Erzincan Üniversitesi
Fen Bilimleri Enstitüsü Dergisi
2019, 12(3), 1513-1525
ISSN: 1307-9085, e-ISSN: 2149-4584
Araştırma Makalesi

Erzincan University
Journal of Science and Technology
2019, 12(3), 1513-1525
DOI: 10.18185/erzifbed.573648
Research Article

Saldırı Tespit Sistemlerinde Makine Öğrenmesi Modellerinin Karşılaştırılması

Cem Berke ÇEBİ[1], Fatma Sena BULUT[1], Hazal FIRAT[1],
Gözde KARATAŞ[2], Özgür Koray ŞAHİNGÖZ[1*]

[1]İstanbul Kültür Üniversitesi, Bilgisayar Mühendisliği Bölümü, İstanbul
[2]İstanbul Kültür Üniversitesi, Matematik-Bilgisayar Bölümü, İstanbul

Geliş / Received: 03/06/2019, Kabul / Accepted: 15/12/2019

Öz

Son yıllardaki gelişen teknolojiler neticesinde her türlü hesaplama cihazının İnternete bağlanması sağlanmıştır. Bu sayede birçok gerçek dünya problemi yeni ağ düzenine aktarılsa da bu tam-kontrol sağlanamayan sanal platform çok sayıda güvenlik açığı içermektedir. Günümüzde ağ yöneticilerin ana görevlerinden biride bu açıkları kapatmak ve sorumlu oldukları bilgisayar ağını saldırılardan korumaktır. Güvenlik duvarlarının kullanımı dışarıdan yapılan saldırıları ciddi boyutta engellese de içeriden yapılabilecek veya daha önceden karşılaşılmayan tipten saldırılara karşı zafiyetler içermektedir. Saldırı Tespit Sistemleri (STS) bu zafiyetleri ortadan kaldırmak için öncelikle tercih edilebilecek uygulamalardır. Son geliştirilen STSleri incelendiğinde dinamik bir güvenlik mekanizması geliştirmek adına özellikle Makine Öğrenmesi tabanlı sistemlere ağırlık verildiği görülmektedir. Bilgisayar donanımları ve paralel hesaplama teknolojilerinde ortaya çıkan gelişmeler ve Büyük Veri işleme teknolojilerinin, Makine Öğrenmesi tabanlı sistemlerle uyumlu kullanıldığı görülmektedir. Bu çalışmada yedi farklı makine öğrenimi algoritmaları kullanarak STSlerin geliştirilmesi amaçlanmıştır. Elde edilen sonuçlar başarım, eğitim süreleri ve çalıştırma süreleri açısından karşılaştırılarak farklı kriterlere göre uygun algoritmanın ortaya konmuştur. Bu karşılaştırma için genel kabul gören NSL-KDD veri setinden faydalanılmıştır. Başarım sonuçlarına bakınca Adaboost algoritmasının en yüksek doğruluk oranına ulaştığı görülmektedir. Ancak gerek eğitim süresi gerekse çalışma zamanı performansı göz önüne alınınca Karar Ağacı algoritmasının daha yüksek performans gösterdiği, doğruluk oranı değeri itibarı ile de Adaboost'a yakın değere sahip olduğu görülmektedir.

Anahtar Kelimeler: Saldırı Tespit Sistemleri, Makine Öğrenmesi, ANN, NSL-KDD, Tensorflow.

Comparison of Machine Learning Based Models in Intrusion Detection Systems

Abstract

As a result of developing technologies in recent years, all kinds of computing devices can be connected to the Internet. In this way, many real-world problems are transferred to the new network layout, but this uncontrollable virtual platform contains many vulnerabilities. One task of network administrators is closing these leaks and protecting the network from attacks. Although use of firewalls can prevent serious attacks from outside, there are many attacks from inside or previously unknown. Intrusion Detection Systems (IDSs) are the most preferable applications to eliminate these vulnerabilities. When recently IDSs are examined, it is seen that Machine Learning-based systems are focused on in order to develop a dynamic security mechanism. It is seen that developments in hardware and parallel computing and Big Data processing technologies are used compatible with these systems. In this study, it is aimed to develop STS using seven different algorithms. Results were compared in terms of performance, training and running times, and appropriate algorithm was determined. NSL-KDD dataset was used as generally accepted-dataset. The results showed Adaboost algorithm achieves the highest accuracy. However, when both training-time and runtime performance are considered, Decision Tree algorithm performs better and close to Adaboost in terms of accuracy.

Keywords: Intrusion Detection Systems, Machine Learning, ANN, NSL-KDD, Tensorflow.

*Sorumlu Yazar: o.sahingoz@iku.edu.tr

1. Giriş

Teknoloji çağında, internetin kullanımı ve internete bağlı cihazların sayısı sürekli artmaktadır. Kurumsal ve kişisel küçük görevleri yerine getirmekten tutun buzdolabınızdaki yiyeceklerin durumunu görüntülemek için bile internet bağlantısı gerekmektedir. Bu küresel ağ kullanılarak veriler yoğun bir şekilde paylaşılabilir, depolanabilir ve etkileşime girilebilir. İnternete olan ihtiyaç arttıkça, güvenlik sorunu; teknoloji ve yetenekler geliştikçe, çeşitli tehditler, güvenlik açıkları ve saldırılar ortaya çıkmaktadır (Zhang, Li ve Wang, 2019), (Karatas ve Sahingoz, 2018).

Kurumlar veya bireyler için kritik önem taşıyan bilgilerin güvenceye alınması bir öncelik haline gelmiştir. Herhangi bir güvenlik ihlali veya veri kaybı ciddi sonuçlara neden olabilmektedir ve bu sonuçlar kişisel verilerin bozulmasına, yasaların ve düzenlemelerin ihlal edilmesine, para ve itibar kaybına yol açabilir (Salo ve ark. 2018). Bu nedenle, özellikle işletme ölçeğinde güvenlik çok önemlidir. Artan bilgi miktarı ile birlikte sistemleri ve verileri güvence altına almak zorlaşmıştır. Mevcut saldırılar ve güvenlik açıkları hakkındaki yöntem, bilgiler ve savunma yöntemleri yayıldıkça, saldırılar daha karmaşık hale gelmektedir. Günümüzdeki bilgiye kolay erişim imkanları sayesinde, bireylerin, bilgi olmadan güvenlik açıklarından yararlanmak için farklı türdeki araçları kullanmalarına olanak sağlamaktadır.

