



MARMARA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ



KORELASYONA DAYALI AYKIRILIK SAPTAMA VE KESTİRİMCİ BAKIM

ANIL ELADAĞ

YÜKSEK LİSANS TEZİ

Bilgisayar Mühendisliği

Anabilim Dalı

Bilgisayar Mühendisliği

Tezli Yüksek Lisans Programı

DANIŞMAN

Prof. Dr. Ali BULDU

EŞ-DANIŞMAN

Dr. Öğr. Üyesi Kazım YILDIZ

İSTANBUL, 2020



MARMARA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ



KORELASYONA DAYALI AYKIRILIK SAPTAMA VE KESTİRİMCİ BAKIM

ANIL ELADAĞ
(523616023)

YÜKSEK LİSANS TEZİ
Bilgisayar Mühendisliği
Anabilim Dalı
Bilgisayar Mühendisliği
Tezli Yüksek Lisans Programı

DANIŞMAN
Prof. Dr. Ali BULDU

EŞ-DANIŞMAN
Dr. Öğr. Üyesi Kazım YILDIZ

İSTANBUL, 2020

MARMARA ÜNİVERSİTESİ FEN BİLİMLERİ ENSTİTÜSÜ

Marmara Üniversitesi Fen Bilimleri Enstitüsü Yüksek Lisans Öğrencisi Anıl ELADAĞ'ın
"Korelasyona Dayalı Aykırılık Saptama ve Kestirimci Bakım" başlıklı tez çalışması, 22 Ocak 2020
tarihinde savunulmuş ve jüri üyeleri tarafından başarılı bulunmuştur.

Jüri Üyeleri

Prof. Dr. Ali BULDU	(Danışman)	(İMZA) 
Marmara Üniversitesi		
Dr. Öğr. Üyesi Kazım YILDIZ	(Eş-Danışman)	(İMZA) 
Marmara Üniversitesi		
Dr. Öğr. Üyesi Erbil AKBAY	(Üye)	(İMZA) 
Haliç Üniversitesi		
Dr. Öğr. Üyesi Önder DEMİR	(Üye)	(İMZA) 
Marmara Üniversitesi		
Dr. Öğr. Üyesi Zehra Aysun ALTIKARDEŞ	(Üye)	(İMZA) 
Marmara Üniversitesi		

ONAY

Marmara Üniversitesi Fen Bilimleri Enstitüsü Yönetim Kurulu'nun 05.02.2020 tarih ve
2020/05-02 sayılı kararı ile Anıl ELADAĞ'ın Bilgisayar Mühendisliği Anabilim Dalı Bilgisayar
Mühendisliği Programında Yüksek Lisans derecesi alması onanmıştır.

Fen Bilimleri Enstitüsü Müdürü

Prof. Dr. Bülent EKİCİ




ÖNSÖZ

Öncelikle tez konusunu seçerken isteklerimi göz önünde bulundurup bana yardımcı olan tez danışmanlarım Prof. Dr. Ali BULDU ve Dr. Öğr. Üyesi Kazım YILDIZ'a teşekkürlerimi sunarım. Bu zorlu tez sürecinde benden desteğini bir an için bile esirgemeyen değerli arkadaşım, Ekrem EROL'a ve tüm eğitim hayatım boyunca benden maddi ve manevi desteklerini esirgemeyen her zaman yanımda olan sevgili aileme teşekkürlerimi bir borç bilirim.

Ocak 2020

Anıl ELADAĞ



İÇİNDEKİLER

ÖNSÖZ.....	I
İÇİNDEKİLER	II
ÖZET	IV
ABSTRACT	V
KISALTMALAR	VI
ŞEKİL LİSTESİ.....	VII
TABLO LİSTESİ.....	VIII
1. GİRİŞ	1
1.1. Kestirimci Bakım Çalışmaları	3
1.2. Kestirimci Bakım ve Yapay Sinir Ağları	3
1.3. Arıza Tespit Yaklaşımları	6
1.3.1. Kural tabanlı arıza tespit yöntemi	6
1.3.2. Tahmine dayalı arıza tespit yöntemi	7
1.3.3. Zaman serisi tabanlı arıza tespit yöntemi.....	7
1.3.4. Öğrenmeye dayalı arıza tespit yöntemi.....	7
1.4. Bakım Yöntemleri	8
1.4.1. Kestirimci bakım yaklaşımı	8
1.5. Korelasyon ve Aykırılık Tespiti.....	9
1.5.1. Korelasyon	9
1.5.2. Aykırılık Tespiti	11
1.6. Kestirimci Bakım Adımları.....	12
2. MATERYAL VE YÖNTEM.....	15
2.1. Çalışmada Kullanılan Donanımlar	15
2.2. Çalışmada Kullanılan Yazılımlar	15
2.3. Çalışmada Kullanılan Teknikler ve Veri Seti	16
2.4. Yapay Sinir Ağı Mimarileri	20
2.4.1. Uzun-Kısa Süreli Hafıza Mimarisi	20
2.4.2. Dense Ağlar.....	24
2.4.3. Evrişimli Sinir Ağı	25
2.4.4. Hibrit (Karma) Sinir Ağı.....	27
3. BULGULAR VE TARTIŞMA.....	29
3.1. Tahmin Durumları ve Yapı Modelleri.....	30
3.1.1. CNN Modeli.....	30

3.1.2. Dense Modeli	32
3.1.3. LSTM Modeli	34
3.1.4. Hibrit Model	36
3.2. Başarı, Kayıp ve Hata Oranları.....	38
3.2.1. Başarı Oranları	38
3.2.2. Kayıp (Loss) Oranları.....	38
3.2.3. Ortalama Mutlak Hata Oranları.....	39
3.2.4. Ortalama Kare Hatası Oranları.....	40
3.3. Tartışma.....	41
4. SONUÇ.....	43
KAYNAKÇA	45
ÖZGEÇMİŞ.....	51

ÖZET

KORELASYONA DAYALI AYKIRILIK SAPTAMA VE KESTİRMÇİ BAKIM

Üretim ekipmanları genellikle planlı bir bakım yaklaşımı olmadan kullanılmaktadır. Böyle bir strateji, beklenmeyen arızalar nedeniyle sıklıkla planlanmamış aksama sürelerine neden olur. Zamanlanmış bakım, beklenmeyen ekipman durmalarını önlemek için bileşenleri sık sık değiştirir, ancak makinenin çalışmaması ve bakım maliyeti ile ilişkili süreyi artırır. Endüstri 4.0 ve akıllı sistemlerin ortaya çıkması, aksama süresini azaltabilecek ve imalat ekipmanının kullanılabilirliğini (kullanım oranını) artırabilecek olan kestirimci bakım, ya da Kestirimci Bakım (Predictive Maintenance - PdM) stratejilerine artan ilgi gösterilmesine yol açmaktadır.

Kestirimci Bakım (Predictive Maintenance - PdM) son yıllarda yaygın olarak kullanılmakta olan bir uygulamadır, veri analizi ile cihazlarda daha sorun oluşmadan önce, durumu izleyen ve arıza çıkarma ihtimallerini değerlendirip, önlem alan bir bakım metodudur. Arıza olasılığını azaltmak için normal çalışma sırasında ekipmanın performansını ve durumunu izler. Kestirimci bakım ayrıca, bileşenlerin kullanım ömrünü en üst düzeye çıkararak üretimde sürdürülebilir uygulamaları geliştirme potansiyeline sahiptir. Üretim, altyapı, sağlık ve enerji gibi çeşitli endüstrilerde verimliliği artırmak ve işletme maliyetlerinden tasarruf sağlayabilmek için kullanılmaktadır.

Bu tez çalışmasında sık kullanılan regresyon metotları yerine yapay sinir ağları kullanılarak sonuçlar gözlemlenmiş ve bunların diğer regresyon modelleri ile performans açısından karşılaştırılması gösterilmeye çalışılmıştır.

Anahtar Kelimeler: Kestirimci bakım, yapay sinir ağı, aykırılık tespiti, korelasyon

ABSTRACT

CORRELATION BASED ANOMALY DETECTION AND PREDICTIVE MAINTENANCE

Production equipment is often used without a planned maintenance approach. Such a strategy often leads to unplanned downtime due to unexpected failures. Scheduled maintenance frequently replaces components to prevent unexpected equipment downtime, but increases the time associated with machine downtime and maintenance cost. The emergence of Industry 4.0 and intelligent systems has led to increased interest in Predictive Maintenance (PdM) strategies that can reduce downtime and increase the availability (utilization) of manufacturing equipment.

Predictive Maintenance (PdM) is a widely used application in recent years. It is a maintenance method that monitors the situation and evaluates the possibility of failure before data problems occur. Monitors the performance and condition of the equipment during normal operation to reduce the possibility of failure. Predictive maintenance also has the potential to develop sustainable applications in production by maximizing the lifetime of the components. It is used in various industries such as production, infrastructure, health and energy to increase productivity and save operating costs.

In this thesis, using artificial neural networks instead of commonly used regression methods, the results were observed and their comparison with other regression models in terms of performance was tried to be shown.

Keywords: Predictive maintenance, artificial neural network, detection of contradiction, correlation

KISALTMALAR

AI	: Artificial Intelligence
YZ	: Yapay Zekâ
CNN	: Convolutional Neural Network (Evrışimsel Sinir Ağları)
GB	: Gigabyte
GDDR5	: Graphics Double Data Rate, version 5 (Grafik, çift kanallı rastgele erişimli hafıza, sürüm 5)
GPU	: Graphics Processing Unit (Grafik İşlemci Ünitesi)
IOT	: Internet of Things (Nesnelerin İnterneti)
IIOT	: Industrial Internet of Things (Endüstriyel Nesnelerin İnterneti)
LCAD	: Latent Correlation Anomaly Detection (Gizli Korelasyon Anomali Tespiti)
LSTM	: Long-Short Term Memory (Uzun-Kısa Süreli Hafıza)
MAE	: Mean Absolute Error (Ortalama Mutlak Hata)
MSE	: Mean Squared Error (Ortalama Kare Hatası)
NLP	: Natural Language Processing (Doğal Dil İşleme)
PdM	: Predictive Maintenance (Kestirimci Bakım)
RUL	: Remaining Useful Life (Kalan Faydalı Ömür)

ŞEKİL LİSTESİ

Şekil 1.1	Yapay Sinir Ağı Modeli	5
Şekil 1.2	Kestirimci Bakım Yöntemlerinin Tercih Edilme Oranları [30]	9
Şekil 1.3	PdM Akış Şeması [38].....	13
Şekil 2.1	Visual Studio Code Arayüzü Örneği.....	16
Şekil 2.2	Turbofan Engine Degradation Simulation Veri Seti Örnek Verileri.....	17
Şekil 2.3	Veri Seti Örnek Sensör Verisi Histogramları.....	19
Şekil 2.4	LSTM Döğümü Yapısı	20
Şekil 2.5	LSTM Unutma Kapısı	21
Şekil 2.6	LSTM Girdi Kapısı.....	22
Şekil 2.7	LSTM Döğüm Hafızası	23
Şekil 2.8	LSTM Çıktısı.....	24
Şekil 2.9	Dense Ağ Mimarisi.....	25
Şekil 2.10	CNN Mimarisi	26
Şekil 3.1	Çeşitli Makine Öğrenmesi Yöntemleri İle Elde Edilen Sonuçlar [50].....	29
Şekil 3.2	Yinelemeli Sinir Ağları ile Tahminlenen RUL ve Kayıp Oranları [51]....	29
Şekil 3.3	CNN İki Tekrar ve Relu Aktivatörlü Tahminleme Sonucu.....	30
Şekil 3.4	Evrişimli Sinir Ağı	31
Şekil 3.5	Dense İki Tekrar ve Relu Aktivatörlü Tahminleme Sonucu	32
Şekil 3.6	Dense Sinir Ağı	33
Şekil 3.7	LSTM İki Tekrar ve Relu Aktivatörlü Tahminleme Sonucu	34
Şekil 3.8	LSTM Sinir Ağı.....	35
Şekil 3.9	Hibrit İki Tekrar ve Relu Aktivatörlü Tahminleme Sonucu	36
Şekil 3.10	Hibrit Sinir Ağı.....	37
Şekil 3.11	Kayıp (Loss) Oranlarının Karşılaştırması	39
Şekil 3.12	Ortalama Mutlak Hata Oranlarının Karşılaştırması	40
Şekil 3.13	Ortalama Kare Hatası Oranlarının Karşılaştırması.....	41
Şekil 3.14	Yapay Sinir Ağları RUL Karşılaştırması	42

TABLO LİSTESİ

Tablo 1.1 LCAD ve Diğer Yöntemlerin Karşılaştırması [40].....	13
Tablo 2.1 Turbofan Motoru Veri Seti Nitelikleri.....	18
Tablo 3.1 Dense, CNN, LSTM ve Hibrit Ağların Başarı Oranı Karşılaştırması	38
Tablo 3.2 Dense, CNN, LSTM ve Hibrit Ağların Kayıp Oranı Karşılaştırması	39
Tablo 3.3 Dense, CNN, LSTM ve Hibrit Ağların MAE Değerlerinin Karşılaştırması	40
Tablo 3.4 Dense, CNN, LSTM ve Hibrit Ağların MSE Değerlerinin Karşılaştırması	41



1. GİRİŞ

Yapay zekâ, büyük veri, veri analizi, bulut bilişim, nesnelerin interneti (Internet of Things – IoT) platformu, vb gibi siber sistemlerin, gelişen bilgi işlem altyapıları ile bütünleşmeleri, 21. yüzyılda akıllı üretimin yani Endüstri 4.0'ın güçlenmesini sağlamıştır.

Birçok elemanın (örneğin insan, maddi ve maddi olmayan kaynakların) birbirleriyle etkileşime girdiği karmaşık imalat alanında [1], imalat işlemleri sırasında büyük miktarda veri toplanır. İşlenmiş üretim verilerinden haberdar olan bir bilgi işlem altyapısı önceden eğitilmiş yapay zekâ (YZ - Artificial Intelligence - AI) algoritmaları ile kontrol edilebilir.

Üretim verilerinden faydalı bilgiler elde etmek için AI teknikleri yaygın olarak kullanılmaktadır. AI teknikleri, eğitim ile tarihsel tecrübeyi kullanarak değişen çevreye otomatik olarak öğrenmek ve adapte olmak için zekâyı sistemlere akıtmaktadır [2]. Buna ek olarak, yüksek boyutlu verileri kullanılması, karmaşıklığın azaltılması, mevcut verilerin geliştirmesi ve ilgili süreç ilişkilerini belirleme becerisi, AI tekniklerin imalat endüstrisinde uygulanabilirliğini göstermek için vurgulanmaktadır [3]. Üretim hattındaki çeşitliliği azaltmak, üretkenliği ve ürün kalitesini iyileştirmek için, üreticinin AI desteği ile konuyu tahmin etme imkânı ortaya çıkmaktadır. Bu sayede, üretim sisteminin gelecekteki davranışına, sisteme AI algoritmaları uygulanarak yaklaşılabılır ve bu oluşturulan bilgi karar vermede yardımcı olabilir.

Çıkarılan anlamlı bilgi; enerji ve kaynak verimliliği [4], atık yönetimi [5], kestirimci bakım (Predictive Maintenance - PdM) [6] gibi imalat endüstrisinde sürdürülebilir uygulamalara doğru dönüşüme yardımcı olabilecek, daha iyi bir karar verilmesine yönelik öngörülerini sağlamaktadır. Akıllı üretim platformunu kullanarak makine bakım görevlerini optimize etmek için makine ve bakım mühendisleri arasında bir iletişim aracı geliştirmek, gelişmiş endüstriyel sürdürülebilirliği sağlamak için bir yaklaşımdır. Üretim tesisinde, sistem güvenilirliğini sağlamak, maliyeti düşürmek, aksaklık süresinden kaçınmak ve bir parçanın kullanım ömrünü en üst düzeye çıkarmak için en uygun bakım stratejileri gereklidir [6].

Planlanmamış ve önleyici bakımlar değiş-tokuş (tradeoff) senaryosuna sahipken, kestirimci bakım, bir bileşenin ve çalışma süresinin eşzamanlı olarak kullanım ömrünü en üst düzeye çıkararak travmayı kırabilen gelecek vaat eden bir tekniktir [7]. Hizmet içi ekipmanın durumunu izlemek ve ardından ekipmanın ne zaman başarısız olacağını tahmin etmek için tasarlanmıştır. Bu, makine bileşenlerinin gelecekteki arızalarının yaklaşılabileceği anlamına gelir ve bu da bakım görevlerini optimize etmeye yardımcı olur. Buna bağlı olarak, makinenin arıza süresi ve bakım maliyeti, bakım sıklığını mümkün olduğu kadar düşük tutarken önemli ölçüde azaltılabilir.

Sensörler yaklaşmakta olan arızayı konvansiyonel kestirimci bakım programlarından çok daha erken bir zamanda tespit etmek için kullanılmaktadır. Sensörler aynı zamanda zorlu üretim ortamlarındaki uygulamaların yanı sıra nakliye, dağıtım ve depo filosu yönetimi potansiyeline sahiptir [8].

