GÜFBED/*GUSTIJ* (2020) 10(2): 425-441 **Gümüşhane Üniversitesi Fen Bilimleri Enstitüsü Dergisi**
DOI: 10.17714/gumusfenbil.612774 Araştırma Makalesi / *Research Article*

Dağıtık Veritabanlarında Saldırı Önleme Metotları

Intrusion Prevention Methods in Distributed Databases

Çiğdem BAKIR*[a], Mehmet GÜÇLÜ[b], Veli HAKKOYMAZ[c], Banu DİRİ[d]
Yıldız Teknik Üniversitesi, Elektrik-Elektronik Fakültesi, Bilgisayar Mühendisliği Bölümü, 34220, İstanbul

• Geliş tarihi / *Received*: 29.08.2019 • Düzeltilerek geliş tarihi / *Received in revised form*: 18.02.2020 • Kabul tarihi / *Accepted*: 28.02.2020

Öz
Dağıtık sistemlerin kullanılmasıyla birlikte verilere farklı kullanıcılar farklı yerlerden anlık erişim sağlayabilmekte ve veriler üzerinde birtakım işlemler yapabilmektedir. Ancak, birden fazla kullanıcının aynı anda farklı noktalardan sisteme yetkisiz olarak erişmek istemesi veri güvenliği ve verinin gizliliği noktasında tehlikeli sonuçlar doğurabilmektedir. Bu çalışma, dağıtık veritabanları üzerine inşa edilmiş saldırı tespit ve önleme sistemleri üzerine olup, kullanılan metotların sınıflamasını yaparak, başarılarını analiz etmekte ve karşılaştırmalı olarak değerlendirmektedir. Üç kategori olarak sınıflandırılan yöntemlerden yapay zeka teknikleri içerisinde yer alan yapay bağışıklık algoritmasının veri madenciliği ve istatistiksel yöntemler içerisinde geçen diğer tekniklere oranla daha başarılı sonuçlar verdiği gözlenmiştir.

Anahtar kelimeler: Dağıtık Veritabanı, İstatistiksel Yöntemler, Saldırı Tespit Sistemleri, Saldırı Önleme Sistemleri, Veri Madenciliği, Yapay Zeka Yöntemleri

Abstract
With the use of distributed systems, different users can instantly access data from different locations and perform some operations on the data. However, the unauthorized access of multiple users to the system from different points at the same time can lead to dangerous results in terms of data security and confidentiality of the data. This study is based on intrusion detection and prevention systems built on distributed databases and classifies the methods used to analyze and evaluate successes comparatively. It is observed that the artificial immunity algorithm we have described in artificial intelligence techniques, which is one of the methods classified as three categories, gives more successful results compared to the other techniques mentioned in the data mining and statistical methods.

Keywords*: Distributed Database, Statistical Method, Intrusion Detection System, Intrusion Prevention System, Data Mining, Artifical Intelligence Method*

*[a] Çiğdem BAKIR; cigdem.bakr@gmail.com; Tel: (0212) 383 5730; orcid.org/0000-0001-8482-2412
[b] orcid.org/0000-0001-7507-5694 [c] orcid.org/0000-0002-3245-4440 [d] orcid.org/000-0002-4052-0049

1. Giriş

Gelişen teknoloji sayesinde verilere erişim ve veri iletişimi oldukça kolaylaşmıştır. Günümüzde Internet kullanım oranının artması, kullanılan alanının genişlemesi ve yapılan işlerin çeşitliliğinin artması neticesinde güvenlik konusu ciddi bir araştırma konusu haline gelmiştir. Dağıtık sistemlerin kullanılmasıyla birlikte verilere farklı kullanıcılar, farklı yerlerden anlık erişim sağlayabilmekte ve veriler üzerinde birtakım işlemler yapabilmektedir. Özellikle güvenlik alanında her gün yeni bir tehdit ile karşılaşılmakta, buna istinaden güvenlik önlemlerinde de hızlı bir gelişim yaşanmaktadır. Bu manada bilgisayar sistemlerinin güvenliğini sağlamak, yetkisiz erişim yapılmasının önüne geçmek bilgi güvenliği kapsamında kimlik doğrulama ve erişim kontrolleri için mekanizmalar geliştirmek gibi amaçlarla birçok yeni uygulamalar geliştirilmiştir. Internet ve haberleşme alanındaki gelişim süreçlerinin hız kazanması aynı zamanda kötü niyetli saldırganların zarar verebilecekleri daha çok sistemi ve buna bağlı olarak da çok daha fazla bilgiyi elde etme ihtimalini doğurmuş, işte bu sebeple saldırı sayısında ve saldırı tespit ve önleme metotlarının kullanımında ciddi ilgi artışları gözlemlenmiştir.

Genellikle saldırıların büyük çoğunluğu kullanılan sistemlerin açıklarından faydalanılarak gerçekleştirilmekte olup, bu tür saldırıların önlenmesi için güvenli bir ortam oluşturma ve de saldırıların zamanında tespit edilip önlenmesi gerekmektedir. Bir sistem ne kadar güvenli olursa olsun burada önemli olan saldırıların erken tespit edilip, olası sızmaların önüne geçmektir. Bu bağlamda, farklı kaynaklardan gelebilecek muhtemel her bir saldırı türü için geliştirilmiş ve geçmişten günümüze kadar araştırma ve geliştirme çalışmalarına konu olan çeşitli saldırı tespit ve önleme sistemleri mevcuttur. Bu çalışmada, saldırı tespit ve önleme sistemlerinin gelişim süreci boyunca varmış olduğu noktaya değinilmiş ve söz konusu sistemler sınıflandırılarak sınıf içi ve sınıflar arası performans değerlendirilmeleri gerçekleştirilmiştir. Makalenin kalan kısmı şu şekilde düzenlenmiştir: İlgili çalışmalar 2., veri seti tanıtımı 3., materyal ve metot 4., bulgular 5., değerlendirme 6., tartışma ve sonuç son bölümde verilmiştir.

2. İlgili Çalışmalar

Veritabanı saldırılarını önceden tahmin etme ve güvenlik açıklarını saldırı oluşmadan fark edebilme amacıyla çok etmenli istatiksel bir tahmin sistemi olan Quickprop sinir ağları geliştirilmiştir (Ramasubramanian vd., 2014). Bu çalışmada gizli katmanları hesaplayabilmek için Pearson korelasyon katsayısı kullanılmış ve bir banka verisi üzerinde yetkisi olmayan kullanıcılar belirlenmeye çalışılmıştır. Kısa vadede gerçekleşen anormal ve hatalı kullanıcı davranışları tespit edilebilmiştir.

Hatalı kullanıcı davranışlarını çözebilmek amacıyla yapılan bir başka çalışma da, genetik algoritmaların kullanımıdır (Romasubramanian vd., 2006). Genetik algoritma, sinir ağlarına dayalı olarak ağ özelliklerinden çeşitli kurallar oluşturarak elde edilen kurallar ile sınıflandırma yapar. Bu çalışmanın sonuçları diğer çalışmalarla karşılaştırılmalı olarak verilmiştir. Ancak, bu çalışma da öncekinde olduğu gibi kısa vadede gerçekleşebilecek saldırılar için bir çözüm önerisi getirmiştir.

Saklı Markov Modeli kullanılarak saldırıyı tahmin ve önleme çalışması, diğer bir çalışmadır (Haslum vd., 2007). Saklı Markov Modeli, verilen durumlardan yola çıkarak gizli durumları bulmak için gerçekleştirilen bir sınıflandırma algoritmasıdır. Dağıtık veriler birbirleriyle çok büyük ağlar üzerinde haberleşir ve bu sebeple ciddi saldırılara açıktır. Bu çalışmada fuzzy tekniği kullanılarak risk analizi yapılmış, tehlikeli olarak giden paket oranı tespit edilmeye çalışılmıştır. Ayrıca, dağıtık çevreler için risk oluşturacak saldırılar belirlenmeye çalışılmıştır.

Deng. ve çalışma arkadaşları (Deng vd., 2003) wireless ve geçici ağlar için (ad hoc networks) saldırı tespit sistemi için Support Vector Machine (SVM) tabanlı bir sistem geliştirmişlerdir. Kablosuz ve geçici ağlar için güvenlik sorunları SVM yöntemi ile çözülmeye çalışılmıştır. Bu çalışmada hiyerarşik ve tamamlayıcı dağıtık sistemler iki yaygın Denial of Service (DoS) saldırısı için geliştirilmiş ve performans ölçümleri yapılmıştır. DoS saldırısının hız, haberleşme mesafeleri arasındaki uzaklık değişimleri ve yer bilgisinin sistem performansına etkisi gözlemlenmiştir.

Jemili ve çalışma arkadaşları (Jemili, 2009) saldırı tahmin ve tespit sistemleri için Bayesian ağı ile olasılıkları birleştiren hibrit yayılma temeline dayalı yeni bir yaklaşım sunmuşlardır. Hibrit yayılma yaklaşımı hem normal hem de ağda oluşan anormal bağlantıları fark etmeye yarar. Bu çalışmanın amacı olası saldırı planlarını, senaryolarını ve aralarındaki ilişkileri ortaya

çıkarmaktır. Ayrıca, bu çalışma barındırma (host) tabanlı saldırı tespit sistemleri ile ağ tabanlı saldırı tespit sistemlerini birleştirerek veri tutarlılığının sağlanmasını gerçekleştirmektedir.

Hu ve çalışma arkadaşları (Hu vd., 2014) ağda meydana gelen değişikliklerden dolayı fark edilemeyen ve sıklıkla değişen ağ saldırıları için Particle Swarm Optimization (PSO) ve SVM tekniklerini birleştiren bir yöntem geliştirmişlerdir. Ayrıca, Adaboost temelli saldırı tespit algoritması olan Gaussian Mixture Model kullanarak dağıtık veritabanının her düğümü için saldırıların bulunmasını sağlamaya çalışmışlardır. Ancak, bu yaklaşım tüm saldırı türlerini özellikle de yeni saldırı türlerini tespit etmeyi tam olarak sağlayamamıştır.

Abraham ve arkadaşları (Abraham vd., 2007) saldırı tespit sistemlerinde evrimsel (genetik) algoritmaların kullanımı amaçlanmıştır. Çalışmada, verilen özellikleri kullanmak suretiyle otomatik bir saldırı tespit programı geliştirilmiştir. Çıktı programı ise küçük ve basittir. Saldırı tespiti için birçok makine öğrenme yöntemlerinin tüm özellikleri kullanılırken, önerilen çalışmada genetik programlamanın, az sayıda özelliği kullanılmıştır. Gelecek zamanda genetik programlama kullanılarak kablolu ağlar için saldırı tespit sistemi geliştiren bir çalışma önerilmiştir.