İnternetin anonim yapısı sayesinde, günümüzde saldırganların sisteme müdahale etmesi kolaylaşmıştır ve bu nedenle, kritik bilgileri korumak için izinsiz giriş tespitine duyulan ihtiyaç gün geçtikçe artmaktadır (Grammatikis ve Sarigiannidis, 2019). Güvenlik duvarları ve antivirüs sistemlerin korunmak istenen ağların veya bilgisayarların güvenliklerini ciddi oranda sağlamaktadır. Antivirüs yazılımları izinsiz uygulama çalışımı konusunda sınırlı güvenlik sağlar ve güvenlik duvarları, ağ trafiğini filtreleyerek kurum ağının ilk savunma hattı olarak görev yapmaktadır (Ruiz-Vanoye vd., 2007). Ancak, bu teknolojiler özellikle içeriden yapılacak veya ilk defa karşılaşılan sıfır-gün (zero-day) saldırısı diyebileceğimiz izinsiz girişleri tespit etmek için yetersizdir. Bu nedenle, izinsiz giriş tespiti için farklı saldırı türlerini iyice incelemek ve tanımlamak için daha sofistike sistemlerin kullanılması gerekmektedir. Bu konuda kural tabanlı sistemlerin kullanılması, geliştirmenin kolaylığı ve hızlı tespit sağlaması sebebi ile yoğun olarak tercih edilmekte olmasına rağmen yeni tip saldırılara karşı kuralların mevcut olmamasından dolayı sistemin korunmasında zafiyet doğurmaktadır. Bu nedenle kendi kendini eğiten, bilgi tabanının dinamik olarak güncellenebildiği sistemlere doğru bir kayış yaşanmaktadır. Bu ikinci tip sistemler için farklı makine öğrenmesi algoritmalarının kullanılması ile gerek normal taleplerin gerekse farklı türdeki saldırıların ayırt edilmesi amaçlanmaktadır (Alhakami vd., 2019). Hangi makine öğrenmesi algoritmasının kullanılmasının daha uygun olduğu ve hangi parametrik değerlerin tercih edilmesi gerektiği araştırmacılar tarafından incelenmektedir.

Bu çalışmada popüler Makine Öğrenmesi algoritmalarından olan Karar Ağacı, Rastgele Orman, K-En Yakın Komşu, Adaboost, Gradyan Artırma, Doğrusal Ayrımcılık Analizi, Yapay Sinir Ağları algoritmalarının kullanımı ile geliştirilen STSlerin Doğruluk Oranları, Eğitim Süreleri ve Çalıştırma Süreleri Performansları açısından incelenmesi amaçlanmıştır. Elde edilen deneysel sonuçlar karşılaştırmalı olarak sunulmuş ve amaçlanan

hedefler doğrultusunda hangi modelin tercih edilmesinin uygun olarak gösterilmiştir. Sistem performansının karşılaştırılması için küresel kabul gören bir veri seti kullanılması amaçlanmıştır. Literatürdeki çoğu çalışma KDDCup99 veri setini kullanmış olmasına rağmen, tekrarlanan kayıtların silinmesi ile elde edilen NSL-KDD veri setinin gerçek dünya verileri ile daha uyumlu olduğu (Dhanabal vd., 2015) değerlendirilmiş ve kullanılmıştır.

Makalenin kalan kısmı şu şekilde organize edilmiştir: bir sonraki kısımda konuyla ilgili çalışmalar bahsedilmiştir. Önerilen sistem detayları kullanılan makine öğrenmesi metotlarıyla birlikte 3ncü kısımda bahsedilmiştir. Elde edilen deneysel sonuçlar 4ncü kısımda listelenmiş, bu sonuçların tartışması 5nci kısımda yapılmış ve son kısımda ise sonuç ve gelecek çalışmalar özetlenmiştir.

2. İlgili Çalışmalar

Bu kısımda; literatürdeki konu ile ilgili çalışmalar, Makine Öğrenmesi, kullanılan veri seti hakkında bilgi verilmiştir.

2.1. Literatürdeki Çalışmalar

Dong ve Wang, geleneksel Makine Öğrenmesi ve Derin Öğrenme algoritmalarını karşılaştırmıştır (Dong vd., 2016). Çalışmanın temel amacı, farklı modelleri test etmek ve problem için en iyi sonuçları bulmaktır. Ayrıca, denetimsiz bir öğrenme algoritması olan Sınırlı Boltzmann Makinesi (RBM), modelin ana yapısı olarak kullanılmıştır. Yaptıkları testler sonucunda, Derin Öğrenme algoritmalarının geleneksel Makine Öğrenmesi algoritmalarından daha iyi sonuç verdiğini görülmüştür.

Aziz ve Hassanien, saldırıların artması ve farklılaşması nedeniyle tek bir tekniğin izinsiz giriş tespiti için yeterli olmadığını vurgulayan bir çalışma geliştirmiştir (Aziz vd., 2014). Bu çalışmada, üç katman ile oluşturulan çok katmanlı bir yapay bağışıklık sistemi önerilmiştir. Sistem sınıflandırıcıları kullanarak belirli işlemler yapar. Sınıflandırıcılar, bulunan anomalileri doğru sınıflarına ve yanlış değerlere etiketlemek için kullanılır. Çalışmaya göre, modelin daha az zamanda çalışabilmesi ve sonuçların en iyi değerde çıkabilmesi için eğitim setinin %20'lik kısmı kullanılmıştır. Bu çalışma sonucunda, her birinin kendi işlevselliğine sahip olduğu farklı tekniklerin bir kombinasyonunu kullanan çok katmanlı bir sistemin daha iyi sonuçlar verdiğini gösterilmiştir.

Sahingoz ve arkadaşları 2019 yılında, yedi farklı sınıflandırma algoritması ve doğal dil işleme (NLP) tabanlı özellikler kullanan gerçek zamanlı saldırılara karşı kimlik tespiti sistemi önermiştir (Sahingoz vd., 2019). Geliştirilen sistem, literatürdeki diğer çalışmalardan ayırt edici özelliklere sahiptir: dil bağımsızlığı, kimlik tespiti ve meşru veri kullanımı, gerçek zamanlı uygulama, yeni web sitelerinin tespiti, üçüncü taraf hizmetlerinden bağımsızlık ve zengin özellikli sınıflandırıcıların kullanımı gibi. Sistemin performansını ölçmek için yeni bir veri seti oluşturarak ve test işlemi bu veri seti üzerinde gerçekleşmiştir. Uygulanan sınıflandırma algoritmalarının deneysel ve karşılaştırmalı sonuçlarına göre, yalnızca NLP tabanlı özelliklere sahip "Random Forest" algoritması, URL'lerinin tespiti için %97,98 doğruluk oranıyla en iyi performansı vermiştir.

