Dokuz Eylül Üniversitesi Mühendislik Fakültesi Fen ve Mühendislik Dergisi
Dokuz Eylul University Faculty of Engineering Journal of Science and Engineering

Basılı/Printed ISSN: 1302-9304. Elektronik/Online ISSN: 2547-958X

Telsiz Duyarga Ağlarda Bizans Saldırılarının Topluluk Öğrenme-tabanlı Tespiti

Ensemble Learning-based Method for Detection of Byzantine Attacks in Wireless Sensor Networks

Vahid Khalilpour Akram [1], Pelin Yıldırım Taser [2*]

[1] Ege Üniversitesi Uluslararası Bilgisayar Enstitüsü, İzmir, TÜRKİYE
[2] İzmir Bakırçay Üniversitesi Mühendislik ve Mimarlık Fakültesi Bilgisayar Mühendisliği Bölümü, İzmir, TÜRKİYE
*Sorumlu Yazar / Corresponding Author **: pelin.taser@bakircay.edu.tr

Geliş Tarihi / Received: 09.03.2020 *Araştırma Makalesi/Research Article*
Kabul Tarihi / Accepted: 05.05.2020 *DOI:10.21205/deufmd.2020226624*
Atıf şekli/ How to cite: AKRAM, V.K., TASER YILDIRIM, P., (2020). Telsiz Duyarga Ağlarda Bizans Saldırılarının Topluluk Öğrenme-tabanlı Tespiti. DEUFMD 22(66), 905-918.

Öz

Telsiz duyarga ağlar (TDA)'da düğümler arasında güvenilir iletişimin sağlanması ve doğru verilerin toplanması birçok açıdan hayati önem taşımaktadır. TDA'ların merkezi iletişim altyapısı olmadığından dolayı, bu ağlar çeşitli saldırılara maruz kalabilmektedirler. TDA'larda yaygın saldırı türlerinden birisi olan Bizans saldırısında, saldırgan ağ alanına yeni bir düğüm ekleyip sahte veriler üreterek ağın güvenilirliğini düşürebilmektedir. Bu çalışma, TDA'da Bizans saldırılarının tespitine yönelik iki yeni topluluk tabanlı yaklaşım önermektedir. Önerilen bu yaklaşımlar, 3 farklı geleneksel sınıflandırma algoritmasının (Naive Bayes, karar ağacı (C4.5) ve k-en yakın komşuluk (İng. k-NN)) voting ve stacking yönetimleri ile bir araya getirilmesinden meydana gelmektedir. Ayrıca, deneysel çalışmalar kapsamında, önerilen iki yeni yaklaşımın yanı sıra, mevcut topluluk öğrenmesi yaklaşımları (C4.5 tabanlı Bagging (Bagging(C4.5)) ve Boosting (AdaBoost)) ile geleneksel algoritmalar (Naive Bayes, C4.5 ve k-NN) da, 66 IRIS düğümünden (60 normal, 6 saldırgan) oluşan örnek ağ üzerinde uygulanmıştır. Her bir algoritmadan elde edilen sınıflandırma sonuçları, doğruluk oranı ve f-ölçüm değerlerine göre karşılaştırılmıştır. Test yatağından elde edilen sonuçlar göstermektedir ki, topluluk tabanlı yöntemler, TDA'da Bizans saldırılarının tespitinde %98.48 doğruluk oranına ulaşırken, geleneksel (tek bir sınıflandırma modeli kullanan) yöntemler %96.97 ile sınırlı kalmaktadır. Çok sayıda düğüm içeren daha büyük ağlarda, bu oranların arasındaki fark artabilir.

Anahtar Kelimeler: *Bizans Saldırıları, Makine Öğrenmesi, Sınıflandırma, Telsiz Duyarga Ağlar, Topluluk Öğrenmesi*

Abstract

Reliable communication and accurate data collection are crucial tasks in Wireless Sensor Networks (WSNs). Due to the lack of having no central communication infrastructure, WSNs can be exposed to various attacks. One of the common attack types in WSNs is Byzantine attack, in which the attacker can reduce the reliability of the network by adding new nodes to the network area and sending fake data. This study proposes two ensemble-based approaches for detecting the Byzantine attacks in WSNs. The proposed approaches combine three different traditional classification algorithms (Naive

Bayes, decision tree (C4.5), and k-NN) with voting and stacking methods. In addition to the proposed methods, the current ensemble learning approaches (C4.5 based Bagging (Bagging(C4.5)) and Boosting (AdaBoost)) and the traditional algorithms (Naive Bayes, C4.5 and k-NN) were applied on a sample network of 66 IRIS nodes (60 normal, 6 malicious) within experimental studies. The classification results obtained from each algorithm were compared according to the accuracy rate and f-measure values. The results gathered from the testbed show that the ensemble-based methods achieve up to 98.48% accuracy rate for detection of the Byzantine attacks in the sample network while this ratio for the traditional methods is limited to the 96.97%. In large networks with more nodes, the difference among these ratios may increase.

Keywords: *Byzantine Attacks, Classification, Ensemble Learning, Machine Learning, Wireless Sensor Networks*

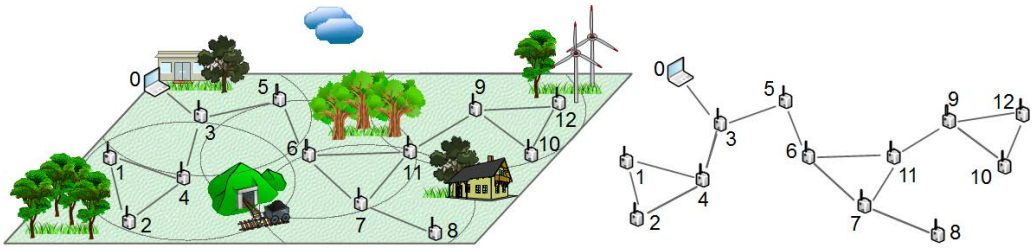
1. Giriş

Telsiz Duyarga Ağ (TDA) 'lar, bağımsız, radyo mesajları üzerinden haberleşebilen ve çeşitli olaylar veya çevre koşullarını algılayabilen düğümlerden oluşmaktadır. Bu düğümler, ortamdan algıladıkları verileri çok sekmeli bağlantılar üzerinden bir işleme merkezine gönderirler. TDA'larda, her düğümün kendine özel işlemci, bellek ve algılayıcısı bulunduğundan, düğümler birbirinden bağımsız ve asenkron bir şekilde çalışırlar ve birbirleri ile iletişime geçebilmek için radyo mesajı gönderirler. Günümüzde TDA'lar, sağlık, güvenlik, otomasyon, askeri operasyonlar, hedef takibi, akıllı yapılar ve arama kurtarma uygulamaları gibi alanlarda sıklıkla kullanılmaktadır [1]. Örneğin; ormanlarda meydana gelebilecek olası yangınların erken tespiti, savaş ortamında askeri birliklerin konumlandırılması, fabrikalarda kimyasal maddelerin tespiti, hastanelerde tıbbi malzeme veya sağlık çalışanlarının konumlandırılması ve büyük binaların güvenliğinin sağlanması amacıyla telsiz duyarga ağlarından faydalanılmaktadır [2-4].

Genel olarak bir TDA'da, ortamdan toplanan veriler, çok sekmeli radyo mesajları aracılığıyla öncelikle bir baz istasyonuna ve ardından bir merkezi işleme birimine aktarılırlar. Baz istasyonu, merkezi işleme birimi ile ağda bulunan diğer düğümler arasında bir köprü görevi görerek, ağdan gelen verileri işleme birimine ve işleme biriminden çıkan komutları da ağda bulunan düğümlere aktarır.

TDA'da her bir düğüm radyo menzilinde bulunan diğer düğümlere bağlanabilir. Böylece, çoklu zıplama bağlantılar aracılığıyla uzak düğümler birbirleri ile iletişim kurabilmektedirler. Dolayısıyla, ana işlevselliğinin yanı sıra, düğümlerin çoğu verilerin ağda yönlendirilmesi için iş birliği yaparlar. Düğümler, dağıtık yönlendirme algoritmaları sayesinde, kendisine gelen mesajları komşularına ileterek, mesajların hedef düğüme ulaşmasını sağlarlar. Şekil 1.a'da örnek bir TDA ve Şekil 1.b'de ise bu ağın çizge modeli gösterilmektedir. Bu örneğe göre, ağda bulunan 0 numaralı düğümün baz istasyonu olduğu varsayılmaktadır. Ağdaki her bir düğüm algıladığı çeşitli verileri baz istasyonuna daha yakın mesafedeki komşularına gönderir. Aradaki düğümler ise, gelen mesajları baz istasyonuna doğru ileterek, tüm verilerin baz istasyonunda toplanmasını sağlarlar. Örneğin; Şekil 1.a'da bulunan 8 numaralı düğümün verileri, 8,7,6,5,3,0 patikasından baz istasyonuna iletilebilir.

