



**KARMAŞIK SÜREÇLERDEKİ
PROBLEMLERE DERİN ÖĞRENME
YAKLAŞIMLARI: QUADROTOR
ARIZALARININ ERKEN TAHMİNİ VE UÇUŞ
SAĞLIĞI İZLEME MODELLERİ**

Doktora Tezi

Mehmet ÖZCAN

Eskişehir 2025

**KARMAŞIK SÜREÇLERDEKİ PROBLEMLERE DERİN ÖĞRENME
YAKLAŞIMLARI: QUADROTOR ARIZALARININ ERKEN TAHMİNİ VE
UÇUŞ SAĞLIĞI İZLEME MODELLERİ**

Mehmet ÖZCAN

DOKTORA TEZİ

Bilgisayar Mühendisliği Anabilim Dalı

Bilgisayar Yazılımı Bilim Dalı

Danışman: Dr. Öğr. Üyesi Cahit PERKGÖZ

Eskişehir

Eskişehir Teknik Üniversitesi

Lisansüstü Eğitim Enstitüsü

Temmuz 2025

Bu tez çalışması BAP Komisyonu tarafından kabul edilen 21DRP053 no.lu proje kapsamında desteklenmiştir.

JÜRİ VE ENSTİTÜ ONAYI

Mehmet ÖZCAN'ın KARMAŞIK SÜREÇLERDEKİ PROBLEMLERE DERİN ÖĞRENME YAKLAŞIMLARI: QUADROTOR ARIZALARININ ERKEN TAHMİNİ VE UÇUŞ SAĞLIĞI İZLEME MODELLERİ başlıklı çalışması 17/07/2025 tarihinde aşağıdaki jüri tarafından değerlendirilerek “Eskişehir Teknik Üniversitesi Lisansüstü Eğitim-Öğretim ve Sınav Yönetmeliği”nin ilgili maddeleri uyarınca, Bilgisayar Mühendisliği Anabilim dalında Doktora Tezi olarak kabul edilmiştir.

Jüri Üyeleri

Unvan Adı Soyadı

İmza

Üye	: Dr. Öğr. Üyesi Cahit PERKGÖZ
Üye	: Prof. Dr. Kemal ÖZKAN
Üye	: Prof. Dr. Tansu FİLİK
Üye	: Doç. Dr. Fatih AYDIN
Üye	: Doç. Dr. Ahmet ARSLAN

Prof. Dr. Harun BÖCÜK

Lisansüstü Eğitim Enstitüsü Müdürü

DANIřMAN ONAYI

Daniřmanlıđını yrttđm Doktora đrencisi Mehmet ZCAN, KARMAřIK SRELERDEKİ PROBLEMLERE DERİN đRENME YAKLAřIMLARI: QUADROTOR ARIZALARININ ERKEN TAHMİNİ VE UUř SAđLIđI İZLEME MODELLERİ bařlıklı tez alıřmasını tamamlamıřtır. Hazırlamıř olduđu tez tarafımda incelenmiř ve đrencinin tez savunma sınavına alınması bilimsel ve etik aıdan uygun grlmřtr.

Tez Daniřmanı
Dr. đr. yesi Cahit PERKGZ

ÖZET

KARMAŞIK SÜREÇLERDEKİ PROBLEMLERE DERİN ÖĞRENME YAKLAŞIMLARI: QUADROTOR ARIZALARININ ERKEN TAHMİNİ VE UÇUŞ SAĞLIĞI İZLEME MODELLERİ

Mehmet ÖZCAN

Bilgisayar Mühendisliği Anabilim Dalı

Bilgisayar Yazılımı Bilim Dalı

Eskişehir Teknik Üniversitesi, Lisansüstü Eğitim Enstitüsü, Temmuz 2025

Danışman: Dr. Öğr. Üyesi Cahit PERKGÖZ

Gelişmiş teknolojik sistemlerde arızaların erken tespiti, güvenilirlik ve güvenlik açısından hayati öneme sahiptir. Yapay zekânın arıza tespiti konusundaki artan popüleritesine rağmen, mevcut yöntemler genellikle kapsamlı sistem bilgilerini ve sensör verilerini etkili bir şekilde kullanmakta yetersiz kalmaktadır. Toplanan verilerdeki gizli arızaların varlığı, dikkatleri gelişmiş analitik tekniklere olan ihtiyaca çekmektedir. Bu çalışma, karmaşık sistemlerde arızaları erken tespit etmek ve sistem sağlığını gerçek zamanlı izleyebilmek amacıyla düşme tahmini ve uçuş sağlığı tahmini olmak üzere derin öğrenme tabanlı iki yöntem sunmaktadır. Geliştirilen yöntemlerde LSTM tabanlı otokodlayıcılardan faydalanılarak özellik boyutları azaltılmış ve verilerin anlamlı özniteliklere indirgenmesi sağlanmıştır. Modellerin tahmin ve sınıflandırma aşamalarında BiLSTM ve LSTM modelleri kullanılmıştır. Her iki model gerçek quadrotor uçuşlarından elde edilen sağlıklı ve arızalı verilerle eğitilmiş ve doğrulanmıştır. Deneysel sonuçlar, düşme tahmini modelinin kritik bir arızadan 30 saniye önce 0,9771 doğrulukla arıza belirtilerini tespit ettiğini, uçuş sağlığı tahmini modelinin ise uçuş sırasında sağlık durumunu ortalama 0,1538 MAE değeriyle gerçek zamanlı olarak tahmin ettiğini göstermektedir. Bu durum, önerilen yöntemlerin karmaşık sistemlerde operasyonel güvenlik ve güvenilirliği artırmadaki ciddi potansiyelini ortaya koyarken, kapsamlı verilerin ve gelişmiş analitik tekniklerin erken arıza tespiti ve sistem sağlığı izlenmesindeki önemini de ortaya koymaktadır.

Anahtar Sözcükler: Quadrotor arıza belirtileri, Erken arıza tespiti, Derin öğrenme

ABSTRACT

DEEP LEARNING APPROACHES TO PROBLEMS IN COMPLEX PROCESSES: EARLY PREDICTION OF QUADROTOR FAILURES AND FLIGHT HEALTH MONITORING MODELS

Mehmet ÖZCAN

Department of Computer Engineering

Programme in Computer Software

Eskişehir Technical University, Institute of Graduate Programs, July 2025

Supervisor: Asst. Prof. Dr. Cahit PERKGÖZ

Early detection of faults in advanced technological systems is crucial for ensuring reliability and safety. Despite the growing interest in artificial intelligence for fault detection, current methods often fall short in effectively utilizing extensive system information and sensor data. The presence of hidden faults in the collected data further directs the attentions to the need for advanced analytical techniques. This study presents two deep learning-based methods, fault prediction and flight health prediction, to early detect faults and monitor system health in real-time in complex systems. In the developed methods, feature dimensions were reduced using LSTM-based autoencoders, enabling the reduction of data into meaningful features. BiLSTM and LSTM models were used in the prediction and classification stages of the models. Both models were trained and validated with healthy and faulty data obtained from real quadrotor flights. Experimental results show that the fault prediction model detected fault indications with 0,9771 accuracy 30 seconds before a critical failure, while the flight health prediction model estimated the health status in real-time with an average MAE value of 0.1538 during flight. This demonstrates the significant potential of the proposed methods to enhance operational safety and reliability in complex systems, while also emphasizing the importance of comprehensive data and advanced analytical techniques in early fault detection and system health monitoring.

Keywords: Quadrotor fault prognostics, Early fault detection, Deep learning.

TEŞEKKÜR

Tez çalışmam boyunca bilimsel rehberliği, sürekli desteği ve çalışmanın zorlu aşamalarında verdiği fikirlerle tezime olan katkılarından dolayı danışmanım Dr. Öğr. Üyesi Cahit PERKGÖZ'e en içten teşekkürlerimi sunarım. Akademik yolculuğumun her aşamasında göstermiş olduğu anlayış ve motivasyon, bu tezin tamamlanmasında belirleyici rol oynamıştır.

Araştırma sürecimde kıymetli görüş ve önerileriyle katkı sağlayan değerli arkadaşım Furkan KARABAYIR'a; eğitim öğretim hayatım boyunca maddi manevi daima yanımda olan aileme; tez çalışma sürecimde bana uygun ortamı sağlamaya çalışan, uzun çalışma saatlerimi anlayışla karşılayıp daima yanımda olmaya çalışan sevgili eşim Merve Nur ÖZCAN'a ve bu tezin ortaya çıkmasında emeği, desteği ve katkısı olan herkese en içten teşekkürlerimi sunarım.

Mehmet ÖZCAN

17/07/2025

ETİK İLKE VE KURALLARA UYGUNLUK BEYANNAMESİ

Bu tezin bana ait, özgün bir çalışma olduğunu; çalışmamın hazırlık, veri toplama, analiz ve bilgilerin sunumu olmak üzere tüm aşamalarında bilimsel etik ve kurallara uygun davrandığımı; bu çalışma kapsamında elde edilen tüm veri ve bilgiler için kaynak gösterdiğimi ve bu kaynaklara kaynakçada yer verdiğimi; bu çalışmanın Eskişehir Teknik Üniversitesi tarafından kullanılan “bilimsel intihal tespit programı”yla tarandığını ve hiçbir şekilde “intihal içermediğini” beyan ederim. Herhangi bir zamanda, çalışmamla ilgili yaptığım bu beyana aykırı bir durumun saptanması durumunda, ortaya çıkacak tüm ahlaki ve hukuki sonuçları kabul ettiğimi bildiririm.

Mehmet ÖZCAN

İÇİNDEKİLER

	<u>Sayfa</u>
BAŞLIK SAYFASI	I
JÜRİ VE ENSTİTÜ ONAYI.....	II
DANIŞMAN ONAYI	III
ÖZET	IV
ABSTRACT.....	V
TEŞEKKÜR	VI
ETİK İLKE VE KURALLARA UYGUNLUK BEYANNAMESİ.....	VII
İÇİNDEKİLER	VIII
TABLolar DİZİNİ.....	X
ŞEKİLLER DİZİNİ	XI
GÖRSELLER DİZİNİ	XIII
SİMGELER VE KISALTMALAR DİZİNİ	XIV
1. GİRİŞ	1
1.1. Motivasyon.....	1
1.2. Çalışmanın Amacı ve Kapsamı	3
1.3. Çalışmanın Önemi.....	5
1.4. Tezin Yapısı	6
2. LİTERATÜR ÖZETİ.....	9
2.1. İHA’larda Arıza Tespiti	9
2.2. Kestirimci Bakım Yaklaşımları	14
2.3. Arıza Tespiti İçin Derin Öğrenme Teknikleri	15
3. YAPAY SİNİR AĞLARI	20
3.1. Yapay Sinir Ağı Mimarisi	22
3.1.1. Optimizasyon ve geri yayılım.....	24
3.1.2. Yapay sinir ağlarında problemlerli durumlar	25
3.2. Derin Sinir Ağları.....	28

3.2.1. Denetimli derin öğrenme	29
3.2.2. Denetimsiz derin öğrenme.....	31
3.3. Otokodlayıcı.....	32
3.3.1. Kodlayıcı	34
3.3.2. Çözümleyici.....	34
3.4. LSTM - BiLSTM.....	35
3.4.1. Otokodlayıcılar ile öznitelik çıkarma	41
4. YÖNTEM	43
4.1. Düşme Tahmini	43
4.2. Uçuş Sağlığı Tahmini	47
4.3. Değerlendirme Ölçütleri.....	50
5. DENEY SONUÇLARI	53
5.1. Veri Toplama.....	53
5.1.1. Veri setinin tanımı.....	53
5.1.2. Normalizasyon	55
5.1.2.1. Z-skor normalizasyonu	55
5.1.2.2. Min-max normalizasyonu	56
5.2. Verilerin Parçalara Ayrılması: Kayan Pencereler Yönteminin Uygulanışı	56
5.3. LSTM-Otokodlayıcı	57
5.4. Düşme Tahmini	60
5.5. Uçuş Sağlığı Tahmini	70
6. TARTIŞMA VE SONUÇ	77
KAYNAKÇA.....	81
EKLER	
ÖZGEÇMİŞ	

TABLÖLER DİZİNİ

	<u>Sayfa</u>
Tablo 5.1. LSTM-Otokodlayıcı modeli eğitim parametreleri	64
Tablo 5.2. BiLSTM sınıflandırıcı modeli eğitim parametreleri	65
Tablo 5.3. Anomali skoru tahmini modeli için eğitim seçenekleri	70
Tablo 5.4. Sağlıklı uçuşlar için gerçekleştirilen anomali skoru tahmininde MAE ve RMSE değerleri	71
Tablo 5.5. Düşen uçuşlar için gerçekleştirilen anomali skoru tahmininde MAE ve RMSE değerleri	72

ŞEKİLLER DİZİNİ

	<u>Sayfa</u>
Şekil 3.1. Nöron yapısı (http-3)	20
Şekil 3.2. Yapay sinir ağının basitleştirilmiş modeli (http-4).....	21
Şekil 3.3. İki gizli katmanlı yapay sinir ağı (http-5).....	22
Şekil 3.4. Erken durdurma (http-6).....	27
Şekil 3.5. Çapraz doğrulama (http-7)	28
Şekil 3.6. Derin öğrenme modellerinin sınıflandırılması (Rhee, Veloso, ve Krishnamurti, 2023).....	29
Şekil 3.7. CNN mimarisi (Erdal ve Korucuk, 2023)	30
Şekil 3.8. RNN yapısı (http-8).....	31
Şekil 3.9. Otokodlayıcı (Ozcan ve Perkgoz, 2024)	33
Şekil 3.10. LSTM birimi (Ozcan ve Perkgoz, 2024).....	36
Şekil 3.11. BiLSTM yapısı (Ozcan ve Perkgoz, 2024)	39
Şekil 3.12. Öznitelik çıkarımı.....	41
Şekil 4.1. Düşme tahmini stratejisi (Ozcan ve Perkgoz, 2024)	44
Şekil 4.2. Kayan pencereler yöntemi (http-9).....	45
Şekil 4.3. Uçuş sağlığı tahmini stratejisi	48
Şekil 5.1. UAV Log Viewer	54
Şekil 5.2. Normalleştirilmiş gerçek sinyaller ve bunların, gizli katmanda 20 nöron bulunan LSTM-otokodlayıcı ile yeniden yapılandırılan temsilcileri. a) sağlıklı bir uçuşun yunuslama sinyali. b) arızalı bir uçuşun yunuslama sinyali. c) sağlıklı bir uçuşun irtifa sinyali. d) arızalı bir uçuşun irtifa sinyali. (Ozcan ve Perkgoz, 2024)	59
Şekil 5.3. Düşme tahmini sınıflandırıcısı doğruluk grafiği	61
Şekil 5.4. Düşme tahmini sınıflandırıcısı F1 skoru grafiği.....	61
Şekil 5.5. Düşme tahmini sınıflandırıcısı kesinlik grafiği	62
Şekil 5.6. Düşme tahmini sınıflandırıcısı duyarlılık grafiği	62
Şekil 5.7. İyileştirilmiş düşme tahmini sınıflandırıcısı doğruluk grafiği.....	67

Şekil 5.8. İyileştirilmiş düşme tahmini sınıflandırıcısı F1 skoru grafiği	67
Şekil 5.9. İyileştirilmiş düşme tahmini sınıflandırıcısı kesinlik grafiği	68
Şekil 5.10. İyileştirilmiş düşme tahmini sınıflandırıcısı duyarlılık grafiği.....	68
Şekil 5.11. 7. kata ait sağlıklı uçuş için anomali skoru tahmin değerleri	74
Şekil 5.12. 7. kata ait düşen uçuş için anomali skoru tahmin değerleri.....	75



GÖRSELLER DİZİNİ

Sayfa

Görsel 1.1. Düşerek parçalanmış quadrotor görüntüleri; üstteki: (http-1), alttaki: (http-2)	2
--	---



SİMGELER VE KISALTMALAR DİZİNİ

ANN	: Yapay Sinir Ağları
BiLSTM	: İki Yönlü LSTM
BNN	: Bayes Sinir Ağları
CAE	: Şartlı Otokodlayıcı
CGAN	: Şartlı Üretici Rekabetçi Ağ
CNN	: Evrimsel Sinir Ağları
DAE	: Gürültü Giderici Otokodlayıcı
DBN	: Derin İnanç Ağı
DCGAN	: Derin Evrimsel Üretici Rekabetçi Ağ
FCN	: Tam Evrimsel Ağ
DL	: Derin Öğrenme
GAN	: Üretici Rekabetçi Ağ
GRU	: Kapılı Tekrarlayan Birim
IMU	: Eylemsizlik Ölçüm Birimleri
İHA	: İnsansız Hava Aracı
KNN	: K-en Yakın Komşu
LQR	: Lineer Kuadratik Regülatör
LSTM	: Uzun Kısa Süreli Bellek
MAE	: Ortalama Mutlak Hata
MLP	: Çok Katmanlı Algılayıcı
MPC	: Model Kestirimci Kontrol
RBM	: Kısıtlanmış Boltzmann Makinesi
RL	: Pekiştirmeli Öğrenme
RMSE	: Kök Ortalama Kare Hata
RNN	: Tekrarlayan Sinir Ağları
SAE	: Seyrek Otokodlayıcı
SOM	: Kendini Organize Eden Haritalar
SVM	: Destek Vektör Makineleri
VAE	: Varyasyonel Otokodlayıcı
WGAN	: Wasserstein Üretici Rekabetçi Ağ

1. GİRİŞ

1.1. Motivasyon

Günümüzde, gelişmiş teknolojik cihazlar günlük yaşamın vazgeçilmez unsurları haline gelmiş, dünyayla olan etkileşim şekillerini köklü bir biçimde değiştirmiştir. Bu yenilikler arasında quadrotorlar, diğer bir deyişle quadcopterlar veya dronelar, geniş uygulama alanlarıyla dikkat çekmektedir. Taşımacılık, arama-kurtarma operasyonları, güvenlik ve tarım gibi pek çok sektörde yaygın kullanım bulmaları, bu cihazların giderek artan önemini ortaya koymaktadır (Belcastro, Klyde, Logan, Newman, ve Foster, 2017; Zhao, Wang, Xu, ve Jiabin, 2018).

Quadrotorların popülaritesinin altında yatan temel nedenlerden biri, sundukları esneklik ve pratikliktir. Bu araçlar, insan müdahalesine gerek kalmadan zorlu koşullarda görev yapabilmeleriyle öne çıkmakta, özellikle de tehlikeli veya erişimi zor alanlarda operasyonel riskleri en aza indirmektedir. Ancak, bu üstün performansın arkasında karmaşık sensörler, navigasyon sistemleri, iletişim modülleri ve gelişmiş yazılımlar yer almakta; bu da hem geliştirme hem de bakım süreçlerinin maliyetini artırmaktadır. Quadrotorların işlevselliği arttıkça, bu cihazların güvenilir şekilde çalışmasını sağlamak için yeni bakım ve denetim protokollerinin geliştirilmesi kaçınılmaz hale gelmiştir (Afshari ve Pourtakdoust, 2018).

Quadrotorların verimli kullanımı için gereken bakım süreçleri, yalnızca düzenli arıza onarımları ile sınırlı kalmaz; bunun ötesinde, önleyici bakım ve erken arıza tespiti gibi yaklaşımların benimsenmesi büyük önem arz etmektedir. Karmaşık donanım ve yazılımların entegrasyonu, bu tür araçların sürekli olarak güvenli ve verimli çalışmasını sağlamak için erken arıza tespiti yöntemlerinin gerekliliğini ortaya koymaktadır. Bu bağlamda, yapay zekâ ve makine öğrenimi gibi ileri teknolojiler, arıza belirtilerini performans kaybına yol açmadan tespit edebilme potansiyeline sahiptir. Böylece, arama-kurtarma operasyonları gibi kritik görevlerde cihaz arızalarından kaynaklanan olumsuz etkiler en aza indirilebilir ve operasyonel güvenlik artırılabilir.



Görsel 1.1. *Düşerek parçalanmış quadrotor görüntüleri; üstteki: (<http-1>), alttaki: (<http-2>)*

Gelişen teknolojiye rağmen, quadrotorların kapsamlı bakım ve onarım süreçleri hâlâ önemli maliyetler doğurabilmektedir. Bu noktada, daha düşük maliyetli ve uzun ömürlü bakım protokollerinin geliştirilmesi büyük bir gereklilik olarak karşımıza çıkmaktadır. Erken arıza tespiti, yalnızca arıza olasılığını önceden belirlemekle kalmaz, aynı zamanda cihazların kalan faydalı ömrünü de tahmin ederek kaynakların doğru kullanımını sağlar. Böylece, beklenmedik arızalardan kaynaklanan ekstra maliyetlerin önlenmesi mümkün hale gelir.

Bu bağlamda, yapılan bu tez çalışmasının motivasyonu, quadrotorların güvenliğini ve performansını artırmak için derin öğrenme temelli arıza tespit tekniklerinin uygulanabilirliğini araştırmaktır. Bu araştırma, hem teknolojik ilerlemelere katkı sunmayı

hem de bu araçların maliyet-etkin bir şekilde işletilmesini sağlayacak çözümler geliştirmeyi amaçlamaktadır.

1.2. Çalışmanın Amacı ve Kapsamı

Quadrotorlar, karmaşık sistemlerden oluşan araçlar olarak, mekanik arızalara, çevresel koşullara ve öngörülemeyen durumlara karşı oldukça hassastır. Bu araçların sahip olduğu doğrusal olmayan ilişkiler, tasarımda karşılaşılan aerodinamik yetersizlikler, rotorların birbirleriyle olan etkileşimlerinden doğan istenmeyen yan kuvvetler, ve aracı dengeleyip yönlendirmeye yönelik karmaşık kontrol algoritmaları, sistemi daha da karmaşıklştırmaktadır. İleri seviye kontrol mekanizmalarına rağmen, kontrol kaybı ve kaza durumlarıyla hâlâ sıklıkla karşılaşılmaktadır.

Bazı arızalar aniden ortaya çıkarken, bazıları ise başlangıçta belirgin olmayan ancak zamanla şiddetlenen gizli arızalar şeklinde gelişir. Bu arızalar, belirli bir süre boyunca farklı alt sistemleri etkileyerek, nihayetinde cihazın düşmesine veya ciddi bir arıza yaşamasına yol açar. Quadrotor kazaları, genellikle geri döndürülemez hasarlar ve önemli finansal kayıplarla sonuçlandığından, etkili erken arıza tespit mekanizmalarının geliştirilmesi zorunlu hale gelmiştir.

İnsansız hava araçları (İHA) alanında hem geleneksel hem de yapay zekâ tabanlı teknikler kullanılarak arıza teşhis ve tespit yöntemleri üzerine birçok çalışma gerçekleştirilmiştir (Ignatovich, Menou, Karuskevich, ve Maruschak, 2013; Yasniy, Mytnyk, Maruschak, Mykytyshyn, ve Didych, 2024). Bu çalışmaların büyük bir kısmı, belirli sensörlerin veya donanımların arızalarının belirlenmesine odaklanmıştır (Puchalski ve Giernacki, 2022). Model tabanlı yaklaşımlar, örneğin Kalman filtreleri, sensör verilerinden gürültüyü ayıklamak için matematiksel modellerden faydalanmaktadır (Hajiyev, 2016; Liu, Ma, Bin, Xiang, ve Yang, 2016; Vural ve Hacızade, 2016; Wang, Datong, Peng, ve Peng, 2019). Ancak, bu tür yöntemler, yalnızca sistem modelinin doğru tanımlanması ve her bileşenin eksiksiz gözlemlenmesi koşuluyla etkili olabilmektedir. Ancak karmaşık quadrotor sistemleri için kusursuz bir matematiksel model tanımlanması oldukça zordur.

Öte yandan, derin öğrenme yöntemleri, uçuş verilerini kullanarak arıza tespitinde önemli başarılar elde etmiştir. Bu yöntemler, titreşim, akustik yayılım, hız, ivme, akım ve sıcaklık gibi çeşitli sensörlerden toplanan verileri işleyerek arızaları belirler. Örneğin, titreşim verileri, rulman ve dişli kutularındaki arızaları tespit etmek için yaygın olarak

kullanılırken, akustik yayılım verileri, özellikle düşük hızda çalışan koşullarda rulman ve dişli hatalarını saptamak için kullanılır. Ancak, mevcut derin öğrenme yöntemleri çoğunlukla spesifik arıza senaryolarına odaklandığından, erken aşamadaki arızaların tespit edilmesinde yetersiz kalabilmektedir. Bu durum, quadrotor kazalarının önlenmesi açısından kapsamlı bir erken teşhis sistemi ihtiyacını ortaya koymaktadır.

Bu çalışma, literatürdeki bu boşluğu doldurmayı amaçlamakta ve quadrotorlar için özel olarak geliştirilmiş proaktif bir arıza tespit yöntemi sunmaktadır. Araştırmamız, sistemde bulunan tüm sensörlerden (örneğin: jiroskop, ivmeölçer, manyetometre, sıcaklık sensörleri, GPS sensörleri ve akım sensörleri) gelen uçuş verilerini kapsamlı bir şekilde analiz etmeyi hedeflemektedir. Amaç, sistemde meydana gelen en küçük anomalileri dahi erken aşamada tespit ederek, herhangi zarara veya mali kayba yol açmadan önce müdahale edilmesine imkân tanımaktır.

Bu çalışmanın yenilikçi yönü, Uzun Kısa Süreli Bellek (LSTM) tabanlı otokodlayıcı (autoencoder) ve sınıflandırma veya regresyon modelini birleştiren bir çerçeve sunmasında yatmaktadır. Bu yapı, özellik boyutlarını azaltarak erken arıza tespitinde doğruluk oranını artırmayı hedeflemektedir. Otokodlayıcının gizli katmanı, çeşitli sensör verileri arasından harmanlanmış ve süzölmüş birleşik özellikleri belirleyerek çok boyutlu verileri sıkıştırır ve anlamlı bilgiye dönüştürür. Düşme tahmini modelinde İki Yönlü LSTM (BiLSTM) sınıflandırıcı, bu özellikleri hem sağlıklı hem de arızalı durumlarda iki yönlü analiz ederek, arızaların oluşumuna dair erken sinyalleri yakalar. Bu çift yönlü analiz yaklaşımı, arızaların başlangıç belirtilerini önceden fark etme kabiliyetini geliştirir ve böylece düşüş gerçekleşmeden önce olası bir düşüşün erken tahmini sağlanır. Uçuş sağlığı modelinde ise aynı özellikler yardımıyla uçuş sağlığının anlık olarak ne durumda olduğu LSTM kullanan bir regresyon modeliyle tahmin edilir. Böylelikle uçuş esnasında gerçek zamanlı olarak uçuşun ne derece sağlıklı ilerlediği takip edilebilir, anomali derecesi kritik seviyelere ulaştığında felaket senaryosu öncesi tedbir alınabilir.

Araştırmanın kapsamı, quadrotorlar için kapsamlı bir arıza tespit sistemi geliştirilmesini içermektedir. Çalışma boyunca, uçuş verilerinden toplanan ham bilgilerin işlenmesi, LSTM-Otokodlayıcı mimarisi ile özelliklerin azaltılması, BiLSTM sınıflandırıcının kullanımı, LSTM-Otokodlayıcı yardımıyla uçuş sağlığının tayini, LSTM regresör ile anomali seviyesinin tahmini gibi aşamalar ayrıntılı olarak ele alınacaktır. Ayrıca, bu yöntemlerin doğruluğunu artırmak için kullanılan ortalama mutlak hata

(MAE), kök ortalama kare hata (RMSE), doğruluk, kesinlik, duyarlılık ve F1 skoru gibi performans ölçütleri incelenecektir. Böylece, hem erken arıza tespitine yönelik çözüm önerileri getirilecek hem de quadrotor kazalarının önlenmesine katkı sağlanacaktır.

Genel anlamıyla bu çalışma, derin öğrenme tabanlı yaklaşımların quadrotor sistemlerine entegrasyonunu sağlayarak, güvenlik, verimlilik ve maliyet-etkinlik açısından önemli kazanımlar sunmayı amaçlamaktadır. Erken arıza tespitini mümkün kılan bu proaktif yaklaşım sayesinde, quadrotorların sürdürülebilirlik performansı artırılacak ve beklenmeyen arızaların önüne geçilecektir. Bu çalışma, hem teknik literatüre hem de uygulamalı mühendislik alanlarına önemli katkılar sunmayı hedeflemektedir.

1.3. Çalışmanın Önemi

Bu çalışmanın deneysel sonuçları, geliştirilen modelin erken arıza tespit yeteneğini açıkça ortaya koymaktadır. Sunulan düşme tahmini modeli, kritik sistem arızalarından yarım dakika önce arızaların başlangıcını 0,9771 doğruluk oranıyla başarılı bir şekilde tahmin etmiştir. Uçuş sağlığı tahmini yönteminde ise tüm uçuş sürecinde uçuş anomali skoru ortalama 0.1538 MAE ve 0,1687 RMSE değeriyle başarılı şekilde tahmin edilmiştir. Erken uyarı sistemi olarak kullanılabilecek bu iki model, yalnızca quadrotorların güvenlik ve güvenilirliğini artırmakla kalmayıp, aynı zamanda benzer karmaşık sistemlerin yönetiminde de büyük bir potansiyel taşımaktadır.

Quadrotor gibi karmaşık sistemlerde meydana gelen beklenmedik arızalar, hem maddi kayıplara hem de geri döndürülemez zararlara yol açabilir. Geleneksel yöntemlerin çoğu, arızalar gerçekleşmeden önce sistemi uyarmakta yetersiz kalırken, bu çalışma sayesinde proaktif bir arıza önleme yaklaşımı geliştirilmiştir. Model, erken aşamadaki arızaların tespit edilmesi yoluyla, uçuş sırasında yaşanabilecek ani arızaların önüne geçerek operasyonel güvenliği artırmayı hedeflemektedir. Bu yaklaşım, İHA'ların daha yaygın kullanıldığı alanlarda da operasyonel verimliliği artırma açısından önem taşımaktadır.

Çalışmanın bir diğer önemli katkısı, büyük veri setleri kullanılarak yapılan analizlerin, arıza belirtilerinin erkenden tespit edilmesindeki rolünü vurgulamasıdır. Uçuş verilerinin kapsamlı ve detaylı analizleri, makine öğrenimi tabanlı modellerin performansını artırmak için kritik bir unsur olarak öne çıkmaktadır. Quadrotor sistemleri gibi çok sayıda sensörün veri ürettiği ortamlarda, bu sensörlerden toplanan veriler doğru

şekilde işlenmediğinde, erken arıza belirtileri makine öğrenmesi yöntemleriyle yakalanamayabilir. Bu bağlamda, çalışmamızda kullanılan gelişmiş veri işleme ve boyut indirgeme teknikleri, bu tür verilerin anlamlı bilgilere dönüştürülmesini sağlamış ve sunulan modellerin verimliliğini artırmıştır.

Bu tez kapsamında yapılan çalışmaların beş temel katkısı şu şekildedir:

1. Çalışmada kullanılan veri seti, gerçek uçuş verilerinden oluşturulmuş ve model, bu verilerle test edilmiştir. Sahada elde edilen gerçek uçuş verileri, modelin gerçek çalışma koşullarında doğruluğunu ve uygulanabilirliğini ortaya koymak açısından kritik bir avantaj sağlamaktadır.

2. Quadrotor sistemlerinde başlangıç aşamasındaki arızalardan yola çıkarak gerçekleşmesi olası düşüşlerin erken tespitini sağlayan derin öğrenme tabanlı bir sınıflandırıcı model geliştirilmiştir.

3. Düşüşe başlama zamanı bilinen uçuşlar için uçuş anomali seviyesini belirlemek üzere derin öğrenme tabanlı etiketleme yöntemi geliştirilmiştir.

4. Uçuş anomali seviyesini tahmin ederek uçuş sağlığının gerçek zamanlı olarak izlenmesine olanak sağlayan derin öğrenme tabanlı bir regresyon modeli geliştirilmiştir.

5. Sunulan modellerin verimliliği, uçuş verilerinin boyutunu azaltmak için kullanılan derin öğrenme tabanlı tekniklerle artırılmıştır. Burada kullanılan otokodlayıcı yapıları, büyük veri kümelerinin anlamlı bir şekilde özetlenmesini sağlayarak, hem hesaplama maliyetini düşürmüş hem de model performanslarını yükseltmiştir.

Bu çalışmanın geliştirdiği yöntemlerle entegre bir erken uyarı sistemi, quadrotor kazalarını önleme ve bakım maliyetlerini azaltma açısından önemli kazanımlar sunma potansiyeline sahiptir. Aynı zamanda, bu çalışma, yalnızca quadrotorlar için değil, diğer karmaşık mühendislik sistemleri için de uygulanabilir modellerin geliştirilmesine ışık tutmaktadır. Bu kapsamda çalışma, güvenli operasyonların sağlanması, bakım süreçlerinin optimize edilmesi ve kullanıcı güvenliğinin artırılması gibi konularda uygulamalı mühendislik alanına değerli katkılar sunmayı amaçlamaktadır.