Akıllı şehirler son yıllarda çok popülerleşmesi sebebi ile Elsaeidy ve arkadaşları 2019 yılında

Sınırlı Boltzmann Makineleri'ne (RBM'ler) dayanan akıllı bir şehir saldırı tespit sistemi önermiştir (Elsaeidy vd., 2019). RBM'ler, ham veriden yüksek seviyeli özellikleri denetimsiz bir şekilde öğrenebilmeleri ve akıllı sayaçlardan ve sensörlerden üretilen gerçek veri gösterimini ele almaları nedeniyle tercih edilmiştir. Önerilen yöntemin performansı akıllı bir su dağıtım tesisinden temin edilen veri seti kullanılarak test edilmiş ve karşılaştırılmıştır. Sonuç olarak, önerilen yöntemin saldırı tespitinde yüksek başarım sağladığını görülmüştür. Ek olarak, önerilen yöntemin, özellik öğrenme aşaması olmadan uygulanan sınıflandırma modelinden daha iyi performans gösterdiği saptanmıştır.

Taher ve arkadaşları, ağ trafiğini kötü niyetli veya iyi niyetli şeklinde sınıflandırmak için yeni bir denetimli makine öğrenme yöntemi geliştirmiştir (Taher vd., 2019). Tespit başarı oranını dikkate alan, en iyi modeli bulmak için denetimli öğrenme algoritması kombinasyonları ve özellik seçme yöntemi kullanılmıştır. Bu çalışma ile ağ trafiğini sınıflandırırken, Yapay Sinir Ağı (ANN) tabanlı makine öğreniminin, özellik seçimi seçiminde, destek vektör makineleri (SVM) yönteminden daha iyi performans gösterdiği fark edilmiştir. Performansı değerlendirmek için, NSL-KDD veri seti, SVM ve ANN denetimli makine öğrenme tekniklerini kullanarak ağ trafiğini sınıflandırmak için kullanılmıştır. Karşılaştırmalı sonuçlarla, önerilen modelin izinsiz giriş tespit başarı oranına göre mevcut diğer modellerden daha verimli olduğunu gösterilmiştir.

Makine Öğrenmesi, sayısal öğrenme ve yapay zekânın model tanıma çalışmaları üzerine geliştirilen bir bilgisayar bilimi alt dalıdır. Makine Öğrenmesi, yapısal bir işlev olarak öğrenebilen ve veriler üzerinde bir tahmin yapabilen algoritmaların yapısını ve işlevini öğrenen bir sistemdir (Sahingoz vd., 2019). Bu tür algoritmalar, sabit program talimatlarını yerine getirmek yerine, veri tabanlı tahminleri ve örnek girdilerden alınan kararları gerçekleştirmek için bir model oluşturarak çalışır (Taher vd., 2019). Makine Öğreniminin 3 farklı alt bölümü vardır; bunlar Denetimli Öğrenme, Denetimsiz Öğrenme ve Pekiştirmeli Öğrenme. Ayrıca Sınıflandırma (İkili ve Çoklu), K-En Yakın Komşuluk, Karar Ağaçları, Destek Vektör Makineleri, Regresyon, Kümeleme gibi algoritmalara sahiptir. Bu algoritmalardan 3 tanesi (Ahmad ve ark. 2018) tarafından karşılaştırılmış ve elde edilen sonuçlar bir makale olarak okuyuculara sunulmuştur.

3. Önerilen Sistem

Saldırı tespit sistemleri, ilerleyen teknoloji ile beraber, artan ihtiyaçlar doğrultusunda istekleri karşılamaya uygun olmalıdır (Thomas ve Pavithran, 2018). Hala gelişmekte ve birçok araştırmaya konu olan Makine Öğrenmesi, saldırı tespit sistemlerinde başvurulan yöntemlerdendir (Athmaja vd., 2017). Bu çalışmada, Makine Öğrenmesi algoritmaları kullanılarak kendini dinamik olarak geliştirebilen bir saldırı tespit sistemi önerilmiştir. Saldırı tespit sistemlerinde bu yöntemlerin kullanılmasının amacı; sistemin, hakkında bilgisi olmadığı bir veriyi hızlı ve yüksek doğruluk oranıyla tahmin edebilmesidir. Çalışmada Makine Öğrenmesi modellerinin uygulandığı veri seti olarak NSL-KDD kullanılmıştır. Karar Ağacı, Rastgele Orman, K-En Yakın Komşu, Adaboost, Gradyan Arttırma, Doğrusal Ayrımcılık Analizi ve Yapay Sinir Ağları; kullanılmıştır. Kullanılan veri seti ve modeller ile ilgili bilgiler aşağıda detaylandırılmıştır.

3.1. Veri Seti

Veri seti seçimi, geliştirilen algoritmaların test edilmesi için çok önem arz etmektedir. Elde edilen sonuçların farklı çalışmalarla karşılaştırılması için dünya çapında kabul gören ve güncel veri setinin kullanılması gerekmektedir. Bu nedenle bu çalışmada NSL-KDD veri setinin kullanılması tercih edilmiştir. NSL-KDD veri seti 3. Veri Madenciliği ve Bilgi Çıkarımı yarışmasında veri olarak kullanılan ve MIT Lincoln Laboratuvarı tarafından toplanılan ve dağıtılan KDD Cup99 veri setinden tekrarlanan ve artık kayıtların çıkarılmasıyla oluşturulmuş olup günümüzde en yaygın kullanılan güncel veri setlerinden kabul edilmektedir (Dhanabal vd., 2015).

NSL-KDD veri seti, 149.470 adet veri içerir ve 41 özellik ile beraber 1 paket/saldırı sınıfından oluşmaktadır. Bu özellikler 3 grupta sınıflandırılmaktadır: *Temel özellikler, Trafik özellikleri ve İçerik özellikleri*. İlgili özelliklerin detaylarına (Dhanabal vd., 2015) kaynağından erişilebilinir.

NSL-KDD veri setindeki saldırı türleri 4 kategori altında incelenir;

- Probe Saldırıları: Satan, Ipsweep, Portsweep, Nmap.
- Denial of Service (DoS) Saldırıları: Pod, Land, Smurf, Neptune, Back, Teardrop.
- Remote to Local (R2L) Saldırıları: Phf, Spy, Imap, ftp_write, Multihop, Guess_passwd, Warezmaster, Warezclient.
- User to Root (U2R) Saldırıları: Rootkit, Perl, loadmodule, Buffer_overflow.