Bir TDA'ya yeni bir düğüm eklemek için, o düğümü ağa bağlı olan mevcut düğümlerden birisinin radyo menziline yerleştirmek yeterlidir. Böylece herhangi bir iletişim altyapısına gerek kalmadan, özellikle, dağ veya orman gibi zorlu ortam şartlarında, TDA'ların kapsama alanını daha hızlı ve kolay bir şekilde genişletebiliriz. TDA'lar, diğer ağlara göre daha kolay kurulup, genişletilebilme avantajına sahip olmalarının yanı sıra, bağlılık kontrolü, güvenlik, yönlendirme ve verimli enerji tüketimi gibi önemli problemlere de sahiptir [5]. TDA'larda, genel olarak pil ile çalışan düğümler, ne kadar fazla mesaj gönderirlerse, o kadar hızlı güç kaynaklarını boşaltırlar. Bir düğümün güç kaynağı tükenirse, sadece o düğüm kapanmakla kalmaz aynı zamanda diğer düğümlerin arasındaki bağlantılar da kesilerek ağın büyük bir kısmı ulaşılamaz hale gelebilir [6]. Örneğin; Şekil 1.a'da bulunan 5 veya 6 numaralı düğümlerden herhangi biri kapanırsa, ağın büyük bir bölümü ile baz istasyonu arasındaki bağlantı tamamen kesilir. Dolayısıyla, TDA'larda gönderilen mesajların sayısı oldukça az ve yönlendirme algoritması da verimli olmalıdır. TDA'ların önemli problemlerinden bir diğeri de, diğer ağ yapılarına kıyasla, bu ağların saldırılara daha açık olmalarıdır.



(a) (b)

Şekil 1. Örnek bir TDA ve bu ağın çizge modeli.

Ağda bulunan bir saldırgan düğüm, tüm düğümlerin arasındaki iletişim protokolünü öğrendikten sonra, ağ alanına yeni düğümler ekleyerek, ağın doğru çalışmasını engelleyebilir, düğümlerin daha hızlı kapanmasını sağlayabilir veya düğümler arası iletilen verileri dinleyebilir [7]. Bu sebeple, TDA'larda düğümler arasında güvenilir iletişimin sağlanması ve doğru verilerin toplanması hayati önem taşımaktadır. Saldırganlar, güvensiz bir TDA'ya kötü amaçlı düğümler ekleme vb. çeşitli yöntemlerle saldırıp, TDA'nın doğru çalışmasını engelleyebilir. Örneğin; askeri bir operasyonda, hedefler veya askeri birliklerin konumu hakkında yanlış bilgi gönderen kötü amaçlı düğümler TDA alanına eklenebilir. Bir başka örnekle, yangın için oluşturulan TDA tabanlı bir erken uyarı sisteminde, ağ alanına yanlış veri gönderen veya mevcut mesajların iletimini engelleyen düğümler eklenirse, sistem yanlış çalışabilir. Bu sebeple, TDA'da var olan saldırgan düğümlerin tespiti oldukça önem taşımaktadır. Mevcut bir TDA'ya yanlış veri gönderen düğümler ekleyerek, ağın düzgün çalışmasına engel olan saldırı türüne Bizans saldırısı ismi verilmektedir [8]. Bu makale kapsamında gerçekleştirilen çalışmada, TDA'da olası Bizans saldırılarının tahminine yönelik topluluk tabanlı bir model önerilmektedir.

Makalenin diğer bölümleri şu şekilde düzenlenmiştir: ikinci bölümde, saldırı türleri ve savunma yöntemleri hakkında ön bilgiler verilmiştir. Üçüncü bölümde ise, bu çalışma kapsamında oluşturulacak olan TDA modeli hakkında detaylı bilgi verilmiştir. Dördüncü bölümde, topluluk öğrenmesi yaklaşımı ve bu yaklaşımda kullanılan başlıca yöntemlerden bahsedilmiştir. Ayrıca, TDA'da Bizans saldırılarının tespiti için önerilen topluluk tabanlı yaklaşım detaylı bir şekilde açıklanmıştır. Önerilen yaklaşımın veri seti üzerinde uygulanmasına yönelik gerçekleştirilen deneysel çalışmalar ve bu çalışmalardan elde edilen karşılaştırma sonuçları tablo ve grafikler halinde beşinci bölümde sunulmuştur. Altıncı bölümde, çalışmanın zafiyetlerinden bahsedilmiştir. Son bölümde ise, sonuç ve gelecekte yapılması planlanan çalışmalar hakkında bilgi verilmiştir.

2. Ön Bilgiler

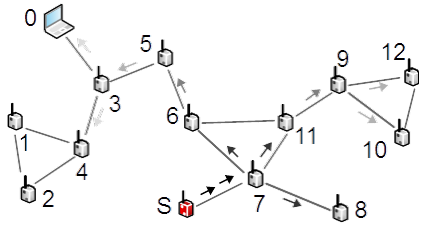
2.1. Saldırı Tipleri

TDA'lara karşı yapılan saldırılar, genel olarak aktif ve pasif olmak üzere ikiye ayrılır. Aktif saldırılarda saldırgan, ağın düzgün çalışmasını engellemek veya gönderilen verileri çalmak için kötü amaçlı yönlendirme yapan veya veri üreten düğümleri kullanır. Pasif saldırılarda ise, saldırgan tarafından hiç bir mesaj gönderilmez. Bu saldırıların amacı daha çok ağda gönderilen verileri dinlemek veya veri akışını analiz etmektir.

TDA'lara karşı Hello Flooding, DDoS, Sybil, Black Hole, Worm Hole, Selective Forwarding ve Bizans atakları gibi çeşitli aktif saldırılar başlatılabilir [9]. Örneğin; Hello Flooding [10]ve DDos [11] saldırılarında, saldırgan ağ alanına bazı düğümler yerleştirip, işe yaramaz mesajlar göndererek ağ trafiğini artırır ve normal verilerin akışını engeller. Bu durum aynı zamanda düğümlerin pilini daha hızlı boşaltarak ağın ömrünü kısaltır. Örneğin; Şekil 2'de saldırgan tarafından yerleştirilen S düğümü, diğer düğümlere sürekli gereksiz mesajlar göndererek bir Hello Flooding saldırısı gerçekleştirmektedir.

Black Hole saldırılarında [12], ağ alanına yerleştirilen kötü amaçlı düğümler, komşularına yanlış yön bilgileri sunarak mesajların kendilerine iletilmesini sağlayıp, gelen mesajları yok ederek ağın güvenilirliğini azaltabilirler. Worm Hole saldırılarında [13], saldırgan ağın çeşitli yerlerine birbiri ile haberleşebilen

düğümler yerleştirir. Saldırganın yerleştirdiği düğümler aralarındaki bağlantılar aracılığıyla, komşularına daha az maliyetli ve daha hızlı yön sunarak, mesajların kendilerine doğru yönlendirilmesini sağlarlar. Böylece ağda gönderilen verilerin önemli bir bölümü saldırganın yerleştirdiği düğümlerden geçer ve saldırgan mesajlar üzerinde istediği müdahaleyi gerçekleştirebilir.



Şekil 2. Yerleştirilen kötü amaçlı bir düğümün gereksiz mesaj gönderimi.

Sybil saldırılarında ise [14], ağa yerleştirilen kötü amaçlı bir düğüm aynı anda birden fazla kimlik bilgileri ile (farklı düğüm Id'si, IP adresi, kullanıcı ismi, vs) istediği verileri gönderip, ağda sanal olarak çoğunluğu kazanır. Bu çoklu sahte kimlikler, diğer düğümler tarafından gerçek ve benzersiz kimlikler gibi görünür. Böylece, çoğunluğa dayalı uygulamalarda, otoriteyi elde eden saldırgan, uygulamayı istediği gibi yönetir. Sybil saldırılarının temel amacı, ağda belirlenen kuralları etkisizleştirmek ve yasadışı eylemler gerçekleştirmektir.