1.4. Tezin Yapısı

Bu tez, quadrotor sistemlerinde erken arıza tespiti ve erken uyarı amacıyla geliştirilen derin öğrenme tabanlı modelin teorik altyapısını, deneysel sonuçlarını ve gelecek çalışmalar için önerileri kapsamlı bir şekilde sunmaktadır. Tezin giriş bölümünde quadrotorların güvenliğini sağlamak ve bakım maliyetlerini azaltmak için erken arıza

tespit mekanizmalarının geliştirilmesi gerektiği vurgulanmıştır. Çalışmanın amacı, düşme tahmini ve uçuş sağlığı tahmini üzerine iki yöntem geliştirerek quadrotor sistemlerinde erken arıza tespiti gerekleřtirilmesi olarak ortaya konulmuřtur. Geliřtirilen modellerin, kritik sistem arızalarını nceden tespit ederek operasyonel gvenlięi artırmak ve maliyetleri azaltmak iin byk bir potansiyel sunduęu belirtilmiřtir. Ařaęıda, tezin yapısı detaylı bir biimde aıklanmaktadır:

Blm 2: Literatr zeti

Bu blmde, hava aralarında arıza tespiti konusundaki mevcut alıřmalar ele alınmakta ve incelenmektedir. İlk olarak, uçuř gvenlięini artırmak ve sistem verimlilięini saęlamak iin kullanılan geleneksel ve yapay zekâ tabanlı arıza tespit yntemleri tanıtılmaktadır. Model tabanlı yntemler ve makine ęrenimi teknikleri gibi eřitli yaklařımların avantajları ve sınırlamaları zerinde durulmaktadır. Ayrıca, erken arıza tespiti nemine vurgu yapılarak, gncel arařtırmalarda ortaya ıkan eksik alanlar ve arařtırma bořlukları tespit edilmektedir. Bylece, bu tezde ele alınan problemin gerekesi ve alıřmanın literatre yapacaęı katkı, net bir řekilde ortaya konulmaktadır.

Blm 3: Yapay Sinir Aęları

Bu blmde, yapay sinir aęlarının temel bileřenleri ve alıřma prensipleri tanıtılmıřtır. alıřmamızda kullanılan derin ęrenme modellerinden olan otokodlayıcılar, LSTM ve BiLSTM mimarilerinin yapı ve iřleyiřlerini, bu aęların ęrenme srelerini, optimizasyon tekniklerini ve farklı kullanım alanları bu blmde ele alınmaktadır. Ayrıca, znitelik ıkarımı ve boyut indirgeme gibi veri iřleme yntemlerine dair aıklamalar da sunulmakta, LSTM tabanlı otokodlayıcıların veri sıkıřtırma ve temsil etme sreleri ile ilgili detaylara bu blmde yer verilmektedir.

Blm 4: Yntem

Bu blm quadrotorlar gibi ok rotorlu hava aralarının arıza teřhis ve uçuř saęlıęı tahmini srelerine dair bir inceleme sunmaktadır. Blmde quadrotorların uçuř sırasında karřılařabilecekleri arızaların erken tespiti ve nlenmesi iin kullanılan yapay zekâ tabanlı yntemler ele alınmıřtır. Bunun yanı sıra, düşme tahmini ve uçuř saęlıęı tahmini bařlıkları altında geliřtirilen modellerin teorik temelleri ve uygulama detayları incelenmektedir. Düşme tahmini modeli, quadrotorun uçuř durumunu analiz ederek olası bir dřüş nceden tahmin etmeyi amalarken, uçuř saęlıęı tahmini modeli, uçuř sresince stabilitiyi izleyerek anomali skorları retir. Her iki modelde de sensr verileri

üzerinden yapılan veri hazırlığı, boyut indirgeme, sınıflandırma ve regresyon gibi aşamalar sonucunda uçuşun güvenliği ve sağlığı hakkında önemli öngörüler elde edilir.

Bölüm 5: Deney Sonuçları

Bu bölümde, geliştirilen modellerin deneysel sonuçları ve bulguları ayrıntılı olarak sunulmaktadır. Gerçek uçuş verileri üzerinde yapılan testlerin sonuçları, modellerin arıza tespit yeteneğini ve doğruluğunu değerlendirmek için kullanılmıştır. Ayrıca, modellerin erken arıza tespit performansı, kritik sistem hatalarının öncesinde ne kadar etkili olduğunu göstermek için analiz edilmektedir. Deneysel bulgular, kullanılan performans ölçütlerine göre ayrı ayrı incelenmekte ve güçlü ve zayıf yönler değerlendirilerek sunulmaktadır. Bu bölüm, aynı zamanda modellerin farklı senaryolarda gösterdiği performansı karşılaştırarak, modelin pratik uygulanabilirliği hakkında çıkarımlar sunar.

Bölüm 6: Tartışma ve Sonuç

Bu bölüm, tezin ana bulgularını özetleyerek, çalışmanın önemini ve katkılarını ortaya koymaktadır. Quadrotorlar için geliştirilen erken arıza tespit modellerinin pratik faydaları ve benzer çalışmalarda sağladığı iyileştirmeler bu bölümde ele alınmaktadır. Ayrıca, çalışmanın sınırlamaları tartışılmakta ve gelecekte yapılacak araştırmalar için öneriler sunulmaktadır. Önerilen geliştirme alanları, arıza tespiti ve İHA operasyonlarının güvenliği konularında yeni çalışmalar için yol gösterici olacaktır. Bu bölüm, tezde ele alınan tüm konuların kapsamlı bir özetini sunarak, okuyucunun çalışmanın genel çerçevesini net bir şekilde anlamasına yardımcı olur.

2. LİTERATÜR ÖZETİ

Quadrorotarlarda arıza nedenli olası düşüşün erken tespiti veya düşme riski tespiti üzerine inceleme yaptığımızda literatürde bu konularla ilgili çalışma bulunmadığı görülmektedir. Ancak düşüşe sebep olabilecek quadrorotor arızalarının tespit edilmesi ve olası bir arıza hakkında bir öngörüye varmak açısından bakıldığında karşımıza birçok çalışma çıkmaktadır. Literatür bu yönleriyle ele alındığında ve incelendiğinde çalışmalar üç ana başlıkta özetlenebilmiştir. Hava araçlarında meydana gelen arızaların gelişmiş sistemler tarafından tespiti “İHA’larda Arıza Tespiti” başlığı altında ele alınmıştır. Hava aracının sürekli olarak izlenerek arıza meydana gelmeden önce bakım zamanının geldiğinin belirlenmesi üzerine yapılan çalışmalar “Kestirimci Bakım Yaklaşımları” başlığı altında incelenmiştir. İlerleyen yapay zekâ teknolojisi ile büyük çapta veriden anlamlı çıkarımlar yapılmasını sağlayan derin öğrenme yöntemlerini arıza tespitinde kullanan çalışmalar ise “Arıza Tespiti İçin Derin Öğrenme Teknikleri” adıyla ayrı bir başlıkta tartışılmıştır.

2.1. İHA’larda Arıza Tespiti

Quadrorotarlarda arıza tespit yöntemleri, genel olarak model tabanlı, veri tabanlı ve hibrit yaklaşımlar olmak üzere üç ana kategoriye ayrılmaktadır. Her bir yaklaşımın kendine özgü avantajları ve sınırlılıkları bulunmaktadır. Bu sınırlılıkları anlamak, sistemlerin daha güvenilir ve etkili hale getirilmesi için büyük önem taşımaktadır.

Model Tabanlı Yaklaşımlar

Model tabanlı yöntemler, insansız hava araçları gibi karmaşık sistemlerde arıza tespiti için etkili bir strateji sağlar. Bu yöntemler, matematiksel bir model kullanarak sistemin dinamiklerini anlamayı ve sistem içindeki potansiyel arızaları tespit etmeyi amaçlar. Model tabanlı arıza tespitinde, sistemin davranışını temsil eden bir matematiksel model geliştirilir ve ardından sensör verileri bu modellere göre analiz edilir. Bu yaklaşımlar, bir sistemin hem doğrusal hem de doğrusal olmayan yapısında kullanılabilir; doğrusal modeller sistemin dinamiklerini kolayca temsil edebilirken, doğrusal olmayan modeller karmaşık etkileşimleri ve dinamikleri ele alabilir. Örneğin, bir İHA'nın motor ve kontrol sistemleri sıklıkla doğrusal olmayan dinamikler sergileyebilir; bu nedenle, bu tür sistemler için doğrusal olmayan modeller kullanılmalıdır. Bu modelleri oluşturmak, sistemin fiziksel ve mühendislik prensiplerinin anlaşılmasını gerektirir. Modelleme süreci, sistemin çalışma koşullarının, bileşenlerinin ve etkileşimlerinin ayrıntılı bir

şekilde incelenmesini, bunların her birinin parametre olarak matematiksel modele doğru şekilde aktarılmasını ve model üzerinde yapılan matematiksel hesaplamalarla arıza tespitinin doğruluğunu arttırarak model etkinliğini sağlamayı içerir.

Sensörler, kuruluşların gerçek zamanlı durumunu izlemede kritik bir rol oynar. Ancak, bu sensörlerden elde edilen veriler genellikle gürültülüdür. Gürültü, arıza tespitinin doğruluğunu etkileyen çevresel faktörlerden veya sensörlerin kendisinden kaynaklanabilir. Matematiksel modellerden daha doğru sonuçlar elde edilebilmesi için sensör verilerindeki gürültünün filtrelenmesi gerekmektedir. Bu filtreleme sırasında Kalman filtreleri sıklıkla tercih edilmektedir (Hajiyeve, 2016; Liu, Ma, Bin, Xiang, ve Yang, 2016; Vural ve Hacızade, 2016; Wang, Datong, Peng, ve Peng, 2019). Kalman filtreleri sistemin durumunu tahmin etmek ve ölçüm verilerinden gürültüyü çıkarmak için kullanılan güçlü bir yöntemdir. Sistem dinamikleri hakkında yeterli bilgiye sahip olunup modele yansıtıldığı takdirde geçmiş ölçümlerden elde edilen bilgiler kullanılarak gelecekteki sistem durumlarına ait tahminler yapılabilir.

Model tabanlı arıza tespiti zayıflıklarından bahsedecek olursak ilk zayıflık, sistem karmaşıklığının matematiksel modelle doğru şekilde karşılanması problemidir. İHA'lar, birden fazla bileşen ve alt sistem içerdiğinden, her bir bileşenin doğru bir şekilde modellenmesi oldukça zordur. Ayrıca, modelin doğruluğu, sistemdeki tüm bileşenlerin ve etkileşimlerin doğru bir şekilde anlaşılmasına bağlıdır. Eğer bir bileşenin modeli hatalıysa, bu durum arıza tespitinde yanlış sonuçlara yol açabilir.

Bunun haricinde, model tabanlı yaklaşımlar genellikle yüksek hesaplama gücü gerektirir. Gerçek zamanlı uygulamalarda, bu hesaplamaların hızlı bir şekilde yapılabilmesi kritik öneme sahiptir. Bu nedenle, model tabanlı arıza tespit sistemlerinin tasarımında, hesaplama verimliliği ve hızının artırılması için optimizasyon tekniklerinin kullanılması önemlidir.

Veri Tabanlı Yaklaşımlar

Veri tabanlı yöntemler, sistemden toplanan geçmiş ve gerçek zamanlı sensör verilerini kullanarak arızaları tespit etmeye odaklanır. Büyük veri analitiği veya makine öğrenimi teknikleri ile bir sistemin olağan çalışma prensibine ait veri akış deseni keşfedilebilir. Veri akışında gerçekleşen olağan dışı hareketlenmeler ise yine bu tekniklerle algılanıp arızanın tespiti veya teşhisine yönelik bir gösterge olarak değerlendirilebilir.

Veri tabanlı yaklaşımlarda, derin öğrenme teknikleri önemli bir yer tutar. Özellikle CNN ve LSTM gibi modeller, zaman serisi verilerini ve karmaşık desenleri analiz etmek için sıklıkla kullanılmaktadır. CNN, görüntü işleme ve uzamsal veri analizi konusunda güçlüdür ve İHA'ların sensör verilerindeki uzamsal ilişkileri keşfetmek için etkili bir şekilde kullanılabilir. Örneğin, bir İHA'nın motor performansını izlemek için sensörlerden elde edilen sinyal verileri zaman-frekans analizi teknikleriyle görüntü haline getirilebilir ve bu görüntülerin CNN ile analiz edilmesi ile potansiyel arızaların tespiti üzerine çıkarımlar yapılabilir.

LSTM, dahili belleğe ve çarpımsal kapılara sahip özel bir tekrarlayan sinir ağı türüdür. Bu model, geçmiş veriler üzerindeki zaman bağımlı ilişkileri dikkate alarak gelecekteki durumları tahmin edebilir ve bu özelliğiyle arıza tespitinde etkin rol oynayabilmektedir. Zhang vd. (2023), quadrotor pervane arızalarını tespit etmek için geliştirdikleri LSTM tabanlı derin öğrenme modelinin yüksek başarı oranları elde ettiğini göstermiştir.

Altınörs vd. (2021), karar ağaçları, destek vektör makineleri (SVM) ve k-en yakın komşu (KNN) algoritmalarıyla, fırçasız doğru akım motorlarının arıza teşhisinde yüksek doğruluk oranlarına ulaşmıştır. Bu yöntemlerin gömülü sistemlerde gerçek zamanlı çalışabilme yeteneği de mevcuttur.

Pose vd. (2023), küçük veri kümeleriyle çalışan ve hızlı tahmin süreleri sağlayan bir sinir ağı tabanlı yöntem tasarlamıştır. Sunulan yaklaşımın gerçek zamanlı uygulamalarda uygulanabilirliği gösterilmiştir.

Veri tabanlı yöntemlerin uygulanması, genellikle büyük miktarda etiketlenmiş veriye ihtiyaç duyar. Burada veri toplama ve etiketleme işlemleri zaman alıcı ve maliyetli olabilmektedir. Etiketlenmiş veri, modelin eğitilmesi için kritik öneme sahiptir; çünkü modelin doğru bir şekilde öğrenebilmesi için hangi verilerin arızalı ve hangi verilerin normal olduğunu bilmesi gerekir. Bu süreç, özellikle büyük ölçekli İHA sistemlerinde zorlu bir görev haline gelebilir.

Veri toplama sürecinde sensörlerin doğru çalışması ve verilerin güvenilir olması gereklidir. Hatalı, eksik veya gürültülü veriler, modelin performansını olumsuz etkileyebilir. Bu nedenle, doğru veri sağlanması için uygun sensörlerin seçilmesi ve bakımının yapılması önemlidir. Gürültülü verilerde ise veri ön işleme tekniklerinin uygulanması ve veri kalitesinin artırılması, modelin başarıya ulaşması açısından kritik öneme sahiptir.

Veri tabanlı yaklaşımların diğer bir sınırlılığı, eğitim verisinde bulunmayan veya önceden tanımlanmamış arıza türlerine karşı yetersiz kalmalarıdır. Derin öğrenme modelleri, eğitildikleri verilerin kalitesine ve kapsamına fazlasıyla bağımlıdır. Eğitim verisi, modelin öğrenme sürecinde kritik bir rol oynar; dolayısıyla, eğer model, belirli bir arıza türü için yeterli veri ile eğitilmemişse, bu tür arızaları tespit etmede yetersiz kalabilir.

Derin öğrenme yöntemleri, yüksek miktarda hesaplama gücü gerektirir. Bu durum, sınırlı donanıma sahip İHA sistemlerinde bu tür modellerin uygulanabilirliğini sınırlandırabilir. Gerçek zamanlı arıza tespit uygulamalarında derin öğrenme modelinin eğitim süreci çevrimdışı olarak tamamlanmış olsa bile modelin karmaşıklığı anlık sonuç çıkarımında fazla yük meydana getirebilir. Bu durumda İHA'larda bulunan sınırlı enerji kaynakları ve işlemci kapasiteleri, karmaşık derin öğrenme modellerinin uygulanmasını zorlaştırabilir (Bondyra, Kołodziejczak, Kulikowski, ve Giernacki, 2022). Dolayısıyla daha hafif ve verimli modellerin geliştirilmesi, veri tabanlı arıza tespit sistemlerinin etkinliğini artırmak için önemlidir.

Hibrit Yaklaşımlar

Hibrit yaklaşımlar, geleneksel model tabanlı yöntemlerle modern veri tabanlı yöntemlerin güçlü yönlerini bir araya getirerek daha etkili bir arıza tespit mekanizması geliştirmeyi amaçlamaktadır. Burada temel felsefe, fiziksel sistemlerin dinamiklerini tanımlayan modellerin sağladığı sezgisel açıklanabilirlik ve nedensellik ile, geçmiş verilerden öğrenme yeteneği sayesinde karmaşık örüntüleri yakalayabilen veri odaklı algoritmaların tahmin gücünü birleştirmektir. Model tabanlı yaklaşımlarda fiziksel sistemler diferansiyel denklemlerle veya dinamik modellerle tanımlanır ve bu modeller aracılığıyla beklenen davranışlar hesaplanır. Bu sistemlerde, model çıktıları ile ölçülen sistem çıktıları arasındaki farklar değerlendirilir ve genellikle belli eşik seviyesi üzerinde olan farklar arıza belirtisi olarak yorumlanabilir. Ancak modelin doğruluğu ve parametrik kesinliği bu yöntemin başarısı için hayati önemdedir. Diğer taraftan veri tabanlı yaklaşımlar, geçmişte kaydedilmiş verilerden örüntüleri ve anormallikleri öğrenebilen algoritmalar üzerine kuruludur. Bu algoritmalar, özellikle çok boyutlu ve doğrusal olmayan sistemlerde başarılı sonuçlar vermektedir.

Hibrit yaklaşımlar, bu iki yöntemin birbirini tamamlamasına dayanır. Anidjar vd. (2023) İHA'larda akustik veri üzerinden anomali tespiti için Wav2BC+ adında hibrit bir çerçeve sunmuştur. Çalışmada önerilen modelde parametre sayısı önemli ölçüde

azaltılmış ve bu şekilde modelin gerçek zamanlı uygulamalarda kullanılabilirliği desteklenmiştir. Lee vd. (2024) quadrotor'un hassas konum takibi için arıza sonrası sapmaları önceden öngören hibrit bozulma tahmin algoritması geliştirmiştir. Önerilen yinelemeli hesaplama yöntemi, quadrotor dinamikleri ile aerodinamikler arasındaki karşılıklı etkileri sayısal nicelleştirebilmektedir.

Tong vd. (2023) tarafından sentetik veri üretimi ile model tabanlı özellik çıkarımı entegre edilerek, rotor hatalarının hassas tespiti sağlanmıştır. Yöntemde quadrotor dinamiklerini temsil eden fiziksel temelli bir simülasyon modeli geliştirilmiştir. Bu modelle farklı arıza senaryolarında uçuş davranışlarını simüle edilerek güvenilir sentetik veri üretilmesi sağlanmıştır. Hibrit yöntemin veri tabanlı bileşeninde ise, simülasyon verilerinden faydalanarak eğitilen LSTM tabanlı bir derin öğrenme modeli, pervane devir sayısını kullanarak meydana gelen arıza durumunda oluşacak itki ve torku tahmin eder. Ardından, bu verilerle eğitilen bir CNN sınıflandırıcı, arızalı pervanenin tespitini ve hasar derecesinin sınıflandırılmasını sağlar.

Bazı çalışmalar, model tabanlı yaklaşımlar ile yapay sinir ağlarını (ANN) birleştirerek etkili arıza sınıflandırma ve teşhis yöntemleri sunmuşlardır (Ouadine, Mjahed, Ayad, ve El Kari, 2020; Jing ve Pebrianti, 2016). Jing vd. (2017), Kalman filtreleri ve ANN kullanarak gecikme süresi düşük bir arıza tespit algoritması geliştirmiştir.

Hibrit sistemlerin başarısı yalnızca model ve algoritmaların tasarımıyla değil, bu sistemlerin eğitildiği ve çalıştırıldığı veri kümeleriyle de doğrudan ilişkilidir. Özellikle uçuş koşullarında elde edilen verilerin etiketli ve çeşitlilik içermesi gerekmektedir. Fakat İHA sistemleri genellikle güvenlik nedeniyle sınırlı test ortamlarında çalıştırıldığından, arızaya dair kapsamlı veri elde etmek zorlaşır. Bu nedenle hibrit sistemler çoğu zaman sentetik veri üretimi, benzetim verisi ya da yarı denetimli öğrenme gibi yöntemlere başvurmak zorunda kalmaktadır.

Model tabanlı ve veri tabanlı sistemlerin her ikisinin de genellikle eğitim süreçlerinde önceden tanımlanmış arıza türlerine göre yapılandırıldığını ve bu nedenle yeni, daha önce gözlemlenmemiş arıza türleri karşısında zayıf kaldıkları literatürde vurgulanmıştır (Liu, Chen, ve Zheng, 2020). İHA uçuşu sırasında karşılaşılan çevresel belirsizlikler, sistemde beklenmedik arızaların ortaya çıkmasına neden olabilir. Bu durumda, sistemin davranışı hem modelin öngörmediği hem de veriye dayalı sistemin

daha önce karşılaşmadığı bir örüntüye dönüşür. Bu durum hibrit yaklaşımda başarısızlığa neden olabilir.

Benzer bir sınırlama, bu sistemlerin gerçek zamanlı çalışabilirliğinde ortaya çıkmaktadır. Derin sinir ağları gibi veri tabanlı bileşenler yüksek hesaplama gücü gerektirirken, fiziksel modellerin simülasyonu da işlem yükü oluşturabilir. Gerçek zamanlı uçuş kontrol sistemlerinde ise hesaplama kaynakları sınırlıdır ve gecikmeler sistem güvenliğini riske atabilir. Bu nedenle hibrit yapıların optimize edilmesi, hem hesaplama açısından verimli çalışacak şekilde sadeleştirilmesi hem de entegrasyon açısından daha kompakt yapılar haline getirilmesi gerekmektedir.

2.2. Kestirimci Bakım Yaklaşımları

Yapay zekâ teknolojilerinin gelişmesiyle birlikte İHA'ların arıza oranlarını azaltmak ve güvenliğini artırmak adına kestirimci bakım stratejisi uygulamaları yaygın hale gelmiştir. Bu strateji, ekipman işletim verilerini analiz ederek, ekipmanların kalan faydalı ömrünü tahmin etmeye, buna bağlı olarak da bakım zamanlaması ve stratejisini belirlemeye dayanmaktadır. Mevcut kestirimci bakım yöntemleri arasında, Lineer Kuadratik Regülatör (LQR), Model Kestirimci Kontrol (MPC) ve pekiştirmeli öğrenme (RL) gibi teknikler yer almaktadır. Ancak her bir optimizasyon döngüsünde, gelecekteki çok adımlı çıktıları tahmin etmek ve çözüm kontrol sırasını optimize etmek gerektiğinden, bu süreç çözüm verimliliğini düşürmektedir. Bu nedenle, İHA'ların zamanında ve güvenilir otonom bakımının gerçekleştirilmesi için hesaplama verimliliğini artırma konusu önemli bir zorluk teşkil etmektedir.

Shen vd. (2024) çalışmalarında, çoklu aktüatör bozulmalarını dikkate alarak, bir quadrotor İHA'nın ömrünü tahmin etmek ve otonom bakım stratejisi geliştirmek üzerine odaklanmaktadır. İki aşamalı kalman filtresi yöntemi ile aktüatörlerin bozulma oranları gerçek zamanlı olarak tahmin edilip, entropi ağırlıklı yöntemle birleştirilen bozulma verileri ile uçuş süresi ve sağlık göstergeleri dikkate alınarak sistemin zaman içerisindeki davranışı öngörülme çalışılmıştır.

Ai vd. (2022), çoklu sensör verilerini kullanarak ve gelişmiş derin öğrenme tekniklerinden yararlanılarak İHA'ların doğru kalan faydalı ömrünün tahmini için çok seviyeli bir füzyon transformatör ağı geliştirmiştir. Çalışmada İHA için kalan faydalı ömür doğrudan batarya performansı ile ilişkilendirilmiştir. Model eğitiminde batarya değerlerine ek olarak uçuş hızı ve toplam ağırlık gibi ek parametreler de kullanılmıştır.

Zongqi vd. (2022), quadrotor itki sisteminin kestirimci bakımı için, sensör sinyallerinden elde edilen zamana bağlı öznitelikleri kullanarak bozulma seviyesi sınıflandırması ve sağlık göstergesi tahmini yapmak amacıyla, LSTM tabanlı bir yöntem geliştirmiştir. Çalışmada itki sistemindeki bozulmayı simule etmek için quadrotor pervane bıçağında devamlı olarak büyüyen bir hasar oluşturulmuştur.

Kestirimci bakım çalışmalarına bakıldığında quadrotor veya İHA sistemlerinin bütünsel olarak ele alınmadığı görülmektedir. Quadrotorların ömrünü ve sağlık göstergelerini doğru bir şekilde tahmin edebilecek, alt sistemler için geliştirilmiş kestirimci bakım yöntemlerini entegre şekilde kullanan bir kestirimci bakım modeli bulunmamaktadır. Ayrıca, mevcut çalışmaların çoğu genellikle tek bir tür sensör verisi veya tek bir aktüatör bozulması üzerine odaklanırken, çoklu sensör verilerinin ve bozulma türlerinin bir arada kullanılması, farklı faktörlerin etkileşimli etkilerini hesaba katan genel ve kapsamlı modellerin eksikliği dikkat çekmektedir.

2.3. Arıza Tespiti İçin Derin Öğrenme Teknikleri

Derin öğrenme yöntemleri, sistemde bulunan farklı sensörlerden toplanan verileri işlemeye dayanır. Bu veriler, titreşim, akustik emisyon, hız, ivme, akım, hareket eksenleri ve sıcaklık gibi çeşitli bileşenlerden sürekli olarak elde edilir. Her bir veri türü, farklı arızaların tespitinde kritik roller üstlenir.

Titreşim verisi: parçaların arızalarını bulmayı ve mekanik bileşenlerin ömrünü uzatmayı sağlar. Sistem elemanlarının hareket halinde olduğunu kanıtlayan göstergedir. Parça ve bağlantılarının gevşek, aşınmış veya dengesiz durumları hakkında bilgi edinmek için titreşim verileri kullanılır. Titreşim ölçümleri, özellikle endüstriyel makinelerde ve taşıma sistemlerinde bakım süreçlerini optimize eder ve arızaları önceden tahmin eder. Hız, ivme ve yer değiştirme gibi değişkenler tipik olarak titreşim verilerinin değerlendirilmesine dahil edilir. Bu verilerin analizi, sistemin tipik çalışma koşullarından farklı olduğunu ortaya çıkarabilir. Belirli bir zaman diliminde titreşim verileri sürekli olarak izlenebilir, bu da anormalliklerin zaman içinde nasıl geliştiğini anlamaya yardımcı olur.

Akustik emisyon verisi: Akustik emisyon, maddenin içerisindeki ses dalgaları ile maddenin gerilme ve deformasyonundan kaynaklanan değişikliklerin tespitini mümkün kılmaktadır. Elde edilen veriler, düşük hızda çalışan sistemler ve düşük frekanslı gürültülerin hakim olduğu ortamlarda dişli ve yataklardaki deformasyonları tespit etmek

için oldukça etkilidir. Özellikle İHA gibi cihazlarda arızaların ses analizi yoluyla hızlı bir şekilde belirlenmesini sağlar.

Hız ve ivme verisi: Sistemdeki ani hareketleri veya çarpışmaları belirlemek için ivme verileri kullanılır. İvme değişiklikleri, bir nesnenin hızının zaman içinde nasıl değiştiğini gösteren bir ölçümdür. Uçuş sırasında meydana gelen bu değişiklikler, motor veya pervane dengesizliği gibi mekanik arızaların teşhis edilmesinde kullanılır. İvme verileri, özellikle hava araçları gibi dinamik sistemler için hayati önem taşır. Bu veriler, sistemin rutin çalışmasında meydana gelen sorunları bulmak için kullanılabilir. Ek olarak, ivme verileri, normal dışı ivmelenmelerin arızalı parçaların neden olduğu anormal durumları gösterebileceği için arıza tespit süreçlerinde önemli bir araçtır. İvme ölçümleri, bakım süreçlerini yönlendirmek ve sistemin genel stabilitesini izlemek için de kullanılabilir.

Akım veya voltaj verisi: Akım ve voltaj değerleri, elektrik motor ve diğer elektrikli parçaların arızalanmaları durumunda çoğunlukla değişiklik gösterir ve bu tarz arızaların teşhis edilmesinde kullanılabilir. Elektrik sistemlerinde akım ve voltaj değişikliklerini izlemek, kısa devreler veya aşırı akım gibi olası sorunların erken teşhisine yardımcı olur. Bu veriler, sistemin elektrikli sağlığını değerlendirmek için kullanılabilir. Akım ve voltaj verilerinin kullanılması, sistemin enerji verimliliğini artırmak ve arıza durumlarını öngörmek için çok önemlidir. Bu tür veriler genellikle sistemin çalışma koşullarında meydana gelen değişiklikleri izlemek için sürekli olarak kayıt altına alınmaktadır. Akım ve voltaj verilerinin analizi, sistemdeki elektrikli yük değişimlerini anlayabilir ve arızaları önleyebilir.

Hareket eksenleri: Uçuş kontrol sistemlerinde, dönme, yuvarlanma ve yalpa hareketleri hava aracının üç ekseninde yaptığı dönüşlere ait bilgileri içerir. Dönüşlerde meydana gelen bozulmalar veya sapmalar; işlem sırasında bir tür denge bozukluğuna işaret edebilecek bir durum göstergesi olarak değerlendirilebilir. İlgili veriler eylemsizlik ölçüm birimleri (IMU) sensörlerinden okunur. Ölçülen ve beklenen dönüş hareketleri arasındaki fark, aracın nasıl hareket ettiğini açıklar ve kontrolcünün isteğine göre havadaki durumdan ne kadar sapma olduğunu gösterir. Bu veri türü; uçuş kontrol sistemlerinin performansını değerlendirmek ve burada meydana gelen sorunları tanımlamak için ideal bir araçtır. Hareket eksen verilerinin analizi, sistemin doğru ve dayanıklı şekilde çalışmasını sağlamak için kullanılır.

Sıcaklık verisi: Sıcaklık verileri, motorlar ve elektronik bileşenlerin sağlığını izlemek için son derece önemlidir. Aşırı ısınma, kısa devreler veya sürtünme kaynaklı problemleri gösterebilir. Sıcaklık verileri, sistemin çalışma koşullarındaki değişiklikleri izlemek ve potansiyel arızaların önceden belirlenmesi için kullanılır. Anormal sıcaklık artışları, arızanın ilerleme hızını gösterebilir ve bu da önleyici bakım yapılmasına olanak tanır. Sıcaklık verileri, sistemin genel verimliliğini değerlendirmek için de yaygın olarak kullanılır. Bu verilerin analizi, sistemdeki aşınma ve bozulma süreçlerini izlemek ve bakım stratejilerini optimize etmek açısından kritik öneme sahiptir.

Bahsi geçen sensörlerden elde edilen veriler üzerinde, derin öğrenme algoritmaları koşturularak, sistemdeki arızaların etkili bir şekilde tespit edilmesi mümkündür. Çalışmaların birçoğunda, titreşim ve akustik veriler üzerinde yapılan analizlerin yanı sıra zaman-frekans analizi gibi teknikler uygulanmış ve arıza tespiti üzerinde başarılı sonuçlar elde edilmiştir. Derin öğrenme yöntemleri kullanılarak yapılan çalışmalara ve önerilen yöntemlere aşağıda yer verilmiştir.

Titreşim Verilerine Dayalı Arıza Tespit Çalışmaları

Titreşim verilerini kullanan Özkat vd. (2023), LSTM tabanlı bir model kullanarak İHA'ların kalan ömür sürelerini tahmin edebilmiştir. Gelecekteki ortalama pik frekans değerleri bu model tarafından tahmin edilebilmektedir. Yine Chen vd. (2021) dalgacık analizi ve kapılı geri dönüşüm birimleri (GRU) kullanarak motor titreşimleri ve rüzgar kaynaklı gürültülerden arızaları belirlemiştir.

Jiang vd. (2015), gövde titreşim sinyallerini kullanarak ANN tabanlı bir teşhis sistemi geliştirmiş ve üç seviyeli dalgacık paket çözümlemesi kullanarak rotor arızalarını belirlemiştir. Bir quadrotorun uçuş deneyleri sırasında elde edilen veriler, bu yöntemi doğrulanması için kullanılmıştır.

Akustik Veriye Dayalı Arıza Tespit Çalışmaları

Çok sayıda çalışma, akustik veri analizlerini kullanarak sistem parçalarının arızalarını bulmaya çalışmaktadır. Bondyra vd. (2022), çalışmalarında tekil aktüatör arızalarının yüksek doğrulukla tespit edilmesini ve izole edilmesini sağlayan bir sistem tasarlamışlardır. Bu çalışma, yerleşik akustik veri toplama ve işleme tekniklerinin gerçek çalışma ortamlarında işe yaradığını göstermektedir. Aynı şekilde, Anidjar vd. (2023) tarafından İHA'lardan yayılan ses dalgalarındaki anormallikleri bulmak için derin öğrenme tekniklerini kullanan bir çalışma sunulmuştur. Bu çalışmada çok sayıda parametrenin birlikte kullanılmasıyla yüksek bir doğruluk elde edilmiştir.

