Bilgisayar Bilimleri ve Mühendisliği Dergisi

Cilt-17
Sayı-1
2024

ADS-B Cihazlarına Yapılan Saldırıların Yapay Öğrenme ile Sınıflandırılması

Classification of Attacks on ADS-B Devices with Artificial Learning

İbrahim MERAL
Milli Savunma Üniversitesi
Atatürk Stratejik Araştırmalar ve Lisansüstü Eğitim Enstitüsü
İstanbul, Türkiye
ibrahim.meral.92@hotmail.com
ORCID: 0000-0002-5053-8613

Elif BOZKAYA
Milli Savunma Üniversitesi, Deniz Harp Okulu
Bilgisayar Mühendisliği Bölümü
İstanbul, Türkiye
ebozkaya@dho.edu.tr
ORCID: 0000-0001-6960-2585

Öz

Havayolu taşımacılığı, başlangıcından itibaren hava araçlarının takibi, uçuşun emniyeti ve hava trafiğinin yönetimi için oldukça önemlidir. Hava taşıtlarının takibinde ise hava taşıtının konumumun bulunduğu alanı yöneten hava sahası işletmecileri, kurumları bulunmaktadır. Bu hava sahasını kontrol eden kurumlar hava taşıtlarını takip edebilmek için çeşitli sistemler kullanmaktadır. Bu sistemler bütününü hava trafik yönetim sistemleri oluşturmaktadır. Hava araçlarının algılanması için kullanılan birçok radar çeşidi bulunmaktadır. Bu radarların dışında hava taşıtlarının konumunu saptamak için Otomatik Bağımlı Gözetim Yayını (Automatic Dependent Surveillance-Broadcast (ADS-B)) cihazları kullanılmaktadır. ADS-B cihazları kurulumu ve maliyeti diğer radar sistemlerine göre daha ucuz olduğundan saldırganlar için daha çok tercih edilir. Bu çalışma kapsamında, ADS-B cihazlarının *verisine yapılan saldırıların sınıflandırılması için hava taşıtı simülasyon araçlarından elde edilen bir yayın üzerinden olası saldırılar incelenmiştir. Elde edilen özgün bir veri kümesi üzerinden olası saldırıların saptanması amacıyla bir sistem modeli önerilmiştir. Amaca uygun olarak, veri kümesinde uygulanan ön işlemler sonrasında, farklı yapay öğrenme teknikleri ile değerlendirmeler yapılmıştır. Bu teknikler, Destek Vektör Makineleri (SVM), İkili Karar Ağacı ve Naive Bayes sınıflandırıcısı makine öğrenme tekniklerini içermektedir. Yapılan sınamalar, doğruluk, tutturma, duyarlılık ve F-ölçüsü ile değerlendirilmiştir.*

Anahtar sözcükler: ADS-B, Siber Güvenlik, Yapay Öğrenme, Siber Saldırı, Hava Trafik Kontrol Yönetimi

Gönderme, düzeltme ve kabul tarihi: 08.06.2023 - 17.10.2023 – 05.12.2023
Makale türü: Araştırma

Abstract

From the beginning of air transport, the tracking of aircraft is crucial for the safety of the flight and the management of air traffic. In the follow-up of aircraft, there are airspace operators and institutions that manage the area where the aircraft is located. Institutions controlling this airspace use various systems to track aircraft. All of these systems constitute Air Traffic Management Systems. There are many types of radars used to detect aircraft. Apart from these radars, Automatic Dependent Surveillance-Broadcast (ADS-B) devices are used to detect the position of aircraft. ADS-B devices are preferred by attackers as they are cheaper to install and cost than other radar systems.

Within the scope of this study, possible attacks were examined through a publication obtained from aircraft simulation tools for the classification of attacks on ADS-B data. A system model has been proposed to detect the possible attacks on a obtained data set. Specifically, after the preprocessing applied on a dataset, the evaluations are performed with different artificial learning techniques. These techniques include Support Vector Machine, Decision Tree and Naive Bayes classifier machine learning techniques. The tests are evaluated with accuracy, precision, sensitivity and F-criteria.

Keywords: ADS-B, Cyber Security, Artifical Learning, Cyber Attacks, Air Traffic Control Management

1. Giriş

Sivil ve askeri havacılığın hızla büyümesi ve hava araçlarının sahip olduğu sistemlerin emniyet kritik sistemler olması nedeniyle havacılıkta kullanılan uygulamaların doğruluğu büyük öneme sahiptir. Hava trafiğinin ve hava sahalarının

yönetimi için kullanılan sistemler bütününe hava trafik yönetim sistemleri denmektedir.

Hava trafik yönetim sisteminde uçakların konumlarını saptamak için birçok sistem kullanılmaktadır; bunlar yer bazlı radarlar veya uydu bazlı radarlardır. Uçakların konum bilgisi, havadaki uçağın yönetimi, hareketleri ve komutları belirlenip uçağa gönderilmektedir. Uçağın konum bilgisi sürekli olarak hava trafiğini yöneten kontrolöre iletilmekte ve kontrolörün kullandığı sistemlerde gösterilmektedir. Uçak konumlarını gösteren radarlar ve Otomatik Bağımlı Gözetim Yayını (Automatic Dependent Surveillance-Broadcast (ADS-B)) cihazlar bu gösterim için kullanılmaktadır. Radarlar, elektromanyetik dalga göndererek bunun karşı nesnelerden yansımalarını gözlemleyip çözümlerler. ADS-B cihazları ise, Küresel Navigasyon Uydu Sistemi (Global Navigation Satellite System (GNSS)) ve Küresel Yer Belirleme Sistemi (Global Positioning System (GPS)) gibi uydulardan aldıkları kendi pozisyonlarının bilgisini ADS-B verisi ile yayınlarlar. Bu sayede konum verisinin doğruluğu, yatay ve dikey uzaklık veya hava durumlarından etkilenmez. ADS-B ayrıca diğer konum bilgisi veren radarlara göre çok daha az maliyetlidir.

ADS-B sistemleri birçok üstünlüğüne karşın temel güvenlik mekanizmaları (örneğin, şifreleme, veri bütünlüğü) açısından saldırılara karşı savunmasızdır. Bunun yanı sıra 1090 MHz bant aralığında açık mesaj kullanılması da dışarıdan açık kaynak yazılımları ile saldırıya açık olmasına sebep olmaktadır. Bu saldırılar dinleme, yayın bozma, mesaj değiştirme, bozma, silme vb. şekilde olabilmektedir. Bu çalışma kapsamında ADS-B cihazına karşı yapılan saldırılardan sahte veri saldırısı ele alınmıştır.

Sahte veri saldırısı; sisteme zarar vermek veya yetkisiz erişim elde etmek amacıyla yanlış veya yanıltıcı verilerin girilmesini içeren bir saldırı türüdür. Bu saldırı veri enjeksiyon saldırısı olarak da adlandırılır. Veriler kasıtlı olarak yanıltıcı olabilir veya tamamen uydurma olabilir ve fiziksel veya sanal yollarla eklenebilir.

Bu çalışma kapsamında, ADS-B verilerinin bulunduğu özgün bir veri kümesi elde edilmiş ve sahte veri saldırılarına karşı yapay öğrenme teknikleri kullanılarak ADS-B cihazından gelen veri üzerinde anomali tespiti yapmak amacıyla bir model önerilmiştir. ADS-B cihazına sahte veri saldırısı yapılmış veri kümesi üzerinde yapay öğrenme tekniklerinden SVM, Naive Bayes ve İkili Karar Ağacı uygulanmıştır.

Bu makale kapsamında ikinci bölümde problemin çözümü için daha önceden yapılmış olan çalışmalar tanıtılmakta, üçüncü bölümde önerilen yapay öğrenme teknik temelli sistem modeli anlatılmakta, dördüncü bölümde elde edilen sonuçlar değerlendirmekte, beşinci bölümde ise çalışma özetlenerek sonuçlar yorumlanmaktadır.

2. Kaynak Taraması

Yeni nesil hava trafik gözetiminde ADS-B cihazları, durumsal farkındalığı sağlamak için gözetim yöntemlerinden biri olmuştur [1]. Ancak ADS-B cihazları, özellikle veri bütünlüğü ve kimlik doğrulamada saldırılara açıktır. Bu kapsamda, saldırılara karşı farklı çözüm önerileri sunulmuştur. Başka bir radar ile doğrulanarak verinin iletimi, ADS-B cihazından gelen verideki açık mesajın şifrelenmesi çözüm yöntemleri arasında verilebilir. Bu çözümlerden biri de yapay öğrenme teknikleri ile akıllı karar verme mekanizmalarının oluşturulmasıdır.

