çalışmanın ilk adımında, sorgu metni verilerinden belirli özellik çıkarılmış ve bu özellikler Rastgele Orman algoritması modeli girişi olarak kullanılmıştır. İkinci aşamada, zararlı ve zararsız olarak etiketlenmiş veriler kullanılarak Rastgele Orman Algoritması modeli eğitilmiştir. Eğitim sırasında model, belirli özellik aracılığıyla zararlı web sorgularını tespit etmeyi öğrenmiştir.

Elde edilen model, test sorgularını zararlı veya zararsız olarak doğru bir şekilde %97,38 başarımla tespit edebilmiştir.

II. MALZEME

A. *Veri Seti*

2017 yılında Faizan vd. [11] özgün veri kümesi oluşturmuşlardır. Web sorgularından oluşmuş veri seti zararlı web sorgu tespiti için toplam 1,310,707 sınıflandırılmış sorgu içermektedir. Veri setindeki zararlı/zararsız sorgu dağılımları Şekil 1'deki gibidir.



Şekil.1. Veri Setindeki Zararlı ve Zararsız Sorgu Dağılımları

B. *Özellik Çıkarımı Adımları*

Veri seti incelendiğinde özellik çıkarımı için aşağıdaki çıkarımlar yapılmıştır.;

1. XSS, LFI, SQLi ve OS Command Injection Şablonu Arama: Web uygulamalarının güvenlik duvarları veya filtreleme kuralları, gelen istekleri bu tür zararlı şablonlara karşı tarayabilir. Zararlı şablonları aramak, bu tür kuralların daha etkili çalışmasına yardımcı olabilir ve saldırıları engelleyebilir.
2. Sorgu ve Parametre Uzunluğu: Saldırganlar, uzun ve karmaşık sorgu parametreleri kullanarak güvenlik kontrollerini aşmayı ve kötü amaçlı işlemlerini gizlemeyi hedefleyebilir.
3. Noktalama İşaretlerinin Sayısı: Birçok web saldırısı, sorgu veya parametreler aracılığıyla zararlı kodlar veya komutlar enjekte etmeye dayanır. Bu zararlı enjeksiyonlar sıklıkla noktalama işaretlerini içerir. Dolayısıyla, sorguda anormal derecede noktalama işareti bulunması, saldırı girişiminin bir göstergesi olabilir.
4. Parametre Sayısı: Fazla sayıda parametre, web uygulamalarının işleyebileceği maksimum veri boyutunu aşabilir ve uygulamada veri bütünlüğü sorunlarına neden olabilir. Çok fazla parametreye sahip sorgular, bir anormallik göstergesi olabilir.
5. Sorgudaki Toplam Harf ve Rakam Sayısı: Web uygulamaları, gelen verilerin bütünlüğünü ve doğruluğunu sağlamak için harf ve rakam gibi güvenilir karakterleri bekler. Sorgudaki anormal karakter veya semboller, veri bütünlüğünü bozabilir veya hassas verilere zarar verebilir. Bu nedenle, sorgudaki harf ve rakam sayısını izlemek, beklenen veri formatına uymayan girişleri tespit etmeye yardımcı olur.

Bu çalışmada uygulanan özellik çıkarımı adımları Şekil 2'de verilmiştir.

Şekil.2. Veri Setine Uygulanan Özellik Çıkarımı Adımları

III. YÖNTEM

Bu kısımda çalışmada kullanılan Makine Öğrenmesi yaklaşımlarından bahsedilmiştir.

Karar Ağacı Algoritması, gözetimli öğrenme kategorisine ait bir algoritmadır ve başta sınıflandırma olmak üzere regresyon problemlerini çözmek amacıyla kullanılır. Bu algoritma, bir veri kümesini küçük alt kümeler veya yapraklar halinde bölen ve her bir yaprakta belirli bir sınıf veya sayısal değerle sonuçlanan bir ağaç yapısı oluşturur. Ağaç yapısının temelini, kök düğümden başlayarak veriyi sınıflandırmak veya tahmin etmek için kullanılan karar kuralı düğümleri oluşturur. Bu karar kuralı düğümleri, özellikleri ve bu özelliklere ilişkin eşik değerlerini içerir ve veriyi bu kriterlere göre bölerek alt düğümlere yönlendirir. Sonuç olarak, veri kümesi her seviyede daha homojen hale gelir ve sonuçları temsil eden yaprak düğümlerine ulaşılır. Karar Ağacı Algoritması, verilerin karmaşıklığını anlama, sınıflandırma ve tahmin yetenekleri sunma, görselleştirilmesinin kolaylığı gibi avantajlara sahip bir algoritmadır ve bu nedenle pek çok uygulama alanında tercih edilir.

Lojistik Regresyon, sınıflandırma problemlerini çözmek için kullanılan bir istatistiksel modeldir. Temel amacı, bir veri noktasının belirli bir sınıfa ait olma olasılığını tahmin etmektir. Bu algoritma, girdi özelliklerinin ağırlıklı toplamını hesaplar ve sonucunu bir sigmoid fonksiyonundan geçirerek 0 ile 1 arasında bir olasılık değeri üretir. Bu olasılık değeri, veri noktasının bir sınıfa ait olma olasılığını temsil eder. Lojistik Regresyon, özellikle iki sınıflı sınıflandırma problemlerinde yaygın olarak kullanılır.