Bir diğer saldırı türü olan Selective Forwarding saldırısında [15], kötü amaçlı düğümler komşularına yanlış yön bilgileri sunarak diğer düğümlerden gönderilen mesajları kendilerine yönlendirirler ve gelen mesajları saldırganın amacı doğrultusunda iletirler veya bırakırlar. Örneğin; bir güvenlik uygulamasında, saldırgan kendisine gelen normal mesajları iletirken tüm uyarı mesajlarını eleyebilir. Mesajların iletilmesi veya elenmesi işlemi sadece ağın paket kayıp oranını artırmak için tamamen rastgele de gerçekleştirilebilir.

Son olarak TDA'larda karşılaşılan bir diğer saldırı türü de Bizans saldırısıdır. Spectrum Sensing Data Falsification (SSDF) adıylada bilinen bu saldırıda, kötü amaçlı düğümler yanlış bilgiler göndererek uygulamanın yanılmasına sebep olurlar. Bu çalışmada, Bizans saldırılarının tespitine yönelik bir topluluk tabanlı model önerilmektedir.

2.2. Bizans Saldırısı

Bizans saldırılarında, sahte veya yanlış bilgiler gönderen kötü amaçlı düğümler ağ alanına yerleştirilir. Gönderilen sahte bilgiler, işleme merkezini yanıltarak yanlış kararlara veya gereksiz eylemlere sebep olabilir. Örneğin; bir yangın kontrol sisteminde, kötü amaçlı yerleştirilen bir düğüm ortam sıcaklığını çok fazla gösteren sahte bilgiler gönderir ise yanlış alarm ve gereksiz operasyonlara sebep olarak sistemin genel güvenilirliğini azaltabilir.

Bizans saldırılarının temel amacı, ağın güvenilirliğini azaltıp, dürüst ve kötü amaçlı düğümlerin ayırt edilmesini zorlaştırmaktır. Kötü amaçlı düğümlerin ayırt edilmesi zor olursa, sistemin yanlış kararlar vermesinin yanı sıra dürüst düğümlerden gelen kritik bilgiler de sahte bilgi olarak düşünülerek göz ardı edilebilir ve böylece ağın temel işlevleri yerine getirilemeyebilir.

Bizans saldırıları, özelikle veri algılama düğümleri sınırlı, algılanan verilerin hata payı yüksek veya verilerin varyansı fazla olan ağlarda daha başarılı olabilir. Zira, bu tür ağlarda doğru verileri ayırt edebilen karar mekanizmasının tasarımı zor ve verilen kararları destekleyen ek bilgilerin sayısı sınırlı olmaktadır.

2.3. Geleneksel Savunma Yöntemleri

Literatürde, Bizans saldırılarına karşı çeşitli istatistiksel savunma yöntemleri bulunmaktadır. Genel olarak, istatistiksel metotlar itibar kavramına dayanarak saldırganı, anormal davranışından tespit etmek ve ondan gelen verileri görmezden gelmektedirler. Örneğin; [16]'da önerilen savunma sisteminde, ağdaki her bir düğümün göndermiş olduğu verilerin ağırlıklı ortalamalarının hesaplanmasıyla verilerin doğruluk olasılığı belirlenir ve düşük olasılığa sahip olan veriler göz ardı edilir. Bunun yanı sıra, ağırlıklı sıralı olasılık oran testi [17], sıklık ve düzen analizi [18], Gaussian ve Regression modelleri [19] de gönderilen verilerin doğrulaması için önerilen diğer istatistiksel yöntemlerdir.

Mevcut diğer çalışmalarda, düğümlerin gönderdiği veriler ve çeşitli istatiksel metotlarla, düğümlerin şüpheli olma olasılıkları hesaplanır. Bu yöntemlerde, yüksek şüpheli düğümlerden gelen veriler göz ardı edilir [5,20-22]. Bu tür yöntemlerin temel özelliklerinden birisi, tutarsız veri gönderen dürüst düğümlere düşük

itibarların atanmasıdır. Ayrıca, bu yöntemlerde sahte verilerin sayısının doğru verilerden az olduğu varsayılmaktadır. Bu nedenle, fazla sayıda kötü amaçlı düğüm tarafından yapılan saldırılarda, itibara dayalı metotların başarı oranı düşüktür.

Literatürde, Bizans saldırılarının tespiti için itibara dayalı olmayan metotlar da bulunmaktadır. Örneğin, [23]'de önerilen yöntemde, düğümlerin arasındaki mesafe dikkate alınarak zararlı düğümler tespit edilmektedir. Bu yöntemde, her bir düğümün dürüst komşu düğümleri ile arasındaki mesafeler belirlenir ve belirsiz mesafelerden gelen veriler göz ardı edilir. Benzer bir şekilde, [24,25]'de önerilen yöntemler de düğümlerin arasındaki haberleşme frekansı ve düğümlerden alınan sinyal gücüne dayanarak zararlı düğümleri tespit etmeye çalışırlar. Ancak, bu yöntemlerde mesafe, frekans veya alınan sinyal gücüyle ilgili bilgiler bir takım genel varsayımlara dayandığı için çeşitli uygulamalarda başarısız olabilmektedirler. Herhangi bir ön varsayımda bulunmayan yöntemler daha çok insan yardımı ile gerçekleştirilmektedir [26]. Fakat insan gücü gerektiren bu işlemler yorucu ve zaman alıcı olmasından dolayı son yıllarda makine öğrenmesi tekniklerine dayalı sistemler geliştirilmeye başlanmıştır. Bu sistemler, varsayımdan uzak gerçek veriler üzerinde yüksek doğruluk oranıyla saldırı tespitini gerçekleştirebilmektedirler.

2.4. Makine Öğrenmesine Dayalı Savunma Yöntemleri

Mevcut durumda, geleneksel yöntemler ile telsiz ağlarda saldırıların tespitinde başarılı sonuçlar elde ediliyor olsa da, bu yöntemler özellikle büyük miktardaki haberleşme verilerinin performanslı bir şekilde işlenmesi ve bu verilerden karmaşık ilişkilerin elde edilmesi aşamalarında yetersiz kalabilmektedir. Bu sebeple son yıllarda, diğer birçok alanda olduğu gibi, telsiz ağlarda saldırıların tespiti alanında da makine öğrenmesi teknikleri yaygın bir şekilde kullanılmaya başlanmıştır.

Makine öğrenmesi kavramı, büyük miktarda ham verinin matematiksel ve istatistiksel yöntemlerle işlenerek, bu veri içerisinden tahminler yapılmasına olanak sağlayan, yapay zekanın bir alt dalıdır. Makine öğrenmesi, teknikleri gözetimli (sınıflandırma ve regresyon) ve gözetimsiz öğrenme (kümeleme ve birliktelik kuralı analizi) olmak üzere iki ana başlık altında toplanmaktadır. Literatürdeki ağ saldırılarının tespitine yönelik birçok çalışmada gözetimli öğrenme algoritmalarının kullanıldığı görülmektedir [27-29]. Mukherjee ve Sharma [27] Naive Bayes sınıflandırma algoritması kullanarak dört farklı (Probe (information gathering), DoS (deny of service), U2R (user to root) and R2L (remote to local)) saldırı türünün tespitini gerçekleştirmişlerdir. Sınıflandırma işlemi öncesi veri setindeki alakasız niteliklerin elenmesi işlemi için ise, CFS (Correlation-based Feature Selection), IG (Information Gain), GR (Gain Ratio) ve önermiş oldukları FVBRM (Feature Vitality Based Reduction Method) yöntemlerini uygulamışlardır. Diğer bir çalışmada ise, mobil ad hoc ağlarda saldırı tespiti için karar ağacı algoritması kullanılmıştır [29].

Topluluk öğrenmesi (ensemble learning) yaklaşımı ise, makine öğrenmesinin son yıllardaki en aktif alanlarından bir tanesidir. Bu öğrenme türünde temel amaç, tek bir sınıflandırıcı yerine birden fazla sınıflandırıcı kullanarak, her bir sınıflandırıcıdan elde edilen çıktıların bir oylama mekanizmasına sokulması sonucu tahminleme işleminin gerçekleştirilmesidir [30]. Özellikle sağlamış olduğu yüksek tahminleme performansı sayesinde, birçok alanda oldukça sık kullanılmaya başlanan topluluk öğrenmesi yaklaşımı, telsiz ağlarda saldırıların tespitinde de yüksek tahminleme becerisi sunmaktadır [31-35]. Örneğin, Tama ve Rhee [31] telsiz ağlarda saldırı tespiti için iki farklı topluluk öğrenmesi yöntemini: voting (oylama) ve stacking (yığılmış genelleme) önermiştir. Bu çalışmada, her iki yöntemde temel sınıflandırıcı olarak karar ağacı (decision tree), rastgele orman (random forest) ve destek vektör makinesi (support vector machine) algoritmaları kullanılmıştır. Bir diğer çalışmada ise, en yaygın kullanılan topluluk öğrenmesi algoritmalarından olan AdaBoost algoritması kullanılarak ağ saldırı tespit sistemi geliştirilmiştir [32].