Çalışma [2]'de ADS-B cihazlarına karşı saldırıların saptama zaman gecikmesini azaltmak ve doğruluğu artırmak için hiyerarşik zamansal belleğe dayalı çevrimiçi bir saldırı saptama stratejisi önerilmiştir. Bunun için; ADS-B verilerinin özelliklerinin analizine dayanarak, ADS-B verilerinin normal dağılımlarını tanımlayan modeller oluşturulmuş ve hiyerarşik zamansal bellek yöntemi ve çevrimiçi öğrenme uygulanmıştır. Asıl ve öngörülen veriler arasındaki sapmaları karşılaştırarak, değerler ve korelasyonlar üzerindeki farklar elde edilmiştir. Farkları ayırt etmek için de dinamik eşik değerleri belirlenmiştir. Böylece hem doğruluk oranı artırılmış hem de işlem gecikmesi azaltılmıştır.

Çalışma [3]'te uçuş planı doğrulama, tek düğüm veri tespiti ve grup veri tespiti dahil olmak üzere çeşitli tespit yöntemlerini entegre eden uçuş ve yer istasyonu yeteneklerine göre ADS-B verileri üzerinde saldırı saptama yöntemi tasarlanmıştır. Pozitif saptama oranını iyileştirmek için, yerden yere, yerden havaya ve havadan havaya işbirlikçi saptama mekanizmaları önerilerek, her bir düğümün algılama yeteneğini geliştirmek amaçlanmıştır. Gerçek ADS-B verileri üzerinde analizler yapılarak, modelin etkinliği sınanmış ve saldırı saptamasında doğruluğun arttığı belirtilmiştir.

Çalışma [4]'te ADS-B protokolünde kimlik doğrulama problemini adreslemek için sertifikasız açık anahtar şifreleme yönetimini kullanan bir hiyerarşik kimlik doğrulama protokolü önerilmiştir. ADS-B mesajlarının doğrulanması için gereken sürenin literatürdeki geleneksel yaklaşımlara göre azaldığı ve önerilen yöntemin daha hızlı olduğu vurgulanmıştır.

Çalışma [5]'te ADS-B veri saldırılarını saptamak için, ADS-B verilerinin zamansal korelasyonlarını ve dağılım karakteristiklerini dikkate alan bir anormallik saptama modeli önerilmiştir. İlk olarak, değişken bir otomatik kodlayıcı (Variational AutoEncoder), ADS-B verilerini yeniden yapılandırmak için kullanılır. Sonrasında, yeniden oluşturulmuş değerler ile gerçek değerler arasındaki fark değerleri, eğitim için Destek Vektörü Veri Açıklamasına (Support Vector Data Description) ve ADS-B anomalisini algılayabilen bir sınıflandırıcıya gönderilir. Uygun yapılandırma değerleri ile yanlış pozitif oranı ile yanlış negatif oranının azaltıldığı belirtilmiştir.

Çalışma [6]'da veri yapısına bakılarak yazılım tabanlı radyo yardımıyla ADS-B cihazına karşı yapılan saldırıların yapay sinir ağları kullanılarak tespiti amaçlanmıştır. Sonrasında görsel radar ve havacılık veri tabanlarını kullanarak görsel resim oluşturulmuştur. Yapay sinir ağları algoritmaları kullanılarak saldırı yapılmış veriler tespit edilmiştir.

Çalışma [7]'de ADS-B sinyalini doğrulamak ve hava aracının konumunu iyileştirmek için, uçağın dinamik uçuş modelini ve uçuş bilgi sistemini kullanan ADS-B/MLAT (Multilateration) veri füzyon gözetim çerçevesi önerilmiştir. Böylelikle, ADS-B'den gelen verilerin kullanılırken doğruluğunu artırmak için MLAT radarı ile kaynaştırılmış ve ADS-B verisinin doğruluğunun arttırıldığı belirtilmiştir.

Ayrıca, ADS-B verisinden gelen anormal verilerin sınıflandırılması için yapılan çalışmalarda SVM [8], K-En Yakın Komşu Algoritması ve Naive Bayes sınıflandırıcı yapay öğrenme teknikleri [9] kullanılarak karşılaştırılmalar yapılmıştır. Başka bir çalışmada ise çoklu etiket sınıflandırılması yapılması amacıyla SVM, Karar Ağacı ve Rassal Orman sınıflandırıcı makine öğrenme teknikleri kullanılarak karşılaştırılmıştır [10]. Kullanılan saldırı tipleri tekrar saldırısı, hayalet uçak enjeksiyonu ve çoklu hayalet uçak enjeksiyonu olarak değerlendirilmiştir. Bu çalışmada en iyi sonucun Rassal Orman yapay öğrenme tekniğinde elde edildiği gözlemlenmiştir. Çalışma [11]'de ise uzun kısa süreli bellek kodlayıcı-kod çözücü mimarisine ve destek vektör etki alanı açıklamasına dayalı bir anomali veri tespit modeli önerilmiştir. Yinelemeli sinir ağı mimarisi ile anomalilerin saptanması amaçlanmış ve doğruluk oranında artış olduğu belirtilmiştir.

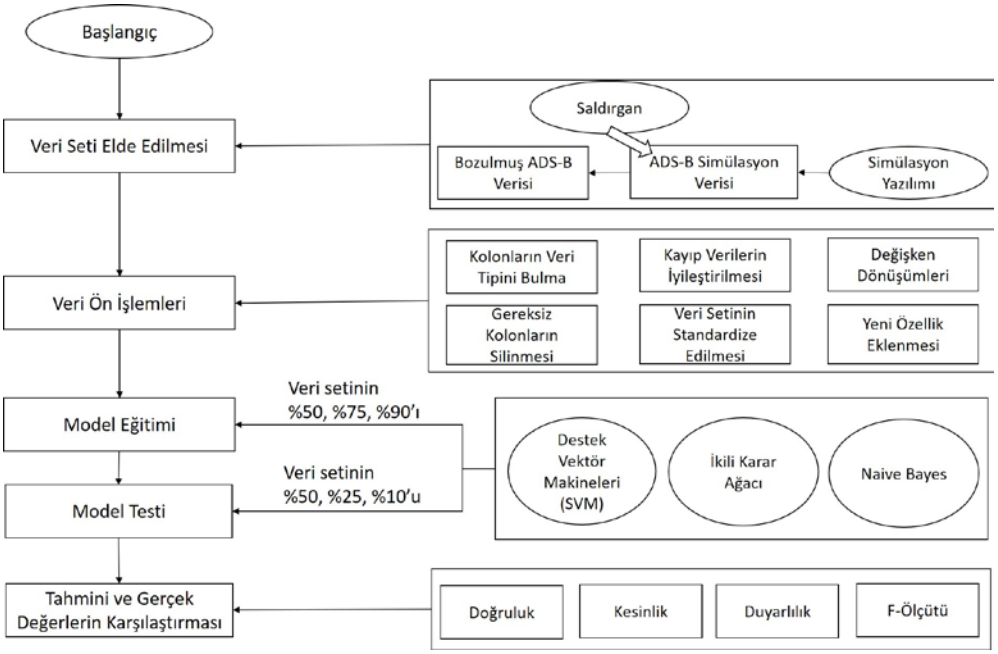
Başka bir çalışmada genel olarak mikro servis mimarisi altında öbek zinciri teknolojisi kullanılarak ADS-B verisinin güvenliği amaçlanmıştır [12]. Her servis bir konteynır üzerinde çalıştırılarak güvenilirlikleri kontrol edilmiştir. Bunu gerçekleştirmek için; üçüncü taraf alıcılardan gelen veri beslemelerinin hava trafik kontrol yer istasyonu alıcısı ile işlenmesi ve ilişkilendirilmesi sağlanmıştır. Önerilen model ile, mesaj sahtekarlığı ve anormal veri trafiği sorunlarının saptanabileceği ve dolayısıyla üçüncü taraf desteği sayesinde hava trafik kontrol altyapısı maliyetinin azaltılabileceği vurgulanmıştır.

Çalışma [13]'te; kimlik sahtekarlığı tespiti için önerilen mevcut derin öğrenme algoritmalarının, kötü amaçlarla oluşturulmuş ADS-B verilerine karşı hassas olduğu gösterilmiştir. Sahte mesajları tespit edilmeden ADS-B sistemine enjekte etmek için, kanal gürültüsünü bastırma ihtiyacını dengelemek ve kod çözme hatasını düşük tutmak gerektiği belirtilmiştir. Simülasyon sonuçları, kod çözme hatasını artırmadan derin öğrenme tabanlı saldırı tespitinden kaçınma yaklaşımının uygulanabilirliğini göstermiştir.