3.2. Veri İşleme

NSL-KDD veri setinin Makine Öğrenmesi modelleriyle uyumlu olabilmesi ve sağladığı bilgilerden faydalanılabilmesi için veri ön-işlemesi gerçekleştirilmiştir. Veri setinde ‘protocol_type’, ‘service’, ‘flag’ özellikleri ve saldırı sınıflandırılması olan ‘label’ sayısal halde bulunmamaktadır. Ön-işleme sırasında bu değerler sayısal hale getirilmiştir. ‘protocol_type’ özelliği veri setinde ‘icmp’, ’tcp’ ve ‘udp’ olmak üzere 3 farklı değeralmaktadır ve bu değerler sırasıyla 0, 1 ve 2 olacak şekilde sayısallaştırılmıştır. Veri setinde ‘service’ özelliği 66, ‘flag’ özelliği ise 11 farklı değer almaktadır. Bu özelliklerin değerleri bulundukları miktar kadar 0’dan başlayarak numaralandırılmıştır. Son olarak, saldırı sınıflandırılmasının bulunduğu ‘label’ için toplam 5 değer bulunmaktadır. Etiket sınıflandırmaları DoS, R2L, Probe ve U2R saldırılarına göre gerçekleştirilmiş ve Tablo 1’deki gibi sayısallaştırılmıştır.

Tablo 1. NSL-KDD veri setinde saldırı sınıflarının numaralandırılması

Etiket Sınıfı	Numara
Normal	100
DoS	200
R2L	300
Probe	400
U2R	500

3.3. Kullanılan Makine Öğrenmesi Yaklaşımları

Bu kısımda çalışmada kullanılan Makine Öğrenmesi yaklaşımlarından bahsedilmiştir.

Karar Ağacı Algoritması (Decision Tree-DT), sayısal ve sınıfsal verilerin sınıflandırılması için kullanılan gözetimli öğrenme algoritmalarından biridir (Gavankar vd., 2017). Karar ağacı, çok sayıda veriden oluşan bir veri kümesinin elemanlarını, karar verme adımları ile uygulayarak küçük kümelere bölünmesini sağlayan bir yapıya sahiptir. Önceden tanımlanmış bir hedef

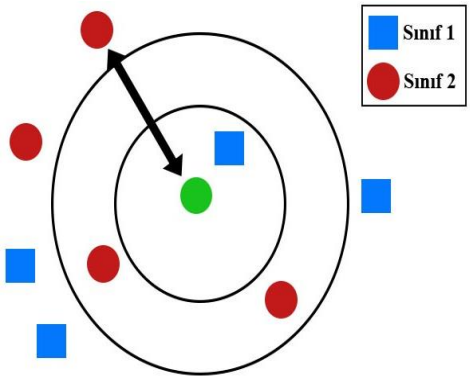
değişkenine sahiptir ve algoritma yapısı açısından yukarıdan aşağıya doğru hedeflerden birine ulaşabilmek için karar verme adımları ile destekli yaprak düğümlere sahiptir. Yapısının basit olmasından doğan avantajı kullanarak yüksek miktardaki veri yığınlarını hızlı bir şekilde işlemeyi sağlar. Bazı durumlarda ise veri kümelerinin sınıflandırılması için daha karmaşık ağaçlar ile başa çıkmak zorunda kalınabilir. Bu gibi bir problem de karar ağaçlarının dalları haliyle daha karmaşık hale gelir ve hedeflenen değişkenlerden herhangi birine ulaşmak daha zorlaşır. Ezbere öğrenme karar ağacı algoritmalarında denk gelinen problemlerden bir diğeridir. Bu problemin çözümü için yaprak düğümlerinden bazıları budanarak karar ağacı içerisinden atılır.

Geliştirilen sitemde "scikit-learn" kütüphanesi içerisindeki karar ağacı algoritması, "Gini" kriteriyle minimum ayrılma değeri 2 ve minimum yaprak sayısı 1 olacak şekilde tasarlanarak kullanılmıştır.

Rastgele Orman Algoritması (Random Forest Algoritması-RF), sınıflandırma veya regresyon problemleri için kullanılan denetimli bir makine öğrenmesi mimarisidir. Kullanımı oldukça kolaydır. Bu algoritma 2001 yılında Leo Breiman tarafından, "Bagging" ve "Random Subspace" yöntemlerinin birleşimi şeklinde ortaya atılmıştır. Algoritma Karar Ağaçlarını kullanarak bir karar ormanı oluşturur ve problem çözümünü bu şekilde gerçekleştirir. Bunun için rastgele bir ağaç topluluğu/orman oluşturur, işlem sırasında birden fazla Karar Ağacı en doğru sınıflandırmayı yapacak şekilde eğitilir. Çoğu zaman hiperparametre kullanımı olmadan dahi oldukça iyi sonuçlar verebilir. İçeriği karışık, eksik veya gürültülü veri setlerinde bile oldukça hızlı ve doğru sonuçlar verdiği için en çok tercih edilen yöntemler arasındadır.

K-En Yakın Komşu Algoritması (K Nearest Neighbor-KNN), hem sınıflandırma hem de regresyon problemlerinin çözümünde kullanılan denetimli öğrenme algoritmasıdır. Diğer denetimli öğrenme algoritmalarından farklı olarak eğitim aşamasına sahip değildir. Makine Öğrenmesi algoritmaları içerisinde en basit olanı olarak bilinir.

K-en yakın komşu algoritması, bir örnek sınıf grubundaki verilerin kullanılmasıyla uygulanır (Taneja vd., 2015). Hangi örnek sınıf grubuna eklenmesi gerektiğine karar verilecek olan yeni veriye en yakın komşu olma özelliği gösteren K adet veri seçilir. K değeri çoğu veri kümesi için 3 ile 10 arasında bir değer olacak şekilde seçilir. Örnek sınıf gruplarından herhangi birine dahil edilecek yeni verinin, K adet en yakın komşu özelliği gösteren veriler ile olan uzaklığı alınır. Uzaklık hesaplamaları için Öklid, Manhattan ve Minkowski fonksiyonları kullanılır (Sharma, 2019) . K-En Yakın Komşu, basit ve gürültülü eğitim verilerine karşı oldukça dayanıklı olmasından dolayı öğrenme algoritmalarının en popüler olanlarından biridir denilebilir. Bununla birlikte, bir dezavantajı da vardır. Algoritma büyük veriler için kullanıldığında mesafe hesaplamalarında tüm durumları sakladığından dolayı çok fazla bellek alanı gerektirir. Şekil 1. K-En Yakın Komşu Algoritması mimarisini göstermektedir. (Ulgen, 2017)



