Süleyman Demirel Üniversitesi
Fen Bilimleri Enstitüsü Dergisi
Cilt 22, Sayı 1, 283-289, 2018

Süleyman Demirel University
Journal of Natural and Applied Sciences
Volume 22, Issue 1, 283-289, 2018

DOI: 10.19113/sdufbed.93423

Topluluk Yöntemlerine Dayalı Dağıtık Hizmet Dışı Bırakma Saldırılarının Algılanması

Ferhat Özgür ÇATAK *[1]

[1]TÜBİTAK BİLGEM, Siber Güvenlik Enstitüsü, Kocaeli

(Alınış / Received: 15.04.2017, Kabul / Accepted: 07.06.2017, Online Yayınlanma / Published Online: 18.06.2017)

Anahtar Kelimeler
Siber güvenlik,
Dağıtık hizmet dışı bırakma saldırıları,
Makine öğrenmesi,
Topluluk algoritmaları

Özet: Dağıtık hizmet dışı bırakma eski bir siber saldırı yöntemi olmasına rağmen günümüzde saldırganlar tarafından hala kullanılmaktadır. Saldırganlar, internet üzerinde yer alan protokollerin mevcut zafiyetleri kullanılarak çeşitli katmanlarda bu tip saldırılar gerçekleştirmektedirler. Günümüzde makine öğrenmesi yöntemleri gelişen teknoloji ile beraber yüksek boyutlu veri kümelerine uygulanabilir olmaktadır. Siber saldırıların algılanması için kullanılacak olan veri kümeleri yüksek sayıda satırlar içeren log dosyalarıdır. Bu çalışmada dağıtık hizmet dışı bırakma saldırılarında elde edilmiş olan logların analiz edilerek tahmin modeli ortaya çıkarılması hedeflenmiştir. Topluluk yöntemleri kullanılarak, siber güvenlik veri kümeleri eğitilebilir duruma getirilmektedir. Farklı parametreler kullanılarak model performans ölçümü uygulanmıştır. Bu şekilde en yüksek doğruluğa sahip model oluşturulması hedeflenmiştir. Ortaya konulan modelin sınıflandırma performans ölçüsü tablo ve şekillerle paylaşılmıştır.

Detection of Distributed Denial of Service Attacks Based on Ensemble Methods

Keywords
Cyber security,
Distributed denial of service attacks,
Machine learning,
Ensemble methods

Abstract: Distributed denial-of-service is still used by attackers today, although it is an old method of cyber attack. Attackers are performing such attacks on various layers using the existing weaknesses of the protocols on the internet. Today, machine learning methods can be applied to high-dimensional data sets together with developing technology. The data sets to be used for the detection of cyber attacks are log files with a high number of rows. In this study, it is aimed to analyze the logs obtained in distributed denial-of-service attacks to build the prediction model. Cyber security data sets are brought into a trainable state using ensemble methods. Model performance measurement was applied using different parameters. It is aimed to create a model with the highest degree of accuracy in this way. The classification performance of the proposed model is shared with tables and figures.

1. Giriş

Bilgisayar kullanımının oldukça artmış olduğu günümüzde, gerçekleşen siber saldırıların ölçeği buna bağlı olarak oldukça artmaktadır. Yapılan siber saldırıların önemli bir kısmını dağıtık hizmet dışı bırakma saldırıları adı verilen, hizmet veren sistemin işlemci, disk, ağ gibi kaynaklarını tüketmeyi hedefleyen saldırılardır [1]. Dağıtık hizmet dışı bırakma saldırıları, adından anlaşıldığı üzere tek saldırgan bilgisayar tarafından değil, birden fazla bilgisayarın koordineli bir şekilde saldırması ile gerçekleşmektedir. Bu şekilde bir koordinasyon içeren saldırgan bilgisayarlar topluluğuna RobotNetwork veya kısaca botnet adı verilmektedir. Günümüzde yapılan dağıtık hizmeti dışı bırakma saldırıları için kullanılan botnet ağında yer alan bileşenler sadece kişisel bilgisayarlardan oluşmayıp, Mirai saldırısında olduğu gibi kişisel bilgisayarların dışında nesnelerin interneti adı verilen akıllı cihazlarda saldırıların bir parçası haline gelmişlerdir. Dağıtık hizmet dışı bırakma saldırılarının en büyük etkisi, verilen hizmetten faydalanması gereken meşru kullanıcıların hizmeti alamaması veya hizmet kalitesinin düşmesi şeklindedir. Bu nedenlerden dolayı, dağıtık hizmet dışı bırakma saldırılarının oldukça kısa sürede algılanması, hizmeti sunan açısından tehditin ortadan kaldırılması için oldukça önem arz etmektedir.

