Bilgisayar ağ güvenliğinde naive bayes algoritmasının kullanımı

Şaban Gülcü

Bilgisayar Mühendisliği Bölümü, Necmettin Erbakan Üniversitesi, Konya, Türkiye, sgulcu@konya.edu.tr

**Özet:** Makine öğrenmesi bilgisayar bilimlerinin önemli bir alt dalıdır ve verilerden öğrenen ve veriler üzerinde tahminler yapan algoritmaların geliştirilmesini araştırmaktadır. Bu tür algoritmalar, çıktı olarak verilere dayalı tahminler yapmak için giriş gözlemlerinin örnek bir eğitim kümesinden bir model oluşturarak çalışırlar. Makine öğrenmesi algoritmalarının kullanıldığı alanlardan bir tanesi de bilgisayar ağ güvenliğidir. Bu çalışmada, ağ saldırı tespitinde naive bayes algoritmasının güncel veriler üzerinde kullanımı incelenmektedir. Deneylerde veri kümesi olarak Kyoto 2016 ve KDD’99 verileri kullanılmıştır. Kyoto 2016 veri kümesinde 309060 tane bağlantı bulunmaktadır ve her bir bağlantı 23 parametreden oluşmaktadır. Algoritmanın başarısını test etmek için 10-katmanlı çapraz doğrulama yöntemi kullanılmıştır. KDD’99 veri kümesinde ise 444616 tane bağlantı bulunmaktadır ve her bir bağlantı 41 parametreden oluşmaktadır. Ayrıca, KDD’99 veri kümesinde saldırının tipi de sınıflandırılmaktadır. Deneylerde, algoritmanın sınıflandırma başarı oranları verilmektedir. Deney sonuçlarına göre, naive bayes algoritmasının saldırı tespitinde başarılı olduğu görülmektedir.

**Anahtar Kelimeler:** ağ saldırısı tespiti, bilgisayar ağ güvenliği, makine öğrenmesi, naive bayes algoritması

The usage of naive bayes algorithm in computer network security

**Abstract:** Machine learning is an important subdivision of computer science and explores the development of algorithms that learn from the data and make estimates on the data. Such algorithms work by creating a model from a sample set of input observations to make predictions based on the output. One of the areas where machine learning algorithms are used is also computer network security. In this paper, the usage of the naive bayes algorithm is analyzed on current data for network intrusion detection. In the experiments, data of Kyoto 2016 and KDD'99 were used as data set. The Kyoto 2016 dataset contains 309060 connections and each connection consists of 23 parameters. The 10-fold cross-validation method was used to test the success of the algorithm. KDD'99 dataset contains 444616 connections and each connection consists of 41 parameters. In addition, the type of attack in the KDD'99 dataset is also classified. In the experiments, the classification success rates of the algorithm are given. According to the experimental results, naive bayes algorithm is determined to be successful in the network intrusion detection.

**Keywords:** network intrusion detection, computer network security, machine learning, naive bayes algorithm

# Giriş

Makine öğrenmesi bilgisayar bilimlerinin önemli bir alt dalıdır ve verilerden öğrenen ve veriler üzerinde tahminler yapan algoritmaların geliştirilmesini araştırmaktadır. 1959 yılında Arthur Samuel makine öğrenmesini şu şekilde tanımlamıştır: “Açıkça programlanmış olmadan bilgisayarlara öğrenme yeteneği veren bir çalışma alanıdır” (Simon, 2013). Bu tür algoritmalar, çıktı olarak verilere dayalı tahminler yapmak için giriş gözlemlerinin örnek bir eğitim kümesinden bir model oluşturarak çalışırlar. Makine öğrenmesi spam filtreleme (Guzella & Caminhas, 2009; Kumar, Gao, Welch, & Mansoori, 2016), biyomedikal (Jones, Ghandehari, & Facelli, 2016), fizik (Ling, Jones, & Templeton, 2016), malzeme bilimi (Chowdhury, Kautz, Yener, & Lewis, 2016), enerji (Sánchez-Oro, Duarte, & Salcedo-Sanz, 2016), imalat (Chou & Thedja, 2016) gibi geniş bir alanda başarılı bir şekilde uygulanmaktadır. Makine öğrenmesi yöntemleri genellikle üç ana grupta toplanmaktadır:

* Gözetimli öğrenme: Bir örnek girdi verisi ve bunların bilinen çıktıları yönteme yüklenmektedir ve amaç bu girdi-çıktılar arasındaki genel kuralı öğrenmektir.
* Gözetimsiz öğrenme: Öğrenme algoritmasına girdi verileri dışında hiçbir uyarı/bilgi verilmez. Algoritmanın kendi kendine bu girdilerdeki yapıyı bulması amaçlanmaktadır.
* Takviyeli öğrenme: Bulunduğu ortamı algılayan ve kendi başına kararlar alabilen bir sistemin, hedefine ulaşabilmesinde doğru kararlar almayı nasıl öğrenebileceğini gösterir.