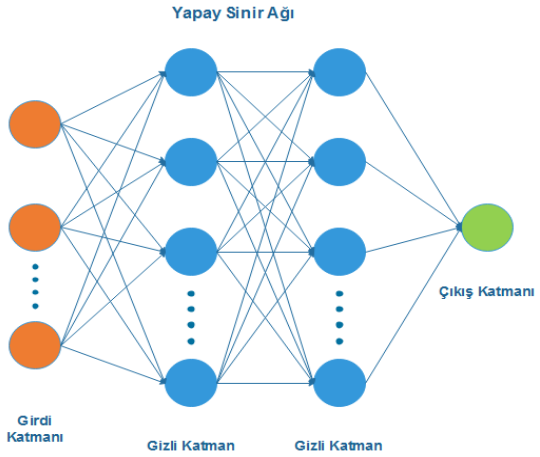
Şekil 1.K-En Yakın Komşu Algoritması Mimarisi

Adaboost Algoritması, sınıflandırma amaçlı kullanılan bir topluluk öğrenme algoritmasıdır. 1996 yılında Schapire ve Freund tarafından geliştirilmiştir. Genel olarak "Boosting" üzerine kuruludur. "Boosting" verilerden çıkan zayıf sonuçları birleştirerek güçlü bir sonuç elde edilmesi işlemidir. Her bir veriyi ilk adımda eşit şekilde dağıtır ve bir sınıflandırma yapar. Bu sınıflandırma sonucunda performansa göre en zayıf olan sınıflandırıcıyı bulur ve ağırlıkları günceller. Bu güncelleme esnasında en kötü sonucu verene odaklanır, bu sayede bir süre sonra kötü sınıflandırıcıları bir araya toplayarak başarılı bir sınıflandırıcı oluşturur. Bunu yapmaktaki amacı sınıflandırmadaki başarısını arttırmaktır. Algoritma ortaya atıldığı yıl, başarımından ötürü "Gödel" ödülünü kazanmıştır (Başar vd., 2016).

Gradyan Arttırma Algoritması (Gradient Boosting-GB), Random Forest Algoritmasının da geliştiricisi olan Leo Braiman tarafından geliştirilmiştir. Bu algoritma, regresyon ve sınıflandırma problemleri için tercih edilebilir. Adaboosting algoritmasına benzer şekilde zayıf sınıflandırma modellerinin bir araya gelmesiyle genel anlamıyla karar ağaçlarından oluşan bir model oluşturur. Buradaki Gradyan arttırmanın amacı; tahminleri öğrenme oranına göre güncelleyerek, hatanın minimum olduğu değerlere ulaşmaktır.

Doğrusal Ayrımcılık Analizi (Linear Discriminant Analysis-LDA), 1936 yılında R. A. Fischer tarafından geliştirilen bir algoritmadır (Mika vd, 1999) ve özellikle boyut sayısını azaltmak için tercih edilir. Bu sayede; hesaplamayı kolaylaştırarak, underfitting ve overfitting problemlerini azaltır. Bir sınıflandırma algoritmasından çok sınıflandırmadan önce, özellik çıkarımının zor olduğu durumlarda veri ön işleme için kullanılabilir. LDA, verileri en iyi şekilde sınıflara ayırabilmek için işlemler yapar. Sınıflandırma için sınıfların dağılımını inceleyerek ve ortalama değerleri arasındaki farklılığı bulur. Bunlar üzerinden de özellik alt uzayları oluşturur.

Yapay Sinir Ağları (Artificial Neural Network-ANN), öğrenme şekillerini çoğaltmayı amaçlayan, var olan bilgiler ile yeni bilgiler üretebilen insan beyninden esinlenmiş sistemlerdir (Baykal, Bulut ve Sahingoz, 2018). Yapay sinir ağları girdi, gizli ve çıktı olmak üzere üç çeşit katmandan oluşan bir sinir ağı modelidir. Aynı zamanda Makine Öğrenimi içerisinde de kullanılan ana modellerden biridir. Tasarlanan bu model ile insan beyninin bulması zaman zaman imkânsız olan karmaşık kalıpları sınıflandırmak, tanımak mümkün hale gelmiştir. Yapay sinir ağları tahmin yürütme, karar verme, sınıflandırma, örüntü tanıma gibi çalışmalar içerisinde sıklıkla tercih edilen bir modeldir. Şekil 2. Yapay Sinir Ağı mimarisini göstermektedir. (Artificial Neural Network, 2019)



Şekil 2.Yapay Sinir Ağı Mimarisi

4. Uygulama ve Başarımlar

Bu çalışmada, belirtilen Makine Öğrenmesi yöntemleri NSL-KDD veri setine Python programlama dili ile uygulanarak, ulaşılan başarımlar kaydedilmiştir. Uygulanan Makine Öğrenmesi yöntemlerinde KNN hariç diğer tüm algoritmalarda parametreler Python içerisinde varsayılan parametreler olarak bırakılmıştır. Bunun sebebi standart kullanım durumlarını görmektir. Sadece KNN algoritmasında sınıf sayısı 5 olarak seçilmiştir çünkü veri setinde normal ve 4 adet farklı atak türü bulunmaktadır. Ayrıca Yapay Sinir Ağları için 3 katman kullanılmış olup bunlar; 1 Girdi Katmanı, 1 Gizli Katman, 1 Çıktı Katmanıdır. Girdi Katmanı ve Gizli Katman arasında 0,3'lük bir seyreltme (dropout) işlemi 100 adım sayısında yapılmıştır. Üzerinde çalışılan bilgisayarın özellikleri Tablo 2'deki gibidir;

Tablo 2. Bilgisayar özellikleri

Donanım	Özellikleri
CPU	INTEL(R) CORE(TM) İ7-8700CPU @3.20GHZ
İşletim Sistemi	64 bit, Windows 10
RAM	16,00 GB (15,90 GB Kullanılabilir)

Tablo 3'te Makine Öğrenmesi modellerinin NSL-KDD veri seti ile çalıştırılması sonucu elde edilen doğruluk değeri başarımları ve modellerin öğrenme süreleri verilmiştir.

Tablo 3. Makine Öğrenmesi modelleri ile elde edilen doğruluk oranı ve modellerin eğitim süreleri

Model	Doğruluk Oranı (%)	Süre (sn)
KNN	99,34	8,58
DT	99,76	0,55
ANN	98,50	55,27
RF	99,85	0,61
ADA	99,88	42,70
GB	99,76	44,51
LDA	93,81	0,69