İnternet ortamında çeşitli hizmetleri sunan kurumlar, sundukları hizmet hakkında kayıtlar tutmaktadırlar. Tutulan bu kayıtlarda içerik olarak hizmet isteğinde bulunan kullanıcının IP adresi, port numarası, hedef sistemin IP adresi, port numarası, gönderilen paketleri paketlerin içerisinde yer alan payload bilgisi, tarih bilgisi gibi kayıtlar bulunmaktadır. Büyük veri teknolojilerinin gelişmesine paralel olarak bu log dosyaları oldukça uzun süreler boyunca kayıt altına alınıp saklanmaktadırlar. Fakat bu kayıtlardan anlamlı bilgilerin çıkarılması işlemi oluşan verinin büyüklüğü nedeniyle kolay olmamaktadır.

*İlgili yazar: ozgur.catak@tubitak.gov.tr

Günümüzde büyük veri teknolojilerinin gelişmesine paralel olarak mevcut makine öğrenme algoritmalarında da gelişmeler gözlemlenmektedir. Geliştirilen yöntemlerle, yüksek boyutlu veri kümelerinin algoritmaların eğitim aşamasında kullanılabilir olmasına imkan sağlamıştır. Özellikle topluluk yöntemleri kullanılarak sadece bir sınıflandırma modeli oluşturmak yerine birden fazla sınıflandırma modeli oluşturulması yaklaşımı kullanılarak hem nihai sınıflandırma modelinin doğruluk oranı artmakta hem de eğitim aşamasında kullanılan veri kümesinin tamamının eğitilebilir olması sağlanabilmektedir.

Literatürde çeşitli makine öğrenme yöntemleri kullanılarak dağıtık hizmet dışı bırakma saldırılarının algılanması ile ilgili çalışmalar mevcuttur.

Bhatia [2] yaptığı çalışmada hizmet dışı bırakma saldırılarını yapılan ağ katmanına göre farklı değerlendirilmesi gerektiğini belirtmiştir. SYN seli [3] saldırısı gibi ağ üzerinde paket trafiğini artırarak hizmetin bulunduğu sunucunun ağ bileşen kaynaklarını tüketmeye yönelik saldırılar olabileceği gibi, uygulama seviyesinde yapılan bir saldırı ile hem ağ bant genişliği hem de CPU kaynak tüketimi hedeflenebilmektedir. Bu çalışma kapsamında ağ seviyesinde gerçekleşen bir saldırının algılanması için gerekli kriterler uygulama seviyesinde yapılan saldırılar ile aynı olmayacağı ifade edilmiştir. Bu nedenle farklı nitelik ve yöntemler kullanılarak her iki tip saldırının algılanması ile modeller oluşturulmuştur.

Diğer bir çalışmada [4], ağ üzerinden yapılan smurf ve teardrop gibi çeşitli hizmet dışı bırakma saldırılarını algılamak için farklı makine öğrenme yöntemleri kullanmışlardır. Kullanılan algoritmalar sırasıyla naive bayes, bayes ağları, karar ağacı ve azalan hata budama (REPTree) şeklindedir. Çalışma kapsamında kullanılan veri kümesi, kddcup99 olarak adlandırılan ağ sızma algılama veri kümesidir. Çalışmada iki ve daha fazla algoritmanın beraber kullanılması ile elde edilen sınıflandırma modeline topluluk yöntemleri adını vermişlerdir. Yazarlar kddcup99 veri kümesi kullanılarak oluşturulan modeller ile %99.94117 oranında doğruluk elde etmişlerdir.

Osanaiye ve diğerleri [5] nitelik seçimlerinde topluluk yöntemleri kullanmışlardır. Bilgi kazanımı, kazanım oranı, ki-kare ve ReliefF yöntemleri kullanılarak kddcup99 veri kümesinin geliştirilmiş bir hali olan NSL-KDD veri kümesi üzerinde test etmişlerdir. Önerdikleri modeli kullanarak 13 nitelik seçmişler, seçilen nitelikler karar ağacı algoritması kullanılarak eğitilmiş ve ortaya çıkan sınıflandırma modelinin doğruluk oranını %99.67 olarak ölçümlemişlerdir.

Livadas ve diğerleri, IRC tabanlı botnet'lerin komuta kontrol trafiğini tespit etmek için ağ akış temelli bir yaklaşım önermişlerdir [6]. Yaptıkları çalışmada, ilk aşamada, sınıflandırma algoritmaları ile trafik akışlarını IRC sohbet veya IRC sohbet-dışı akışlara sınıflandırırken, ikinci aşamada, IRC akışlarını kötü amaçlı veya kötü amaçlı olmayan olarak sınıflandırmaktadırlar. Modelin performans ölçüm değerlerine bakıldığında nispeten yüksek kabul edilebilecek %10-20 oranında hatalı negatif oranı ve %30-40 dolaylarında hatalı pozitif oranı elde etmişlerdir.

Zhao ve diğerleri trafik davranış analizi ve akış aralıklarına dayalı bir botnet algılama sistemi önermişlerdir [7]. Bu çalışma kapsamında kötü niyetli ve kötü amaçlı olmayan ağ trafiğini sınıflandırmak için azalan hata budama algoritmasını (REPTree) kullanmışlardır.