Sağıroğlu ve arkadaşları çalışmalarında (Sağıroğlu vd., 2011) yapay sinir ağı kullanılarak bir ağ üzerinde akan paketlerin hangi saldırı yöntemini kullandığını tespit etme amaçlanmıştır. Bu saldırılardan "Neptune" ve "the ping of death" bulunması için Çok Katman Algılayıcılı yapay sinir ağ modeli kullanılmıştır. DARPA veri setleri örnek alınarak ağlar için eğitmişlerdir. Bu çalışmanın neticesinde internetten gelebilecek DoS ataklarının algılanması başarı ile sağlanmıştır.

Saldırı tespit sistemleri için geliştirilmiş ve günümüze kadar kullanılan çeşitli veri madenciliği yöntemleri, yapay sinir ağları ve istatistiksel yöntemler gibi birçok yöntem kullanılmıştır.

3. Veri Seti Tanıtımı

Saldırı tespit sistemleriyle ilgili uygulamalarda en çok kullanılan veri setlerinden biri "KDD Cup'99" (Knowledge Discovery and Data Mining Cup'99) veri kümesidir. Bu veri kümesi üzerinde çok katmanlı yapay sinir ağlarının uygulanabilirliği test edilmiş ve paralel programlama ile performans analizleri yapılmıştır (Yıldırım vd., 2014). KDD-Cup 1999 verisi DoS, R2L, U2R ve Probe olmak üzere farklı türden saldırıları içeren bir veri setidir. Bu veri seti toplam da 972780 örnek içerir. DoS saldırı türünde saldırgan sunucuya sürekli sahte istekler göndererek sunucuyu meşgul eder. Bu durumda sunucu resmi yollarla istekte bulunan kullanıcıların isteklerine cevap veremez hale gelir. R2L (Remote to Local Attack) saldırı türünde erişim yetkisi bulunmayan saldırgan ağ üzerinden paket gönderir. Böylelikle sisteme erişimi sağlayarak gerekli verileri izinsiz olarak kullanır. U2R (User to Root Attack) saldırı türünde saldırgan kullanıcının şifresini ele geçirerek sisteme yöneticiymiş gibi erişir. Probe (Probing Attack) saldırılarında ise saldırgan sistemdeki güvenlik kontrollerini bozarak ağ üzerinden istediği verileri alır.

KDD-Cup 1999 veri seti, Darpa Bsm (Defense Advanced Research Projects Agency Basic Security Module) verisinden türetilen bir veri setidir. Darpa Bsm verisi Amerikan Hava Kuvvetlerine ait bir ağ trafiğinin simülasyonu sonucu oluşturulmuştur ve saldırı tespitinde yaygın bir şekilde kullanılmıştır. Toplamda 7 haftalık eğitim ve 2 haftalık test verisi olmak üzere 38 farklı saldırı türü içerir. Veri setinin içeriği saldırıya uğrayan makineden gece ve gündüz ağ dinleyicisinden alınan tcpdump dosyaları, saldırıya uğrayan makinelerden alınan kayıtların, log kayıtları ve güvenlik modülünden alınan dosyaları oluşturur (Liao vd., 2002).

4. Materyal ve Method

Dağıtık veritabanlarında saldırı tespit ve önleme sistemlerinde kullanılan yöntemler Tablo 1'de gösterildiği gibi üç ana grupta toplanmıştır: Veri madenciliği yöntemleri, İstatiksel yöntemler ve Yapay Zeka yöntemleri. Her bir yöntemdeki tekniklere izleyen alt bölümlerde yer verilmiştir. Ayrıca, tekniklerin performans değerleri grup içi ve gruplar arası olmak üzere değerlendirilmiştir.

4.1. Veri Madenciliği Yöntemleri

Bu grup altında K-Ortalama Kümeleme, En Yakın Komşuluk, Karar Ağaçları, Destek Vektör Makineleri ve Birliktelik Kuralları teknikleri yaygın bir şekilde kullanılmaktadır.

Tablo 1. Saldırı tespit ve önlemede yöntemler

Kullanılan Yöntemler	
Veri Madenciliği	K-Ortalama kümeleme K En Yakın Komşuluk (kNN) Karar Ağacı Destek Vektör Makineleri (SVM) Birliktelik Kuralları
İstatistiksel Yöntemler	Öğrenmeli Vektör Kuantalama (LVQ) Saklı Markov Modelleri (SMM) Naive Bayes Bulanık Mantık
Yapay Zeka Yöntemleri	Genetik Algoritma Yapay Sinir Ağları Yapay Bağışıklık Teknikleri

4.1.1. K-Ortalama Kümeleme

K-Ortalama Kümeleme tekniği nesneleri benzerliklerine göre gruplandıran eğitici olmayan öğrenme yöntemlerinden biridir. Bölünmeye dayalı bir yöntem olan K-Ortalama kümeleme, N tane nesneyi k küme merkezi uzaklığına göre hesaplar ve nesne hangi küme merkezine yakın ise o kümeye dahil eder. Küme merkezi başlangıçta rastgele bir veya daha fazla örneğin ortalaması alınarak belirlenir ve her iterasyonda yeniden hesaplanılır. Her defasında yeni küme merkezlerine göre tüm verilerin benzerlikleri bulunur. Bu şekilde benzer nesneler aynı kümeye diğer nesneler farklı kümelere alınır. Bu adımlar iteratif olarak tekrar eder. Kümeleme hata oranı (amaç fonksiyonu)minimum olduğunda adımlar sona erer. Saldırı tespit ve önlemede k-Ortalama; N adet saldırı türünü k adet kümeye böler. Bölme işlemi sonucunda elde edilen kümelerin, küme içi benzerlikleri maksimum olurken; kümeler arası benzerlikleri minumum olmalıdır. Bu yöntemin başarısı ve performansı başlangıçta rastgele seçilen k adet küme sayısı, küme merkezleri ve kullanılan benzerlik ölçütlerine göre değişmektedir. Bu sebeple yanlış alarm (false alarm) oranını bulmada çok başarılı sonuçlar vermemiştir (Sharma vd., 2018). Yanlış alarm oranı, saldırı olmayan bir verinin saldırıya uğrayan bir veri olarak kabul edilmesi yani hatalı olarak sınıflandırmasıyla da saldırıya uğrayan bir verinin normal bir veri olarak sınıflandırılma oranıdır.

K-Ortalama yöntemi saldırı tespit ve önleme sistemlerinde hesaplama karmaşıklığını azaltmak ve sınıflandırma başarısını arttırmak için kullanılmıştır (Faroun vd., 2007). KDD-Cup 1999 verisi üzerinde yapılan çalışmalarda k-Ortalama Kümeleme yöntemi yaklaşık %92 başarı sağlamıştır (Sharma vd., 2018). Nadiammai ve arkadaşları ise (Nadiammai vd., 2012) KDD-Cup 1999 verisi üzerinde k-Ortalama Kümeleme yöntemi ile %92.05 başarı elde etmişlerdir.

4.1.2. K En Yakın Komşuluk

K En Yakın Komşu yöntemi, en eski ve en basit eğiticili sınıflandırma yöntemidir. Verilen giriş vektörleri arasındaki uzaklığı hesaplar ve k adet en yakın komşusunun sınıfını seçer. Farklı k değerleri için farklı sınıflar bulunabilir. Bu sebeple sınıflandırma zamanı ve sınıflandırma doğruluğu için k parametresi oldukça önemlidir (Malhotra vd., 2017). Bu yöntemde sınıfı bulunmak istenen verinin, bilinen tüm verilere olan uzaklıkları hesaplanılır. Uzaklık ölçütü olarak genelde Öklid uzaklığı kullanılır. Rastgele k adet komşu sayısı belirlenir. Veri, en yakın k adet komşusuna bakılarak dahil olduğu sınıf belirlenir. Bu yöntemde, k-Ortalama Kümeleme yönteminden farklı olarak kullanılan veri setinin sınıfları bellidir ve yeni gelen veri bu sınıflara bakılarak bulunur.

kNN algoritması saldırı tespitinde, normal ve saldırgan türlere ait veri örneklerinin sınıflandırılması için kullanılmıştır (Malhotra vd., 2017). Sınıfı bulunacak verinin tüm veriye olan uzaklığı hesaplanılır. K adet verinin ortalamasına bakılarak eldeki verinin saldırı sınıfı belirlenir. Bu yöntemin başarısını ve performansını k adet komşu sayısı ve verinin sınıf sayısının dengeli olup olması etkiler.

Darpa Bsm verisi üzerinde sistem çağrılarının frekansına bakarak saldırgan sınıfa ait örnekleri tespit etmeye çalışmışlardır ve düşük yanlış pozitif oranı sağlamışlardır. Yanlış pozitif oranı saldırı içermeyen bir verinin saldırı türü olarak bulunması yani yanlış olarak sınıflandırılma oranıdır.

Saldırı tespitinde ve önlenmesinde hatalı ve normal olmayan verileri belirleyecek yanlış alarm oranını azaltmak için kNN sınıflandırıcı kullanılmıştır (Law vd., 2004). Bu çalışmada normal ve saldırgan olan verinin yanlış alarm modellerinin 5 yakın komşuluğu alınmıştır ve yüksek oranda başarı sağlamışlardır. KDD-Cup 1999 verisi üzerinde yapılan çalışmalarda kNN En Yakın Komşuluk yöntemi yaklaşık %96 başarı sağlamıştır (Senthilnayaki vd., 2019). Ayrıca, Aburomman ve arkadaşları (Aburonman vd., 2016) tarafından KDD-Cup 1999 verisi üzerinde yapılan çalışmada da yaklaşık %96 başarı elde

edilmiştir. Chen ve arkadaşları (Chen vd., 2016) ise KDD-Cup 1999 verisi üzerinde yaptıkları çalışmada %91.96 başarı sağlamışlardır.

4.1.3. Karar Ağaçları

Karar Ağacı tekniği veri madenciliğinde kullanılan ilk sınıflandırma algoritmalarından biridir. Sınıflandırma sonuçları daha kolay ve daha hızlı bir şekilde elde edilir. Bu teknikte her veri setinde sütunlar özellikleri gösterirken, her bir satırda bu özelliklerin değerleri tanımlanır. Ayrıca, her bir kayıt için özelliklerin değerlerine bağlı olarak sınıflar tanımlanır. Karar ağacında ilk yaklaşım özellikleri seçmektir. Daha sonrasında bu seçilen özelliğe göre sınıflar bölünür ve bu işlem iteratif olarak tekrarlanır.