Özet olarak, kaynaklardaki çalışmalardan farklı olarak bu çalışmada, ADS-B cihazına karşı olası saldırılardan sahte veri saldırısı ele alınmış ve anomaliyi saptama amacıyla farklı yapay öğrenme algoritmalarının sınandığı ve birbiriyle karşılaştırıldığı bir sistem modeli önerilmiştir. Önemli işlem adımlarından biri olarak, sistem modelinin değerlendirilmesi için ADS-B verilerinin bulunduğu özgün bir veri kümesi oluşturulmuştur.

3. Araçlar ve Yöntem

Bu bölümde ADS-B cihazlarına yapılan saldırılara karşı önerilen sistem modeli sunulmuştur. Bu çalışma kapsamında sunulan sistem modelinde; veri kümesinin elde edilmesi, veri kümesine uygulanan ön işlemler ve sonrasında kullanılan yapay öğrenme tekniklerindeki modelin eğitimi ve modelin sınama aşamaları bulunmaktadır. Anomali saptaması için önerilen modelimiz Şekil 1'de verilmiştir.



Şekil-1: ADS-B Cihazlarına Yönelik Anomali Saptaması için Önerilen Sistem Modeli

3.1 Veri Kümesinin Elde Edilmesi

Çalışmada kullanılacak veri kümesi için ilk olarak ADS-B verisi elde edilmiştir. Hava aracı simülasyon yazılımından elde edilen bu orijinal veriye daha sonra sahte veri saldırısı yapılarak anomali verisi elde edilmiştir. Detaylar aşağıda açıklanmıştır.

3.1.1 ADS-B Veri Kümesinin Elde Edilmesi

Çalışmada kullanılan veri kümesinin elde edilmesi için ATCTRSIM hava trafik kontrol yazılımı simülasyon ortamı kullanılmıştır. Simülasyon ortamını kullanırken belirli bir alan ve bölge seçilip buradan örnekleme yapılması amaçlanmıştır. Türkiye hava sahasındaki bir konumu merkez alarak bir radar (ADS-B) varsayımı ve bu radarın dinlediği hava araçlarının

sabit hızda ilerlediği bir ortam sunularak çalışma yapılmıştır. Bu simülasyon ortamı 64 adet hava aracı kullanılarak 39 dakika 2 saniye boyunca çalıştırılmıştır. Çalıştırılan simülasyon ortamında uçaklar "enroute" (havada düz bir şekilde ilerleme) safhasında bulunmaktadırlar. Yaklaşık 6 saniyede bir her uçaktan bir veri alınmıştır. Simülasyon yazılımından elde edilen simülasyon nesneleri ATCSIMTEST yazılımı aracılığı ile saldırı yapılmış CSV formatında ADS-B veri kümesine dönüştürülmüştür. Elde edilmiş olan ADS-B veri kümesinde 23194 kayıt bulunmaktadır. Bu kayıtların 4406 tanesi bozulmuş veri (anomali), 18788 tanesi ise bozulmamış/saldırı yapılmamış veri (normal) olarak etiketlenmiştir. Bozulmuş verilerin (anomali) kullanılan toplam veri kümesine oranının %18,9 olduğu gözlemlenmiştir. Kullanılan veri kümesinde veri oluşturma zamanı olan TimeStamp, uçağın tekil olmasını sağlayan Callsign, uçak ve kontrolörün haberleşmesi için kullanılan Mode-S, hava trafiğinde kullanılan SSR Code, Enlem, Boylam, Uçuş Seviyesi, hız ve saldırı yapılmış etiketi isSpoofed bulunmaktadır.

ADS-B verilerinde uçuş trafiğine ait bir kayıttaki bilgilerin detayları aşağıda verilmiştir:

- TimeStamp: Radardan gelen verinin anlık zamanı milisaniye cinsindendir.
- Callsign: İlgili trafiğin telsiz çağrı kodudur. Uçuşa verilen tekil bir isimdir.
- Mode-S: Uçakların 24 bit genişliğinde bireysel seçici (S= Selective) adresleme ile kodlanmasına dayanır. Her hava aracına bireysel bir kod ataması yapılır [14].
- SSR (Special Service Request) Code: Bir uçuşa ait 4 haneli 8'lik tabanda verilen koddur. Uçak tarafından değiştirilebilir. Acil kod numaraları bulunmaktadır.
- Latitude: Uçağın anlık enlemi.
- Longtitude: Uçağın anlık boylamı.
- Speed: Uçağın anlık hızı.
- Flight Level: Uçağın anlık uçuş seviyesi, deniz seviyesinden yüksekliği.

3.1.2. Saldırı Yapılmış ADS-B Veri Kümesinin Elde Edilmesi

Bu çalışmada, ADS-B mesajında bulunan SSR kod ve hız bilgileri düzenlenmiştir. Elde edilen veri kümesinde 'spoofed' etiketi ile verinin bozulup bozulmadığı bilgisi de eklenmiştir.

Oluşturulan ADS-B verisinden saldırı yapılmış veri kümesi elde etmek için saldırıların konfigürasyonu da ATCSIMTEST yazılımı içerisinde bulunan konfigürasyon dosyasından alınmaktadır. Bu dosya *attackMode* ve *attackLevel* değerleri ile ayarlanmaktadır. Eğer *attackMode* 1 ise yalnızca uçuş yüksekliğinin bozulmasını, *attackMode* 2 ise yalnızca uçuş SSR kod değerinin yanlış gönderilmesini, eğer *attackMode* 3 ise hem uçuş yüksekliği hem de SSR kod değerlerinin bozulmasını sağlayacaktır. Konfigürasyonda bulunan *attackLevel* değeri ile 100 değeri çarpılıp uçuşun yüksekliğinin feet cinsinden değeri eklenmiştir.

Bu çalışmada *attackMode* değeri 3 olarak ayarlanarak her iki saldırı da modellenmiştir. *attackLevel* değeri ise 4 olarak ayarlanmıştır. Bu da her bozulmuş olan trafiğe 400 feet eklenmesine sebep olmuştur.

Çalışma kapsamında incelenen ve yapılan saldırılar aşağıda verilmiştir:

- *Sahte SSR Kod Saldırısı*: SSR Kod her trafiğin kendisine ait bir tekil koddur. Bu nedenle, burada yapılacak olan değişiklik hem hava trafiğini yöneten tarafta hem de uçak tarafında problemlere sebep olacaktır. Hava trafiğinin yönetiminde kontrolör tarafında farklı bir SSR kod ile trafik verisinin gelmesi, kontrolörün önünde bulunan yardımcı araçlarda uçuş planları ile ilgili izin korelasyonun kırılmasına sebep olacaktır. Ayrıca yeni geliştirilen veri bağlantısı araçları da bunlardan etkilenecektir. SSR kod, 8'lik tabanda değerler ile verilmektedir. Ancak 7 ile başlayan SSR kod değerleri alarm anlamına gelmektedir. Bunlar;

- 7500 – uçak kaçırma,
- 7600 – iletişim problemi ve
- 7700 – acil durumdur.

Bu çalışmada, bazı SSR kod değerlerini 7600 ile değiştirip yayınlayarak saldırılar analiz edilmiştir. Hava trafiğini yöneten tarafa yanlış bir ikaz gitmesine sebep olan bu hata kontrolörün ve karar destek yazılımlarının yanlış karar vermelerine sebep olacaktır.

Sahte Uçuş Seviyesi Saldırısı: Hava taşıtının ADS-B mesajında bulunan anlık uçuş yüksekliğinin bozulması trafiğin anlık uçuş seviyesine bağlı problemleri beraberinde getirecektir. Hava trafiği yönetiminde bulunan kontrolör ile hava taşıtı arasındaki iletişimde sürekli yanlış bilgiye dayalı olarak iletişim gerçekleşecektir. Bu saldırı, hava aracının kaza kırımına dahi sebep olabilir.

Veri kümesi oluşturulduktan sonra ilk olarak aşağıda belirtilen ön işlemler gerçekleştirilmiştir.

3.1.3. Veri Kümesine Uygulanan Ön İşlemler

Şekil-1'de de verilen sistem modelinde veri kümesi üzerinde uygulanan ön işlemler aşağıda sunulmuştur.

Yeni Özellik Ekleme: Callsign bilgisi uçağın tekil anahtar değeridir. Burada kullanılan değer ise karakter tipindedir. Bu nedenle sistem tarafından kullanılabilmesi için bu kolon nümerik değerlere dönüştürülmüştür ve 'CallSignNumeric' adında bir kolon oluşturulmuştur.