Siber güvenlik veya IT güvenliği olarak da bilinen bilgisayar güvenliği; donanıma ve yazılıma verilen zararlardan ve hırsızlıktan bilgi sistemlerinin korunmasını amaçlamaktadır. Bunun yanı sıra, bu sistemlerin sağladıkları hizmetlerin yanlış yönlendirilmesi, aksatılması veya bozulması gibi konuları da kapsamaktadır. Bu alan; birçok toplumun bilgisayar otomasyon sistemlerine bağlılığının olmasından dolayı artan bir öneme sahiptir. Ayrıca günümüzde internetin, kablosuz ağların ve akıllı telefonlar, televizyonlar gibi akıllı cihazların kullanımının artması özellikle ağ güvenliği alanının önem derecesini artırmaktadır. Ağ güvenliği; ağ kaynaklarının hatalı kullanımını, değişikliklerini ve izinsiz girişlerini gözlemeyi/engellemeyi içeren uygulama ve kurallardan oluşur. Ağ saldırısı, bir bilgisayar ağındaki herhangi bir izinsiz aktivite olarak tanımlanabilir. Bu izinsiz aktiviteyi tespit edebilme, saldırıların nasıl çalıştığını iyi bilen savunuculara bağlıdır. Bu tür istenmeyen aktivite, diğer kullanımlar için tasarlanmış ağ kaynaklarını tüketir ve ağın güvenliğini tehdit eder. McAfee’nin raporuna (Labs, 2017) göre, 2017 yılının üçüncü çeyreğinde dünyada en fazla yapılan ağ saldırı tipi Şekil 1’de gösterilmiştir. Bu nedenle, ağ saldırı tespit sistemlerine ihtiyaç duyulmaktadır ve bunların düzgün tasarlanması davetsiz misafirleri engellemede yardımcı olmaktadır. Bu sistemler ağdaki bütün cihazların trafiğini izlemekle ve analiz etmekle sorumludur. Bir saldırı tespit edilir edilmez veya anormal bir davranış algılanır algılanmaz sistem yöneticisine hemen bir uyarı gönderilir.

Makine öğrenmesi son yılların önemli araştırma konularından biridir ve ağ güvenliği alanında da uygulanmaktadır. Yapay bağışıklık sistemi (Chen, Chang, & Wu, 2016), bulanık mantık (Balan, Priyan, Gokulnath, & Devi, 2015), naive bayes (Scott, 2004), destek vektör makinesi (Chitrakar & Huang, 2014) gibi farklı birçok makine öğrenmesi algoritması ağ saldırı tespitinde kullanılmaktadır. Bu çalışmada ağ saldırı tespitinde naive bayes sınıflandırıcısının güncel veriler üzerinde kullanımı incelenmektedir. KDD Kupa 1999 verileri (KDD’99) (UCIKDD, 1999) ve Kyoto 2006+ (Song et al., 2011) ağ saldırı tespitinde sıkça kullanılan ve iyi bilinen veri kümeleridir. KDD’99 veri kümesi bir askeri ağ ortamında simüle edilen günlük hayatta karşılaşılabilecek çok çeşitli saldırıları içermektedir. Fakat KDD’99 ve Kyoto 2006+ veri kümeleri, çok önceleri oluşturulduğu için siber saldırıların son zamanlardaki eğilimlerini yansıtmamaktadır. Kyoto Üniversitesi 2017 yılında Kyoto 2016 (Tada, Kobayashi, Shimada, & Takakura) isimli veri kümesini yayınlamıştır. Bundan dolayı bu çalışmadaki deneylerde veri kümesi olarak Kyoto 2016 ve KDD’99 verileri kullanılmaktadır.

|  |
| --- |
|  |

Şekil 1. 2017’de en fazla yapılan ağ saldırı çeşitleri.

Makalenin devamı şu şekilde organize edilmiştir: Bölüm 2 literatürdeki ağ saldırısı tespitinde kullanılan yöntemler hakkında detaylı bir bilgi vermektedir. Naive bayes sınıflandırıcısı Bölüm 3’de anlatılmıştır. Bölüm 4 deneysel sonuçları göstermektedir. Bölüm 5’de makale sonuçlandırılmıştır.