Her düğüm bir özelliği gösterir ve bu düğümlerin çocuk düğümleri vardır (Moon vd., 2017). Karar ağacı tekniği kısacası iki aşamada gerçekleştirilir. İlk aşamada kök ve çocuk düğümlerden oluşan bir ağaç oluşturulur. İkinci aşamada ise bu ağacın yapısına göre çeşitli sınıflandırma kuralları çıkarılır. Bu sınıflandırma kuralları ağacın kök düğümü ile yaprakları arasında kalan düğümleri gösterir.

Ağ tabanlı saldırı tespit ve önleme sistemlerinde her düğüm, kullanıcı ya da verideki saldırı türlerini ya da normal olayları gösterir. Karar ağacı veri setini modelleyerek sınıflandırma problemini çözer. Hatalı saldırı tespitinde oluşturulan modele göre gelecekteki saldırı türlerini tahmin eder. Gerçek zamanlı saldırı tespitinde yüksek performans sağlar. Kural tabanlı çeşitli modelleri kullanır. Yeni saldırı türlerinin tespitinde ise oldukça başarılı sonuçlar verir. KDD-Cup 1999 verisi üzerinde yapılan çalışmalarda karar ağacı yöntemi yaklaşık %97 başarı sağlamıştır (Ugochukwu vd., 2018). Rachburee ve arkadaşları (Rachburee vd., 2017) KDD-Cup 1999 verisi üzerinde saldırı tespit sistemi gerçekleştirmişlerdir. Bu çalışmada Chi-Square yöntemi ile bu veri setinde özellik seçimi yapılarak saldırı türleri Karar Ağaçları ve Yapay Sinir Ağları ile sınıflandırılmıştır.

4.1.4. Destek Vektör Makineleri

Destek Vektör Makineleri (Support Vector Machines–SVM) verileri analiz etmede, sınıflamada kullanılan eğiticili öğrenmeye dayanan bir tekniktir. SVM başlangıçta iki sınıflı veri problemini çözmek için kullanılsa da daha sonra genişletilerek çok sınıflı veri probleminde kullanılmıştır. İki sınıflı veri probleminde iki sınıfı birbirinden ayırmak için bir model oluşturulur. Bu model fonksiyon oluşturularak elde edilir. Yeni gelen verinin hangi sınıfa ait olduğu bu fonksiyona göre belirlenir. SVM tekniğindeki amaç; iki sınıfı birbirinden ayıracak en uygun hiper düzlemi elde edecek fonksiyonu bulmaktır. Ayrıca, iki sınıfa ait destek vektörleri sınıfları birbirinden ayırmak için mümkün olduğunca maksimum olmalıdır.

SVM yüksek boyutlu uzayda her bir eğitim vektörünün sınıfını belirleyen sınıflandırma algoritmasıdır. SVM verinin destek vektörlerini belirleyecek sınıflar ve hiper düzlem ile sistemin çıkışını belirler. Eğitim anında destek vektörlerini doğrusal, polinomsal ya da sismoid fonksiyonlar ile belirler. SVM'in sınıfları birbirinden ayırması çeşitli parametrelere bağlı olarak değişir (Shams vd., 2018).

Saldırı tespit ve önlenmesinde Darpa Bsm verisi üzerinde saldırı türlerini sınıflandırmak için SVM kullanılmıştır (Mukkamala ve Januski, 2002). SVM'in saldırı tespitinde iki önemli rolü vardır. İlk ve en önemlisi saldırı tespiti için gerçek zamanlı performans sağlayarak sistemi eğitmesidir ve sistemin başarısını hesaplamasıdır. İkinci rolü ise sistemde meydana gelebilecek ölçekleme problemlerinin üstesinden gelmesidir. Ayrıca, SVM yüksek boyutlu uzayda ve karmaşık sınıflandırma problemlerinde oldukça başarılı sonuçlar vermektedir (Mukkamala vd., 2002). Chen ve arkadaşları (Chen vd., 2016) ise KDD-Cup 1999 verisi üzerinde yaptıkları çalışmada %92.46 başarı sağlamışlardır. Ayrıca, Aburomman ve arkadaşları (Aburomman vd., 2016) tarafından KDD-Cup 1999 verisi üzerinde yapılan çalışmada da %93.9 başarı elde edilmiştir.

4.1.5. Birliktelik Kuralları

Birliktelik Kuralları algoritması geçmiş verilere bakarak veriler arasındaki birliktelik davranışlarını inceleyen veri madenciliği tekniğidir. **T** transaction olmak üzere veritabanında tüm transaction işlemleri { T_1, T_2,…,T_n } ve işlem yapılan nesneler { i_1, i_2,…,i_m } ile ifade edelim. Veriler arasındaki ilişkisel kural X →Y (c,s) 'dir. Burada **s** veriler arasındaki destek (suport), **c** ise güven (confidence) aralığını gösterir. Destek bir nesnenin tüm nesneler için kullanılma sıklığını gösterir. Güven ise bir nesnenin diğer nesneyle kullanılma sıklığını gösterir. **s** birlikte yapılan X ve Y transaction işlemlerinin yüzdesini; **c** ise oranını belirtir (Tajbakhsh vd., 2009). Birliktelik kuralları en çok ürün satışında kullanılır. Örneğin; marketlerdeki

birlikte satılan ürünler bulunarak müşterilerin daha fazla satış yapılması sağlanabilir. Bu algoritmada ilk adım her bir transaction işlemi için sık kullanılan veri nesnelerini bulmak ve **s** güven aralığı için eşik değer belirlemektir. İkinci adım ise veri seti için uygun kuralları oluşturmaktır. Apriori algoritmasındaki temel problem çok sayıda kuralı oluşturmaktır. Ancak sık kullanılan veri nesneleri seçilerek bu veri nesneleri üzerinde oluşturulan kurallar sınırlandırılabilir. Algoritma adımları aşağıdaki gibidir:

- En küçük destek ve güven aralığı belirlenir.
- Her nesnenin destek değeri bulunur.
- En küçük destek değeri ile her nesnenin destek değeri kıyaslanır ve en küçük değerden küçük olanlar nesneler kümesinden atılır.
- İkili birliktelik kuralları oluşturulur ve aynı işlemler tekrarlanılır.

Saldırı tespit ve önlemede birliktelik kuralları kullanılarak yeni bir yaklaşım KDD-99 veri seti kullanılarak gerçekleştirilmiştir (Tajbakhsh vd., 2009). Bulanık ilişkisel kural seti ile birlikte farklı sınıfları belirleyen yeni sınıflandırma yaklaşımı oluşturulmuştur. Ayrıca, yeni veri nesneleri üzerinde ölçümler yapılabilecek etkili bir algoritma gerçekleştirilmesi hedeflemişlerdir. Tajbakhsh ve arkadaşları (Tajbakhsh vd., 2009) tarafından yapılan bu çalışmada %91 başarı sağlanmıştır. KDD-Cup 1999 verisi üzerinde yapılan çalışmalarda Birliktelik Kuralı yöntemi yaklaşık %96 başarı sağlamıştır (Abraham vd., 2007).

Olay kayıtları üzerinden okuma ve yazma işlemleri arasında ilişkisel kurallar oluşturarak güvenlik analizleri yapılmıştır (Hu vd., 2004). Bu çalışmada çok büyük veritabanı üzerinde güvenilir okuma ve yazma transaction işlemleri belirlemeye çalışarak sistemin performansı ölçülmüştür.

4.2. İstatiksel Yöntemler

Bu grup altında Öğrenmeli Vektör Kuantalama, Saklı Markov Modelleri, Naive Bayes ve Bulanık Mantık teknikleri yaygın bir şekilde kullanılmaktadır.

4.2.1. Öğrenmeli Vektör Kuantalama

Öğrenmeli Vektör Kuantalama (Learning Vector Quantization-LVQ) bir eğiticili sınıflandırma yöntemidir. Bu teknik Kohonen öğrenme kuralına göre gerçekleştirilir. LVQ tekniği diğer sınıflandırma algoritmalarından farklı olarak; n boyutlu girdi vektörünün hangi işlem elemanına ait vektörler ile temsil edilebileceğini bulur. Sınıfı bilinmeyen bir girdi vektörü için en yakın özellik vektörü bulunur. Bu vektörün bulunması öğrenme oranına ve maksimum eğitim sayısına bağlı olarak değişir. Girdi vektörüne en yakın işlem elamanı ödüllendirilir ve girdi vektörünün sınıfı bu işlem elemanına yaklaştırılır. Böylelikle bu işlem elemanı ödüllendirilir. Değil ise uzaklaştırılarak cezalandırılır. Bu şekilde özellik vektörleri güncellenir.

LVQ saldırı tespit ve önleme sistemlerinde sisteme giriş olarak verilen saldırı türlerini sınıflandırmak için bir katman kullanır. Bu katman giriş vektörlerinden bağımsızdır ve birbirine benzeyen giriş vektörlerini aynı sınıfta toplar (Noum vd., 2012). LVQ ağları ilk ve sonraki katman olmak üzere iki katmana sahiptir. İlk katman giriş vektörü olarak verilen saldırı türlerini sınıflandırmak için eğitir. İkinci katman kullanıcının belirlediği hedef katmana dönüştürülür. Eğitilen sınıflar sistemin alt sınıflarını ve hedefte oluşması gereken sınıflarını gösterir.

LVQ telekomünikasyondan robotiğe kadar birçok uygulamada başarılı bir şekilde kullanılmaktadır (Hamman vd., 2014). Ayrıca bu sınıflandırma algoritması sezgiseldir ve LVQ2, LVQ3 gibi çeşitli formları vardır. Sınıflar arasındaki uzaklık hesaplanarak maliyet fonksiyonu hesaplanır (Hamman vd., 2014). Öğrenme oranı ve iterasyonların sayısı performansı belirler ve iteratif olarak seçilir.

Saldırı türlerini sınıflandırmak için LVQ kullanılmıştır (Noum vd., 2012). Normal, DoS, U2R, R2L ve Prob olmak üzere 5 saldırı türünü sınıflandırmak için iki katman kullanmışlardır. Bu katmanlar alt sınıf ve ana sınıfı belirler. KDD-Cup 1999 verisi üzerinde yapılan çalışmalarda LVQ yöntemi yaklaşık %81 başarı sağlamıştır (Soleiman vd., 2014). Degang ve arkadaşları (Degang vd., 2007) aynı veri seti üzerinde veriyi normalize edip, özellik seçimi yaptıktan sonra LVQ kullanarak saldırı türlerini sınıflandırarak %76.3 başarı elde etmişlerdir.

