Yüzüncü Yıl Üniversitesi Fen Bilimleri Enstitüsü Dergisi/ Journal of the Institute of Natural & Applied Sciences 23 (3): 297-307, 2018

Araştırma Makalesi / Research Article

Geliş Tarihi: 25.09.2018
Kabul Tarihi: 27.11.2018

Sınıflandırma ve Regresyon Ağaçları ile Rastgele Orman Algoritması Kullanarak Botnet Tespiti: Van Yüzüncü Yıl Üniversitesi Örneği

Duygu Kormaz, H. Eray Çelik*, Mesut Kapar

Van Yüzüncü Yıl Üniversitesi, Bilgisayar Bilimleri Araştırma ve Uygulama Merkez Müdürlüğü
*Sorumlu yazar e-posta: ecelik@yyu.edu.tr

Öz: Bir botnet, kötü amaçlı yazılım kodunun bulaşmış olduğu, bir veya daha fazla makineden oluşan bir ağdır. Botnet, Botmaster denilen kişiler tarafından yönetilir ve DDos, Spam, Kimlik Hırsızlığı gibi faaliyetler için kullanılmaktadır. Bu çalışmanın amacı, bir Network üzerinde botnet bulaşmış network cihazı olup olmadığını, Makine Öğrenmesi Algoritmalarından, Sınıflandırma Ağaçları ve Regresyon Ağacı (CART) ile Rastgele Orman teknikleriyle tespit etmek ve sınıflandırmaktır. Modellerin sınıflandırma performansları bazı performans ölçütleri bakımından ölçülmüş ve kıyaslanmıştır. Ele alınan değişkenler, ekleyip çıkarılarak doğruluk ve bazı performans ölçütleri üzerindeki değişimler Sınıflandırma Ağaçları Yöntemi ve Rastgele Orman Algoritması Yöntemi ile incelenmiştir ve bir ağda Botnet tespiti yapmak için önemli olan değişkenler önerilmiştir.
Anahtar kelimeler: Düğüm, Gini, Hata matrisi, Phyton, Siber güvenlik

Botnet Detection by Using Classification and Regression Trees with Random Forest Algorithms: Example of Van Yüzüncü Yıl University

Abstract: A botnet is a network of malware code infected by one or more machines. Botnet is managed by Botmaster and is used for activities such as Ddos, Spam, and Identity Theft. The purpose of this study is to identify and classify whether or not there is a network device infected by a botnet on a network using Classification and Regression Trees and Random Forest techniques from Machine Learning Algorithms. Classification Performance of Models are measured and compared in terms of some performance measures. Variations on accuracy and some performance measures were examined on Classification and Regression Trees and Random Forest techniques, by adding and subtracting variables and this study suggests variables that are important for Botnet Detection in a Network.
Keywords: Confusion matrix, Cyber Security, Gini, Node, Phyton

Giriş

Teknolojik gelişmeler yaşam kalitesini yükseltmekle beraber, bilişim alanındaki gelişmeler; yeni tip suçların ortaya çıkmasına sebep olmuştur. Bu yeni tip suçlara siber suç adı verilmektedir (Shinder ve Tittel, 2002).

Siber, bilgisayar ve bilgisayar ağlarını ilgilendiren veya içeren kavram ya da varlıkları tanımlamak için kullanılan bir kelimedir. Siber suçlar temel anlamıyla, bilgisayar sistemleri üzerinden yasadışı bir şekilde bilgi alınması ve bu sistemlere izinsiz erişim sağlanmasıdır (Shinder ve Tittel, 2002).

Sosyal Medya ve Mobil Kullanıcı İstatistikleri'ne göre dünya nüfusunun % 53'ü aktif bir şekilde internet kullanmaktadır (Anonim, 2018a). Her geçen yıl bu oran artmakta ve buna bağlı olarak siber suçların da oranı ve bu suçlara maruz kalan kullanıcılar artmaktadır.

Türkiye, dünya genelinde en çok siber saldırı alan ülkelerden biri konumundadır. Avrupada en fazla siber saldırının yaşandığı ülke Türkiye iken, dünya genelinde ABD ve Brezilya'dan sonra üçüncü sırada yer almaktadır. Bu saldırıların çoğu bilgisayar, akıllı telefon veya diğer ağ cihazlarından botnet yoluyla gerçekleştirilmektedir. Fortinet şirketi tarafından yapılan araştırmanın sonuçlarına göre botnet, exploit kit ve fidye yazılım tehdidi altındaki dünya ülkeleri arasında Türkiye beşinci sırada yer almaktadır (Anonim, 2018b).

Botnet saldırıları, spam, Dağıtık Hizmet Reddi Saldırıları (DDos), kimlik hırsızlığı ve kimlik avı gibi bir çok internet saldırısının temel platformudur (Gu ve ark., 2008).

Bir botnet, kötü amaçlı yazılım (bot) kodunun bulaşmış olduğu, bir veya daha fazla makineden oluşan bir ağdır. Botnetler, botmaster denilen kişiler tarafından yönetilir ve DDos, Spam, Phishing, Kimlik Hırsızlığı gibi faaliyetler için kullanılırlar (Gu ve ark., 2008).