Zaman-Frekans Analizi ve Sensör Verileriyle Arıza Tespiti

Guo vd. (2018), sensör arızalarına yönelik kalıntı sinyallerinden özellikler çıkarmak için CNN tabanlı bir yöntem geliştirmiştir. Kısa süreli Fourier dönüşümünü kullanarak sinyalleri zaman-frekans haritalarına dönüştüren bu model, arıza tespitinde umut verici sonuçlar vermiştir. Olyaei vd. (2018), zaman-frekans-genlik grafiklerinden elde edilen renkli görüntüleri derin sinir ağlarında işleyerek başarılı bir sınıflandırma algoritması geliştirmiştir. Bu yöntem, sensör arızalarına yönelik kalıntı sinyallerinden özellikler çıkarmak için CNN tabanlı bir yöntem kullanır. Bu yöntem sayesinde sensör ve aktüatör arızalarının tespitinde olumlu sonuçlar elde edilmiştir.

Fu ve Che (2021) tarafından Fourier dönüşüm teknikleri ve kendini organize eden özellik haritaları olarak da bilinen bir arıza teşhis modeli oluşturulmuştur. Bu model, desen tanıma ve arıza tespitine yönelik kümeleme operasyonlarında başarıyla çalıştırılmış ve kullanılabilirliği ortaya konulmuştur. Huang vd. (2021) çalışmalarında zaman-frekans analizi ve derin öğrenme teknolojilerini birleştiren yeni bir arıza tespit ve sınıflandırma yöntemi geliştirmiştir. Bu yaklaşım, önce rastgele oluşturulan veri kümelerini zaman-frekans alanına dönüştürür. Ardından derin ağlar kullanılarak arıza türleri doğru ve hızlı bir şekilde sınıflandırılır.

Diğer İlgili Çalışmalar ve Önerilen Yaklaşımlar

Zhang vd. (2023), quadrotor pervane arızalarını belirlemek için derin öğrenme modeli kullanarak simülasyon ile gerçek şartlar arasında ortaya çıkan fark üzerine başarılı bir çözüm öne sürmüştür. Araştırmacılar tarafından geliştirilen model, yüksek doğruluk oranlarına ulaşmak için yeni özelliklerin adaptasyonunu sağlamıştır.

Liu vd. (2020), CNN ve transfer öğrenme tekniklerini kullanarak pervane arızalarının tespiti üzerine çalışmış ve bu tekniklerle yüksek doğruluk oranları elde etmişlerdir. Iannace vd. (2019) tarafından ANN kullanılarak dengesiz pervaneleri sınıflandırmak için bir model geliştirilmiştir. Yang vd. (2021) derin kalıntı küçültme ağı olarak bilinen bir ağ kullanarak erken pervane arızalarını belirlemeye çalışmıştır. Zheng vd. (2021), birleştirilmiş arıza sözlükleri ve makine öğrenimi modellerini kullanarak çoklu arıza modlarını aynı anda etiketleyebilen bir yöntem geliştirmiştir.

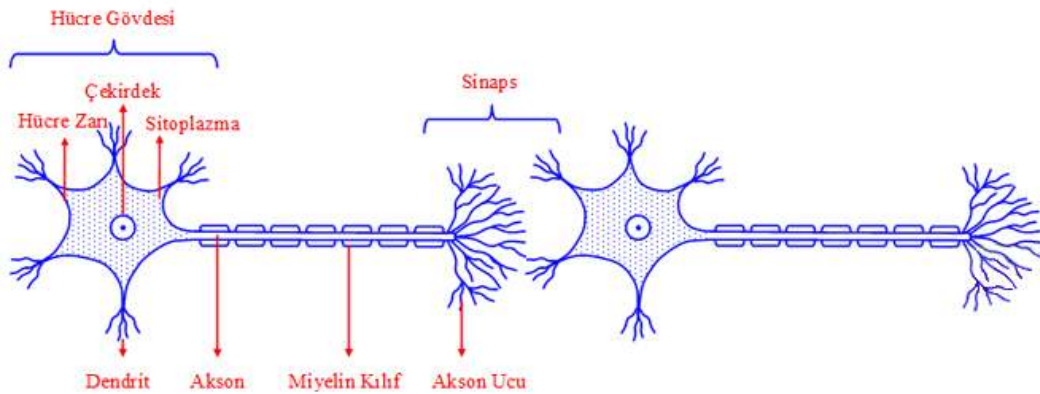
Son olarak, Al Younes vd. (2016) sensörlerdeki bias türü arızaları bulmak için bir arıza toleranslı teşhis ve kontrol yöntemi önermiştir. Ai vd. (2021), arıza özelliklerini çıkarmak için dalgacık paket çevirisi kullanmış ve derin orman algoritmasını arıza teşhisi

iin optimize etmiřtir. Simlasyon ortamında alıřan Erfanian ve Ramezani (2022), BiLSTM algoritmasını kullanarak aktatr arızalarını bařarıyla tespit etmiřtir.



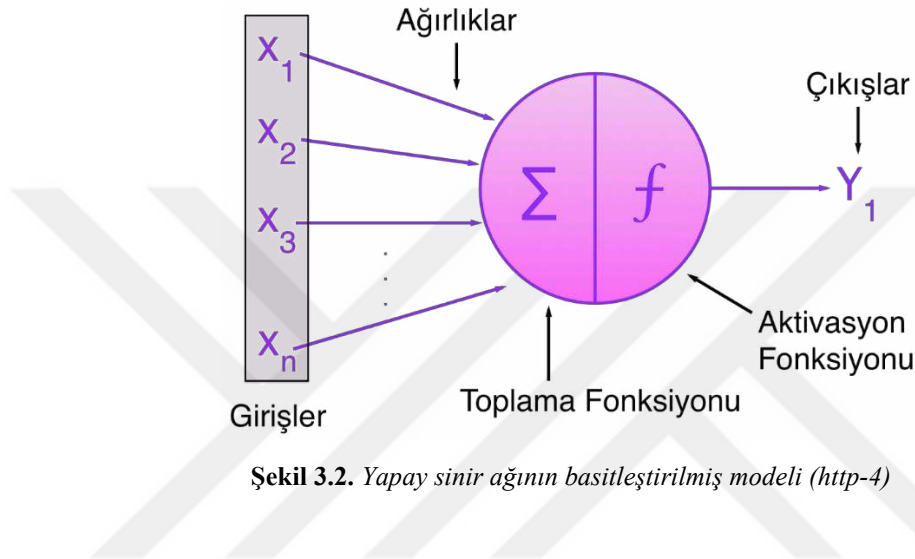
3. YAPAY SİNİR AĞLARI

Yapay sinir ağı, dışsal girdilere verdikleri dinamik durum tepkisiyle bilgiyi işleyen, çok sayıda basit ve yüksek derecede birbirine bağlı işlem elemanından oluşan bir hesaplama sistemidir. Bu fikir, insan beynindeki nöronların çalışma prensibine dair bir modelden türemiştir. İnsan beyni, gerçek dünya durumlarının bağlamını bilgisayarların yapamayacağı şekilde yorumlar. Yapay sinir ağı, insan beynindeki nöron ağını simüle etmeye çalışan bir yapıdır. Böylece bir bilgisayarın, insanların düşündüğü gibi öğrenme ve karar verme yetisine sahip olması amaçlanır. Biyolojik bir nöronun yapısı Şekil 3.1’de gösterilmiştir. Nöron, sinir hücresi olarak da bilinir ve sinaps adı verilen bir bağlantı noktası aracılığıyla diğer hücrelerle iletişim kuran elektriksel olarak uyarılabilir hücrelerdir. Bir nöron üç ana parçaya ayrılabilir. Birincisi, Şekil 3.1’de görülebileceği gibi hücre gövdesidir. İkincisi, hücre gövdesinden çıkan dallanmış yapılar olan dendritlerdir. Sonuncusu ise aksondur; hücrenin en uzun parçası olup birkaç santimden birkaç metreye kadar olabilir ve nöronlar arasındaki ana "telefon hattı" olarak düşünülebilir. Sinaps, dendrit ile akson terminalinin birleşerek iki nöronun birbiriyle iletişim kurmasını sağlayan bağlantıdır. Bu işlem, bir nöron ile nöron olmayan bir hücre arasında da gerçekleşebilir. Bir nörondan diğerine geçen şeye aksiyon potansiyeli denir; bu aksiyon potansiyelini alan nörona postsinaptik nöron/hücre, gönderen nörona ise presinaptik nöron denir. Bu işlem gerçekleştiğinde, postsinaptik nöron da kendi aksiyon potansiyelini başka bir nörona veya hücreye gönderebilir. Bu sürecin anlaşılması, Yapay sinir ağları kavramlarının ortaya çıkmasını sağlamıştır.



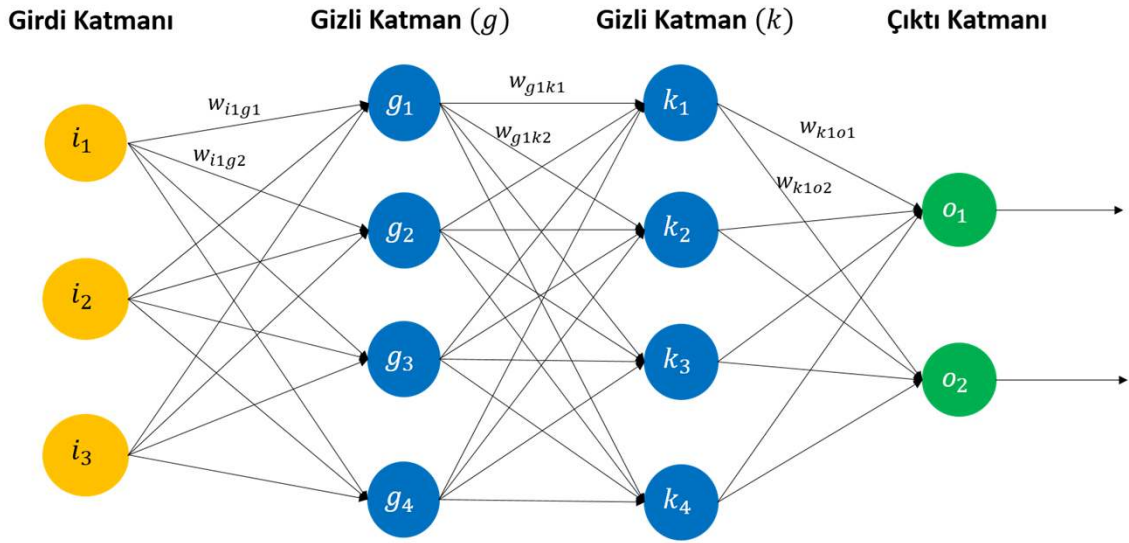
Şekil 3.1. Nöron yapısı (<http-3>)

İlk yapay nöron tasarımı, 1943'te matematikçi Pitts ve nörofizyolog McCulloch tarafından geliştirilmiştir (McCulloch ve Pitts, 1943). Bu yapay nöron tasarımı, biyolojik nörona biraz farklı olarak iki parçaya ayrılmıştır: birinci parça soma, ikinci parça aksondur (McCulloch ve Pitts, 1943). İlk bölümün işlevini girdileri alan ve toplayan bir toplama fonksiyonu üstlenir. Bu toplama sonucunda elde edilen değere göre ikinci bölümde bulunan aktivasyon fonksiyonu bir çıktı üretir. Bunun görseli Şekil 3.2'de gösterilmiştir.



Şekil 3.2. Yapay sinir ağının basitleştirilmiş modeli (<http-4>)

1949'da Kanadalı psikolog Hebb, her sinir yolunda, bahsedilen aksiyon potansiyelinin her geçişte güçlendiğini öne sürerek Pitts ve McCulloch'un fikirlerini geliştirmiştir (Hebb, 2005). Hebb'in çalışmasında yaptığı çıkarıma göre, bir hücre tekrar tekrar diğerini tetiklemeye yardımcı olduğunda, ilk hücrenin aksonu, ikinci hücrenin somasıyla temas eden sinaptik düğmeler geliştirir, veya zaten varsa onları büyütür. Yani, sinaptik boğumun boyutuna bağlı olarak aksiyon potansiyelinin bir nörondan diğerine geçme olasılığı artar. Bu durum, bir ANN'nin çalışma prensibine ve yapısına da yansıtılabilir. Bir ANN'nin temel mimarisi üç katmandan oluşur: girdi katmanı, çıktı katmanı ve aradaki gizli katman(lar). Şekil 3.2 basit bir ANN modeli gösterirken, Şekil 3.3'te üç bölüm daha ayrıntılı görülebilir.



Şekil 3.3. İki gizli katmanlı yapay sinir ağı (<http-5>)

Her bölümün katmanları, tıpkı nöronlar ve hücrelerin iletişim kurabilmesi için temas etmesi gerektiği gibi, birbirine bağlı olmalıdır. Girdi katmanındaki düğümler, gizli katmandaki düğümlere; oradan da çıkış katmanına bağlanır. Girdi katmanı, verileri alır ve gizli katmanlara iletir. Çoğu durumda, girdileri gerektiği gibi ayarlamak için ağırlıklar kullanılır. Bu, sinir hücresindeki dendritin yaptığı işlemin bir benzeridir. Burada gerçekleştirilen işlem çarpma işlemidir. Bu şekilde girdi katmanından gelen ham verileri güçlendirerek veya zayıflatarak alan gizli katmanlar kendi içlerinde bu verileri işler. Girdilerin toplamını alır ve ardından bir aktivasyon fonksiyonundan geçirir. Elde edilen değer çıkış katmanına aktarılır ve orada da işlenerek nihai çıktı üretilir. ANN'nin temel yapısını genel hatlarıyla bu şekildedir. Şimdi ANN mimarisini daha ayrıntılı olarak ele alalım.

3.1. Yapay Sinir Ağı Mimarisi

Bir sinir ağı üç katmandan oluşur. İlk katman girdi katmanıdır ve verileri gizli katmana ileten giriş düğümlerini içerir. Gizli katman, aldığı veriler üzerinde hesaplamalar yapar ve ürettiği çıktıyı çıktı katmanına iletir. Şekil 3.3'te gösterildiği gibi, her düğüm bir sonraki katmandaki düğümlere bağlıdır; bu bağlantıya kanal (channel) denir. Her kanal üzerinde sayısal bir değer atanır, bunlara ağırlık (weight) denir. Ayrıca ağırlıkların yanında, bilgiyi bir düğümden diğerine aktarırken etkili olan sapma (bias) değerleri de

bulunur. ANN'deki ağırlıklar, bir girdinin çıktıyı ne kadar etkileyeceğini belirler. Bu, doğrusal regresyondaki eğime benzer; ağırlık, girdiye çarpılır ve çıktıyı oluşturur. Ağırlıklar, iki düğüm arasındaki bağlantının gücünü belirleyen sayısal parametrelerdir. Yani, bir ağırlık, girdinin çıktıyı ne kadar etkileyeceğine karar verir. Tipik bir durumda, $[x_1, x_2, x_3, \dots, x_n]$ şeklinde girdilerimiz varsa, gizli katmandaki i 'inci nöron için uygulanacak ağırlıklar $[w_{1i}, w_{2i}, w_{3i}]$ olarak gösterilir. Gizli katmanlarda, ağırlıklarla çarpılan girdilerin toplamı alınır, bu aşağıdaki denklemle gösterilir:

$$g(x) = \sum_{j=1}^n x_j w_{ji} \quad (3.1)$$

Burada n , girdi sayısını gösterir ve j , $[1, n]$ aralığındadır. Böylece ağırlıklı toplamı matris çarpımıyla elde edebiliriz. Bias'a gelince, bu değer doğrusal bir denklemdeki sabit terim gibidir. Bias, düğüme giren ağırlıklı toplamın üstüne eklenen ve çıktıyı ayarlamaya yarayan ek bir parametredir. Bu işlem şöyle gösterilir:

$$y = g(x) + b \quad (3.2)$$

Burada $g(x)$, 3.1 denkleminde gelir, b bias'tır ve y çıktıdır. Bias her zaman sabittir. Ağırlıklardan farklı olarak, bias önceki katmandan etkilenmez; ancak yine de çıkış bağlantılarına sahiptir ve sayısal değerler atanır. Böylece, giriş sıfır olsa bile bias sayesinde yine de aktivasyon gerçekleşir.

Denklem 3.2'den elde edilen sonuç, Şekil 3.2'de f olarak görülen gizli katmanlardaki aktivasyon fonksiyonuna getirilir. 3.2 denkleminde çıkan değere bir fonksiyon uygulanır; buna aktivasyon fonksiyonu denir. Aktivasyon fonksiyonları ile, çıktıyı alır ve o düğümün ne derecede aktive olacağına karar verilir. Böylece bilgi, girişten gizli katmanlara, oradan çıkışa doğru ağda ileriye doğru akar; bu sürece ileri yayılım (forward propagation) denir. Gizli katmandan çıktı katmanına geçişte gerçekleşen süreç girdi katmanından gizli katmana geçişteki süreç ile aynıdır. Çıktı katmanındaki fonksiyonlarının çalıştırılması sonucunda nihai çıktılar belirlenmiş olur ve ileri yayılım süreci tamamlanır. İlk ileri yayılımın sonucunda elde edilen çıktı veya çıktılar çoğunlukla yanlış değerlere sahip olur. Bunun sebebi ilk başta her kanalda bulunan ağırlıkların rastgele değerler atanarak sürecin başlatılmasıdır. Yanlış bir çıktı üretildiğinde, ağ yeniden eğitilebilir; bu süreçte çıktı, gerçek çözümlerle karşılaştırılır. Hatanın büyüklüğü anlaşılır ve elde edilen tüm bilgiler ağda geriye doğru iletilir; buna geri yayılım (back propagation) denir. Geri yayılım sürecinde ağırlıklar hata oranları doğrultusunda tekrar güncellenir.

Aktivasyon fonksiyonlarını biraz daha açacak olursak isminin de ima ettiği gibi, bu fonksiyonlar düğümün ne seviyede aktif olacağına karar verir. ANN'nin gizli katmanlarında bulunurlar ve 3.2 denkleminde çıkan değeri giriş olarak alırlar. Burada yapılan, bir çıktı üretmektir; bu çıktı küçük bir değer de olabilir, büyük de. Eğer girdi küçükse, aktivasyon fonksiyonu da küçük bir çıktı üretir; eğer girdi değeri yüksekse, büyük bir çıktı üretir. Üretilen çıktılar bir sonraki düğüme aktarılır. Aktivasyon fonksiyonları, ANN'ye doğrusal olmayan bir yapıya sokar ve ağı daha karmaşık işlemleri öğrenmesini sağlar. Eğer ağıdan aktivasyon fonksiyonu çıkarılırsa, ağ yalnızca basit doğrusal işlemler veya matris dönüşümleri yapabilir ve karmaşık görevleri yerine getiremez. Bölüm 3.1'de anlatıldığı gibi, ANN'deki birçok kavram gibi aktivasyon fonksiyonları da nörobilimden esinlenmiştir. Biyolojik bir nöronda, iç ve dış arasındaki elektriksel potansiyel belirli bir değeri aşarsa, nöron bir zincirleme reaksiyon başlatır ve komşu nöronlara veya hücrelere sinyal iletir. ANN'deki aktivasyon fonksiyonu da tam olarak bu süreci gerçekleştirir.

3.1.1. Optimizasyon ve geri yayılım

Optimizasyon, bir fonksiyonu belirli bir kısıtlama miktarına maruz bırakarak minimize etme veya maximize etme sürecidir. ANN için optimizasyonda amaç, ileri yayılım sonucu elde edilen çıktı değerlerinin, çıkması beklenen değerlerle arasındaki farkı minimize etmektir. Burada ortaya çıkan minimizasyon problemi, bahsi geçen farkı azaltabilecek en iyi ağırlık ve bias değerlerini veya parametreleri bulmak şeklinde ifade edilebilir.

Matematiksel optimizasyon teorisinde, kayıp fonksiyonu veya maliyet fonksiyonu, hata fonksiyonu olarak adlandırılır. Ancak tüm bu isimlerine rağmen, hata fonksiyonu, bir olayın veya bir veya daha fazla değişkenin değerlerini bir gerçek sayıya bağlayarak, o olaya bağlı olabilecek bir tür maliyeti temsil eder. Basit bir ifadeyle, hata fonksiyonu, sinir ağılarımızın veri setimizi ne kadar iyi modellediğini değerlendirmede kullanılan yöntemdir. Tahminler tamamen yanlışsa, hata fonksiyonu yüksek bir sayı çıkartır. Eğer tahminler doğruya daha yakınsa, düşük bir sayı çıkartır. Sinir ağındaki parçalar değiştikçe ve modeli geliştirmeye çalıştıkça, hata fonksiyonu, ilerleme kaydedip etmediğimizi gösterir. Sıklıkla kullanılan hata fonksiyonu bu tezin deneyler kısmında da faydalanan Ortalama Kare Hata (MSE) fonksiyonudur. MSE'yi hesaplamak için tahminler ile gerçek

değer arasındaki farkların kareleri alınır ve tüm veri seti boyunca elde edilen bu karelerin ortalaması alınır. MSE aşağıdaki formülle temsil edilebilir:

$$\min \left(\sum_{j=1}^n (y_j - x_j)^2 \right). \quad (3.3)$$

Burada:

x_j , çıktı değerini,

y_j , gerçek değeri ifade eder.

Geri yayılımda, mevcut ağırlıklar ile kayıp fonksiyonu arasındaki gradyan hesaplanır. Geri yayılımda stokastik gradyan inişi (stochastic gradient descent) olarak adlandırılan yöntem yaygın olarak kullanılır. Stokastik gradyan inişi, bir hedef fonksiyonunu optimize etmek için gradyanlarla birlikte uygun yumuşatma özelliklerini kullanan iteratif bir yöntemdir.

Optimizasyon sürecinde, geri yayılım algoritması kullanılarak ağırlıklar güncellenir. Geri yayılım algoritması, ağırlıkların her adımda hatayı hesaplayarak, bu hatayı azaltacak şekilde ağırlıkların gradyan inişi yöntemiyle günceller. Bu sayede, model zamanla giriş verisindeki önemli özellikleri daha doğru şekilde öğrenir ve modelin performansını artırır.

3.1.2. Yapay sinir ağlarında problemliler

ANN, temel bileşenleri olan nöronların dinamik etkileşimleri sayesinde karmaşık ilişkileri öğrenme ve modelleme yeteneğine sahiptir. Ancak, eğitilmeleri sırasında; kaybolan veya patlayan gradyanlar, aşırı öğrenme, sınırlı veri miktarı gibi bazı güçlükler meydana gelebilir.

Daha önce de bahsedildiği üzere, ANN modelinin eğitimi gerçekleşirken geri yayılım algoritmasında ağırlıkların güncellenmesi gradyan çarpımlarına göre yapılır. Bu gradyan değerleri birçok katmanda aşırı büyüme veya aşırı küçülme eğilimine girdiğinde ağırlık değerlerinin güncellenme sürecinde çok hızlı veya çok yavaş değişmesine sebep olur. Bu durum optimizasyonda sorunlara yol açabilir. Gradyanların aşırı büyümesi sonucu değerlerdeki değişim hızlı gerçekleştiğinde, modelin hatayı yerel minimuma ulaştırma hassasiyeti bozulmuş olur ve öğrenme yeteri kadar doğru gerçekleşmeyebilir. Bu durum literatürde patlayan gradyanlar problemi olarak anılmaktadır. Diğer taraftan gradyanların aldığı aşırı küçük değerler sonucunda ağırlık değerleri çok yavaş şekilde değişmesi, dahası gradyanların sürekli daha da küçülerek etkisiz hale gelmesi optimum

ağırlık değerlerine ulaşamamasına sebep olur. Bu durum ise kaybolan gradyanlar problem olarak ele alınır. Bu tarz durumlar, ağ modelinin öğrenmesi veya kararlı kalmasında zorluklara neden olabilir.

Bu gibi sorunları karşılamak amacıyla yine literatürde farklı farklı yöntemler önerilmiştir. Gradyan kırpma, gelişmiş ağırlık başlatma ve bağlantıları atlama, bu çözüm yöntemlerinin başlıcaları olarak sıralanabilir. Gradyan kırpma yönteminde belirli bir eşik değeri belirlenir. Gradyanlar belirlenen sınırı geçtiğinde ölçeklenerek eşik değerine çekilirler. Ağırlıkların aşırı güncellenmesi veya patlayan gradyanlar problemine karşı bu yöntem etkili bir rol oynar.

Gelişmiş ağırlık başlatma, eğitimin daha hızlı ve stabil başlamasını sağlamak için ağırlıklara ilk değerler atanırken rastgele olarak değil, gelişmiş yöntemlerin kullanılmasını ifade eder. Xavier (Glorot) Initalization, He Initalization ve Orthogonal Initialization bu gelişmiş yöntemlere örnek olarak verilebilir. Gelişmiş ağırlık başlatma yöntemleri kullanıldığında atanan ilk ağırlık değerleri tarafından öğrenmenin yavaş gerçekleşmesinin önüne geçilmiş olur.

Bağlantıları atlama yönteminde bazı katmanların çıktıları doğrudan sonraki katmanlara eklenir. Bu sayede düğüme ulaşan bilgi doğrudan sonraki katmana taşınmış olur. Derin ağlarda katmanların artması gradyanların zayıflamasına ve kaybolan gradyanlar problemine sebep olabilirken bu yöntemin kullanılması gradyan kaybını önler.

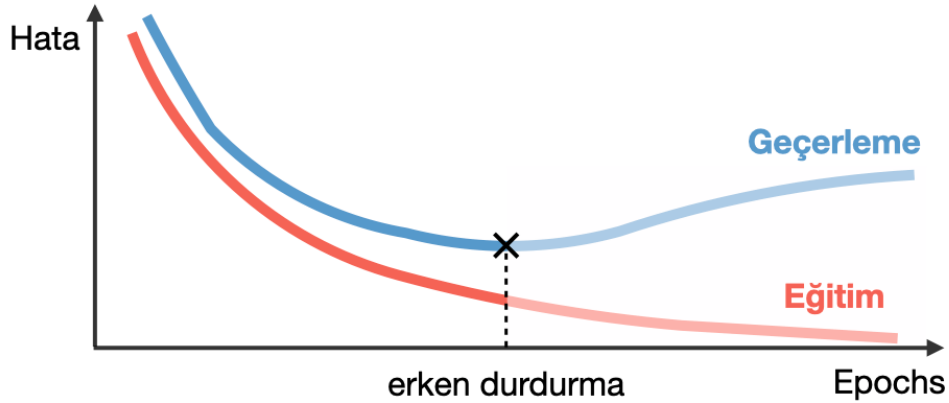
Makine öğreniminde, kimi durumlarda model eğitim esnasında belli bir kalıptan dışarı çıkamaz ve eğitimde kullandığı veriler üzerinden henüz karşılaşmadığı veriler için iyi bir genelleme yapamaz. Bu duruma aşırı öğrenme (overfitting) denir. Aşırı öğrenme nedeniyle, model eğitim setinde mükemmel bir performans sergilerken, test setinde kötü bir uyum gösterir. Bunun nedeni, aşırı öğrenme sağlamış bir modelin, eğitim setindeki bilgilere benzemeyen test setindeki bilgi parçalarıyla başa çıkmada zorluk yaşamasıdır. Öte yandan, aşırı öğrenme gerçekleştirmiş modeller, verilerin arkasındaki düzeni öğrenmek yerine, eğitim setindeki gürültü de dahil olmak üzere tüm verileri ezberleme eğilimindedir. Modelin görülmemiş örneklerle iyi genelleme yapmasını sağlamak için düşürme (dropout), erken durma, düzenleme (regularization), çapraz doğrulama ve daha çeşitli veri kümelerinden faydalanma gibi teknikler kullanarak aşırı uyumu ele almak önemlidir.

Düşürme, eğitim sırasında rastgele olarak belirli nöronların devre dışı bırakılmasıdır. Bu şekilde modelin her eğitim döneminde farklı bir alt küme ile çalışması

garanti edilmiş olur. Böylece model, belirli nöronlara aşırı bağımlı hale gelmez ve daha genel bir şekilde öğrenir ve test setinde daha iyi performans gösterir.

L1 ve L2 Düzenlemeleri aşırı öğrenmeyi önleyici yöntemler arasında yaygınca kullanılır. L1 Düzenlemesi (Lasso), modelin ağırlıklarının toplamının mutlak değerinin bir sabit ile sınırlandırılmasıdır. Bu yöntem bazı ağırlıkların sıfıra düşmesinin yolunu kolaylaştırır. Yapılan düzenleme böylelikle modelin daha basit hale gelmesine ve gereksiz özellikleri ortadan kaldırmasına yardımcı olur. L2 Düzenlemesi (Ridge) ise, modelin ağırlıklarının karelerinin toplamının bir sabit ile sınırlandırılmasıdır. Bu yöntem, L1 düzenlemesine benzer şekilde, ağırlıkları küçük tutarak aşırı öğrenmeyi önlemeye yardımcı olur, ancak ağırlıkları sıfıra düşürmez.

Erken durdurma, modelin eğitim sürecinin, doğrulama setindeki performansın düşmeye başladığı noktada sonlandırılmasıdır. Bu yöntemde amaç eğitim setine aşırı uyum sağlama riskine girmeden önce eğitimin durdurulması ile en iyi genel performansı elde etmeye çalışmaktır. Yöntemin çalışması prensibini ifade eden görsel Şekil 3.4'te paylaşılmıştır.

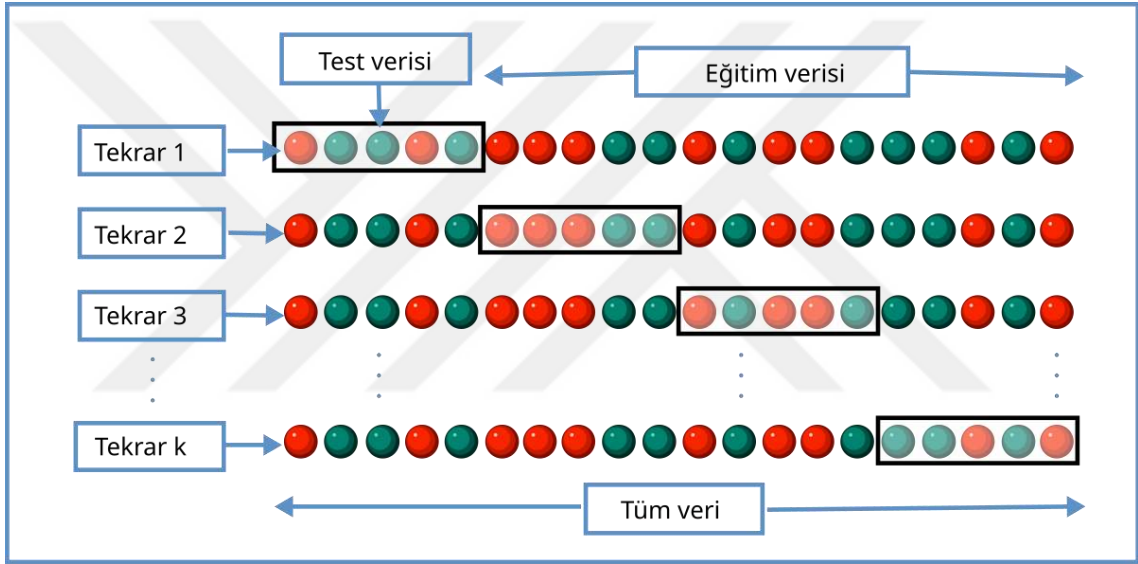


Şekil 3.4. Erken durdurma (http-6)

Doğrulama ya da geçerleme, öğrenme algoritmalarını değerlendirmek ve karşılaştırmak için verileri iki ayrı parçaya ayıran istatistiksel bir yöntemdir. Bu yöntemde ayrılan parçalardan biri modeli öğrenmek veya eğitmek için kullanılırken, diğeri modeli doğrulamak için kullanılır. Çapraz doğrulamada, eğitim ve doğrulama setleri, her veri noktasının doğrulama için bir şansa sahip olmasını sağlamak amacıyla ardışık turlarda yer değiştirmelidir. Çapraz doğrulamanın temel şekli k-katlı çapraz doğrulama (k-fold cross-validation) olarak bilinir. Diğer çapraz doğrulama biçimleri, k-katlı çapraz

doğrulamanın özel durumlarıdır veya k-katlı çapraz doğrulamanın tekrarlanan turlarını içerir.

K-katlı çapraz doğrulama yönteminde, veriler öncelikle k eşit (veya neredeyse eşit) boyutlu kümelere ayrılır. Ardından, k iterasyon eğitim ve doğrulama işlemi gerçekleştirilir; bu iterasyonların her birinde, verilerin farklı bir kümesi doğrulama için ayrılırken, kalan $k - 1$ adet kümede bulunan verilerin hepsi eğitim için kullanılır. Şekil 3.5'te yöntemin uygulanması ile ilgili görsel verilmiştir. Yöntem aşırı öğrenmeyi doğrudan engellemek için kullanılmaz ancak, modelin genellenebilir şekilde uygulanabilirliğini, üzerinde çalışılan kısıtlı veri seti üzerinden göstermek için ideal bir istatistiksel yöntemdir.