4.2.2. Saklı Markov Modelleri

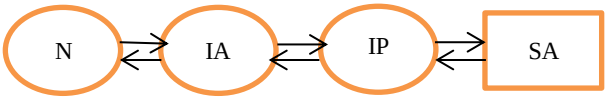
Saklı Markov Modelinde (Hidden Markov Model-HMM) sisteme giriş olarak mevcut durumlar verildiğinde oluşabilecek gelecek durumlar tahmin edilmeye çalışılır. Her çalıştırıldığında farklı bir çıkış ürettiğinden HMM stokastik bir süreçtir. Ayrıca, Markov modellerinde sistem

olasılık dağılımına bağlı olarak kendi durumundan başka bir duruma geçebilir ya da aynı durumda kalabilir. Durumda meydana gelen olasılıklar geçiş olasılıkları olarak adlandırılır. HMM normal Markov modelinden farklı olarak durumlar gözlemci tarafından görülmez. Ancak, duruma bağlı olan geçişler görülebilir.

Saldırı tespit sistemleri dağıtık sistemlerde ağ trafiğindeki kötü paketleri fark eden savunma mekanizmalarıdır. Saldırı önleme sistemleri ağ tabanlı ve host tabanlı saldırı önleme sistemleri olmak üzere 1'ye ayrılır. Saldırı önleme sistemlerinin asıl amacı normal ağ trafiğindeki şüpheli akışı ve şüpheli paketleri gözlemlemektir. Ayrıca, şüpheli ağ trafiğinin yolunu yeniden düzenleyerek DoS saldırısı gibi saldırılarının önlenmesini sağlamaktadır. Aynı zamanda tüm ağ performansını ve yüksek paket işleme oranlarını gözlemler. Yanlış pozitif oranlarını DoS aşamasında azaltmaya çalışır (Haslum vd., 2007).

HMM biyoinformatik, el yazısı tanıma, görüntü işleme ve ses işleme gibi birçok alanda yaygın bir şekilde kullanılmaktadır. Dağıtık veritabanında saldırı tahmin önleme için HMM kullanılmıştır ve risk değerlendirilmesi yapılmıştır (Haslum ve Abraham, 2007). HMM'de iki stokastik işlemi vardır. Biri sistemin durumu (x_t; t=1,2,…) diğeri ise gözlemlenebilen işlemler (y_t; t=1,2,…)'dir. **T** saldırı tespit temsilcileri arasında ardışık olan gözlemleri ifade etmektedir. HMM birimleri aşağıdaki gibidir (Haslum ve Abraham, 2007):

- S= { s_1, s_2,…,s_N } sistemde olası durumları tanımlar. Bu çalışmada Normal (N), Intrusion Attempt (IA), ağda oluşan şüpheli aktiviteler Intrusion in Progress (IP) ve bir ya da daha fazla saldırının sistemde olması Successful Attack (SA) olmak üzere 4 durum vardır.
- Sistemde oluşan gözlemler ise V= {v_1, v_2,...,v_M} ile ifade edilir. Bu çalışmada hiç şüpheli aktivite olmaması (N), ağda şüpheli bir aktivitenin olması (P) ve şüpheli bir aktivitenin başarılı olarak gerçekleştirilme gözlemi (SA) olmak üzere üç gözlem vardır.
- İlk dağıtık vektör π= { π_i } ve π_i= P (x_i=i) olarak sistemde tanımlanır. Bu sistem N-durumlu olarak farz edilir.
- Geçiş olasılık matrisi P= { p_{ij} } ve p_{ij}=P (x_t=j I x_{t-1}=i)'dir. Bu sisteme izinsiz olarak giren kullanıcılar ile sistem arasındaki etkileşimi gösterir.
- Gözlemlenebilen olasılık matrisi ise her saldırı tespit sisteminin temsilcileri için güvenliği ya da kalitesi için tanımlanır.



Şekil 1. HMM'de kullanılan güvenlik durumları

Saklı Markov modelinin ağlardaki güvenlik modellemesi Şekil 1'deki gibidir (Haslum vd., 2007). Durumlar güvenlik durumlarını daireler olarak gösterirken eğer sisteme zarar veren bir durum oluşmuşsa bu durum SA ile gösterilmiştir. Gözlemler mevcut durumlardan bağımsızdır. Her iterasyon sonucundan hesaplanan olasılık dağılımları güncellenir (Rabier, 1990). Deshmukh ve arkadaşları (Deshmukh vd., 2015) tarafından KDD-Cup 1999 verisi üzerinde yaptıkları çalışmada %93.4 başarı elde etmişlerdir. KDD-Cup 1999 verisi üzerinde yapılan çalışmalarda Saklı Markov modeli yöntemi yaklaşık %92 başarı sağlamıştır (Shanmugauadiu vd., 2014).

4.2.3. Naive Bayes Sınıflandırıcılar

Naive Bayes tekniği Bayesian olasılığına dayalı bir tekniktir. Naive Bayes sınıflandırıcısı birbirinden bağımsız olayları işler. Bunun en önemli nedeni bir özelliğin olasılığının diğer özelliklerin olasılığından etkilenmemesidir. Metin dokümanlarını sınıflandırmada, e-postadaki gelen mailleri spam olup olmadığını sınıflandırmada yaygın bir şekilde kullanılmaktadır. Verilerin hangi sınıfa ait olduğunu belirlemek için tüm verilerin olasılıklarını hesaplar. Bir verinin olasılığı diğer verilerin gerçekleşme durumunun meydana gelme olasılığına ve tüm verilerin olma olasılığına bağlıdır. Hangi sınıfın olasılık değeri büyükse veri o sınıfa dahil edilir. Naive Bayes sınıflandırıcısı tekniğinin sonuçları genelde doğrudur. Ancak verinin gürültülü olması, varyansı gibi bazı sebeplerden dolayı hatalı sonuçlar üretebilir. Naive Bayes giriş özelliklerini indirgeyerek önemli özelliklerin bulunmasını sağlar. Böylelikle etkili ve etkin saldırı tespit sistemi oluşturulabilir (Mukherjee vd., 2012).

Naive Bayes sınıflandırma yöntemi ile ağda meydana gelen saldırılar tespit edilmeye çalışılmıştır (Sharma, 2012). KDD'cup'99 veri seti üzerinde ağda meydana gelen yeni saldırı türlerini tespit etmek amacıyla yaptıkları çalışmada oldukça başarılı sonuçlar elde etmişlerdir. Deshmukh ve arkadaşları (Deshmukh vd., 2015) tarafından KDD-Cup 1999 verisi üzerinde yaptıkları çalışmada %88.02 başarı elde etmişlerdir. KDD-Cup 1999 verisi üzerinde yapılan çalışmalarda Naive Bayes yöntemi yaklaşık %91 başarı sağlamıştır (Obeidat vd., 2019).

4.2.4. Bulanık Mantık

Bulanık Mantık (Fuzzy Logic) teknikleri bilgisayar güvenliği alanında 90'lı yıllardan beri kullanılmaktadır. Bu teknikte kesinlik kavramı yoktur. Bir veri birden fazla sınıfa üyelik derecesine göre dahil edilir. Üyelik derecesi 0 ve 1 arasındadır. Örnek verecek olursak havanın veya suyun sıcak, soğuk ve ılık olması kişilere göre değişiklik gösterir. Bu sebeple belli değer aralıkları bu üç sınıfı kapsar. Bulanık Mantık tekniğinin adımları aşağıda verilmiştir:

- Tüm veriler tanımlanır.
- Üyelik fonksiyonları belirlenir.
- Uygulanacak kurallar tanımlanır.
- Kurallar değerlendirilir ve birleştirilir.
- Veriler gösterilen üyelik değerlerine göre sınıflandırılır.

Karışık sistemlerde bilgisayar güvenliğini sağlamak için gelen verileri sürekli analiz eder. Ayrıca, bu teknik kullanıcı imzalarını ya da klasik örüntü tanımayla saldırıları tespit etmeye çalışır. Hatalı ya da normal olmayan olayları tespit eder. İki aşamadan oluşur. İlk aşama kural üretimi; ikinci aşama ise saldırıların tespiti ve önlenmesidir. Bulanık Mantık, düşük, orta ve yüksek saldırı türlerini birbirini kapsayacak biçimde ele aldığı için birçok keskin sınır probleminin üstesinden gelir ve yanlış Pozitif olarak oluşan hataları azaltır. Gerçek zamanlı sistemlerde sistem optimizasyonunu sağlar (Rizvi vd, 2016).

Bulanık Mantık tekniği kullanılarak Tian ve arkadaşları (Tian, 2005) büyük veri setleri alt veri setlerine bölünmüştür ve farklı veri setleri için TCP verisi izlenerek veri setinin performans analizi gerçekleştirilmiştir. KDD-Cup 1999 verisi üzerinde yapılan çalışmalarda Bulanık Mantık yöntemi yaklaşık %94 başarı sağlamıştır (Shanmugauadiu vd., 2014). Nadiammai ve arkadaşları ise (Nadiammai vd., 2012) KDD-Cup 1999 verisi üzerinde k-Ortalama Kümeleme yöntemi ile %81.54 başarı elde etmişlerdir.

4.3. Yapay Zeka Yöntemleri

Bu grup altında Genetik Algoritma, Yapay Sinir Ağları ve Yapay Bağışıklık teknikleri yaygın bir şekilde kullanılan yöntemlerdir.

4.3.1. Genetik Algoritma

Genetik algoritma arama ve optimizasyon problemlerinin çözümünde ayrıca, modelleme yapabilmek için kullanılan biyolojik süreçlerin esas alındığı yöntemlerdir.

Genetik algoritmalar geleneksel yöntemlerle çözümü zor veya imkansız olan problemlerin çözümünde kullanılır. Genetik algoritmalar rastgele arama metodu olduğu için problemin optimum çözümü için tek bir çözüm aramak yerine birden fazla çözüm kümesi üzerinden çalışır. Bu nedenle problem çözümünde sonuçlar her zaman en iyi olmaz. Genetik algoritmanın tercih edilmesinin nedeni, genetik algoritmanın problemin doğasıyla ilgili herhangi bir bilgiye ihtiyaç duymamasıdır. Genetik algoritmanın temel adımları aşağıdaki gibidir (Kannan, 2016):

- Rastgele popülasyon oluşturulur.
- Genetik işlemler (seçme, çaprazlama) uygulanarak bu popülasyondan yeni bireyler oluşturulur. Seçme ve çaprazlama işlemleri popülasyondaki bireylerden gen alışverişi yaparak yeni bireyleri oluşturur.
- Bu bireyler içerisinde problemi çözebilecek en uygun bireyler seçilir.
- Popülasyon boyu tüm iterasyonlar için aynıdır ve gelecek iterasyon için fonksiyonun en yüksek olasılığı seçilir. İterasyon en uygun eşik değerine gelince durur.