Botnet saldırıları, yasal iletişim kanalları aracılığıyla gerçekleştirilir. Internet Relay Chat (IRC) uygulamaları 2000'lerin başlarına kadar geleneksel botnetler arasında en yaygın iletişim yolu olarak kullanılmaktadır. Ağ içerisindeki makinelere botnet yazılımları bulaştıktan sonra, botnetler bir IRC sunucusuyla bağlantı kurar ve bu kötü amaçlı yazılım Botmaster tarafından yerleşik IRC komutları ve kontrol (C&C) kanallarını kullanır. Botmasterın amacı, bulaştırılan zararlı yazılımın (botların) sistemde sürekliliğini sağlamaktır. Botlar güncelleme almak için belirli aralıkla bot yöneticisine bağlanır ve botlar merkezi olarak yönetildikleri için, bot yöneticisinin kapatılması durumunda devre dışı kalmaktadırlar.

Ağ trafiğinin izlenebilmesiyle birlikte botnet tespiti konusunda daha fazla araştırma yapılmıştır. 2000'li yıllardan itibaren Peer to Peer (p2p) protokolünün gelişmesi ile bot yazılımları istemci ve sunucu arasında hareket eder ve merkezi bir nokta olmadığı için çalışması sekteye uğramaz. P2P botnetleri yönetilebilirlik açısından ve sınırlandırmalarının olmasından ötürü HTTP tabanlı C &C protokolüyle uygulanmaya başlanmıştır. HTTP tabanlı botnetlerin, güvenlik duvarı ve benzeri sistemlere takılmaması için botnet paketleri şifrelenerek bu güvenlik duvarlarının geçilmesi sağlanmaktadır (Zhao ve ark., 2013).

Bu çalışmanın amacı, bir ağ üzerinde Botnet bulaşmış network cihazı olup olmadığını, sanal ortamda belli bir zaman diliminde botnet bulaştırılmış ve bulaştırılmamış ağ trafiği akışı birlikte izlenerek toplanan veriler üzerinde, Makine Öğrenmesi Algoritmalarından Sınıflandırma ve Regresyon Ağacı (CART) ile Rastgele Orman teknikleri kullanarak botnetli ve normal akışı tespit etmek, bu iki akışı ayırt eden bir sınıflandırma modeli oluşturmak ve modele ait performans ölçülerini hesaplamaktır.

Materyal ve Yöntem

Materyal

Botnet trafiği için gerçek network kullanmak networke zarar verebileceğinden, sanal ortamda bir network kurarak (demo ortamında) paket toplaması yapılmıştır. Bu paketlere ek olarak internet üzerinden hazır botnet paketleri de indirilip, diğer botnet paketlerine eklenmiştir. Sanal ortamın kurulması için Microsoft Windows 10 üzerinde kurulu gelen Hyper-v sanallaştırma ortamı kullanılmıştır. Bu sanal ortamda toplamda dört adet bilgisayar bulunmaktadır. Bu bilgisayarlardan üçü botnetli akışa sahipken biri Van Yüzüncü Yıl Üniversitesine ait normal ağ akışına sahiptir.

Network üzerindeki anormal aktiviteleri incelemek için network trafiğinin sniff (süzülmesi) edilmesi gerekmektedir. Network üzerinden paket toplamak için ve paket üzerinden filtrelemeler yapabilmek için Wireshark adlı network sniffing aracı kullanılmıştır. Bunun için ağ katmanı yani layer 2' de çalışılmıştır Sanal ortamda bulunan bilgisayarlara botnet malwareler

bulaştırılıp, bu dört trafik arasındaki trafik 40 bin veri akışı toplanana kadar izlenmiştir. Normal trafik verisinin akışı için ise botnetsiz olan üniversiteye ait normal trafik akışı izlenmiştir.

Toplamda 40 bin botnet trafik paketi, 40 bin normal trafik paketi toplanıp makine öğrenmesi algoritmalarıyla sınıflandırılması gerçekleştirilmiştir. Ancak 1 gözlem botnet sınıfına, 1 gözlem de normal akış sınıfına ait olmak üzere, toplam iki gözlem isimleri dışında değişken içermediğinden analizden çıkarılmıştır ve analiz 79998 gözlem üzerinden yapılmıştır. Veri hazırlama aşamasında literatürde önemli olduğu düşünülen özellikler PHP yazılım dili yardımı ile MYSQL veri tabanına aktarılarak veriler burada düzenlenip (sayısal hale getirilip) Excell formatına çevrilmiştir. Uygulama Phyton'da gerçekleştirilmiştir. Uygulamaya ait değişkenler ve elde edilen bulgular bu bölümde verilmiştir.

Çalışmada Kullanılan Nitelikler: Özellik seçimi, sınıflandırıcı oluşturmada çok önemli bir rol oynamaktadır. Varolan veri kümesi, başlangıç zamanı, kaynak ve hedef bağlantı noktası belirtme sırasında kullanılan süre, ağ trafiğinde yer alan paket için kullanılan kaynak ve hedef IP adresi, varlıklar arasındaki etkileşimi belirleyen protokol, toplam bayt gibi temel özellikleri içerir. Bu özellikler, normal trafikten botnet trafiğini ayırt etmek için yeterli değildir. Bu nedenle, yüksek sınıflandırma doğruluğuna ulaşmak için botnet ve normal trafiğin özelliklerini açıklayan özellikler seçilir. Dolayısıyla, ortalama bayt oranı, ortalama paket hızı, paket boyutu gibi botnet trafiğini belirlemede önemli olan özellikler seçilmiştir (Kalaivani ve Vijaya, 2016).

Veri setinde yer alan özellikler ve onların sayısallaştırma işlemlerine değinilmiş ve analize ilişkin bulgular sunulmuştur.