Üretim tesislerinin operasyonel maliyetlerinin büyük bir kısmı bakım giderleridir ve sektöre bağlı olarak üretim maliyetinin yüzde 15 ile 60'lık kısmını oluşturmaktadır [9]. Bu giderlerin azaltılması ve karlılığı arttırmak için düzgün bir bakım planı ve üretimi durdurmamak önemlidir. Buna yönelik olarak bakım planları yapılmaktadır. Bakım planları işletmeden işletmeye farklılık gösterebilir. Düzeltici, periyodik, önleyici ve kestirimci bakım olarak üzere 4'e ayrılmaktadır [10].

Tez çalışmasının Giriş bölümünde, kestirimci bakım ve sensörlerin günümüz kullanımına ilişkin bilgiler ve bu tez çalışmasının amacı açıklanmaktadır. Kestirimci bakım, akıllı sensörler, IIoT (Industrial Internet of Things - Endüstriyel Nesnelerin İnterneti) kavramı ve arıza tespit yaklaşımları ile ilgili bilgilere yer verilmektedir.

Malzeme ve Yöntem bölümünde, kullanılan programlar ve yöntemler ile ilgili bilgiler paylaşılmıştır. Tez çalışmasında kullanılan platformlar, veri setleri, uygulama sürecinde izlenen yollar ve kullanılan donanımlar bu bölümde sunulmaktadır.

Bulgular, Tartışma ve Sonuç bölümünde ise, veri setleri ile yapılan analizler ve bunun sonucunda maliyet, performans kategorileri üzerinden sonuçları paylaşılmaktadır. Tez çalışması ile elde edilen bulgulara dayanarak çıkarılan sonuçlar verilmektedir.

1.1. Kestirimci Bakım Çalışmaları

Literatürde “klasik” kestirimci bakım ve durum izlemesi [11, 12] ile ilgili çeşitli makaleler ve çalışmalar bulunmaktadır. Tahmini bakım stratejisi, genellikle uygun şekilde seçilmiş bir harici sensör seti ve çevrimiçi durum izlemeye dayanan kural tabanlı bir bakımdır. Ne yazık ki, sensörlerin kendi durumunu izlemenin gerekli olması durumunda, bu tekniklerin hiçbiri kullanışlı olmamaktadır. Bu süreçte uygun sensör seti anahtar rol oynamaktadır [12]. Dahası, yayınlanan birçok çalışma yaklaşımlarını sensör ağlarına dayandırır. Arıza ağdaki sensörlerin ölçümleri kullanılarak tespit edilebilir.

Bu tez çalışmasının amacı, sensörlerden anlık olarak zaman serisi şeklinde gelen verileri kullanarak, cihazlar arasındaki korelasyonun bulunması ve bu korelasyona göre normalden farklı olan aykırı durumları bulma temelli bir PdM yapısı kurulmasıdır. Bu yolla, gerçek zamanlı olarak normal bir PdM yapısında, çıkan sorunu bulmanın yanı sıra, buna ek olarak bu sorunun kaynağının başka bir cihaz olup olmadığı bulunarak sorunun temelden çözülmesine imkân sağlanması amaçlanmaktadır. Geliştirilen bu sistem ile sorunların tekrarının azalacak olması ve kök sebeplerinin bulunabilir olması önemlidir.

1.2. Kestirimci Bakım ve Yapay Sinir Ağları

Endüstri 4.0 ve akıllı sistemlerin ortaya çıkması, aksama süresini azaltabilecek ve imalat ekipmanının kullanılabilirliğini (kullanım oranını) artıracabilecek olan öngörücü bakım, ya da PdM stratejilerine artan ilgi gösterilmesine yol açmaktadır. Kestirimci bakım, arıza olasılığını azaltmak için normal çalışma sırasında ekipmanın performansını ve durumunu izleyen bakımdır. Kestirimci bakım ayrıca, bileşenlerin kullanım ömrünü en üst düzeye çıkararak üretimde sürdürülebilir uygulamaları geliştirme potansiyeline sahiptir. Artık prognostik; kalan işlevsel zaman (Remaining Useful life – RUL) tahminlemesi gibi kritik hasar ve maliyetlerden kaçınan bakım stratejilerinde ana işlem olarak kabul görmektedir [13]. Bu bağlamda karmaşık üretim alanlarında; insan, robot, sensör gibi somut ve soyut birçok kaynağın birbiri ile etkileşim içerisinde [1] olması dolayısı ile de çeşitli prognostik yöntemler ortaya çıkmıştır. Bu yöntemler; model temelli, veri temelli ve deneyim temelli olmak üzere üç kategoride sınıflandırılmaktadırlar [14-16]. Üretimdeki çeşitlilik ve araçların fazlalığı sebebi ile model temelli yapıları oluşturmak oldukça zordur

ve deneyim temelli bir yöntem ortaya koymak da oldukça uzun zaman ve bilgi birikimi gerektirmektedir. Bu yüzden sıklıkla veri temelli yöntemler kullanılmaktadır. Bu yöntemler genellikle Chao Hu ve arkadaşlarının çalışmasında [17] kullanılan kolektif öğrenmeye dayalı bir regresyon modeli, Wo Jae Lee ve arkadaşlarının çalışmasında [4] ise yapay sinir ağları yöntemleri kullanılarak gerçekleştirilmiştir.

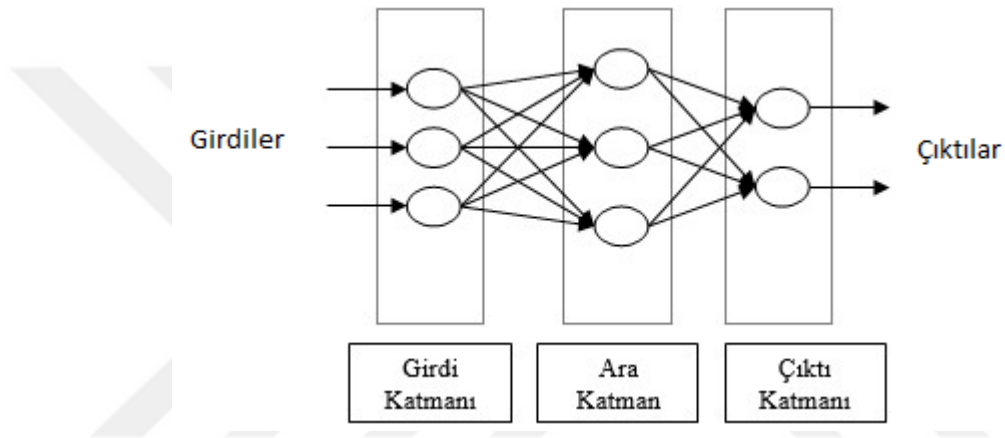
Yapay sinir ağları beyinin biyolojik yapısı örnek alınarak geliştirilmiş bir yapay zekâ metodudur [18]. Matematiksel işlem yapan hücreler, girdi karşılığında çıktı üretirler. Yapay sinir ağları genel olarak sınıflandırma, modelleme, gruplama, örüntü tanıma, görüntü işleme gibi alanlarda kullanılırlar. Görsel korteks temel alınarak ilk olarak Kunihiko Fukushima tarafından oluşturulmuş olan [19] ve görüntü işleme işlemlerinde sıkça kullanılan yapay sinir ağı türü olan Evrişimsel Yapay Sinir Ağı (Convolutional Neural Network - CNN), genel matris çarpım işlemleri yerine konvolüsyon (evrişim) kullanır. Buradaki evrişim işlemi; iki serinin birbirine göre benzerliğidir ve matematiksel olarak serilerin iç çarpımlarına/çapraz korelasyonlarına eşittir. Bir yineleyen yapay sinir ağı türü olan ve Sepp Hochreiter tarafından oluşturulan [20] Uzun Kısa-Vadeli Hafıza (Long Short-Term Memory - LSTM) ise zaman bağımlı serilerde aykırılık tespiti için tercih edilmektedir [21].

Şekil 1.1’de örnek yapay sinir ağı görülmektedir. Girdi katmanı aracılığıyla sisteme alınan veriler ara katmanlara ve son olarak çıktı katmanına aktarılır. Katmanlardaki düğümler matematiksel hesaplamalar ile çıktı üretirler. Genellikle bir düğümdeki bu matematiksel işlemler; ağırlıkların hesaplandığı Denklem 1.1’deki lineer fonksiyon ve genelde lineer olmamasına özen gösterilen Denklem 1.2’deki gibi bir aktivasyon fonksiyonundan oluşmaktadır.

$$\sum_{i=0}^n w_i x_i + b \quad (1.1)$$

$$f(x) = \frac{1}{(1 + e^{-x})} \quad (1.2)$$

Denklem 1.1’de W değişkeni ağırlıkları ifade etmektedir. Her bir düğüme ait ağırlık bilgisi bu değişkende tutulmaktadır. X değişkeni düğümün girdisidir. B ise lineer bir sayıdır, başlangıçta 0 olabileceği gibi denklemin hata oranına göre değiştirilmektedir. Denklem 1.2’de “e” Euler değerini ifade etmektedir.



Şekil 1.1 Yapay Sinir Ağı Modeli

Geliştirilmiş olan bu yapay sinir ağı modellerinden yola çıkılarak kestirimci bakım uygulamalarında kalan ömrün tahmin edilmesini iyileştirmek için sensörlerin korelasyonları ve sensör verilerinin aykırılıklarının kullanılması amaçlanmıştır. Bu sayede kalan ömür tahminlemesine ek olarak, aykırılık kaynaklı sorunların, kalan ömüre etki eden kök sebebin bulunması ve sorun tekrarının azaltılması hedeflenmiştir.

Tahmine Dayalı Bakım (Predictive Maintenance, PdM), geniş bir sensör yelpazesinden gelen verilerin toplanmasına dayanır ve bu, oluşmadan önce olası arızaları belirleyebilecek istatistiksel bir model oluşturur. Bu yaklaşım, ekipmanın sağlık trendlerini takip etmek için genellikle bulutta, istatistiksel analiz ve makine öğrenimi gibi geniş bir yelpazedeki araçları kullanabilir [22].

Bu günlerde, kestirimci bakım; IIoT teknolojisinin ortaya çıkmasından kaynaklanan, endüstri oyun değiştiricisi olarak görülmektedir. Accenture, New York City

(accenture.com) ve GE, Boston (ge.com) tarafından yapılan son bir araştırma; kestirimci bakımın, bakım maliyetlerinde %30'luk bir düşüş ve ekipman arızaları nedeniyle üretimin aksama süresinde %70'lik bir düşüş sağlayabileceğini göstermektedir [23].

1.3. Arıza Tespit Yaklaşımları

Optimum olarak seçilip uygulandığında, en yeni nesil akıllı sensörler, çeşitli uygulamalar için kestirimci bakım stratejileri üzerinde olumlu bir etkiye sahip olabilmektedir. Akıllı sensörler aracılığıyla gerçek zamanlı veri toplama, bakım işlemlerini gerçekleştirmek için kesinti zamanını tasarlanmasıyla ilgili kararları etkileyerek, üretkenliği en üst düzeye çıkarmaya yardımcı olabilmektedir [9].

Günümüzde bakım personelleri akıllı sensörü bir bilgisayara bağlayarak sorunun nedenini belirleyebilmektedir. Bilgisayar, izlenen hangi makine bileşeninin arızayı ürettiğini gösterebilmektedir. Gerekli bakım daha sonra programlanabilir. Arızalı makine bileşeni onarıldıktan veya değiştirildikten sonra, sensör çıkışı etkin olmayan bir duruma dönebilmektedir.

Arıza tespiti için dört farklı yaklaşım sınıfı tartışılmaktadır [24]. Kural Tabanlı Yöntemler, Tahmine Dayalı Yöntemler Zaman Serisi Tabanlı Yöntemler ve Öğrenmeye Dayalı Yöntemler. Bu bölümde bu yöntemler açıklanmaktadır.

1.3.1. Kural tabanlı arıza tespit yöntemi

Kural Tabanlı Yöntemler, sensör okumalarının yerine getirmesi gereken sezgisel kısıtlamaları geliştirmek için alan bilgisini kullanır. Bu kısıtlamaların ihlali, hata anlamına gelir.

Denklem 1.3'de örnek okumaların w_{size} bir pencere içindeki varyansı (σ^2 veya standart sapma σ) hesaplanır. Belli bir eşik üstünde ise, örnekler gürültü hatası nedeniyle bozulur. Varyans sıfır ise, örnekler sabit hata nedeniyle bozulur. Kısa gürültü arızalarını tespit etmek için verilerin uygun şekilde önceden işlenmesi gerekiyordu. Değişim oranı bir eşik üstünde ise, verilerin kısa arızalardan etkilendiği varsayılabilir [25].

$$\sigma^2 = \sum_{i=k}^{w_{size}+k} \frac{(X_i - \bar{X})^2}{w_{size} - 1} \quad (1.3)$$

Bu yöntemin performansı büyük oranda w_{size} ve eşik değerlerine bağlıdır. Parametre ayarı önemsiz değildir ve genellikle incelenen sorunun alan bilgisini gerektirir.

1.3.2. Tahmine dayalı arıza tespit yöntemi

Fiziksel bir olgu aynı anda birden fazla sensör tarafından algılandığında tahmine dayalı yöntemler kullanılabilir ve bireysel sensör ölçümleri için tahminler üretmek için sensör ölçümleri arasındaki bağımlılıktan yararlanılabilir [9]. Bağımlılık mekansal korelasyon ile ifade edilebilmektedir. Korelasyonun nedeninden bağımsız olarak normal davranışı modellemek için kullanılmaktadır. Tahmin, örneğin Doğrusal En Küçük Kareler Tahmini ile yapılmaktadır [26]. Bu yöntem, olguların neredeyse aynı sensörler tarafından algılandığı durumlar için en uygun yöntemdir. Örnek olarak, tek bir uçakta birden fazla barometrik altimetre bulunur. Bu altimetrelerden okunan değerlerin güçlü bir şekilde ilişkilendirildiğine dair güçlü bir varsayım vardır.

1.3.3. Zaman serisi tabanlı arıza tespit yöntemi

Zaman Serisi Tabanlı Yöntemler, bir sensörün ölçümlerinin rastgele olmadığı ve bu nedenle bir tür düzenli desenler içerdiği gerçeğini kullanır [27]. Bu modeller, tek bir sensör tarafından toplanan ölçümlerde oto korelasyonlar yoluyla tanımlanmaktadır. Bunlar, algılanan olayların gerileyen bir modelini oluşturmak için kullanılmaktadır. Bir sensör ölçümü, hatalı olup olmadığını belirlemek için öngörülen değer ile karşılaştırılabilir [28]. Bu yaklaşımın sınıflandırmadan daha genel olması ve etiketli veri mevcut olmasa veya çok güçlü şekilde ilişkili algılayıcılar olmasa bile kullanılabilir olması diğer yöntemlerden avantajlı kılmaktadır.

1.3.4. Öğrenmeye dayalı arıza tespit yöntemi

Öğrenmeye Dayalı Yöntemler, "normal" sensör davranışının modelini ortaya çıkarmak için eğitim verilerini kullanır. "Normal" sensör davranışı ve sensör arızalarının

etkileri iyi anlaşılıyorsa, sensör arızalarını tespit etmek ve sınıflandırmak için öğrenme tabanlı yöntemler uygun olmaktadır. [29]'de Sharma ve arkadaşları başarıyla bir sensör ölçüm modeli oluşturmak için Saklı Markov Modellerini kullanmaktadır. Öğrenmeye dayalı yöntemlerin temel avantajı, hataları aynı anda tespit edip sınıflandırabilmeleridir.

1.4. Bakım Yöntemleri

İşletmeden işletmeye farklılık gösterse de bakım yöntemleri genel olarak 4 ana başlık altına toplanmaktadır [9];

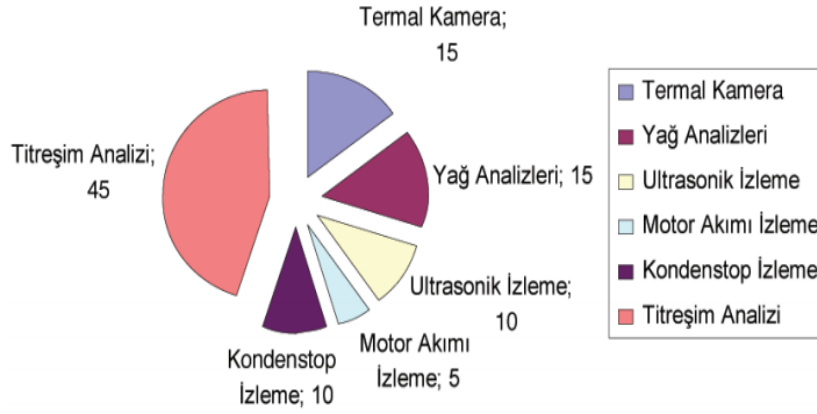
- Düzeltici Bakım: Makineler arızalandıktan sonra arızaların düzeltilmesi için yapılan bakımdır.
- Periyodik Bakım: Makineler arızalana kadar değil belirli aralıklarla yapılan bakım çeşididir.
- Önleyici Bakım: Bu bakım çeşidinde; makinelerde sıklıkla sorun çıkarabilecek parçalar belirlenir ve parçaların sorun çıkarmasına sebep olabilecek problemler ortadan kaldırılır.
- Kestirimci Bakım: Periyodik olarak çalışır durumdaki makinalardan toplanan veriler üzerinde analizler yapılarak, sistemi durdurmadan, arıza oluşmadan önce arızanın tahminlenmesi ve uygun bir zamanda bakımının yapılmasıdır.