Rastgele Orman, birçok karar ağacının bir araya getirilerek oluşturduğu bir topluluk öğrenimi yöntemidir. Her bir ağaç, veri kümesinin rastgele örneklemeleri üzerinde bağımsız olarak eğitilir ve sonuçları bir araya getirilerek nihai tahmin yapılır, Bu yöntem, sınıflandırma ve regresyon problemleri için etkili bir şekilde kullanılır ve aşırı uydurmaya karşı dayanıklıdır. Rastgele Orman, veri kümesindeki önemli özellikleri belirleme yeteneği ve yüksek tahmin doğruluğu ile bilinir.

XGBoost, bir “Gradient Boosting” yöntemini temel olan bir topluluk öğrenimi yöntemidir. Bu algoritma, birçok zayıf tahminci kullanarak güçlü bir tahminci oluşturur. Gradient Boosting algoritmasına kıyasla daha hızlı ve daha iyi performans sağlar ve aşırı uydurmayı önlemek için bazı düzenlemeler içerir. Sonuçları optimize etme yeteneği ile bilinir.

IV. DENEYSEL SONUÇLAR

A. Model Eğitimi

Rastgele Orman, Karar Ağacı, Lojistik Regresyon ve XGBoost modellerine uygulanan eğitim ve test veri setleri Tablo 2’de verilmiştir. Veri setinin %80’i eğitim ve %20’si test veri seti olarak ayrılmıştır.

TABLO II. UYGULANAN EĞİTİM VE TEST VERİ SETİ

Sınıf	***Eğitim***	***Test***
Zararlı (1)	35,675	9,038
Zararsız (0)	1,012,890	253,104

Veri setindeki bu dengesizliği gidermek için SMOTE yöntemi uygulanmıştır. SMOTE, azınlık sınıfına ait örnekleri sentetik olarak üretir, böylece sınıflar arası dengesizlik azaltılır. Veri arttırma işleminden sonra sayısal özellikler ölçeklendirilerek modeller eğitilmiştir. Elde edilen skorlar Tablo 3’te verilmiştir. En iyi sonucu Rastgele Orman algoritması modeli elde etmiştir.

TABLO III. EĞİTİLEN MODELLERİN SKORLARI

Model	***Doğruluk***	***Kesinlik***	***Duyarlılık***	***F1 Skoru***
Lojistik Regresyon	0.9296	0.9703	0.8864	0.9264
Karar Ağacı	0.9683	0.9928	0.9434	0.9674
Rastgele Orman	0.9684	0.9925	0.9439	0.9676
XGBoost	0.9618	0.9860	0.9369	0.9609

B. Hiper Parametre Uygulanması

Fit edilecek bağımsız ağaç sayısı için kullanılan hiper parametre aralığı 100, 200 ve 300 olarak belirlenmiştir. Bir düğümün bölünmeden önce sahip olması gereken minimum örnek sayısı aralığı 2, 5 ve 10 olarak belirlenmiştir. Bir yaprağın sahip olması gereken minimum örnek sayısı içinse 1, 2 ve 4 olarak belirlenmiştir. İyileştirme aşamasında uygulanan optimum parametre değerleri Tablo 4’te verilmiştir.

TABLO IV. İYİLEŞTİRME İÇİN UYGULANAN OPTİMUM PARAMETRİK DEĞERLER

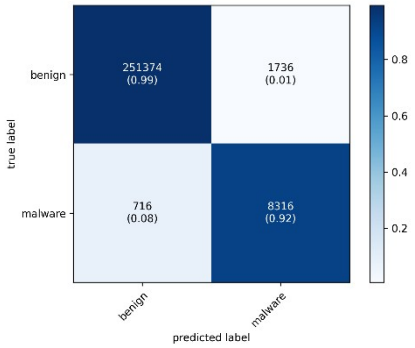
Parametre Adı	***Parametrik Değer***
criterion	gini
n_estimators	100
min_samples_leaf	1
min_samples_split	2
max_feature	sqrt

C. Model Başarım Sonuçları

Optimum değerlerle elde edilen modelin eğitim veri seti için doğruluk oranı % 97,38 elde edilmiştir. Modelin test başarımı Tablo 5’te verilmiştir.