Veri setini kullanarak inceleme yapmak için K Katlamalı Çapraz Doğrulama (K-Fold Cross-Validation) kullanılmıştır. Buradaki K değeri 5 olarak şeçilmiş ve bunlara bağlı olarak Tablo 3 üzerinde bulunan doğruluk oranlarına ulaşılmıştır. Uygulanan makine öğrenmesi modelleri arasında en yüksek doğruluk oranı %99,88 ile Adaboost algoritmasına aittir. Rastgele Orman algoritması %0,03'lük bir fark ile ikinci başarılı algoritma olmuştur. Bu algoritma hem doğruluk oranı hem de süresi ile öne çıkmaktadır. Ardından %99,69 doğruluk oranı ile Karar Ağaçları ve Gradyan Arttırma algoritmaları gelmektedir. Bu algoritmalar zaman açısından incelendiğinde Karar Ağaçları çok daha hızlı bir şekilde gerçekleşmiştir. %99,34 doğruluk oranı ile KNN algoritması ve %98,50 doğruluk oranı ile Yapay Sinir Ağları algoritmaları gelmektedir. En düşük doğruluk oranına sahip model %93,81 ile Doğrusal Ayrımcılık Analizi algoritmasıdır. Makine Öğrenmesi modelleri içerisinde 0,55 saniye ile en hızlı öğrenen Karar Ağaçları; en yavaş öğrenen ise 55,27 saniye ile Yapay Sinir Ağları olmuştur. Bu bağlamda doğruluk oranları ve süreler

incelendiğinde en optimal sonucu Rastgele Orman algoritması vermektedir. Ayrıca sistemin doğruluğunu kontrol etmek için Kesinlik (Precision), Duyarlılık (Recall) ve F1-Değerlendirme (F1-Score) Denklem (1)-(3)'deki gibi hesaplanmıştır.

$$Kesinlik = \frac{\mathrm{DP}}{\mathrm{DP} + \mathrm{YP}} \quad (1)$$

$$Duyarlılık = \frac{\mathrm{DP}}{\mathrm{DP} + \mathrm{YN}} \quad (2)$$

$$F1 - Değ = \frac{\mathrm{Kesinlik} * \mathrm{Duyarlılık}}{\mathrm{Kesinlik} + \mathrm{Duyarlılık}} \quad (3)$$

olmak üzere DP; Doğru Pozitif (True Positive), YP; Yanlış Pozitif (False Positive) ve YN; Yanlış Negatif (False Negative) anlamına gelmektedir. Kesinlik ve duyarlılık formüllerinden yola çıkarak F1-Değerlendirme formülü oluşmaktadır.

Tablo 4'te çalışmanın bu değerlere bağlı sonuçları gösterilmektedir. Bu sonuçlar incelendiğinde doğruluk oranlarına bağlı olarak en iyi başarımın Adaboost olduğu görülmüştür.

Tablo 4. Makine Öğrenmesi modelleri ile elde edilen Kesinlik, Duyarlılık ve F1-Değerlendirme Oranları

Model	Doğruluk oranı (%)		
	Kesinlik	Duyarlılık	F1 Değr.
KNN	95,18	84,57	87,42
DT	91,58	89,55	90,25
ANN	77,68	69,13	72,15
RF	92,46	87,05	89,09
ADA	96,22	91,14	93,13
GB	94,70	86,48	88,94
LDA	67,77	81,72	72,47

Makine Öğrenmesi modellerinin gelen saldırı sınıflarını tahmin etme sürelerini öğrenmek amacıyla öğrenilen modellere sırasıyla 15, 50, 100 ve 1.000 adet veri yollanarak alınan sonuçlar Tablo 5'de gösterilmiştir.

Tablo 5. Makine Öğrenmesi modellerinin örnek verilerin sınıfını tahmin etme süreleri

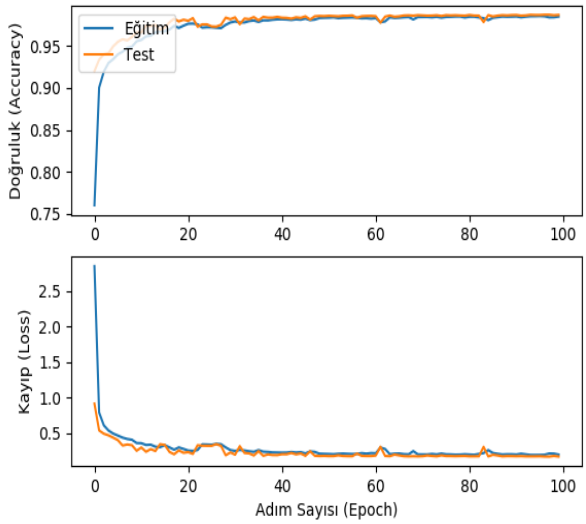
Model	Süre (milisaniye)			
	15 Veri	50 Veri	100 Veri	1000 Veri
KNN	4	10	20	186
DT	1	1	1	1
ANN	1	2	2	19
RF	1	1	1	2
ADA	6	6	8	20
GB	1	1	1	5
LDA	1	1	1	1

Her algoritma için atakların dağılımına bağlı doğruluk oranları Tablo 6 ile gösterilmiştir. Bu tablo incelendiğinde U2R atağının tespitinde oldukça düşük başarım oranı elde edildiği görülmektedir. Bunun sebebi U2R atağı tipinde çok az örnek veri bulunmasıdır.

Tablo 6. Makine Öğrenmesi modelleri ile elde edilen doğruluk oranlarının saldırı tiplerine göre dağılım

Model	Doğruluk Oranı (%)				
	Normal	DoS	R2L	Probe	U2R
KNN	99,61	99,64	97,26	94,77	32,69
DT	99,81	99,91	99,40	96,28	55,76
ANN	99,74	97,88	97,33	50,65	0,0
RF	99,94	99,94	99,53	95,07	44,23
ADA	99,94	99,95	99,70	96,98	61,53
GB	99,88	99,93	99,04	94,97	40,38
LDA	93,58	96,35	85,85	87,03	50,00

Bu algoritmalardan Yapay Sinir Ağları modeli yinelemeli (iteratif) bir algoritmadır. Bu algoritmanın çalışma performansının görülmesi açısından her adımdaki değişikliklerinde görülmesi anlam ifade etmektedir. Şekil 3'te her yineleme için değişen doğruluk ve kayıp değerleri gösterilmektedir.



Şekil 3.Yapay sinir ağı çalışımında doğruluk ve kayıp değerlerinin adımlardaki değişimi

Grafikten görüleceği gibi eğitim ve test verileri için doğruluk oranları birbirine oldukça yakındır. Belli bir noktadan sonra doğruluk oranının %99 üzerinde kaldığı görülmektedir bu da bize bu tip veri setleri için çok uzun adım sayılarına ihtiyaç duymadan kısa sürede eğitim yapabileceğimizi göstermektedir.

Farklı saldırı tiplerinin olmasından dolayı hangi saldırı tipinin daha yüksek doğruluk ile yakalandığının belirlenebilmesi açısından bir de Karmaşıklık Matrisi (Confusion Matrix) karşılaştırması yapılmıştır. Bu algoritmalardan yaygın olarak kullanılan Yapay Sinir Ağının Karmaşıklık Matrisi Tablo 7'de gösterilmektedir.