Kolonların Veri Tipini Bulma: Bu analiz işleminde veri kümesinde bulunan tüm kayıtların durumu gösterilmektedir. Bizim kullandığımız bozulmamış veriye ait bilgiler, boyut ve tipleri ile birlikte her bir kolonun bilgileri aşağıda Şekil 2'de gösterilmiştir. Şekilde görüldüğü gibi 9 numaralı 'isSpoofed' etiketi ikili değişken olarak tanımlanmış ve modelin çıktısını anomali yok ya da anomali var şeklinde (0 ve 1) değerlendirmektedir. Diğer sütun bilgileriyse Bölüm 3.1.1'de ayrıntılı biçimde verilmiştir.

#	Column	Non-Null Count	Dtype
0	TimeStamp	23194 non-null	object
1	Callsign	23194 non-null	object
2	CallSignNumeric	23194 non-null	int64
3	Mode-S	23194 non-null	object
4	SSR Code	23194 non-null	int64
5	latitude	23194 non-null	float64
6	Longtitude	23194 non-null	float64
7	Flight Level	23082 non-null	float64
8	Speed	23194 non-null	float64
9	isSpoofed	23194 non-null	bool

Şekil-2: Orjinal Veri Kümesi Özellikleri

Gereksiz Kolonların Silinmesi: Burada kullanılacak modelde modele etkisi olmayan kolonlar silinmektedir. Bu sayede işlemin performansı artırılıp kullanılacak verinin de alanı azaltılmış olur. Bu çalışmada trafiğin 'latitude', 'longtitude', 'TimeStamp', 'Callsign', 'Mode-S' kolonları değerlendirmeyi etkilemediği için silinmiştir.

Eksik Verilerin İşlenmesi: Kullanılacak olan kayıtlardaki verilerin bazıları boş kalmış olabilir. Bunlara hesaplamayı bozmaması için bir değer atanabilir. Bu çalışmada ortalama değer atanmıştır. 'Flight Level' kolonunun eksik bulunan kayıtlarına 'Flight Level' kolonunun ortalama değeri atanmıştır.

Veri Kümesinin Standardize Edilmesi: Bu aşama ile birlikte kullanılan değerler standardize edilmiş olacaktır. StandardScaler ortalamayı kaldırır ve her özelliği/değişkeni birim varyansa göre ölçeklendirir. Bu işlem, özellik bazında bağımsız bir şekilde gerçekleştirilir. StandardScaler, ampirik ortalamanın tahminini ve her özelliğin standart sapmasını içerdiğinden (veri kümesinde varsa) aykırı değerlerden etkilenebilir [15].

Bu çalışmada veri kümesi standardize edilmeden önce öğrenme ve doğrulama veri kümeleri olarak ayrılmıştır. 'isSpoofed' etiketi olan kolon üzerinden çalışılacağı da belirtilmiştir.

3.2. Model Eğitimi ve Model Sınaması

Bu kısımda modelin eğitilmesi ve sınamasında kullanılan yapay öğrenme teknikleri ve bu teknikler için kullanılan değişkenlerin değerleri tartışılmıştır.

3.2.1. Kullanılan Yapay Öğrenme Teknikleri

Çalışmada gözetimli öğrenme teknikleri kullanılmıştır. Gözetimli yapay öğrenme, makine öğrenmesinin bir alt dalıdır ve veri analizinde sıkça kullanılan bir tekniktir. Bu yaklaşım, modelin giriş verileri ile bu verilere karşılık gelen hedef çıktıları arasındaki ilişkiyi öğrenmeyi amaçlar. Temel olarak, bir öğrenme algoritması, verileri inceleyerek girişleri hedef çıktılara eşlemeye çalışır. Bu aşamada, algoritma veri noktalarındaki desenleri ve ilişkileri yakalamayı hedefler, böylece daha sonra verilen yeni giriş verileri üzerinde tahminlerde bulunabilir.

Gözetimli öğrenme aşaması aşağıdaki adımlarla gerçekleşir:

1. *Veri Toplama ve Hazırlama*: İlk adım veri toplamak ve hazırlamaktır. Bu veriler, giriş özelliklerini (bağımsız değişkenler) ve hedef çıktılarını (bağımlı değişkenler) içerir. Örnek bir veri kümesi, özellikleri ve bunlara karşılık gelen hedef çıktıları içerir.
2. *Model Seçimi*: Veri toplandıktan sonra, kullanılacak model türü seçilir. Bu modeller genellikle matematiksel ve istatistiksel fonksiyonlardan oluşur. Örnek olarak, bizim de bu makalede kullandığımız Naive Bayes, ikili karar ağaçları, destek vektör makineleri (SVM) gibi modeller kullanılabilir.
3. *Model Eğitimi*: Seçilen model, veri kümesi üzerinde eğitilir. Bu, modelin giriş verilerini hedef çıktılara eşlemesini sağlayan parametreleri ayarlamak anlamına gelir. Eğitim süreci, veri kümesinin üzerinden geçerek modelin hedefi tahmin etmesi ve gerçek hedefle karşılaştırmasıyla gerçekleşir. Model, tahminlerindeki hataları en aza indirecek şekilde güncellenir.
4. *Model Değerlendirmesi*: Eğitim tamamlandıktan sonra, modelin performansını değerlendirmek önemlidir. Bunun için ayrı bir sınama veri kümesi kullanılabilir. Model, bu veri kümesindeki girişleri kullanarak hedef çıktıları tahmin eder ve gerçek hedeflerle karşılaştırarak ne kadar doğru tahmin yaptığını ölçer.
5. *Tahminler ve Sonuç Çıkarma*: Eğitilmiş model, yeni ve görülmemiş giriş verileri üzerinde tahminlerde bulunabilir. Model, öğrendiği desenleri kullanarak bu tahminleri yapar.

Bu çalışma kapsamında kullanılan yapay öğrenme teknikleri aşağıda detaylı olarak açıklanmıştır:

İkili Karar Ağacı: İkili Karar Ağacı, verileri sınıflandırmak için kullanılan bir dizi kuralın grafiksel bir temsilidir. İkili Karar Ağacı, bir durdurma kriterine ulaşılana kadar belirli bir özelliğin değerine dayalı olarak veriyi yinelemeli olarak alt kümelere bölerek oluşturulur [16].

Naive Bayes Sınıflandırıcı: Naive Bayes, Bayes teoremine dayanan olasılıksal bir algoritmadır. Girdi özellikleri, verilen her sınıfın olasılığını hesaplar ve en yüksek olasılığa sahip sınıfı seçer [17].

Destek Vektör Makineleri (Support Vector Machine-SVM): Sınıflar arasındaki marjı maksimize ederek sınıflar arasındaki en iyi sınırı bulan bir sınıflandırma algoritmasıdır. Verileri, doğrusal bir sınırın bulunabileceği daha yüksek boyutlu bir alana dönüştürerek çalışır [18].

Bu bölümde kullanmış olduğumuz SVM, Naive Bayes ve İkili Karar Ağaçları tekniklerinin uygulama detaylarını ayrıntılı olarak anlatacağız. Tüm teknikler kullanılmadan önce veri yukarıda bahsedilen ön işlemlerden geçmiştir.

Analizi yapılmış ve sonrasında ön işlemlerden geçmiş olan veri kümesi öğrenme ve doğrulama veri kümeleri olarak bölünmüştür. Bu çalışmada 3 farklı türde bir ayırma yapılarak teknikler denenmiştir.

- Veri kümesinin %50'si modelin eğitimi, %50'si modelin sınaması
- Veri kümesinin %75'i modelin eğitimi, %25'i modelin sınaması

• Veri s kümesinin %90'ı modelin eğitimi, %10'u modelin sınaması

3.2.2. Destek Vektör Makineleri Tekniği Uygulaması

SVM, sınıflandırma ve regresyon analizi için yaygın olarak kullanılan bir makine öğrenme algoritmasıdır. Veri kümesindeki sınıfları maksimum düzeyde ayıran yüksek boyutlu bir uzayda hiper düzlemi bularak çalışır.

Basit bir ifadeyle, SVM, sınıflar arasındaki marjı en üst düzeye çıkaracak şekilde sınıfları ayıran en iyi çizgiyi veya sınırı bulmaya çalışır. Kenar boşluğu, hiper düzlem ile her sınıftan en yakın veri noktaları arasındaki mesafedir. SVM'nin amacı, daha sağlam ve genelleştirilebilir bir modelle sonuçlanan en büyük marja sahip hiper düzlemi bulmaktır [19].

SVM, orijinal veri kümesini bir çekirdek işlevi kullanarak daha yüksek boyutlu bir alana dönüştürerek çalışır. Bu; algoritmanın, orijinal özellik uzayında doğrusal olarak ayrılamaz durumda olsalar bile sınıfları ayıran bir hiper düzlem bulmasını sağlar. SVM, doğrusal, polinom ve radyal temel işlev çekirdekleri dahil olmak üzere çeşitli çekirdek işlevlerine sahiptir. Bu çekirdek tipleri verilerin koordinatlarını daha yüksek boyutlu bir uzayda hesaplamadan orijinal özellik uzayında işlem yapmamızı sağlar. Bu çalışmada SVM, doğrusal (Linear) ve radyal (RBF) temel işlevi olarak kullanılmıştır.