# İlgili çalışmalar

Ağ saldırı tespiti konusunda hem konunun ilgi çekici olması hem güvenliğin hayatımızda vazgeçilmez bir unsur olması hem de adli bilişimin bir alt çalışma alanı olmasından dolayı birçok çalışma yapılmıştır. Saldırı türlerinin sürekli değişmesinden dolayı bu saldırıların tespitinde makine öğrenmesi yöntemlerinin kullanılması gittikçe önem arz etmektedir. Makine öğrenmesinde naive bayes sınıflandırıcıları; özellikler arasındaki güçlü bağımsız varsayımları ile bayes teoremine dayalı basit olasılık sınıflandırıcılarının bir ailesidir. 1950’den itibaren naive bayes üzerine çalışmalar yapılmaktadır ve bunlardan birkaçı belge sınıflandırma (Jiang, Li, Wang, & Zhang, 2016), spam mail filtreleme (Esmaeili, Arjomandzadeh, Shams, & Zahedi, 2017; Guzella & Caminhas, 2009) ve ağ saldırı tespitidir (Scott, 2004). Scott ağ saldırı tespit sistemi tasarımında naive bayes teorem tabanlı bir model geliştirmiştir (Scott, 2004). Birçok saldırı tespiti algoritması bireysel bağlantıları modellemeye odaklanmıştır. Kullanıcı davranışına göre modelleme hesaba özel olmalıdır ve kullanıcı davranışı değiştikçe kolay bir şekilde güncellenebilmelidir. Ayrıca geniş bir kullanıcı kitlesini kapsayacak şekilde oldukça esnek olmalıdır. Scott, deneylerde bir bilgisayar ağının verileri ve bir haberleşme ağının verileri olmak üzere iki çeşit veri kümesi kullanmıştır. Li ve Guo (Li & Guo, 2007), k-enyakın komşu (k-Nearest Neighbour) algoritmasına dayalı gözetimli ağ saldırı tespit yöntemi önermişlerdir. Önerilen yöntem geleneksel gözetimli ağ saldırı tespit yöntemleri ile karşılaştırılmıştır. Deneyler KDD’99 verileri üzerinde gerçekleştirilmiş ve önerilen yöntemin diğer yöntemlerden daha etkili ve kararlı olduğu gösterilmiş. Hanguang ve Yu (Hanguang & Yu, 2012), makine öğrenmesi birliktelik kuralları algoritmasından biri olan Apriori algoritmasını web tabanlı saldırı tespit sistemi için önermişlerdir. Apriori algoritması kuralları üretmektedir ve bu kurallar saldırının çeşidini belirlemede kullanılmaktadır. Modi ve arkadaşları (Modi et al., 2013) bulut kaynakları ve hizmetlerinin kullanılabilirliğini, gizliliğini ve bütünlüğünü etkileyen farklı saldırıları incelemişlerdir. Ayrıca, buluttaki saldırı tespit sistemlerini ve saldırı engelleme sistemlerini içeren önerileri araştırmışlardır. Modi ve arkadaşları gelecek nesil ağlardaki arzu edilen güvenliği sağlamak için bulutta çalışan bir saldırı tespit ve saldırı engelleme sistemi önermişlerdir. Chitrakar ve Huang (Chitrakar & Huang, 2014), ağ saldırı tespiti için Aday Destek Vektör isimli bir destek vektör makinesi algoritması önermişlerdir. Deneylerde KDD’99 ve Kyoto 2006 veri setini kullanmışlar ve sonuçları Artımsal Destek Vektör Makinesi ile karşılaştırmışlardır. Deney sonuçları önerilen yöntemin gerçek zamanlı ağ saldırı tespitinde başarılı olduğunu göstermiştir. Ravale ve arkadaşları (Ravale, Marathe, & Padiya, 2015) saldırı tespitinde K-Means algoritmasını ve destek vektör makinesi algoritmasını kullanarak hibrit bir yöntem önermişlerdir. Önerilen yöntemin amacı saldırı tespitinde kontrol edilen parametrelerin sayısını azaltmaktır. Böylece daha hızlı karar veren bir sistem oluşturmuşlardır. Deneylerde, önerilen yöntemi KDD’99 veri kümesi üzerinde test etmişler. Önerilen yöntem, belirleme oranı ve doğruluk açısından başarılı sonuçlar elde etmiştir. Son yıllardaki popüler konulardan birisi bulut bilişimdir ve günümüzde IT şirketleri alt yapılarını bulut bilişime taşıyarak teknoloji masraflarını düşürmektedirler. Rajendran ve arkadaşları (Rajendran, Muthukumar, & Nagarajan, 2015) bulut bilişim alanında kullanılan hibrit bir saldırı tespiti yöntemi önermişlerdir. Mobil ağların öneminin artmasından dolayı, Balan ve arkadaşları mobil ağlar için bulanık mantık tabanlı bir saldırı tespit sistemi önermişlerdir (Balan et al., 2015). Son yıllarda, bilgisayar bilimlerinde biyolojik ve evrimsel algoritmalar üzerine çokça çalışmalar yapılmaktadır. Chen ve arkadaşları (Chen et al., 2016), biyolojik tabanlı bir algoritma olan yapay bağışıklık sistemi algoritmasına dayalı bir ağ saldırı tespiti yöntemi önermişlerdir. Deneylerde KDD’99 veri setini kullanmışlar ve sonuçları birkaç yapay zeka yöntemi ile karşılaştırarak önerilen yöntemin başarısını test etmişlerdir. Varma ve arkadaşları (Varma, Kumari, & Kumar, 2016), karınca koloni algoritması ve bulanık entropi yöntemlerine dayalı gerçek zamanlı saldırı tespit sistemi önermişlerdir.