Dağıtık sistemlerde ağ yapısı G=(N,E) ağırlıklı graf ile gösterilir. E düğümler arasındaki haberleşme bağlantılarını gösterirken N ise düğümleri gösterir. Çoklu ağ yapılarında T değişkeni maliyet olmak üzere ağ yapısı T= (N_T,E_T) gösterilir. P_T (s,u) ise; s kaynağından u hedef düğüme olan yolu gösterir. T maliyeti eşitlik 1'teki gibi hesaplanır (Kannan, 2016).

$$C(T_e) = \sum e\text{Є}E_T C(e), e\text{Є}E_T \quad (1)$$

s kaynak düğümünden u hedef düğüme olan minimum bant genişliği ise eşitlik 2'teki gibi hesaplanılır.

$$B_T = \min(B(e), e\text{Є}E_T) \quad (2)$$

KDD-Cup 1999 verisi üzerinde yapılan çalışmalarda genetik algoritma yaklaşık %85 başarı sağlamıştır (Hassan, 2013).

4.3.2. Yapay Sinir Ağları

Yapay Sinir Ağları (Artifical Neural Network-ANN) insan beynini modelleyen bilgi sistemleridir ve öğrenme yoluyla verileri sınıflandırır. İnsan beyninin çalışma prensibinden hareketle geliştirilmiştir. Bir başka deyişle; ANN

biyolojik sinir ağlarına benzer bir mantıkla geliştirilmiş ve birbirine ağırlıklar ile bağlantılı olan bilgi işleme yapılarıdır.

Bir ANN girdi, çıktı ve gizli katmanlardan oluşur. Girdi katmanıyla veriler yapay sinir ağına alınır, çıktı katmanıyla da veriler dışarıya aktarılır. Girdi ile çıktı katmanı arasındaki katmanlar ise gizli katmanları oluşturur.

İleri beslemeli Yapay Sinir Ağlarında nöronlar sadece ileriye doğru bağlıdır. Nöron, birbirine bağlanmış tüm verileri ifade eder. Nöron ağının herbir katmanı gelecek katmanın bağlantısını içerir ve bu bağlantılar geriye doğru değildir. Yani nöronlar arasında hiyerarşik bir yapı vardır ve bir katmandaki nöronlar sadece kendinden sonraki katmana veri iletir. Şekil 3'te ileri beslemeli ANN yapısı gösterilmiştir. İleriye geçiş, çıkış katmanına ulaşan aktivasyon akışı ve giriş örneklerinden oluşur. Sigmoid, Gauss fonksiyonu gibi aktivasyon fonksiyonları kullanılabilir. Geriye geçiş aşamasında ise ağdaki asıl çıkış hedef çıkış ile karşılaştırılır ve çıkış birimlerinde oluşan hata hesaplanır (Alhello vd., 2017). Bu yapı saldırı tespit ve önleme sistemleri için eşitlik 3, eşitlik 4 ve eşitlik 5'deki gibi hesaplanır (Tong vd., 2009).

$$x(t) = f\left(W^A x_C(t) + W^B u\,(t-1)\right) \quad (3)$$

$$x_C(t) = x(t-1) \quad (4)$$

$$y(t) = g\left(W^C X(t)\right) \quad (5)$$

Bu ifadelerde x(t) gizli katman çıkışını, y(t) çıkış katmanının çıkışını, u(t-1) ise ağın girişini, W^A birimler ve gizli katman arasındaki bağlantının ağırlığını, W^B giriş ve çıkış katmanı arasındaki bağlantının ağırlığını, W^C gizli ve çıkış katmanı arasındaki bağlantının ağırlığını, f(.) ve g(.) ise gizli katman ve çıkış katmanı arasındaki aktivasyon kodunu gösterir (Tong vd., 2009).

Geri yayılım (backprogation) ağ ise nöronun nasıl eğitildiğini gösterir. ANN eğitici öğrenmenin bir türüdür. Eğitici methot kullanıldığında ağ hem örnek girişleri hem de beklenen çıkışlar ile sağlanır. Girişleri verilen ağlar için beklenen çıkışlar asıl çıkışlarla karşılaştırılır. Beklenen çıkışların kullanılması durumunda hata hesaplanır ve çıkış katmanından giriş katmanına geriye doğru çeşitli katmanların ağırlıkları ayarlanır. Ağ beklenen çıktıyı elde etmek üzere katsayılarını günceller.

ANN'da her iterasyon sonucunda çıkış katmanındaki hata hesaplanarak bu hata çıkış katmanından giriş katmanına doğru bütün nöronlara iletilir ve ağırlıklar hata payına göre tekrar düzenlenir. Bu hata payı nörona ait kendinden önceki nöronlara ağırlıklarıyla orantılı olarak dağıtılır.

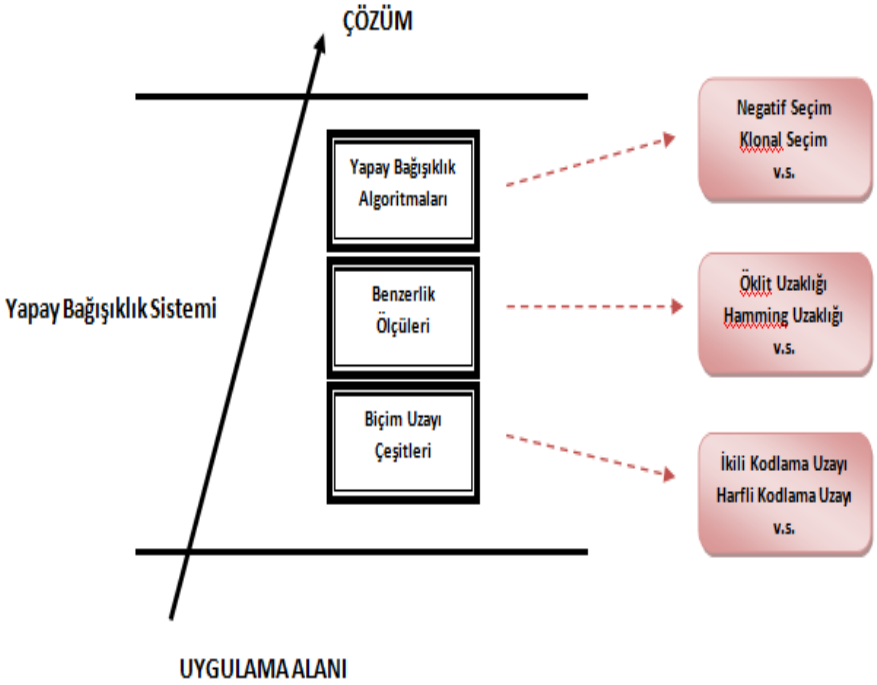
ANN dağıtık sistemlerde güvenlik açıklarının çözümlenebileceği bir yöntemdir. Dağıtık hesaplamalarda ANN düğümler, düğümler arasındaki bağlantılar ve işlem birimlerinden oluşur. İki birim arasındaki bağlantı bir birimin diğer birimi etkileyen ağırlıklarından oluşur. Bu birimler giriş düğümlerinden çıkış düğümlerine doğru toplanarak ve eşik değerden geçerek hareket eder. ANN farklı ağ güvenliklerinde, tıpta, pazarlamada, bankacılık ve finansta, telekomünikasyonda, işlemler yönetiminde ve diğer endüstrilerde uygulanır (Alhello vd., 2017).

Tong ve arkadaşları saldırı tespit sisteminde ANN modelini kullanmışlardır. Elman sinir ağları ile hem hatalı hem de anomali saldırıları tespit etmeye çalışmışlardır. Bu sinir ağları düğümlerin içeriğine sahiptir ve her düğümün içeriği tek gizli katmandan girişini alır ve her düğüm için çıkış gizli katmana bağlıdır (Tong vd., 2009). Mahit ve arkadaşları (Mahit vd., 2015) KDD-Cup 1999 verisi üzerinde ANN ile yapmış oldukları çalışmada yaklaşık %94 başarı sağlamıştır. Ayrıca, Aburomman ve arkadaşları (Aburonman vd., 2016) tarafından KDD-Cup 1999 verisi üzerinde yapılan çalışmada da %98.5 başarı elde edilmiştir.

4.3.3. Yapay Bağışıklık Sistemi

Genetik algoritma arama ve optimizasyon problemlerinin çözümünde ayrıca, modelleme yapabilmek için kullanılan biyolojik süreçlerin esas alındığı yöntemlerdir.

Bağışıklık sistemini, vücudu hastalıklara karşı koruyarak ona savunma kazandıran bir koruyucu mekanizma olarak tanımlayabiliriz. Bağışıklık sisteminin biyolojik tanımı ve çalışma mantığından esinlenerek ortaya çıkan Yapay Bağışıklık Sistemi (Artificial Immune System-AIM), özellikle son yıllarda yapay zeka temelli yöntem olarak araştırma çalışmalarında sıklıkla yer aldığı gözlemlenmiştir. Bahse konu bu araştırma alanlarından biri de AIM'nin saldırı tespitinde kullanılmasıdır. Yapay bağışıklık adını verdiğimiz bu sistemin ilk esinlenme kaynağının bilgisayar virüsleri olduğunu söyleyebiliriz.



Şekil 2. AIM'nin katmanlı yapısı (Shen vd., 2011)

Yapay Bağışıklık Sistemi (AIM), katmanlı bir yapıda olup, Şekil 2'de gösterilmiştir (Shen vd., 2011). Sistemin temel parçası olarak kullanılan antijen ve antikorlar arasındaki benzerlik, tanıma, şekil uzayında gösterim gibi Afinite Ölçümü (benzerlik ölçümü) ile sağlanır. Afinite ölçümü için Öklid, Hamming veya Manhattan mesafeleri kullanılır. Afinite ölçümü düşük olan bireyler daha fazla mutasyona uğratılarak bir antijeni tanıyacak en iyi antikor bulunur ve antikor belirli oranlarda mutasyon ile çoğaltılır (klonal seçim algoritması), çoğaltılan hücreler antikor hücrelerine eklenir. Antijen, antikor hücrelerine sunulur. Antijeni en iyi tanımlayan ve belirli bir eşik değerinin üzerinde olan antikorlar alınır ve aralarında yarıştırılır (Afinite ölçümü ile benzerlik oranları hesaplanır). Yarışma sonucunda Afinitesi düşük olan belirli sayıda bireyler alınarak bellek hücresine eklenir. Böylece, bir antijene karşı en iyi antikor hücrelerini temsil edecek bellek hücresini üretilmiş olur. Bu süreç sisteme sunulacak diğer antijenler için de ayrı ayrı uygulanır (Farhaoui, 2017). Shen ve arkadaşları'nın (Shen vd., 2011) KDD-Cup 1999 verisi üzerinde yapılan çalışmalarında yapay bağışıklık yöntemi yaklaşık %99 başarı sağlamıştır. Chen ve arkadaşları (Chen vd., 2016) ise KDD-Cup 1999 verisi üzerinde yaptıkları çalışmada %92.41 başarı sağlamışlardır.