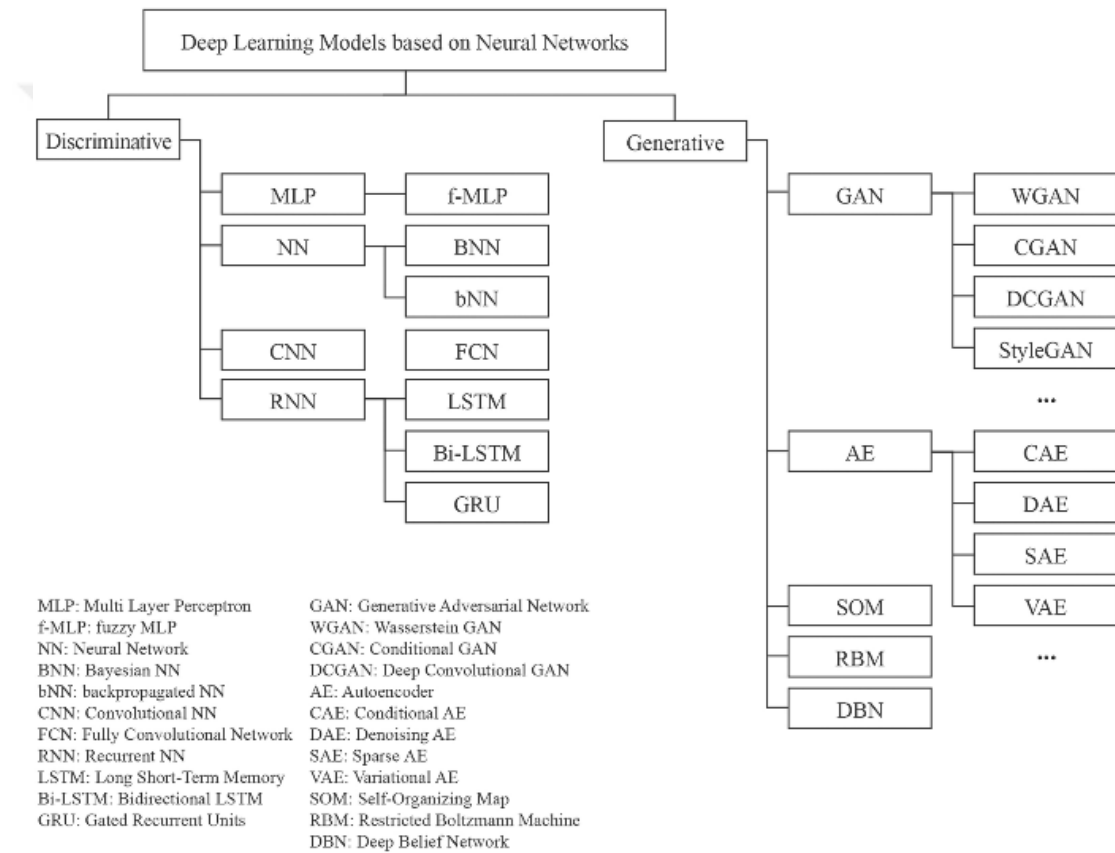


Şekil 3.5. Çapraz doğrulama (<http-7>)

3.2. Derin Sinir Ağları

Derin Sinir Ağları veya Derin Öğrenme, üç katmandan fazla olan sinir ağlarını ifade eden bir terimdir. Yüksek boyutlu verileri işlemek ve giderek daha karmaşık modeller öğrenmek için birden fazla gizli katman ve nöron gerektirir. Derin Öğrenme (DL) sürecinde, nöronlar önceki katmanın çıktısına dayalı olarak bir özellik temsili eğitir. Bu nedenle, büyük ölçekli yüksek boyutlu veri setleriyle çalışırken geleneksel makine öğrenimi yöntemlerinden daha iyi performans gösterir. Ancak, DL yöntemleri, DL modellerini makul bir sürede eğitmek için daha fazla hesaplama gücü gerektirir; bu genellikle birden fazla GPU kullanımı ile sağlanır. DL model geliştirmelerindeki ilerlemeler, GPU'ların artan kullanılabilirliği sayesinde mümkün olmuştur; bu da önemli

ölçüde daha hızlı hesaplama ve bağımsız olarak eğitilebilen katmanlar sağlar. Böylece, milyonlarca parametreye sahip büyük bir model, daha az kaynak gerektiren küçük, yönetilebilir parçalar halinde optimize edilebilir. DL'nin, özellikle yüksek boyutlu verilerde karmaşık problemlerin üstesinden gelme kabiliyeti ile, arıza tespit uygulamaları özelinde geniş bir yelpazede oldukça iyi performans gösterdiği kanıtlanmıştır. Genel bir sınıflandırma yapılacak olursa, DL tabanlı yöntemler, denetimli (supervised-discriminative) ve denetimsiz (unsupervised-generative) yöntemler olarak iki ana gruba ayrılabilir. Şekil 3.6'da bu gruplar altında bulunan DL modelleri ve bu modellerin kendi içinde sınıflandırılması verilmiştir



Şekil 3.6. Derin öğrenme modellerinin sınıflandırılması (Rhee, Veloso, ve Krishnamurti, 2023)

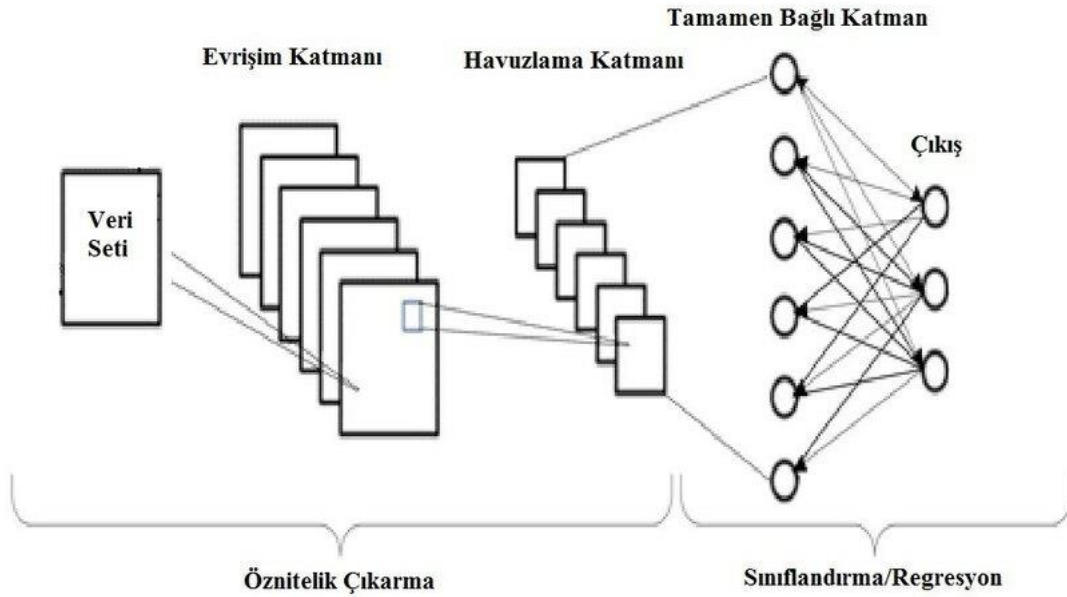
3.2.1. Denetimli derin öğrenme

Bu DL modelleri kategorisi, etiketli veri setleri kullanılarak eğitilmektedir. Denetimli mimariler temel olarak Çok Katmanlı Algılayıcı (MLP), CNN ve RNN ile bunların varyantlarını içerir.

MLP, işlenmek üzere giriş verilerini alan tamamen bağlı bir giriş katmanı, girişle ilgili tahmin veya sınıflandırma yapan bir çıkış katmanı ve bu iki katman arasında

hesaplama işlevlerini gerçekleştiren bir veya daha fazla gizli katmandan oluşan bir ileri beslemeli yapay sinir ağı modelidir. MLP'deki nöronlar, geri yayılım algoritması ile eğitilmektedir. MLP ağının çıktısını belirlemek için çeşitli aktivasyon fonksiyonları kullanılır. Birden fazla katmanı ve doğrusal olmayan aktivasyonları, MLP'yi doğrusal algılayıcıdan ayırır ve doğrusal olarak ayıramayan problemleri çözebilme kabiliyeti sağlar.

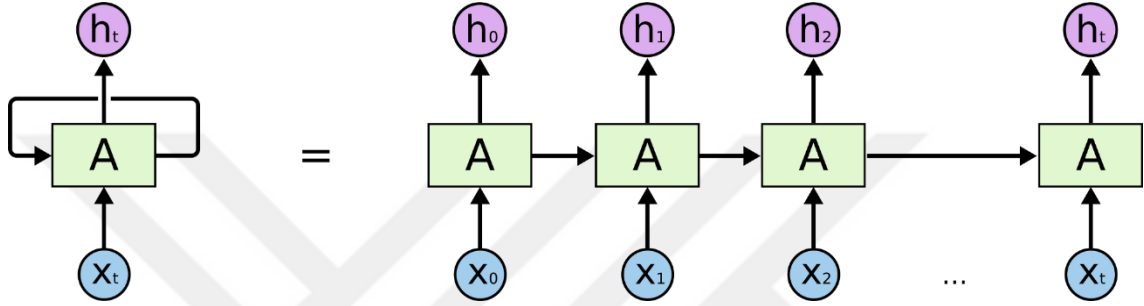
CNN, özellik çıkarımı ve sınıflandırma olmak üzere iki ana kısımdan meydana gelir. Özellik çıkarımı, konvolüsyon ve havuzlama katmanlarını içerir. Konvolüsyon katmanı, bir girişi alır ve üzerinde bir dizi konvolüsyon işlemi gerçekleştirirken, havuzlama katmanı, girişin mekansal boyutları boyunca örnekleme yaparak parametre sayısını azaltır. Ayrıca, tipik sinir ağlarında meydana gelebilecek aşırı öğrenme problemini de çözüme kavuşturan bir yapıya sahiptir. En sonda bulunan sınıflandırma katmanı, önceki adımlarda konvolüsyon katmanından alınan özellikleri kullanarak her sınıf için puan hesaplayan tamamen bağlı bir katmandır. CNN'in mimarisi Şekil 3.7'de gösterilmektedir.



Şekil 3.7. CNN mimarisi (Erdal ve Korucuk, 2023)

RNN, şekil 3.8'de gösterildiği gibi, zaman adımları boyunca sıralı verileri öğrenme yeteneği ile öne çıkan dinamik bir ileri beslemeli sinir ağıdır. Geleneksel ileri beslemeli sinir ağlarında, her bir birimin çıktısı mevcut girdi ile bağımlıdır ve aynı birimin önceki çıktısı ile doğrudan bir bağımlılık yoktur. Ancak, ses tanıma ve zaman serisi verileri gibi

bazı uygulamalar, ardışık örneklerin birbirine bağımlı olduğu sıralı verilere dayanır. LSTM ve GRU, RNN'nin popüler varyantlarıdır. LSTM, önceki verileri hatırlamak için bir bellek hücresi içerir ve hücreye bilgi akışını yönetmek için üç kapı (girdi, çıktı ve unutma) kullanır. Bu varyant, çok sayıda küçük veya büyük türev çarpımları nedeniyle meydana gelebilecek kaybolan ve patlayan gradyanlar sorununu çözer. LSTM modeli ve bu modelin bir çeşit türevi olan BiLSTM modelinden tez çalışmamız kapsamında kullanılmıştır. Bu modeller kendi yöntem ve özellikleriyle LSTM-BiLSTM başlığı altında daha detaylıca incelenecektir.



Şekil 3.8. RNN yapısı (<http-8>)

GRU, RNN'nin bir başka varyantıdır. Kaybolan veya patlayan gradyanlardan kaçınmak için LSTM, mimarisinde birden fazla bellek hücresi gerektirdiğinden yüksek bir bellek gereksinimine sahiptir. LSTM'ye benzer şekilde, GRU, sinir ağındaki hücreler arasındaki bilgi akışını kontrol etmek ve yönetmek için kapı yöntemlerini kullanır; ancak ayrı bellek hücrelerine sahip değildir. Mimari yapısına baktığımızda GRU, LSTM'ye benzer, ancak bir çıktı kapısı bulunmamaktadır ve bu yapı sayesinde, daha önceki bölümlerden bilgi kaybetmeden geniş veri dizilerinden bağımlılıkları uyumlu bir şekilde yakalama yeteneğine sahiptir.

3.2.2. Denetimsiz derin öğrenme

Bu DL kategorisi, etiketlenmemiş veri setlerinde manuel veri etiketlemesine ihtiyaç duymadan gizli kalıpları veya veri gruplarını keşfeder. Denetimsiz öğrenme için yaygın olarak kullanılan derin sinir ağı teknikleri, Üretici Rekabetçi Ağ (GAN), Otokodlayıcı ve Kendini Organize Eden Haritalar (SOM) olmak üzere üç alt kategoride incelenebilir.

GAN, modelin orijinal veri setindeki örneklerle benzer örnekler üretebilmesi için giriş verilerindeki düzenlilikleri veya kalıpları öğrenir. GAN, esasen denetimsiz öğrenme

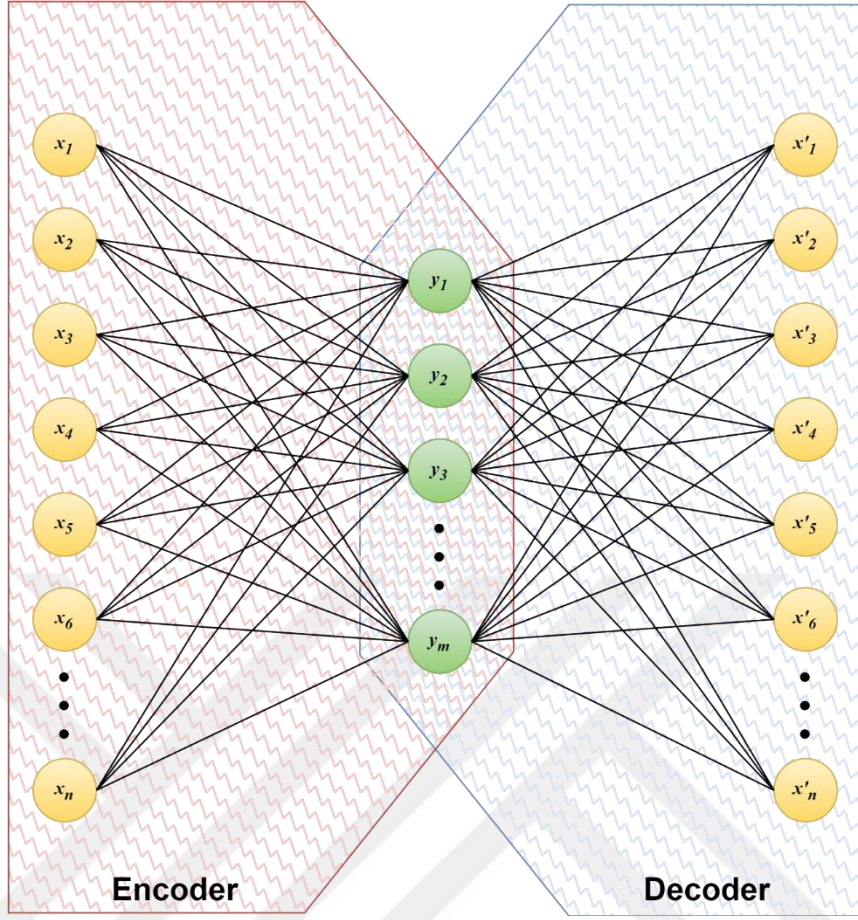
problemleri için tasarlanmış olmasına rağmen, bazı yarı denetimli ve pekiştirmeli öğrenme görevleri için de iyi performans gösterdiği kanıtlanmıştır.

Otokodlayıcı, giriş verilerinin boyutunu azaltarak gürültüsüz veri temsili oluşturur. Yapısında bulunan kodlayıcı verinin boyutunu azaltırken, kod çözücü boyutu tekrar orijinal büyüklüğüne getirir. Giriş ve çıkış arasındaki ilişkiyi öğrenmek için kodlayıcı ile kod çözücü arasında birkaç gizli katman bulunmaktadır ve bu öğrenme, geri yayılım algoritması kullanılarak gerçekleştirilir. Tezimizde kullanılan bu model Otokodlayıcılar başlığı altında daha detaylı şekilde ele alınacaktır.

SOM, veri kümesinin topolojik yapısını korurken düşük boyutlu bir temsil oluşturmak için kullanılan, ileri beslemeli bir yapıya sahip başka bir denetimsiz öğrenme tekniğidir. SOM, gizli katman olmaksızın giriş ve çıkış katmanlarına sahiptir. Her nöron, giriş alanına dayalı olarak aynı anda bir ağırlık değeri alır ve bu ağırlık değeri, nöronlarda herhangi bir aktivasyon fonksiyonu kullanılmadan çıkışa iletilir.

3.3. Otokodlayıcı

Otokodlayıcılar (Şekil 3.4), ANN içinde yer alan özel bir mimari olarak, giriş katmanına verilen verileri çıkış katmanında yeniden oluşturmayı amaçlar. Temel olarak, giriş ve çıkış katmanındaki veriler aynıdır; bu da otokodlayıcıların en belirgin özelliğini oluşturur. Klasik ileri beslemeli sinir ağlarından farkı, otokodlayıcıların giriş ve çıkış katmanlarında aynı sayıda nörona sahip olmasıdır. Diğer bir deyişle, ağın temel amacı veriyi yeniden yapılandırarak aslına en yakın bir şekilde üretmektir.



Şekil 3.9. Otokodlayıcı (Ozcan ve Perkgöz, 2024)

Otokodlayıcı mimarisinde, giriş katmanındaki nöron sayısı genellikle gizli katmanlardaki nöron sayısından fazladır. Bu durum, verinin sıkıştırılması ve kritik özelliklerin öğrenilmesi için gereklidir. Otokodlayıcı, veriyi daha düşük boyutlu bir alanda temsil ederek, gürültü temizleme, boyut indirgeme, görüntü sıkıştırma ve anomalilerin tespiti gibi çeşitli uygulamalarda kullanılabilir. Bu sinir ağı, kodlayıcı (encoder) ve çözümleme (decoder) işlemleri ile birlikte, yeniden yapılandırma hatasını en aza indirme sürecinden oluşur. Bu bileşenler aşağıda ayrıntılı olarak ele alınmıştır.

Kodlayıcı aşaması: Giriş verisinin gizli katmana (latent layer) aktarılması.

Çözümleme aşaması: Gizli katmandaki bilgilerin çıkış katmanında yeniden yapılandırılması.

Optimizasyon süreci: Yeniden yapılandırma hatasının en aza indirildiği süreç.

3.3.1. Kodlayıcı

Bu aşamada, giriş katmanına verilen veriler, gizli katmana aktarılır. Amaç, girişteki tüm özelliklerin daha düşük boyutlu ve anlamlı bir şekilde temsil edilmesini sağlamaktır. Giriş katmanındaki her bir nöron, gizli katmandaki nöronlara ağırlıklandırılmış bağlantılarla aktarılır. Aşağıdaki denklem, şifreleme işleminin matematiksel ifadesini gösterir:

$$y_i = f\left(\sum_{j=1}^n x_j w_{ji}\right), \quad (3.4)$$

Burada:

x_j , giriş katmanındaki j -inci nöronun değerini,

y_i , gizli katmandaki i -inci nörona aktarılan değeri ifade eder.

n , giriş katmanındaki toplam nöron sayısını belirtir.

w_{ji} , giriş katmanındaki j -inci nöron ile gizli katmandaki i -inci nöron arasındaki ağırlığı ifade eder.

f , aktivasyon fonksiyonudur.

Aktivasyon fonksiyonları, verileri doğrusal olmayan bir alanda dönüştürerek daha karmaşık ilişkilerin öğrenilmesini sağlar. En yaygın kullanılan aktivasyon fonksiyonları arasında sigmoid, ReLU ve tanh bulunur. Bu işlem sonucunda, giriş katmanından elde edilen bilgiler sıkıştırılarak, otokodlayıcının gizli katmanında temsil edilir.

3.3.2. Çözümleyici

Şifreleme aşamasından sonra elde edilen veriler, çözümleme (decoding) aşamasında çıkış katmanına iletilir. Bu aşamanın amacı, gizli katmandaki verileri çözümleyerek, giriş verisine mümkün olduğunca yakın bir çıktı üretmektir. Çözümleme işlemi şu şekilde ifade edilir:

$$x'_j = f\left(\sum_{i=1}^m y_i w_{ji}\right), \quad (3.5)$$

Burada:

x'_j , çıkış katmanındaki j -inci nöronun yeniden oluşturulan değerini,

y_i , gizli katmandaki i -inci nöronun değerini,

m , gizli katmanındaki toplam nöron sayısını belirtir.

w_{ji} , gizli katmanındaki j -inci nöron ile çıkış katmanındaki i -inci nöron arasındaki ağırlığı ifade eder.

f , aktivasyon fonksiyonudur.

Bu aşamada, otokodlayıcı giriş verisinde yer alan tüm bilgileri yeniden oluşturmaya çalışır. Ancak, otokodlayıcının yeniden yapılandığı veriler her zaman mükemmel bir kopya olmayabilir. Burada, yeniden yapılandırma kalitesini artırmak için ağırlıkların optimize edilmesi önemlidir.

Otokodlayıcılar, giriş verilerini sıkıştırarak ya da gizli katmanlarda özetleyerek çok boyutlu verilerin analizi için etkili bir araç sunar. Aşağıda otokodlayıcıların kullanıldığı başlıca alanlar verilmiştir:

- Boyut İndirgeme: Yüksek boyutlu verilerin daha düşük boyutlarla temsil edilmesi için kullanılır.
- Gürültü Temizleme: Veriye eklenen gürültünün temizlenmesinde etkili bir yöntemdir.
- Anomali Tespiti: Beklenmeyen davranışları ya da arızaları tespit etmek için, özellikle üretim ve siber güvenlik alanlarında kullanılır.
- Veri Görselleştirme: Karmaşık verilerin görselleştirilmeden önce boyut indirgeme yöntemleri ile sadeleştirilmesine yardımcı olur.
- Görüntü Sıkıştırma: Yüksek çözünürlüklü görüntüleri sıkıştırarak veri depolama alanından tasarruf sağlar.

3.4. LSTM - BiLSTM

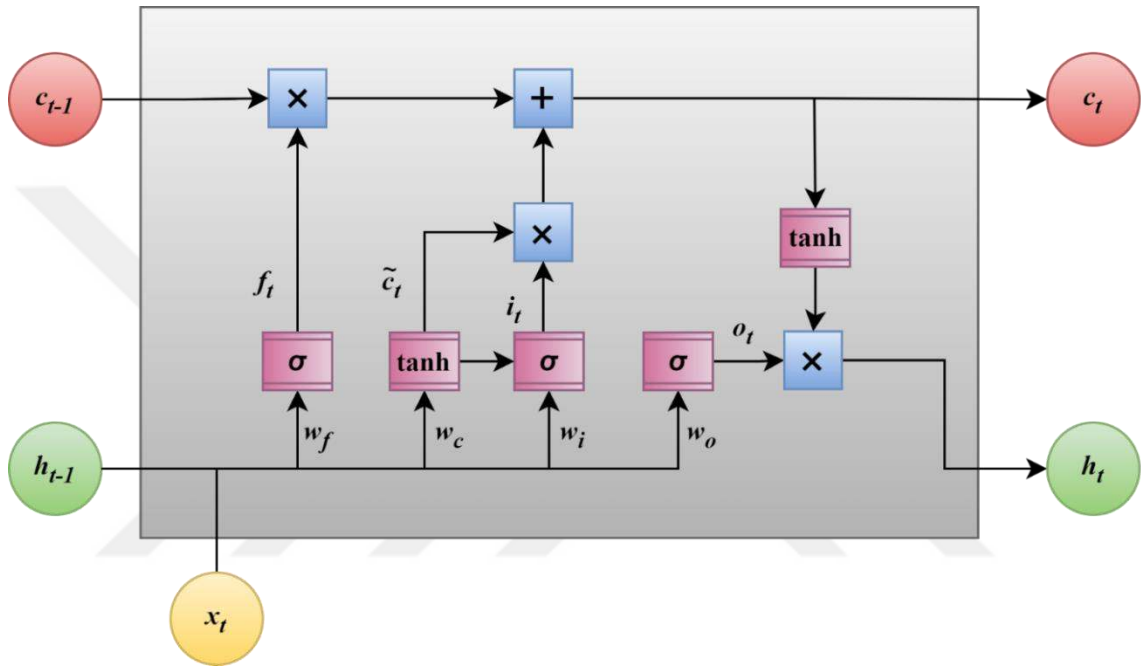
LSTM, RNN'nin geliştirilmiş bir versiyonu olarak öne çıkar ve sinir ağı mimarisinde önemli bir yere sahiptir. RNN, "kısa süreli bellek" sunarak, yalnızca son dönemdeki bilgileri işleyebilir ve kısa süreli bağımlılıkları yakalamak için kullanılır. Ancak, RNN'deki zamanla uzun dizilerde karşılaşılan unutma problemi nedeniyle, eski bilgiler giderek daha az erişilebilir hale gelir. LSTM, bu soruna çözüm getiren bir mimari olarak geliştirilmiştir.

LSTM'nin en büyük farkı, yalnızca kısa vadeli bilgileri değil, geçmişteki tüm verileri bellekte uzun vadeli olarak saklayabilmesi ve gerektiğinde yeniden erişebilmesidir. Bu gelişmiş hafıza yapısı, LSTM'yi özellikle zaman serisi verileri veya ardışık bağımlılıklar gerektiren görevlerde güçlü bir araç haline getirir. Bu nedenle, dil modeli, makine arızası tespiti, uçuş verilerinin analizi ve sinyal işleme gibi birçok alanda yaygın şekilde kullanılmaktadır. Aşağıda, LSTM'nin mimarisi, temel bileşenleri ve çalışma prensibi detaylı olarak açıklanmıştır.

LSTM biriminde (Şekil 3.10), bilginin akışını ve bellek yönetimini sağlayan bir hücre durumu (cell state) ile üç ana kapı (gate) yer alır:

- Giriş Kapısı (Input Gate)
- Çıkış Kapısı (Output Gate)
- Unutma Kapısı (Forget Gate)

Bu kapılar, bilgi akışını düzenlemek ve ağ belleğini dinamik olarak yönetmek için birlikte çalışır.



Şekil 3.10. LSTM birimi (Ozcan ve Perkgöz, 2024)

Hücre Durumu (Cell State): Ağın uzun vadeli belleği olarak işlev görür. Hücre durumu, geçmişteki önemli bilgileri uzun süre koruyarak, ağırlıklandırılmış şekilde güncel verilerle birleşir.

Önceki Gizli Durum (Hidden State): LSTM’de bir önceki zaman adımında üretilen çıkışı saklar ve ağın kısa vadeli belleği olarak görev yapar.

Giriş Verisi: Her zaman adımında, sisteme yeni bir bilgi eklenir ve bu veri, hem mevcut hem de geçmiş bilgilerle işlenir.

Aşağıdaki bölümlerde LSTM'nin çalışma prensibi adım adım ayrıntılı olarak açıklanmaktadır.

Adım 1: Unutma Kapısı (Forget Gate)

Bu adımda, hücrede saklanan bilginin ne kadarının korunacağına veya silineceğine karar verilir. Unutma kapısı, bir önceki gizli durum (h_{t-1}) ile mevcut giriş verisini (x_t)

değerlendirilerek, bir sigmoid aktivasyon fonksiyonu kullanır. Çıktı olarak elde edilen f_t vektörü, 1'e yakınsa bilginin korunması gerektiğini, 0'a yakınsa bilginin silineceğini ifade eder.

Unutma kapısının matematiksel ifadesi şu şekildedir:

$$f_t = \sigma(w_f[h_{t-1}, x_t] - b_f). \quad (3.6)$$

Burada:

w_f ve b_f , sırasıyla unutma kapısının ağırlık matrisi ve bias değerini ifade eder.

h_{t-1} , bir önceki gizli durumdur.

x_t , mevcut zaman adımıdaki giriş verisidir.

σ , sigmoid aktivasyon fonksiyonudur ve çıktıyı 0 ile 1 arasında sıkıştırır.

Adım 2: Giriş Kapısı (Input Gate)

Giriş kapısı, iki aşamalı bir işlemle çalışır. İlk olarak, mevcut girişin hücreye ne kadar dahil edileceğine karar verir ve yeni bilginin hücre durumuna nasıl eklenmesi gerektiğini belirler.

Yeni Bilgi Vektörü: Mevcut veriler, tanjant hiperbolik ($\tanh$) aktivasyon fonksiyonundan geçirilerek $\tilde{c}_t$ vektörü oluşturulur:

$$\tilde{c}_t = \tanh(w_c[h_{t-1}, x_t] + b_c). \quad (3.7)$$

Giriş Kapısı Çıktısı: Aynı zamanda, bir sigmoid aktivasyon fonksiyonu ile, yeni bilginin ne kadarının hücre durumuna eklenmesi gerektiğine karar verilir:

$$i_t = \sigma(w_i[h_{t-1}, x_t] - b_i). \quad (3.8)$$

Son olarak, bu iki işlem sonucunda noktasal çarpım ile yeni bilginin hücre durumuna etkisi hesaplanır:

$$c_t = f_t * c_{t-1} + i_t * \tilde{c}_t. \quad (3.9)$$

Bu denklem, unutma kapısı tarafından korunan eski bilgiyi ve giriş kapısının eklediği yeni bilgiyi birleştirir.

Adım 3: Çıkış Kapısı (Output Gate)

Çıkış kapısı, her bir zaman adımıda ağın yeni gizli durumunu üretir. İlk olarak, sigmoid aktivasyon fonksiyonu, bir önceki gizli durum ve mevcut girişten bir filtre vektörü (o_t) oluşturur:

$$o_t = \sigma(w_o[h_{t-1}, x_t] - b_o). \quad (3.10)$$

Sonrasında, güncellenmiş hücre durumu tanh aktivasyon fonksiyonundan geçirilir:

$$h_t = o_t * \tanh(c_t). \quad (3.11)$$

Bu işlem, ağın hem uzun vadeli bellekten hem de mevcut bilgidен yararlanarak güncellenmiş çıkış üretmesini sağlar.

LSTM’de, gizli durum ve hücre durumu, her bir zaman adımında güncellenerek bir sonraki birime aktarılır. Bu yapı sayesinde LSTM, dizilerdeki uzun vadeli bağımlılıkları öğrenir ve kaybolan gradyan sorununu önleyerek RNN'lerden çok daha etkili çalışır.

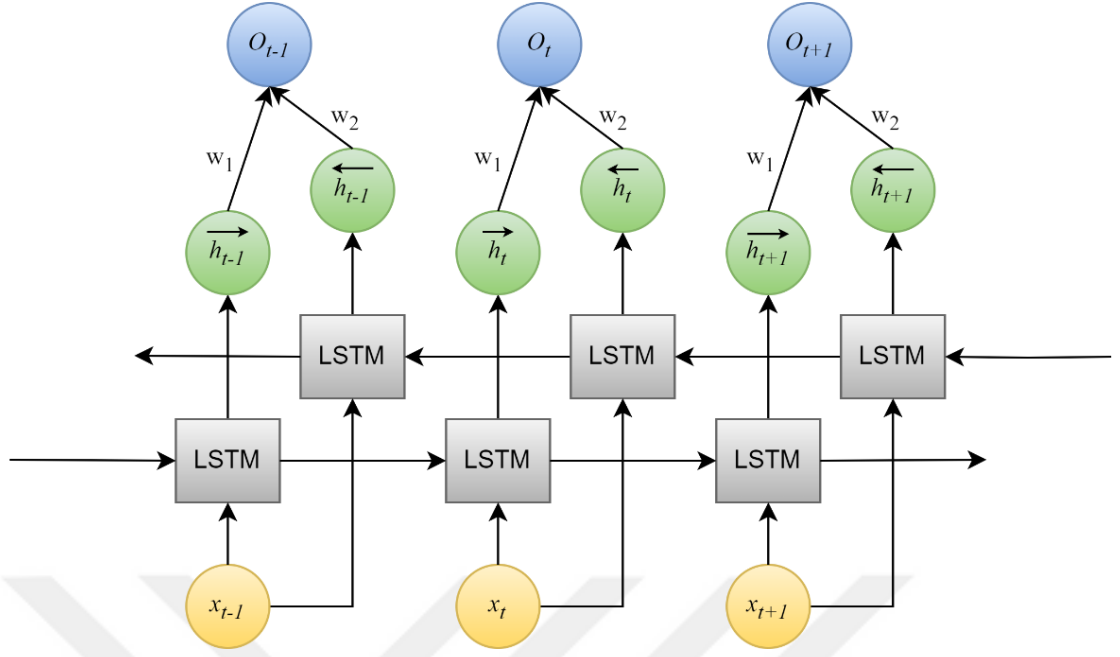
LSTM’nin, ardışık veri setlerinin analizinde oldukça başarılı olması, birçok farklı uygulama alanında kullanılmasını sağlamıştır:

- Zaman serisi analizi: Finansal veriler, hava tahminleri ve üretim süreçleri.
- Doğal dil işleme: Metin sınıflandırma, makine çevirisi ve konuşma tanıma.
- Makine arızası tespiti: Titreşim, hız ve sıcaklık verileri gibi sensörlerden gelen verilerin analizi.
- İHA ve robotik sistemler: Uçuş kontrolü ve hareket tahmini.

LSTM’nin güçlü belleği ve ardışık bağımlılıkları yakalama yeteneği, arızaların erken teşhisi ve önleyici bakım için oldukça kritik avantajlar sunar. Bu nedenle, özellikle İHA'larda pervane ve motor arızalarının teşhisi gibi görevlerde yaygın olarak tercih edilir.