1.4.1. Kestirimci bakım yaklaşımı

Kestirimci bakım yaklaşımının bir işletmede uygulanabilmesi için çeşitli yöntemler kullanılmalıdır. Bu yöntemleri şu şekilde sıralanmaktadır [9]:

- Kızıl Ötesi Termografi: Termal kamera ile sıcaklığın takibi
- Yağ Analizleri: Yağların akışkanlık, ısı vb. takibinin yapılması
- Ultrasonik İzleme: Ultrasonografi yöntemi ile elde edilen görüntülerin incelenmesi
- Motor Akımı İzleme
- Kondensatör İzleme: Sistemde oluşan ve atılan buharın takibi
- Titreşim Analizi: Hareketli parçaların oluşturduğu titreşimlerin takibi

Bu yöntemler içerisindeki tercih edilme oranları Şekil 1.2 Kestirimci Bakım Yöntemlerinin Tercih Edilme Oranları 'de görülmektedir.



Şekil 1.2 Kestirimci Bakım Yöntemlerinin Tercih Edilme Oranları [30]

1.5. Korelasyon ve Aykırılık Tespiti

1.5.1. Korelasyon

Olasılık kuramı ve istatistikte iki rassal değişken arasındaki doğrusal ilişkinin yönünü ve gücünü belirten değişken korelasyondur [27]. İstatistiksel kullanımda korelasyon, bağımsızlık durumundan ne kadar uzaklaşıldığını gösterir [28]. Bu tez kapsamında korelasyon; sensörler ve makineler arasındaki ilişkiyi belirtmektedir. Makineler genelde birden çok sensörden oluşmaktadır ve bu sensörler de birbirleri ile bağıntılıdır. Aynı şekilde makineler de birbirleri ardı sıra ya da beraber çalıştığı için aralarında ilişki içindedirler.

İki değişken arasındaki ilişki miktarı hesaplanırken, ikili ya da basit korelasyon teknikleri kullanılmaktadır. Çoklu korelasyon tanımlanırken, bir değişkenin iki ya da daha çok değişken ile olan ilişkisi incelenir; değişkenlerden biri sabitlenerek diğer değişkenler ile olan ilişkisi kısmi korelasyon teknikleriyle hesaplanır.

Korelasyon katsayısının kullanılması belirlenme aşamaları şu şekilde sıralanmaktadır;

- Sürekli veya süreksiz değişken olması,
- Değişkenlerin ölçek düzeyi ölçümü
- Verilerin doğrusallığına göre değişmektedir.

Basit (ikili) korelasyon, bir bireye ait iki ölçüm arasındaki ilişkiyi belirlemektedir. Korelasyon katsayısı kullanılarak ilişki olup olmadığı ve varsa bu ilişkinin derecesi hesaplanır.

Korelasyon katsayısı “r” ile ifade edilmektedir ve -1 ile +1 arasında değerler alır.

a) Pearson Momentler Çarpımı Korelasyon Katsayısı (r)

İki sürekli değişken arasındaki en az eşit aralıklı ölçek düzeyinde ölçülen doğrusal ilişkinin derecesinin belirlenmesinde kullanılmaktadır. ‘*değişkenler arasındaki bir ilişki var mıdır?*’ sorusunun cevabı aranır [31]. Değer aralığı -1 ile +1 arasında değişmektedir.

- $r = 0$ ise iki değişken arasında ilişki yoktur.
- $r = -1$ ise tam negatif doğrusal bir ilişki vardır.
- $r = +1$ ise tam pozitif doğrusal bir ilişki vardır.

b) Spearman Brown Sıra Farkları Korelasyon Katsayısı (ρ , r_s)

İki değişken arasındaki doğrusal ilişkiyi sıralı puanlar kullanarak açıklamaktadır [30]. Eşit aralıklı ya da eşit oran düzeyindeki ölçümler sıralı ölçeğe dönüştürülerek analiz edilmesi gerektiğinde de ölçümler sıralı ölçeğe dönüştürülerek uygulanabilir. Pearson korelasyonundaki değişkenler, normal dağılım varsayımını sağlamadığında Pearson korelasyon yerine kullanılmaktadır.

c) Nokta Çift Serili Korelasyon Katsayısı

Sürekli (en az aralık ölçeğinde) değişken ile iki kategorili gerçek süreksiz (cinsiyet gibi) değişken arasındaki doğrusal ilişkiyi tanımlamaktadır [32].

d) Çift Serili Korelasyon Katsayısı

Sürekli ancak yapay olarak iki kategorili (yaşı iki kategoriye ayırmak gibi) süreksiz duruma getirilen bir değişken ile sürekli bir değişken arasındaki ilişkinin hesaplanmasında kullanılmaktadır [32].

e) Dörtlü Korelasyon Katsayısı

İki kategorili iki süreksiz değişken arasındaki ilişkinin ölçülmesinde kullanılmaktadır [28].

1.5.2. Aykırılık Tespiti

Aykırı değer tespiti, aykırı değerlerin belirli bir veri kümesinde belirlenmesi ve ardından bu kümeden çıkarılması işlemidir.

Aykırılık, veri setinin verilen norm veya ortalamasından büyük ölçüde sapan bir veri parçası veya gözlem olarak tanımlayabilir. Aykırılık, basitçe tesadüfen kaynaklanmış olabilir, ancak aynı zamanda ölçüm hatasını veya verilen veri setinin yoğun kuyruklu bir dağılıma sahip olduğunu gösterebilir [33].

Bir aykırılığı belirlemek için standardize edilmiş, katı bir matematiksel yöntem yoktur. Bunun sebebi; aykırılığın küme veya veri popülasyonuna bağlı olarak değişmesi ve bu yüzden de belirlenmesi ve tespitinin öznelleşmesidir. Fakat belirli bir veri alanında sürekli örnekleme yoluyla, algılamayı kolaylaştırmak için bir ayracın özellikleri belirlenebilir [34].

Aykırı değerlerin algılanması için model tabanlı yöntemler vardır ve verilerin tümünün normal bir dağılımdan alındığını ve aykırı değerler olarak ortalama veya standart sapmalara dayanmalarının muhtemel olmadığı gözlemleri veya noktaları tanımlayacağını varsaymaktadırlar. Aykırı değer tespiti için kullanılan yöntemler 4'e ayrılmaktadır:

a) Grubb'un Outliers Testi:

Verilerin normal bir dağıtım olduğu varsayımına dayanmaktadır ve testte artık aykırı değer bulunamayana kadar yinelenen test ile bir aykırı değeri kaldırmaktadır [34]. Denklem 1.4'de görünen $\bar{Y}$ örneklerin ortalamasını ve S standart sapmayı ifade etmektedir

$$G = \frac{\max_{i=1,\dots,N} |Y_i - \bar{Y}|}{S} \quad (1.4)$$

b) Dixon'ın Q Testi:

Veri setinin normal olmasına bağlı olarak, bu yöntem kötü verileri test etmektedir. Bunun bir veri setinde az miktarda ve asla bir defadan fazla kullanılması gerektiği not edilmiştir [27]. Denklem 1.5'te görüldüğü gibi en yakın komşu ile olan uzaklığın referans aralığına oranı şeklinde hesaplanmaktadır.

$$Q = \frac{\text{gap}}{\text{range}} \quad (1.5)$$

c) Chauvenet'in Kriteri:

Aykırlığın sahte olup olmadığını veya hala sınırlar dâhilinde olup olmadığını ve setin bir parçası olarak kabul edilip edilmediğini analiz etmek için kullanılmaktadır [35]. Ortalama ve standart sapma alınır ve aykırı oluşma olasılığı hesaplanır. Sonuçlar dâhil edilip edilmemesi gerektiğini belirleyici özelliktedir.

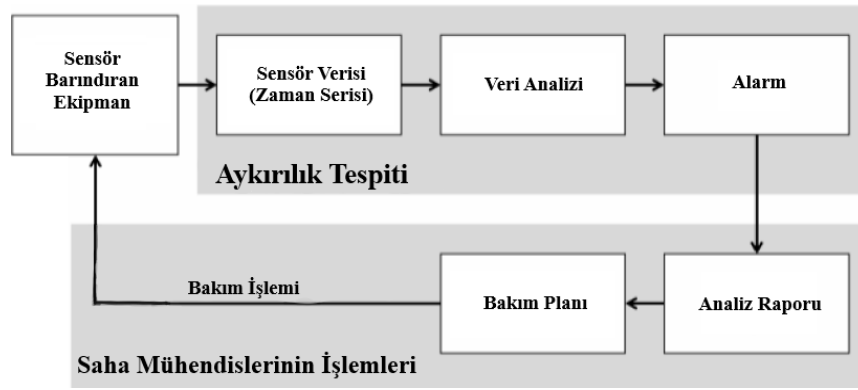
d) Pierce'in Kriteri:

Çok büyük bir hatayı içerdikleri için tüm gözlemlerin ötesine atılacağı bir dizi gözlem için bir hata limiti belirlenmektedir. [36]

Aykırlıklar bir düzen içerisinde normalden farklı olan durumlardır. Bir zaman serisi içerisindeki anormallikleri bulmak için; istatistiksel veya makine öğrenmesi yöntemleri kullanılmaktadır [37]. Zaman serisi içindeki aykırılıkların tespiti için verinin sınıflandırılmasından sonra aykırılıkların bulunması, verinin kesitleri içerisinde kendisi ile eşleşmeyen en yakın komşusuna en büyük uzaklığa sahip olan kesitin bulunması gibi birden çok yöntem bulunmaktadır.

1.6. Kestirimci Bakım Adımları

Korelasyona bağlı olarak aykırılıkların bulunması ve kestirimci bakım yapılması için Şekil 1.3'de görüldüğü gibi; öncelikle veri incelenir, düzeltilir, analiz edilir. Ardından makine öğrenmesi ya da yapay sinir ağları vasıtası ile korelasyonlar bulunur ve aykırılıklar tespit edilir. Aykırılıkların tespitinin ardından uyarılar oluşturulur ve raporlanır. Raporlamanın ardından bakım planı yapılır ve saha teknisyenleri plana göre bakımı gerçekleştirir.



Şekil 1.3 PdM Akış Şeması [38]

S. Zhong ve arkadaşları [39] anormalliklerin bulunması için Jianwei Ding ve arkadaşları [40] tarafından geliştirilen Latent Correlation Anomaly Detection (LCAD) yöntemini kullanmıştır. Bu yöntemi kullanmalarının sebebi diğer anormallik tespiti yöntemlerinin belirli zaman aralıklarındaki belirli veri setleri ile düzgün çalışabiliyor olmalarıdır. Sadece birkaç yöntem karmaşık ve çok boyuttaki zaman serilerine uygulanmaktadır.

Tablo 1.1’de görüldüğü üzere gerçek bir veri setinde yapılan testler sonucu LCAD normal anormallik tespit yöntemlerinden alarm oluşturma ve alarmları kaçırmama yönünde çok daha iyi bir performans sergilemiştir.

Tablo 1.1 LCAD ve Diğer Yöntemlerin Karşılaştırması [40]

Algoritma	A	FAR	MAR
Geliştirilmiş LCAD	70.9%	7.1%	2.2%
İstatistik Bazlı	64.8%	4.8%	5.6%
Uzaklık Bazlı	51.4%	6.5%	7.4%

P. Zhao ve arkadaşları [38] çalışmalarında S. Zhong ve arkadaşlarının metodunu [39] kullanmışlardır. S. Zhong ve arkadaşlarının yönteminde korelasyon katsayıları hesaplanırken tüm sensörler hesaba katılmasından dolayı sensör sayısı fazla olduğunda boyutsallık problemi ortaya çıkmaktadır. İkinci bir sorun ise olasılık modelinin anomalileri daha doğru algılamak için optimize edilmesi gerekmektedir. P. Zhao ve arkadaşları bu sorunların üstüne giderek yöntemlerini uygulamışlardır.



2. MATERYAL VE YÖNTEM

2.1. Çalışmada Kullanılan Donanımlar

Bu tez çalışması sırasında donanım olarak Intel Core i7-7700K işlemcili, NVIDIA GTX 1070 grafik kartlı ve 32GB belleğe sahip bilgisayar kullanılmıştır.

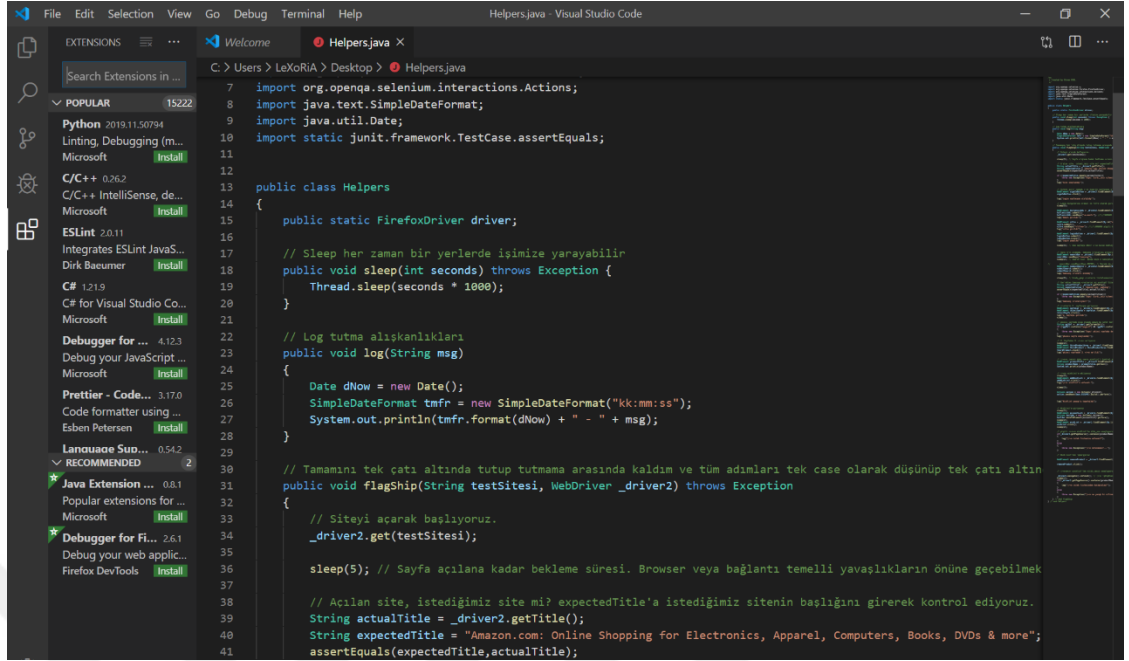
Intel i7-7700K işlemci, Intel'in 7. nesil işlemci serisinin üst sınıf ürünlerindendir. Performans için tasarlanan bu işlemci, çekirdek baz frekansı 4.2 GHz olan 4 yüksek performanslı çekirdeğe ve 8MB önbelleğe sahiptir. Intel® Turbo Boost 2.0 Teknolojisi ile maksimum turbo frekansı 4,5 GHz gibi bir hıza ulaşmaktadır. 8 yollu çoklu görev çalıştırmayı mümkün kılan Intel® Hyper-Threading Teknolojisi'ni destekleyen işlemcinin çarpan kilitsiz özelliği sayesinde hız aşırıya olanak sağlamaktadır [41].

NVIDIA GTX 1070, NVIDIA'nın oyun GPU mimarisi olan NVIDIA Pascal™'ı destekleyen 10 serisi grafik kartıdır. 1920 çekirdek ile 8GB GDDR5 çerçeve arabelleği sunan grafik kartı, 8 Gbps bellek hızına ve 256-bit bellek arayüzü genişliğine sahiptir [42].

2.2. Çalışmada Kullanılan Yazılımlar

Tez çalışmasında kod yazım ortamı olarak Visual Studio Code; programlama dili olarak TensorFlow 2 ve Keras kütüphaneleriyle birlikte Python 3.7 kullanılmıştır.

Visual Studio Code Microsoft tarafından Windows, Linux ve MacOS için geliştirilen bir kaynak kodu düzenleyicisidir. Hata ayıklama, gömülü Git kontrolü, sözdizimi vurgulama, akıllı kod tamamlama, snippet'ler ve kod yeniden yapılandırma desteği içermektedir [43]. Şekil 2.1'te Visual Studio Code arayüzü görülmektedir. Arayüz, bir kod yazım ortamı olarak gayet basit ve sade olmakla birlikte, kişiselleştirilebilir renk ve tema ayarlarına sahiptir.



Şekil 2.1 Visual Studio Code Arayüzü Örneği

Python, nesne yönelimli, yorumlamalı, birimsel ve etkileşimli yüksek seviyeli, girintilere dayalı basit sözdizimine sahip bir programlama dilidir [44].

TensorFlow, bir dizi dataflow ve farklılaştırılabilir programlama için ücretsiz ve açık kaynaklı bir yazılım kütüphanesidir. Sembolik bir matematik kütüphanesidir ve ayrıca sinir ağları gibi makine öğrenme uygulamaları için de kullanılmaktadır [45].