Ele alınan özellikler; Devam Süresi, Protokol Tipi, Kaynak ve Hedef IP, Kaynak ve Hedef Port, Syn Bayrak Durumu, Reset Bayrak Durumu, Ack Bayrak Durumu, Toplam Paket, Reset Bağlantı Sayısı, Ortalama Bayt, Ortalama Paket Oranı, Paket Boyutu ve Botnet'tir.

Devam süresi: (Akışın süresi), akışı tamamlamak için alınan toplam süreyi gösterir. Bu süre, ortalama paket oranını ve ortalama bayt oranını hesaplamak için kullanılır (Kalaivani ve Vijaya, 2016).

Protokol: iletişim kurduklarında telekomünikasyon bağlantı kullanımındaki noktaları sonlandıran özel kurallar kümesidir. Protokoller, iletişim kurucu varlıklar arasındaki etkileşimleri belirler.

Kullanılan farklı protokol türleri vardır (Kalaivani ve Vijaya,2016). Bu çalışmada yer alan protokoller; TCP, UDP, ICMPV6, IGMP ,DATA, IPV6,.HOPOPTSI ve ICMP'dir.

Kaynak IP: İnternet akış alanlarının temel özelliklerinden biridir. Kaynak IP adresi kullanıcı bilgisayarının IP adresidir (Karasaridis ve ark.,2007). Kaynak IP adresi, daha fazla işlem için ondalık formata PHP yazılım dilinde IP2LONG (IP adreslerini sayısal değerlere dönüştürüp veritabanına kayıt işlemlerini yapan fonksiyondur) fonsiyonu kullanılarak dönüştürülmektedir.

Hedef IP: bir mesajın gönderildiği IP adresidir. IP adresleri, bir ağ üzerinden veri paketlerini iletmek için kullanılır (Karasaridis ve ark., 2007; Kalaivani ve Vijaya, 2016). Hedef IP adresi, daha fazla işlem için ondalık formata için ondalık formata Php yazılım dilinde IP2LONG fonksiyonu kullanılarak dönüştürüldü.

Kaynak ve hedef port: Port numarası, verdiğimiz hizmeti veya uygulamayı tanımlamamıza, isteğimizin gönderilmesine ve yapılmasına izin veren önceden belirlenmiş numaradır. Port numaraları saldırıları amaçlayan uzak sistemlere ilişkin bilgi edinmek için

kullanılabilirler. 80, 53, 25 numaralı bağlantı noktası, farklı botnet saldırıları türlerine sahip kötü amaçlı akış olarak işaretlenir; bunlar sırasıyla, HTTP tabanlı botnet, spam botnet ve DNS sunucu tabanlı botnettir (Karasaridis ve ark., 2007; Kalaivani ve Vijaya, 2016).

Bayraklar: Ağ akışını temsil eden SYN, RST, CON, ACK, FIN olmak üzere farklı bayrak türleri vardır. Bunların her biri farklı bir durumu temsil eder (Kalaivani ve Vijaya, 2016).

Toplam paket: Toplam paket özelliği, belirli bir akış sırasında aktarılan paketlerin sayısı olarak anlamına gelir. Belirli bir süre veya akışta iletilen paketlerin sayısı kaydedilir (Karasaridis ve ark., 2007).

Reset bağlantı sayısı: Bir akışın reset bağlantısı sayısı, sunucunun bağlantı yapmayı reddetmesidir. Reset bağlantısının sayısı, aynı IP adresinden tekrarlanan RST bağlantısını analiz ederek hesaplanır. Çok fazla RST bağlantısının alınması, alıcının enfekte olduğu anlamına gelmektedir (Kalaivani ve Vijaya, 2016).

Ortalama bayt: Ortalama bayt oranı, toplam bayt ve süre gibi var olan özellikler aracılığı ile hesaplanır.Ortalama bayt,belirli bir sürede akış içinde aktarılan ortalama baytları ifade eder (Karasaridis ve ark., 2007; Kalaivani ve Vijaya, 2016).

$$\text{Ortalama Bayt} = \frac{\text{Toplam bayt}}{\text{Devam Süresi}}$$

Burada toplam bayt: İstemcinin istek dâhilinde gönderdiği toplam bayt sayısı anlamına gelir (Kalaivani ve Vijaya, 2016).

Ortalama paket oranı: Belirli bir zaman aralığındaki akışta aktarılan ortalama paket sayısıdır (Kalaivani ve Vijaya, 2016; Chen ve ark., 2017).

$$\text{Ortalama Paket Oranı} = \frac{\text{Toplam paket}}{\text{Devam Süresi}}$$

Paket boyutu: Belirli bir zaman aralığındaki akışta aktarılan paketin bayt cinsinden boyutudur (Kalaivani ve Vijaya, 2016; Chen ve ark., 2017).

Yöntem

Sınıflandırma ve regresyon ağaçları (CART)

Bir sınıflandırma çalışmasında amaç, probleme bağlı olarak doğru bir sınıflandırıcı üretmek veya problemin yapısına uygun kestirimci bir yapı ortaya çıkarmaktır. Eğer araştırmacı; kestirimci bir model bulma çabasındaysa, veri setindeki değişkenlerden hangisinin veya hangilerinin olayı basit bir şekilde karakterize ettiğini ve diğer sınıflardan ayırdığını anlamaya çalışmalıdır. Problemin türüne göre hem veriyi anlamak hem de kestirimci bir model elde etmek amaçlanabilir ve bazen bu amaçlardan biri diğerine göre daha önemli hale gelebilir (Bock, 2002; Breiman ve ark., 2017).