Saad ve diğerleri, kötü niyetli e-postalar, web siteleri, dosya paylaşım ağları ve geçici kablosuz ağlar aracılığıyla P2P botnet komutu ve kontrol aşamasını radar altında tespit etmek için ağ trafiği davranışının özelliklerini incelemişlerdir [8]. ISOT botnet veri kümesini kullanarak botnet trafiğini tanımlamak için beş farklı makine öğrenme algoritması kullanmışlar ve %89'luk en yüksek doğruluk oranı elde etmişlerdir.

Lu ve diğerleri, kötü amaçlı IRC bot trafiğini normal aralıktan ayırmak için önceden tanımlanmış bir zaman aralığı boyunca yük üzerindeki 256 ASCII baytın zamansal sık özelliklerini analiz eden bir algılama sistemi önermişlerdir [9].

Masud ve diğerleri ana makine üzerinde kurulu birden çok günlük dosyası arasındaki korelasyonu göz önüne alarak, akış temelli bir algılama yöntemi kullanan botnet algılama sistemi önermişlerdir [10].

Literatürde yer alan çalışmalar genellikle sadece bir sınıflandırıcı fonksiyondan oluşan modellerin geliştirilmesine odaklanmışlardır. Bu çalışmanın amacı dağıtık hizmet dışı bırakma saldırılarının incelenmesi ve bu tip saldırıların tahmin edilmesi amacıyla ağ trafiğinin sınıflandırma algoritmalarında kullanılması şeklindedir. Çalışmanın katkıları şu şekilde sıralanabilir:

- Dağıtık hizmet dışı bırakma saldırılarında oluşan log dosyasında bulunan alanlardan önemli olanları ortaya çıkarmak.
- Yüksek boyutlu ağ trafiği dosyasından topluluk yöntemlerine dayalı sınıflandırma modelleri çıkararak ileride yaşanabilecek saldırıların tahmin edilmesi.
- Literatürde yer alan çalışmalar genellikle oldukça eski sayılabilecek kddcup99 verikümesini kullanmaktadır. Bu çalışma ile daha güncel bir veri kümesi kullanılmaktadır.

Bu çalışmada kullanılan veri kümesi, internet ortamında erişilebilen dağıtık hizmet dışı bırakma saldırıları sonucundan elde edilmiş kayıt dosyalarından oluşmaktadır. Bu veri kümesi kullanılarak topluluk yöntemlerine dayalı çeşitli sınıflandırma modelleri oluşturulmuştur. Oluşturulan modellerin sınıflandırma performans değerleri, kullanılan niteliklerin önemleri tablo ve grafiklerle paylaşılmıştır.

2. Materyal ve Metot

Çalışmanın bu kısmında topluluk yöntemlerine dayalı sınıflandırma algoritmalarının temeli, kullanılan veri

kümesi, PCAP dosya formatı hakkında bilgiler paylaşılacaktır.

2.1. Gradyan artırma sınıflandırma algoritması

Topluluk yöntemlerinden bir tanesi olan gradyan artırma sınıflandırma algoritması, türevlenebilir kayıp fonksiyonlarının artırma genelleştirilmesi için kullanılan bir yöntemdir [11]. Bu yöntem hem regresyon hem de sınıflandırma problemleri için kullanılabilen, sınıflandırma performansı yüksek ve etkili bir algoritmadır. Bu algoritma kullanılarak oluşturulan modeller, web arama sıralaması [12] ve ekoloji [13] dahil olmak üzere çeşitli alanlarda kullanılmaktadır. Yöntemin avantajları şu şekilde sıralanabilir:

- Karışık tipteki verileri ele alması.
- Yüksek tahmin gücü.
- Çıktı uzayında bulunan sapmalara karşı güvenilirlik.

Gradyan Artırma temel olarak üç bileşen içerir:

- Optimize edilecek bir *kayıp fonksiyonu*
- Tahminler yapacak bir *zayıf sınıflandırıcı*
- Kayıp fonksiyonunu minimize etmek için zayıf sınıflandırıcıları birleştirien *ekleme modeli*

Aşağıda yer alan bölümlerde bu üç bileşen açıklanmıştır.

2.1.1. Kayıp fonksiyonu

Yöntemin kullanacağı kayıp fonksiyonu türevlenebilir olmalıdır. Sınıflandırma işlemininde kullanılması durumunda logarithmic kayıp (LogLoss), regresyon işlemi için ise ortalama hataların karesi (RMSE) fonksiyonu kullanılabilir. Bu fonksiyonlar Denklem 1 ve 2'de gösterilmiştir.

$$LogLoss = -\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{M} y_{i,j} log(p_{i,j}) \tag{1}$$

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{n}(y_i - \hat{y}_i)} \tag{2}$$

2.1.2. Zayıf sınıflandırıcı

Gradyan artırmada, karar ağaçları yöntemi zayıf sınıflandırıcı olarak kullanılmaktadır. Karar ağaçları, Gini gibi saflık skorlarına dayalı en iyi bölünme noktalarını seçerek veya kaybı en aza indirgemek için açgözlü (greedy) bir şekilde inşa edilmektedir.