Keras, Python'da yazılmış açık kaynaklı bir sinir ağı kütüphanesidir. TensorFlow, Microsoft Cognitive Toolkit, R, Theano veya PlaidML'in üzerinde çalışabilmektedir. Derin sinir ağları ile hızlı deneyler sağlamak üzere tasarlanan, modüler ve genişletilebilir olmaya odaklanmaktadır [46].

2.3. Çalışmada Kullanılan Teknikler ve Veri Seti

Bu tez çalışmasında veri madenciliği teknikleri kullanılarak verilerin seçimi ve elemesi yapılmıştır. Bu aşamada; zaman serisi şeklindeki sensor verileri sisteme alınmıştır. Bunun yanında veri korelasyonu kısmında ayrı bilgi parçaları arasındaki istatistiksel korelasyonu bulmak için olay işleme ve örüntü tanıma teknikleri kullanılmıştır [25]. Korelasyonların bulunmasının ardından, zaman serisi şeklindeki veri üzerinde korelasyonlar göz önünde bulundurularak aykırılıklar tespit edilmiş ve elde edilen

sonuçlara göre sistem üzerindeki, revizyonlar önceden tahmin ile gerçekleştirilmiştir. Korelasyonlar bulunurken aynı zamanda sensörler, kalan ömrün oranına bakmaktadır.

Çalışmada birden çok sensör verisi ve kalan ömür bilgisini içeren zaman bağımlı bir veri seti olması itibarıyla ve PdM için gerekli olan *kalan ömür bilgisini* içeren bir veri seti olması sebebiyle NASA'nın "Turbofan Engine Degradation Simulation Data Set" [47] veri seti kullanılarak yapay sinir ağı oluşturulmuştur. Bu veri setinin seçilmesinde şu ana sebepler yer almaktadır:

- Sistem içindeki birimlerin verileri doğrudan parçadan gelmektedir.
- Birden çok sensör kullanılmıştır.
- Cihazın ne kadar çalıştığı bilgisini içermektedir.

Ayrıca bu veri seti üzerinde çalışmalar yapılmış olması ve ilgili alanda başka sonuçlar olması nedeniyle ilgili veri setinin bu çalışma için kullanılması uygun görülmüştür.

Şekil 2.2'de örneği gösterilen bu veri seti; farklı konfigürasyonlar ile oluşturulmuş dört veri seti içermektedir. Tablo 2.1'de gösterildiği gibi; her bir veri seti birden çok turbofan motorunun üç operasyon, yirmi bir sensör verisi ve cycle değeri ile ters ilişkili olan RUL etiketi bulunduran 25 boyutlu zaman bağımlı verisini içermektedir.

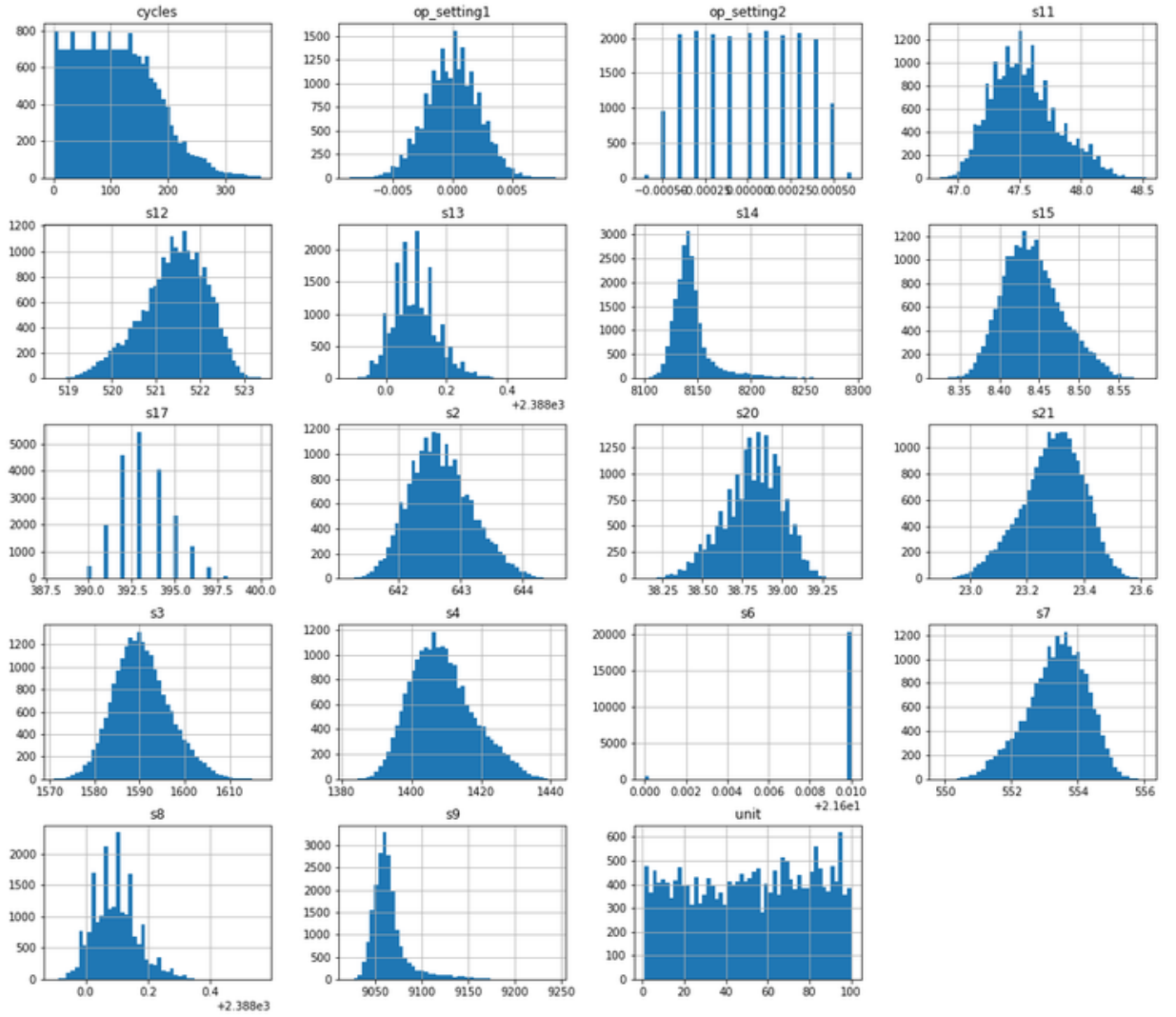
dataset_id	unit_id	cycle	setting 1	setting 2	setting 3	sensor 1	sensor 2	sensor 3	sensor 4
FD001	1	1	0.0023	0.0003	100.0	518.67	643.02	1585.29	1398.21
FD001	2	4	0.0025	0.0001	100.0	518.67	642.31	1583.43	1408.23
FD001	2	5	0.0004	-0.0004	100.0	518.67	642.77	1585.03	1407.6
FD001	13	14	0.0003	0.0005	100.0	518.67	642.22	1590.06	1402.01
FD001	13	15	-0.0004	-0.0001	100.0	518.67	642.12	1579.08	1403.83
FD001	46	131	0.0012	-0.0002	100.0	518.67	642.91	1595.1	1411.27
FD003	75	44	4	0.0003	100.0	518.67	642.17	1577.98	1391.86
FD004	14	87	249.994	0.6205	60.0	462.54	537.04	1253.68	1054.84
FD004	14	88	42003	0.8418	100.0	445.0	549.71	1352.98	1129.37
FD004	14	89	100.069	0.25	100.0	489.05	604.84	1501.09	1314.3

Şekil 2.2 Turbofan Engine Degradation Simulation Veri Seti Örnek Verileri

Tablo 2.1 Turbofan Motoru Veri Seti Nitelikleri

Operasyonel Ayarlar		Açıklama
Setting 1		Rakım
Setting 2		Mach Değeri
Setting 3		Gaz Kelebeği Açısı
Sensör Değerleri		
Sensör No		Açıklama
1		Fan Girişindeki Sıcaklık
2		LPC Çıkışındaki Sıcaklık
3		HPC Çıkışındaki Sıcaklık
4		LPT Çıkışındaki Sıcaklık
5		Fan Girişindeki Basınç (psia)
6		Baypas Kanalındaki Basınç (psia)
7		HPC Çıkışındaki Basınç (psia)
8		Fiziksel Fan Hızı (rpm)
9		Fiziksel Çekirdek Hızı (rpm)
10		Motor Basınç Oranı
11		HPC Çıkışındaki Statik Basınç (psia)
12		HPC Çıkışına Yakıt Akış Oranı (pps/psia)
13		Düzeltilmiş Fan Hızı (rpm)
14		Düzeltilmiş Çekirdek Hızı (rpm)
15		Baypas Oranı
16		Yakıcı Yakıt-Hava Oranı
17		Hava Alma Buharı Sıcaklığı
18		Beklenen Fan Hızı (rpm)
19		Düzeltilmiş Beklenen Fan Hızı (rpm)
20		HPT Soğutucu Buhar Çıkışı (lbm/s)
21		LPT Soğutucu Buhar Çıkışı (lbm/s)
Eklenmiş Etiketler		
1		RUL

Şekil 2.3’de veri seti içerisindeki örnek sensörlerin değer dağılımları histogram şeklinde gösterilmektedir.



Şekil 2.3 Veri Seti Örnek Sensör Verisi Histogramları

Sensörlerden anlık olarak zaman serisi şeklinde gelen sensör verileri ile sık kullanılan regresyon metotları yerine cihazlar arasındaki korelasyon ve aykırı durumların tespiti ile veriyi zenginleştirerek, kalan ömür tespitinin yapılmasını amaçlayan bir yapay sinir ağı oluşturmak ve bu ağı karşılaştırmak için temel (LSTM-CNN-Dense) yapay sinir ağları kullanılarak sonuçlar gözlemlenmiş ve bunların performans açısından karşılaştırılması gösterilmeye çalışılmıştır. Bu sayede verinin işlenmesi ile elde edilen işlenmiş veri ile zenginleştirilen modelin daha başarılı sonuçlar aldığı gözlemlenmiş ve daha iyi RUL tahminlemeleri yaparak üretim maliyetlerinin düşmesine ve üretim aksamamasına katkıda bulunulmuş olunacaktır.

2.4. Yapay Sinir Ağı Mimarileri

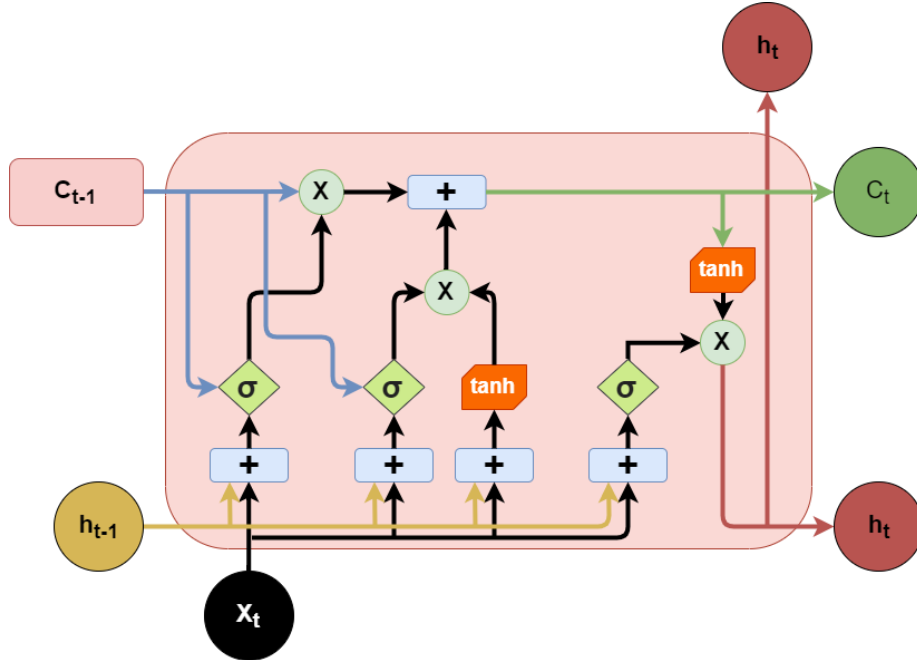
2.4.1. Uzun-Kısa Süreli Hafıza Mimarisi

Uzun-Kısa Süreli Hafıza (Long-Short Term Memory- LSTM), derin öğrenme alanında kullanılan yapay tekrarlayan sinir ağı mimarisidir. Standart ileri beslemeli sinir ağlarının aksine, LSTM'nin geri besleme bağlantıları bulunmaktadır ve yalnızca tek veri noktalarını değil, tüm veri dizilerini de işleyebilmektedir.

LSTM birimleri veya blokları, tekrarlayan bir sinir ağı yapısının bir parçasıdır. Bu yapay zekâ programlarının insan düşüncesini daha etkin bir şekilde taklit etmesine yardımcı olabilecek belirli yapay bellek işlemlerinden faydalanmak için tekrarlayan sinir ağları yapılmaktadır.

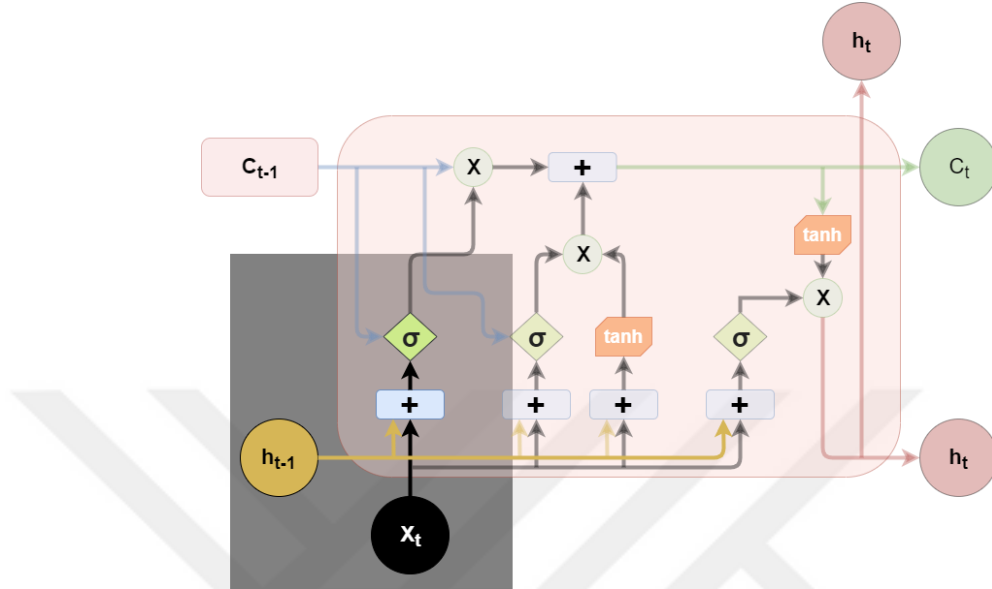
Tekrarlayan sinir ağı; bir bağlam çerçevesinde, programın girdileri alma ve çıktıları oluşturması için uzun kısa süreli hafıza blokları kullanmaktadır. Uzun kısa süreli hafıza bloğu, ağırlıklı girdiler, aktivasyon fonksiyonları, önceki bloklardan gelen girdiler ve nihai çıktılar gibi çeşitli bileşenlere sahip karmaşık bir ünitedir.

Şekil 2.4'de gösterildiği üzere X_t şu anki bloğun girdileri, h_t şu anki bloğun çıktıları ve C_t ise şu anki bloğun hafızası olarak ifade edilmektedir.



Şekil 2.4 LSTM Düğümü Yapısı

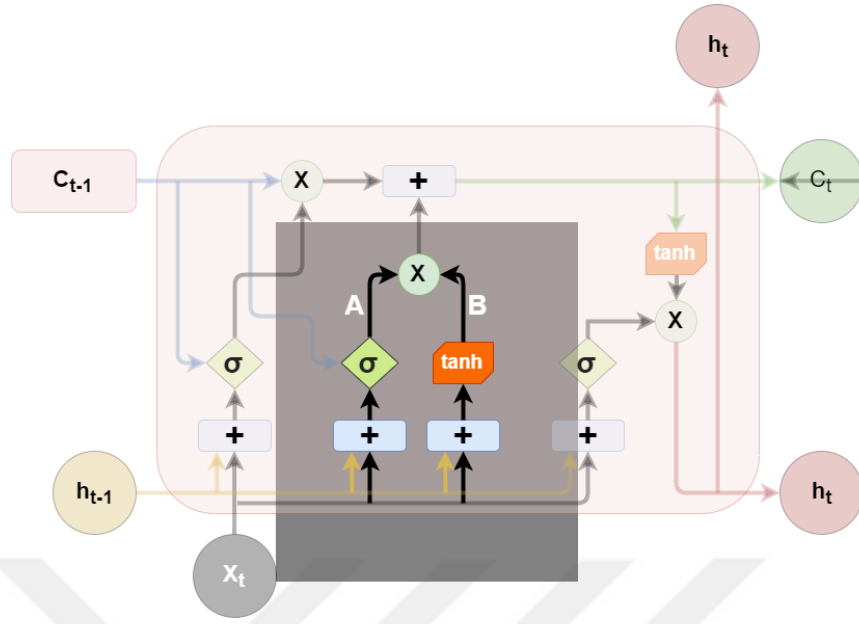
İlk olarak “Unutma Kapısı” denilen işlem ile bilgiyi saklayıp saklamayacağına karar verir. Denklem 2.1’de gösterildiği üzere önceki düğümün çıktısı h_{t-1} ile girdi X_t toplanır ve Sigmoid Fonksiyonu uygulanarak hafızaya iletilir.