SVM tekniğinde C parametresinin önemli bir rolü bulunmaktadır. C parametresi, SVM'nin ihlaller konusunda ne kadar ciddi olduğunu belirler. C; 0 ise, ceza süresi ortadan kalktığı için SVM ihlalleri hiç umursamaz. C çok büyükse, küçük ihlaller amaç fonksiyonunda büyük bir artışa yol açacaktır [20].

SVM uygulanırken C değişkeni {0.1, 0.01, 0.001} değerlerini alarak Kernel tipi 'RBF' ve 'Linear' iken sınanacaktır. Bulgular bölümünde çizelgeler halinde kıyaslaması yapılacaktır.

3.2.3. Naive Bayes Sınıflandırıcı Tekniği Uygulaması

Naive Bayes, Bayes Teoremine dayalı istatistiksel bir sınıflandırma tekniği ve denetimli öğrenme algoritmalarından biridir [21]. Naive Bayes sınıflandırıcısı hızlı, doğru ve güvenilir bir algoritmadır. Naive Bayes sınıflandırıcıları, büyük veri kümelerinde yüksek doğruluk ve hıza sahiptir. "Saf" olarak adlandırılır, çünkü bir veri kümesinin özelliklerinin, gerçekte ilişkili olsalar bile, birbirinden bağımsız olduğu varsayımını yapar. Bu basitleştirici varsayıma rağmen, Naive Bayes birçok uygulamada etkili olabilir.

Naive Bayes sınıflandırmasında algoritma, özelliklerin değerlerine dayalı olarak her bir sınıfın olasılık dağılımını öğrenmek için eğitim verilerini kullanır. Ardından, bir dizi özellik değerine sahip yeni bir örnek verildiğinde, algoritma, öğrenilen dağılımlara dayanarak örneğin her bir sınıfa ait olma olasılığını hesaplar. En yüksek olasılığa sahip sınıf daha sonra örneğe atanır.

Bu çalışma kapsamında Gauss Naive Bayes, Kategorik Naive Bayes ve Multinomial Naive Bayes teknikleri uygulanmıştır.

Gaussian Naive Bayes, Multinomial Naive Bayes ve Kategorik Naive Bayes, Naive Bayes algoritmasının sınıflandırma problemlerinde yaygın olarak kullanılan üç çeşididir. Bu üç değişken arasındaki temel fark, işleyebilecekleri veri türünde ve varsaydıkları olasılık dağılımında yatmaktadır [22].

Gaussian Naive Bayes (GNB): Gaussian Naive Bayes, sürekli veriler için kullanılır ve özelliklerin bir Gauss dağılımını (yani normal dağılımı) takip ettiğini varsayar. Naive Bayes'in bu varyantı, özelliklerin kelime frekansları veya tf-idf puanları olduğu metin sınıflandırma problemlerinde sıklıkla kullanılır.

Multinomial Naive Bayes (MNB): Multinomial Naive Bayes, kelime sayıları veya belge frekansları gibi ayrık veriler için kullanılır. Özniteliklerin çok terimli bir dağılımdan üretildiğini varsayar. Naive Bayes'in bu varyantı, metin sınıflandırma problemlerinde yaygın olarak kullanılır [23].

Kategorik (Categorical) Naive Bayes (CNB): Categorical Naive Bayes, özelliklerin ikili olduğu (yani, 0 veya 1 gibi yalnızca iki değer alabildikleri) Multinomial Naive Bayes'in özel bir durumudur. Özelliklerin bir Bernoulli dağılımından üretildiğini varsayar. Naive Bayes'in bu biçimi genellikle spam filtrelemede, duyarlılık analizinde ve diğer ikili sınıflandırma problemlerinde kullanılır.

Çalışma esnasında GNB, MNB ve CNB algoritmaları verilen sınama oranları üzerinden modellenip çalıştırılmıştır. Yapay öğrenme tekniklerinin karşılaştırmalı çizelgesi ve sonuçları Bulgular bölümünde gösterilecektir.

3.2.4. İkili Karar Ağacı Tekniği Uygulaması

İkili Karar Ağacı algoritması, eğitim veri kümesinden oluşturulan karar ağacını kullanarak sınıflandırılacak verinin sınıfını belirler. İkili karar ağacı oluşturulurken öncelikle kök düğüm belirlenir. Kök düğümü belirlenirken örnekleri en iyi ayıran özellik seçilir. Sonrasında bu işlem yaprak düğümlerde tekrarlanarak ağacın yapısı belirlenir [24].

İkili Karar Ağacı, makine öğrenimi ve veri madenciliğinde kullanılan denetimli öğrenme algoritmalarındandır. Ağacın her bir iç düğümünün bir öznitelik üzerindeki sınamayı temsil ettiği, her dalın sınamanın sonucunu temsil ettiği ve her yaprak düğümün bir kararı veya sonucu temsil ettiği bir karar verme sürecinin grafiksel bir temsilidir.

Bu çalışmada ikili karar ağacı sınama oranları değişkenleri ile birlikte uygulanmıştır. Uygulanan İkili Karar Ağacı modellerinin sonuçları Bulgular bölümünde kıyaslanıp tartışılacaktır.

4. Bulgular

Çalışmanın bu bölümünde yapılan çalışmaların ve hesaplamaların değerlendirmesi tartışılacaktır.

İkili Karar Ağaçları, amacın niteliklerine dayalı olarak yeni bir sınıf etiketini tahmin etmek olduğu sınıflandırma görevleri için veya amacın sürekli bir hedef değişkeni tahmin etmek olduğu regresyon görevleri için kullanılabilir. İkili Karar Ağaçları, basitlikleri, yorumlanabilirlikleri ve çok çeşitli veri kümelerinde başarılı olmaları nedeniyle finans, sağlık ve mühendislik gibi çeşitli alanlarda yaygın olarak kullanılmaktadır.

Bu çalışma, 4 çekirdekli 10. nesil i7 Windows işletim sistemi, 32 GB bellek, 1TB hafıza bellek özellikli bir donanım üzerinde gerçekleştirilmiştir. Simülasyon ortamından verinin elde edilmesi için kullanılan ATCSIMTEST yazılımında Java 8 versiyonu kullanılıp IntelliJ geliştirme ortamında derlenmiştir. Elde edilen verilerle Python programlama dili ve PyCharm geliştirme ortamı kullanılarak, Scikit-learn (Sklearn) kütüphanesindeki yapay öğrenme teknikleri uygulanmıştır.

Bu çalışma kapsamında kullanılan yapay öğrenme teknikleri doğruluk, kesinlik, duyarlılık ve F-ölçüsü kıyaslanarak değerlendirilmiştir. Öncelikli olarak kullanılan yapay öğrenme tekniklerinin sonuçları değerlendirilecektir. Daha sonra en iyi sınama oranı sabit tutularak yapay öğrenme teknikleri birbirleriyle karşılaştırılacaktır.

SVM yapay öğrenme tekniğinde C değerinin Kernel bazında farklı değerler ile uygulanması sonucunda Çizelge 1 elde edilmiştir.

Çizelge-1'de görüldüğü gibi Kernel tiplerinde 'RBF' veya doğrusal (Linear) olması durumunda ve C parametresinin 0,1 değerine sahip olduğunda duyarlılık oranının en yüksek olduğu, doğruluk değerinin de en yüksek değerlere sahip olduğu görülmüştür.

Naive Bayes yapay öğrenme tekniğinde C değerinin 0,1 olarak çalıştırılması daha doğru sonuçlar verdiği için bu değer sabit tutularak Kernel bazında RBF ve Doğrusal değişkenleri farklı sınama oranları ile uygulanmıştır. Çizelge 2'de bu çalışmanın karşılaştırması gösterilmiştir. Çizelge 2'de gösterilen sonuçlarda sınama oranı her iki Kernel tipinde de en iyi sonuca 0,1 olduğunda ulaşılmıştır. 0,1 sınama oranında ise Kernel tipi RBF olması daha doğru sonuçlar elde edilmesini sağlamıştır.

Çizelge 3'te Gauss, Multinomial ve Kategorik Naive Bayes tekniklerinin sınama oranı ile beraber karşılaştırılmaları verilmiştir. Çizelgede GNB ve MNB tekniklerinin CNB tekniğine oranla daha iyi sonuçlar verdiği görülmektedir. GNB ve MNB değerlerinin ise 0,1 sınama oranında Doğruluk ve Tutturma sonuçlarında daha iyi değer elde ettiği gözlemlenmiştir.