Geleneksel LSTM ağları, geçmiş verilere dayalı bağlamı dikkate alırken, gelecek verilerden elde edilebilecek bilgileri göz ardı eder. Zaman serisi verilerini yeniden üretmek amacıyla LSTM kullanımı, otokodlayıcılar için etkili bir stratejidir. Ancak, zaman serisi verilerinin sınıflandırma performansını daha da artırmak için çift yönlü veri işleme yönteminden faydalanmak mümkündür. Zaman serisi sensör verilerinde belirgin olan geçici bağımlılıklar, tek yönlü bir yaklaşım ile tam olarak yakalanamayabilir. Bu sorunun üstesinden gelmek ve hem geçmiş hem de gelecek zamana ait bilgileri eşzamanlı olarak kullanmak için, BiLSTM yapısı tercih edilir.

Bu özel mimari, veri dizisini ileri ve geri yönlerde işlemekten geçirerek, iki farklı gizli katmandan elde edilen bilgileri tek bir çıktı katmanında birleştirir. BiLSTM yapısının genel akışını ve veri işleme kapasitesini görselleştiren bir model, Şekil 3.11’de sunulmaktadır. Bu yapının çift yönlü veri akışı, giriş verilerinin zaman dinamiklerini daha iyi kavramasını sağlar.



Şekil 3.11. BiLSTM yapısı (Ozcan ve Perkgöz, 2024)

BiLSTM, girdi dizilerini hem ileri hem de geri yönlerde işleyen LSTM birimlerine dayanır. BiLSTM'nin iki yönlü veri işleme özelliği sayesinde, ağ hem geçmişten gelen bağımlılıkları hem de gelecekteki olası bağlamları öğrenir. BiLSTM'nin ileriye doğru geçiş (forward pass) aşamasında, veri dizisi başlangıçtan sona doğru akarken, geriye doğru geçiş (backward pass) aşamasında veri tersine çevrilir ve sondan başa doğru işlenir. Bu iki süreçte elde edilen bilgiler, iki ayrı gizli durum olarak tutulur.

Her iki yönde hesaplanan bu gizli durumlar, geçmiş ve geleceğe ait bilgileri içerir ve ReLU gibi bir aktivasyon fonksiyonuyla birleştirilerek tek bir çıktı haline getirilir. Çift yönlü bu yaklaşım sayesinde ağ, girdi dizisinin tüm zaman boyutlarına ilişkin anlamlı bir temsil üretir. İki yönlü veri işleme süreci, aşağıdaki denklemde gösterilmektedir:

$$O_t = g(w_1 \vec{h}_t + w_2 \overleftarrow{h}_t). \quad (3.12)$$

Burada:

O_t her iki yönlü bilgi akışından birleştirilmiş çıktıyı,

$\vec{h}_t$, ileri yönde hesaplanan gizli durumu,

$\overleftarrow{h}_t$, geri yönde hesaplanan gizli durumu,

w_1 ve w_2 , ilgili yönlerde ait ağırlıkları,

g , genellikle ReLU aktivasyon fonksiyonunu ifade eder.

Bu denklem, ileri ve geri yönlerde elde edilen bilgilerin ağırlıklandırılarak birleştirildiğini ve böylece ağırların hem geçmişe hem de geleceğe ait bağlamları içeren bütüncül bir temsil oluşturduğunu gösterir.

Oluşan birleşik temsil, BiLSTM'nin çıktı katmanına aktarılır ve belirli bir görev için uygun hale getirilir. Bu görev, sınıflandırma, tahmin veya farklı bir öngörü işlemi olabilir. BiLSTM'nin eğitim sürecinde, modelin ürettiği tahminler ile gerçek hedefler arasındaki hata hesaplanır. Bu hatayı en aza indirmek için geri yayılım ve optimizasyon teknikleri kullanılır.

Optimizasyon sırasında, ağırlıklar güncellenir ve hatanın sürekli olarak minimize edilmesi sağlanır. Eğitim süreci tamamlandığında, BiLSTM modeli, hem geçmiş hem de gelecek bağlamlara dayalı olarak veriyi daha etkili bir şekilde öğrenir ve yüksek doğrulukla tahmin yapabilir. Eğitimden sonra çıkarım (inference) aşamasında, BiLSTM'nin çift yönlü veri işleme kabiliyeti sayesinde, zaman serisi verisi üzerinde başarılı tahminler yapılır.

Zaman serisi verilerinin en temel özelliklerinden biri, geçmiş ve gelecekteki durumların birbirine bağımlı olmasıdır. Örneğin, bir sensör verisinin anlık değişimi, hem önceki hem de sonraki anların durumlarından etkilenebilir. Tek yönlü LSTM ağları, yalnızca geçmişe ait bilgileri işlediği için, gelecekteki bağımlılıkları öğrenme konusunda sınırlı kalır. BiLSTM, çift yönlü veri işleme özelliği sayesinde bu sorunun üstesinden gelir ve hem geçmiş hem de gelecek bağlamları kullanarak daha güçlü bir model oluşturur.

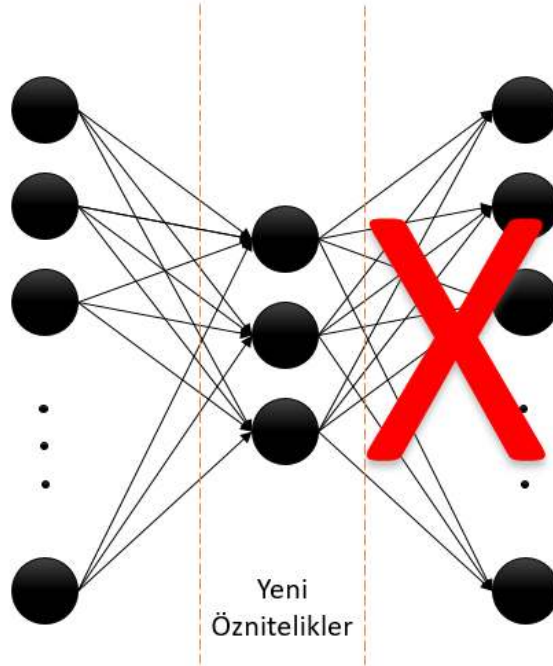
BiLSTM'nin zaman serisi verilerindeki başlıca avantajları şunlardır:

1. Geliştirilmiş Zaman Dinamiği Kavrayışı: Geçmiş ve gelecek bağlamların birleştirilmesi, modelin kapsamlı bir zaman anlayışı geliştirmesini sağlar.
2. Yüksek Sınıflandırma Performansı: BiLSTM, sensör verileri gibi karmaşık zaman serisi verilerinde tek yönlü LSTM'ye göre daha yüksek doğruluk sağlar.
3. Kapsamlı Anomali Tespiti: BiLSTM, hem geçmiş hem de gelecekteki verilerden faydalandığı için anomalileri tespit etme konusunda etkilidir.
4. Çift Yönlü Tahmin Kapasitesi: Zaman serilerindeki gelecekteki olayları tahmin etme veya eksik veriyi tamamlama gibi işlemlerde BiLSTM üstün performans gösterir.

3.4.1. Otokodlayıcılar ile öznitelik çıkarma

ANN algoritmalarının performansı, özellik uzayındaki birden fazla boyutla gerçek zamanlı bir durumda çalışırken önemli ölçüde etkilenebilir. Büyük veri setleriyle çalışırken, boyut indirgeme veya öznitelik çıkarımı için birçok farklı yaklaşım kullanılmaktadır. Bu sayede, ANN algoritması daha hızlı öğrenebilirken, aynı zamanda model basitleştirilir ve anlaşılması kolaylaşır. Çalışmalarımızda öznitelik çıkarımı için kullandığımız yaklaşım, LSTM-Otokodlayıcıdır.

Otokodlayıcı, ham girdi verisinin sıkıştırılmış bir versiyonunu öğrenebilen bir tür sinir ağıdır. Otokodlayıcı modelinde girdi verisinin sıkıştırılması görevi kodlayıcı tarafından yerine getirilirken, girdinin yeniden yapılandırılması çözücü tarafından yerine getirilir. Çözücü bu esnada sıkıştırılmış girdiyi kullanır. Model eğitimi tamamlandığında sıkıştırılmış girdi tüm girdi verisini temsil etme özelliğine sahip hale getirilmiş olur. Model eğitildikten sonra kodlayıcı modeli korunur, ancak çözücü kaldırılır. Kodlayıcı, ham veriyi başka bir makine öğrenmesi modelini eğitmek için hazırlamak amacıyla kullanıldığında öznitelik çıkarma işlemi gerçekleşmiş olur. Ortaya çıkan yeni öznitelikler girdi verisini ifade eden özniteliklerden farklı olarak onların bilinmeyen birleşimleridir (Şekil 3.12).



Şekil 3.12. Öznitelik çıkarımı

LSTM-Otokodlayıcı modelinde, kodlayıcı kurulduktan sonra, veri görselleştirmelerinde veya denetimli öğrenme modellerinin eğitilmesinde kullanılmak üzere veri dizilerinin kodlanması veya sıkıştırılması işlemi yapılabilir. LSTM-Otokodlayıcı modeli kodlayıcı-çözümleyici yapısını kullanarak giriş dizilerini, herhangi bir diziler kümesi için çoğaltılabilecek bir çıkış dizisine dönüştürmektedir. Modelin, giriş dizisini doğru bir şekilde tahmin etme yeteneği, performansını tanımlar. Kodlayıcı modeli, dizinin yeniden üretme görevini yerine getirebildiği sürece uzun süre kullanılabilir. Bu yaklaşım, giriş dizilerinden sabit uzunlukta vektörler üretmek için de uygulanabilir. Bu vektörler, başka bir denetimli öğrenme modeline giriş olarak kullanılabilir ve dizilerin sıkıştırılmış bir temsili oluşturulması gibi birçok farklı amaç için daha ileri düzeyde kullanılabilir.

4. YÖNTEM

Tarihsel olarak, quadrotorlerin arıza teşhisinde büyük ölçüde insan uzmanlığına dayalı yöntemler kullanılmıştır. Ancak, yapay zekâ algoritmalarının gelişmesiyle birlikte, akıllı arıza teşhisi (intelligent fault diagnosis) adı verilen sistemler sayesinde quadrotorlar artık kendi arıza koşullarını otomatik olarak tanımlama, sınıflandırma ve hatta önceden tahmin etme yeteneğine kavuşmuştur. Bu durum, yalnızca insan müdahalesine olan ihtiyacı azaltmakla kalmayıp, aynı zamanda quadrotor operatörlerinin araç sağlığını proaktif olarak yönetmesine imkân tanır ve güvenli ve kesintisiz operasyonlar için gerekli önlemlerin zamanında alınmasını sağlar.

Quadrotorlar gibi karmaşık sistemlerde arıza tespiti, arızaların anlık olarak tanınmasını ve o an meydana gelen bileşenlere yönelik teşhis süreçlerini içerir. Ancak, literatürdeki önceki çalışmalar genellikle yalnızca tek bir bileşen veya birkaç bileşen için arıza tespitine odaklanmış ve arıza anında teşhis yapmayı hedeflemiştir. Bu durum, gelecekte meydana gelebilecek arızaların erken teşhisi için yeterli bir yaklaşım sunmamaktadır. Özellikle, uçuş sırasında kapsamlı verilerden faydalanarak zaman içinde ortaya çıkabilecek potansiyel arızaların önceden tahmin edilmesi büyük bir gereklilik hâline gelmiştir.

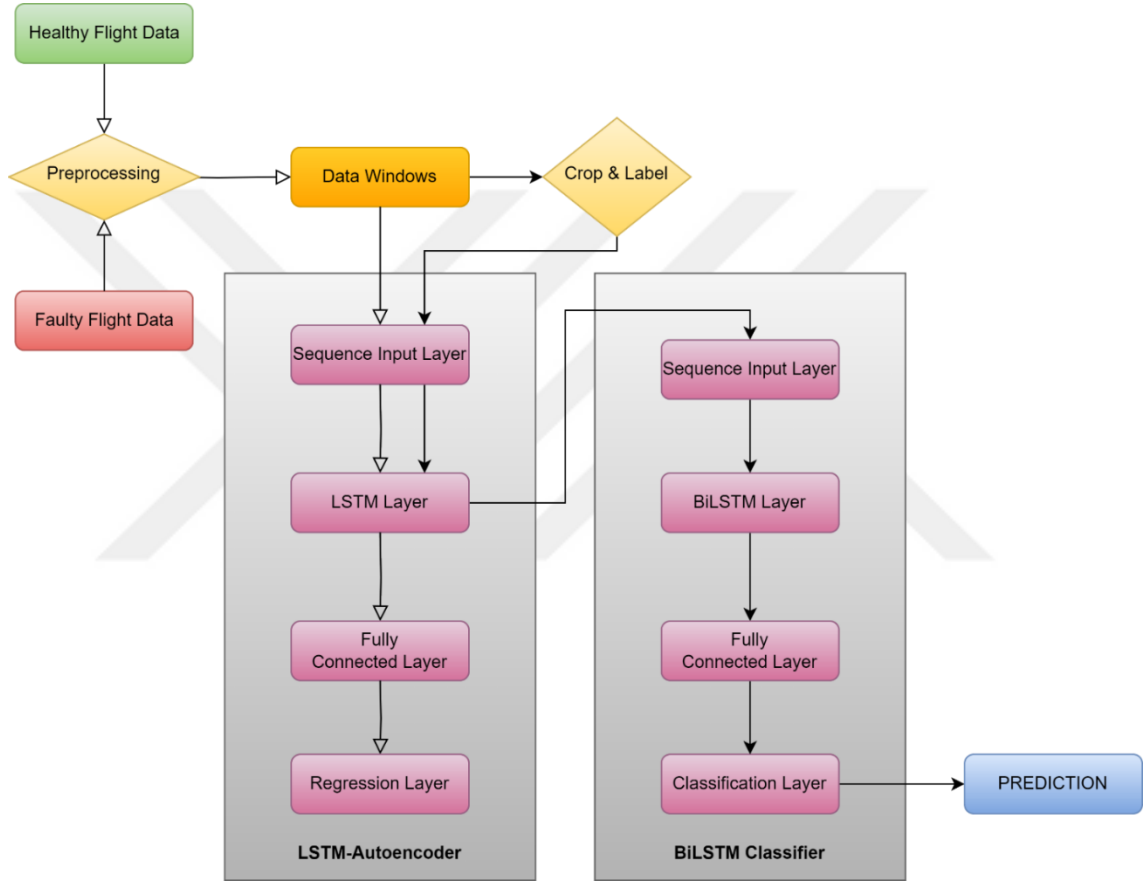
Bu çalışmanın temel amacı, çok rotorlu hava araçlarının (quadrotorların) uçuş süreçlerinde meydana gelebilecek arızaları erken safhada tespit etmek ve bu sayede daha güvenli bir uçuş deneyimi sağlamaktır. Buna yönelik olarak düşme tahmini ve uçuş sağlığı tahmini başlıklı iki farklı yöntem geliştirilmiştir. Her iki modelin tasarımı, bu modellerin ardındaki teorik çerçeve ve uygulama detayları aşağıda detaylıca ele alınmıştır.

4.1. Düşme Tahmini

Düşme tahmini modeli, bir quadrotor'un kontrolsüz bir düşüşe geçmeden önceki belli bir süreçteki davranışını analiz ederek bu sürecin başında bir arıza belirtisi olup olmadığını tespit etmeyi amaçlar. Bu model, quadrotorun uçuş durumunu "düşecek" veya "düşmeyecek" olarak iki sınıfa ayırarak çift sınıflı bir sınıflandırma problemi çözer.

Düşme tahminine yönelik bu çalışmanın katkılarından biri, arızaların kontrolsüz düşüşün öncesindeki ilk belirtilerini erken safhada yakalayarak, pilot veya otonom kontrol sisteminin gerekli önlemleri zamanında alabilmesine imkân sağlamasıdır. Bu amaçla geliştirilen modelde BiLSTM tabanlı bir sinir ağı kullanılmıştır.

Bu çalışmada geliştirilen düşme tahmini stratejisi; veri hazırlığı, boyut indirgeme ve düşme tahmini için sınıflandırma olmak üzere üç temel aşamadan oluşmaktadır. Çalışmanın genel yapısının şematik gösterimi Şekil 4.1’de sunulmuştur. Her bir aşama, arızaların yalnızca anlık olarak tanınmasını değil, aynı zamanda zaman içinde ortaya çıkabilecek sorunların önceden tahmin edilmesini sağlamak üzere kurgulanmıştır. Böylece, quadrotorun arızalanmadan önce gerekli müdahalelerin yapılabilmesi hedeflenmektedir.



Şekil 4.1. Düşme tahmini stratejisi (Ozcan ve Perkgoz, 2024)

Veri Hazırlığı

Düşme tahmini sürecinin ilk aşaması, farklı uçuşlardan toplanan geniş miktarda sensör verisinin analiz edilerek ilgili modellerde kullanılmak üzere hazırlanmasıdır. Quadrotorlarda, hız, ivme, batarya durumu, sıcaklık gibi çok çeşitli parametrelerden elde edilen sensör verileri, verimli bir şekilde analiz edilmediğinde gürültülü ve gereksiz olabilmektedir. Bu nedenle, ham sensör verilerinin işlenmesi ve daha anlamlı bir yapıya dönüştürülmesi için uygun bir veri hazırlığı süreci gereklidir.

Veri hazırlığı aşamasında, gereksiz bilgilerden arındırılmış veriler, daha sonra boyut indirgeme yöntemine aktarılacak üzere düzenlenir. Farklı uçuş senaryolarından elde edilen büyük veri kümeleri, boyut indirgeme ve modelleme süreçlerinde kullanılacak şekilde temizlenir, gerekli normalizasyon işlemleri uygulanır ve kayan pencereler yöntemi ile veri segmentleri oluşturulur.

Kayan pencereler (sliding windows) yöntemi zaman serisi tahmininde ardışık verileri sabit uzunlukta girdi-çıkı çiftlerine yapılandıran bir tekniktir. Bu yöntemde tüm zaman serisi tek bir dizi olarak ele alınmaz. Bunun yerine, girdi olarak kullanılacak verilerden ve karşılık gelen çıktı veya hedef değerlerden daha küçük, örtüşen pencereler oluşturur. Bu şekilde veri seti, üzerinde çeşitli makine öğrenmesi algoritmalarının koşturulmasına uygun hale getirilmiş olur. Pencerelerin nasıl oluşturulduğuna yönelik temsili bir görsel Şekil 4.2’de verilmiştir.

	col_1	col_2	col_3	col_4	col_5
row_1	53	15	73	59	75
row_2	76	31	18	21	51
row_3	50	27	76	99	34
row_4	61	46	13	37	74
row_5	50	99	13	68	79
row_6	18	59	41	32	42
row_7	46	91	66	23	20
row_8	79	20	81	23	54
row_9	10	43	13	97	13
row_10	85	76	96	38	69

Şekil 4.2. Kayan pencereler yöntemi (<http-9>)

Kullandığımız uçuş verilerini şekle göre yorumladığımızda, her bir satır sensörlerden okunan bir özelliğe karşılık gelirken, her bir sütun ise anlık zamanı ifade eder. Dolayısıyla burada her bir pencere, uçuşa ait belli bir zaman aralığındaki sensör değerlerini belirtir. Pencere sağa doğru kaydırıldığında zamanda ilerleme gerçekleşmiş olur ve sensör değerleri farklılaşır. Böylelikle uçuş karakteristiğinin zamanla nasıl değiştiğine dair anlam çıkarılması mümkün olur.

Kaydırma penceresi yönteminin önemli bir yönü, pencere boyutu ve kaydırma miktarının yapılandırılmasındaki esnekliğidir. Pencere boyutu, modelin tahmin yapmak için ne kadarlık bir seriyi kullanacağını belirler. Çok küçük bir pencere uzun vadeli desenlerin yakalanamamasına sebep olabilir. Diğer taraftan çok büyük bir pencere ise beraberinde gürültü veya hesaplama yükü getirebilir. Dolayısıyla pratik uygulamalarda dengenin sağlanması gereklidir.

Pencerenin kaydırılma miktarı öğrenmede kullanılacak örneklem sayısını doğrudan değiştirir. Kaydırma miktarı 1 olarak ayarlandığında eğitim örneklerinin sayısı maksimuma ulaşır. Ancak aynı zamanda art arda alınan iki pencerenin birbirleri arasındaki veri kesişimi de maksimum hale gelmiş olur. Bu durum tekrarlı verilerin çok fazla kullanılmasına sebep olur. Bu sorundan kurtulmak için daha büyük adımlar kullanılabilir; bu durumda, pencere kaydırıldığında daha az örtüşme olur ve veri tekrarı azaltılmış olur.

Boyut İndirgeme

Bu çalışmada, büyük veri kümesinden elde edilen bilgilerin daha kompakt bir hale getirilmesi ve gereksiz verilerin elenmesi amacıyla LSTM tabanlı bir otokodlayıcı eğitilmiştir. Otokodlayıcı, verilerin gizli katmanlardan geçirilmesiyle, yalnızca daha anlamlı ve yararlı bilgileri öne çıkarır. Böylece boyut indirgeme işlemi gerçekleştirilir.

Otokodlayıcı yapısında, sensör verileri, öncelikle LSTM katmanlarında işlenerek zamansal bağımlılıkları yakalar. LSTM-Otokodlayıcının kullanımı, yalnızca verilerin boyutunu azaltmakla kalmaz, aynı zamanda gereksiz bilgilerin ortadan kaldırılmasını ve arıza tahmininde kullanılacak önemli özelliklerin korunmasını sağlar. Bu şekilde boyutları azaltılmış ve daha anlamlı hale getirilmiş veriler, sonraki aşamada kullanılmak üzere hazırlanmış olur.

Düşme Tahmini için Sınıflandırma

Boyut indirgeme aşamasından elde edilen şifrelenmiş (encoded) veriler, çalışmanın en kritik aşaması olan arıza tahmini için kullanılır. Burada, boyutları azaltılmış ancak bilgi açısından zengin veriler, BiLSTM tabanlı bir sınıflandırma algoritmasına aktarılır. BiLSTM, zaman serisi verilerindeki geçmiş ve gelecek bağımlılıkları aynı anda işleyerek, olası arızaların erken teşhis edilmesine olanak tanır.

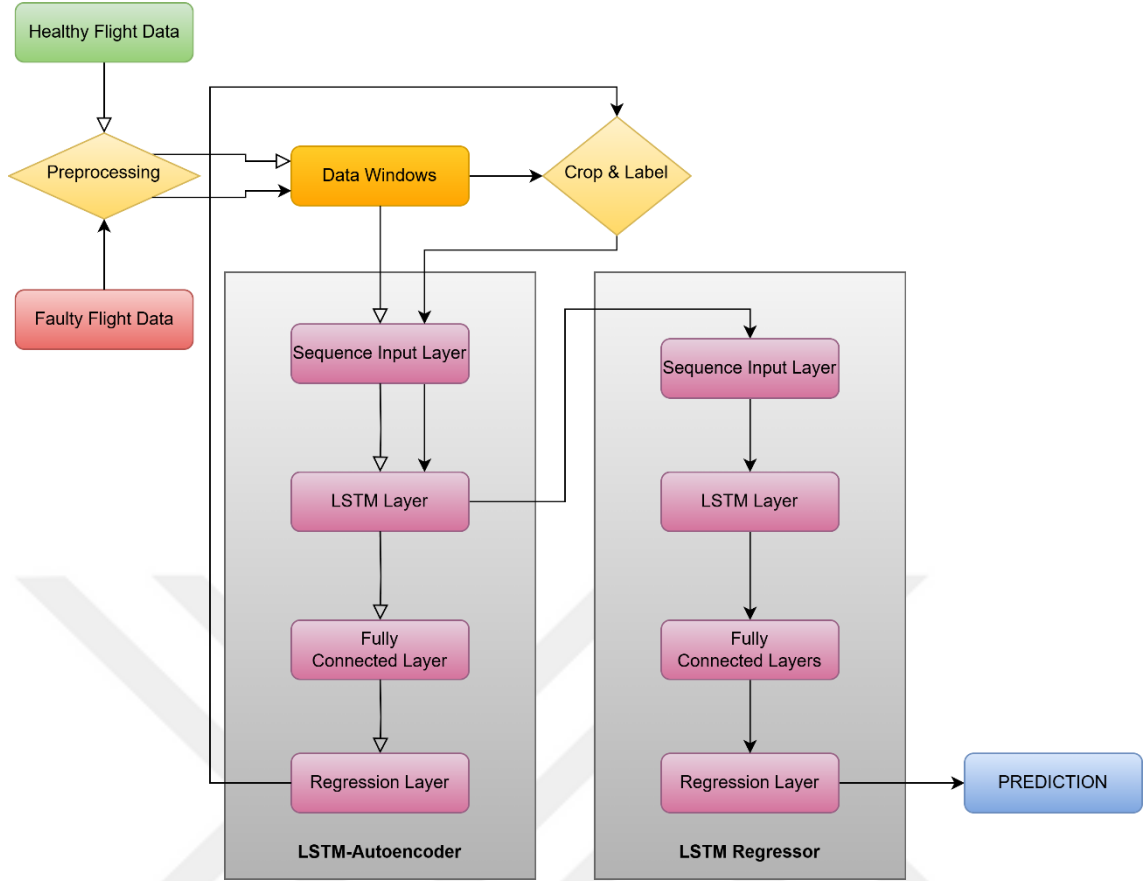
BiLSTM yapısının avantajı hem geçmiş hem de geleceğe yönelik verilerden öğrenerek çift yönlü veri akışı sağlamasıdır. Bu, özellikle zaman serisi sensör verilerinde

önemli bir fark yaratır, çünkü arızaların yalnızca geçmişe değil, geleceğe bağlı olarak da ortaya çıkabileceği göz önünde bulundurulmalıdır.

BiLSTM modeli, veri akışını ileri ve geri yönde işleyerek, her iki yönden elde edilen bilgileri tek bir çıktıda birleştirir. Böylece, potansiyel arızalar yalnızca meydana geldikten sonra değil, oluşmadan önce tahmin edilir ve gerekli önlemler alınabilir. Bu aşamada kullanılan geri yayılım ve optimizasyon algoritmaları, modelin ağırlıklarını sürekli olarak güncelleyerek en doğru tahminleri yapmasını sağlar.

4.2. Uçuş Sağlığı Tahmini

Uçuş sağlığı tahmini, bir uçuş sürecinin zamanla nasıl performans kaybına uğradığını modellemek ve uçuşun sağlıklı bir durumdan anormal bir duruma geçişini [0-1] aralığında sürekli bir değerle ifade etmek için geliştirilmiştir. Bu değer uçuş anomali skoru olarak isimlendirilmiştir. Geliştirdiğimiz uçuş sağlığı tahmini modeli, quadrotorun uçuş performansının zamana bağlı olarak nasıl değiştiğini analiz ederek, uçuşun belirli bir anda ne kadar stabil olduğunu ya da ne kadar bozulmaya yaklaştığını bu değere bağlı olarak kantitatif bir şekilde ifade etmeyi amaçlar. Hesaplanan uçuş anomali skorunda çıkan daha yüksek değerler, uçuş stabilitesinin bozulduğuna işaret etmektedir. Uçuş sağlığı tahmini, düşme tahminine benzer şekilde veri hazırlığı, boyut indirgeme ve uçuş sağlığı tahmini olmak üzere üç aşamadan oluşur. Şekil 4.2 uçuş sağlığı tahmini stratejisinin genel işleyişini göstermektedir.



Şekil 4.3. Uçuş sağlığı tahmini stratejisi

Veri Hazırlığı

Quadrotor uçuşlarından elde edilen log kayıtları, kaydın başladığı andan bitirildiği ana kadar olan zaman içerisinde ilgili sensörlerden ölçülen akım, sıcaklık, basınç, ivme gibi değerleri barındırmaktadır. Veri hazırlığı sürecinde ilk iş, bu kayıtlardaki gereksiz bilgilerin arındırılması, boyut indirgeme yöntemine uygulanmak üzere verinin hazır hale getirilmesidir. Bu aşamada kayıtlardan sadece uçuş sürecindeki veriler toplanır. Aynı zamanda quadrotorda düşüş başladıktan sonra stabilizasyon bozulmuş olur ve sensörlerden okunan veriler kontrolden çıkar. Bu sırada elde edilen verilerin model eğitiminde kullanılması anlamsız olacağından veri temizliği aşamasında çıkarılmıştır. Sonrasında verilerin normalizasyonu gerçekleştirilir ve kayan pencereler yöntemine göre veri segmentleri oluşturulur.

Uçuş sağlığı tahmini için etiketleme, veri segmentlerinin eğitime sokulmadan önce anomali skorlarının belirlenmesi işlemidir. Bu işlem otokodlayıcının yalnızca sağlıklı verilerle eğitilmesi sayesinde, anormal uçuş durumlarını tanımlamak için yeniden yapılandırma RMSE hatasının kullanılmasıyla gerçekleştirilir. Yüksek RMSE,

otokodlayıcının uçuş verilerini doğru bir şekilde modelleyemediğini ve bunun uçuş sağlığının azaldığını gösterir. Bu hatalar kullanılarak uçuş veri segmentleri, 0-1 aralığında sürekli bir değerle etiketlenir. Etiketleme adımları şu şekildedir:

1. Otokodlayıcı Hata Hesaplama: Uzman tecrübesiyle belirlenmiş öznitelikler seçilir. Seçilmiş öznitelikler üzerinden her bir veri segmenti için otokodlayıcı modeli tarafından yeniden oluşturma RMSE hatası hesaplanır. RMSE değeri, segmentin normallik derecesini anlamak için temel ölçüttür.

2. Eşik Değeri Kontrolü: Eğer bir segmentin RMSE değeri, önceden belirlenen bir eşik değerden büyükse, bu segmentle birlikte uçuşun anomali skoru hata oranında artırılır. Eşik değeri, modelin hassasiyetine göre deneysel olarak belirlenmiştir.

3. Kümülatif Hata Hesabı ve Etiket Atama: Anormal segmentlerin RMSE değerleri birikimli bir toplam etiket değeri olarak kaydedilir. Her segment, toplam hata birikimiyle etiketlenir. Sağlıklı segmentler için etiket değeri düşük, anormal segmentler için ise daha yüksek olur.

4. Normalizasyon: Tüm etiket değerleri, toplam hata değeri üzerinden normalleştirilir. Bu normalizasyon, uçuş sağlığı değerlerini 0 ile 1 arasında bir sürekli aralıkta ifade etmeyi sağlar.

Boyut İndirgeme

LSTM-Otokodlayıcı, uçuş verilerini boyut indirgeme amacıyla kullanılmıştır. Otokodlayıcı, girdi verisini sıkıştırarak daha küçük bir boyutta temsil ederken, aynı zamanda bu veriden orijinal girdi verisini yeniden oluşturmayı öğrenir. Uçuş sağlığı tahmini yöntemi için Bölüm 3.3'te anlatılan süreç işletilmiş, gizli katmandaki nöron sayısı 20 olarak belirlenmiştir.

Uçuş Sağlığı Tahmini için Regresyon

Uçuş sağlığı tahmini için geliştirilen modelde, LSTM tabanlı bir yapay sinir ağı yapısı kullanılmıştır. Bu yapıda giriş katmanından sonra bir LSTM katmanı, 10 nöronlu tam bağlı bir katman ve tek nöronlu çıkış katmanı eklenmiştir. Modelin temel mantığı, uçuş süresince elde edilen sensör verilerindeki zamansal örüntüleri analiz ederek, uçuş anomali skorunu doğru bir şekilde tahmin etmektir. Uçuş anında zamanla değişen bu skor, uçuş esnasındaki anlık durumların genel bir özeti olarak düşünülebilir ve sistemin stabilitesindeki olası bozulmaları öngörmek için kullanılır. Bu yapıda, LSTM katmanı, zamansal verilerin örüntülerini yakalarken, regresyon katmanı, uçuş sağlığını temsil eden

sürekli bir değer üretir. Çıkış katmanı, uçuş süresinin her bir segmenti için bir skor tahmin eder ve bu skorlar, uçuşun genel stabilitesi hakkında bilgi sağlar.

Bu yaklaşım, uçuş sağlığı değerlendirme sürecine önemli katkılar sunmaktadır. Özellikle uçuş sırasında meydana gelen bozulmaların erken tespiti, hem pilotların hem de otonom sistemlerin uçuş güvenliğini sağlamak için zamanında müdahale edebilmesine olanak tanır. Modelin performans değerlendirmesi, RMSE metriği kullanılarak yapılmış ve yüksek doğrulukta tahmin sonuçları elde edilmiştir. Model, uçuş güvenliği açısından yenilikçi bir yaklaşım sunmakla birlikte gelecekte farklı tipteki hava araçları ve farklı uçuş koşulları için de genelleştirilebilir ve uyarlanabilir bir yapı taşımaktadır.

4.3. Değerlendirme Ölçütleri

Sunulan modellerin performanslarını ölçmek ve değerlendirmek üzere bir takım değerlendirme ölçütü kullanılmıştır. Sınıflandırma performansının değerlendirmesinde doğruluk, kesinlik, duyarlılık ve f-skoru kullanılmıştır. Regresyon performansı değerlendirmelerinde ise MAE, RMSE ve standart sapma ölçütlerinden faydalanılmıştır.

Doğruluk: Doğru tahmin edilen örneklerin toplam örneklere oranıdır ve aşağıdaki formülle hesaplanır:

$$Doğruluk = \frac{TP+TN}{TP+TN+FP+FN} \times 100 \quad (4.1)$$

Burada:

TP (Doğru Pozitifler): Doğru şekilde arızalı olarak tahmin edilen uçuşların sayısı.