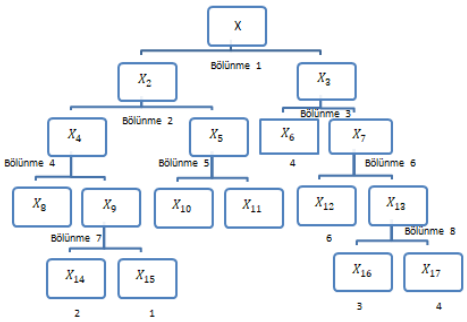
Bu amaca uygun bir yöntem olan Sınıflandırma ve Regresyon ağaçları Morgan ve Sonquist'in AID (Automatic Interaction Detection) adlı karar ağacı algoritmasının devamı niteliğinde olup Breiman tarafından 1984 yılında önerilmiştir (Breiman ve ark., 2017).

Sınıflandırma ve Regresyon Ağaçları (CART) yöntemi, çoklu bağlantı, sapkın gözlem, kayıp verinin etkilerine büyük direncinden ve modelde hangi değişkenin içerilmesi gerektiğinin belirlenmesinden önce kestirimciler/ tahminciler arasında yüksek seviyeli interaksiyonları tanımlamaya yeteneğinden dolayı parametrik yaklaşımlara tercih edilebilir (De'ath ve Fabricius, 2000).

Sınıflandırma ve Regresyon Ağaçları Metodolojisi 3 kısımdan oluşmaktadır: maksimum ağacın oluşturulması, uygun ağaç genişliğinin seçimi, oluşturulan ağaçtan hareketle yeni

verilerin sınıflandırılması (Timofeev, 2004).

Ağaç yapılı sınıflandırıcılar: Ağaç yapılı sınıflandırıcılar ya da daha doğru bir şekilde ikili ağaç yapılı sınıflandırıcıların bir X kümesini, kendisinden başlayarak tekrarlı bir şekilde X'in alt kümelere bölünmesi ile inşa edilir (Loh, 2011; De'ath ve Fabricius, 2000). Bu süreç altı sınıflı bir



ağaç için Şekil 1'de gösterilmiştir.

Şekil 1. Sınıflandırma ağacının yapısı

Şekilde X_2 ve X_3 ayrıktır .$X = X_2 \cup X_3$ olup benzer şekilde X_4 ve X_5 de ayrıktır ve $X_2 = X_4 \cup X_5$ ve $X_3 = X_6 \cup X_7$ olur.

Bilinmeyen alt kümeler $X_6, X_8, X_{10}, X_{11}, X_{12}, X_{14}, X_{15}, X_{16}$ ve X_{17} terminal alt kümeler olarak adlandırılır. Şekilde belirtilen terminal düğümler dikdörtgen ile, terminal olmayanlar ise dairelerle gösterilmiştir (Loh, 2011; Breiman ve ark., 2017).

Terminal alt kümeler X'in bir bölünmesini temsil eder. Her bir terminal alt kümesi bir sınıf etiketi ile gösterilir. Aynı sınıf etiketine sahip iki ya da daha fazla terminal alt kümeler olabilir. Sınıflandırıcıya karşılık gelen bölünme aynı sınıfa karşılık gelen tüm terminal alt kümelerin birleştirilmesiyle elde edilir. Yani, $A_1 = X_{15}$,$A_2 = X_{11} \cup X_{14}$, ,$A_3 = X_{10} \cup X_{16}$, ,$A_4 = X_6 \cup X_{17}$, $A_5 = X_8$, $A_6 = X_{12}$'dir. Bölünmeler $X = (X_1, X_2 \ldots)$ noktasının koordinatlarının koşulları ile oluşturulur. Örneğin, X'in X_2 ve X_3'e bölünmesi;

$X_2 = \{x;\ x_4 \leq 7\}, X_3 = \{x;\ x_4 > 7\}$

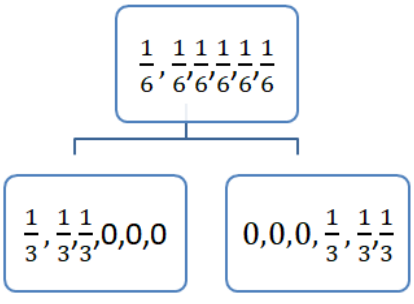
şeklinde olabilir (Loh 2011; Breiman ve ark., 2017).

Ağaç sınıflandırıcısı x ölçüm vektörü için bir sınıfı şöyle tahmin eder: İlk bölünmenin tanımından x'in X_2'ye ya da X_3'e gidip gitmeyeceği belirlenir. Örneğin (Bkz. 4.7) $X_4 \leq 7$ ise x, X_2 kümesine, eğer $X_4 > 7$ ise X_3 kümesine gider. x terminal kümeye ulaştığında, x'in sınıfı gittiği terminal alt sınıfın kümesi olarak tahmin edilir.

Teorik olarak X'in bir alt kümesine t düğümü, X'in kendisine t_1 kök düğümü, terminal alt kümelere terminal düğümler, terminal olmayan alt kümelere de terminal olmayan düğümler denir.

Ağaç sınıflandırıcı inşası: Ağaç inşasında ilk problem L verisinin X'in ikili bölünmelerini sürdürerek daha küçük parçalara bölmek için nasıl kullanılacağıdır. Temel fikir bir alt kümenin her bir bölünmesini aşağıdaki alt kümeleri (child), yukardaki (parent) alt kümelere göre daha saf olacak şekilde seçebilmektir (Guttman,1984).