# Naive Bayes Sınıflandırıcısı

Bayes Teoremi 1701-1761 yılları arasında yaşayan Thomas Bayes tarafından geliştirilmiştir. Bu teorem, bir rassal değişken için olasılık dağılımı içinde koşullu olasılıklar ile marjinal olasılıklar arasındaki ilişkiyi gösterir. Koşullu olasılıkları hesaplayan basit bir matematiksel formüldür. Bir olayın ortaya çıkmasında birden fazla bağımsız nedenin etkili olması durumunda, bu nedenlerden herhangi birinin o olayı meydana getirme olasılığını hesaplamada kolaylık sağlar. Bayes teoremi; bir stokastik süreç sırasında ortaya çıkan bir rastgele A olayı ile bir diğer rastgele B olayı için koşullu olasılıkları ve marjinal olasılıkları arasındaki ilişkidir. Koşullu olasılık kavramı, bir olayın gerçekleşme olasılığının hesaplanmasında ek bilginin kullanılmasına olanak tanır. Bayes teoremi Denklem (1)’de gösterilmiştir.

|  |  |
| --- | --- |
|  | (1) |

Bayes teoremi formülü içinde bulunan her bir terime özel isimler verilmektedir:

* P(*A*); *A* için önsel olasılık veya marjinal olasılık adı verilir. Bu önseldir, çünkü *B* olayı hakkında önceden herhangi bir bilgiyi içermemektedir.
* P(*A*|*B*); verilmiş *B* olayı için *A* olayının koşullu olasılığı adını alır. Yani, *B* olayının olduğu bilindiğinde *A* olayının olma olasılığıdır.
* P(*B*|*A*); *A* olayının olduğu bilindiğinde *B* olayının olma olasılığıdır.
* P(*B*) terimi *B* olayı için önsel olasılıktır.

Naive bayes sınıflandırıcısı bayes teoreminin bağımsızlık önermesiyle basitleştirilmiş halidir. Naive bayes sınıflandırması makine öğrenmesinde gözetimli öğrenmenin alt sınıfındadır. Naive bayes sınıflandırmasında, sınıflandırılması gereken sınıflar (kümeler) ve örnek verilerin bu sınıflardan hangisine ait olduğu bellidir. Örnek olarak, ağa gelen her bir isteğin saldırı olup olmadığının tespiti işlemi verilebilir. “Saldırı” ve “normal” bağlantı ayrıştırılacak sınıfları temsil eder. Daha önceki ağ bağlantısı parametrelerinden yararlanarak gelecekteki ağ bağlantılarının saldırı olup olmadığına karar verecek bir algoritma da gözetimli makine öğrenmesine örnektir. Naive bayes sınıflandırıcısının kavram fonksiyonu Denklem (2)’de gösterilmektedir.

|  |  |
| --- | --- |
|  | (2) |

Burada *X* örnek kümesini ve *C* sonuç sınıf kümesini göstermektedir ve c ϵ *C*’dir. Ayrıca, **x** ϵ *X* olmak üzere **x**=<*x*1,*x*2,…,*x*n> nitelik vektörü ile tanımlıdır. Naive bayes sınıflandırıcısı matematiksel olarak şu şekilde ifade edilmektedir:

|  |  |
| --- | --- |
|  | (3) |

Bütün nitelikler birbirinden bağımsız olduğu için naive bayes varsayımı yeniden Denklem (4)’deki gibi yazılabilir.

|  |  |
| --- | --- |
|  | (4) |

Böylece naive bayes sınıflandırıcısı **x** nitelik vektörünü en yüksek olasılığa sahip olan c sınıfına atamaktadır. Naive bayes sınıflandırıcının sözde kodu Şekil 2’de gösterilmektedir.

|  |
| --- |
|  |

Şekil 2. Naive bayes sınıflandırıcının sözde kodu (Mitchell, 1997).

# Deneysel çalışmalar

Biz bu bölümde naive bayes sınıflandırıcı algoritmasını KDD’99 ve Kyoto 2016 verileri üzerinde kullanarak ağ saldırı tespitinin deneysel sonuçlarını sunmaktayız.

Amerika Savunma Bakanlığı İleri Araştırma Projeleri Ajansı (DARPA) sponsorluğu ile MIT Lincoln laboratuvarı, bilgisayar ağ saldırı tespiti sistemlerinin gelişimi için verileri toplamış ve hizmete sunmuştur (Mahoney & Chan, 2003). DARPA veri kümesi birçok saldırı tespit sisteminin değerlendirilmesinde ve testinde en çok kullanılan veri setidir. KDD’99 veri kümesi ise DARPA veri kümesinin bir alt kümesidir (Elkan, 2000). Bu veri kümesi 1998 DARPA veri kümesindeki tcpdump verilerinden 41 özelliğin seçilerek alınmasıyla oluşturulmuştur. Bu veri kümesindeki bir bağlantı, belirli zamanda başlayıp biten TCP paketlerinin dizisidir. Her bir bağlantı ya normal ya da bir saldırı (kesin olarak belirli bir saldırı tipi ile) olarak etiketlenmektedir. Saldırılar dört ana kategoriye ayrılır ve Tablo 1’de ana ve alt kategoriler gösterilmektedir. 22 saldırı türü ve bir de normal bağlantı olmak üzere veriler toplamda 23 sınıfta gruplanmaktadır. Dört ana kategori şu şekildedir:

* DoS: Servisleri iş göremez hale getirir.
* Remote to Local: Makinedeki kullanıcı yetkilerini ele geçirme
* User to Root: Normal kullanıcı hesabından root hesabına geçmeye çalışma
* Probing: Bilgisayar veya ağ bilgisini elde etmeye çalışma

Bağlantı kayıtları için tanımlanmış 41 tane özelliğin (parametre) tam bir listesi Tablo 2’de verilmiştir. Deneyler KDD’99 veri kümesinin 10% üzerinde yani 494021 tane bağlantı üzerinde gerçekleştirilmiştir. Algoritma Intel i3 işlemcili, 2.3 GHz hıza sahip ve 3 GB hafızalı bir bilgisayar üzerinde çalıştırılmıştır.

Tablo 1. Saldırı türleri.

|  |  |
| --- | --- |
| **4 ana saldırı sınıfı** | **22 saldırı sınıfı/türü** |
| Denial of service | back, land, neptune, pod, smurt, teardrop |
| Remote to local | ftp\_write, guess\_passwd, imap, multihop, phf, spy, warezclient, warezmaster |
| User to root | buffer\_overflow, perl, loadmodule, rootkit |
| Probing | ipsweep, nmap, portsweep, satan |

Tablo 2. Bağlantı kayıtları için tanımlanmış 41 tane özellik.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No** | **Özellik adı** | **Tip** | **No** | **Özellik adı** | **Tip** |
| 1 | duration | cont. | 22 | is\_guest\_login | disc. |
| 2 | protocol\_type | disc. | 23 | count | cont. |
| 3 | service | disc. | 24 | srv\_count | cont. |
| 4 | flag | disc. | 25 | serror\_rate | cont. |
| 5 | src\_bytes | cont. | 26 | srv\_serror\_rate | cont. |
| 6 | dst\_bytes | cont. | 27 | rerror\_rate | cont. |
| 7 | land | disc. | 28 | srv\_rerror\_rate | cont. |
| 8 | wrong\_fragment | cont. | 29 | same\_srv\_rate | cont. |
| 9 | urgent | cont. | 30 | diff\_srv\_rate | cont. |
| 10 | hot | cont. | 31 | srv\_diff\_host\_rate | cont. |
| 11 | num\_failed\_logins | cont. | 32 | dst\_host\_count | cont. |
| 12 | logged\_in | disc. | 33 | dst\_host\_srv\_count | cont. |
| 13 | num\_compromised | cont. | 34 | dst\_host\_same\_srv\_rate | cont. |
| 14 | root\_shell | cont. | 35 | dst\_host\_diff\_srv\_rate | cont. |
| 15 | su\_attempted | cont. | 36 | dst\_host\_same\_src\_port\_rate | cont. |
| 16 | num\_root | cont. | 37 | dst\_host\_srv\_diff\_host\_rate | cont. |
| 17 | num\_file\_creations | cont. | 38 | dst\_host\_serror\_rate | cont. |
| 18 | num\_shells | cont. | 39 | dst\_host\_srv\_serror\_rate | cont. |
| 19 | num\_access\_files | cont. | 40 | dst\_host\_rerror\_rate | cont. |
| 20 | num\_outbound\_cmds | cont. | 41 | dst\_host\_srv\_rerror\_rate | cont. |
| 21 | is\_host\_login | disc. |  |  |  |

Naive bayes sınıflandırıcısını eğitmek için veri kümesinden rasgele seçilmiş 90%’lık kısmı -yani 444616 tane bağlantı- kullanılmıştır. Geriye kalan 49405 adet bağlantı verisi naive bayes sınıflandırıcının performansını test etmek için kullanılmıştır. Deney sonuçları Tablo 3’de gösterilmiştir. Tabloda her bir sınıf için başarı oranları verilmiştir. Test verisinin 99.62%’si yani 49218 adet veri doğru sınıflandırılmıştır. Test verisinin 187 tanesi hatalı sınıflandırılmıştır ki bu da 0.38%’e tekabül etmektedir.