2.1.3. Ekleme modeli

Karar ağaçları birer birer eklenir ve modeldeki mevcut ağaçlar değiştirilmez. Yeni ağacın çıktısı olan sınıflandırıcı mevcut ağaç dizisine eklenir. Bu şekilde nihani topluluk modelinin sınıflandırma performansının iyileştirilmesi hedeflenmektedir.

2.1.4. Matematik modeli

Yöntem aşağıdaki denklemde bulunan sınıflandırma fonksiyonlarını göz önüne alarak sınıflandırma modeli oluşturmaktadır.

$$F(\mathbf{x}) = \sum_{m=1}^{M} \gamma_m h_m(\mathbf{x}) \tag{3}$$

Burada $h_m(\mathbf{x})$ ifadesi, artırma bağlamında genellikle zayıf sınıflandırma hipotezleri için adlandırılan temel sınıflandırma fonksiyonudur. Gradyan artırma yöntemi, zayıf sınıflandırma hipotezleri olarak genellikle sabit boyuttaki karar ağaçlarını kullanmaktadır. Karar ağaçlarının, farklı türdeki verileri işleyebilme yetenekleri ve karmaşık işlevleri modelleme becerisi gibi önemli yetenekleri vardır.

Diğer artırma algoritmalarına benzer şekilde gradyan artırma, ileri modele göre eklemeli sınıflandırma modeli geliştirmektedir.

$$F_m(\mathbf{x}) = F_{m-1}(\mathbf{x}) + \gamma_m h_m(\mathbf{x}) \tag{4}$$

Her bir aşamada, karar ağacı (veya sınıflandırma hipotezi), $h_m(\mathbf{x})$, mevcut model F_{m-1} ve onun uyumu $F_{m-1}(\mathbf{x}_i)$ ile verilen kayıp fonksiyonunu (L) en aza indirgemek üzere seçilmektedir.

$$F_m(\mathbf{x}) = F_{m-1}(\mathbf{x}) + \arg\min_{h} \sum_{i=1}^{n} L(y_i, F_{m-1}(\mathbf{x}) \tag{5}$$

Gradyan artırma yöntemi, bu minimizasyon problemini en dik inişle sayısal olarak çözmeye çalışmaktadır. Optimizasyon çözülürken en dik iniş yönü, herhangi bir türevlenebilir kayıp fonksiyonu için hesaplanabilen mevcut F_{m-1} modelinde değerlendirilen kayıp fonksiyonunun negatif eğimidir.

$$F_m(\mathbf{x}) = F_{m-1}(\mathbf{x}) + \gamma \sum_{i=1}^{n} \nabla_f L(y_i, F_{m-1}(\mathbf{x}_i)) \tag{6}$$

Burada adım uzunluğu γ_m ise hat arama yöntemi ile seçilmektedir.

$$\gamma_m = \arg\min_{\gamma} \sum_{i=1}^{n} L\left(y_i, F_{m-1}(\mathbf{x}_i) - \gamma \frac{\delta L(y_i, F_{m-1}(\mathbf{x}_i))}{\delta F_{m-1}(\mathbf{x}_i)}\right) \tag{7}$$

Gradyan artırma yönteminin sözde kodu Algoritma 1'de gösterilmiştir.

2.2. Veri kümesi

Bu çalışma kapsamında kullanılan veri kümesi, Avustralya Siber Güvenlik Merkezi'nde bulunan Siber Güvenlik Laboratuvarından alınmıştır [14]. Veri kümesi oluşturulurken IXIA PerfectStorm aracı kullanılmış, bu şekilde normal aktiviteler ve saldırı davranışlarını içeren ağ trafiği PCAP dosya formatında kayıt edilmiştir [15]. Veri kümesi *Fuzzers*, *Analysis*, *Backdoors*, *DoS*, *Exploits*, *Normal*, *Reconnaissance*, *Shellcode* ve *Worms* şeklinde 9 farklı etikete

Algoritma 1 Gradyan artırma sınıflandırıcısı sözde kodu.

Girdi: Eğitim seti $(\mathbf{x}_i, y_i)_{i=1}^{n}$, türevlenebilir kayıp fonksiyonu $L(y, F(y))$, iterasyon sayısı M

1: sabit bir değer ile modeli ilklendir

$$F_0 = \arg\min_{r} \sum_{i=1}^{n} L(y_i, r)$$

2: **for** i = 1 to M **do**

3: Sözde-reziduali hesapla

$$r_{im} = -\left[\frac{\partial L(y, F(\mathbf{x}))}{\partial F(\mathbf{x})}\right]_{F(\mathbf{x})=F_m(\mathbf{x})}, \forall i \in n$$

4: Sözde-rezidual kullanarak $\{(\mathbf{x}_i, r_{im})\}_{i=1}^{n}$, zayıf sınıflandırıcı $h_m(\mathbf{x})$ hesaplanır