Örneğin altı sınıflı bir gemi probleminde, herhangi bir düğümde $p_1, p_2, \ldots, p_6$ ile herhangi bir düğümde sınıf $1,2,\ldots 6$' dan birine eşit olma oranlarını gösterelim. t_1 kök düğüm için; $p_1, p_2 \ldots p_6 = \left(\frac{1}{6}, \frac{1}{6} \ldots \frac{1}{6}\right)$ 'dır. t_1'in iyi bir bölünmesi ile sınıf $1,2,3$ 'e sahip gemilerin sol düğüme, sınıf $4,5,6$ ' ya sahip gemilerin sağ düğüme ayrılması olabilir.



Şekil 2. Belirli bir sınıfa ait olma oranları

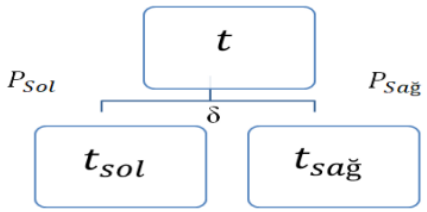
t_1'in iyi bir bölünmesi bulunduğunda, arama t_2 ve t_3'ün iyi bir bölünmesini bulmak ile devam eder. Düğümlerin daha saf düğümler oluşturması için bölünmesi fikri şu şekilde uygulanır (Breiman ve ark., 2017); $p(j\backslash t)\,, j = 1{,}2\,\ldots,6$ düğüm oranları, $x_n \in t$, j sınıfına dahil olma oranı $p\left(\frac{1}{t}\right) + \cdots p\left(\frac{6}{t}\right) = 1$ olacak şekilde tanımlanır.

t'nin safsızlık ölçüsü $i(t)$, $\mathrm{p}\left(\frac{1}{\mathrm{t}}\right) + \cdots \mathrm{p}\left(\frac{6}{\mathrm{t}}\right)$'nin negatif olmayan bir θ fonksiyonu olarak tanımlanır. Bu θ fonksiyonu:

- $\theta\left(\frac{1}{6}, \frac{1}{6}, \ldots, \frac{1}{6}\right) = max,$
- $\theta(1{,}0, \ldots, 0) = 0,$
- $\theta(0{,}0, \ldots, 1) = 0$ şartlarını sağlar.

Yani, düğüm safsızlığı tüm sınıflar bir düğümde eşit şekilde yer aldığında maksimum, bir düğüm sadece bir sınıf içerdiğinde minimumdur (Alpaydın, 2014).

Herhangi bir t düğümü için varsayalım ki bu düğümü, durumların $p_{sağ}$ oranında $t_{sağ}$'a p_{sol} oranında t_{sol}'a ayıran bir δ aday bölünmesi olsun;



Şekil 3. Aday Bölünme

Bu durumda bu bölünmelerin iyiliği safsızlıktaki azalma olarak tanımlanır ve şöyle verilir;

$$\Delta \mathrm{i}(\delta, \mathrm{t}) = \mathrm{i}(\mathrm{t}) - \mathrm{p}_{\mathrm{sol}}\mathrm{i}(t_{sol}) - \mathrm{p}_{\mathrm{sağ}}\mathrm{i}(t_{sağ}) \quad (1)$$

Son olarak, her düğümde δ ikili bölünmelerin bir aday seti S tanımlanır. Genellikle,ϑ sorular kümesinde her sorunun $x \in A,\ (A \subset X)$ şeklindeki sorularla üretilerek bölünmelerin S setini kavramak kolaydır. Daha sonra δ'ye ilişkin bölünme, t'deki tüm x_n'leri yanıt "evet" ise t_{sol}'a yanıt "hayır" ise $t_{sağ}$'a gönderir.

6 sınıflı bir sınıflandırma probleminde düğüm safsızlığı;

$$\mathrm{i(t)} = \sum_1^6 \mathrm{p(j|t)} \log \mathrm{p\,(j|t)} \quad (2)$$

şeklinde olabilir.

Her bir düğümdeki ikili bölünmelerin bir aday kümesi tanımlanır ve ağaç şu şekilde oluşturulur. t_1 kök düğümünde safsızlıktaki en yüksek azalmayı sağlayan δ^* bölünmesi bulunur;

$$\Delta \mathrm{i}(\delta^*, \mathrm{t}_1) = \max_{\delta \in S} \Delta \mathrm{i}(\delta, \mathrm{t}_1) \quad (3)$$

Burada S kümesi olası tüm bölünmeler kümesidir. Daha sonra t_1, δ^* bölünmesi kullanılarak t_2 ve t_3 düğümlerine bölünür ve aynı prosedür t_2 ve t_3 için en iyi $\delta \in S$ ile tekrarlanır (Loh, 2011).

Ağaç oluşturmasını durdurmak için, sezgisel bir kural oluşturulmuştur. Eğer bir t düğümü için safsızlıkta önemli derecede bir azalma olmuyorsa bu düğüm terminal düğüm olarak belirlenir. Terminal düğümün sınıfı, çokluk kuralı ile belirlenir (Chipman ve ark., 1998; Breiman ve ark., 2017).

<u>Bölme kuralları</u>**:** Bölme kuralları, $\emptyset(\delta, t)$ bölme iyiliği fonksiyonunun belirlenmesi ile oluşur. İki sınıf problemi için bölme kriteri ise;

$\emptyset(p_1, p_2) = 1 - max(p_1, p_2) = min(p_1, p_2) = min(p_1, 1 - p_1)$

olmak üzere, Gini Kriteri;

$$\emptyset(\mathrm{p}_1, \ldots, \mathrm{p}_\mathrm{j}) = -\sum_\mathrm{j} \mathrm{p}_\mathrm{j} \log \mathrm{p}_\mathrm{j} \quad (4)$$

olarak belirlenir.