Çizelge-1: C Değerinin SVM'de Kernel bazında karşılaştırılması

C parametresi	Kernel					
	RBF			Linear		
	Doğruluk	Tutturma	Duyarlılık	Doğruluk	Tutturma	Duyarlılık
0,1	0,907	0,898	1	0,895	0,899	0,982
0,01	0,891	0,898	0,872	0,895	0,899	0,982
0,001	0,894	0,878	0,875	0,895	0,899	0,982

Çizelge-2: Veri Kümesinde Sınama Verisinin Oranının Etkisi

Sınama Oranı	Kernel					
	RBF			Linear		
	Doğruluk	Tutturma	Duyarlılık	Doğruluk	Tutturma	Duyarlılık
0,1	0,907	0,898	1	0,895	0,899	0,862
0,25	0,891	0,898	0,872	0,891	0,898	0,872
0,5	0,894	0,897	0,875	0,894	0,897	0,875

Çizelge-3: Naive Bayes Algoritmaları Sınama Oranı Karşılaştırması

Tipi	Sınama Oranı								
	0,1			0,25			0,5		
	Doğruluk	Kesinlik	Duyarlılık	Doğruluk	Kesinlik	Duyarlılık	Doğruluk	Kesinlik	Duyarlılık
GNB	0,895	0,899	0,981	0,891	0,893	0,983	0,894	0,897	0,982
MNB	0,891	0,895	0,899	0,982	0,893	0,983	0,894	0,897	0,982
CNB	0,894	0,887	0,878	1	0.868	1	0,846	0,84	1

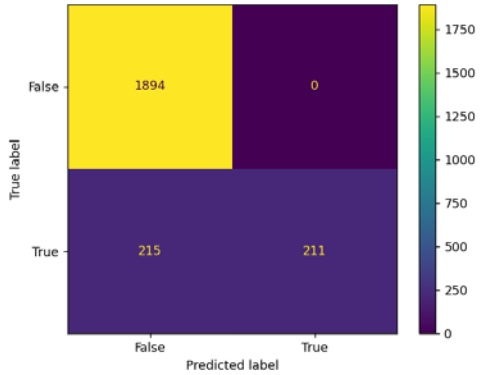
İkili Karar Ağacı yapay öğrenme tekniğinin sınama oranları bazında karşılaştırılması sonucunda Çizelge 4 elde edilmiştir. Çizelgede sınama oranı 0,1 olduğunda en iyi sonuçların elde edildiği gözlemlenmiştir.

Çizelge-4: İkili Karar Ağacı Sınama Oranı

Sınama Oranı	Karar Ağacı		
	Doğruluk	Tutturma	Duyarlılık
0,1	0,99828	0,9979	1
0,25	0,99776	0,99724	1
0,5	0,99828	0,99787	1

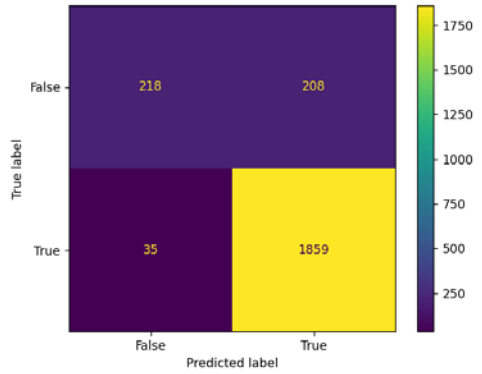
Özet olarak Çizelge 1'de görüldüğü üzere C değeri 0,1 olduğunda en iyi sonucu vermiştir. Bu nedenle de sınama oranları Kernel tipi üzerinden çalıştırılırken C değeri 0,1 sabiti kullanılarak hesaplanmıştır. Çizelge 2'de görüldüğü gibi sınama oranı azaltılıp veri kümesinin eğitilme oranı artırılınca doğruluk oranı artmıştır. Ancak yanlış pozitif hiç tespit edilememiş yanlış değerleri tamamen hatalı tespit edilmiştir. Burada görülen sınama oranı 0,1 ve Kernel tipi de RBF olduğunda en iyi sonucu verdiği görülmüştür. Karmaşıklık matrisi olarak tüm hesaplamaların sonuçları yerine en iyi modelin yani sınama oranı 0,1 olan, C değeri 0,1 olan ve RBF

Kernel tipinde yapılan modelin veri kümesi üzerindeki sonucu Şekil 3'de gösterilmiştir.



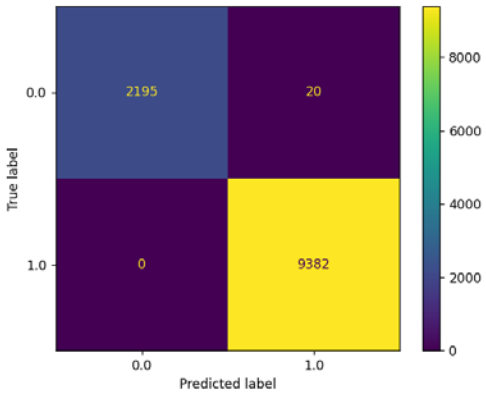
Şekil-3: SVM Karmaşıklık Matrisi

Çizelge 3'de gösterilen Naive Bayes yapay öğrenme teknikleri ve sınama oranı değerlendirmesinde görüldüğü üzere en iyi sonuçları Multinomial Naive Bayes tekniği ve sınama oranı 0,1 olan model vermiştir. Bu kapsamda, en iyi modelin karmaşıklık matrisi sonucu Şekil 4'te gösterilmiştir.



Şekil-4: Multinomial Naive Bayes Karmaşıklık Matrisi

Çizelge 4'te görüldüğü gibi İkili Karar Ağacı tekniği 0,1 oranında sınama veri kümesi kullanılınca daha doğru sonuçlar elde ettiği gözlemlenmiştir. Burada elde edilen karmaşıklık matrisi de Şekil 5'te gösterilmiştir.



Şekil-5: İkili Karar Ağacı Karmaşıklık Matrisi

Yukarıda SVM, Naive Bayes ve İkili Karar Ağacı yapay öğrenme tekniklerinin değerlendirmeleri bulunmaktadır. Tüm değerlendirmelerdeki sınama oranı ortak olarak 0,1 en iyi sonucu vermiştir. Bu nedenle, farklı sınıflandırıcılar arasında karşılaştırma, sınama oranı 0,1 olarak yapılacaktır. Bu karşılaştırma yapılırken en iyi sonuçları veren modellerin değerlendirmesinde; doğruluk, kesinlik, duyarlılık ve F-Ölçütü kullanılacaktır.

Bu kapsamda, yapay öğrenme tekniklerinin sonuçları farklı sınama oranları ile değerlendirilmiş ve en iyi karşılaştırma sonuçları Çizelge 5'de verilmiştir. Çizelge 5'te görüldüğü gibi kullanılan veri kümesi üzerinde sınama oranı 0,1 olduğunda İkili Karar Ağacı yapay öğrenme tekniğinin tüm parametrelerde üstünlük sağladığı görülmüştür. En düşük sonuç ise Naive Bayes yapay öğrenme tekniğinde elde edilmiştir. Naive Bayes daha çok özellik barındıran veri kümelerinde daha iyi tahmin yeteneğine sahiptir. Bu çalışmada SSR kodu ve uçuş seviyesi değerleri anormal verilere sahiptir. Uçuş seviyesi değeri her uçakta farklı değerlerde olduğu için buraya yapılan aynı seviyede bozulma SVM ve Naive Bayes için daha düşük doğruluk değerlerine neden olmuştur. Buna karşılık, ikili karar ağacında ise, SSR kodu ve uçuş seviyesi değerlerine bağlı olarak 'isSpoofed' kolonunda anomali var/anomali yok değerlendirmesi kuralların üretilmesi aşamasında güçlü bir yaklaşım sunmuştur. Karar ağacına dahil olan her bir girdinin ağacın yapısına katkısı anlamlandırılmış ve 0,998 oranında doğruluk elde edilmesi sağlanmıştır.