5: Optimizasyon problemi çözülerek çarpan γ_m hesaplanır.

$$\gamma_m = \arg\min_{\gamma} \sum_{i=1}^{n} L(y_i, F_{m-1}(\mathbf{x}_i) + \gamma h_m(\mathbf{x}_i)$$

6: Sınıflandırma modelini güncelle

$$F_m(\mathbf{x}) = F_{m-1}(\mathbf{x}) + \gamma_m h_m(\mathbf{x})$$

7: Çıktı: $F_m(\mathbf{x})$

sahip veriler içermektedir. PCAP dosyasından 49 farklı niteliğin çıkarılması işlemi için Argus aracı kullanılmıştır [16]. Bahsedilen 49 nitelik arasında *kaynak IP adresi*, *protokol* gibi sayısal olmayan niteliklerde mevcuttur. Bu çalışmada kullandığımız sınıflandırma algoritması sebebiyle seçilen niteliklerin sayısal olması gerekmektedir. Bu nedenle, bu çalışma kapsamında bahsedilen 49 nitelik arasından sınıflandırma algoritmalarında kullanılmak üzere 25 adet sayısal değer içeren nitelik seçilmiştir. Sayısal değer içermeyen diğer nitelikler çalışmaya dahil edilmemiştir. Seçilen nitelikler ve açıklamaları Tablo 1'de bulunmaktadır.
Tablo 2'de veri kümesinde yer alan kayıtların *normal* ve *saldırı* dağılımları gösterilmektedir.

Tablo 2. Veri kümesinde bulunan nitelikler.

Trafik	Toplam kayıt
Normal	37000
Saldırı	45332

2.3. Sınıflandırma model ölçümü

Bu çalışma kapsamında önerilen modelin sınıflandırma performansının ölçülmesi için hassaslık (precision), geri çağırma (recall), F_1-ölçüsü ve doğruluk olmak üzere dört farklı sınıflandırma model değerlendirme metriği kullanılmıştır. Sınıflandırma algoritmalarının öğrenme aşamasında kullanılan eğitim veri kümesinin örnek boyutunu dikkatle seçilerek çok yüksek doğruluğu elde etmek kolaydır. Yalnızca doğruluk metriğinin kullanılması, önerilen modelin sınıflandırma performansını test edilmesinde hatalı yorumlamalara neden olabilmektedir. Bu sorunun üstesinden gelmek için, eğitimin veri kümesinin büyüklüğüne ve test örneklerine bağlı olmayan hassaslık (pozitif öngörme değeri) ve geri çağırma değerleri dikkate alınarak model değerlendirilmesi yapılması gerekmektedir.

Hassaslık elde edilen pozitif örneklerden (bu çalışma için botnet trafiği) gerçek pozitif olanların elde edilen değerlere olan oranıdır.

$$Hassaslik = \frac{Gercek\,Pozitif}{Toplam\,Pozitif\,Etiketlenen} \tag{8}$$

Geri çekilme oranı ise elde edilen pozitif örneklerden gerçek pozitif olanların örneklem kümesi içerisinde bulunan toplam pozitif örnek sayısına olan oranıdır.

$$GeriCekilme = \frac{Gercek\,Pozitif}{Toplam\,Pozitif} \tag{9}$$

Hassaslık ve geri çağırma değerleri birbirlerine ters orantılı olarak davranmaktadır ve normalde bu iki değer arasında bir dengelenme mevcuttur. Bu nedenle hassaslık ve geri çekilmenin harmonik ortalaması olan F_1-ölçüsü denilen bir başka değerde oluşturulan modelin değerlendirmesinde dikkate alınmıştır.

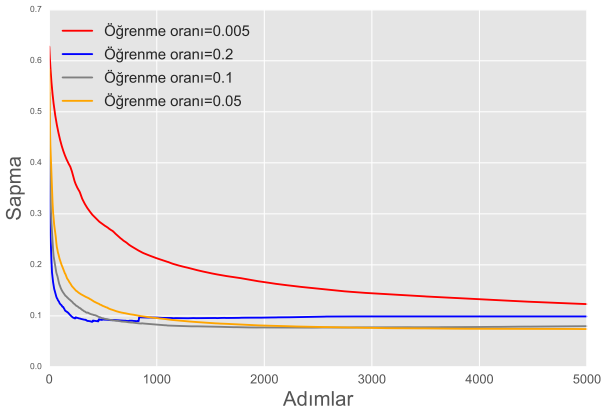
$$F_1 = 2 \times \frac{(Hassaslik) \times (GeriCagirma)}{(Hassaslik) + (GeriCagirma)} \tag{10}$$

Bu çalışma kapsamında ortaya konulan modellerin doğruluğunu göstermek amacıyla bahsedilen dört farklı metrik kullanılacaktır.