Gini İndeksi;

$$\mathrm{i(t)} = \sum_{\mathrm{j} \neq \mathrm{i}} \mathrm{p(j|t)\ p(i|t)} \quad (5)$$

Ayrıca,

$$i(t) = \sum_j (p(j|t))^2 - \sum_j p^2(j|t) = 1 - \sum_j p^2(j|t) \quad (6)$$

şeklinde de yazılabilir. İkili sınıflandırma problemlerinde bu indeks;

$$i(t) = 2p(1|t)p(2|t) \quad (7)$$

şeklindedir (Alpaydın, 2014).

Rastgele Orman

Rastgele Orman yöntemi oluşturulmak istenen ağaç sayısınca Sınıflandırma Ağacının veya amaca uygun olarak Regresyon Ağacının topluluklarından oluşmaktadır. Bu yüzden topluluk yöntemlerinden en yaygın olarak kullanılan algoritmalardan biri de Rastgele Ormandır. Yöntemin altında yatan temel fikir, çok sayıda tahminci ağaçlar arasından rastgele seçilen bir alt kümesi yardımıyla topluluklar oluşturmaktır (Breiman, 2001).

Rastgele Orman Yöntemi hem kategorik hem sürekli hem de her ikisinin yer aldığı veri setlerinde; aynı zamanda büyük veya küçük boyutlu veri setlerinde rahatlıkla kullanılabilmektedir. Yöntemin dez avantajı olarak, Sınıflandırma Ağacı Yönteminin aksine çıktı olarak bir ağaç vermemesidir (Akman ve ark., 2011).

Bu şekilde rastgele tahminci seçmenin avantajı, topluluktaki ağaçlar arasında daha az korelasyon elde edildiği için oluşan modelin doğruluğu daha yüksektir (Suchetana ve ark., 2017).

Bu yöntemde de Sınıflandırma ve Regresyon ağaçlarında olduğu gibi bölünme kriteri olarak daha önce Eşitlik 5' te verilmiş olan Gini indeksi kullanılmaktadır. Gini indeksinin azalması istenen bir durumdur çünkü saflığın arttığına işarettir ve bu indeksin nihai olarak sıfıra eşit olması demek maksimum saflık demektir (Watts ve ark., 2011).

Bulgular

Öncelikle Botnet Sınıflandırma problemimizde ele aldığımız açıklayıcı değişkenler; Devam Süresi, Protokol Tipi, Kaynak IP, Hedef IP, Kaynak Port, Hedef Port, Syn Bayrak Durumu, Reset Bayrak Durumu, Ack Bayrak Durumu, Toplam Paket, Reset Bağlantı Sayısı, Ortalama Bayt, Ortalama Paket Oranı ve Paket Boyutudur. Yanıt değişkenimiz ise botnettir. Sınıflandırma Ağacı oluşturmak için analize tüm veri setinden başlanarak performans sonuçları incelenmiş, ağacın önemli bulduğu ve ağaç inşasında tek bir bölünmeyle yanlılığa sebep olan değişkenler tek tek çıkarılarak doğruluktaki azalma ele alınmıştır. Bu kapsamda Phyton Programlama dilinden elde edilen bulgular yardımıyla hata matrisi oluşturulmuş ve hata matrisinden hareketle bazı model performans ölçütleri hesaplanmıştır.

Tüm değişkenler modele dâhil edildiğinde ve önem sırasına (ağacın ilk bölünmesine) göre Ortalama Bayt Oranı, Süre, Reset Bağlantı Sayısı, Ortalama Paket Oranı değişkenleri teker teker çıkarıldığında hata matrisi hep aynı sonucu vererek doğru sınıflandırma oranı % 100, yanlış pozitif ve yanlış negatif oranı sıfır olmuştur.

Çizelge 1. Ortalama Bayt Oranı, Süre, Reset Bağlantı Sayısı, Ortalama Paket Oranı değişkenleri modelden teker teker çıkarıldığında sınıflandırma ve regresyon ağacı ve rastgele orman yöntemine ilişkin hesaplanan bazı model performans ölçütleri.

Yöntem	Doğruluk	TPR	SPC	FPR	FNR	PPV	NPV
CART	% 100	1.000	1.000	0.000	0.000	1.000	1.000
RF	% 100	1.000	1.000	0.000	0.000	1.000	1.000

ART: Sınıflandırma ve Regresyon Ağacı (Classification and Regression Tree), RF: Rastgele Orman (Random Forest), TPR: Doğru Pozitif Oran (True Positive Rate),SPC: Duyarlık (Specivity), FPR: Yanlış Pozitif Oran (False Positive Rate), FNR: Yanlış Negatif Oran (False Negative Rate), PPV: Pozitif Tahmin Değeri (Positive Predictive Value), NPV: Negatif Tahmin Değeri (Negative Predictive Value).

Çizelge 2. Toplam Bayt değişkeni modelden çıkarıldığında sınıflandırma ve regresyon ağacı ve rastgele orman yöntemine ilişkin hesaplanan bazı model performans ölçütleri.

Yöntem	Doğruluk	TPR	SPC	FPR	FNR	PPV	NPV
CART	% 100	1.000	1.000	0.000	0.000	1.000	1.000
RF	% 99.858	0.997	1.000	0.000	0.003	1.000	0.997

Çizelge 3. Kaynak IP değişkeni modelden çıkarıldığında sınıflandırma ağacı ve rastgele orman yöntemine ilişkin hesaplanan bazı model performans ölçütleri.