Tablo 7. Yapay Sinir Ağları Karmaşıklık Matrisi

	Nor.	Dos	R2L	Prb.	U2R
Nor.	67190	21	76	945	1
Dos	963	44958	3	0	0
R2L	4236	3	11377	0	0
Prb	494	3	0	497	1
U2R	50	0	2	0	0

Bu tablodan da görüleceği üzere YSA yaklaşımı her ne kadar yüksek başarım gösterse de U2R saldırılarının tespitinde başarısız olmaktadır. Bu durum özellikle kullanılan eğitim ve test veri seti sayısından kaynaklanmaktadır.

Ancak diğer makine öğrenme algoritmalarının özellikle U2R tipli saldırılara karşı daha yüksek başarımları olduğu görülmektedir. Genel başarım oranları açısından ayırt edici olan Karar Ağaçları ve Adaboost algoritmalarının Karmaşıklık Matrisleri sıralı olarak Tablo 8 ve Tablo 9'da sunulmaktadır.

Tablo 8. Karar Ağaçları Karmaşıklık Matrisi

	Nor.	Dos	R2L	Prb	U2R
Nor.	67223	29	43	37	10
Dos	38	45885	4	0	0
R2L	51	12	11592	0	1
Prb	35	1	1	957	1
U2R	16	1	1	1	33

Tablo 9. Adaboost Karmaşıklık Matrisi

	Nor.	Dos	R2L	Prb	U2R
Nor.	67303	11	19	4	5
Dos	20	45907	0	0	0
R2L	34	1	11621	0	0
Prb	32	0	0	960	3
U2R	18	0	0	0	34

5. Tartışma

Geliştirilen algoritmaların performansları incelendiğinde, yüksek doğruluk oranlarına ulaşıldığı görülmektedir. Ancak kullanılan veri setindeki bazı tipten verilerin örnek sayısının az olmasının ilgili kaydın sınıflandırılmasında düşün başarıma ulaştığı görülmektedir. Örneğin U2R ve Probe tipli saldırıların tespit oranı gerçeklenen tüm algoritmalarda diğer sınıflardan daha düşük çıktığı görülmektedir. Ancak ilgili tipten kayıtların test setinde de az miktarda

bulunmasından dolayı toplam verim konusunda fazla bir etkisi bulunmadığı değerlendirilmektedir.

Aynı zamanda geliştirilen sistem benzer şekilde yapılmış olan 2 çalışma ile karşılaştırılmıştır. Belavagi ve arkadaşları yapmış oldukları çalışmada 4 farklı algoritmayı test ederek Tablo 10'da görüleceği üzere en yüksek %99,00 doğruluk oranına ulaştıkları görülmektedir (Belavagi vd., 2016).

Tablo 10. Sistemin Başarım Ölçüleri

Model	Doğruluk oranı (%)			
	Kesinlik	Duyarlılık	F1 Değr.	Doğr. Oranı
LR	83,00	85,00	82,00	84,00
GNB	79,00	81,00	78,00	79,00
SVM	76,00	79,00	77,00	75,00
RFC	99,00	99,00	99,00	99,00

Choudhury ve arkadaşlarının yaptıkları çalışmada ise 9 farklı algoritmayı test ederek Tablo 11'de görüleceği üzere en yüksek % 91,52 lik doğruluk oranına ulaştıkları görülmektedir (Choudhury vd., 2015)

Tablo 11. Sistemin Başarım Ölçüleri

Model	Doğruluk oranı (%)			
	Kesinlik	Duyarlılık	F1 Değr.	Doğr. Oranı
Bayes Net	96,20	85,50	90,70	90,66
Logistic	84,80	87,34	86,00	84,96
IBk	93,60	88,68	91,00	90,73
JRip	91,50	84,47	87,80	87,54
PART	93,80	85,32	89,30	89,21
J48	93,60	86,48	89,90	89,67
Random Forest	95,10	88,68	91,70	91,52
Random Tree	94,70	87,74	91,10	90,87
REPTree	93,30	79,78	86,00	86,21

Aynı başarım parametrelerini bizim geliştirdiğimiz sistem ile karşılaştırdığımız zaman, Tablo 12'de görüldüğü üzere, özellikle doğruluk oranı kapsamında diğer çalışmalara göre daha iyi bir performans ortaya koyduğumuz görülmektedir.

Tablo 12. Geliştirilen Sistemin Başarım Ölçüleri

Model	Doğruluk oranı (%)			
	Kesinlik	Duyarlılık	F1 Değr.	Doğr. Oranı
KNN	95,18	84,57	87,42	99,34
DT	91,58	89,55	90,25	99,76
ANN	77,68	69,13	72,15	98,50
RF	92,46	87,05	89,09	99,85
ADA	96,22	91,14	93,13	99,88
GB	94,70	86,48	88,94	99,76
LDA	67,77	81,72	72,47	93,81

6. Sonuç ve Gelecek Çalışmalar

Bu makalede, farklı Makine Öğrenmesi algoritmaları kullanarak Saldırı Tespit Sistemleri üzerindeki etkilerini incelenmiştir. Sistemin temel amacı, sık kullanılan makine öğrenmesi algoritması olan Karar Ağacı, Rastgele Orman, K-En Yakın Komşu, Adaboost, Gradyan Arttırma, Doğrusal Ayrımcılık Analizi, Yapay Sinir Ağları algoritmalarının ayrı ayrı kıyaslayarak, ağ saldırılarını tespit etmelerindeki başarımlarını karşılaştırmalı olarak ölçmektir. Sistemin performansını değerlendirmek için literatürde yoğun kabul gören NSL-KDD veri seti kullanılmıştır. Bu veri seti hem yaygın kullanılması hem de KDDCup99 veri setindeki tekrarlanan kayıtların çıkarılması ve toplan veri miktarının azaltılmış olması sebebi ile tercih edilmiştir.

Sistemin temel başarım ölçütü olarak doğruluk oranı kabul edilmiş ve rasgele seçim etkisini azaltmak için 5-fold tekniği kullanılarak 5 ayrı test yapılarak ortalama değerler alınmıştır. Her bir modelin doğruluk değerleri, eğitim süreleri ve test süreleri ölçülmüştür ayrıca test aşamasında 15, 50, 100 ve 1000 adet veri yollanarak kontrol süresi sonuçları alınmıştır. Bu doğruluk

oranları ve süreler doğrultusunda neredeyse her algoritmanın %99'un üzerinde bir başarım elde ettiği görülmektedir. Ayrıca mimarileri kendi içerisinde incelediğimiz zaman en yüksek doğruluk oranına Adaboost algoritmasının %99,88 doğruluk oranına ulaştığı görülmektedir. Ancak aynı algoritmanın 42,70 sn gibi alternatiflerine göre daha uzun bir eğitim süresi olduğu görülmektedir. Eğitim Süresi ve çalışma süresi açısından incelendiğinde ise Karar Ağacı algoritmasının diğer modellerden daha hızlı eğitim sonucu verdiği görülmektedir. Doğruluk oranı açısından Adaboost algoritması ile ciddi bir fark olmamasından dolayı sistem geliştiriminde tercih edilebileceği değerlendirilmektedir.