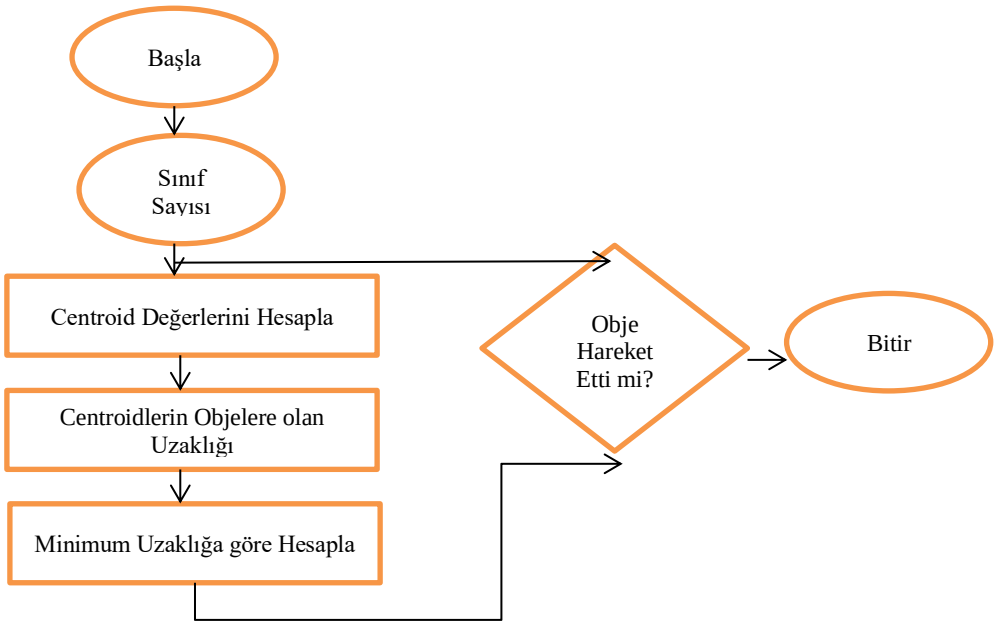
5. Materyal ve Method

Dağıtık veritabanlarında saldırı tespit ve önleme sistemlerinde kullanılan yöntemler Tablo 1'de gösterildiği gibi üç ana grupta toplanmıştır: Veri madenciliği yöntemleri, İstatiksel yöntemler ve Yapay Zeka yöntemleri. Her bir yöntemdeki tekniklere izleyen alt bölümlerde yer verilmiştir. Ayrıca, tekniklerin performans değerleri grup içi ve gruplar arası olmak üzere değerlendirilmiştir.

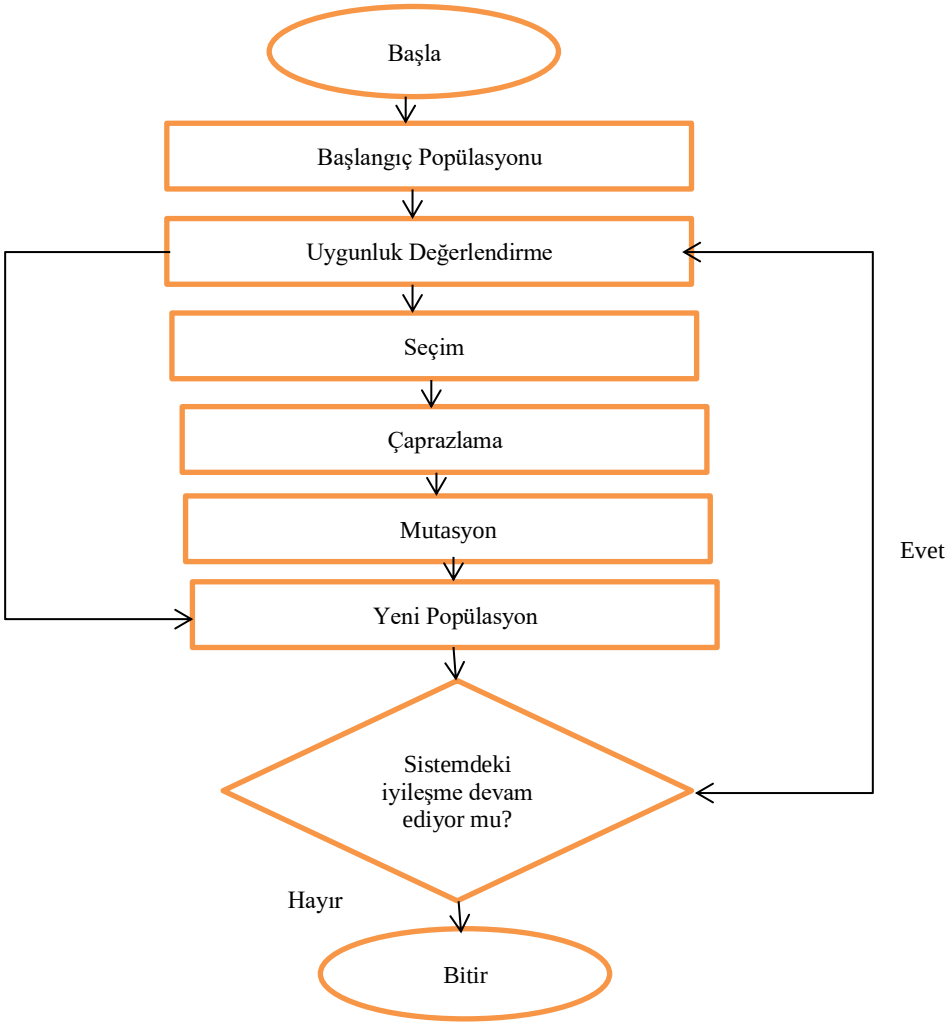
Şekil 3'de K Ortalama Kümeleme tekniğinin akış şeması verilmiştir. Bu teknikte tüm verilerin merkeze olan uzaklığı hesaplanır ve uzaklığı en küçük olan veriler aynı sınıfta toplanır.

Şekil 4'te genetik algoritmanın adımları verilmiştir. Genetik algoritmada iyileştirme süreci en iyi popülasyon bulununcaya kadar devam eder.

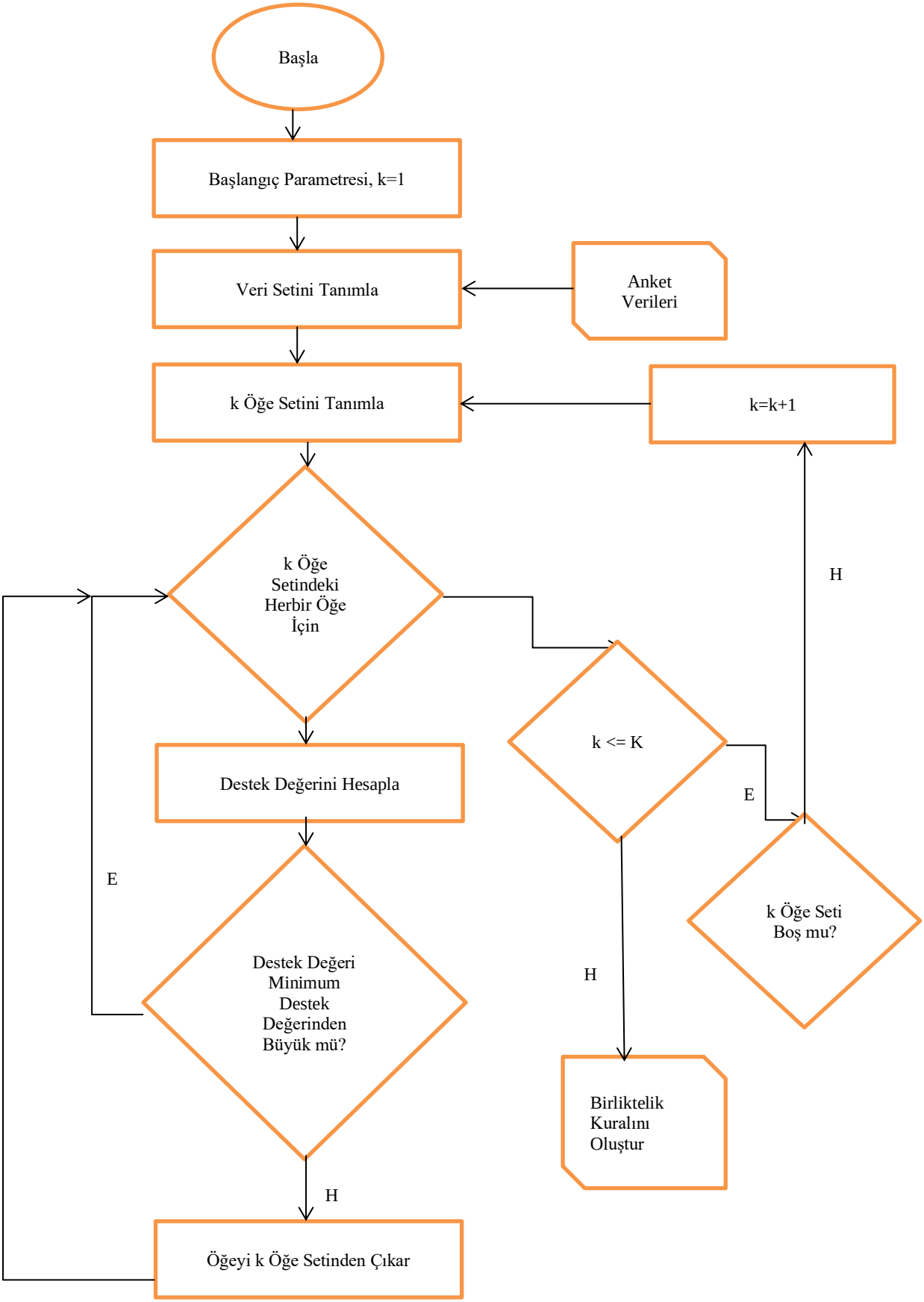
Şekil 5'de birliktelik kuralının akış şeması verilmiştir. Her nesnenin destek değeri bulunur ve en küçük destek değeri ile her nesnenin destek değeri karşılaştırılır ve küçük olanlar kümeye alınarak birliktelik kuralları oluşturulur.