3. Bulgular

3.1. Sınıflandırma modeli eğitim aşamaları

Model eğitilirken algoritmanın sahip olduğu zayıf sınıflandırıcı sayısı ve öğrenme oranı parametreleri kullanılmıştır. Zayıf sınıflandırıcı sayısı 100 seçilerek oluşturulan modelin doğruluk oranını yüksek olması hedeflenmiştir. Şekil 1'de modelin oluşturulmasında kullanılan algoritmanın 0.2, 0.1, 0.05 ve 0.005 şeklinde dört farklı öğrenme oranı değeri ile model oluşturma esnasında her bir adımda sapma oranı verilmiştir.



Şekil 1. Veri kümesinde bulunan niteliklerin önem değerleri.

Şekil 1'de en yüksek öğrenme oranı olan 0.2'ye sahip olan sınıflandırma modelinin eğitim aşamasında yer

Tablo 1. Veri kümesinde bulunan nitelikler.

Nitelik Adı	Açıklama
ackdat	TCP bağlantı kurulum zamanı SYN_ACK ve ACK paketleri arasındaki süre.
ct_flw_http_mthd	Http hizmetinde Get ve Post gibi yöntemler olan akış sayısı.
ct_ftp_cmd	Ftp oturumunda bir komut olan akışların sayısı.
dbytes	Hedef işlem boyutu
dinpkt	Hedef katmanlararası varış süresi (ms)
djit	Hedef jitter (ms)
dload	Hedef bit / saniye
dloss	Yeniden aktarılan veya silinen hedef paket sayısı
dmean	Hedef tarafından iletilen Ham paket boyutunun ortalaması.
dpkts	Hedef paket sayısı
dur	Toplam süre
dwin	Hedef TCP pencere değeri
is_ftp_login	Ftp oturumu
response_body_len	Sunucu http servisi tarafından cevap boyutu
sbytes	Kaynak işlem boyutu
sinpkt	Kaynak katmanlararası varış süresi (ms)
sjit	Kaynak jitter (ms)
sload	kaynak bit / saniye
sloss	Yeniden aktarılan veya silinen kaynak paket sayısı
smean	Kaynak tarafından iletilen Ham paket boyutunun ortalaması.
spkts	Kaynak paket sayısı
swin	Kaynak TCP pencere değeri
synack	TCP bağlantısı kurulum zamanı SYN ve SYN_ACK paketleri arasındaki süre.
tcprtt	TCP bağlantısı kurulumu gidiş-dönüş süresi 'SYN_ACK' ve 'ACKDAT' toplamı.
trans_depth	Http request / response transaction bağlantısındaki derinlik.

alan adımları incelendiği zaman, sapma oranı hızlı bir şekilde azalmasına rağmen 800. adımdan sonra artmaya başlamaktadır. Sapma değeri eğitim aşamasında kullanılmayan örneklerin sınıflandırma performansının ölçülmesi ile hesaplanmış olmasından dolayı 0.2 öğrenme değerine sahip olan sınıflandırma modelinin aşırı öğrenme veya ezberleme problemine sahip olduğu sonucuna ulaşılabilmektedir.

Aynı şekilde en düşük öğrenme oranı olan 0.005'e sahip olan sınıflandırma modelinin eğitim aşamasında sapma değerinin daha akıcı bir şekilde ilerlemesine rağmen değerin düşük olması sebebiyle diğer modeller gibi düşük sapma değerine ulaşamadığı gözlenmektedir.

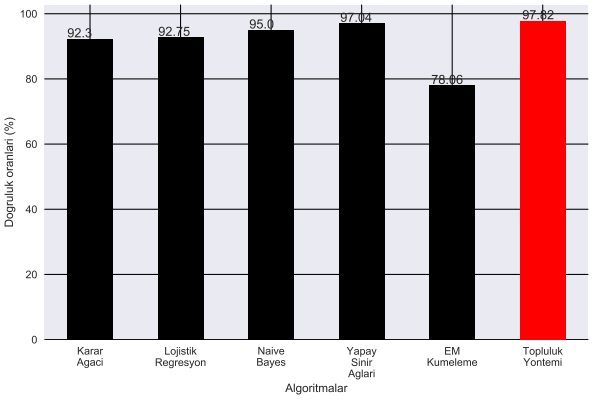
Bu çalışma kapsamında en optimal değerin 0.05 öğrenme oranına sahip olan model ile elde edildiği sonucuna ulaşılmıştır. Şekil 1 incelendiği zaman 5000. adım olan son yineleme aşamasında algoritmanın oluşturmuş olduğu sınıflandırma modeli en düşük sapma değerine ulaştığı gözlemlenmektedir.

Tablo 3'te en güçlü sınıflandırma performansına sahip olan modelin değerlendirme sonuçları yer almaktadır.

Tablo 3. En güçlü sınıflandırma performansına sahip modelin değerlendirme sonuçları.

Sınıf	Hassaslık	Geri Çekilme	F_1	Doğruluk
Normal	0.98	1.00	0.99	0.9782
Saldırı	0.97	0.79	0.87	0.9782

Şekil 2 orjinal yayında [14] elde edilen sonuçlarla bu çalışma kapsamında elde edilen sonuçları karşılaştırmaktadır. Bu çalışma kapsamında daha yüksek doğruluk oranlarına ulaşılmıştır.