Şekil 2.5 LSTM Unutma Kapısı

$$f(t) = \sigma(W_f \cdot [h_{t-1}, X_t] + b_f) \quad (2.1)$$

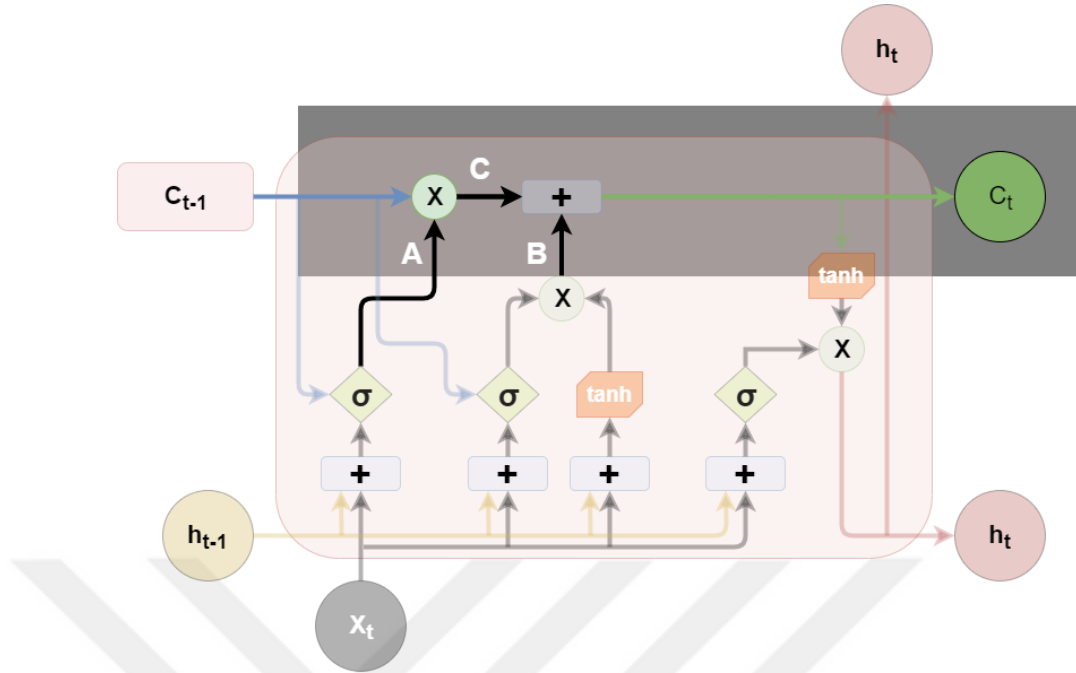
Şekil 2.6’da önceki düğümün sonucuna ve şu anki girdiye göre düğüm içerisinde hangi bilginin tutulması gerektiğini belirlediği “Girdi Kapısı” gösterilmektedir. Bunun için Denklem 2.2’de gösterildiği gibi sigmoid ve tanh aktivasyon fonksiyonlarından geçen değerler matris çarpımı ile çarpılır ve hafızaya gönderilir.



Şekil 2.6 LSTM Girdi Kapısı

$$\begin{aligned} A &= \sigma(W[h_{t-1}, x_t] + b) \\ B &= \tanh(W[h_{t-1}, x_t] + b) \end{aligned} \quad (2.2)$$

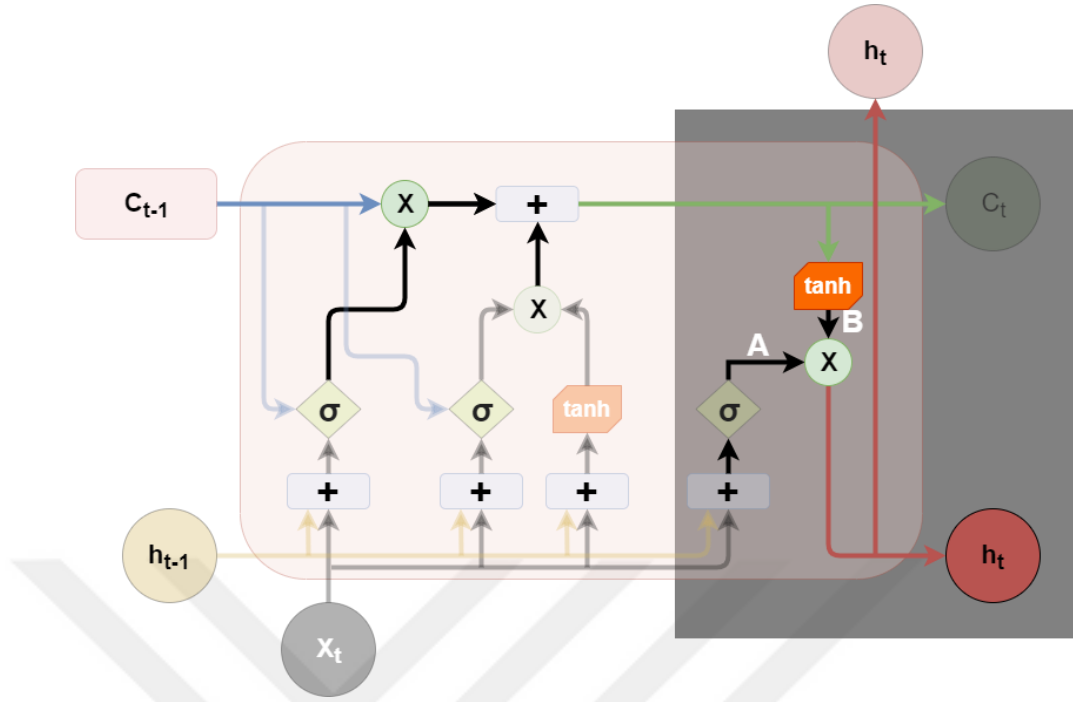
Unutma ve Girdi kapılarında hesaplanan değerler artık hafızaya eklenmeye hazırdır. Bunun için Şekil 2.7'deki gibi önceki düğümün hafızası ve Denklem 2.1'de LSTM Unutma Kapısı'ndan elde edilen değerlerin ürünü ile Denklem 2.3'den elde edilen değerlerin ürünü toplanarak düğümün hafızasını oluşturur.



Şekil 2.7 LSTM Düğüm Hafızası

$$\begin{aligned}
 C &= C_{t-1} * A \\
 B &= (\sigma(W[h_{t-1}, X_t] + b)) * (\tanh(W[h_{t-1}, X_t] + b)) \\
 C_t &= C + B
 \end{aligned} \tag{2.3}$$

Şekil 2.8’de gösterildiği gibi düğüm hafızası A ve girdi ile tanh fonksiyonu uygulanan önceki düğümün çıktısı C_t ’nin işleme sokulması ile düğümün çıktısı h_t elde edilir. Denklem 2.4’te çıktının hesaplanması gösterilmektedir.



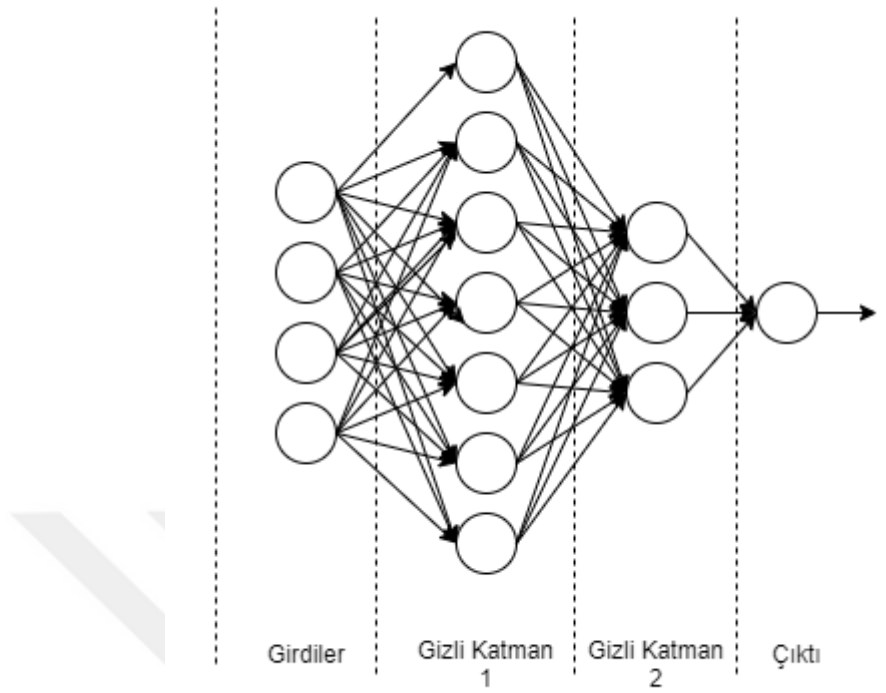
Şekil 2.8 LSTM Çıktısı

$$\begin{aligned}
 A &= \sigma(W_f \cdot [h_{t-1}, X_t] + b_f) \\
 B &= \tanh(C_t) \\
 h_t &= A * B
 \end{aligned}
 \tag{2.4}$$

Üniteye uzun süreli bir hafıza bloğu denmektedir. Bunun nedeni; programın daha uzun süreli hafıza oluşturmak için kısa süreli hafıza işlemlerine dayanan bir yapı kullanmasıdır. Bu sistemler, örneğin doğal dil işlemede sıklıkla kullanılmaktadır. Tekrarlayan sinir ağı, belirli bir kelimeyi ya da fonemi almak için uzun kısa süreli hafıza bloklarını kullanır ve hafızanın bu tip girdileri ayırmada ve kategorize etmede yararlı olabileceği bir dizgede başkalarının bağlamında değerlendirir [48]. Genel olarak, LSTM, tekrarlayan sinir ağlarında öncü olarak kabul edilmiş ve yaygın bir kavramdır [20].

2.4.2. Dense Ağlar

Dense (yoğun) ağlara bu ismin verilmesinin sebebi, katmanların bir ağ katmanındaki nöronlar tarafından tam olarak bağlandığını (yoğun olduğunu) gösteriyor olmasıdır. Bir katmandaki her nöron, önceki katmandaki tüm nöronlardan bir girdi almaktadır ve böylece yoğun şekilde bağlanmış olurlar. Şekil 2.9’da Dense ağ mimarisi görülmektedir.



Şekil 2.9 Dense Ağ Mimarisi

Başka bir deyişle, yoğun katman tamamen bağlı bir katmandır, yani bir katmandaki tüm nöronlar bir sonraki katmandakilere bağlanmaktadır.

Yoğun katmanlar, ilginç bir doğrusal olmayan özellik ekler, böylece herhangi bir matematiksel işlevi modelleyebilmektedirler. Bununla birlikte, aynı giriş vektörü için her zaman aynı çıkış vektörünü elde edilen anlamda hala sınırlıdır. Zamanında tekrarlamayı tespit edemezler veya aynı girişte farklı cevaplar üretmezler [19].

2.4.3. Evrişimli Sinir Ağı

Derin öğrenmede, evrişimli (convolutional – CNN veya ConvNet) bir sinir ağı, görsel görüntüleri analiz etmek için en sık uygulanan derin sinir ağı sınıflarındandır. Ortak ağırlık mimarisi ve çeviri değişmezliği özelliklerine bağlı olarak uzay değişmez yapay sinir ağları olarak da bilinir. Görüntü ve video tanıma, öneri sistemleri, görüntü sınıflandırma, tıbbi görüntü analizi ve doğal dil işleme alanlarında uygulamaları bulunmaktadır [49].

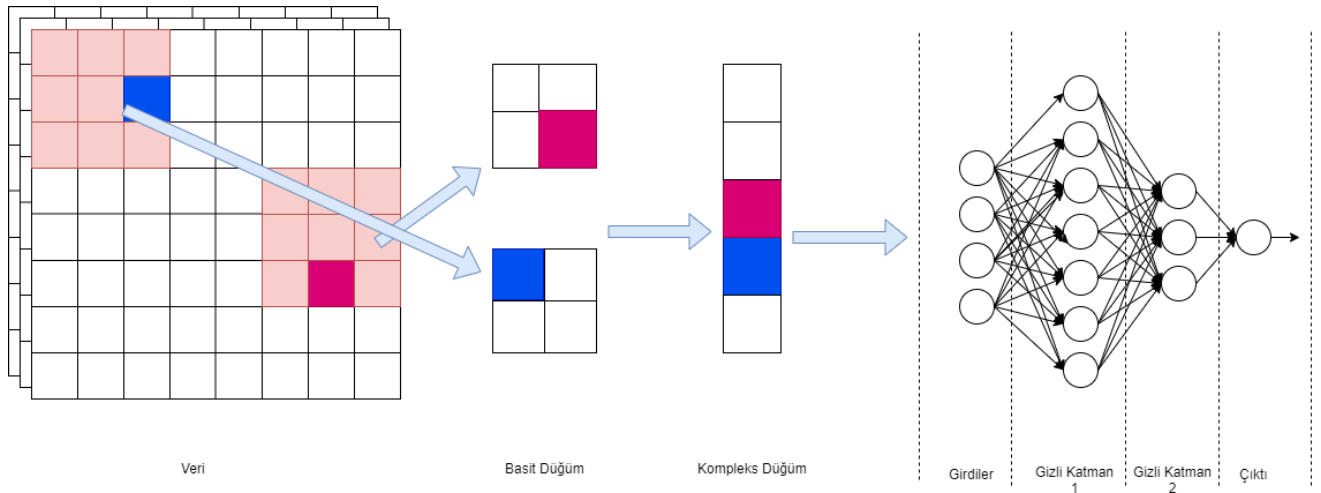
Bir CNN, özellikle piksel verilerini işlemek için tasarlanmış, görüntü tanıma ve

işlemede kullanılan bir tür yapay sinir ağıdır.

CNN'ler, hem üretici hem de tanımlayıcı görevleri yerine getirmek için derin öğrenmeyi kullanan, genellikle görüntü ve video tanıma içeren makine vizyonunu, öneri sistemleri ve doğal dil işlemeyi (Natural Language Processing – NLP) içeren, güçlü bir görüntü işleme yapay zekâsıdır.

Bir sinir ağı, insan beynindeki nöronların çalışmasından sonra düzenlenmiş bir donanım ve/veya yazılım sistemidir. Geleneksel sinir ağları görüntü işleme için ideal değildir ve görüntüleri düşük çözünürlüklü parçalarla beslemelidir. CNN, insanlarda ve diğer hayvanlarda görsel uyaranların işlenmesinden sorumlu alan olan ön lobdaki gibi düzenlenmiş “nöronlarına” sahiptir. Nöron tabakaları, geleneksel sinir ağlarının parça parça görüntü işleme probleminden kaçınarak tüm görsel alanı kaplayacak şekilde düzenlenmektedir.

CNN, azaltılmış işlem gereksinimleri için tasarlanmış çok katmanlı bir algılayıcıya benzer bir sistemi kullanmaktadır. Şekil 2.10'da CNN ağ mimarisi gösterilmektedir. Bir CNN'nin katmanları, bir giriş katmanı, bir çıkış katmanı ve çoklu katlamalı katmanlar, havuz katları, tamamen bağlı katmanlar ve normalizasyon katmanları içeren gizli bir katmandan oluşmaktadır. Sınırlamaların kaldırılması ve görüntü işleme için verimliliğin artırılması, görüntü işleme ve doğal dil işleme için sınırlı trenler için daha basit, çok daha etkili bir sistemle sonuçlanmaktadır [19].



Şekil 2.10 CNN Mimarisi

2.4.4. Hibrit (Karma) Sinir Ağı

CNN, LSTM ve Dense katmanları kullanılarak oluşturulan karma yapay sinir ağı; korelasyonların bulunması için CNN katmanı, aykırılıkların tespiti için ise LSTM katmanı kullanılmıştır. Oluşturulan bu karma yapay sinir ağının kalan ömür tahminlemesi, diğer yapay sinir ağı modelleri ile karşılaştırılmıştır.

Korelasyonun bulunması kısmında CNN katmanlardan yararlanılmıştır. *Convolution* yapılarak veriler arasındaki konvolüsyon hesaplanmış ve buna göre işlem yapılmıştır.

PdM yapabilmek amacıyla geleneksel ve mevcut çalışmalarda kullanılan regresyon metotları yerine yapay sinir ağları kullanılmıştır.