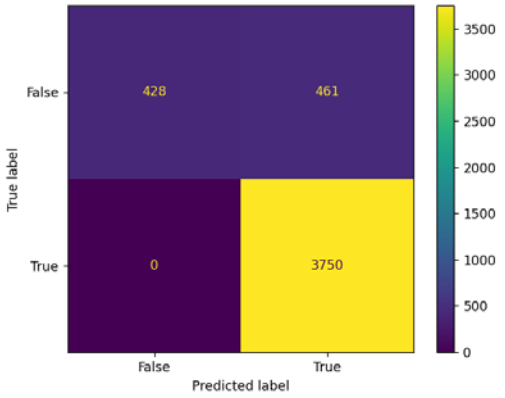
Çizelge-5: Farklı Yapay Öğrenme Tekniklerinin Sonuçlarının Karşılaştırılması

SVM				Naive Bayes				İkili Karar Ağacı			
Doğruluk	**Kesinlik**	**Duyarlılık**	**F-Değeri**	**Doğruluk**	**Kesinlik**	**Duyarlılık**	**F-Değeri**	**Doğruluk**	**Kesinlik**	**Duyarlılık**	**F-Değeri**
0,907	0,898	1	0,946	0,895	0,899	0,982	0,939	0,998	0,998	1	0,999

Elde edilen sonuçları literatürdeki bir çalışma ile karşılaştırmak için Wahlgren ve Thorn çalışması ele alınmıştır [8]. Wahlgren ve Thorn, bu çalışmasında ADS-B verisinden gelen verileri bir saldırı simülasyonu üzerinden değiştirerek bir veri kümesi elde etmiştir. Araştırmacıların saldırı simülasyonundan elde ettiği ADS-B veri kümesinde*; sahte rota, sahte bilgi* gibi saldırıların doğru yanlış etiketleri bulunmaktadır. Yapmış oldukları modelde %30 oranında sınama verisi, %70 oranında ise eğitim verisi kullanmışlardır. Kullandıkları SVM yapay öğrenme tekniğini de Python'da gerçekleştirmişlerdir. Çalışma sonucunda tüm etiketlerin doğru tahminleme ortalaması, SVM kullanarak %91,4 olarak gözlemlenmiştir. Bu çalışmada elde edilen sonuçlar saldırı bazında değerlendirilerek toplam sonuç elde edilmiştir. Elde etmiş olduğu değerlerde dikkat çekici olan %100 başarılı tahmin ile *yanlış rota* saldırısında gözlemlenmektedir. Diğer taraftan dikkat çeken kötü sonuç ise *sahte rota* etiketinde %61 gibi düşük doğrulukta bir tahmin gerçekleşmiştir. Buna

karşılık; bizim çalışmamızda önerilen sistem modelinden elde ettiğimiz sonuçta, İkili Karar Ağacı yapay öğrenme tekniği %10 sınama oranı, %90 eğitim oranı ile en iyi sonucu vermiş olup, doğruluk ölçütünde %99,8 başarı elde edilmiştir.

Bu çalışmamızı, Wahlgren ve Thorn [8]'un yaptığı çalışma ile kıyaslamak maksadıyla %30 sınama verisi, %70 eğitim verisi olacak şekilde değerlendirdik. Bu maksatla, İkili Karar Ağacı yapay öğrenme tekniğini kullanarak Şekil 6'da belirtilen Karmaşıklık Matrisi elde edilmiş ve doğruluk ölçütü %90 olarak hesaplanmıştır.



Şekil-6: 0,3 Sınama Oranı ile İkili Karar Ağacı için Karmaşıklık Matrisi

Wahlgren ve Thorn'un yapmış olduğu çalışma [8] ile bu çalışmada önerilen sistem modeli karşılaştırıldığında; yapılan saldırılar sonucunda elde edilen veriler ve etiketler farklılık göstermektedir. Saldırı sonucu oluşan etiketlemelerin kullanılan yapay öğrenme tekniğinin de doğruluk oranlarına ve seçimine etki ettiği tespit edilmiştir. SVM kullanarak çok daha fazla etiket kullanıldığında daha iyi sonuçlar elde edildiği görülmüştür. Ancak bazı etiketlerde (ham veride boş değerleri çok olan) çok düşük doğruluk skorlarına sebep olabilmektedir; sahte veri tahmininindeki %61'lik düşük oranda doğruluk skoru vermesi gibi. Bu çalışmada birden fazla saldırı yapılmış olsa da ortak bir şekilde değerlendirilerek tek kolonda bozulup bozulmadığı etiketlenmiştir. Bu nedenle İkili Karar Ağacı yapay öğrenme tekniği ile oluşturulmuş modelin ADS-B cihazına yapılan sahte veri saldırılarında daha iyi sonuçlar verdiği tespit edilmiştir.

5. Tartışma ve Sonuç

Havacılığın hızla büyümesiyle, yoğunlaşan hava trafiğinin yönetimi de çok büyük önem kazanmıştır. Hava trafiğinin yönetiminde kullanılan yardımcı yazılımlar hava taşıtının konumunu kontrolörlere yayınlamaktadır. Hava taşıtlarının konumları ile birlikte kontrolörler hava trafiğini yönetebilmektedir. Hava taşıtının konum bilgisi radar ve ADS-B cihazları üzerinden gelen veriler ile kontrolörün ekranına yansıtılmaktadır.

ADS-B cihazları radarlara oranla finansal maliyeti daha ucuz ve kurulum maliyeti de daha düşüktür. Bu nedenle ADS-B cihazının kendisi veya ADS-B cihazı üzerinden gelen hava trafiği dışarıdan gelebilecek saldırılara diğer radarlara göre daha açıktır. Bu saldırılar; hava trafiğini önemli ölçüde etkileyebilecek, kontrolörün yanlış davranmasına hatta uçak kaza kırımlarına sebep olabilecek saldırılardır. Bu makale kapsamında ADS-B üzerinden hava trafiğini yöneten ve kontrolöre gönderilen sahte veri saldırıları ele alınıp anomali tespiti gerçekleştirilmiştir.

Özetle, bu çalışmada ADS-B cihazlarına karşı yapılan saldırıların çözümlerinde yapay öğrenme teknikleri kullanılarak çözüm modeli sunulmuştur. Önerilen modelde en iyi sonucu elde etmek için bir hava taşıtı simülasyon yazılımından özgün olarak elde edilen veriye sahte veri saldırıları yapılmıştır. Bu sahte veri saldırıları ADS-B verisinde bulunan SSR kod ve uçuş seviyesi alanlarına yapılmıştır. Bu kapsamda hem orijinal hem de saldırılmış veri kümeleri elde edilmiştir. Elde edilen bu veride gereksiz kolonların silinmesi, boş alanların doldurulması, standartizasyon gibi bazı ön işlemler uygulanmış ve yapay öğrenme tekniklerinin kullanımına hazır hale getirilmiştir. Kullanılan yapay öğrenme teknikleri; SVM, Naive Bayes ve İkili Karar Ağacı sınıflandırıcılarıdır. Bu sınıflandırıcılar, farklı sınama oranları üzerinden modellenerek en uygun sınama oranlarının elde edilmesi sağlanmıştır. Yapay öğrenme tekniklerinin kendi içerisinde farklı sınama oranları ve değişkenleri ile elde edilen sonuçların performans değerlendirmeleri tartışılmıştır.

Başlangıç aşamasında çalışma ortamı daha az sürelerde çalıştırılarak değerlendirmeye başlanılmış ve mevcut veri miktarının artmasının etkileri tartışılmış ve sunulan veri miktarı ile optimal sonuçların elde edildiği görülmüştür. Bunun nedeni; çalışma ortamında süreyi artırmamız halinde elde edilecek veri sayısı artarken, bu makalede ele alınan tehdit modeli ve bir ADS-B kaydındaki içerik standart olduğu için alınan sonuçların değişmediği gözlemlenmiştir. Bunu detaylandırmak gerekirse; bir ADS-B verisinde uçak tipi gibi bir özellik bulunmadığı için farklı uçak tiplerinde karşımıza çıkabilecek farklı sonuçları değerlendirmek mümkün olmamaktadır. Böyle bir veri kaydında farklı uçak tipleri için veri miktarının artırmak değerlendirmeyi etkileyecektir. Ancak, mevcut bir ADS-B veri kümesinde; Timestamp (gelen verinin zamanı), Callsign (uçağın tekil adı), Mode-S (uçağın kodu), SSR (hava trafikte kullanılan uçağın kodu), Latitude ve Longitude (enlem ve boylam), hız ve uçuş seviyesi özellikleri bulunmaktadır. Bu parametreler uçak tipinden bağımsız olarak kaydedilen özelliklerdir. Bu nedenle, bu makalede ele alınan sahte veri saldırısı için veri sayısının artması ile elde edilen sonuçlarda da değişiklik olmayacağı değerlendirilmiştir. ADS-B cihazının kapsam ve kapasitesinin yanı sıra simülasyon ortamının nesne üretme gücü de düşünülerek ortam hazırlanmış ve çalışmada kullanılan simülasyon ortamı üzerinden değerlendirmeler yapılmıştır.