Şekil 2. Önerilen yöntem ile UNSW veri kümesinin kullanıldığı çalışma ile elde edilen sınıflandırma sonuçlarının karşılaştırılması.

3.2. Nitelik önemleri

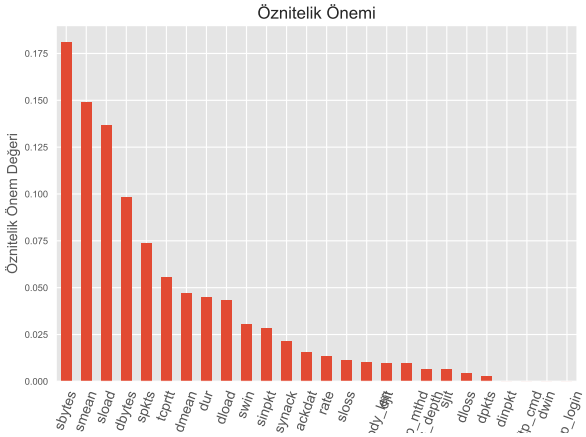
Nitelik önemi, bir sınıflandırma modeli içinde yer alan karar ağaçlarının oluşturulmasında her bir niteliğin, sınıflandırma performansına ne kadar faydalı veya değerli olduğunu gösteren bir değerdir. Bu önem değeri, veri kümesinde yer alan her özellik için açıkça belirtilerek, niteliklerin sıralanmasına ve birbirleriyle karşılaştırılmasına olanak tanımaktadır.

Önem, tek bir karar ağacı için, her nitelik ayırım nok-

tasının performans ölçüsünü artırdığı ve ilgili düğümün sorumlu olduğu gözlem sayısına göre ağırlık olarak hesaplanmaktadır. Burada ifade edilen performans ölçütü, bölme noktalarını seçmek için kullanılan bilgide saflık (Gini dizini) veya başka daha spesifik bir hata fonksiyonu olabilmektedir.

Özellik önemleri daha sonra model içindeki tüm karar ağaçlarında normalizasyon işlemi tabii tutulmaktadır. Önem değerleri 0-1 arasında normalize edilirse, nitelik filtreleme işlemi için önem puanları için bir eşik değeri belirlenebilir.

Şekil 3 ve Tablo 4'de eğitim aşamasında kullanılan veri kümesinde yer alan niteliklerin bu çalışma kapsamında oluşturulan modele göre önem değerleri gösterilmektedir. Son üç nitelik olan "ct_ftp_cmd", "dwin" ve "is_ftp_login" niteliklerinin önem değerinin 0 olduğu elde edilmiştir. Bu nedenle bu üç niteliğin sınıflandırma modeli oluşturulması esnasında kullanılmasının, performansa herhangibir katkısı olmadığı sonucuna ulaşılmaktadır.



Şekil 3. Veri kümesinde bulunan niteliklerin önem değerleri.

4. Tartışma ve Sonuç

Bu çalışma kapsamında günümüzde en sık karşılaşılan siber saldırılardan biri olan dağıtık hizmet dışı bırakma saldırısının makine öğrenme yöntemleri kullanılarak algılanabilmesi için bir model önerilmiştir. Önerilen model topluluk modeli temel alınarak geliştirilmiş olması sebebiyle sınıflandırma performansının yüksek olduğu sonuçlarına varılmıştır.

Açık veri kümesinin alındığı üniversitenin ilgili çalışmasında karar ağacı, lojistik regresyon, naive bayes, yapay sinir ağları ve beklenti maksimizasyonu şeklinde beş farklı sınıflandırma algoritması kullanılmıştır. Elde ettikleri en yüksek doğruluk oranının %97.04 şeklinde olduğunu ifade etmişlerdir. Bu çalışma kapsamında elde edilen sonuç ise %97.82 şeklindedir. Bu şekilde az bir miktarda olsa model doğruluğu artmıştır.

Önerilen yöntemin yüksek boyutlu siber güvenlik alanında kullanılan veri kümelerine uygulanabilir olduğu gösterilmiştir. Bu yöntem kullanılarak hizmet dışı bırakma saldırısının algılanmasında kaynak IP adresi gibi yanıltıcı alanlara bakılmadan, gelen paketler içerisinde yer alan büyüklük, varış süresi, yük boyutu gibi hizmet sunucuya gelen isteklerin niteliklerine bakılarak model oluşturmaktadır. Bu sebeple paketler üzerinde saldırganlar tarafında IP, MAC adres sahteciliği gibi yöntemlerden etkilenmemektedir.

Tablo 4. Veri kümesinde bulunan nitelikler.