TABLO V. MODEL TEST BAŞARIM SONUÇLARI

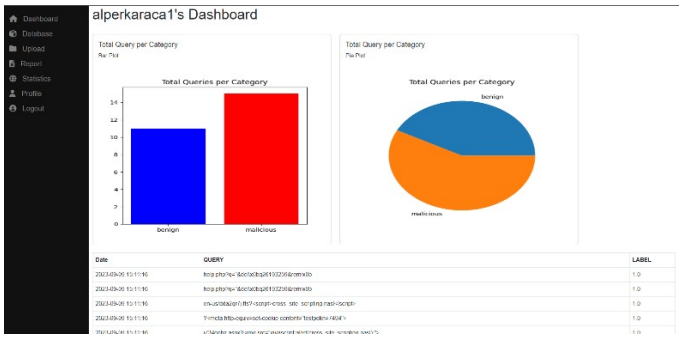
Sınıf	***Hassasiyet***	***Geri Çağırma***	***F1-skor***
Zararlı (1)	%94,00	%99,92	%97,00
Zararsız (0)	%99,00	%99,93	%97,00



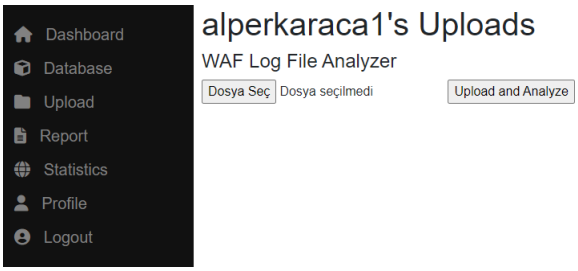
Şekil.3. Test veri seti için karışıklık matrisi

D. Modelin Canlıya Alınması

Bu çalışmada amaçlanan kullanıcılarının web sorgu istekleri veri ön işleme adımları ile işlenerek Rastgele Orman modeli ile zararlı ve zararsız olarak tespit edilebilmektedir. Kullanıcıların web sorgu isteklerinin kaydedildiği bir dosyayı servise yükleyerek isteklerin zararlı ve zararsız olarak tespiti için geliştirilen modele canlı ortamda Flask uygulaması ile ulaşılabilmektedir [12].



Şekil.4. Kullanıcı Arayüzü



Şekil.5. Log dosyası yükleme ekranı

Şekil.6. Modelin tahminleri sonrası oluşturulan sonuç grafikleri

V. SONUÇ VE GELECEK HEDEFLERİ

Zararlı WAF sorgusu tespitinde hazır veri seti kullanılarak zararlı/zararsız sorgu tespiti Rastgele Orman algoritması ile modellenmiştir. Model test başarımı %97,38 olarak elde edilmiştir. Model canlı ortamda kullanıcıların hizmetine açılmıştır. Modelin iyileştirilmesinde farklı yöntemler kullanılarak test başarımının arttırılması hedeflenmektedir.

VI. TEŞEKKÜR

Proje geliştirilmesinde yaptıkları yardımlardan dolayı AISecLab [13] organizasyonuna teşekkür ederiz.

KAYNAKLAR

[1] Bayrak, Ş., & Ardanuç, B. Web Uygulamaları için Derin Öğrenme Yöntemiyle Güvenlik Duvarı Uygulaması Firewall Application with the Deep Learning Method for the Web Applications.

[2] Sezer, T., & Ali, Y. (2022). Derin Öğrenme Tekniği Kullanarak Anomali Tabanlı Web Uygulama Güvenlik Duvarı. *Acta Infologica*, *6*(2), 219-244.

[3] İncir, R. (2020). *Derin öğrenme yöntemi kullanarak web tabanlı kimlik avı saldırılarının sınıflandırılması* (Master's thesis, Fen Bilimleri Enstitüsü).

[4] Demir, S., & Aslan, Z. (2023). K-NN, NN ve Feature Selection yöntemleri ile firewall verilerinin sınıflandırması. *Anadolu Bil Meslek Yüksekokulu Dergisi*, *17*(66), 139-148.

[5] Erçin, M. S., & Yolaçan, E. (2022). SQLi ve XSS Saldırı Tespitinde Kullanılan Yeni Bir Özellik Çıkarma Yöntemi. *Uluslararası Bilgi Güvenliği Mühendisliği Dergisi*, *8*(1), 1-11.

[6] Özekes, S., & Karakoç, E. N. (2019). Makine öğrenmesi yöntemleriyle anormal ağ trafiğinin tespit edilmesi. *Düzce Üniversitesi Bilim ve Teknoloji Dergisi*, *7*(1), 566-576.

[7] Demir, F. (2021). Siber saldırı tespiti için makine öğrenmesi yöntemlerinin performanslarının incelenmesi. *Balıkesir Üniversitesi Fen Bilimleri Enstitüsü Dergisi*, *23*(2), 782-791.

[8] Özer, Ç., & Takaoğlu, M. SALDIRI TESPĠT SĠSTEMLERĠNE MAKĠNE ÖĞRENME ETKĠSĠ.

[9] Şahingöz, O. K., Çebi, C. B., Bulut, F. S., Fırat, H., & Karataş, G. (2019). Saldırı tespit sistemlerinde makine öğrenmesi modellerinin karşılaştırılması. *Erzincan University Journal of Science and Technology*, *12*(3), 1513-1525.

[11]*https://github.com/faizann24/Fwaf-Machine-Learning-driven-Web-Application-Firewall*

[12] *https://github.com/thealper2/AISECLAB-cyber-inspector*

[13] https://aiseclab.org