Bu çalışmanın literatüre sağlayacağı başlıca yenilikçi yönleri: (1) TDA'da saldırıların tespitinde 7 farklı sınıflandırıcının (Naive Bayes, C4.5, k-NN, Bagging (C4.5), Boosting (AdaBoost), Voting ve Stacking) kullanılması (2) topluluk öğrenmesi yaklaşımının ilk kez TDA'da Bizans saldırılarının tespitinde uygulanıyor olması

(Bagging (C4.5), Boosting (AdaBoost), Voting ve Stacking) (3) Voting ve stacking yöntemleri ile 3 farklı geleneksel sınıflandırma algoritmasının (Naive Bayes, C4.5 ve k-NN) bir araya getirilmesiyle iki yeni yaklaşımın elde edilmesi (4) deneysel çalışmalar kapsamında uygulanan 7 farklı sınıflandırıcının, örnek bir TDA'da Bizans saldırılarının tespitinde sağlamış oldukları doğruluk oranlarının karşılaştırılması.

3. Ağ Modeli

TDA'ları bir G(V,E) çizge ile modelleyebilmek mümkündür. Bu modelde, V düğümlerin kümesini ve E ise düğümlerin arasında bulunan bağlantıların kümesini sembolize etmektedir. Bu ağ modelinde, birbirinin radyo iletişim alanında olan düğümlerin arasında bir haberleşme kanalının olduğunu varsayabiliriz. Şekil 1.b'de gösterilen örnek TDA'da düğüm kümesi V={0,1,2,…,12} iken, bağlantı kümesi ise E={(0,3),(1,2),(1,4),…(10,12,)} şeklindedir. Bu çalışmada TDA'lar ile ilgili aşağıdaki varsayımlar bulunmaktadır:

- Her düğümün kendine ait bir kimliği (id) vardır.
- Tüm düğümler hareketsizdir. Dolayısıyla algoritma çalışırken düğümlerin arasındaki bağlantılar değişmez.
- Düğümlerin arasındaki bağlantılar simetriktir. Örneğin; eğer u düğümünden v düğümüne bir bağlantı varsa, v'den u'ya da bir bağlantı vardır.
- Düğümler GPS alıcısı gibi pozisyon izleyici modülüne sahip değillerdir ve bu sebeple konum bilgilerini bilmeyebilirler.
- Düğümlerin işlemcileri, iletişim donanımları, enerji tüketim oranları ve bellek kapasiteleri aynıdır.
- Düğümler ağ alanına rasgele dağıtılmıştır ve aralarında herhangi bir düzen yoktur.
- En az bir baz istasyonu, bir merkezi veri işleme bilgisayarına bağlıdır ve bu baz istasyonu verileri ağdan toplayıp, bilgisayara aktarır.
- Verileri toplayan bilgisayar ağdan gelen tüm verileri kendi belleğinde veya bulut üzerinde depolayabilir ve böylece her an bu verilere ulaşabilir.

Ayrıca TDA'da düğümlerin arasında fiziksel ve ortam erişim katmanlarına (İng. Medium Access Control(MAC)) uygun protokollerin çalıştığı ve düğümlerin mesajları bir minimum kapsayan ağaç üzerinden baz istasyonuna doğru yönlendirdikleri varsayılmaktadır.

4. Topluluk Öğrenmesi

Topluluk öğrenmesi yönteminde, geleneksel makine öğrenmesi yöntemlerinden farklı olarak, bir örneklemin çıktı değerini tahmin etmek için tek bir sınıflandırıcı yerine birden fazla sınıflandırıcı kullanılır [36]. Örneğin; bu çalışmada, TDA'da yer alan bir düğümün saldırgan olup olmadığının tahmin edilmesi için geleneksel sınıflandırma yöntemlerinden farklı olarak birden fazla sınıflandırıcı kullanılır. Her bir sınıflandırıcı eğitim veri seti (TDA'dan elde edilen veriler) ile eğitilir ve birden fazla sınıflandırma modeli elde edilir. Tahmin edilecek olan örneklem modellere girdi olarak verilir ve her bir modelden elde edilen çıktı bir oylama mekanizmasına sokulur. Bu oylama mekanizmasında modellerden elde edilen çıktılar arasında çoğunluğa sahip olan değer nihai sınıf etiketi (saldırgan ya da dürüst) olarak seçilir.

Literatürdeki mevcut birçok çalışmada [37-39], topluluk öğrenmesi yaklaşımının, klasik tek sınıflandırıcının kullanıldığı makine öğrenmesi yöntemlerine göre daha yüksek doğruluk oranına sahip sınıflandırma performansı gösterdiği görülmüştür. Topluluk öğrenmesi teknikleri kendi içerisinde bagging, boosting, stacking ve voting gruplarına ayrılmaktadır.

4.1. Bagging

Bagging (Bootstrap Aggreating) yönteminde ilk olarak veri seti içerisinden rastgele kayıtlar seçilerek birden fazla eğitim veri seti oluşturulur. Bu yönteme önyükleme (bootstrap) denilmektedir. Daha sonrasında ise, sınıflandırıcılar farklı eğitim setleri ile eğitilerek farklı sınıflandırma modelleri elde edilir. Son olarak da, sınıf etiketi tahmin edilecek olan örneklem farklı sınıflandırma modellerine girdi olarak verilir ve her bir modelden elde edilen çıktılar oylama mekanizmasına sokularak sınıf etiketi belirlenmiş olur. Bu çalışmada, C4.5 algoritması tabanlı Bagging yöntemi kullanılmıştır.

4.2. Boosting

Bu yöntemdeki ana fikir, birden fazla zayıf öğreniciyi bir araya getirerek güçlü bir öğrenici

elde etmektir. Bagging yönteminden farklı olarak bu teknikte, sınıflandırıcılar birbiri ardına eğitilir ve bu sayede öğrenicilerin zayıftan güçlüye dönüştürülmesi işlemi gerçekleştirilir.

Bu çalışmada, boosting yönteminin en bilinen algoritması olan AdaBoost algoritması kullanılmıştır. Bu algoritmada, eğitim setindeki her bir örnekleme başlangıçta bir ağırlık değeri atanır. Her yinelemede, doğru sınıflandırılan örneklemlerin ağırlık değeri düşürülürken, yanlış sınıflandırılan örneklemlerin ağırlık değerleri ise artırılmaktadır. Bu yaklaşıma göre, yanlış sınıflandırılan örneklemlerin, sahip oldukları ağırlık değerleri sebebiyle, eğitim verisine seçilme şansı yükselir. Böylece, sınıflandırıcının tahminleme başarısı artırılmış olur.

4.3. Stacking

Bagging ve Boosting yöntemlerinden farklı olarak Stacking yönteminde birden fazla sınıflandırma algoritması aynı eğitim seti ile eğitilir ve bu sayede birden fazla sınıflandırma modeli elde edilir. Her bir modelden elde edilen çıktılar, ara bir katmandaki meta sınıflandırıcıya girdi olarak gönderilir ve elde edilen çıktı sınıf etiketi olarak belirlenir.

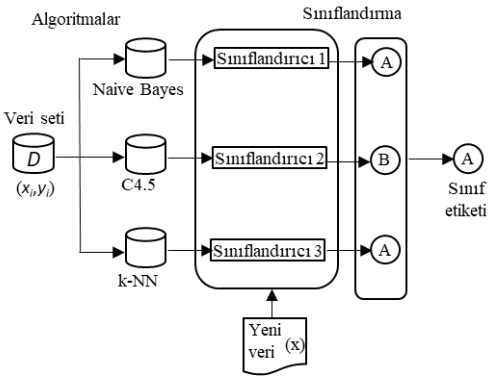
4.4. Voting

Voting yönteminde de Stacking'e benzer şekilde birden fazla sınıflandırma algoritması aynı eğitim seti ile eğitilebilir ya da tek bir algoritma aynı veri seti ile farklı parametre değerleri kullanılarak eğitilebilir. Böylece farklı sınıflandırma modelleri oluşturulur ve modellerden elde edilen çıktılar oylama mekanizmasına sokularak nihai çıktı değeri üretilir.

Bu makalede, sınıflandırma alanında en çok tercih edilen üç farklı algoritma (Naive Bayes, C4.5 ve k-NN) [40-42] kullanılarak voting ve stacking yöntemi ile iki yeni bir topluluk öğrenmesi modeli oluşturulmuştur.