Çalışmanın konusu dâhilinde sensörlerin korelasyonu ve anomalilerin, PdM için kalan zaman hesaplamasına etkisi olup olmadığı gözlemlenmeye çalışılmıştır. Çalışmada karşılaştırma yapabilmek adına 4 temel yapay sinir ağı oluşturularak kalan ömür tahminlemesi yapılmaya çalışılmıştır. Bu 4 yapay sinir ağının 3 tanesi olan dense, convolutional ve recurrent neural networkler karşılaştırma için hazırlanmıştır. 4. yapay sinir ağı ise hibrit bir yapay sinir ağıdır ve çalışmanın amacına uygun olarak convolutional nodelar ile korelasyon bilgisi, LSTM nodelar ile anomali tahminleri yapay sinir ağına eklenmiş ve bu ilave özelliklerin kalan ömür tahminlemesine olan etkisi gözlemlenmiştir.



3. BULGULAR VE TARTIŞMA

Oluşturulan yapay sinir ağlarında etkili bir karşılaştırma yapılabilmesi adına; aşırı uymadan (*overfitting*) kaçınılarak, olabildiğince hiperparametreler (*hyperparameter*) ile oynanarak en iyi sonuçları almaya odaklanılmıştır. Yapay sinir ağlarının ne kadar çok eğitilirse o kadar çok başarılı sonuçlar vermesinden dolayı; bu karşılaştırmalar arasında eşitsizlik olmaması adına eğitim (*training*) sürelerinin mümkün olduğunca yakın olması ve çok uzun sürmemesine özen gösterilmiştir. Sonuç olarak kullanılan hibrit ağ, %95 olarak en yüksek başarıyı ve en düşük kaybı (*loss*) yakalamıştır.

Literatürde aynı veri seti ile RUL tahminlemesi yapan çalışmalar mevcuttur. Mathew ve arkadaşlarının yaptığı çalışmada Decision Tree, SVM, Random Forest, KNN, K-Means, Gradient Boosting, Adaboost, Anova gibi makine öğrenmesi teknikleri kullanılarak Şekil 3.1'deki gibi sonuç elde edilmeye çalışılmış ve karşılaştırmıştır.

<i>Algorithm</i>	<i>Data Set 1</i>	<i>Data Set 2</i>	<i>Data Set 3</i>	<i>Data Set 4</i>	<i>Mean</i>
Linear Regression	29.91	31.49	45.64	39.81	36.71
Decision Tree	28.48	34.52	27.74	45.91	34.17
SVM	48.17	31.12	61.53	34.65	43.86
Random Forest	24.95	29.64	30.55	33.79	29.73
KNN	30.79	34.79	34.44	44.70	36.18
K Means	78.30	90.19	72.92	95.45	84.21
Gradient Boost	27.45	33.35	31.78	39.30	32.97
Ada boost	28.82	33.84	30.91	39.16	33.18
Deep Learning	29.62	42.41	46.82	38.11	39.24
Anova	33.50	41.14	35.46	51.44	40.38

Şekil 3.1 Çeşitli Makine Öğrenmesi Yöntemleri İle Elde Edilen Sonuçlar [50]

Narendhar Gugulothu ve arkadaşlarının yaptığı çalışmada ise tekrarlı sinir ağları kullanılarak Şekil 3.2'de gösterildiği gibi RUL tahminlemesi yapılmıştır.

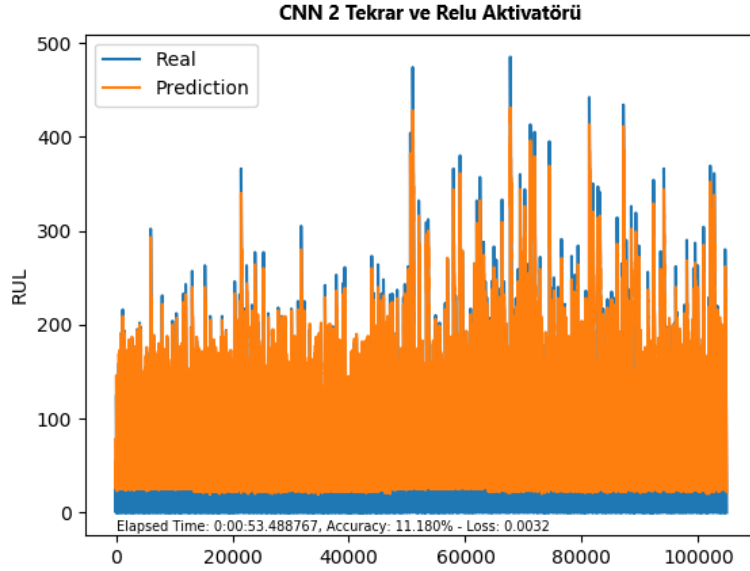
Metric	Recon-RUL	Embed-RUL (proposed)	Recon-LR ₁	Embed-LR ₁ (proposed)	Recon-LR ₂	Embed-LR ₂ (proposed)	RNN-Reg.
S	1263	810	477	219	256	232	352
MSE	546	456	288	155	164	167	219
A(%)	36	48	65	59	67	62	64
MAE	18	17	12	10	10	10	11
³ MAPE	39	39	20	19	18	19	17
FPR(%)	34	23	19	14	13	15	22
FNR(%)	30	29	16	27	20	23	24

Şekil 3.2 Yinelemeli Sinir Ağları ile Tahminlenen RUL ve Kayıp Oranları [51]

3.1. Tahmin Durumları ve Yapı Modelleri

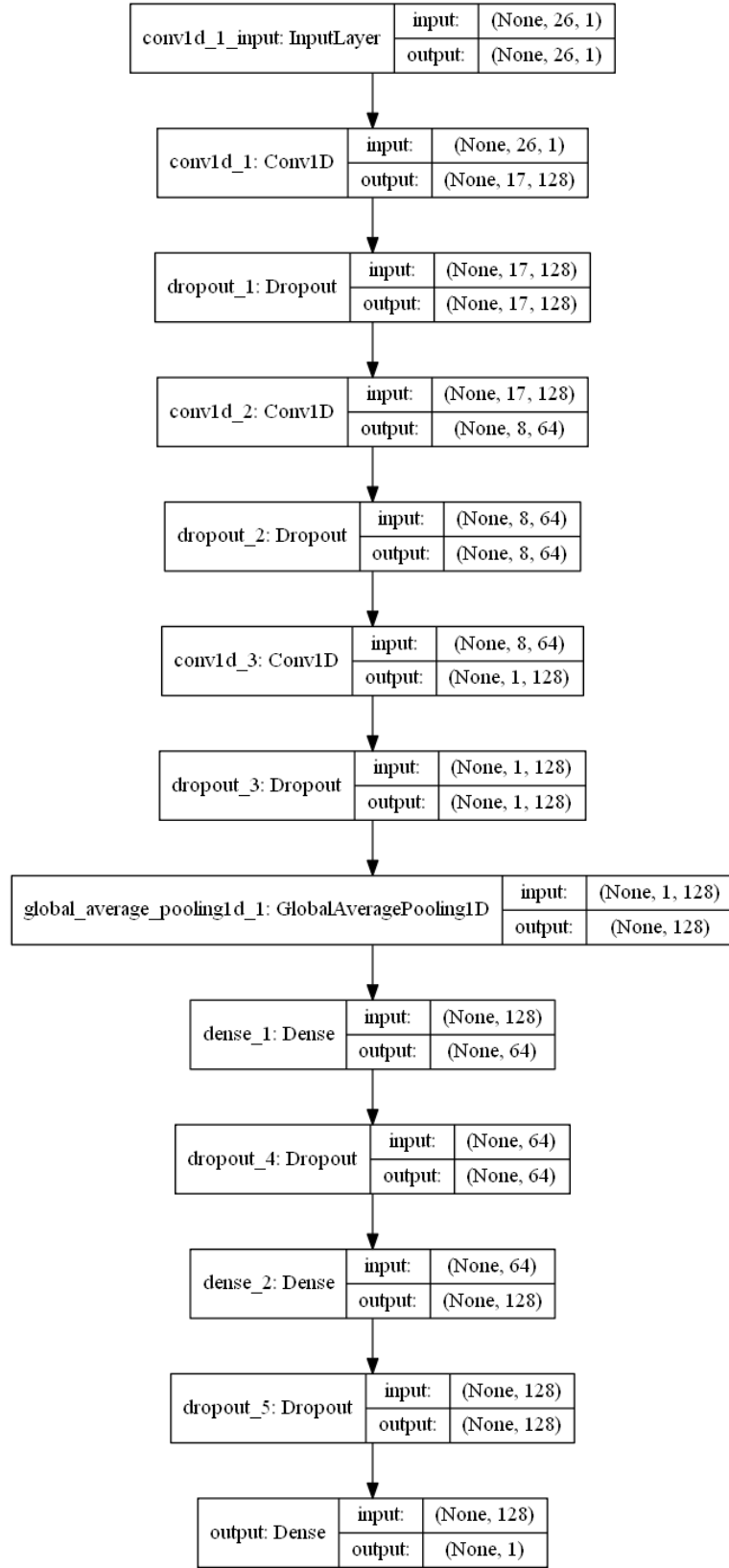
3.1.1. CNN Modeli

CNN yapının iki tekrarda (epoch) RELU aktivasyon fonksiyonu kullanılarak ortaya çıkan tahmin sonucu şekil Şekil 3.3’de gösterilmiştir. Burada mavi ile gösterilen kısım datanın kendisi RUL değerini, turuncu kısım ise tahmin edilen değeri göstermektedir. Görüldüğü üzere tahminlemenin kapsayamadığı alanlar, özellikle kalan ömrün azaldığı zamanları tahminlemekte zorlanmıştır. Burada 100.000’den fazla sensör değeri olduğu görülmektedir.



Şekil 3.3 CNN İki Tekrar ve Relu Aktivatörlü Tahminleme Sonucu

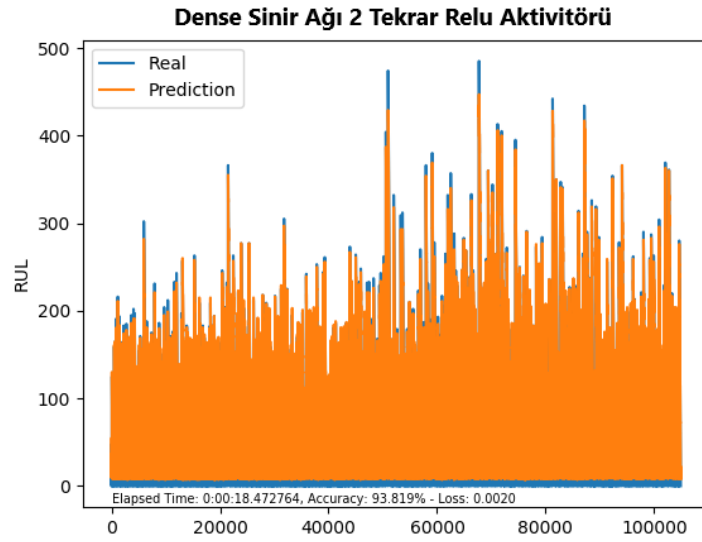
Şekil 3.4’de CNN yapı modeli görülmektedir. Conv1d katmanı konvolüsyon yapan katmandır. İlk iki conv1d katmanında filtre (1,10)’luk bir diziyken üçüncü convd’de (1,8)’lik bir filtre uygulanmaktadır. Dropout kısmı her bir katmandaki ağırlıkların her bir seferinde %20’sini sıfırlamaktadır. Böylece aşırı uyma (overfitting) durumundan kaçınılmaktadır. Burada GlobalAveragePooling1D; boyut (dimension) küçültmek amacıyla kullanılmıştır. Birden çok dizinin ortalamasını almaktadır. Bu aşamadan sonra dense katmanına geçiş yapılmaktadır.



Şekil 3.4 Evrişimli Sinir Ağı

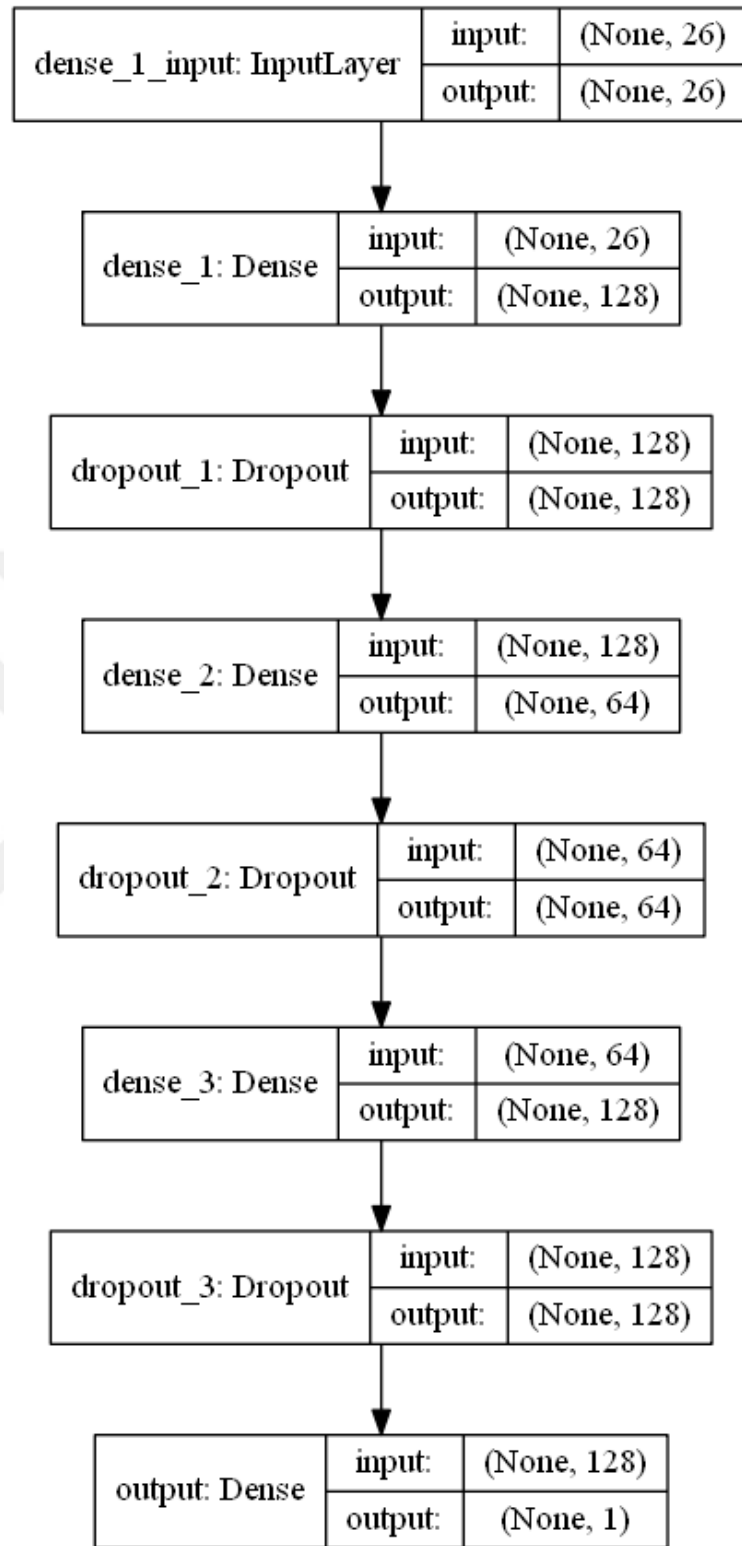
3.1.2. Dense Modeli

Şekil 3.5’de Dense yapının iki tekrar ve RELU aktivasyon kullanılarak elde edilen çıktısı gösterilmiştir. Grafik incelendiğinde, hızlı ve başarılı bir şekilde çözdüğü görülmektedir.



Şekil 3.5 Dense İki Tekrar ve Relu Aktivatörlü Tahminleme Sonucu

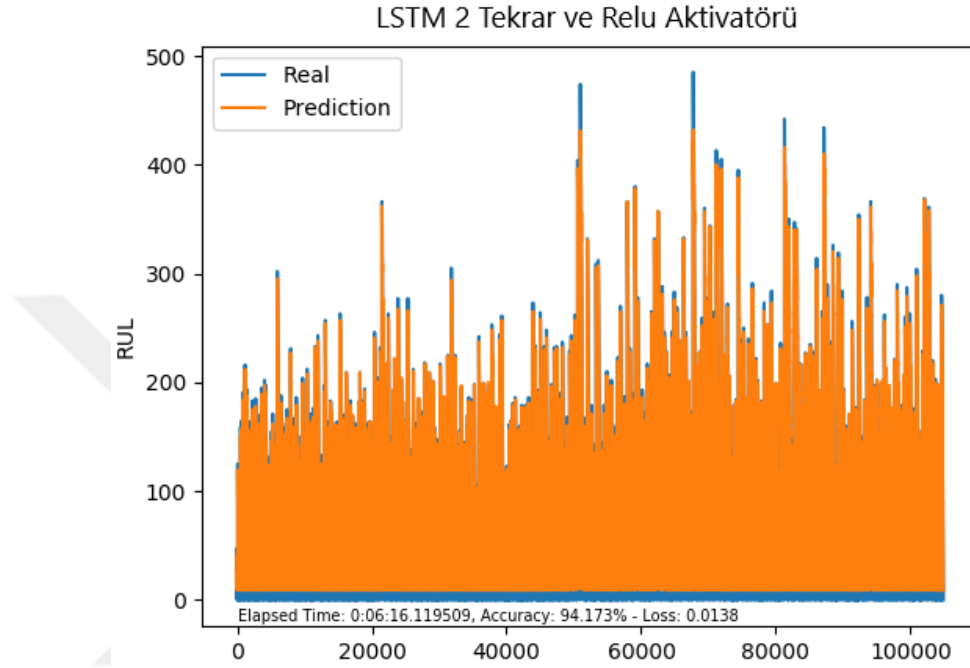
Şekil 3.6’deki Dense yapı modelinde; 26 sensör/girdi ile başlanmıştır ve 128 düğüm, 64 düğüm ve 128 düğüm olmak üzere üç katmanı bulunmaktadır.