Tablo 3. Deney sonuçları

|  |  |  |  |
| --- | --- | --- | --- |
| **Sınıf adı** | **Doğru sayısı** | **Yanlış sayısı** | **Başarı oranı %** |
| back | 211 | 1 | 99.5 |
| buffer\_overflow | 1 | 1 | 50.0 |
| ftp\_write | 1 | 3 | 25.0 |
| guess\_passwd | 3 | 1 | 75.0 |
| imap | 0 | 1 | 0.0 |
| ipsweep | 101 | 24 | 80.8 |
| land | 2 | 0 | 100.0 |
| loadmodule | 1 | 6 | 14.3 |
| multihop | 0 | 16 | 0.0 |
| neptune | 10785 | 19 | 99.8 |
| nmap | 21 | 1 | 95.5 |
| normal | 9739 | 59 | 99.4 |
| perl | 1 | 0 | 100.0 |
| phf | 1 | 0 | 100.0 |
| pod | 25 | 1 | 96.2 |
| portsweep | 76 | 26 | 74.5 |
| rootkit | 1 | 19 | 5.0 |
| satan | 107 | 3 | 97.3 |
| smurf | 27961 | 0 | 100.0 |
| spy | 0 | 1 | 0.0 |
| teardrop | 96 | 0 | 100.0 |
| warezclient | 85 | 5 | 94.4 |
| warezmaster | 0 | 0 | - |
| Toplam: | 49218 | 187 | 99.6 |

Kyoto 2016 veri seti Kyoto Üniversitesindeki ağ trafik verilerinden elde edilmiştir ve Tablo 4’de verilen 24 tane özellikten oluşmaktadır. Veri setindeki *label* özelliği bağlantının bir saldırı olup olmadığını belirtmektedir. *label* değeri 1 ise bağlantı normal, -1 ise bağlantıda bilinen bir saldırı gözlenmiş, değeri -2 ise bağlantıda bilinmeyen bir saldırı gözlenmiştir. Biz bu çalışmada saldırı olup olmadığını tespit etmek istediğimiz için bilinen ve bilinmeyen saldırı tiplerini tek bir grup altında topladık ve -1 olarak etiketledik. Kyoto 2016 veri setinde veriler gün bazında tutulmaktadır. Deneylerde 31 Aralık 2015 gününün verileri kullanılmıştır ve bu veride 309060 tane bağlantı bulunmaktadır. Yöntemin başarısını test etmek için verileri eğitim verisi ve test verisi olarak ayırmak gerekmektedir. Bu ayırma işlemini 10-katmanlı çapraz doğrulama yöntemi ile gerçekleştirdik. Bu yöntemde veri kümesi 10 parçaya bölünmektedir ve 9 parça eğitim için, kalan 1 parça ise test için kullanılmaktadır. Yani 278154 adet veri eğitim için ve 30906 adet veri test için kullanılmıştır. Bu işlem 10 kez tekrar etmekte ve her seferinde daha önce test için kullanılmamış parça test için kullanılmaktadır. Böylece her bir parçanın test için kullanılması sağlanmaktadır.

Tablo 4. Kyoto 2016 veri setindeki 24 tane özellik.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No** | **Özellik adı** | **Tip** | **No** | **Özellik adı** | **Tip** |
| 1 | duration | cont. | 13 | dst\_host\_srv\_serror\_rate | cont. |
| 2 | service | disc. | 14 | flag | disc. |
| 3 | src\_bytes | cont. | 15 | ids\_detection | disc. |
| 4 | dst\_bytes | cont. | 16 | malware\_detection | disc. |
| 5 | count | cont. | 17 | ashula\_detection | disc. |
| 6 | same\_srv\_rate | cont. | 18 | label | disc. |
| 7 | serror\_rate | cont. | 19 | srv\_ip\_address | disc. |
| 8 | srv\_serror\_rate | cont. | 20 | srv\_port\_number | cont. |
| 9 | dst\_host\_count | cont. | 21 | dst\_ip\_address | disc. |
| 10 | dst\_host\_srv\_count | cont. | 22 | dst\_port\_number | cont. |
| 11 | dst\_host\_same\_src\_port\_rate | cont. | 23 | start\_time | disc. |
| 12 | dst\_host\_serror\_rate | cont. | 24 | protocol\_type | disc. |

Performans değerlendirmesi Denklem (5)’deki tespit oranı (TO) ve Denklem (6)’deki yanlış alarm oranı (YAO) performans metrikleri yardımıyla hesaplanmaktadır. Tespit oranı, duyarlılık olarak da tanımlanmaktadır ve doğru olarak tanımlanan pozitiflerin oranını ölçmektedir. Yanlış alarm oranı ise, belirli bir test için H0 hipotezini yanlışlıkla reddetme ihtimalini ölçmektedir.

|  |  |
| --- | --- |
|  | (5) |
|  | (6) |

Burada DP doğru pozitifi (saldırı, saldırı olarak tespit edildi), DN doğru negatifi (normal, normal olarak tespit edildi), YP yanlış pozitifi (normal, saldırı olarak tespit edildi) ve YN yanlış negatifi (saldırı, normal olarak tespit edildi) temsil etmektedir.

Kyoto 2016 veri seti üzerindeki deneysel sonuçlar Tablo 5’de sunulmaktadır. k-katman sütunu, 10-katmanlı çapraz doğrulama yönteminin kaçıncı katmanının gerçekleştirildiğini göstermektedir. DP, DN, YP, YN sütunları 30906 adet test verisinin doğru sınıflandırılıp sınıflandırılmadığını göstermektedir. Başarı oranı sütunu ise yöntemin doğru sınıflandırma yüzdesini vermektedir. TO ve YAO sütunları ise sırasıyla tespit oranını ve yanlış alarm oranını vermektedir. Sonuçlar incelendiğinde naive bayes algoritmasının saldırı tespitinde başarılı olduğu görülmektedir. Ayrıca, TO ve YAO’nun grafiksel gösterimi Şekil 3 ve Şekil 4’de sunulmaktadır.