- **Naive Bayes:** Naive Bayes algoritması, Thomas Bayes tarafından sunulan Bayes Teoremi'ne dayalı istatistiksel bir sınıflandırma algoritmasıdır. Bu algoritmaya göre, sınıf etiketi önceden belli eğitim verilerindeki her bir öznitelik birbirinden bağımsız ele alınarak, bu öznitelik değerlerinin sınıf etiketine göre olasılıkları elde edilir. Tahminlenecek olan örneklemin sınıf etiketi, elde edilen tüm bu olasılıklardan en yüksek değere sahip olan seçilerek belirlenir.
- **Karar Ağacı (C4.5):** Karar ağacı algoritmasında ise, yeni gelen örneklemin sınıf etiketini belirlemek için eğitim verilerinin öznitelik ve değerlerinden oluşan bir ağaç yapısı oluşturulur**.** Bu ağaç yapısında, düğümler eğitim verisinin özniteliklerinden, dallar öznitelik değerlerinden ve yapraklar ise sınıf etiketlerinden meydana gelmektedir. Literatürde birçok karar ağacı algoritması bulunmaktadır: C4.5, C5, ID3, CART ve CHAID. Sağlamış olduğu yüksek sınıflandırma becerisi sayesinde en çok tercih edilen karar ağacı algoritmalarından biri olması sebebi ile bu çalışmada C4.5 algoritmasından yararlanılmıştır.
- **K-En Yakın Komşuluk (İng. k-NN):** Uygulaması en kolay sınıflandırma algoritmalarından biri olan k-en yakın komşuluk algoritmasında, yeni bir örneklem sınıflandırılacağı zaman kendisine en yakın k (kullanıcı tanımlı sabit bir sayı) kayda bakılır. Bu k kaydın sahip olduğu sınıf etiketlerinin çoğunluğuna bakılarak yeni örneklemin sınıf etiketi belirlenir. Sınıflandırılacak olan örnekleme en yakın k kaydın bulunmasında ise literatürdeki mevcut uzaklık ölçütleri (sayısal veriler için: Euclidean, Manhattan ve Minkowski, kategorik değerler için: Hamming) kullanılmaktadır.

Bu çalışmada önerilen yeni topluluk tabanlı yaklaşımlarda (stacking ve voting) , yukarıda bahsedilen üç geleneksel sınıflandırma algoritması temel algoritma olarak seçilmiştir. Geliştirilen bu yöntemde, çalışma kapsamında oluşturulan örnek TDA'dan elde edilen eğitim verileri her bir sınıflandırma algoritması ile eğitilip birden fazla sınıflandırma modeli (sınıflandırıcı) elde edilir. Bir düğümün saldırgan olup olmadığının belirlenmesi için, o düğüme ait öznitelikler her bir modele girdi olarak verilir ve bu modellerden elde edilen çıktılar bir oylama mekanizmasına sokulur. Düğüm, bu oylama mekanizmasında, çoğunluğa sahip olan sınıf değeri ile etiketlenir. Sistemin genel mimarisi Şekil 3'te sunulmuştur.



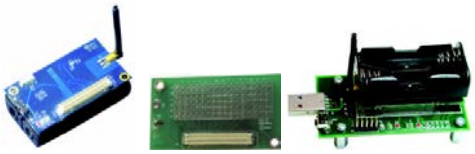
Şekil 3. Önerilen sistem mimarisi.

5. Deneysel Çalışma

Bu çalışmada, 7 farklı sınıflandırıcı (Naive Bayes, C4.5, k-NN, Bagging (C4.5), Boosting (AdaBoost), Voting ve Stacking) oluşturulan örnek TDA'da Bizans saldırısının tespiti amacıyla uygulanarak test edilmiş ve göstermiş oldukları doğruluk oranlarına göre karşılaştırılmıştır. Bu çalışma kapsamındaki sınıflandırma uygulaması, Weka açık kaynak kodlu veri madenciliği kütüphanesi kullanılarak geliştirilmiştir[43].

5.1. Veri Seti

Bu çalışma kapsamında uygulanan makine öğrenmesi tekniklerinin tahminleme başarısını değerlendirmek için, Ege Üniversitesi Uluslararası Bilgisayar Enstitüsü binasının çeşitli yerlerine 66 IRIS düğümü (Şekil 4.a) yerleştirerek, havanın sıcaklığını ölçen bir TDA oluşturulmuştur. IRIS düğümleri TinyOS işletim sistemini destekleyip, 2.4 GHz frekansında, 50 m iç mekan radyo menzili ve 250 kbps radyo veri hızında çalışmaktadır. Yerleştirilen 66 adet düğümün, 60 tanesi dürüst ve 6 tanesi ise saldırgan düğüm olarak belirlenmiştir. Dürüst düğümler her zaman ölçtükleri gerçek sıcaklık değerlerini gönderirken, saldırgan düğümler ölçtükleri sıcaklık değerleri ile beraber ara sıra sahte veriler de göndermektedirler. Çalışma kapsamında önermiş olduğumuz yöntemle, bir düğümün dürüst ya da saldırgan olma durumu tahmin edilmektedir.

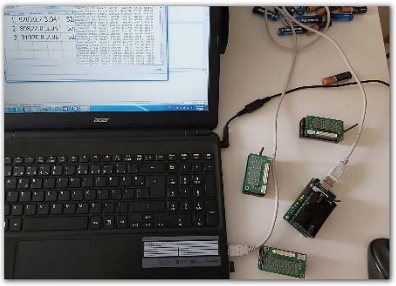


(a) (b) (c)

Şekil 4. a) IRIS düğümü, b) MDA100 algılayıcı, c) MIB520 baz istasıyonu.

Oluşturduğumuz ağda, IRIS düğümler periyodik olarak dakika başı hava sıcaklığını ölçüp, bu verileri çok sekmeli bağlantılar üzerinden baz istasyonuna göndermişlerdir. Hava sıcaklığı, IRIS düğümlerin üzerine takılabilen bir MDA100 (Şekil 4.b) algılayıcı tarafından ölçülmüştür. Baz istasyonu için bir IRIS düğümü, MIB520 donanım aracılığıyla (Şekil 4.c) USB portu üzerinden bilgisayara bağlanmış ve gelen tüm veriler bir Java uygulaması aracılığıyla USB portundan okunup veri tabanına eklenmiştir.

(a)



(b)

Şekil 5. Test yatağında kullanılan düğümler ve baz istasyonu.

Şekil 5.a test yatağında kullanılan düğümleri göstermektedir. Düğümler tarafından gönderilen veriler Şekil 5.b'de gösterilen baz istasyonu aracılığıyla bilgisayarda çalışan Java uygulamasına aktarılmıştır. Düğümler tarafından gönderilen verilerin formatı aşağıdaki gibidir:

```
Message {
  int data;      // 2 baytlık hava sıcaklığı
  int sID;       //2 baytlık kaynak düğüm kimliği
  int mNo;       // 2 baytlık mesaj numarası
}
```

Her mesajda, algılanan 2 baytlık hava sıcaklığı değeri, veriyi algılayan kaynak düğümün kimlik numarası ve mesaj numarası tutulmuştur. Baz istasyonunun kimlik numarası 0, diğer düğümlerin kimlik numaraları ise 1 ile 66 arasındaki değerler olarak seçilmiştir. Her düğümün göndermiş olduğu ilk mesaj numarası 1 ve ardından gönderdiği her mesajın numaraları da birer artan değerler olarak belirlenmiştir.

Geliştirilen dağıtık uygulamada, düğümler sıcaklık ölçmeye başlamadan önce baz istasyonundan bir Start mesajının gelmesini beklerler. Baz istasyonu Start mesajı göndererek düğümlerin aynı zamanda başlamasını ve aynı zamanda düğümlerin arasında bir kapsayan ağacın oluşmasını sağlar. İlk defa Start mesajını alan her düğüm, gönderen düğümü kendi ebeveyni olarak seçip, mesajı komşularına yayınlar (Broadcast) ve böylece düğümler arasında kökü baz istasyonu olan bir kapsayan ağaç oluşur. Her düğüm ölçtüğü sıcaklık değeri veya diğer düğümlerden gelen mesajları ebeveyn düğümüne göndererek mesajların baz istasyonuna ulaşmasını sağlar. Mesajların çakışma durumunu azaltmak için, düğümler her mesajı göndermeden önce 0 ile 10s arasında rastgele değişen sürelerde beklerler. Bu çalışma kapsamında, çakışan mesajlar göz ardı edilmektedir.