Yöntem	Doğruluk	TPR	SPC	FPR	FNR	PPV	NPV
CART	% 92.129	0.875	0.993	0.007	0.125	0.995	0.846
RF	% 99.854	0.997	1.000	0.000	0.003	1.000	0.997

Çizelge 4. Hedef IP değişkeni analizden çıkarıldığında sınıflandırma ağacına ilişkin hesaplanan bazı model performans ölçütleri.

Yöntem	Doğruluk	TPR	SPC	FPR	FNR	PPV	NPV
CART	% 93.916	0.892	1.000	0.000	0.108	1.000	0.877
RF	% 96.47	0.937	0.996	0.004	0.063	0.996	0.932

Çizelge 5. Kaynak port değişkeni analizden çıkarıldığında sınıflandırma ağacına ilişkin hesaplanan bazı model performans ölçütleri.

Yöntem	Doğruluk	TPR	SPC	FPR	FNR	PPV	NPV
CART	% 78.383	0.854	0.735	0.265	0.146	0.687	0.881
RF	% 86.141	0.930	0.806	0.194	0.070	0.774	0.949

Çizelge 6. Paket boyutu değişkeni analizden çıkarıldığında sınıflandırma ağacına ilişkin hesaplanan bazı model performans ölçütleri.

Yöntem	Doğruluk	TPR	SPC	FPR	FNR	PPV	NPV
CART	% 62.224	0.818	0.570	0.430	0.182	0.242	0.928
RF	% 62.224	0.818	0.570	0.430	0.182	0.320	0.928

Çizelge 7. Statesfin değişkeni analizden çıkarıldığında sınıflandırma ağacına ilişkin hesaplanan bazı model performans ölçütleri.

Yöntem	Doğruluk	TPR	SPC	FPR	FNR	PPV	NPV
CART	% 59.3	0.557	0.00026	0.999	0.443	0.930	0.251
RF	% 59.3	0.557	0.00026	0.999	0.443	0.930	0.251

Çizelge 8. Statesack değişkeni analizden çıkarıldığında sınıflandırma ağacına ilişkin hesaplanan bazı model performans ölçütleri.

Yöntem	Doğruluk	TPR	SPC	FPR	FNR	PPV	NPV
CART	% 56.26	0.541	0.647	0.353	0.459	0.858	0.263
RF	% 56.26	0.541	0.647	0.353	0.459	0.858	0.263

Çizelge 9. Protokol değişkeni analizden çıkarıldığında sınıflandırma ağacına ilişkin hesaplanan bazı model performans ölçütleri.

	Doğruluk	TPR	SPC	FPR	FNR	PPV	NPV
CART	% 56.26	0.0006	0.517	0.483	0.999	0.106	0.960
RF	% 56.26	0.541	0.647	0.353	0.459	0.858	0.263

Çizelge 10. Statessyn değişkeni analizden çıkarıldığında sınıflandırma ağacına ilişkin hesaplanan bazı model performans ölçütleri.

Yöntem	Doğruluk	TPR	SPC	FPR	FNR	PPV	NPV
CART	% 53.495	0.000	0.490	0.510	1.000	0.000	1.00
RF	% 53.495	0.0006	0.517	0.483	0.999	0.106	0.96

Çizelge 11. Staterst değişkeni analizden çıkarıldığında sınıflandırma ağacına ilişkin hesaplanan bazı model performans ölçütleri.

Yöntem	Doğruluk	TPR	SPC	FPR	FNR	PPV	NPV
CART	% 49.68	0.000	0.490	0.510	1.000	0.000	0.000
RF	% 49.68	0.000	0.490	0.510	1.000	0.000	0.000

Tartışma ve Sonuç

Siber güvenlik alanında önemli bir yere sahip olan botnet tespiti ve sınıflandırılması için, botnet akışı ile normal akışı bir birinden ayırt edeceği düşünülen değişkenler ele alınarak, Sınıflandırma ve Regresyon Ağaçları ile Rastgele Orman Yöntemleri ile analiz edilmiştir.

Sınıflandırma Ağacında önemli bulunan değişkenler ağacın ilk bölünmesine sebep olan değişkenler olduğu için bir sonraki ağacı oluşturmada bu değişken analizden çıkarılmıştır. Böylece ağacın bir tek değişkene göre bölünmesine bağlı olarak ortaya çıkan yanlılık (% 100 doğruluk), değişkenleri tek tek analizden çıkarılarak azaltılmaya çalışılmıştır. Her bir değişken çıkarıldıktan sonra modelde oluşan değişimler bazı model performans ölçütleri ile hesaplanmıştır. Buradaki amaç, ağ akışının botnetli mi yoksa normal mi olduğuna karar verirken önemli olan özelliklerin belirlenmesi ve seçimi ve performansa katkısını ölçmektir.

Yapılan iki analiz sonucu da Ortalama Bayt Oranı, Süre, Reset Bağlantı Sayısı, Ortalama Paket Oranı değişkenleri Botnetli bir trafiği belirlemede % 100 başarı gösterdiği (Çizelge 1) için önemli bulunmuştur, bu değişkenlerden her biri modelden çıkarıldığında sınıflandırma modelinin doğruluğunda bir azalma olmamıştır. Dolayısıyla bu çalışmada kullanılan ağ akışını botnetli mi, normal mi olup olmadığını ayırt etmek için Ortalama Bayt Oranı, Süre, Reset Bağlantı Sayısı, Ortalama Paket Oranı değişkenlerinden birini baz almak yeterli olacaktır.