Tablo 5. Kyoto 2016 veri setindeki deney sonuçları.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **k-katman** | **DP** | **DN** | **YP** | **YN** | **Başarı oranı (%)** | **TO (%)** | **YAO (%)** |
| 1 | 29564 | 1169 | 35 | 138 | 99.44 | 99.54 | 2.91 |
| 2 | 28607 | 2212 | 42 | 45 | 99.72 | 99.84 | 1.86 |
| 3 | 26407 | 4324 | 8 | 167 | 99.43 | 99.37 | 0.18 |
| 4 | 25764 | 4987 | 4 | 151 | 99.50 | 99.42 | 0.08 |
| 5 | 25606 | 5133 | 14 | 153 | 99.46 | 99.41 | 0.27 |
| 6 | 28847 | 1926 | 20 | 113 | 99.57 | 99.61 | 1.03 |
| 7 | 30015 | 853 | 34 | 4 | 99.88 | 99.99 | 3.83 |
| 8 | 29484 | 1355 | 56 | 11 | 99.78 | 99.96 | 3.97 |
| 9 | 30485 | 365 | 56 | 0 | 99.82 | 100.00 | 13.30 |
| 10 | 30430 | 401 | 67 | 8 | 99.76 | 99.97 | 14.32 |

|  |
| --- |
|  |

Şekil 3. Naive bayes yönteminin Tespit Oranı metriği açısından değerlendirilmesi.

|  |
| --- |
|  |

Şekil 4. Naive bayes yönteminin Yanlış Alarm Oranı metriği açısından değerlendirilmesi.

# Sonuç

Makine öğrenmesi, matematiksel ve istatistiksel yöntemler kullanarak mevcut verilerden çıkarımlar yapan ve bu çıkarımlarla bilinmeyene dair tahminlerde bulunan yöntemleri konu edinen bir bilim dalıdır. Günlük hayatımızın birçok alanında kullandığımız makine öğrenmesi algoritmaları ağ saldırı tespitinde de kullanılmaktadır. Bir bilgisayar ağındaki herhangi bir izinsiz aktivite olarak tanımlanan ağ saldırısı ağ kaynaklarını tüketmekte ve ağın güvenliğini tehdit etmektedir. Bu nedenle, ağ saldırı tespit sistemlerine ihtiyaç duyulmaktadır ve bunların düzgün tasarlanması davetsiz misafirleri engellemede yardımcı olmaktadır.

Biz bu çalışmamızda makine öğrenmesi algoritmalarından naive bayes sınıflandırıcı algoritmasını ağ saldırı tespitinin belirlenmesinde kullandık. Naive bayes sınıflandırıcı algoritması, daha önceki ağ bağlantısı parametrelerinden yararlanarak gelecekteki ağ bağlantılarının saldırı olup olmadığına karar vermektedir. Deneylerde veri kümesi olarak Kyoto 2016 ve KDD Kupa 1999 verileri kullanılmıştır. Kyoto 2016 veri kümesinde, her bir ağ bağlantısının 23 tane parametresi bulunmaktadır. Algoritma bu 23 parametreyi inceleyerek bağlantının saldırı olup olmadığına karar vermektedir. KDD Kupa 1999 veri kümesinde, her bir ağ bağlantısının 41 tane parametresi bulunmaktadır. Algoritma bu 41 parametreyi inceleyerek bağlantının saldırı olup olmadığını tespit etmekte ve 23 sınıftan birine atamaktadır. 23 sınıftan bir tanesi normal bağlantıyı temsil etmektedir ve diğer 22 sınıf saldırının çeşidini göstermektedir. Deneylerin sonucuna göre naive bayes sınıflandırıcısının ağ saldırı tespitinde başarılı olduğu görülmektedir.

Algoritmanın başarısı algoritmanın eğitimi aşamasındaki kullandığı verilere bağlıdır. Eğitim aşamasında ne kadar kapsamlı veri kümesi kullanılırsa, algoritma test ve kullanım aşamasında ağ saldırılarını başarılı bir şekilde tespit etmektedir. Bu nedenle, gelecekteki çalışma olarak farklı veri kümeleri kullanarak algoritmanın performansı test edilmelidir. Ayrıca, ağ saldırı tespitinde yapay sinir ağı gibi farklı makine öğrenmesi algoritmaları kullanılarak naive bayes sınıflandırıcı ile karşılaştırmaları yapılabilir.

**Referanslar**:

Balan, E. V., Priyan, M., Gokulnath, C., & Devi, G. U. (2015). Fuzzy based intrusion detection systems in MANET. *Procedia Computer Science, 50*, 109-114.