TN (Doğru Negatifler): Doğru şekilde sağlıklı olarak tahmin edilen uçuşların sayısı.

FP (Yanlış Pozitifler): Yanlışlıkla arızalı olarak tahmin edilen sağlıklı uçuşların sayısı.

FN (Yanlış Negatifler): Yanlışlıkla sağlıklı olarak tahmin edilen arızalı uçuşların sayısı.

Kesinlik: Doğru şekilde pozitif (arızalı) olarak tahmin edilen gözlemlerin, toplamda tahmin edilen pozitifler içindeki oranını gösterir. Özellikle yanlış pozitiflerin yüksek maliyet oluşturduğu senaryolarda önemlidir. Aşağıdaki formülle hesaplanır:

$$Kesinlik = \frac{TP}{TP+FP} \quad (4.2)$$

Duyarlılık: Doğru şekilde pozitif olarak tahmin edilen gözlemlerin, gerçek pozitif sınıftaki tüm gözlemlere oranıdır. Arıza tespitinde kritik bir rol oynar, çünkü mümkün olduğunca fazla gerçek arızanın tespit edilmesini sağlar. Aşağıdaki gibi tanımlanır:

$$Duyarlılık = \frac{TP}{TP+FN} \quad (4.3)$$

F1 Skoru: Kesinlik ve duyarlılığın ağırlıklı ortalamasıdır ve her ikisi arasında bir denge sağlar. Sınıfların dengesiz olduğu durumlarda özellikle kullanışlıdır, örneğin arızaların sağlıklı duruma kıyasla nadir olduğu durumlar gibi. Aşağıdaki formülle hesaplanır:

$$F1 Skoru = 2 \times \frac{kesinlik \times duyarlılık}{kesinlik + duyarlılık} \quad (4.4)$$

MAE: Bir modelin tahmin ettiği değerlerle gerçek değerler arasındaki mutlak farkların ortalamasını veren bir hata ölçütüdür. Anomali skoru tahmini hatalarını ölçmek için kullanılmıştır. Aşağıdaki formülle hesaplanır:

$$MAE = \frac{1}{n} \times \sum_{i=1}^n |y'_i - y_i| \quad (4.5)$$

Burada y'_i ve y_i terimleri sırasıyla tahmin edilen anomali skoru ve gerçek anomali skorunu ifade ederken n terimi ise tahmin edilen toplam veri segmenti sayısını ifade eder.

RMSE: Bir modelin tahmin ettiği değerlerle gerçek değerler arasındaki kareli farkların ortalamasının karekökünü alarak hesaplanan bir hata ölçütüdür. Anomali skoru tahmini modelinin hata hesaplamasında kullanılmıştır. MAE formülünde kullanılan terimler burada da kullanılmaktadır ancak bu ölçütte yüksek hatalar MAE ölçütüne kıyasla daha fazla cezalandırılır. Aşağıdaki formülle hesaplanır:

$$RMSE = \sqrt{\frac{\sum_{i=1}^n (y'_i - y_i)^2}{n}} \quad (4.6)$$

Standart Sapma: Bir veri kümesindeki değerlerin ortalamadan ne kadar saptığını ölçen, verilerin yayılma derecesini gösteren bir istatistiksel ölçüdür. Anomali skoru tahminini yapılırken katmanlarda ölçülen hataların standart sapmasının ölçülmesiyle modelin tutarlılığı gözlemlenmiştir. Formülü aşağıdaki şekilde ifade edilir:

$$Standart Sapma = \sqrt{\frac{\sum_{i=1}^n (x_i - \mu)^2}{n}} \quad (4.7)$$

Burada x_i , tahminin hata deęerini ifade ederken μ hesaplanan tahmin hatalarının ortalamasını, n ise standart sapması hesaplanacak hata deęerlerinin toplam sayısını ifade eder.



5. DENEY SONUÇLARI

5.1. Veri Toplama

Quadrotorlar ile ilgili kazalar, operatörlerin yaşadıkları deneyimleri çevrimiçi bloglar aracılığıyla paylaşımlarıyla sıklıkla gündeme gelmektedir. Bu paylaşımlar, benzer zorluklarla karşılaşan kullanıcılar arasında bilgi alışverişine imkân tanır ve bu tür bilgiler, arıza tespitine yönelik araştırmalar için değerli bir veri kaynağı olarak öne çıkar. Bu çalışmada, kazalara neden olan belirsiz sebeplerin aydınlatılmasına yönelik veri toplama süreci sistematik bir şekilde gerçekleştirilmiş ve analiz edilmiştir.

5.1.1. Veri setinin tanımı

Veri toplama sürecinin birinci aşamasında, çevrimiçi bloglarda paylaşılan quadrotor kazalarına ilişkin bilgiler incelenmiştir. Özellikle, kazaların kesin nedeninin belirlenemediği durumlara odaklanılmıştır. Bu kapsamda, toplamda 10 uçuşa ait veri toplanmıştır. Bu uçuşlar, quadrotorların kontrolsüz bir şekilde düşmeye başladığı noktadan sonraki verileri içermeyecek şekilde düzenlenmiştir. Bu sayede, analizde yalnızca kazaya sebep olabilecek uçuş süresi boyunca toplanan sensör verileri kullanılmıştır.

İkinci aşamada, kampüs içerisinde quadrotor deneyleri gerçekleştiren bir ekip ile iş birliği yapılmıştır. Bu iş birliği kapsamında, sağlıklı (kazasız) 10 uçuşa ait veri toplanmıştır. Bu uçuşlar, quadrotorların arızasız ve stabil bir şekilde çalıştığı koşullarda gerçekleştirilmiş olup, sensör verileri doğru arıza tahmini için bir referans olarak kullanılmıştır.

Bu iki veri kaynağı birleştirilerek toplamda 20 uçuşluk bir veri seti oluşturulmuştur. Veri seti şu şekilde sınıflandırılmıştır:

- Arızalı Uçuşlar (10 adet): Çevrimiçi bloglardan toplanan ve kazayla sonuçlanan uçuşlardan oluşur. Bu uçuşların toplam süresi 126 dakika 16 saniye olarak kaydedilmiştir.
- Sağlıklı Uçuşlar (10 adet): Kampüs deneylerinden elde edilen ve kazasız tamamlanan uçuşlardan oluşur. Bu uçuşların toplam süresi ise 70 dakika 25 saniye olarak ölçülmüştür.

Quadrotorların uçuşları sırasında kaydedilen log dosyaları, çeşitli sensörlerden alınan ölçümleri içerir. Ancak, bu tür geniş ve kapsamlı veriler, arıza tespit sürecini karmaşıktırabilecek gereksiz, alakasız veya tekrarlayan bilgiler içerebilir. Bu nedenle,

erken arıza tespitinde anlamlı bilgiler sunabilecek özelliklerin dikkatlice seçilmesi önemli bir adım olarak gerçekleştirilmiştir.

Uçuş kayıtlarının incelenmesi sırasında UAV Log Viewer isimli açık kaynak kodlu bir web uygulamasından yararlanılmıştır. Uygulamaya dair bir ekran görüntüsü Şekil 5.1’de paylaşılmıştır. Uygulama üzerinden uçuşlara ait log kayıtlarının incelenmesi yapılabilmekte ve sensörlerden okunan verilerin zamanla değişimleri grafik üzerinden çoklu şekilde incelenebilmektedir. Grafik üzerinde zamanla ilerlerken aynı zamanda hava aracının uçuşu ayrı bir ekranda simüle edilebilmekte ve aracın yaptığı hareketler eş zamanlı olarak gözlemlenebilmektedir.



Şekil 5.1. UAV Log Viewer

Log dosyalarındaki sensör verileri üzerinde yapılan filtreleme ve düzenleme işlemleri ile uçuş sağlığı ve arıza tespiti açısından kritik bilgiler öne çıkarılmıştır. Bu süreçte, yalnızca faydalı olduğu düşünülen özellikler seçilmiş ve gereksiz bilgiler veri setinden çıkarılmıştır.

Veri hazırlama sürecinin sonucunda, farklı sensörlerden elde edilen toplamda 50 özellik öncelikli olarak belirlenmiştir. Verisetine alınan bu özelliklerin neler olduğu, birimleri ve açıklamaları Ek Tablo-1’de verilmiştir. Bu özellikler, quadrotor uçuşlarının genel sağlığını ve performansını değerlendirebilmek için geniş bir parametre aralığını kapsamaktadır. Belirlenen özellikler şu temel kategorilerde toplanmıştır:

Tahmini Konumlandırma Bilgileri: Roll, pitch, yaw (yönelim açıları), enlem ve boylam koordinatları.

Araç Performans Ölçütleri: Jiroskop sürüklenme hatası, irtifa, tırmanma hızı.

Kontrolcü Parametreleri: Kontrol kazançları, servo ve throttle giriş/çıkışları.

Batarya Durumu: Voltaj ve şarj durumu.

Hız Bileşenleri: X, Y, Z eksenlerindeki hız verileri.

İvme Ölçümleri: İvmeölçerden (akselerometre) elde edilen veriler.

Seçilen bu 50 özellik, hem gerçek dünyadaki kaza raporlarından türetilen bilgilerden hem de kampüs deneylerinden elde edilen sensör verilerinden oluşmaktadır. Böylece, veri seti yalnızca geniş kapsamlı değil, aynı zamanda gerçek dünya senaryolarıyla uyumlu olacak şekilde zenginleştirilmiştir.

5.1.2. Normalizasyon

ANN için doğrudan farklı dağılımlara sahip veri setleri sağlandığında, bu durum ağın giriş verilerinin ölçeği ve dağılımına duyarlılığı nedeniyle hata oranlarının artmasına ve modelin daha yavaş bir şekilde yakınsamasına yol açabilir. Bu sorun, girdilerin ölçeklerinin farklı olması nedeniyle modelin belirli özellikleri diğerlerine göre daha fazla dikkate almasına neden olabilir. Bu tür problemlerin önüne geçmek ve modelin daha dengeli bir şekilde öğrenmesini sağlamak için, veri setinin eğitimden önce bir normalizasyon işlemine tabi tutulması gerekmektedir.

Normalizasyon, her bir özelliğin (bu çalışmada sensör verileri) benzer ölçek ve dağılıma sahip olacak şekilde dönüştürülmesi işlemidir. Bu dönüşüm, ANN'nin daha hızlı ve etkili bir şekilde öğrenmesini sağlamak için kritik bir adımdır. Çalışmamızda kullanılan iki normalizasyon yöntemi şunlardır:

- Z-Skor Normalizasyonu
- Min-Max Normalizasyonu

5.1.2.1. Z-skor normalizasyonu

Z-skor normalizasyonu, bir veri noktasının ortalamadan ne kadar uzakta olduğunu standart sapma cinsinden ifade eden bir dönüşüm yöntemidir. Bu işlem, aşağıdaki matematiksel formülle gerçekleştirilir:

$$Z = \frac{x - \mu}{\sigma} \quad (5.1)$$

Bu denklemde:

x , veri noktasını,

μ , veri setinin ortalamasını,

σ , veri setinin standart sapmasını ifade eder.

Bu dönüşüm, her bir sensör verisinin veri setindeki diğer verilerle olan ilişkisini, veri noktalarının veri kümesi içerisindeki bağlamını göz önünde bulundurarak standart bir ölçekte yeniden ifade eder. Böylece, farklı büyüklüklere sahip olan özelliklerin ağırlıkları eşitlenir ve ANN'nin öğrenme sürecinde bu tür farklılıkların neden olabileceği yanlılık (bias) etkisi ortadan kaldırılır.

5.1.2.2. Min-max normalizasyonu

Min-max normalizasyonu ise veriyi belirli bir aralığa, genellikle $[0, 1]$ aralığına, yeniden ölçeklendiren bir yöntemdir. Bu yöntem, her bir veri noktasının, veri setindeki en küçük ve en büyük değerler arasındaki konumunu ifade edecek şekilde dönüştürülmesini sağlar. Min-max normalizasyonunun matematiksel ifadesi şu şekildedir:

$$x' = \frac{x - x_{min}}{x_{max} - x_{min}} \quad (5.2)$$

Bu denklemde:

x , normalizasyon uygulanacak veri değerini,

x_{min} , veri setindeki en küçük değeri,

x_{max} , veri setindeki en büyük değeri,

x' , $[0, 1]$ aralığına ölçeklendirilmiş veri değerini temsil eder.

Verilerin belirli bir aralığa ölçeklendirilmesi, ANN'de kullanılan ağırlıkların daha istikrarlı bir şekilde güncellenmesini sağlar. $[0, 1]$ aralığında yer alan veriler, optimizasyon algoritmalarının daha hızlı çalışmasına katkıda bulunur. Özellikler arası dengelenmeyle büyük değerlerin daha baskın olması engellenir ve tüm özellikler model tarafından eşit önemde değerlendirilebilir. Böylelikle farklı büyüklükteki sensör ölçümleri arasında uyum sağlanır ve bu ölçümlerin öğrenme sürecine eşit katkıda bulunmasını temin edilmiş olur.

5.2. Verilerin Parçalara Ayrılması: Kayan Pencere Yönteminin Uygulanışı

Normalizasyon işlemi tamamlandıktan sonra, veri üzerinde daha ayrıntılı analiz yapılabilmesi ve zaman serisi tabanlı algoritmaların etkin bir şekilde eğitilmesi için veriler kayan pencere yöntemi kullanılarak küçük segmentlere ayrılmıştır. Bu yöntemde,

ham veriler belirli bir pencere boyutuna göre bölünmekte ve bu pencereler belirli bir kaydırma oranıyla ilerletilmektedir. Böylece, her bir pencere verinin yalnızca bir kısmını içerse de, pencereler arasında örtüşme oluşturularak zaman serisinin sürekliliği korunmaktadır.

Kayan pencere yönteminde kritik adımlardan biri, verinin en uygun boyutlarda bölünmesini sağlamaktır. Bu çalışmada farklı pencere boyutları üzerinde denemeler yapılmış ve sonuçlar doğrultusunda pencere boyutu 10 veri noktası olarak belirlenmiştir. Bu, 10 Hz örnekleme oranına sahip bir sistemde, pencere boyutunun 1 saniyelik veriyi temsil etmesi anlamına gelir. Bu seçim, sistemin zaman serisi tabanlı özelliklerini kaybetmeden yeterli bilgi yoğunluğunu sağlayacak şekilde yapılmıştır.

Pencere boyutuna ek olarak, pencerelerin birbirini ne kadar örtüştüreceği, yani kaydırma oranı, modelin performansını doğrudan etkileyen bir diğer önemli parametredir. Bu bağlamda, farklı kaydırma oranları test edilmiş ve en uygun değer 5 veri noktası olarak belirlenmiştir. Bu değer, 10 Hz örnekleme oranı için 0.5 saniyelik bir zaman dilimi anlamına gelmektedir. Böylece, ardışık pencereler arasında hem bilgi paylaşımı sağlanmış hem de veri setinin zaman boyutundaki sürekliliği korunmuştur.

Her bir pencere segmentinin içerdiği veri miktarı, pencere boyutunun sensör sayısı ile çarpılmasıyla hesaplanabilir. Çalışmada kullanılan 50 sensör özelliği göz önüne alındığında, her bir pencere segmenti şu şekilde yapılandırılmıştır:

$$\text{Veri Pencere Segmenti} = \text{Pencere Boyutu} \times \text{Sensör Sayısı} = 10 \times 50 = 500$$

Bu nedenle, her bir veri penceresi segmenti toplamda 500 değer içermektedir. Bu yapılandırma, modelin girdilerini optimize ederek hem hesaplama sürecini kolaylaştırmış hem de algoritmaların veri içindeki ilişkileri daha verimli bir şekilde öğrenmesine olanak tanımıştır.

5.3. LSTM-Otokodlayıcı

BiLSTM sınıflandırıcısı kullanarak arıza tespiti yapılırken, modelin zaman karmaşıklığını azaltmak ve gereksiz giriş özelliklerini elimine ederek daha hassas sonuçlar elde etmek adına boyut indirgeme işlemi uygulanmıştır. Bu bağlamda, LSTM-Otokodlayıcı yapısının gizli katmanından elde edilen çıktılar, BiLSTM sınıflandırıcı ağına giriş verisi olarak kullanılmıştır.

Bu yaklaşım, ilk olarak LSTM-Otokodlayıcı üzerinde deneyler gerçekleştirilerek başlatılmıştır. LSTM-Otokodlayıcı aşamasında, uçuş verilerinden elde edilen tüm ilgili

bilgilerin korunması hedeflenmiş ve verinin yeniden yapılandırılması sırasında hata minimizasyonu esas alınmıştır. Bu amaçla aşağıdaki bileşenlerden oluşan bir model yapısı geliştirilmiştir:

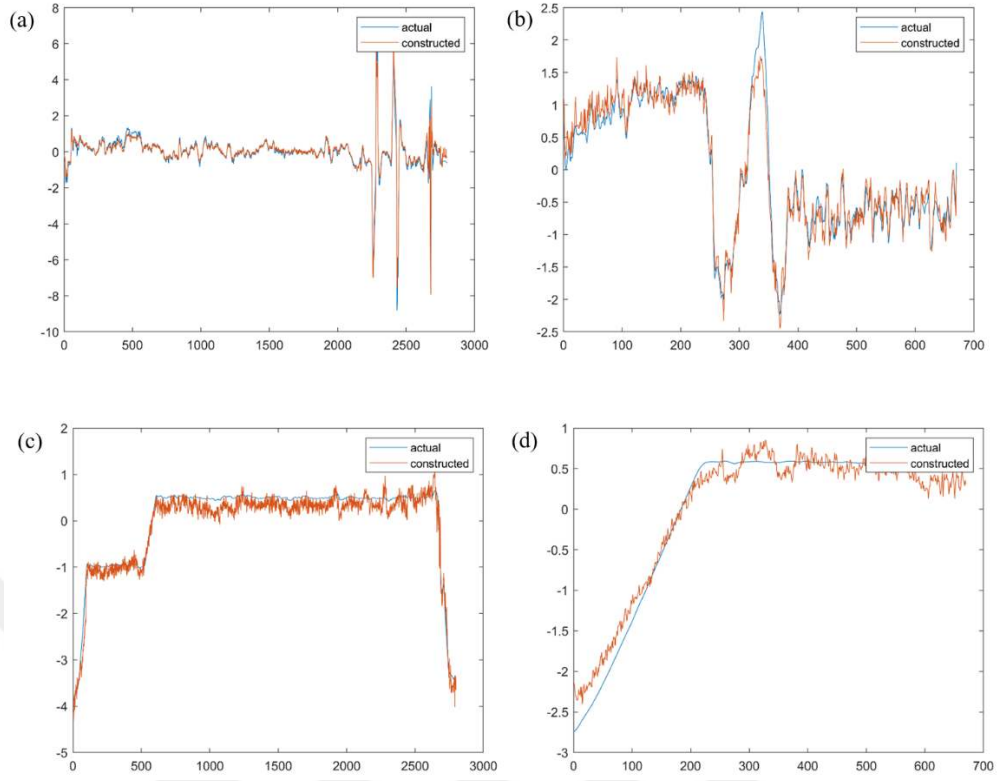
1. Dizi giriş katmanı
2. Sigmoid geçit aktivasyonu ve tanh durum aktivasyonuna sahip LSTM katmanı
3. Tam bağlantılı katman
4. Regresyon katmanı

Modelin eğitimi sırasında, uyarlamalı moment tahmini (ADAM) optimizasyon algoritması kullanılmış; öğrenme oranı 0,01 olarak belirlenmiştir. Eğitim, 50'lik mini veri grupları (batch size) ile 200 epoch boyunca gerçekleştirilmiştir. Kayıp fonksiyonu olarak Ortalama Kare Hata (MSE) kullanılmış ve L2 düzenleme eklenerek modelin genelleştirme yeteneği artırılmıştır.

Eğitim sürecinde hem sağlıklı hem de arızalı veriler kullanılmış, bunun yanı sıra tüm veri seti, %80 eğitim, %10 test, %10 doğrulama şeklinde üçe bölünmüştür. Bu bölünme, aşırı öğrenme problemini önlemek adına kritik bir adım olarak uygulanmıştır.

LSTM-Otokodlayıcı yapısında, gizli katmandaki nöron sayısının ağ performansı üzerindeki etkisini değerlendirmek amacıyla çeşitli deneyler gerçekleştirilmiştir. Bu süreçte, gizli katmandaki nöron sayısı 5'ten başlayarak 45'e kadar 5'er artışlarla yükseltilmiş ve toplamda dokuz farklı ağ yapısı oluşturulmuştur. Bu deneyler sonucunda, nöron sayısındaki artışın ağın yeniden yapılandırma performansı üzerindeki etkileri analiz edilmiştir.

Bu yöntem ve yapı sayesinde, arıza tespit modelinin veri dengesizliği sorununu minimize ederek daha hassas ve güvenilir sonuçlar elde etmesi sağlanmıştır. Çalışma, bu yaklaşımla quadrotor sistemlerinde erken arıza tespiti için önemli bir katkı sunmaktadır.



Şekil 5.2. Normalleştirilmiş gerçek sinyaller ve bunların, gizli katmanda 20 nöron bulunan LSTM-otokodlayıcı ile yeniden yapılandırılan temsilcileri. a) sağlıklı bir uçuşun yunuslama sinyali. b) arızalı bir uçuşun yunuslama sinyali. c) sağlıklı bir uçuşun irtifa sinyali. d) arızalı bir uçuşun irtifa sinyali. (Ozcan ve Perkgöz, 2024)

Şekil 5.2, ilgili sensörlerden elde edilen normalize edilmiş pitch ve irtifa sinyalleri için gerçek değerler ile otokodlayıcı tarafından yeniden yapılandırılan değerlerin grafiğini göstermektedir. Bu deneyde kullanılan otokodlayıcı, gizli katmanında 20 nöron içermektedir. Benzer şekilde, her bir sensör veri seti, gizli katmandaki nöron sayısı değiştirilerek incelenmiş ve uçuşlara ait RMSE değerleri hesaplanmıştır. Tüm veri seti için hesaplanan ortalama RMSE değeri 0.3898 olarak elde edilmiştir.

Elde edilen sonuçlar, kazalı uçuşlar ve sağlıklı uçuşların, özellikle gizli katmandaki nöron sayısının özellik sayısına yakın olduğu durumlarda LSTM-Otokodlayıcılar tarafından yüksek doğrulukla temsil edilebildiğini göstermektedir. Gizli katmandaki nöron sayısının artmasıyla birlikte, otokodlayıcının hata oranının azaldığı gözlemlenmiştir. Bu durum, boyut indirgeme sürecinde, nöron sayısının belirli bir eşik değerin üzerinde tutulması gerektiğini, aksi halde önemli bilgilerin kaybolabileceğini ortaya koymaktadır.

Bu bulgular, otokodlayıcı yapısının, uçuş verilerinin özellik uzayını hassas bir şekilde küçültme kapasitesine sahip olduğunu ve gizli katmandaki nöron sayısının doğru

seçilmesinin, modelin performansı üzerinde kritik bir etkiye sahip olduğunu göstermektedir. Böylelikle, boyut indirgeme süreci, veri kaybını minimize ederek arıza tespit modellerinin etkinliğini artırmakta önemli bir rol oynamaktadır.

5.4. Düşme Tahmini

Bu bölümde BiLSTM sınıflandırıcının düşme tahminindeki başarısını değerlendirmek amacıyla bir dizi deney gerçekleştirilmiştir. Bu deneylerde kullanılan BiLSTM sınıflandırıcı ağı, şu mimari bileşenlerden oluşmaktadır: bir sıra giriş katmanı, 30 gizli birimli BiLSTM katmanı (durum aktivasyon fonksiyonu olarak tanh, kapılama fonksiyonu olarak sigmoid kullanılmıştır), bir tam bağlı katman ve bir sınıflandırma katmanı. Eğitim sürecinde ADAM optimizasyon algoritması kullanılmış olup, 0,01 öğrenme oranı, 50 batch boyutu ve maksimum 100 epoch değeri atanmıştır. Çapraz entropi kayıp fonksiyonu olarak tercih edilirken, aşırı öğrenmeyi önlemek için L2 düzenleme uygulanmıştır.

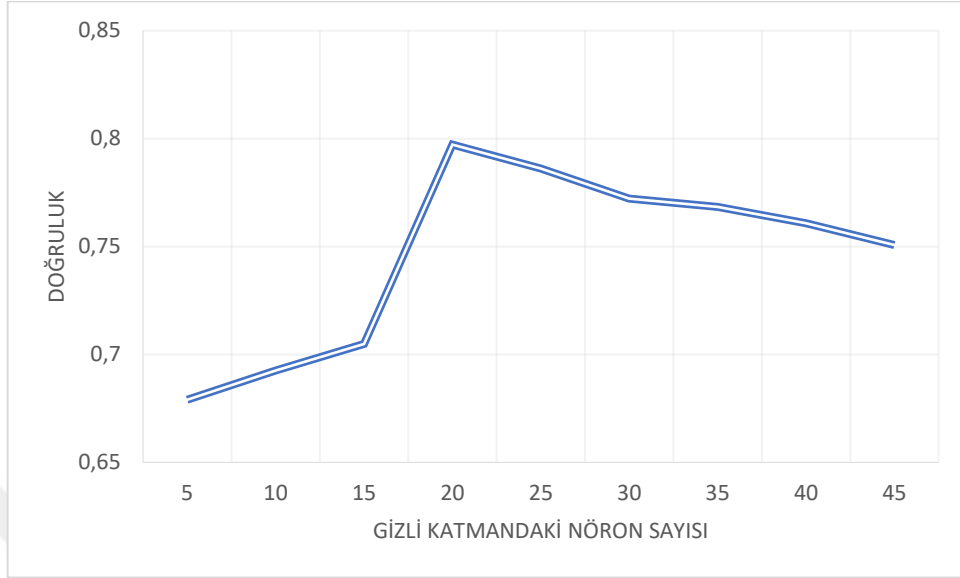
Boyut indirgeme için optimal gizli değişken sayısını belirlemek amacıyla, daha önce oluşturulan dokuz farklı LSTM-Otokodlayıcı ağı kullanılmış, bu ağların yalnızca kodlayıcı bileşenleri dahil edilerek işlem yapılmıştır. Bu yöntem, LSTM-Otokodlayıcıların boyut indirgemedeki etkinliğini arıza tespiti performansı üzerinden değerlendirmek için kullanılmıştır.

Deney aşamasında, 20 uçuş verisi üzerinde özel olarak 10 katlı çapraz doğrulama yöntemi uygulanmıştır. Her bir kat, biri sağlıklı diğeri arıza içeren iki farklı uçuştan oluşmuştur. Her kat içerisinde, veri setinin yarısı doğrulama seti olarak ayrılmış ve böylece aşırı öğrenme riski azaltılmıştır. Ayrıca, doğrulama esnasında erken durdurma tekniği kullanılmış ve doğrulama doğruluğu 7 ardışık epoch boyunca iyileşmediğinde eğitim süreci sonlandırılmıştır.

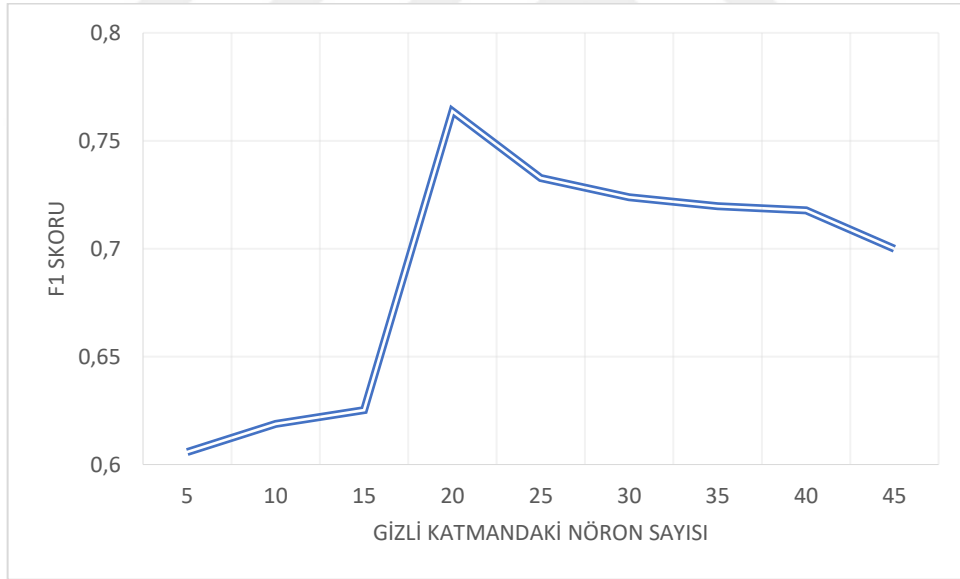
Deneyler kapsamında, LSTM-Otokodlayıcı ağlarının gizli katmanındaki nöron sayısı değiştirilerek dokuz farklı ağ oluşturulmuş ve bu ağlar, BiLSTM sınıflandırıcı ağı ile entegre edilerek test edilmiştir. Her bir ağ, dört kez tekrar edilerek performans ölçümleri alınmış ve ortalama sınıflandırma performansı sonuçları Şekil 5.3-5.6'de sunulmuştur. Bu deneylerde kullanılan farklı ölçüm metrikleri, modelin genel performansını kapsamlı bir şekilde değerlendirmeye olanak tanımıştır.

Elde edilen bulgular, LSTM-Otokodlayıcı kullanımıyla gerçekleştirilen boyut indirgeme sürecinin, BiLSTM sınıflandırıcı performansını artırdığını ve arıza tespitinde

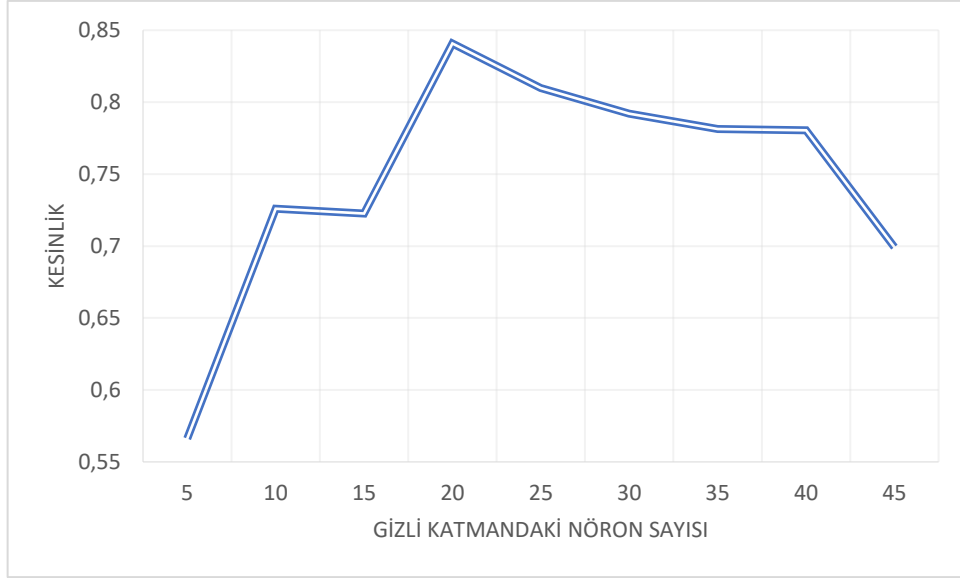
etkin bir yöntem olduğunu göstermektedir. Bu sayede, doğruluk, genelleştirilebilirlik ve hesaplama verimliliği açısından tatmin edici sonuçlar elde edilmiştir.



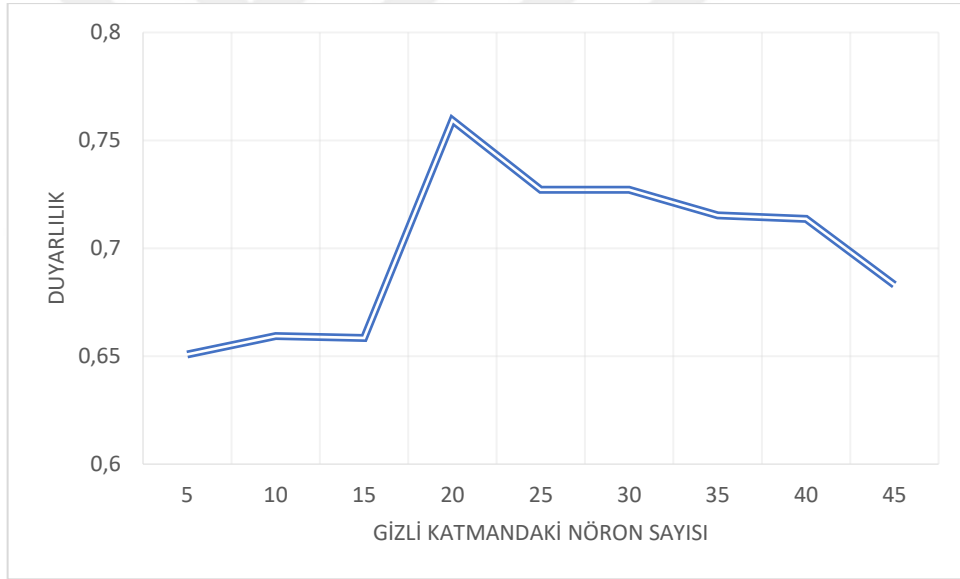
Şekil 5.3. Düşme tahmini sınıflandırıcısı doğruluk grafiği



Şekil 5.4. Düşme tahmini sınıflandırıcısı F1 skoru grafiği



Şekil 5.5. Düşme tahmini sınıflandırıcısı kesinlik grafiği



Şekil 5.6. Düşme tahmini sınıflandırıcısı duyarlılık grafiği

Elde edilen sonuçlara göre, boyut indirgeme uygulanmadan gerçekleştirilen arıza tespiti deneylerinde doğruluk oranı %73,57 olarak belirlenmiştir. Boyut indirgeme işleminin uygulandığı sonraki deneylerde, gizli katmandaki nöron sayısının 45'ten 20'ye düşürülmesi, sınıflandırma doğruluğunu %74,83'ten %79,48'e yükseltmiştir. Ancak, nöron sayısının 15'e daha fazla azaltılması durumunda sınıflandırma başarısında önemli bir düşüş gözlenmiş ve doğruluk oranı %70,48'e gerilemiştir. Nöron sayısının daha da

azaltılması ise sınıflandırma başarısını daha da olumsuz etkilemiş ve doğruluk oranı %67.95 seviyesine kadar düşmüştür.