Gelecek çalışmalarda güncel bir veri seti kullanılarak mimariler kıyaslanacak ayrıca Derin Öğrenme yaklaşımları ile karşılaştırılması ve sistemin hızlı eğitilmesi açısından paralel bir mimari olan GPU altyapısı üzerinden çalıştırılması planlanmaktadır.

7. Kaynaklar

Ahmad, I., Basheri, M., Iqbal, M. J. and Rahim, A. 2018. "Performance Comparison of Support Vector Machine, Random Forest, and Extreme Learning Machine for Intrusion Detection," in IEEE Access, 6, 33789-33795.

Alhakami, W., Alharbi A., Bourouis S., Alroobaea R. and Bouguila N. 2019. "Network Anomaly Intrusion Detection Using a Nonparametric Bayesian Approach and Feature Selection," in IEEE Access, 7, 52181-52190.

Artificial neural network. 2019. URL: https://en.wikipedia.org/wiki/Artificial_neural_network.

Athmaja, S., Hanumanthappa, M. and Kavitha, V. 2017. "A survey of machine learning algorithms for big data analytics," 2017 International Conference on Innovations in Information, Embedded and Communication Systems (ICIIECS), Coimbatore, 1-4.

Aziz, A. S. A., Hassanien, A. E. 2014. "Multilayer Machine Learning-Based Intrusion Detection System", In Bio-inspiring Cyber Security and Cloud Services: Trends and Innovations, 225-247.

Başar, M. D., Sarı, P., Kılıç, N., and Akan, A. 2016. "Detection of chronic kidney disease by using Adaboost ensemble learning approach", In 2016 24th Signal Processing and Communication Application Conference (SIU), 773-776.

Baykal, S. I., Bulut, D., Sahingoz, O. K. 2018. "Comparing deep learning performance on bigdata by using CPUs and GPUs", In 2018 Electric Electronics, Computer Science, Biomedical Engineerings' Meeting (EBBT), 1-6.

Belavagi, M. C., and Muniyal, B. 2016. "Performance evaluation of supervised machine learning algorithms for intrusion detection", Procedia Computer Science, 89, 117-123.

Choudhury, S., and Bhowal, A. 2015. "Comparative analysis of machine learning algorithms along with classifiers for network intrusion detection", In 2015 International Conference on Smart Technologies and Management for Computing, Communication, Controls, Energy and Materials (ICSTM), 89-95.

CUP-99 Task Description. (n.d.). URL: https://kdd.ics.uci.edu/databases/kddcup99/task.html

Dhanabal, L., Shantharajah, S. P. 2015. "A study on NSL-KDD dataset for intrusion detection system based on classification algorithms", International Journal of Advanced Research in Computer and Communication Engineering, 4(6), 446-452.

Dong, B., Wang, X. 2016. "Comparison deep learning method to traditional methods using for network intrusion detection", In 2016 8th

IEEE International Conference on Communication Software and Networks (ICCSN), 581-585.

Elsaeidy, A., Munasinghe, K. S., Sharma, D., and Jamalipour, A. 2019. “Intrusion detection in smart cities using Restricted Boltzmann Machines”, Journal of Network and Computer Applications.

Gavankar, S. S., Sawarkar, S. D. 2017. “Eager decision tree”, In 2017 2nd International Conference for Convergence in Technology (I2CT), 837-840.

Karatas, G., Demir, O., Sahingoz, O. K. 2018. “Deep Learning in Intrusion Detection Systems”, In 2018 International Congress on Big Data, Deep Learning and Fighting Cyber Terrorism (IBIGDELFT), 113-116.

Karatas, G., Sahingoz, O. K. 2018. “Neural network based intrusion detection systems with different training functions”, In 2018 6th International Symposium on Digital Forensic and Security (ISDFS), 1-6.

Mika, S., Ratsch, G., Weston, J., Scholkopf, B., and Mullers, K. R. 1999. “Fisher discriminant analysis with kernels”, In Neural networks for signal processing IX: Proceedings of the 1999 IEEE signal processing society workshop, 41-48.

Radoglou-Grammatikis, P. I. and Sarigiannidis, P. G. 2019. "Securing the Smart Grid: A Comprehensive Compilation of Intrusion Detection and Prevention Systems," in IEEE Access, 7, 46595-46620.

Ruiz-Vanoye, J. A., Diaz-Parra, O., Penna, A. F., Castro, J. O. C. and Olivares-Rojas, J. C. 2007. "An Alternative Solution Initiative to Problematic of Computer Science Security of Virus and Malware with Experimentation of Firewalls and Antivirus," 2007 International Multi-Conference on Computing in the Global Information Technology (ICCGI'07), Guadeloupe City, 34-34.

Sahingoz, O. K., Buber, E., Demir, O., Diri, B. 2019. “Machine learning based phishing detection from URLs”, Expert Systems with Applications, 117, 345-357.

Salo, F., Injadat, M., Nassif, A. B., Shami, A. and Essex A. 2018. "Data Mining Techniques in Intrusion Detection Systems: A Systematic Literature Review," in IEEE Access, 6, 56046-56058.

Sharma, N. 2019. “Importance of Distance Metrics in Machine Learning Modelling” https://towardsdatascience.com/importance-of-distance-metrics-in-machine-learning-modelling-e51395ffe60d

Taneja, S., Gupta, C., Aggarwal, S., Jindal, V. 2015.” MFZ-KNN—A modified fuzzy based K nearest neighbor algorithm”, In 2015 International Conference on Cognitive Computing and Information Processing (CCIP), 1-5.

Thomas, R. and Pavithran, D. 2018. "A Survey of Intrusion Detection Models based on NSL-KDD Data Set," 2018 Fifth HCT Information Technology Trends (ITT), Dubai, United Arab Emirates, 286-291.

Ulgen, K. 2017. Makine Öğrenimi Bölüm-2 (k-En Yakın Komşuluk). URL: https://medium.com/@k.ulgen90/makine-öğrenimi-bölüm-2-6d6d120a18e1

Zhang, Y., Li, P. and Wang, X. 2019. "Intrusion Detection for IoT Based on Improved Genetic Algorithm and Deep Belief Network," in IEEE Access, 7, 31711-31722.