Şekil 3. K ortalama kümeleme tekniğinin akış diagramı



Şekil 4. Genetik algoritma akış diagramı



Şekil 5. Birliktelik kuralları akış diagramı

6. Bulgular

Bu çalışmada, dağıtık veritabanı saldırı tespit ve önleme yöntemleri doğruluk, hız, performans, kullanılan veri setinin büyüklüğü gibi bazı metriklere göre karşılaştırılmıştır. Buna göre kullanılan yöntemlerin avantaj ve dezavantajları Tablo 2'de gösterilmiştir. Veriler arasındaki benzerlikleri bulmak, ilişkiyi tespit etmek, verileri tanıma, geçmiş verilerden yararlanacak tüm veriler için ortak model oluşturma, benzer özeelliklere göre gruplama yapmak veri madenciliği yöntemlerinin ortak özelliğidir. Saldırı türlerini kolay ve doğru bir şekilde tespit eder. Bu tekniklerin ortak özelliği faydalı ve mantıklı veriyi elde etmek için birden fazla basamaktan oluşan bir veri analizi yapar. Saklı ilişkileri ortaya koyduğundan küçük verilerde oldukça başarılı sonuçlar verir. Ayrıca birden fazla veri madenciliği tekniğini kullanarak daha başarılı sonuçlar eşde edilebilir. Ancak bu yöntemlerde geçmiş verilerden yararlanılarak analiz yapıldığı için yeni saldırı türlerinin bulunması oldukça zordur. Ayrıca çok büyük veri setleri için uzun süren hesaplamalar gerektirir. Bu da ölçekleme problemi, zaman ve hesaplama karmaşıklığı gerektirir. İstatiksel yöntemler ise veri madenciliği yöntemlerine aksine geçmiş verilere ihtiyaç duymaz. Bu sebeple yeni saldırı türlerini bulmada veri madenciliği yöntemlerine göre daha etkilidir. Ancak veri madenciliği yöntemlerinde ek parametrelere ihtiyaç duyulmaz. Bu yöntemlerde ise ek parametreler gerekir ve tüm veriler için ortak ve doğru parametreleri belirlemek zordur. Yapay zeka yöntemleri çok büyük veri setleri için kullanılır. Diğer iki gruptaki yöntemlerin tersine oldukça pratiktir ve esnektir. Sezgiye dayalı teknikleri içerir. Ayrıca diğer yöntemlerden farklı olarak saldırı türlerini tespit ederken özel bilgilere ihtiyaç duymaz. Karmaşık yapıya sahip veri türlerini bulmada başarılı sonuçlar verir. Ancak bu gruptaki yöntemlerde tek bir çözüm yolu kullanılmadığı için yüksek miktarda kaynak tüketir.

Tablo 2. Dağıtık sistemlerde kullanılan yöntemlerin avantaj ve dezavantajları

Veri Madenciliği Yöntemleri	**Avantaj:** Küçük veriler için oldukça kullanışlı ve başarısı yüksektir. Sağlamdır ve sonuçları kesine yakındır.
	Dezavantaj: Sisteme uygulanması bir takım hesaplama ve zaman karmaşıklığını beraberinde getirmektedir.
İstatiksel Yöntemler	**Avantaj:** Normal hareketleri gözlemlemek için genelde önceki bilgilere ihtiyaç duymaz. Bu sebeple yeni saldırı türlerini bulmada etkindir.
	Dezavantaj: Sistemin başarısını hesaplayabilmek için gerekli olan parametreleri ve metrikleri belirlemek oldukça zordur ve her veri seti için farklılık gösterir.
Yapay Zeka Yöntemleri	**Avantaj:** Esnektir ve hemen hemen tüm veri setlerine uygulanabilir.
	Dezavantaj: Saldırı tespit ve önleme sistemleri için yüksek miktarda kaynak tüketir.

Yapay zeka teknikleri ve veri madenciliği yöntemleri istatiksel yöntemlere göre daha başarılı sonuçlar vermektedir. Ayrıca, yapay zeka yöntemleri büyük veriler için daha kolay uygulanmaktadır. Bu teknikler birleştirilerek daha güvenli ve daha hızlı veri gönderimini gerçekleştirmek mümkün olabilir.

7. Değerlendirme

Dağıtık veritabanlarında saldırı tespit ve önleme sistemlerinde kullanılan yöntemler Tablo 1'de gösterildiği gibi üç ana grupta toplanmıştır: Veri madenciliği yöntemleri, İstatiksel yöntemler ve Yapay Zeka yöntemleri. Her bir yöntemdeki tekniklere izleyen alt bölümlerde yer verilmiştir. Ayrıca, tekniklerin performans değerleri grup içi ve gruplar arası olmak üzere değerlendirilmiştir.

Teknolojinin gelişmesiyle birlikte veriye erişim ve veri iletişimi oldukça kolaylaşmıştır. Dağıtık sistemler veriye her yerden hızlı ve kolay bir şekilde erişim sağlanmaktadır. Ancak, birden fazla kullanıcının aynı anda farklı yerlerden sisteme erişmek istemesi veri güvenliği, veri gizliliği, servis sürekliliği, yetkilendirme ve veriyi güvenli saklama gibi birtakım problemleri de beraberinde getirmektedir. Kullanıcıların veriye erişim sorunları, ağın dinlenmesi, hizmetin engellenmesi, güvensiz ağlar, yetkisiz kişilerin önemli bilgileri ele geçirerek saklaması bu problemlerden birkaçını oluşturmaktadır. Bu problemlerin üstesinden gelebilmek için Veri Madenciliği, İstatiksel ve Yapay Zeka alanında kullanılan saldırı tespit ve önleme tekniklerine dayalı olarak gerçekleştirilmiş çeşitli sistemler kullanılmaktadır.

Dağıtık veritabanlarında saldırı tespit ve önleme sistemleri için kullanılan tekniklerin KDD-Cup-1999 verisi üzerindeki performans sonuçları çeşitli çalışmalardan derlenerek Tablo 3'de verilmiştir. Yapılan çalışmaların sonuçları normalizasyon, gürültüden temizleme, özellik çıkarımı ve özellik seçimi tekniklerine bağlı olarak değişmektedir. Çünkü yapılan çalışmalarda gürültü giderimi, özellik çıkarımı ve seçme yöntemleri kullanılması sınıflandırma başarısını arttırdığı gözlemlenmiştir. Bu yöntemler ile eldeki veri daha kullanışlı hale getirilmektedir.

Dağıtık veritabanlarında saldırı tespit ve önleme sistemleri için kullanılan tekniklerin gerçek veri seti üzerindeki performans sonuçları ise Tablo 4'de verilmiştir. Bu deneysel çalışmada kullanılan veri seti bir bankanın 2 yıllık verileri gözönüne alınarak hesaplanmıştır.

Tablo 3. Dağıtık sistemlerde kullanılan yöntemlerin başarıları

Yöntemler	Teknikler	Başarı Oranı(%)
Veri Madenciliği Yöntemleri	Destek Vektör Makineleri	93.9 (Aburomman vd., 2016)
	k En Yakın Komşuluk	96.24 (Senthilnayaki, 2019)
	Karar Ağacı	97.7 (Ugochhukwu vd., 2018)
	Birliktelik Kuralları	96.9 (Abraham vd., 2007)
	k-Ortalama Kümeleme	91.84 (Sharma vd., 2018)
İstatiksel Yöntemler	Bulanık Mantık	94.92 (Abraham vd., 2007)
	Saklı Markov Modelleri	93.4 (Shanmuyavadiu vd., 2014)
	Naive Bayes	91.23 (Obeidat vd., 2019)
	Öğrenmeli Vektör Kuantalama	81 (Soleiman vd., 2014)
Yapay Zeka Yöntemleri	Yapay Bağışıklık	99.74 (Shen vd., 2011)
	Yapay Sinir Ağları	94 (Aburonman vd., 2016)
	Genetik Algoritma	85.7 (Hassan, 2013)

Tablo 4. Banka verisi üzerinde kullanılan yöntemlerin başarıları

Yöntemler	Teknikler	Başarı Oranı(%)
Veri Madenciliği Yöntemleri	Destek Vektör Makineleri	87.64
	k En Yakın Komşuluk	88.16
	Karar Ağacı	82.75
	Birliktelik Kuralları	81.01
	k-Ortalama Kümeleme	83.61
İstatiksel Yöntemler	Bulanık Mantık	78.20
	Saklı Markov Modelleri	81.37
	Naive Bayes	79.59
	Öğrenmeli Vektör Kuantalama	75.88
Yapay Zeka Yöntemleri	Yapay Bağışıklık	93.26
	Yapay Sinir Ağları	90
	Genetik Algoritma	91.83

8. Tartışma ve Sonuç

Günümüzde eğitim, ekonomi, askeri ve iş hayatının birçok yerinde bilgisayarlar yaygın bir şekilde kullanılmaktadır. Ağ üzerinden gizli, özel ve korunması gereken birçok değerli bilgiler kullanıcılar arasında paylaşılmaktadır. Teknolojinin hızla artması, bilginin paylaşılması ağ ve bilgisayar güvenliğinde saldırıların oluşmasına neden olmaktadır. Şifreleme, yetkilendirme, yetkisiz kişilerin sisteme zarar vermesi, sistemi çökertme gibi birçok güvenlik sorununun ortaya çıkması gerekli güvenlik tedbirlerinin de alınmasını gerekli kılmaktadır. Bu durumda sistemi kötüye kullanan kullanıcıları tespit etmek ve normal olmayan anormal davranışları önlemek için saldırı tespit ve önleme sistemleri ortaya çıkmıştır.

Veri madenciliği yöntemlerinde kullanılan tekniklerle veriler arasındaki benzerlikler, ilişkiler tam olarak belirlenir. Ancak özellikle Destek Vektör Makineleri, k-Ortalama gibi teknikler çok uzun süren hesaplamalar gerektiği için büyük

verilerde zaman ve maliyet gerektirir. Bu sorunu çözebilmek için kullanılan veride özellik çıkarımı, özellik seçimi ve boyut indirgeme yapılabilir. Böylelikle büyük verilerde algoritma karmaşıklığı azaltılır. Bu grupta en başarılı sonucu k En Yakın Komşuluk tekniği vermiştir. Çünkü bu teknikte her hesaplamada birbirine benzer saldırı türleri daha belirgin bir şekilde ortaya çıkar. Ayrıca Destek Vektör Makinelerinde saldırı türlerini birbirinden ayıran sınıfları temsil edecek en uygun fonksiyon elde edildiği için bu teknik bu gruptaki diğer tekniklere oranla daha başarılıdır.

İstatiksel yöntemlerde ise Saklı Markov Modeli bu gruptaki en başarılı tekniktir. Çünkü Saklı Markov Modelinde mevcut durumlara göre çıkışlar stokastik olarak üretilir ve en doğru çıkışlar böylelikle elde edilir. Veri madenciliği yöntemlerine göre bu gruptaki tekniklerde yeni saldırı türlerini bulmak için geçmiş veriler gibi ek parametrelere bakılmaz. Bu sebeple yeni saldırı türlerini bulmada daha başarılı sonuçlar verir.