Ağacın sadece bir düğüm oluşturmasına sebep olan değişken, yanıt değişkeni ile yüksek korelasyonlu olduğundan diğer değişkenleriniz ne olursa olsun, ağacınız her zaman bu değişkene ait bir düğümle sonuç verecektir. Bu çalışmada bu sorunun

çözümü için değişkenleri tek tek analizden çıkarmak uygun bulunmuştur. Ayrıca her bir değişken çıkarıldığında sınıflandırmaya ait doğrulukta meydana gelen azalma, o değişkenin modele ne kadar katkısı olduğuna dair bilgiler vermektedir.

Sınıflandırma Ağaçları ve Regresyon Ağaçları ile toplanan tüm akış analiz edildiğinde akışın normal olup olmadığını belirlemede değişkenler, analizden çıkarma sırasına göre önemli bulunmuştur. Hem Sınıflandırma Ağaçları ve Regresyon Ağaçları hem de Rastgele Orman Yöntemi, bu akışta benzer ve başarılı bir sınıflandırıcı performansı sergilemişlerdir. Bu çalışmada yakalanan bu başarı ile kullanılan yöntemlerin, ağ akışı incelemelerinde ve diğer zararlı yazılımları tespit etmede etkin olduğunu söylenebilir.

Çalışmanın, Siber Güvenlik Alanına dikkat çekmesi, Sınıflandırma ve Regresyon Ağaçlarının yorumunun kolay olduğunun gösterilmesi, ağ akışında botnet tespitinde kullanılan açıklayıcı değişkenler için bir fikir oluşturması, sanal bilgisayar kurulumu gibi konulara değinmesi bakımından önemli olduğu düşünülmektedir. Diğer Makine Öğrenmesi Tekniklerinden Sınıflandırma Algoritmaları, ağ akışı incelemelerinde kullanılabilir ve yöntemlerin gösterdiği performanslar ayrı ayrı ele alınarak bu minvaldeki çalışmaların kapsamı genişletilebilir.

Kaynaklar

Akman, M., Genç, Y., Ankaralı, H., 2011. Random forests yöntemi ve sağlık alanında bir uygulama, Türkiye Klinikleri Journal of Biostatistics, 3 (1): 36-48.

Alpaydın, E., 2014. Introduction to Machine Learning, MIT Press, 3rt edition.

Anonim, 2018a. Sosyal Medya ve Mobil Kullanıcı İstatistikleri. https://dijilopedi.com/2018-internet-kullanimi-ve-sosyal-medya-istatistikleri/ Erişim tarihi: 01.03.2018.

Anonim, 2018b. Avrupa'daki En Fazla Siber Saldırı Türkiye'de. http://www.sigortacigazetesi.com.tr/avrupadaki-en-fazla-siber-saldiri-turkiyede/ Erişim tarihi: 01.03.2018.

Bock, H. H., 2002. Data mining tasks and methods: Classification: the goal of classification, In Handbook of Data Mining and Knowledge Discovery, 254-258.

Breiman, L., 2001. Random Forests, Machine Learning, 45 (1): 5-32.

Breiman, L., Friedman, J. H., Olshen, R. A., Stone, C. J., 2017. Classification and Regression Trees, Taylor Francis, Berkeley, California.

Chen, R., Niu, W., Zhang, X., Zhuo, Z., Lv, F., 2017. An effective conversation-based botnet detection method, Mathematical Problems in Engineering, Article ID 4934082:9.

Chipman, H. A., George, E. I., McCulloch, R. E., 1998. Bayesian CART modelsearch, Journal of the American Statistical Association, 93 (443): 935-948.

De'ath, G., Fabricius, K. E., 2000. Classification and regression trees: a powerful yet simple technique for ecological data analysis, Ecology, 81(11): 3178-3192.

Gu, G., Zhang, J., & Lee, W., 2008. BotSniffer: Detecting botnet command and control channels in network traffic, 17th USENIX Security Symposium.

Guttman, A., 1984. R-trees: A dynamic index structure for spatial searching, 47-57: SIGMOD'84,

Proceedings of Annual Meeting, Boston, Massachusetts, June 18-21.

Kalaivani, P., Vijaya, M., 2016. Mining based detection of botnet traffic in network flow, International Journal of computer Science and information Technology & Security, 6: 535-540.

Karasaridis, A., Rexroad, B., Hoeflin, D. A., 2007. Wide-Scale Botnet Detection and Characterization. HotBots, 7: 7.

Loh, W. Y., 2011. Classification and regression trees. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, 1 (1): 14-23.

Shinder, D. L., Tittel, E. 2002. Scene of the Cybercrime: Computer Forensics Handbook, Syngress Publishing.

Suchetana, B., Rajagopalan, B., Silverstein, J., 2017. Assessment of wastewater treatment facility compliance with decreasing ammonia discharge limits using a regression tree model, Science of the Total Environment, 598: 249-257.

Timofeev, R., 2004. Classification and Regression Trees (CART) Theory and Applications (master thesis). Humboldt University, Berlin.

Watts, J. D., Powell, S. L., Lawrence, R. L., Hilker, T., 2011. Improved classification of conservation tillage adoption using high temporal and synthetic satellite imagery, Remote Sensing of Environment, 115 (1): 66-75.

Zhao, D., Traore, I., Sayed, B., Lu, W., Saad, S., Ghorbani, A., Garant, D., 2013. Botnet detection based on traffic behavior analysis and flow intervals. Computers & Security, 39: 2-16.