Kesinlik (precision) ve duyarlılık (recall) ölçütleri de doğruluk metriğinde gözlemlenen eğilimleri destekler niteliktedir. Kesinlik değeri, nöron sayısı 45 iken 0,70 civarında başlamış ve nöron sayısının 20'ye düşürülmesiyle 0,84 ile zirve yapmıştır. Bu eğilim, optimum nöron sayısının modelin arıza içeren uçuşları doğru bir şekilde tespit etme yetisini (doğru pozitifler) artırırken, yanlış pozitifleri en aza indirdiğini göstermektedir. Benzer şekilde, duyarlılık değeri de nöron sayısı 45 iken 0,68 seviyesinde başlamış, 20 nöronda 0,76 ile en yüksek seviyeye ulaşmış, ardından nöron sayısının azaltılmasıyla birlikte düşüş göstermiştir. Bu durum, modelin tüm gerçek arızaları tespit etme yetisinin, optimum gizli katman boyutu ile maksimize edildiğini göstermektedir.

F1-skora bakıldığında, diğer metriklerde gözlemlenen eğilimlerin aynısı görülmektedir. Veri setinin, sağlıklı ve arızalı uçuşlardan son 30 saniyelik veriler kullanılarak dengelenmiş olması, F1-skorunun istikrarlı bir ölçüm sağladığını göstermektedir. F1-skora, nöron sayısı 45 iken 0,70 seviyesinde başlamış, 20 nöronda yaklaşık 0,76 ile zirveye ulaşmış, ardından nöron sayısının daha da azaltılmasıyla düşüş eğilimi göstermiştir. 20 nöronda elde edilen en yüksek F1-skora, modelin hem yüksek kesinlik hem de yüksek geri çağırma değerlerini dengeli bir şekilde sağladığını ortaya koymaktadır.

Düşme tahmini stratejisinde tahmin için sınıflandırma yapılmadan önce eğitilmiş bir LSTM-Otokodlayıcı modeline girdi olarak verilen verilerin sıkıştırılmış bir temsili elde edilir. Modelin her türlü uçuş verisini doğru şekilde temsil edebilme yeteneğine sahip olması için eğitim esnasında girdi örnekleminin kapsayıcı olması gerekir. Düşme tahmininin gerçekleştirildiği BiLSTM sınıflandırıcı modelinde ise modele giren uçuş verilerindeki anomalilikler tespit edilmeye çalışılır ve model bu tespit üzerinden düşüşün gerçekleşip gerçekleşmeyeceğine karar verir. Burada sınıflandırıcı modele giren veri normal uçuş karakteristiğinden ne kadar uzak veya farklı olursa sınıflandırıcının başarısının da o derecede yüksek olması beklenir. Bu düşünceden yola çıkarak bakıldığında düşme tahmininde doğruluğun artırılması sınıflandırıcı modele giren sağlıklı ve sağlıklı uçuş verilerinin birbirinden ne kadar ayrıştırılabilir olduğuyla doğrudan ilgilidir. Uyguladığımız stratejide BiLSTM sınıflandırıcı modeline verilen girdiler, LSTM-Otokodlayıcı modeli sonucu sıkıştırılan temsili uçuş verileri olduğu için

sağlıklı ve sağlıksız uçuşlara ait verilerin birbirinden daha fazla ayrıştırılması bu modelle gerçekleştirilebilir.

Sağlıklı uçuşlarla eğitilen bir LSTM-Otokodlayıcı modeli sağlıklı uçuş karakteristiğini doğru şekilde temsil edebilme kabiliyetine sahip olur. Modelin gizli katmanında da benzer şekilde sağlıklı bir uçuşu doğru şekilde temsil edebilen bir örneğin elde edilebilmesi mümkün olur. Model sağlıklı uçuş verileriyle eğitildiği için, sağlıksız bir uçuşa ait bir veri paketi böyle bir modele sokulduğunda, modelin çıktısı ile gerçek veri arasında hesaplanan hata miktarı çok daha yüksek olur. Dolayısıyla modelin gizli katmanından elde edilen sağlıklı ve sağlıksız uçuşlara ait verilerin karakteristiği birbirinden daha da ayrıştırılmış olur.

Bu durum gözetilerek elde edilen sınıflandırma sonuçları sonrasında düşme tahmini başarısının daha yüksek seviyelere taşınabilmesi için LSTM-Otokodlayıcı modeli sadece sağlıklı uçuş verileri ile eğitilmiş ve çalışmalara bu şekilde devam edilmiştir. Bununla birlikte uygulanan normalizasyon yöntemi min-max normalizasyonu olarak değiştirilmiş, modelin parametreleri de bu model üzerinden tekrar optimize edilmiştir. ADAM optimizasyon algoritmasıyla eğitilen LSTM-Otokodlayıcı modelinin eğitiminde kullanılan parametre değerleri Tablo 5.1’de verilmiştir.

Tablo 5.1. LSTM-Otokodlayıcı modeli eğitim parametreleri

Eğitim Seçenekleri	Değerler
GradientDecayFactor	0,9
MaxEpochs	60
InitialLearnRate	0,01
LearnRateSchedule	'none'
LearnRateDropFactor	0,1
LearnRateDropPeriod	10
MiniBatchSize	128
Shuffle	'every-epoch'
CheckpointFrequencyUnit	'epoch'
PreprocessingEnvironment	'serial'
Verbose	1
VerboseFrequency	50
ValidationFrequency	50
ObjectiveMetricName	'loss'
ExecutionEnvironment	Auto

Tablo 5.1. (Devam) LSTM-Otokodlayıcı modeli eğitim parametreleri

SequenceLength	'longest'
SequencePaddingValue	0
SequencePaddingDirection	'right'
InputDataFormats	Auto
TargetDataFormats	Auto
ResetInputNormalization	1
BatchNormalizationStatistics	Auto
OutputNetwork	Auto
Acceleration	Auto
CategoricalInputEncoding	'integer'
CategoricalTargetEncoding	Auto
L2Regularization	0,0001
GradientThresholdMethod	'l2norm'
GradientThreshold	Inf
Epsilon	0,00000001

LSTM-Otokodlayıcı modeli eğitimi için sağlıklı uçuşlara ait veri segmentlerinin tamamı, %80 eğitim, %10 test, %10 doğrulama şeklinde üç parçaya bölünmüş ve modelin eğitiminde bu %80'lik kısım kullanılmıştır. BiLSTM sınıflandırıcı modeli eğitimi için yine ADAM optimizasyon algoritması kullanılmış, eğitim sırasında kullanılan parametrelerin değerleri ise Tablo 5.2'de verilmiştir.

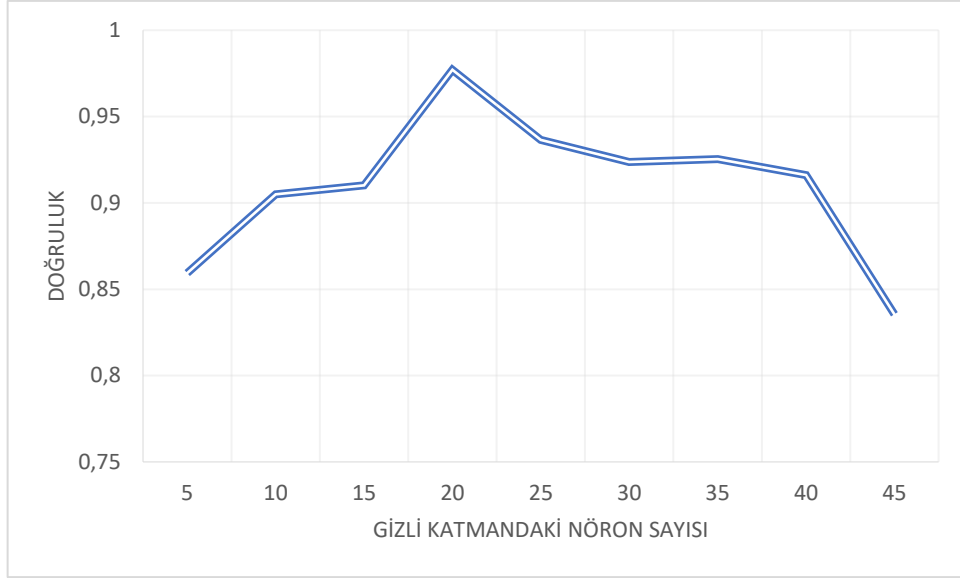
Tablo 5.2. BiLSTM sınıflandırıcı modeli eğitim parametreleri

Eğitim Seçenekleri	Değerler
GradientDecayFactor	0,9
MaxEpochs	100
InitialLearnRate	0,01
LearnRateSchedule	'none'
LearnRateDropFactor	0,1
LearnRateDropPeriod	10
MiniBatchSize	50
Shuffle	'every-epoch'
CheckpointFrequencyUnit	'epoch'
PreprocessingEnvironment	'serial'
Verbose	0
VerboseFrequency	50

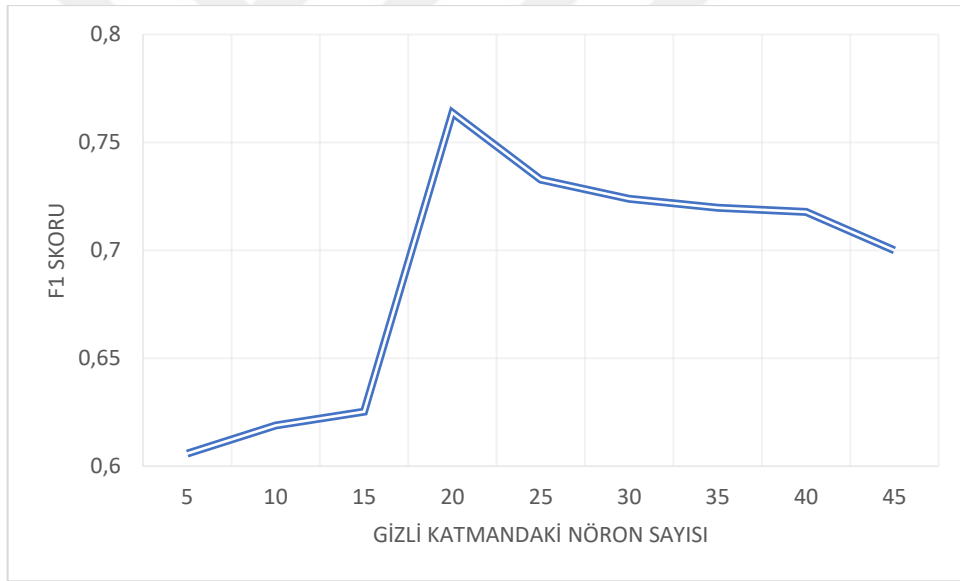
Tablo 5.2. (Devam) BiLSTM sınıflandırıcı modeli eğitim parametreleri

ValidationFrequency	10
ValidationPatience	7
ObjectiveMetricName	'loss'
ExecutionEnvironment	'cpu'
SequenceLength	'longest'
SequencePaddingValue	0
SequencePaddingDirection	'right'
InputDataFormats	Auto
TargetDataFormats	Auto
ResetInputNormalization	1
BatchNormalizationStatistics	'auto'
OutputNetwork	'best-validation-loss'
Acceleration	Auto
CheckpointFrequency	1
CategoricalInputEncoding	'integer'
CategoricalTargetEncoding	'auto'
L2Regularization	0,01
GradientThresholdMethod	'l2norm'
GradientThreshold	1
SquaredGradientDecayFactor	0.9990
Epsilon	0,00000001

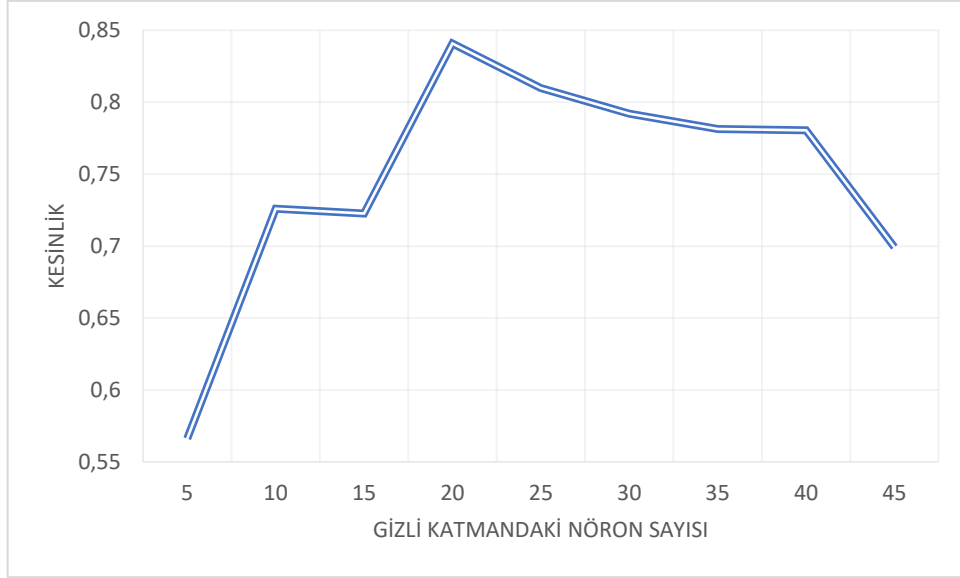
Düşme tahmini için sınıflandırma yapılmadan önce LSTM-Otokodlayıcı modelinin gizli katmanındaki nöron sayıları 45'ten 5'e kadar 5'er 5'er azaltılarak değiştirilmiş ve 9 farklı LSTM-Otokodlayıcı modeli oluşturulmuş ve kaydedilmiştir. Daha sonra bu modellerin her biri ayrı ayrı BiLSTM sınıflandırıcı ile kullanılmıştır. Düşme tahmini için yapılan bu sınıflandırma deneylerinde her bir katta bulunan uçuşlarda değiştirme yapılmadan daha önce bahsedildiği şekliyle 10 katlı çapraz doğrulama yapılmış ve farklı değerlendirme ölçütlerine göre elde edilen ortalama sınıflandırma başarıları Şekil 5.7-5.10 arasında gösterilmiştir.



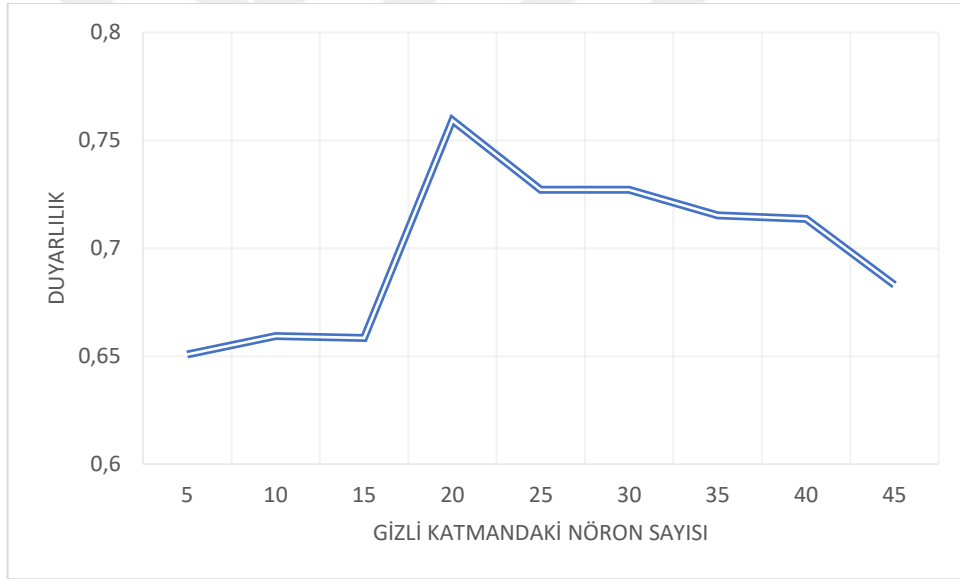
Şekil 5.7. *İyileştirilmiş düşme tahmini sınıflandırıcısı doğruluk grafiği*



Şekil 5.8. *İyileştirilmiş düşme tahmini sınıflandırıcısı F1 skoru grafiği*



Şekil 5.9. İyileştirilmiş düşme tahmini sınıflandırıcısı kesinlik grafiği



Şekil 5.10. İyileştirilmiş düşme tahmini sınıflandırıcısı duyarlılık grafiği

Deney sonuçlarındaki doğruluk grafiğine bakıldığında, LSTM-Otokodlayıcı modelinin gizli katmanındaki nöron sayısı 45 iken 0,8355 doğruluk değeri elde edildiği görülmektedir. Gizli katmandaki nöron sayısı 45'ten 20'ye kadar azaldıkça doğruluk değerinin sürekli olarak arttığı gözlemlenmektedir. Nöron sayısı 20 iken 0,9771 ile en yüksek sınıflandırma doğruluğu başarıları elde edilmiştir. Nöron sayısı 20'den 5'e doğru azaltıldığında doğruluk değerinin yine azalarak devam ettiği ve 0,8593'e kadar düştüğü gözlemlenmiştir.

F1 skoru, kesinlik ve duyarlılık grafikleri incelendiğinde, bu ölçütlere ait grafiklerin gösterdiği trendlerin doğruluk grafiğiyle çok benzer şekilde ilerlediği gözlemlenmektedir. Gizli katman nöron sayısı 45'ten 20'e gelirken F1 skoru değeri 0,7789'dan başlayarak 0,9786'ya kadar yükselmiştir. Benzer şekillerde kesinlik ve duyarlılık değerlerinin de sırasıyla 0,9000'dan 0,9750'ye ve 0,7627'den 0,9831'e kadar arttığı gözlemlenmektedir. Nöron sayısı 20'den 5'e kadar azalırken ise yine doğruluk grafiğine benzer şekilde F1 skor, kesinlik ve duyarlılık değerlerinde azalma gözlemlenmiştir. Nöron sayısı 5'e ulaştığında F1 skoru, kesinlik ve duyarlılık değerleri sırasıyla 0,8635, 0,8563 ve 0,8512 olarak elde edilmiştir.

Sonuçlar incelendiğinde sağlıklı bir uçuş sergileyen bir quadrotorun arıza kaynaklı bir düşme ihtimalinin bulunmadığını, uçuş sağlığı bozulan veya arızalı bir quadrotorun ise düşüşüne son 30 saniye kaldığı andan itibaren düşeceğini %97 doğrulukla tespit edebilen bir model oluşturulduğu görülmektedir. Bu doğruluk değeriyle birlikte F1 skorunun da %97 seviyesinde olması modelin güvenilir ve genellenebilir olduğunu göstermektedir. Aynı zamanda modelin kesinlik seviyesinin yüksek olması, bir quadrotor üzerine entegre edilecek geliştirdiğimiz modeli kullanan gerçek zamanlı düşme tespit sistemine sistemin düşme tehlikesi alarmı verirken daha temkinli ve seçici davranacağını gösterir. Sistem alarm durumuna geçtiğinde düşme tehlikesi durumunun önemli olduğu ve eğer sistem tarafından alarm verildiyse bu durumun gerçekten ciddiye alınması ve sağlıklı bir iniş sağlayan veya düşmeyi önleyen bir sistemin acilen devreye sokulması gerektiği çıkarımı yapılabilir.

Bulgular, gizli katmandaki nöron sayısının fazla olması durumunda sisteme gereksiz bilgi yüklendiğini ve bunun sınıflandırma performansı üzerinde olumsuz bir etki yarattığını göstermektedir. Öte yandan, nöron sayısının azaltılması, sınıflandırma başarısını artırarak gereksiz veya yinelenen bilgilerin elimine edildiğini ve önemli bilginin daha etkin bir şekilde kullanıldığını göstermektedir. Bununla birlikte, boyut indirgeme sürecinin daha da ilerletilmesi, arıza açıklayıcı bilginin kaybolmasına neden olarak sınıflandırma performansında belirgin bir düşüşe yol açmaktadır.

Genel olarak elde edilen sonuçlarla, boyut indirgeme işleminin performans iyileştirmesi için kritik bir rol oynadığı ortaya konulmuştur. Gizli katmanda bulunan optimum nöron sayısının belirlenmesi, modelin daha yüksek doğruluk, daha az karmaşıklık ve daha etkili öğrenme sağlaması açısından ciddi öneme sahiptir.

5.5. Uçuş Sağlığı Tahmini

Uçuş sağlığı tahmini için deney sonuçları bölümü, anomali skoru tahmini modeli için gerçekleştirilen deneylerin analiz edilmesi ve sonuçların tartışılması amacıyla sunulmuştur. Düşme tahmini modelinde en yüksek sınıflandırma başarısı gizli katmandaki 20 nöron sayısı ile elde edildiği için uçuş sağlığı modelinde de boyut indirilmesi için aynı yapı kullanılmış, öznitelik sayısı öncelikle 50'den 20'ye indirilmiştir. Bu eğitim süreci uçuş öncesinde çevrimdışı olarak gerçekleştirilir. Quadrotor uçuşu esnasında sensörlerden gerçek zamanlı olarak akan veriler segmentler halinde bu modele sokularak 20 özniteliğe indirgenmiş veri segmentlerine dönüştürülür.

Anomali skoru tahmini modelinin giriş katmanı indirgenmiş veriye uygun şekilde 20 nöron olarak belirlenmiştir. Sorasında 30 gizli birimden oluşan ve son zaman adımının gizli durumunu çıktı olarak veren LSTM katmanı kullanılır. Ardından aşırı öğrenmeyi engelleme amacıyla girdi elemanlarını 0,5 olasılıkla eleyen bir dropout katmanı eklenmiştir. Bu katmanın çıktısı 10 nöronlu tam bağımlı bir katmana bağlanır. Sürekli bir değer çıktısı üretmek üzere son olarak bu katman tek bir nörona bağlanır ve buradan regresyon yöntemiyle son çıkış sağlanır.

Modelin eğitiminde kullanılan verilerin etiketlenme süreci Bölüm 4.2. de detaylıca anlatılmıştır. Öncelikle etiketleme için LSTM-Otokodlayıcı modelinin eğitimi sağlıklı uçuşlar üzerinden gerçekleştirilmiştir. Sonrasında sağlıklı uçuşlarda anomali skoru 0 olarak etiketlenirken, düşen uçuşlarda hata hesaplaması için yapılan deneyler sonrasında Pitch, RMSPitchD, DCRt, VD ve C2_out özellikleri baz alınmıştır. Hata eşik değeri için de çeşitli deneyler gerçekleştirilmiş olup ideal eşik değeri 0,08 olarak belirlenmiştir. Eğitim sırasında kullanılan diğer seçenekler ve parametreler Tablo 5.3'de sunulmuştur.

Tablo 5.3. *Anomali skoru tahmini modeli için eğitim seçenekleri*

Eğitim Seçenekleri	Değerler
Input Size	20
Num Hidden Units	30
Algorithm	ADAM
Dropout Layer Probability	0,5
Max Epoch	100
Mini Batch Size	50
Initial Learn Rate	0,01
Gradient Threshold	1

Tablo 5.3. (Devam) Anomali skoru tahmini modeli için eğitim seçenekleri

Shuffle	Every-epoch
SequenceLength	Longest
Validation Frequency	10
Output Network	Best validation loss
Output Mode	Last
Validation Patience	50

Modelin tahmin performansını ölçmek için MAE ve RMSE ölçütleri kullanılmıştır. Ayrıca, modelin tahmin ettiği değerlerin yumuşatılmış versiyonları için de MAE2 ve RMSE2 olarak ifade edilmiştir. Bu yumuşatılmış tahminler, son 20 segment için üretilen tahminlerin ortalaması alınarak elde edilmiştir. Model güvenilirliğini test etmek için 10 katlı çapraz doğrulama kullanılmıştır. Her bir katta bir sağlıklı ve bir düşen uçuş test kümesi için ayrılmış, geri kalan uçuşlar ise eğitim kümesi olarak kullanılmıştır. Tablo 5.2 ve Tablo 5.3'te, sırasıyla her kattaki sağlıklı ve düşen uçuşlar için elde edilen hata değerleri ile bu uçuşlar için üretilen yumuşatılmış tahminlerin hata değerleri yer almaktadır.

Tablo 5.4. Sağlıklı uçuşlar için gerçekleştirilen anomali skoru tahmininde MAE ve RMSE değerleri

Kat	MAE	MAE 2	RMSE	RMSE 2
1	0,0982	0,0970	0,1034	0,1007
2	0,1242	0,1239	0,1346	0,1305
3	0,0484	0,0452	0,0587	0,0528
4	0,0188	0,0139	0,0236	0,0168
5	0,0429	0,0402	0,0533	0,0492
6	0,1271	0,1281	0,1290	0,1296
7	0,0234	0,0196	0,0321	0,0261
8	0,1204	0,1178	0,1262	0,1227
9	0,0982	0,0970	0,1034	0,1007
10	0,0279	0,0277	0,0305	0,0293
Ortalama:	0,0729	0,0710	0,0794	0,0758
S. Sapma:	0,0447	0,0459	0,0443	0,0455

Sağlıklı uçuşlar için modelin performansı incelendiğinde, en düşük MAE değeri 0,0188 ile 4. katta elde edilmiştir. Bu kat için RMSE değeri de 0,0236 ile minimum

seviyededir. Öte yandan, en yüksek MAE değeri 0,1271 ile 6. katta gözlemlenmiştir. Ortalama MAE değeri 0,0729, ortalama RMSE değeri ise 0,0794 olarak hesaplanmıştır.

Yumuşatılmış tahminlerin sonuçları incelendiğinde, yumuşatma işlemi hata oranlarını hafifçe azaltmıştır. Örneğin, 4. katta MAE2 değeri 0,0139'a, RMSE2 değeri ise 0,0168'e düşmüştür. Genel olarak, yumuşatılmış tahminlerin ortalama MAE ve RMSE değerleri sırasıyla 0,0710 ve 0,0758'dir. Bu, modelin tahmin doğruluğunu artırmak için yumuşatma işleminin faydalı olduğunu göstermektedir.

Standart sapma değerleri, modelin farklı katlar arasındaki performans dalgalanmalarını yansıtmaktadır. Sağlıklı uçuşlar için MAE'nin standart sapması 0,0447, RMSE'nin standart sapması ise 0,0443'tür. Burada elde edilen düşük sapmalar, modelin tutarlı bir performans sergilediğini göstermektedir.

Tablo 5.5. Düşen uçuşlar için gerçekleştirilen anomali skoru tahmininde MAE ve RMSE değerleri

Kat	MAE	MAE 2	RMSE	RMSE 2
1	0,2741	0,2737	0,3199	0,3190
2	0,2822	0,2735	0,3308	0,3139
3	0,2396	0,2393	0,2784	0,2781
4	0,2062	0,1921	0,2401	0,2261
5	0,2464	0,2448	0,0923	0,2907
6	0,2385	0,2362	0,2773	0,2734
7	0,1680	0,1642	0,2312	0,2209
8	0,2235	0,2180	0,2549	0,2485
9	0,2741	0,2737	0,3199	0,3190
10	0,1944	0,1935	0,2340	0,2337
Ortalama:	0,2347	0,2309	0,2588	0,2733
S. Sapma:	0,0373	0,0384	0,0689	0,0384

Düşen uçuşlar için modelin performansı incelendiğinde, en düşük MAE değeri 0,1680 ile 7. katta elde edilmiştir. Bu kat için RMSE değeri ise 0,2312 olarak hesaplanmıştır. En yüksek MAE değeri 0,2822 ile 2. katta gözlemlenmiştir. Ortalama MAE değeri 0,2347, ortalama RMSE değeri ise 0,2588 olarak hesaplanmıştır.

Yumuşatılmış tahminler, yine düşen uçuşlar için de hata oranlarını hafifçe azaltmıştır. Örneğin, 2. katta MAE2 değeri 0,2735'e, RMSE2 değeri ise 0,3139'a

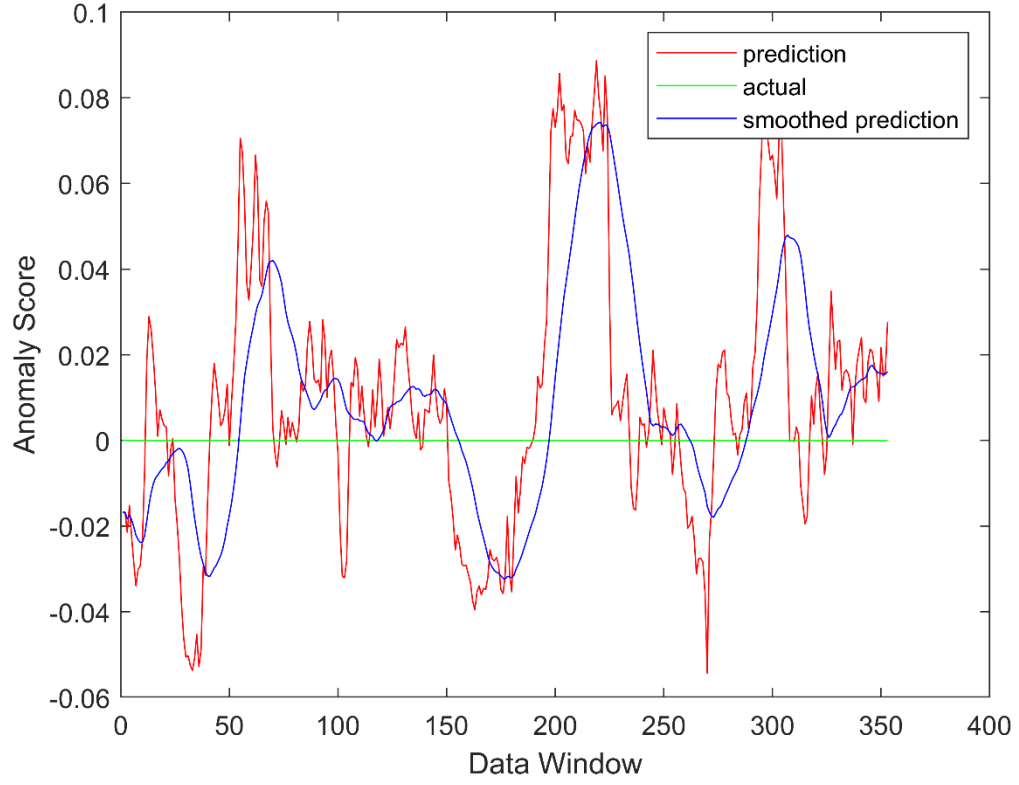
düşmüştür. Genel olarak, yumuşatılmış tahminlerin ortalama MAE ve RMSE değerleri sırasıyla 0,2309 ve 0,2733'tür.

Düşen uçuşlar için MAE'nin standart sapması 0,0373, RMSE'nin standart sapması ise 0,0689'dur. Bu değerler, modelin performansının katlar arasında sağlıklı uçuşlara göre daha değişken olduğunu göstermektedir. Özellikle RMSE değerlerinde gözlemlenen yüksek sapma, modelin bazı katlarda diğerlerine göre daha az tutarlı tahminler yaptığını göstermektedir.