Çizelge 1'de görüldüğü gibi SVM uygulanırken Kernel tipi 'RBF' veya 'Linear' iken; kullanmış olduğumuz veride C parametresi kullanılan tekniğin ihlallere karşı hassasiyetini belirlemektedir. C parametresi büyüdükçe duyarlılık oranının da arttığı görülmektedir. C parametresi 0,1 olduğunda duyarlılık oranının çok yüksek olduğu ve doğruluk değerinin de en yüksek değerlere sahip olduğu görülmüştür.

Çizelge 2'de gösterilen sonuçlarda SVM uygulanırken sınama oranı her iki Kernel tipinde (RBF ve Linear); kullanılan verinin eğitim oranı daha yüksek, sınama oranı daha düşük olduğunda daha iyi sonuçlar elde edildiği görülmüştür. Veri

kümesinin sınanan kısmının tüm veri kümesine oranı 0,1 olduğunda doğruluk oranı en yüksek sonuca ulaşılmıştır. Bu kullanılan sınama oranında elde edilen anomalilerde bir doğrusal ayrım olmadığından Kernel tipi RBF olması daha doğru sonuçlar elde edilmesini sağlamıştır.

Çizelge 3'te Gaussian, Multinomial ve Kategorik Naive Bayes tekniklerinin sınama oranı ile beraber karşılaştırmaları verilmiştir. Naive Bayes algoritmasında eğer özellikler sayılabilir değerler bulundurmakta ise MNB tekniği daha iyi sonuçlar vermektedir. CNB tekniğinde ise özelliklerin kategorilere ayrılmış olması gerekmektedir. Kullanılan veri kümesinde özelliklerin sayısal değerler içermesi nedeniyle çizelgede de görüldüğü üzere MNB tekniğinin CNB ve GNB tekniğine göre doğruluk açısından daha iyi sonuçlar verdiği görülmektedir. MNB tekniğinde kullanılan sınama veri kümesinin tüm veri kümesine oranı 0,1 olduğunda Doğruluk ve Kesinlik sonuçlarında daha iyi sonuç elde edildiği görülmüştür.

Çizelge 4'te görüldüğü gibi İkili Karar Ağacı tekniğinde kullanılan sınama veri kümesinin tüm veri kümesine oranı 0,1 olduğunda daha iyi sonuç verdiği gözlemlenmiştir.

Özet olarak, performans değerlendirmesi kapsamında yapay öğrenme tekniklerinden elde edilen modellerin sonuçlarını değerlendirmek için doğruluk, kesinlik, duyarlılık ve F-ölçütü parametreleri kullanılmıştır. Bu çalışma kapsamında Çizelge 5'te de görüldüğü üzere; en iyi sonuç %10 sınama verisi oranı ile İkili Karar Ağacı yapay öğrenme tekniğinde %99,8 doğruluk oranı olarak elde edilmiştir.

Kaynakça

[1] Khandker, S., Turtiainen, H. Costin A. ve Hämäläinen T., *On the (In)Security of 1090ES and UAT978 Mobile Cockpit Information Systems–An Attacker Perspective on the Availability of ADS-B Safety- and Mission-Critical Systems,* in IEEE Access, vol. 10, pp. 37718-37730, 2022, doi: 10.1109/ACCESS.2022.3164704.

[2] Li, T., Wang B., Shang, F., Tian, J., Cao, K., *Online sequential attack detection for ADS-B data based on hierarchical temporal memory,* Computers & Security, vol. 87 (2019) 101599.

[3] Li, T., Wang, B., *Sequential collaborative detection strategy on ADS-B data attack,* International Journal of Critical Infrastructure Protection, vol. 24 (2019), pp. 78-99.

[4] Asari, A., Alagheband, M. R., Bayat, M., Asaar, M.R., *A new provable hierarchical anonymous certificateless authentication protocol with aggregate verification in ADS-B systems,* Computer Networks, vol. 185 (2021), 107599.

[5] Luo, P., Wang, B., Li, T., Tian, J., *ADS-B anomaly data detection model based on VAE-SVDD,* Computers & Security, vol. 104 (2021), 102213.

[6] TajDini, M., Sokolov V., Skladannyi, P., *Performing Sniffing and Spoofing Attack Against ADS-B and Mode S using Software Define Radio,* in 2021 IEEE International Conference on Information and Telecommunication Technologies and Radio Electronics (UkrMiCo), IEEE, 2021, pp. 1–5.

[7] El Marady, A. A. W., *Enhancing accuracy and security of ADS-B via MLAT assisted-flight information system,* in 2017 12th International Conference on Computer Engineering and Systems (ICCES), IEEE, 2017, pp. 182–187.

[8] Wahlgren, A., Thorn, J., *Detecting ADS-B spoofing attacks: using collected and simulated data,* 2021.

[9] Khan, S., Thorn, J., Wahlgren, A., Gurtov, A., *Intrusion detection in automatic dependent surveillance-broadcast (ADS-B) with machine learning,* in 2021 IEEE/AIAA 40th Digital Avionics Systems Conference (DASC), IEEE, 2021, pp. 1–10.

[10] Kacem, T., Kaya, A., Keceli, A. S., Catal, C., Wijsekera, D., Costa, P., *ADS-B Attack Classification using Machine Learning Techniques,* in 2021 IEEE Intelligent Vehicles Symposium Workshops (IV Workshops), IEEE, 2021, pp. 7–12.

[11] Li, N., Lin, L., Li, F., *ADS-B Anomaly Data Detection Using SVDD-based LSTM Encoder-Decoder Algorithm,* in 2021 IEEE 3rd International Conference on Civil Aviation Safety and Information Technology (ICCASIT), IEEE, 2021, pp. 1295–1300.

[12] Damis, H. A., Shehada, D., Fachkha, C., Gawanmeh, A., Al-Karaki, J. N., *A microservices architecture for ADS-B data security using blockchain,* in 2020 3rd International Conference on Signal Processing and Information Security (ICSPIS), IEEE, 2020, pp. 1–4.

[13] Shang, F., Wang, B., Li, T., Tian, J., Cao, K., Guo, R., *Adversarial examples on deep-learning-based ADS-B spoofing detection,* IEEE Wirel. Commun. Lett., vol. 9, no. 10, pp. 1734–1737, 2020.

[14] SHGM, *Mode-S Tahsis İşlemleri | Sivil Havacılık Genel Müdürlüğü.* http://172.16.10.52:81/tr/hava-araci-islemleri/2233-mode-s-tahsis-islemleri (Erişim Tarihi: 02.04. 2023).

[15] Manikanth, *What is the use of data standardization and where do we use it in machine learning,* Analytics Vidhya, Mar. 19, 2021. https://medium.com/analytics-vidhya/what-is-the-use-of-data-standardization-and-where-do-we-use-it-in-machine-learning-97b71a294e24 (Erişim Tarihi: 02.04. 2023).

[16] Nguyen, T. T., Armitage, G., *A survey of techniques for internet traffic classification using machine learning,* IEEE Commun. Surv. Tutor., vol. 10, no. 4, pp. 56–76, 2008.

[17] Deshmukh, D. H., Ghorpade, T., Padiya, P., *Improving classification using preprocessing and machine learning algorithms on NSL-KDD dataset,* in 2015 International Conference on Communication, Information & Computing Technology (ICCICT), IEEE, 2015, pp. 1–6.

[18] Yiğidim, H. A., *Makine Öğrenme Algoritmalarını Kullanarak Ağ Trafiğinin Sınıflandırılması,* Master's Thesis, TOBB Ekonomi ve Teknoloji Üniversitesi Fen Bilimleri Enstitüsü, 2012.

[19] Berwick, R., *An Idiot's guide to Support vector machines (SVMs)*.

[20] *Support Vector Machine Explained-Theory, Implementation, and Visualization*, https://www.linkedin.com/pulse/support-vector-machine-explained-theory-visualization-zixuan-zhang (Erişim Tarihi: 02.04. 2023).

[21] *Naive Bayes Classifier Tutorial: with Python Scikit-learn*, https://www.datacamp.com/tutorial/naive-bayes-scikit-learn (Erişim Tarihi: 02.04. 2023).

[22] Shoba R., Kenta N., Christian S., Micheal G., *Encyclopedia of Bioinformatics and Computational Biology: ABC of Bioinformatics,* Amsterdam: ELSEVIER Yayınları, 2019, 405, books.google.com.tr [Erişim Tarihi: 29.03.2023].

[23] *Naive Bayes Classifiers,* GeeksforGeeks, Mar. 03, 2017. https://www.geeksforgeeks.org/naive-bayes-classifiers/ (Erişim Tarihi: 02.04. 2023).

[24] Aksu G., Dogan, N., *Comparison of Decision Trees Used in Data Mining",* Pegem J. Educ. Instr., vol. 9, no. 4, pp. 1183–1208, 2019.