Şekil 3.6 Dense Sinir Ağı

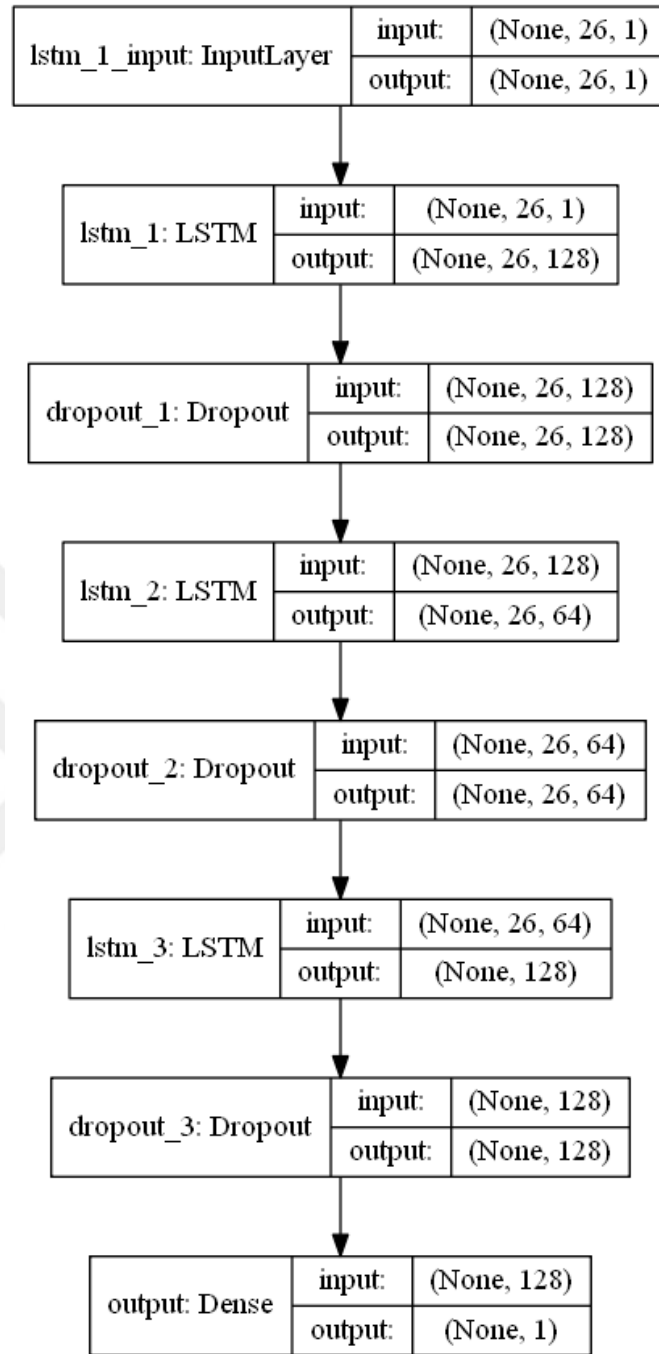
3.1.3. LSTM Modeli

Şekil 3.7’de ise LSTM ağı tahminlemesi gösterilmektedir. Tahmin oranı gayet başarılı olmuş olsa da, tahminleme süreci diğer ağlara kıyasla çok daha uzun sürmüş ve tahminleme sınırlardaki düşük ve yüksek ömürleri doğru olarak tahminleyememiştir.



Şekil 3.7 LSTM İki Tekrar ve Relu Aktivatörlü Tahminleme Sonucu

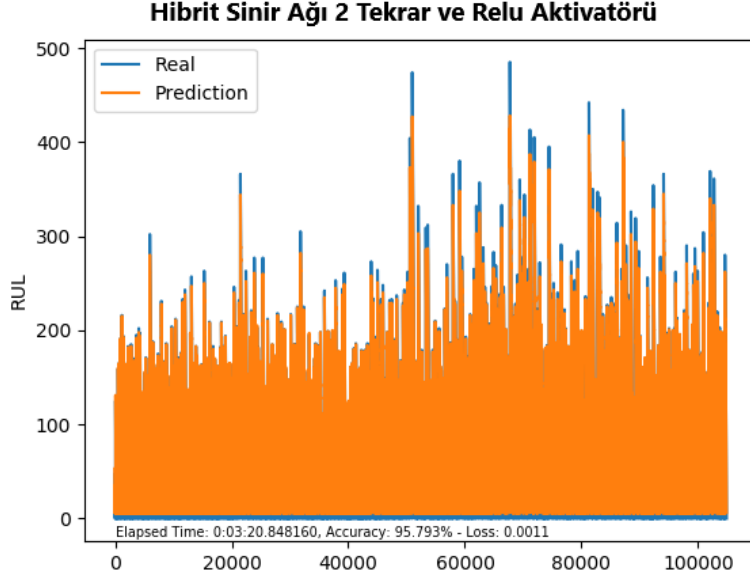
Şekil 3.8’de LSTM yapı modeli gösterilmektedir, bu model 128 düğüm, 64 düğüm ve 128 düğüm şeklinde üç katmandan oluşmaktadır. Bu yapı aynı zamanda ufak bir AutoEncoder yapısıdır. AutoEncoder (otomatik kodlayıcı), verimli veri kodlamalarını denetimsiz bir şekilde öğrenmek için kullanılan bir tür yapay sinir ağıdır [52].



Şekil 3.8 LSTM Sinir Ağı

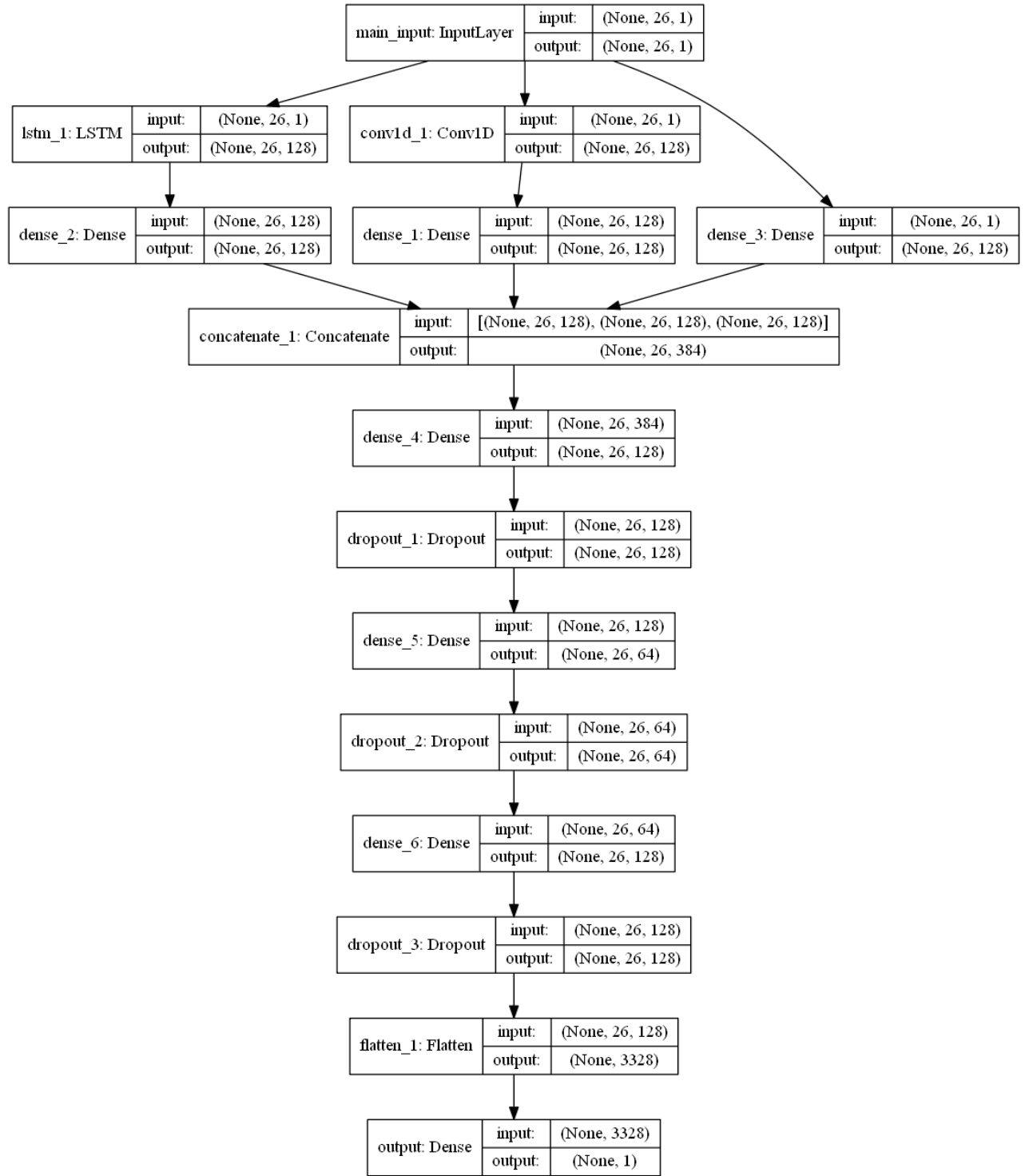
3.1.4. Hibrit Model

Hibrit yapının tahmin durumu Şekil 3.9’te gösterilmektedir. Burada zaman dense ağına göre biraz daha uzun olmasına rağmen başarı oranında artış görülmektedir. Küçük olan RUL değerlerini de, kıyasla daha çok kapsadığı gözlemlenmektedir.



Şekil 3.9 Hibrit İki Tekrar ve Relu Aktivatörlü Tahminleme Sonucu

Son olarak Şekil 3.10’de ise hibrit yapı modeli görülmektedir. Burada amaç; var olan veriden olabildiği kadar çeşitli özellik (*feature*) çıkarmaktır. LSTM yapı, gelen sensör yapısına göre tahminleme yaparak 0 ile 1 arasında olan çıktılar vermektedir ve 128 düğümden oluşmaktadır. Buradaki çıktı aslında kalan ömrün tahminlemesidir. LSTM ile bir sonraki adımda oluşacak RUL değeri tahminlendiği için 0 değerine olan yakınlık olması, hata ihtimalini arttırmaktadır. conv1d konvülyasyonları bulurken; paralel üç katman ise concatenate aşamasında birleşmektedir. Hibrit yapıdaki özellik sayısı, diğer katmanlarda 128 değerinde iken, bu katmada 391 değerine çıkmaktadır ve ardından dense yapı tahminlemeyi yapmaktadır.



Şekil 3.10 Hibrit Sinir Ağı

3.2. Başarı, Kayıp ve Hata Oranları

3.2.1. Başarı Oranları

Yapay sinir ağları içerisinde gerçekleşen normalizasyon, denormalizasyon işlemleri tahminlemelerin ondalık sayılar olarak ortaya çıkmasına sebep olmaktadır. Örneğin gerçek RUL 200 iken tahmin edilen RUL 200,3 olarak hesaplanmaktadır. Bu nedenle yüzdelik başarı oranları hesaplanırken gerçek RUL değerine $\pm 2\%$ yakınlığa bakılmıştır. Tablo 3.1’de verilen başarı oranları incelendiğinde Hibrit ağın daha başarılı sonuç verdiği görülmektedir.

Tablo 3.1 Dense, CNN, LSTM ve Hibrit Ağların Başarı Oranı Karşılaştırması

Algoritma	Başarı Oranı
Dense	93,819%
CNN	11,180%
LSTM	94,173%
Hibrit	95,793%

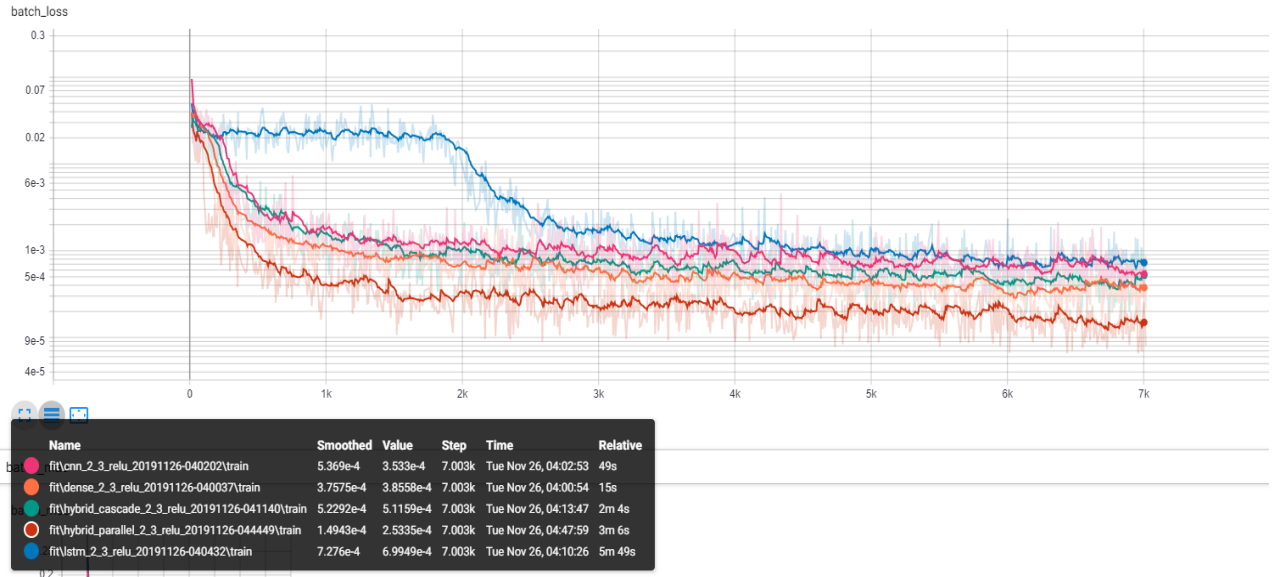
3.2.2. Kayıp (Loss) Oranları

Şekil 3.11 Kayıp (Loss) Oranlarının Karşılaştırması’te farklı yapıların kayıp (loss) oranlarının karşılaştırması verilmektedir. Bu çalışmada kayıp oranı ne kadar düşük olursa, sistem o kadar yakın bir tahminleme yapmış anlamına gelmektedir. Tablo 3.2’de verilen kayıp oranları incelendiğinde Hibrit ağın daha başarılı sonuç verdiği görülmektedir.

Denklem 3.1’de ortalama kare hatası (MSE – Mean Squared Error) görülmektedir [28].

$$MSE = \frac{1}{n} \sum_{i=1}^n (Y_i - \hat{Y}_i)^2 \quad (3.1)$$

Denklem 3.1’deki ifade, Y_i ve $\hat{Y}_i$ ile gösterilen ardışık iki değer arasındaki farkın karekökünü göstermektedir.



Şekil 3.11 Kayıp (Loss) Oranlarının Karşılaştırması

Tablo 3.2 Dense, CNN, LSTM ve Hibrit Ağların Kayıp Oranı Karşılaştırması

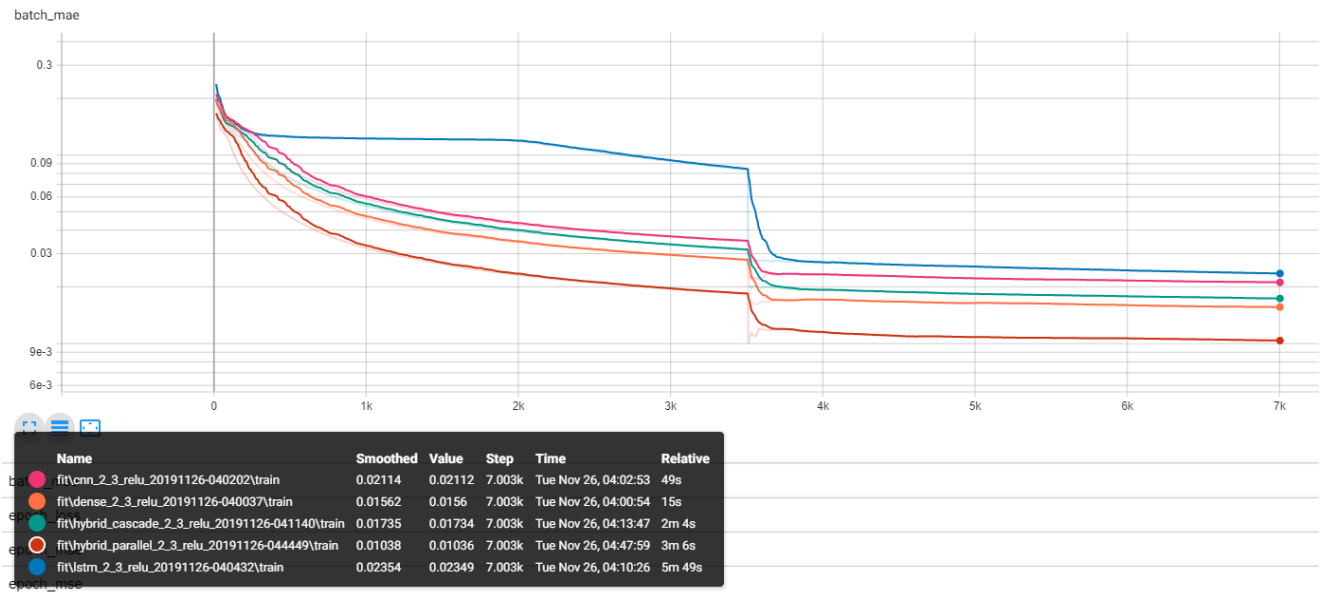
Algoritma	Kayıp Oranı
Dense	0.000385
CNN	0.000353
LSTM	0.000699
Hibrit	0.000253

3.2.3. Ortalama Mutlak Hata Oranları

Şekil 3.12’de farklı yapıların Denklem 3.2’de gösterilen ortalama mutlak hata oranlarının [28] (MAE – Mean Absolute Error) karşılaştırması verilmiştir ve formül olarak kullanılan

$$MAE = \frac{\sum_{i=1}^n |y_i - \hat{y}_i|}{n} = \frac{\sum_{i=1}^n |e_i|}{n} \quad (3.2)$$

ifadesi de, ardışık iki değerin ya da girdi ile çıktının (farkının) mutlak değerini göstermektedir. Tablo 3.3’de verilen mutlak hata oranları incelendiğinde Hibrit ağı daha başarılı sonuç verdiği görülmektedir.