Nitelik	Önem değeri
smean	0.163028
sbytes	0.138580
sload	0.115149
dbytes	0.076136
dmean	0.060468
dur	0.056705
tcprtt	0.055894
spkts	0.052250
sinpkt	0.041140
synack	0.031071
response_body_len	0.030887
dpkts	0.030480
dload	0.022183
ackdat	0.021479
ct_flw_http_mthd	0.021305
rate	0.020139
djit	0.017830
swin	0.015987
dloss	0.009649
trans_depth	0.007482
sjit	0.005635
sloss	0.004320
dinpkt	0.002203
ct_ftp_cmd	0.000000
dwin	0.000000
is_ftp_login	0.000000

Literatürde yer alan diğer makine öğrenme yöntemlerine dayalı saldırı tespit sistemlerinde bir sınıflandırma hipotezine dayalı model oluşturulmaya çalışılmaktadır. Bu çalışma kapsamında geliştirilen model ise birden fazla, bulgular kısmında gösterildiği gibi 5000 adet gibi oldukça yüksek sayıda sınıflandırma aracı kullanarak sonuca ulaşmaktadır. Bu nedenle diğer yapılan çalışmalardan oldukça farklılık göstermektedir.

Gelecek çalışma olarak nitelik çıkarımının danışmansız olarak yapılabildiği auto-encoder yöntemleri kullanılarak oluşturulacak olan veri kümesinin derin öğrenme yöntemleri ile eğitilmesi amaçlanmaktadır. Bu şekilde model sınıflandırma performansının artacağı değerlendirilmektedir.

Kaynakça

[1] Zargar,S. T., Joshi, J., Tipper, D. 2013. A Survey of Defense Mechanisms Against Distributed Denial of Service (DDoS) Flooding Attacks. IEEE Communications Surveys Tutorials, 15(4), 2046-2069.

[2] Bhatia, S. 2016. Ensemble-based model for DDoS at-

tack detection and flash event separation. Future Technologies Conference (FTC), 2016, 958-967

[3] Kshirsagar, D., Sawant, S., Rathod, A., Wathor, S. 2016. CPU Load Analysis &; Minimization for TCP SYN Flood Detection. Procedia Computer Science, 85(2016), 626-633.

[4] Katkar,V. D., Kulkarni, S. V. 2013. Experiments on detection of Denial of Service attacks using ensemble of classifiers. International Conference on Green Computing, Communication and Conservation of Energy (ICGCE), 2013, 837-842

[5] Osanaiye, O., Cai, H., Choo, K.R., Dehghantanha, A., Xu, Z., Dlodlo, M. 2016. Ensemble-based multi-filter feature selection method for DDoS detection in cloud computing. EURASIP Journal on Wireless Communications and Networking 1(2016), 130-140.

[6] Livadas, C., Walsh, R., Lapsley, D., Strayer, W. T.2006, Usilng Machine Learning Technliques to Identify Botnet Traffic. International Conference on Local Computer Networks, 1-16 Kasım,967-974.

[7] 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(A), 2-16.

[8] Saad, S., Traore, I., Ghorbani, A., Sayed B., Zhao, D., Lu, W., Felix, J., Hakimian, P. 2011, Detecting P2P botnets through network behavior analysis and machine learning. International Conference on Privacy, Security and Trust, 19-21 Temmuz, Montreal, 174-180.

[9] Lu, W. , Rammidi, G., Ghorbani, A. A. 2011. Clustering botnet communication traffic based on n-gram feature selection. Computer Communications, 34(3), 502-514.

[10] Masud, M. M., Al-khateeb, T., Khan, L. , Thuraisingham, B., Hamlen, K. W. 2008, Flow-based identification of botnet traffic by mining multiple log files. International Conference on Distributed Framework and Applications, 21-28 Ekim, Penang, 200-206.

[11] Friedman, J. H. 2001. Greedy function approximation: a gradient boosting machine. Annals of statistics, 2001, 1189-1232.

[12] Wang, H., He, X., Chang, M.-W., Song, Y., White, R. W., Chu, W. 2013, Personalized Ranking Model Adaptation for Web Search. International ACM SIGIR Conference on Research and Development, New York, 323-332.

[13] Brillante, L., Gaiotti, F., Lovat, L., Vincenzi, S., Giacosa, S., Torchio, F., Segade, S. R., Rolle, L., Tomasi,D. 2015. Investigating the use of gradient boosting machine, random forest and their ensemble to predict skin flavonoid content from berry physical–mechanical characteristics in wine grapes. Computers and Electronics in Agriculture, 117, 186-193.

[14] Moustafa, N., Slay, J., 2016. The evaluation of Network Anomaly Detection Systems: Statistical analysis of the UNSW-NB15 data set and the comparison with the KDD99 data set. Information Security Journal: A Global Perspective, 25, 1-3.

[15] https://www.ixiacom.com/products/perfectstorm. 2017 PerfectStorm. (Erişim Tarihi: 01.06.2017)

[16] qosient.com. 2017 Argus-3.0.8.2. (Erişim Tarihi: 01.06.2017)

[17] tcpdump.org. 2015. tcpdump and libpcap latest release. (Erişim Tarihi: 15.04.2017).