Chen, M.-H., Chang, P.-C., & Wu, J.-L. (2016). A population-based incremental learning approach with artificial immune system for network intrusion detection. *Engineering Applications of Artificial Intelligence, 51*, 171-181.

Chitrakar, R., & Huang, C. (2014). Selection of Candidate Support Vectors in incremental SVM for network intrusion detection. *computers & security, 45*, 231-241.

Chou, J.-S., & Thedja, J. P. P. (2016). Metaheuristic optimization within machine learning-based classification system for early warnings related to geotechnical problems. *Automation in Construction, 68*, 65-80.

Chowdhury, A., Kautz, E., Yener, B., & Lewis, D. (2016). Image driven machine learning methods for microstructure recognition. *Computational Materials Science, 123*, 176-187.

Elkan, C. (2000). Results of the KDD'99 classifier learning. *ACM SIGKDD Explorations Newsletter, 1*(2), 63-64.

Esmaeili, M., Arjomandzadeh, A., Shams, R., & Zahedi, M. (2017). An Anti-Spam System using Naive Bayes Method and Feature Selection Methods. *International Journal of Computer Applications, 165*(4).

Guzella, T. S., & Caminhas, W. M. (2009). A review of machine learning approaches to spam filtering. *Expert Systems with Applications, 36*(7), 10206-10222.

Hanguang, L., & Yu, N. (2012). Intrusion detection technology research based on apriori algorithm. *Physics Procedia, 24*, 1615-1620.

Jiang, L., Li, C., Wang, S., & Zhang, L. (2016). Deep feature weighting for naive Bayes and its application to text classification. *Engineering Applications of Artificial Intelligence, 52*, 26-39.

Jones, D. E., Ghandehari, H., & Facelli, J. C. (2016). A review of the applications of data mining and machine learning for the prediction of biomedical properties of nanoparticles. *Computer Methods and Programs in Biomedicine, 132*, 93-103.

Kumar, S., Gao, X., Welch, I., & Mansoori, M. (2016). *A machine learning based web spam filtering approach.* Paper presented at the Advanced Information Networking and Applications (AINA), 2016 IEEE 30th International Conference on.

Labs, M. (2017). Threats report. *https://www.mcafee.com/us/resources/reports/rp-quarterly-threats-dec-2017.pdf*.

Li, Y., & Guo, L. (2007). An active learning based TCM-KNN algorithm for supervised network intrusion detection. *computers & security, 26*(7), 459-467.

Ling, J., Jones, R., & Templeton, J. (2016). Machine learning strategies for systems with invariance properties. *Journal of Computational Physics, 318*, 22-35.

Mahoney, M. V., & Chan, P. K. (2003). *An analysis of the 1999 DARPA/Lincoln Laboratory evaluation data for network anomaly detection.* Paper presented at the International Workshop on Recent Advances in Intrusion Detection.

Mitchell, T. M. (1997). Machine learning. 1997. *Burr Ridge, IL: McGraw Hill, 45*, 37.

Modi, C., Patel, D., Borisaniya, B., Patel, H., Patel, A., & Rajarajan, M. (2013). A survey of intrusion detection techniques in cloud. *Journal of Network and Computer Applications, 36*(1), 42-57.

Rajendran, P. K., Muthukumar, B., & Nagarajan, G. (2015). Hybrid intrusion detection system for private cloud: a systematic approach. *Procedia Computer Science, 48*, 325-329.

Ravale, U., Marathe, N., & Padiya, P. (2015). Feature Selection Based Hybrid Anomaly Intrusion Detection System Using K Means and RBF Kernel Function. *Procedia Computer Science, 45*, 428-435.

Sánchez-Oro, J., Duarte, A., & Salcedo-Sanz, S. (2016). Robust total energy demand estimation with a hybrid Variable Neighborhood Search–Extreme Learning Machine algorithm. *Energy Conversion and Management, 123*, 445-452.

Scott, S. L. (2004). A Bayesian paradigm for designing intrusion detection systems. *Computational statistics & data analysis, 45*(1), 69-83.

Simon, P. (2013). *Too Big to Ignore: The Business Case for Big Data* (Vol. 72): John Wiley & Sons.

Song, J., Takakura, H., Okabe, Y., Eto, M., Inoue, D., & Nakao, K. (2011). *Statistical analysis of honeypot data and building of Kyoto 2006+ dataset for NIDS evaluation.* Paper presented at the Proceedings of the First Workshop on Building Analysis Datasets and Gathering Experience Returns for Security.

Tada, R., Kobayashi, R., Shimada, H., & Takakura, H. Kyoto 2016 Dataset.

UCIKDD. (1999). The third international knowledge discovery and data mining tools competition dataset KDD Cup 1999 data.

Varma, P. R. K., Kumari, V. V., & Kumar, S. S. (2016). Feature Selection Using Relative Fuzzy Entropy and Ant Colony Optimization Applied to Real-time Intrusion Detection System. *Procedia Computer Science, 85*, 503-510.