Saldırgan düğümler, sahte veri gönderme sıklığına göre üç gruba (her grup ikişer düğüm) ayrılmıştır. Birinci grupta, saldırgan düğümler her bir doğru veriden sonra bir sahte veri gönderdiler. İkinci gruptaki saldırgan düğümler, her 3 doğru veriden sonra ve üçüncü gruptaki saldırgan düğümler ise her 5 doğru veriden sonra bir sahte veri gönderdiler. Sahte veriler, gerçek ölçülen değerlere 2 ile 6 arasında değişen rastgele değerler eklenerek üretilmiştir. Düğümler 48 saat boyunca, her dakikada bir mesaj gönderdiler. Böylece toplam $48 \times 60 \times 66 = 190080$ mesaj baz istasyonuna gönderilmiştir. Elde edilen veri setinde toplam 66 kayıt (her düğüm için bir kayıt) bulunmaktadır. Her bir kayıt $48 \times 60 = 2880$ ortam sıcaklık verisi, düğümlerin kimlik numaraları ve saldırgan veya normal olduklarını gösteren sınıf etiketleriyle beraber toplam 2882 öznitelik barındırmaktadır. Test yatağın özellikleri tablo 1'de verilmiştir.

Tablo 1. Geliştirilen test yatağın özellikleri.

Özellik	*Sütun Başlığı*
Deney süresi	48 saat
Düğüm sayısı	60 dürüst, 6 saldırgan, (66 toplam)
Sahte verilerin sıklığı	1, 3, 5
Veri gönderme oranı	1 dakika
Veri tipi	Ortam sıcaklığı
Yönlendirme	Kapsayan Ağaç

5.2. Değerlendirme

Deneysel çalışma kapsamında, 7 farklı sınıflandırıcı (Naive Bayes, C4.5, k-NN, Bagging (C4.5), Boosting (AdaBoost), Voting ve Stacking), havanın sıcaklığını ölçmek için oluşturulan bir telsiz duyarga ağı üzerinde ayrı ayrı uygulanmış ve 10-katlı çapraz geçerlilik tekniği kullanılarak test edilmiştir. Elde edilen sınıflandırma sonuçları doğruluk oranı ve f-ölçüm değerlerine göre karşılaştırılmıştır.

Doğruluk oranı, test verisi içerisindeki doğru sınıflandırılan kayıtların, toplam kayıt sayısına oranını ifade etmektedir (1). Denklemde DP, DN, YP ve YN sırasıyla doğru pozitif, doğru negatif, yanlış pozitif ve yanlış negatif değerlerini belirtmektedir. Doğru pozitif ve doğru negatif değerler, doğru sınıflandırılan pozitif ve negatif değerleri, yanlış pozitif ve yanlış negatif değerler ise yanlış sınıflandırılan değerleri göstermektedir.

$$Doğruluk = \frac{DP + DN}{DP + DN + YP + YN} \qquad (1)$$

F-ölçüm ise, sınıflandırma algoritmalarının başarısını ortaya koymak için kullanılan bir diğer doğruluk ölçütüdür. Bir algoritmanın F-ölçüm değeri kesinlik (precision) ve hassasiyet (recall) değerlerinin harmonik ortalaması alınarak hesaplanmaktadır (2).

$$F - \text{ölçüm} = \frac{2 * Kesinlik * Hassasiyet}{Kesinlik + Hassasiyet} \qquad (2)$$

Kesinlik değeri, doğru pozitif değerlerin tüm pozitif değerlere oranını verirken (3), hassasiyet değeri ise doğru pozitif değerlerin, doğru pozitif

ve yanlış negatif değerlere oranı ile hesaplanmaktadır (4).

$$Kesinlik = \frac{DP}{DP + YP} \quad (3)$$

$$Hassasiyet = \frac{DP}{DP + YN} \quad (4)$$

Her bir sınıflandırıcıdan elde edilen DP, DN, YP ve YN değerleri Tablo 2'de, sınıflandırma doğruluk oranları ile f-ölçüm, kesinlik ve hassasiyet değerleri Tablo 3'te sunulmuştur. Sonuçlara göre, uygulanan algoritmalar arasında %98.48 ile en yüksek doğruluk oranına sahip algoritmanın voting yöntemi ile geliştirilen topluluk tabanlı sınıflandırıcı olduğu görülmektedir. Dolayısıyla, örnek ağ üzerindeki Bizans saldırısının tespitinde en başarılı yöntemin, önerilen yaklaşımlardan olan voting yöntemi olduğu anlaşılmaktadır. Ayrıca, önerilen bir diğer yaklaşım olan stacking yönteminin de %95.45 ile yüksek bir sınıflandırma becerisi sunduğu görülmektedir. Bunun yanı sıra, bu çalışma kapsamında uygulanan tüm sınıflandırma algoritmalarının %90 'ın üzerinde doğruluk oranı sunması, makine öğrenmesi tekniğinin telsiz duyarga ağlarda Bizans saldırılarının tespitinde oldukça başarılı olduğunu ortaya koymaktadır.

Tablo 2. Sınıflandırma algoritmalarının DP, DN, YP ve YN değerleri.

Algoritma	*DP*	*DN*	*YP*	*YN*
Naive Bayes	54	9	1.91	1.09
C4.5	54.09	9.91	1	1
k-NN (k=3)	54.73	7.27	3.64	0.36
Bagging (C4.5)	54.73	7.27	3.64	0.36
Boosting (AdaBoost)	54.82	8.18	2.73	0.27
Voting	55	10	0.91	0.09
Stacking	54.82	8.18	2.73	0.27

Tablo 3. Sınıflandırma algoritmalarının doğruluk oranı ve f-ölçüm değerlerine göre karşılaştırılması.

Algoritma	*Kesinlik*	*Hassasiyet*	*F-ölçüm*	*Doğruluk Oranı(%)*
Naive Bayes	0.95	0.96	0.95	95.45
C4.5	0.97	0.97	0.97	96.97
k-NN (k=3)	0.94	0.94	0.93	93.94
Bagging (C4.5)	0.94	0.94	0.93	93.94
Boosting (AdaBoost)	0.96	0.96	0.95	95.45
Voting	0.99	0.99	0.98	**98.48**
Stacking	0.96	0.96	0.95	95.45

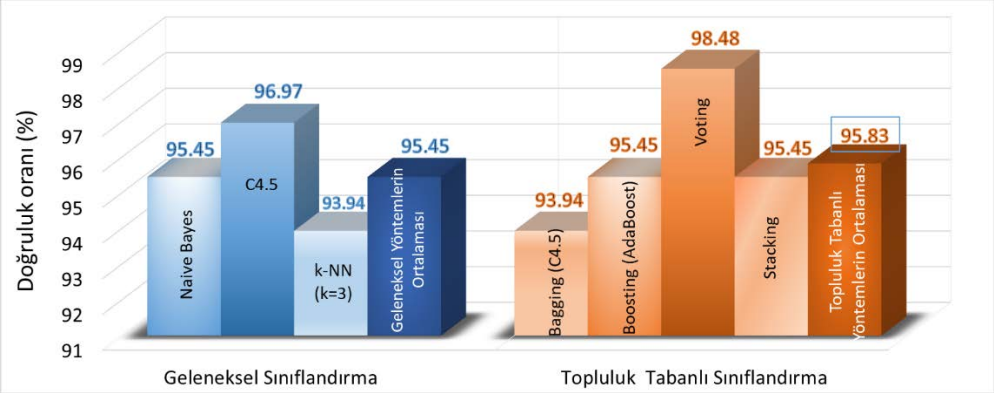
Ayrıca, geleneksel sınıflandırma algoritmaları (Naive Bayes, C4.5 ve k-NN) ile topluluk tabanlı yöntemlerin (Bagging, Boosting, Voting ve Stacking) sunmuş olduğu doğruluk oranlarının ortalaması Şekil 6'da sunulmuştur. Grafiğe göre, topluluk tabanlı yöntemlerin, telsiz duyarga ağlarda Bizans saldırılarının tespitinde geleneksel (tek bir sınıflandırma modeli kullanan) yöntemlere göre yaklaşık %2 daha başarılı olduğu görülmektedir. Çok sayıda düğüm içeren daha büyük ağlarda, topluluk tabanlı yöntemler ile geleneksel yöntemlerin başarı oranlarının arasındaki fark gözle görülür şekilde artabilir. Dolayısıyla, daha büyük ağlarda, önerilen topluluk tabanlı yöntemler, daha fazla sayıda saldırıyı önleyebilir. Ayrıca, çalışma kapsamında kullanılan sınıflandırma algoritmaları çalışma süreleri ve süre karmaşıklıklarına göre kıyaslanmıştır. N kayıt sayısı, D öznitelik sayısı, T ağaç sayısı olmak

üzere, geleneksel sınıflandırma algoritması olan Naive Bayes, C4.5 ve k-NN algoritmalarının karmaşıklıkları sırasıyla *O(ND), O(ND²)* ve *O(ND+kN)* iken, topluluk tabanlı Bagging (C4.5) ve Boosting (AdaBoost) algoritmalarının *O(TD²N²)* ve *O(TN)*'dir. Voting ve Stacking yöntemlerinin karmaşıklıkları ise, kullanılan algoritmaların karmaşıklığına ve sayısına göre değişiklik göstermektedir. Stacking yönteminde ayrıca meta sınıflandırıcı karmaşıklığı da eklenmektedir.