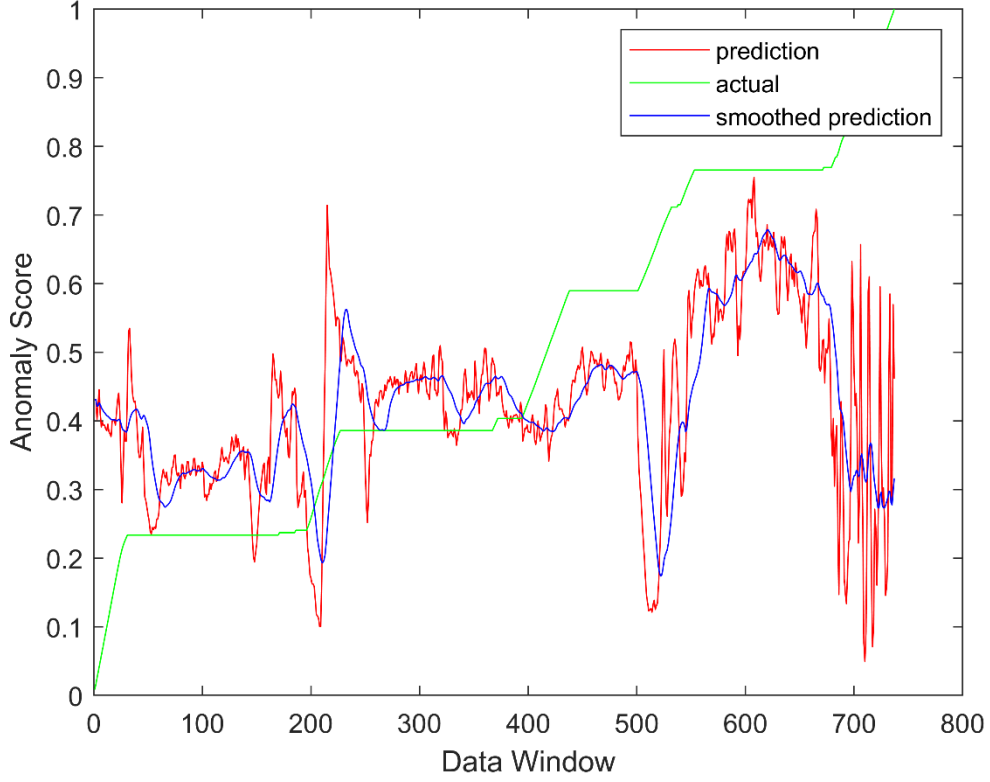
Genel olarak, her iki uçuş türü için de elde edilen ortalama hata değerleri karşılaştırıldığında, sağlıklı uçuşlarda tahmin doğruluğunun daha yüksek olduğu ve düşen uçuşlarda modelin daha büyük hata değerlerine sahip olduğu anlaşılmaktadır. Genel ortalamalara bakıldığında, MAE ve RMSE değerleri sırasıyla 0,1538 ve 0,1687 olarak hesaplanmıştır. Yumuşatılmış tahminler için bu değerler sırasıyla 0,1510 ve 0,1741 olarak bulunmuştur. Bu sonuçlar, modelin genelde istikrarlı bir performans sergilediğini ve yumuşatma işleminin tahmin doğruluğunu bir miktar artırdığını göstermektedir.

Zamana bağlı olarak oluşturulmuş veri segmentlerine ait etiketlerin, tahminlerin ve yumuşatılmış tahminlerin nasıl değiştiğini eş zamanlı olarak gözlemlemek üzere örnek olarak 7. katta bulunan sağlıklı ve düşen uçuşa ait anomali skoru tahmini grafikleri sırasıyla Şekil 5.11 ve Şekil 5.12'de verilmiştir. Yatay ekseninde verilen veri segmentleri kayan pencereler yöntemiyle üretilmiştir. Dolayısıyla 1 numaralı veri segmenti uçuşun başlangıcına karşılık gelirken her bir sonraki veri segmenti uçuşun 0,5 sn sonrasında karşılık gelmektedir. Quadrotorlarda ilk kalkış esnasında stabilizasyonun sağlanabilmesi için belli bir süreye ihtiyaç vardır. Bu süreç boyunca elde edilen sensör verileri aykırı değer olarak nitelendirildiği için uçuşların ilk 10 saniyelik kısımları eğitime dahil edilmemiş ve grafik sonuçlarında gösterilmemiştir. Düşen uçuşlar için de düşüş

başlangıcına kadar olan kısım eğitimlerde kullanılmış, grafiklerde de yalnızca bu segmentler gösterilmiştir.



Şekil 5.11. 7. kata ait sağlıklı uçuş için anomali skoru tahmin değerleri



Şekil 5.12. 7. kata ait düşen uçuş için anomali skoru tahmin değerleri

Sağlıklı uçuştaki grafiğe baktığımızda anomali skorunun 0 olarak etiketlendiği görülmektedir. Tahminlere baktığımızda ise sonuçların 0 civarında artan ve azalan şekillerde ilerlediği görülmektedir. Uçuşun en sağlıklı hali anomali skorunun 0 değeriyle ifade edildiğinden grafikte 0'ın altına inen değerler burada 0'a yuvarlanabilir. Bu yuvarlama gerçekleştirildiğinde sağlıklı uçuşlar için üretilen tahmin değerlerinin hatasının daha fazla düşeceği göz önünde bulundurulabilir. Bu da sağlıklı uçuşlar için uçuş sağlığının daha doğru şekilde tahmin edilebildiğini göstermektedir.

Düşen uçuş grafiğini izlediğimizde uyguladığımız etiketleme stratejisi sonrasında anomali skoru etiket değerlerinde basamak basamak artış gerçekleştiği görülmektedir. Başlangıçtaki anomali skoru ile üretilen tahmin farkına baktığımızda düşen uçuşlar için başlangıçta da belli seviyede bir uçuş sağlığı bozukluğu olduğu varsayılabilir. Üretilen tahminlerin uçuşun son kısmına kadar yapılan etiketlemeyi takip etme eğiliminde olduğu görülürken belli bölümlerde kısa zamanlı yüksek sapmalar gözlemlenmektedir. Yapılan yumuşatma işleminin ise özellikle bu anlarda tahmin doğruluğunu arttırdığı görülmektedir. Tahmin hatasının en olduğu değerler ise düşüş başlangıcına en yakın

noktalarda ölçülmüştür. Her iki uçuştaki yumuşatılmış tahmin değerlerine bakıldığında ise sağlıklı ve düşen uçuşlar için üretilen tahminlerin tamamıyla birbirlerinden ayrıştıkları görülmektedir.



6. TARTIŞMA VE SONUÇ

Bu çalışma, derin öğrenme tabanlı modellerin uçuş sağlığı ve düşme tahmini üzerindeki öğrenme yeteneklerini inceleyerek, karmaşık sistem üzerinden elde edilen verilerin gerçek zamanlı ve kapsamlı analiziyle sistem sağlığına yönelik çıkarımlar kazanılmasını sağlamıştır. Yapılan çalışma kapsamında quadrotorların potansiyel arıza durumlarını öngörerek güvenli inişlerini sağlamak amacıyla düşme tahmini ve uçuş sağlığı tahmini olmak üzere derin öğrenme tabanlı iki yöntem önerilmiştir. Önerilen yaklaşımlar, erken uyarı sistemi aracılığıyla ciddi arızalar meydana gelmeden önce gerekli tedbirlerin alınmasına olanak tanımaktadır. Çalışma, daha önceki çalışmalardan farklı olarak tüm mevcut sensör verilerinin kapsamlı bir şekilde kullanılmasını sağlamış ve bu sayede hiçbir önemli bilginin göz ardı edilmemesi hedeflenmiştir. Ancak bazı sensör verilerinin gereksiz ya da tekrarlayan bilgi içerdiği bilinmekte olup, sınıflandırma ve tahmin işlemlerinden önce boyut indirgeme yöntemi uygulanarak bu tür verilerin negatif yönlü etkileri ortadan kaldırılmıştır. Bu adım, hem arıza tespit doğruluğunu artırmış hem de modelin hesaplama karmaşıklığını azaltarak performansı optimize etmiştir.

Önerilen yöntemlerde şu üç derin öğrenme modeli kullanılmıştır:

LSTM-Otokodlayıcı: Bu bileşen, özellik boyut indirgeme işlemlerini gerçekleştirmekle birlikte, her bir quadrotor uçuşuna ait yeniden yapılandırma hatalarını hesaplamak için kullanılmıştır.

BiLSTM sınıflandırıcı: Eğitim sürecinde hem sağlıklı hem de arızalı uçuş verilerini kullanarak başlangıç aşamasındaki arızaların öğrenilmesini sağlar.

LSTM regressor: Uçuş sağlığını tayin etmek üzere anomali skorunu tahmin etmek için sürekli bir değer üretilmesini sağlar.

Kullanılan modeller, hem internet üzerinden toplanan, hem de üniversite kampüsünde gerçekleştirilen uçuşlardan elde edilen verilerle oluşturulan quadrotor uçuş veri seti üzerinde eğitilmiş ve doğrulanmıştır. Sonuçların güvenilirliğini arttırmak adına 10 katlı çapraz doğrulama gerçekleştirilmiştir. Her katta sağlıklı ve düşen uçuşlardan birer tanesi test için ayrılmış, geriye kalan 9 sağlıklı ve 9 düşen uçuş verileri eğitim için kullanılmıştır.

Düşme tahminini yöntemine yönelik deneysel sonuçlar, bu yöntemin başlangıç aşamasındaki arızaları ciddi arızaların tetiklenmesinden 30 saniye önce tespit ettiğini göstermiştir. Elde edilen sonuçlara göre, modelin ortalama doğruluk değeri 0,79 olarak

belirlenmiştir. LSTM-Otokodlayıcı modelinin eğitimi için kullanılan verilerin sadece sağlıklı uçuşlarla seçilmesiyle bu doğruluk başarısı 0,97 seviyesine kadar yükseltilmiştir. Bu bulgu, modelin quadrotor uçuş operasyonlarının güvenliğini ve emniyetini önemli ölçüde artırma potansiyeline sahip olduğunu ortaya koymaktadır. 0,76 olarak ölçülen F1 skoru değeri yine aynı şekilde 0,97 seviyesine çıkarılmış, elde edilen bu yüksek değerle modelin düşen uçuş sınıfını doğru şekilde tespit etme yeteneğinin yüksek olduğu ortaya konulmuştur.

Uçuş sağlığı tahmini yöntemine yönelik bulunan sonuçlar, anomali skoru tahminine yönelik geliştirilen modellerin performansını detaylı bir şekilde değerlendirme olanağı sunmuştur. Deneyler sırasında kullanılan eğitim seçenekleri ve optimizasyon parametreleri, modellerin farklı uçuş durumlarında gösterdiği başarıyı doğrudan etkilemiştir. Özellikle sağlıklı uçuşlarda elde edilen düşük hata oranları anomali skoru tahmini modelinin belirli koşullarda yüksek doğrulukla çalıştığını göstermektedir. Ancak, düşen uçuşlarda gözlemlenen daha yüksek hata oranları, modelin performans sınırlarını ve bu sınırların aşılabilmesi için gelecekte atılması gereken adımları işaret etmektedir.

Sağlıklı uçuşlarda elde edilen MAE ve RMSE değerlerinin düşük olması, uçuş sağlığı tahmini yönteminde, anomali skorlarını belirgin bir doğrulukla tahmin edebildiğini ortaya koymaktadır. Bu sonuç, özellikle quadrotor gibi karmaşık sistemlerde operasyonel güvenliği artırma ve bakım maliyetlerini düşürme açısından umut verici bir gösterge sunmaktadır. Öte yandan, düşen uçuşlarda elde edilen yüksek hata oranları, bu tür uçuşlar için geliştirilecek modellerde daha fazla öznetelik mühendisliği ve daha fazla veriye ihtiyaç duyulabileceğini göstermektedir. Bu bağlamda, düşen uçuşlardaki performans iyileştirmesi, veri setinin genişletilmesi veya farklı uçuş senaryolarının modele entegre edilmesiyle mümkün olabilir.

Uçuş sağlığı tahmini modelinin genel performansını değerlendirirken, yumuşatılmış tahminlerin tahmin doğruluğuna yaptığı katkılar da dikkate alınmalıdır. Model tarafından üretilen tahminlerin son 20 tanesinin ortalamasını almak, daha dengeli ve istikrarlı bir tahmin süreci sağlamaktadır. Bu yaklaşım, özellikle yüksek varyansa sahip verilerle çalışılırken modelin kararlılığını artırabilir. Ancak, yumuşatma işleminin bazı durumlarda anomali algılamayı az da olsa geciktirebileceği de unutulmamalıdır. Gelecekte, yumuşatma parametrelerinin adaptif bir şekilde ayarlanması ve uçuş durumuna göre optimize edilmesi, bu tür sorunların önüne geçebilir.

Önerilen modeller, giriş boyutları ve nöron sayıları gibi parametrelerin ayarlanabilir olması sayesinde yalnızca quadrotor sistemlerinde değil, diğer karmaşık sistemlerdeki arıza tespit uygulamalarına da kolaylıkla uyarlanabilir bir yapıya sahiptir. Bu esneklik, gelecekte modelin farklı sistemlere adapte edilerek çok daha geniş bir yelpazede uygulanmasını sağlayacaktır.

Bu çalışmanın önemli bir diğer katkısı, LSTM tabanlı otokodlayıcıların uçuş verilerindeki anlamlı özellikleri çıkararak hem boyut indirgeme hem de anomali tespiti açısından yüksek verimlilik sağlamasıdır. Bununla birlikte, BiLSTM ve LSTM tabanlı modellerin, uçuş süresince zaman serisi verilerindeki ardışık bağımlılıkları başarılı bir şekilde öğrenmesi, tahmin performansını artırmıştır. Bu modeller, yalnızca uçuş sırasında meydana gelen anomalileri tespit etmekle kalmayıp, aynı zamanda olası arızaların zamanında tespiti için de bir temel sunmaktadır.

Arıza tespit çalışmalarındaki en yaygın zorluklardan biri, sağlıklı operasyon verilerine kıyasla arıza verilerinin yetersiz kalması, dolayısıyla verilerin dengesiz dağılım göstermesidir. Bu çalışmada, veri dengesizliğini gidermek amacıyla veriler dengeleyici işlemlerden geçirilmiş, ancak bu durum deneylerde bir sınırlamaya neden olmuştur. Sağlıklı uçuş verilerinin önemli bir kısmı tam anlamıyla kullanılamamış olup, bu durum arıza tespit sürecinde bir eksiklik oluşturmuştur.

Çalışmanın sınırlamaları arasında, kullanılan veri setinin boyutu ve dağılımı yer almaktadır. Sağlıklı ve arızalı uçuş segmentlerinin dengelenmesi amacıyla yapılan seçimler, bazı uçuş durumlarının temsil edilmesini sınırlamış olabilir. Veri dengesizliği sorununa yönelik özel yöntemlerin adapte edilmesi bu probleme yönelik bir çözüm yöntemi olarak değerlendirilebilir. Bu doğrultuda, dengesiz veri setleri için geliştirilen yeni yöntemlerin modeller üzerinde uygulanması sağlanarak arıza tespit başarısı daha kapsamlı bir şekilde optimize edilebilir. Ayrıca, çevresel faktörlerin (rüzgâr, sıcaklık, nem gibi) etkisi çalışmada ele alınmamış, yalnızca uçuş dinamiklerine odaklanılmıştır. Gelecekte, bu tür değişkenlerin modele entegre edilmesi, daha kapsamlı bir analiz yapılmasını sağlayabilir.

Sonuç olarak, bu çalışma, derin öğrenme tabanlı erken arıza tespit yöntemlerinin quadrotor sistemlerinde uygulanabilirliğini göstermiştir. Önerilen yöntemler, yalnızca uçuş güvenliğini artırmakla kalmayıp, bakım maliyetlerini düşürme ve sistem ömrünü uzatma açısından da değerli katkılar sunmaktadır. Çalışmanın, sunduğu bulgular ve

yöntemler aracılığıyla literatürdeki mevcut potansiyelin genişletilmesine katkı sağlaması ve gelecekteki araştırmalar için önemli bir referans noktası oluşturması beklenmektedir.



KAYNAKÇA

- Afshari, S. S., ve Pourtakdoust, S. H. (2018). Probability density evolution for time-varying reliability assessment of wing structures. *Aviation*, 45-54.
- Ai, S., Shang, W., Song, J., ve Cai, G. (2021). Fault diagnosis of the four-rotor unmanned aerial vehicle using the optimized deep forest algorithm based on the wavelet packet translation. *2021 8th International Conference on Dependable Systems and Their Applications (DSA)* (s. 581-589). IEEE.
- Ai, S., Song, J., ve Cai, G. (2022). Sequence-to-sequence remaining useful life prediction of the highly maneuverable unmanned aerial vehicle: A multilevel fusion transformer network solution. *Mathematics*, 10(10), 1733.
- Al Younes, Y., Rabhi, A., Noura, H., ve El Hajjaji, A. (2016). Sensor fault diagnosis and fault tolerant control using intelligent-output-estimator applied on quadrotor UAV. *2016 International Conference on Unmanned Aircraft Systems (ICUAS)* (s. 1117-1123). IEEE.
- Altınors, A., Yol, F., ve Yaman, O. (2021). A sound based method for fault detection with statistical feature extraction in UAV motors. *Applied Acoustics*, 183, 108325.
- Anidjar, O. H., Barak, A., Ben-Moshe, B., Hagai, E., ve Tuvyahu, S. (2023). A stethoscope for drones: Transformers-based methods for UAVs acoustic anomaly detection. *IEEE Access*, 11, 33336-33353.
- Belcastro, C. M., Klyde, D. H., Logan, M. J., Newman, R. L., ve Foster, J. V. (2017). Experimental flight testing for assessing the safety of unmanned aircraft system safety-critical operations. *17th AIAA Aviation Technology, Integration, and Operations Conference*, (s. 3274).
- Bondyra, A., Kołodziejczak, M., Kulikowski, R., ve Giernacki, W. (2022). An acoustic fault detection and isolation system for multirotor UAV. *Energies*, 15(11), 3955.
- Chen, B., Peng, Y., Gu, B., Luo, Y., ve Liu, D. (2021). A fault detection method based on enhanced GRU. *2021 International Conference on Sensing, Measurement & Data Analytics in the era of Artificial Intelligence (ICSMD)* (s. 1-4). IEEE.
- Erdal, H., ve Korucuk, S. (2023). Bütünleşik Dalgacık Dönüşümü-Evrişimsel Sinir Ağları Tabanlı Derin Öğrenme Yaklaşımı ve Borsa Tahmini Üzerine Bir Uygulama. *Karadeniz Sosyal Bilimler Dergisi*, 15(29), 387-404.
- Erfanian, A. M., ve Ramezani, A. (2022). Using deep learning network for fault detection in UAV. *2022 8th International Conference on Control, Instrumentation and Automation (ICCIA)* (s. 1-5). IEEE.
- Fu, J., ve Che, G. (2021). Fusion fault diagnosis model for six-rotor UAVs based on conformal Fourier transform and improved self-organizing feature map. *IEEE Access*, 9, 14422-14436.
- Guo, D., Zhong, M., Ji, H., Liu, Y., ve Yang, R. (2018). A hybrid feature model and deep learning based fault diagnosis for unmanned aerial vehicle sensors. *Neurocomputing*, 319, 155-163.
- Hajiyeve, C. (2016). An innovation approach based sensor fault detection and isolation. *IFAC-PapersOnLine*, 49(17), 420-425.

- Hebb, D. O. (2005). *The organization of behavior: A neuropsychological theory*. Psychology press.
- Huang, J., Li, M., Zhang, Y., Mu, L., Ao, Z., ve Gong, H. (2021). Fault detection and classification for sensor faults of UAV by deep learning and time-frequency analysis. *2021 40th Chinese Control Conference (CCC)* (s. 4420-4424). IEEE.
- Iannace, G., Ciaburro, G., ve Trematerra, A. (2019). Fault diagnosis for UAV blades using artificial neural network. *Robotics*, 8(3), 59.
- Ignatovich, S. R., Menou, A., Karuskevich, M. V., ve Maruschak, P. O. (2013). Fatigue damage and sensor development for aircraft structural health monitoring. *Theoretical and applied fracture mechanics*, 65, 23-27.
- Jiang, Y., Zhiyao, Z., Haoxiang, L., ve Quan, Q. (2015). Fault detection and identification for quadrotor based on airframe vibration signals: a data-driven method. *2015 34th chinese control conference (ccc)* (s. 6356-6361). IEEE.
- Jing, C. S., ve Pebrianti, D. (2016). Fault detection and identification in Quadrotor system (Quadrotor robot). *2016 IEEE International Conference on Automatic Control and Intelligent Systems (I2CACIS)* (s. 11-16). IEEE.
- Jing, C. S., Pebrianti, D., Qian, G. M., ve Bayuaji, L. (2017). Fault detection in Quadrotor MAV. *2017 7th IEEE International Conference on System Engineering and Technology (ICSET)* (s. 65-70). IEEE.
- Lee, C., Son, J. J., Yoon, S., Jeon, S., ve Han, S. (2024). Hybrid model-based and data-driven disturbance prediction for precise quadrotor trajectory tracking. *Engineering Applications of Artificial Intelligence*, 136, 108895.
- Liu, L., Ma, Y., Bin, X., Xiang, C., ve Yang, X. (2016). Fault detection and isolation based on UKFs for a novel ducted fan UAV. *IEEE International Conference on Aircraft Utility Systems (AUS)* (s. 212-218). IEEE.
- Liu, W., Chen, Z., ve Zheng, M. (2020). An audio-based fault diagnosis method for quadrotors using convolutional neural network and transfer learning. *2020 American Control Conference (ACC)* (s. 1367-1372). IEEE.
- McCulloch, W. S., ve Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *bulletin of mathematical biophysics*, 5, 115-133.
- Olyaei, M. H., Jalali, H., Noori, A., ve Eghbal, N. (2018). Fault detection and identification on UAV system with CITFA algorithm based on deep learning. *Electrical Engineering (ICEE), Iranian Conference on* (s. 988-993). IEEE.
- Ouadine, A. Y., Mjahed, M., Ayad, H., ve El Kari, A. (2020). UAV quadrotor fault detection and isolation using artificial neural network and Hammerstein-Wiener model. *Studies in Informatics and Control*, 29(3), 317-328.
- Ozcan, M., ve Perkgoz, C. (2024). Deep learning-based proactive fault detection method for enhanced quadrotor safety. *Aviation*, 28(3), 175-187.
- Ozkat, E. C., Bektas, O., Nielsen, M. J., ve Cour-Harbo, A. I. (2023). A data-driven predictive maintenance model to estimate RUL in a multi-rotor UAS. *International Journal of Micro Air Vehicles*, 15, 17568293221150171.

- Pose, C., Giribet, J., Torre, G., ve Marzik, G. (2023). Neural network-based propeller damage detection for multirotors. *International Conference on Unmanned Aircraft Systems (ICUAS)* (s. 17-23). IEEE.
- Puchalski, R., ve Giernacki, W. (2022). UAV fault detection methods, state-of-the-art. *Drones*, 6(11), 330.
- Rhee, J., Veloso, P., ve Krishnamurti, R. (2023). Three decades of machine learning with neural networks in computer-aided architectural design (1990–2021). *Design Science*, 9, e25.
- Shen, F.-Y., Li, W., Jiang, D.-N., ve Mao, H.-J. (2024). Autonomous predictive maintenance of quadrotor UAV with multi-actuator degradation. *The Aeronautical Journal*, 1-25.
- Tong, J. J., Zhang, W., Liao, F., Li, C. F., ve Zhang, Y. F. (2023). Machine learning for uav propeller fault detection based on a hybrid data generation model. *arXiv preprint arXiv*, 2302.01556.
- Vural, S. Y., ve Hacızade, C. (2016). Sensor/actuator fault detection, isolation and accommodation applied to UAV model. *Journal of Aeronautics and Space Technologies*, 9(2), 1-12.
- Wang, B., Datong, L., Peng, Y., ve Peng, X. (2019). Multivariate regression-based fault detection and recovery of UAV flight data. *IEEE Transactions on Instrumentation and Measurement*, 69(6), 3527-3537.
- Yang, P., Geng, H., Wen, C., ve Liu, P. (2021). An intelligent quadrotor fault diagnosis method based on novel deep residual shrinkage network. *Drones*, 5(4), 133.
- Yasniy, O., Mytnyk, M., Maruschak, P., Myktyshyn, A., ve Didych, I. (2024). Machine learning methods as applied to modelling thermal conductivity of epoxy-based composites with different fillers for aircraft. *Aviation*, 28(2), 64-71.
- Zhang, W., Tong, J., Liao, F., ve Zhang, Y. (2023). Simulation-to-reality UAV Fault Diagnosis with Deep Learning. *arXiv preprint arXiv:2302.04410*.
- Zhao, Z., Wang, X., Xu, J., ve Jiabin, Y. (2018). A performance evaluation algorithm of stochastic hybrid systems based on fuzzy health degree and its application to quadrotors. *IEEE Access*, 37581-37594.
- Zheng, K., Jia, G., Yang, L., ve Wang, J. (2021). A Compound fault labeling and diagnosis method based on flight data and BIT record of UAV. *Applied Sciences*, 11(12), 5410.
- Zongqi, L., Fuyang, C., ve Zili, W. (2022). A data-driven predictive maintenance method for quadrotor propulsion system based on LSTM network. *International Conference on Guidance, Navigation and Control* (s. 6240-6251). Singapore: Springer Nature Singapore.
- http-1: <https://phantompilots.com/attachments/prop-failure-20200929a-jpg.120226/> (Erişim tarihi: 12.05.2025)
- http-2: https://www.donanimhaber.com/images/images/haber/96665/ABDde-alkolluyken-drone-ucurmak-yasaklandi96665_0.jpg (Erişim tarihi: 21.01.2025)
- http-3: <https://www.bilimup.com/vucudumuzdaki-en-uzun-hucreler-sinir-hucrelerinin-yapisi-ve-islevi> (Erişim tarihi: 18.06.2025)

http-4: <https://medium.com/@yasinguzel/yapay-zeka-ders-notlar%C4%B1-03-biyolojik-sinir-sistemi-ve-yapay-sinir-a%C4%9F%C4%B1-h%C3%BCcresi-6555add68d80> (Erişim tarihi: 18.06.2025)

http-5: <https://www.derinogrenme.com/2018/06/28/geri-yayilim-algoritmasina-matematiksel-yaklasim> (Erişim tarihi: 18.06.2025)

http-6: <https://medium.com/software-development-turkey/derin-%C3%B6%C4%9Frenme-hakk%C4%B1nda-neredeyse-her-%C5%9Fey-2-5d35ec484d49> (Erişim tarihi: 20.06.2025)

http-7: https://tr.wikipedia.org/wiki/%C3%87apraz_do%C4%9Frulama_%28istatistik%29 (Erişim tarihi: 23.06.2025)

http-8: <http://colah.github.io/posts/2015-08-Understanding-LSTMs/> (Erişim tarihi: 23.06.2025)

http-9: <https://medium.com/@data-overload/sliding-window-technique-reduce-the-complexity-of-your-algorithm-5badb2cf432f> (Erişim tarihi: 28.06.2025)

EKLER

Ek Tablo 1. *Veri Setine Alınan Özellikler*

Özellik	Birim	İngilizce Tanımı	Türkçe Tanımı
Roll_est	deg	Estimated roll	Tahmini yuvarlanma
Pitch_est	deg	Estimated pitch	Tahmini yunuslama
Yaw_est	degheading	Estimated yaw	Tahmini dönme
Lat	deg	Estimated latitude	Tahmini enlem
Lng	deg	Estimated longitude	Tahmini oylam
Roll	deg	Achieved vehicle roll	Erişilen araç yuvarlanması
Pitch	deg	Achieved vehicle pitch	Erişilen araç yunuslaması
Yaw	deg	Achieved vehicle yaw	Erişilen araç dönmesi
ErrRP	deg	Lowest estimated gyro drift error	Tahmin edilmiş en düşük jiroskopik kayma hatası
ErrYaw	deg	Difference between measured yaw and DCM yaw estimate	Ölçülen dönme ile Yön Kosinüs Matrisi dönme tahmini arasındaki fark
Alt	m	Calculated altitude from GPS	GPS'ten ölçülen yükseklik
CRt	m/s	Derived climb rate from primary barometer	Temel barometreden türetilmiş tırmanma oranı
RMSRollP	-	LPF Root-Mean-Squared Roll Rate controller P gain	Düşük Geçirgen Filtre ortalama karekök yuvarlanma oranı kontrolcüsü P kazancı
RMSRollD	-	LPF Root-Mean-Squared Roll rate controller D gain	Düşük Geçirgen Filtre ortalama karekök yuvarlanma oranı kontrolcüsü D kazancı
RMSPitchP	-	LPF Root-Mean-Squared Pitch Rate controller P gain	Düşük Geçirgen Filtre ortalama karekök yunuslama oranı kontrolcüsü P kazancı
RMSPitchD	-	LPF Root-Mean-Squared Pitch Rate controller D gain	Düşük Geçirgen Filtre ortalama karekök yunuslama oranı kontrolcüsü D kazancı
RMSYaw	-	LPF Root-Mean-Squared Yaw Rate controller P+D gain	Düşük Geçirgen Filtre ortalama karekök dönme oranı kontrolcüsü P+D kazancı
ThI	-	Throttle input	Gaz girişi

Ek Tablo 1. (Devam) Veri Setine Alınan Özellikler

ABst	-	Angle boost	Açı artışı
ThO	-	Throttle output	Gaz çıkışı
ThH	-	Calculated hover throttle	Hesaplanan havada kalma gazı
DAlt	m	Desired altitude	İstenen yükseklik
Alt	m	Achieved altitude	Erişilen yükseklik
BAlt	m	Barometric altitude	Barometrik yükseklik
DCRt	cm/s	desired climb rate	İstenen tırmanma oranı
CRt	cm/s	climb rate	Tırmanma oranı
BatVolt	-	Ratio between detected battery voltage and maximum battery voltage	Batarya voltajının maksimum batarya voltajına oranı
VN	m/s	Estimated velocity (north component)	Tahmini hız (kuzey bileşeni)
VE	m/s	Estimated velocity (east component)	Tahmini hız doğu bileşeni)
VD	m/s	Estimated velocity (down component)	Tahmini hız (alt bileşeni)
dPD	m/s	Filtered derivative of vertical position (down)	Dikey pozisyonun filtrelenmiş türevi (alt)
GX	deg/s	Estimated gyro bias, X axis	X eksen tahmini jiroskop kayması
GY	deg/s	Estimated gyro bias, Y axis	Y eksen tahmini jiroskop kayması
GZ	deg/s	Estimated gyro bias, Z axis	Z eksen tahmini jiroskop kayması
AZbias	cm/s/s	Z accelerometer bias	Z ivmeölçer kayması
PN	m	Estimated distance from origin (North component)	Orijinden tahmini uzaklık (kuzey bileşeni)
PE	m	Estimated distance from origin (East component)	Orijinden tahmini uzaklık doğu bileşeni)
PD	m	Estimated distance from origin (Down component)	Orijinden tahmini uzaklık alt bileşeni)
Vcc	V	Flight board voltage	Uçuş kartı voltajı
C1_in	us	RC channel 1 input	Uzaktan kumanda kanal 1 girişi

Ek Tablo 1. (Devam) Veri Setine Alınan Özellikler

C2_in	us	RC channel 2 input	Uzaktan kumanda kanal 2 girişi
C3_in	us	RC channel 3 input	Uzaktan kumanda kanal 3 girişi
C4_in	us	RC channel 4 input	Uzaktan kumanda kanal 4 girişi
C1_out	us	Servo channel 1 output	Servo kanal 1 çıkışı
C2_out	us	Servo channel 2 output	Servo kanal 2 çıkışı
C3_out	us	Servo channel 3 output	Servo kanal 3 çıkışı
C4_out	us	Servo channel 4 output	Servo kanal 4 çıkışı
VibeX	m/s/s	Primary accelerometer filtered vibration, x-axis	Temel ivmeölçer filtrelenmiş titreşim, x eksen
VibeY	m/s/s	Primary accelerometer filtered vibration, y-axis	Temel ivmeölçer filtrelenmiş titreşim, y eksen
VibeZ	m/s/s	Primary accelerometer filtered vibration, z-axis	Temel ivmeölçer filtrelenmiş titreşim, z eksen

ÖZGEÇMİŞ

ORCID NO: 0000-0001-8250-3143

Ad Soyad : Mehmet ÖZCAN

Yabancı Dil : İngilizce

Eğitim ve Mesleki Geçmişi:

- Lisans Mezuniyeti: 2014, Anadolu Üniversitesi, Mühendislik Fakültesi, Bilgisayar Mühendisliği Bölümü
- Yüksek Lisans Mezuniyeti: 2018, Anadolu Üniversitesi, Fen Bilimleri Enstitüsü, Bilgisayar Mühendisliği Anabilim Dalı
- 2014-2015, Web Yazılımı Mühendisi, S&G Innova
- 2015-2018, Araştırma Görevlisi, Anadolu Üniversitesi, Fen Bilimleri Enstitüsü
- 2018-2019, Araştırma Görevlisi, Balıkesir Üniversitesi, Mühendislik Fakültesi
- 2019-2023, Araştırma Görevlisi, Eskişehir Teknik Üniversitesi, Lisansüstü Eğitim Enstitüsü
- 2023-, Araştırma Görevlisi, Balıkesir Üniversitesi, Mühendislik Fakültesi

Yayınları ve/veya Bilimsel/Sanatsal Faaliyetleri:

- Ozcan, M., ve Perkgoz, C. (2024). Deep learning-based proactive fault detection method for enhanced quadrotor safety. Aviation, 28(3), 175-187.
- Başlangıç Tarihi: 20.05.2021, Bitiş Tarihi: 20.05.2023, 21DRP053, 301-DR1 Tez Projesi, Karmaşık Süreçlerdeki Problemlere Derin Öğrenme Yaklaşımları, Eskişehir Teknik Üniversitesi
- Ozcan, M., ve Goz, F. (2017). An Educational Mobile City Learning Application for Kids. The Eurasia Proceedings of Educational and Social Sciences, 7, 24-29.
- Goz, F., ve Ozcan, M. (2017). An entertaining mobile vocabulary learning application. The Eurasia Proceedings of Educational and Social Sciences, 7, 63-66.
- Ozcan, M., Goz, F., Temel, T., (2016). New Recommender System Using Naive Bayes for E-Learning. W. Wu, S. Alan. ve M. T. Hebeci, (Ed.). Research Highlights in Education and Science, 62-68