Yapay zeka yöntemleri ise diğer iki gruptaki tekniklere oranla çok büyük hesaplamalar gerektirmez. Bundan dolayı daha kısa zamanda sonuç üretir. Sezgisel bir yöntem olduğundan dolayı istatistiksel yöntemler gibi ek bilgilere ihtiyaç duymaz. Ek bilgiler olmadan da doğru sonuçlar üretebilir. Yapay Bağışıklık tekniği saldırı türlerini bulmada bu gruptaki en başarılı tekniktir. Çünkü Genetik Algoritma ve Yapay Sinir Ağları, tek bir en iyi adayı bulmaya çalışırken, Yapay Bağışıklık Sistemi süreç boyunca oluşturulan tüm popülasyonlarda da en iyi bireyleri bulmaya çalışır. Ayrıca üç gruptaki tekniklerin hepsinde de gürültü giderimi, veri ön işleme yapılarak başarı arttırılır.

Sonuç olarak, yapay zeka tekniklerinden özellikle Yapay Bağışıklık tekniği dağıtık veritabanları için saldırı tespit ve önleme sistemlerinde oldukça başarılı sonuçlar vermiştir. Yapay zeka tekniklerinin saldırı tespit ve önlemede etkin bir şekilde kullanılması ileride yapılacak çalışmalar için büyük önem taşır. Özellikle geliştirilmiş yapay zeka tekniklerini de içeren hibrit yöntemler kullanılarak daha başarılı sonuçlar elde edilebileceği göz ardı edilmemelidir.

Kaynaklar

Abraham, A., Grosan, C. ve Martiv-Vide, C., 2007. Evolutionary Design of Intrusion Detection Programs. International Journal of Network Security, 4, 328-339.

Aburonman, A. ve Reaz, M., 2016. A Novel SVM-kNN-PSO Ensemble Method for Intrusion Detection System. Elseiver Applied Soft Computing, 38, 360-372.

Alhello, Z., Abdul, A. ve Harleen, K., 2017. On Applicatiablity of Neural Network in Intrusion Detection and Prevention. International Journal of Advanced Research in Computer Science, 8(7), 494-498.

Bakir, C. ve Hakkoymaz, V., 2015. Veritabanı Güvenliğinde Saldırı Tahmini ve Tespiti için Kullanıcıların Sınıflandırılması, ISCTurkey 2015 8.Uluslararası Bilgi Güvenliği ve Kriptoloji Konferansı (VIII. Int'l Conference on Information Security and Cryptology), Ankara, Türkiye, s.1-6.

Castro, L. ve Timmis J., 2003. Artificial İmmune Systems as a Novel Soft Computing Paradigm. Soft Computing, Springer, 7(8), 526-544.

Chen, M., Chang, P. ve Wu, J., 2016. A Population-Based İncremental Learning Approach with Artificial İmmune System for Network Intrusion Detection. Elseiver Engineering Applications of Artificial Intelligence, 51, 171-181.

Degang, Y. ve Guo, C., 2007. Learning Vector Quantization Neural Network Method for Network Intrusion Detection. Wuhan University Journal of Natural Sciences, 12(1), 147-150.

Deng, H.ve Zeng, Q., 2003. SVM-baseed Detection System for Wireless Ad Hoca Networks, Vehicular Technology Conference, Orlando, USA, 2147-2151.

Faraoun, K.M. ve Boukelif, A., 2007. Neural Networks Learning Improvement Using the K-Means Clustering Algorithm to Detect Network Intrusions. International Journal of Computer and Information Engineering, 1(10), 3138-3145.

Farhaoui, Y., 2017. Design and Implementation of an Intrusion Prevention System. International Journal of Network Security, 19(5), 675-683.

Hamman, B. ve Hoffman, D., 2014. Learning Vector Quantization for (dis-)-similarities. Elsevier Neurocomputing, 131, 43-51.

Haslum, K. ve Abraham, A., 2007. Disp: A Framework for Distributed Intrusion Prediction and Prevention Using Hidden Markov Models and Online Fuzzy Risk Assesment. 3rd International Symposium on Information Assurance and Security, Manchester, United Kingtom, 183-190 pp.

Hassan, M., 2013. Network Intrusion Detection System Using Genetic Algorithm and Fuzzy Logic. International Journal of Innovative Research in

Computer and Communication Engineering, 1(7), 435-1445.

Hu, W. ve Jun, G., 2014. Online Adaboost-Based Parameterized Methods for Dynamic Distributed network Intrusion Detection. IEEE Transactions on CyberNetics, 44(3), 66-82.

Hu, Y. ve Panda, B., 2004. A Data Mining Approach for Database Intrusion Detection. ACM Symposium on Applied Computing, 711-716 pp.

Jemili, F., 2009. Hybrid Intrusion Detection and Prediction multiAgent System, HIDPAS, (IJCSIS) International Journal of Computer Science and Information Security, 5(1), 62-71.

Kannan S., Ruban M. ve Arun, M., 2016. Intelligent Intrusion Detection System using Genetic Algorithm. Journal of Advances in Chemistry, 12(17), 5020-5025.

Law, K. ve Kwok, F., 2004. IDS False Alarm Filtering using KNN Classifier. Springer Information Security Applications Lecture Notes in Computer Science, 114-121 pp.

Mahit, D., 2015. Using Artifical Neural Network Classification and Invertion of Intrusion in Classification and Intrusion Detection System. International Journal of Innovative in Computer and Communication Engineering, 3(2), 1102-1108.

Malhotra, S., Bali, V. ve Paliwal, K., 2017. Genetic Programming and K-nearest Neighbour Classifier Based Intrusion Detection Model. 7th International Conference on Cloud Computing, 42-46 pp.

Moon, D., Im, H. ve Kim, I., 2017. Dtb:Ids: An Intrusion Detection System based on Decision Tree using Behavior Analysis for Preventing Apt Attacks. The Journal of Supercomputing, 73(7), 2881-2895.

Mukherjee, D.S. ve Sharma, N., 2012. Intrusion Detection using Naive Bayes Classifier with Feature Reduction. Elsevier Procedia Technology, 4, 119-128.

Mukkamala, S. ve Janoski, G., 2002. Intrusion Detection using Neural Networks and Support Vector Machines, IJCNN'02 Proceedings of the 2002 International Joint Conference on, 1702-1707.

Nadiammai, G.U. ve Hemalathen, M., 2012. An evaluation of clustering technique over intrusion detection system, ICACCI'12 Proceedings of the International Conference on Advances in Computing, Communications and Informatics, 1054-1060 pp.

Noum, R. ve Al-Sultani, Z., 2012. Learning Vector Quantization (LVQ) and k-Nearest Neighbor for Intrusion Classification. World of Computer Science and Information Technology Journal (WCSIT), 2(3), 105-109.

Obeidat M., Hamadreh N. ve Alakasassbeh M., 2019. Intensive Pre-Processing of KDD Cup 99 for Network Intrusion Classification Using Machine Learning Techniques. International Journal of Interactive Mobile Tehnologies, 16(1), 71-84.

Rabier, L.R., 1990. A tutorial on Hidden Markov Models and Selected applications speech recognition. Ready in Speech Recognition, 267-296.

Rachburee, N. ve Punlumjeak, W., 2017. Big Data Analytics: Feature Selection and Machine Learning for Intrusion Detection on Microsoft Azure Platform. Journal of Telecommunication Electronic and Computer Engineering, 9(1-4), 1-5.

Ramasubramanian, P. ve Kannan, A., 2014. Multi-Agent based Quickprop Neural Network Short-term Forecasting Framework for Database Intrusion Prediction System. CiteSeerX.

Rizvi, S., Labrador, G. ve Guyan, M., 2016. Advocating for Hybrid Intrusion Detection Prevention System and Framework Improvement. Elsevier Procesia Computer Science, 95, 369-374.

Romasubramanian, P. ve Kannan, A., 2006. A Genetic-Algorithm Based Neural Network Short-Term Farecasting Framework for Database İntrusion Prediction System. Soft Computing, 10(8), 699-714.

Sağıroğlu, Ş., Yolaçan, E.N ve Yavanoğlu, U., 2012. Zeki Saldırı Tespit Sistemi Tasarımı ve Gerçekleştirilmesi. Gazi Mühendislik-Mimarlık Fakültesi Dergisi, 26(2), 325-340.

Sharma P., Sengupta J. ve Suri P.K., 2018. Wli-Fcm and Artificial Neural Network Based Cloud Intrusion Detection System. International Journall Advanced Networking and Applications, 10(1), 3698-3703.

Senthilnayaki, B., Venkatalakshmi, K., Kannan, A., 2019. Intrusion Detection System using Fuzzy Rough Set Feature Selection and Modified KNN Classifier, The International Arab Journal of Information Technology, 16(4), 746-753.

Shams, E.A., Rizaer, A. ve Ulusoy, A.H., 2018. Trust aware Supporrt Vector Machine Intrusion Detection and Preventin System in Vehicular ad hoc Networks. Elsevier Computers&Security, 78, 245-254.

Shanmugavadivu, R. ve Nagarajan, N., 2014. Network Intrusion Detection System using Fuzzy Logic. Indian Journal of Computer Science and Engineering (IJCSE), 2(1), 101-111.

Sharma, S., 2012. An Improved Network Intrusion Detection Technique based on k-means clustering via Naive Bayes Classification, IEEE-International Conference on Advances In Engineering, Science and Management (ICAESM-2012), Nagapattinum, India, 417-422 pp.

Shen, J. ve Wang, J., 2011. Network Intrusion Detection by Artificial Immune System, IEEE Power and Energy General Meeting, 1-8.

Soleiman, E. ve Fetarat, A., 2014. Using Learning Vector Quantization (LVQ) in Intrsuion Detection Systems. International Journal of Innovative Research in Advanced Engineering (IJIRAE), 1(10).

Tajbakhsh, A. ve Rahmati, M., 2009. Intrusion Detection Using Fuzzy Assocation Rules. Elsevier Applied Soft Computing, 9, 462-469.

Tian, J., 2005. Intrusion Detection Combining Multiple Decision Trees by Fuzzy Logic, Proceedings of the sixth International Conference on Parallel and Distributed Computing. Applications and Technologies (PDCAT'05).

Tong, X. ve Wang, Z., 2009. A Research Using Hybrid RBF/Elman Neural Networks for Intrusion Detection System Secure Model. Elsevier Computer Physics Communications, 180, 1795-1801.

Ugochukwu, C. ve Bennett E.O., 2018. Intrusion Detection System using Machine Learning Algorithm. International Journal of Computer Science and Mathematical Theory, 4(1), 39-47.

Yıldırım, M.Z., Çavuşoğlu, A., Şen, B. ve Budak, İ., 2014. Yapay Sinir Ağları ile Ağ Üzerinde Saldırı Tespiti ve Paralel Optimizasyonu, XVI, Akademik Bilişim, Şubat 2014, Mersin, Türkiye, s.671-677.