Algoritmaların çalışma süreleri Tablo 4'te sunulmuştur. Sonuçlar göstermektedir ki, geleneksel yöntemler topluluk tabanlı yöntemlere göre daha düşük doğrulukla daha kısa sürede işlem gerçekleştirebilmektedir. Bunun sebebi, topluluk tabanlı yöntemlerde, geleneksel yöntemlerin aksine birden fazla model oluşturularak doğruluk oranının arttırılmasıdır. Dolayısıyla, topluluk tabanlı yöntemlerin işlem süresindeki artış öngörülebilir düzeyde olup, bu yöntemler daha yüksek sınıflandırma becerisi sunmaktadır.

Tablo 4. Sınıflandırma algoritmalarının çalışma sürelerine göre karşılaştırılması.

Algoritma	*Çalışma Süresi (sn.)*
Naive Bayes	0.08
C4.5	0.09
k-NN (k=3)	0.02
Bagging (C4.5)	0.63
Boosting (AdaBoost)	0.05
Voting	0.12
Stacking	1.5



Şekil 6. Sınıflandırma yöntemlerinin ortalama doğruluk oranlarına göre karşılaştırılması.

6. Zafiyetler

Bu çalışmada, test yatağı olarak kullanılan TDA'da, IRIS düğümleri sabit konumlara yerleştirilip, her düğümün rolü (saldırgan veya normal) ve veri gönderme oranı deney boyunca sabit tutulmuştur. Dolayısıyla, deneylerin farklı topolojilere sahip çeşitli TDA 'lar üzerinde tekrarlanması ve her deneyde, düğüm sayısı, düğümlerin radyo iletişim menzili, veri toplama oranı, normal ve saldırgan düğümlerin rolleri ve düğümlerin konumlarının değişmesi, daha tarafsız sonuçlar verebilir.

Elimizde sınırlı sayıda IRIS düğümü bulunduğundan dolayı, deneyler 66 düğümden oluşan bir ağ üzerinde gerçekleştirilmiştir. Önerilen yaklaşımların performanslarını değerlendirebilmek için, deneylerin daha büyük ağlar üzerinde tekrarlanması gerekmektedir. Taşınabilir veya hareketli (mobil) ağlarda düğümler tarafından gönderilen veriler sürekli olarak ve daha sık değişebildiğinden dolayı, çalışmada önerilen yaklaşımların bu ağlardaki sınıflandırma becerisi sınırlı kalabilir.

Bu çalışmada, düğümlerin Id ve IP numaralarının rastgele dağıtılması ve bu bilgilerin kimlik

doğrulamasında kullanılmaması varsayılmıştır. Böylece, saldırgan rastgele bir Id ve IP kullanarak yanlış veriler gönderebilir. Id ve IP üzerinden kimlik doğrulaması yapılan sistemlerde, saldırganın ilk adımı geçerli bir Id ve IP değeri bulmak olmalıdır. Saldırgan, daha sonraki aşamalarda, ele geçirdiği bu bilgileri kullanarak yanlış veriler gönderebilir.

7. Sonuç

TDA'larda önemli çelişkilerden birisi iletilen verilerin güvenilirliğidir. Bu ağlarda, düğümler birbirleriyle radyo mesajları üzerinden dağıtık bir şekilde haberleştikleri için çeşitli saldırılara maruz kalabilmektedirler. Bizans saldırılarında, saldırgan TDA'ya sahte veri üreten düğümler ekleyerek, yanlış bilgi elde edilmesine sebep olabilmektedir. Bu makale kapsamında gerçekleştirilen çalışmada, TDA'larda Bizans saldırılarının tespitine yönelik topluluk tabanlı yaklaşımlar önerilmiştir. Bu yaklaşımlar, 3 farklı geleneksel sınıflandırma algoritmasının (Naive Bayes, C4.5 ve k-NN) voting ve stacking yönetimleri ile bir araya getirilmesinden oluşmaktadır. Önerilen yaklaşımların sınıflandırma performansını değerlendirmek için 66 IRIS düğümü içeren bir TDA oluşturulmuştur. Oluşturulan TDA'da ortam sıcaklık verileri, düğümler tarafından algılanıp, merkezi bir baz istasyonuna iletilmiştir. Yapılan deneysel çalışmalarda, önerilen yaklaşımlar ile birlikte 7 farklı sınıflandırıcı (Naive Bayes, C4.5, k-NN, Bagging (C4.5), Boosting (AdaBoost), Voting ve Stacking), oluşturulan örnek TDA üzerinde uygulanmıştır. Her bir algoritmadan elde edilen sınıflandırma sonuçları, doğruluk oranı ve f-ölçüm değerlerine göre karşılaştırılmıştır. Deneysel sonuçlarda, TDA'da Bizans saldırılarının tespitinde en yüksek doğruluk oranına sahip olan yöntemin önerilen %98.48 ile voting olduğu ve önerilen bir diğer yaklaşım olan stacking yönteminin de %95.45 ile yüksek bir sınıflandırma becerisi sunduğu görülmektedir. Ayrıca, topluluk tabanlı yöntemlerin (Bagging (C4.5), Boosting (AdaBoost), Voting ve Stacking) ortalama doğruluk oranı %95.83 iken, geleneksel sınıflandırma algoritmalarının (Naive Bayes, C4.5 ve k-NN) %95.45 olarak elde edilmiştir. Dolayısıyla, TDA'da Bizans saldırılarının tespitinde topluluk tabanlı yaklaşımların, geleneksel sınıflandırma algoritmalarına göre daha başarılı olduğunu söylemek mümkündür. Gelecek çalışma olarak, topluluk tabanlı yöntemlerin Black Hole, Hello Flooding, Sybil gibi farklı saldırı türlerinin tespitine yönelik uygulanması amaçlanmaktadır. Ayrıca, sınıflandırma ve kümeleme algoritmalarını birleştiren yeni bir hibrit yaklaşım ile TDA saldırılarının yüksek doğruluk oranı ile tespitinin sağlanması hedeflenmektedir.

Kaynakça

[1] Akyildiz, I. F., Su, W., Sankarasubramaniam, Y., Cayirci, E. 2002. Wireless sensor networks: a survey. Computer networks, Cilt. 38(4), s. 393-422.

[2] Yu, L., Wang, N., Meng, X. 2005. Real-time forest fire detection with wireless sensor networks. International Conference on Wireless Communications, Networking and Mobile Computing, Cilt. 2, s. 1214-1217, IEEE.

[3] Arslan, S., Challenger, M., Dagdeviren, O. 2017, Wireless sensor network based fire detection system for libraries. International Conference on Computer Science and Engineering (UBMK) s. 271-276, IEEE.

[4] Karimpour, N., Karaduman, B., Ural, A., Challenger, M., Dagdeviren, O, 2019, IoT based Hand Hygiene Compliance Monitoring. In International Symposium on Networks, Computers and Communications (ISNCC), s. 1-6, IEEE.

[5] Karlof, C., Wagner, D. 2003, Secure routing in wireless sensor networks: Attacks and countermeasures. Ad hoc networks, Cilt 1(2-3), s. 293-315.

[6] Dağdeviren, O., Akram, V. K. 2017. TinyOS Tabanlı Telsiz Duyarga Ağları için Bir Konumlandırma ve k-Bağlılık Denetleme Sistemi. Bilişim Teknolojileri Dergisi, Cilt. 10(2), s.139-152.

[7] Pathan, A. S. K., Lee, H. W., Hong, C. S. 2006. Security in wireless sensor networks: issues and challenges. 8th International Conference Advanced Communication Technology, Cilt. 2, s. 1043-1048. IEEE.

[8] Rawat, A. S., Anand, P., Chen, H., Varshney, P. K. 2010, Collaborative spectrum sensing in the presence of Byzantine attacks in cognitive radio networks. IEEE Transactions on Signal Processing, Cilt. 59(2), s. 774-786.

[9] Padmavathi, D. G., Shanmugapriya, M. 2009. A survey of attacks, security mechanisms and challenges in wireless sensor networks. arXiv preprint arXiv:0909.0576.

[10] Salam, M. A., Halemani, N. 2016. Performance evaluation of wireless sensor network under hello flood attack. International Journal of Computer networks & Communications (IJCNC), Cilt 8(2).

[11] Abidoye, A. P., Obagbuwa, I. C. 2017. DDoS attacks in WSNs: detection and countermeasures. IET Wireless Sensor Systems, Cilt. 8(2), s. 52-59.

[12] Otoum, S., Kantarci, B., Mouftah, H. T. 2017. Hierarchical trust-based black-hole detection in WSN-based smart grid monitoring. 2017 IEEE International Conference on Communications (ICC) (pp. 1-6). IEEE.

[13] Amish, P., Vaghela, V. B. 2016. Detection and prevention of wormhole attack in wireless sensor

network using AOMDV protocol. Procedia computer science, Cilt. 79, s. 700-707.

[14] Alsaedi, N., Hashim, F., Sali, A., Rokhani, F. Z. 2017. Detecting sybil attacks in clustered wireless sensor networks based on energy trust system (ETS). Computer communications, Cilt. 110, s. 75-82.

[15] Ren, J., Zhang, Y., Zhang, K., Shen, X. 2016. Adaptive and channel-aware detection of selective forwarding attacks in wireless sensor networks. IEEE Transactions on Wireless Communications, Cilt. 15(5), s. 3718-3731.

[16] Oh, S. H., Hong, C. O., Choi, Y. H. 2012. A malicious and malfunctioning node detection scheme for wireless sensor networks. Wireless sensor network, Cilt. 4(03), s. 84-90.

[17] Alizadeh, H., Sharifi, A. A., Niya, M., Javad, M., Seyedarabi, H. 2017. Attack-aware cooperative spectrum sensing in cognitive radio networks under Byzantine attack. Journal of Communication Engineering, Cilt. 6(1), s. 81-98.

[18] He, X., Dai, H., Ning, P. 2013. A Byzantine attack defender in cognitive radio networks: The conditional frequency check. IEEE Transactions on Wireless Communications, Cilt. 12(5), s. 2512-2523.

[19] Zhang, P., Koh, J. Y., Lin, S., Nevat, I. 2014. Distributed event detection under byzantine attack in wireless sensor networks. 2014 IEEE Ninth International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP) s. 1-6, IEEE.

[20] Curiac, D. I., Banias, O., Dragan, F., Volosencu, C., Dranga, O. 2007. Malicious node detection in wireless sensor networks using an autoregression technique. In International Conference on Networking and Services (ICNS'07) s. 83-83. IEEE.

[21] Wang, W., Li, H., Sun, Y., Han, Z. 2009. Securing collaborative spectrum sensing against untrustworthy secondary users in cognitive radio networks. EURASIP Journal on Advances in Signal Processing, 2010, s. 1-15.

[22] Kaligineedi, P., Khabbazian, M., Bhargava, V. K. 2010. Malicious user detection in a cognitive radio cooperative sensing system. IEEE Transactions on Wireless Communications, Cilt. 9(8), s. 2488-2497.

[23] Li, H., Han, Z. 2010. Catch me if you can: An abnormality detection approach for collaborative spectrum sensing in cognitive radio networks. IEEE Transactions on Wireless Communications, Cilt. 9(11), s. 3554-3565.

[24] Adelantado, F., Verikoukis, C. 2011. A non-parametric statistical approach for malicious users detection in cognitive wireless ad-hoc networks. 2011 IEEE international conference on communications (ICC) s. 1-5. IEEE.

[25] Min, A. W., Shin, K. G., Hu, X. 2009. Attack-tolerant distributed sensing for dynamic spectrum access networks. 17th IEEE International Conference on Network Protocols, s. 294-303. IEEE.

[26] Li, S., Zhu, H., Yang, B., Chen, C., Guan, X. 2011. Believe yourself: A user-centric misbehavior detection scheme for secure collaborative spectrum sensing. 2011 IEEE International Conference on Communications (ICC), s. 1-5. IEEE.

[27] Mukherjee, S., Neelam, S. 2012. Intrusion Detection using Naive Bayes Classifier with Feature Reduction, Procedia Technology, Cilt. 4, s. 119-128. DOI: 1 0.1016/j.protcy.2012.05.017

[28] Hodo, E., Bellekens, X., Hamilton, A., Dubouilh, P.L. 2016. Threat analysis of IoT networks using artificial neural network intrusion detection system. 2016 International Symposium on Networks, Computers and Communications (ISNCC), 11-13 Mayıs, Yasmin Hammamet, 1-6.

[29] Jim, L.E., Chacko, J. 2019. Decision Tree based AIS strategy for Intrusion Detection in MANET. 2019 IEEE Region 10 Conference (TENCON), 17-20 Ekim, Kochi, 1191-1195.

[30] Yıldırım, P., Birant, D. 2018. The Relative Performance of Deep Learning and Ensemble Learning for Textile Object Classification. 2018 3rd International Conference on Computer Science and Engineering (UBMK), 20-23 Eylül, Saraybosna, 22-26.

[31] Tama, B.A., Rhee, K. 2016. Classifier Ensemble Design with Rotation Forest to Enhance Attack Detection of IDS in Wireless Network. 2016 11th Asia Joint Conference on Information Security (AsiaJCIS), 4-5 Ağustos, Fukuoka, 87-91.

[32] Hu, W., Hu, W., Maybank, S. 2008. AdaBoost-Based Algorithm for Network Intrusion Detection, IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), Cilt. 38, s. 577-583. DOI: 10.1109/TSMCB.2007.914695

[33] Chebrolu, S., Abraham, A., Thomas, J.P. 2005. Feature deduction and ensemble design of intrusion detection systems, Computers & Security, Cilt. 24, s. 295-307. DOI: 10.1016/j.cose.2004.09.008

[34] Cabrera, J.B.D., Guiterrez, C., Mehra, R.K. 2008. Ensemble methods for anomaly detection and distributed intrusion detection in Mobile Ad-Hoc Networks, Information Fusion, Cilt. 9, s. 96-119. DOI: 10.1016/j.inffus.2007.03.001

[35] Ma, T., Wang, F., Cheng, J., Yu, Y., Chen, X. 2016. A Hybrid Spectral Clustering and Deep Neural Network Ensemble Algorithm for Intrusion Detection in Sensor Networks, Sensors, Cilt. 16, s. 1-23. DOI: 10.3390/s16101701

[36] Yildirim, P., Birant, K.U., Radevski, V., Kut, A., Birant, D. 2018. Comparative analysis of ensemble learning methods for signal classification. 26th Signal Processing and Communications Applications Conference (SIU), 2-5 Mayıs, İzmir, 1-4.

[37] Yu, L., Shouyang, W., Lai, K.K. 2008. Credit risk assessment with a multistage neural network ensemble learning approach, Expert Systems with Applications, Cilt. 34, s. 1434–1444. DOI:10.1016/j.eswa.2007.01.009

[38] Yu, H., Ni, J. 2014. An Improved Ensemble Learning Method for Classifying High-Dimensional and Imbalanced Biomedicine Data, IEEE/ACM Trans Comput Biol Bioinform, Cilt. 11, s. 657-666. DOI: 10.1109/TCBB.2014.2306838

[39] Wang, G., Hao, J, Ma, J., Jiang, H. 2011. A comparative assessment of ensemble learning for credit scoring, Expert Systems with Applications, Cilt. 38, s. 223-230. DOI: 10.1016/j.eswa.2010.06.048

[40] Nikam, S. S. 2015. A comparative study of classification techniques in data mining algorithms. Oriental journal of computer science & technology, 8(1), 13-19.

[41] Pechenizkiy, M. 2005. The impact of feature extraction on the performance of a classifier: kNN, Naïve Bayes and C4. 5. In Conference of the Canadian Society for Computational Studies of Intelligence (pp. 268-279). Springer, Berlin, Heidelberg.
[42] Kumar, R., & Verma, R. (2012). Classification algorithms for data mining: A survey. International Journal of Innovations in Engineering and Technology (IJIET), 1(2), 7-14.
[43] Weka. https://www.cs.waikato.ac.nz/ml/weka/ (Erişim Tarihi: 08.03.2020).