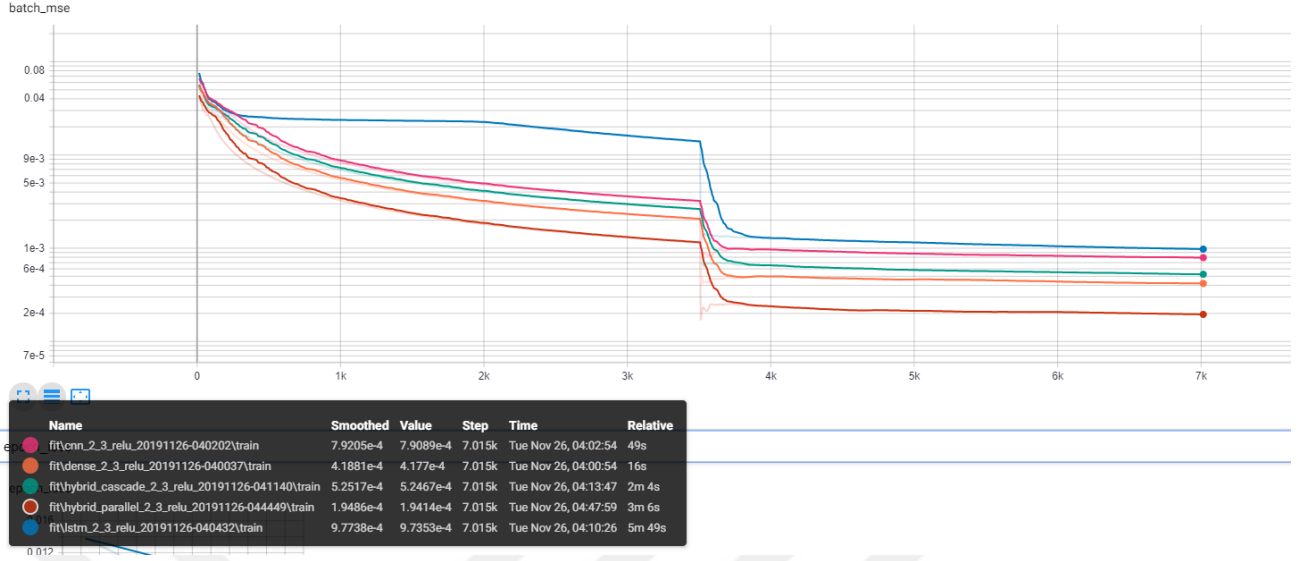
Şekil 3.12 Ortalama Mutlak Hata Oranlarının Karşılaştırması

Tablo 3.3 Dense, CNN, LSTM ve Hibrit Ağların MAE Değerlerinin Karşılaştırması

Algoritma	MAE
Dense	0.0156
CNN	0.0211
LSTM	0.0235
Hibrit	0.0103

3.2.4. Ortalama Kare Hatası Oranları

Şekil 3.13’de Denklem 3.1’deki ortalama kare hatası kullanılarak elde edilen RUL tahminleri arasındaki kayıp karşılaştırmaları gösterilmiştir. İlgili yapının value (değer) kısmında 0(sıfır)’a yakın olması, doğru tahminleme yaptığı anlamına gelmektedir. Tablo 3.4’de verilen kare hatası oranları incelendiğinde Hibrit ağın daha başarılı sonuç verdiği görülmektedir.



Şekil 3.13 Ortalama Kare Hatası Oranlarının Karşılaştırması

Tablo 3.4 Dense, CNN, LSTM ve Hibrit Ağların MSE Değerlerinin Karşılaştırması

Algoritma	MSE
Dense	0.000417
CNN	0.000790
LSTM	0.000973
Hibrit	0.000194

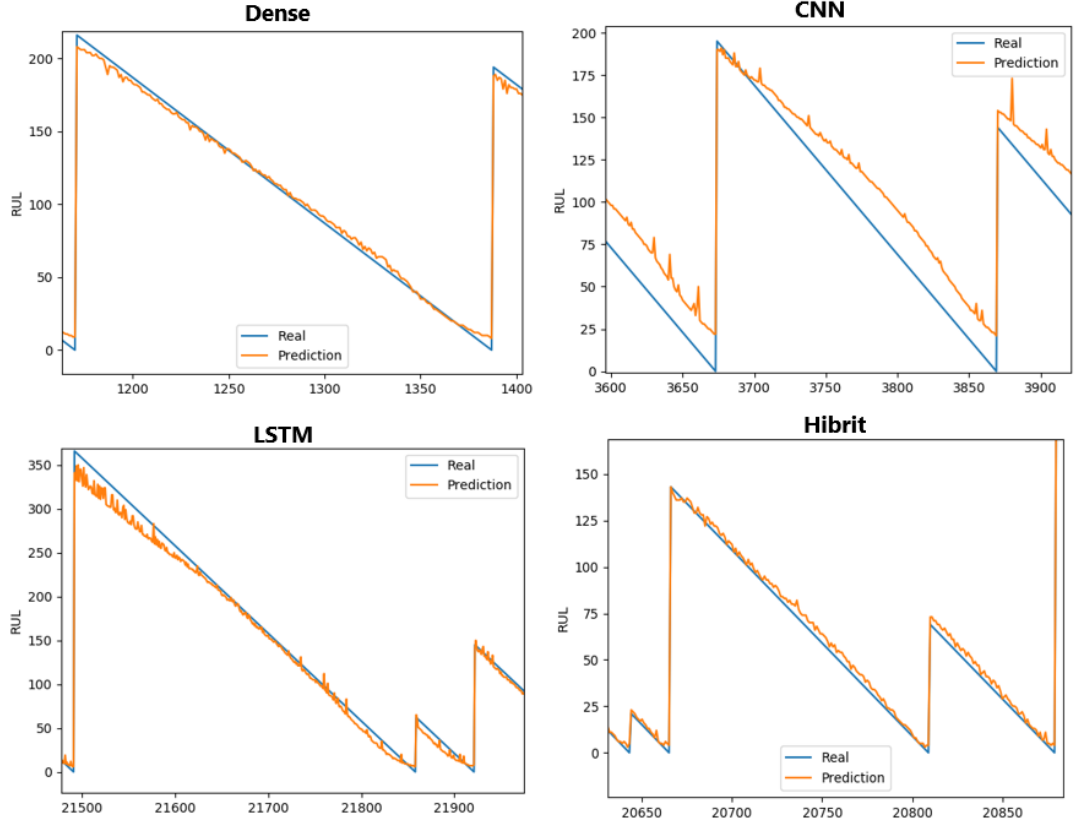
3.3. Tartışma

Çalışmada, daha önce benzer çalışmalarda kullanılan LSTM, Dense ve CNN yapıları ilgili veri seti ile denenerek, oluşturulan hibrit yapının diğer yapılarla karşılaştırılmasına imkân sağlanmıştır. Böylece bu çalışmanın diğer çalışmalardan farkı ve kullanılan hibrit yapının diğer yapılara göre ne derece avantajlı olduğu gösterilmek istenmiştir.

Mathew ve arkadaşlarının yaptığı çalışmada [50] tek tek ele alınan veri setlerine rağmen başarı oranları tez çalışmasından düşüktür. Narendhar Gugulothu ve arkadaşlarının yaptığı çalışmada [51] ise MSE oranları karşılaştırıldığında tez kapsamındaki çalışmanın daha düşük kayıp oranına sahip olduğu; dolayısı ile daha iyi RUL tahminleme yaptığı görülmektedir.

Çalışmada hibrit yapının performans olarak mevcut yapılarla kıyaslanması sonucu, gelecekteki çalışmalara ışık tutması amaçlandığından dolayı; aynı veri seti üzerinde çeşitli

mimariler denenerek sonuçlar açık bir şekilde gösterilmiştir.



Şekil 3.14 Yapay Sinir Ağları RUL Karşılaştırması

Şekil 3.14’de yapay sinir ağları ile yapılan RUL tahminlemelerinden örnek bir karşılaştırma verilmiştir. Hibrit yapının tahminlerinin diğer yapılara göre gerçek RUL ile örtüştüğü görülmektedir. CNN ve LSTM yapılar tepe ve alt noktaları kapsayamazken hibrit yapı ve dense yapı bu noktaları da tahminleyebilmiştir.

Kullanılan hibrit yapı farklı veri setleri için revize edilerek, farklı alanlarda kullanılabilir durumdadır. Yapılacak yeni çalışmalar ve o alanlardaki sonuçların paylaşılması da bu yapının performansı ve kullanılabilirliği konusunda yol gösterici nitelikte olacaktır.

Ayrıca tekrar sayısı ve kontrol verisi oranlarında değişiklikler yapılarak eğitilmesi durumunda elde edilecek yeni sonuçların da gelecekteki çalışmalara ışık tutacak nitelikte olacağı öngörülmektedir.

4. SONUÇ

Kestirimci bakım uygulamaları var olan veri içerisinde korelasyon, aykırılık gibi anlamlı öznitelikler çıkarılması ve bu çıkarılan özniteliklerin sensör değerleri ile birlikte kullanılması, tahminlemenin başarısını arttırmıştır. Çıkarılan özniteliklerin veya sensör değerlerinin farklılıkları, tahminleme sonuçlarını ve dolaylı olarak da başarı durumunu olumlu veya olumsuz etkileyecektir. RUL tahminlemesindeki artan başarı daha doğru bakım planlamaları yapılmasına imkan sağlayabilecek ve bu da maliyetlerin azalmasında etkili olacaktır.

Aynı sayıda katmanlardan oluşturulan 4 yapay sinir ağı, aynı tekrar sayısı ile ve eğitim setinin %30'u kontrol verisi olacak şekilde eğitilmiştir. 0.001 öğrenme oranı ve %20 ağırlık seyreltme oranı ile beraber, ADAM [53] optimizasyon algoritmasının daha hızlı ve başarılı olduğu görülmüştür. Performans kriteri olarak kayıp oranlarına bakılmış ve bu oran hesaplanmasında ortalama kare hatası (Mean Squared Error - MSE) kullanılmıştır. Denklem 3.1'de ki gibi formüle edilen MSE yöntemiyle; LSTM ağına kayının 0,0138, CNN ağına kayının 0,0032, Dense ağına 0,0020 ve önerilen karma ağına kayının 0,0011 olduğu gözlemlenmiştir. Tahminleme doğruluk kriteri olarak; tahmin edilen değerin, gerçek değere ± 2 yakınlığına bakılmıştır. Bunun sebebi normalizasyon ve denormalizasyon sonucu ortaya çıkan yuvarlama hatalarını gidermektir. Bu bağlamda oluşturulan karma ağı doğruluğu 95,79%, LSTM ağı doğruluğu 94,17%, dense ağı doğruluğu 93,81% ve CNN ağı doğruluğu 11,18% olarak görülmüştür.

Tablo 4.1 Dense, CNN, LSTM ve Hibrit Ağların Karşılaştırması

Algoritma	Başarı Oranı	Kayıp Oranı	MSE	MAE
Dense	93,819%	0.000385	0.000417	0.0156
CNN	11,180%	0.000353	0.000790	0.0211
LSTM	94,173%	0.000699	0.000973	0.0235
Hibrit	95,793%	0.000253	0.000194	0.0103

Performans kriteri olarak bu çalışmada kullanılan kayıp oranları yerine farklı metrikler kullanılması ise, çok daha farklı sonuçları doğuracak ve bu da yeni ve başka çalışmalar için fikir sunacak nitelikte olacaktır.

Kestirimci bakım uygularken sadece sensör değerlerinin değil, sensörlerin korelasyon ve aykırılıkların tespitinden elde edilen değerlerin de sisteme eklenmesi ile kalan ömüre etkisi olduğu tespit edilmiştir. Bu etkinin ve sonuçların başarısını arttırmak için Yu ve arkadaşlarının da bahsettiği [54] sensör seçimi yapmak, gereksiz sensörleri dahil etmemek faydalı olabilir. Shi ve arkadaşlarının geliştirdiği [55] ConvLSTM sinir ağını kullanmanın sonuçlarının araştırılması ve bu çalışma ile karşılaştırma yapılması bu alandaki çalışmalara katkı sağlayacaktır.



KAYNAKÇA

- [1] T. Klaus-Dieter, S. A. Wiesner ve T. Wuest, «"Industrie 4.0" and Smart Manufacturing – A Review of Research Issues and Application Examples,» International Journal of Automation Technology , pp. 4-19, 2017.
- [2] S. C. Lu, "Machine learning approaches to knowledge synthesis and integration tasks for advanced engineering automation," Computers in Industry, vol. 20, no. 1, pp. 15-105, 1990.
- [3] T. Wuest, D. Weimer, C. Irgens and K.-D. Thoben, "Machine learning in manufacturing: advantages, challenges, and applications," Production & Manufacturing Research, vol. 4, no. 1, pp. 23-45, 2016.
- [4] J. Lee, B. Bagheri ve H.-A. Kao, «A Cyber-Physical Systems architecture for Industry 4.0-based manufacturing systems,» Manufacturing Letters, cilt 3, no. 5, pp. 18-23, 2015.
- [5] J. Moyne ve J. Iskandar, «Big Data Analytics for Smart Manufacturing: Case Studies in Semiconductor Manufacturing,» Processes, cilt 3, no. 5, p. 39, 2017.
- [6] K. A. Kaiser ve N. Z. Gebraeel, «Predictive Maintenance Management Using Sensor-Based Degradation Models,» IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans, cilt 39, no. 4, pp. 840 - 849, 2009.
- [7] W. J. Lee, H. Wu, H. Yun, H. Kim, M. B. Jun and J. W. Sutherland, "Predictive Maintenance of Machine Tool Systems Using Artificial Intelligence Techniques Applied to Machine Condition Data," Procedia CIRP, vol. 80, no. 26, pp. 506-511, 2019.
- [8] Sensors-Works, «Condition Monitoring, Predictive Maintenance,» Sensors-Works, 04 01 2020. [Çevrimiçi]. Available: <http://sensor-works.com/>. [Erişim Tarihi: 04 01 2020].

- [9] R. K. Mobley, *An Introduction to Predictive Maintenance*, 2nd dü., Elsevier Science, 2002, pp. 01-02.
- [10] S. Cornelius ve G. Paresh, *Practical Machinery Vibration Analysis and Predictive Maintenance*, Elsevier Science, 2004.
- [11] M. A. Inc., «Common maintenance strategies,» 2015. [Çevrimiçi]. [Erişim Tarihi: 17 01 2016].
- [12] S. Kennedy, «New tools for PdM,» 2006. [Çevrimiçi]. [Erişim Tarihi: 17 01 2016].
- [13] R. Gouriveau ve E. Ramasso, «From real data to Remaining Useful Life estimation: an approach combining neuro-fuzzy predictions and evidential Markovian classifications,» 38th ESReDA Seminar Advanced Maintenance Modelling, p. 13, 2010.
- [14] C. Byington, M. Roemer and T. Galie, "Prognostic enhancements to diagnostic systems for improved condition-based maintenance," in *Proceedings, IEEE Aerospace Conference*, Big Sky, MT, 2002.
- [15] G. Vachtsevanos, F. L. Lewis, M. Roemer, A. Hess ve B. Wu, *Intelligent Fault Diagnostic and Prognosis for Engineering Systems*, New Jersey: John Wiley & Sons, 2006.
- [16] A. Heng, S. Zhang, A. C. Tan ve J. Mathew, «Rotating machinery prognostics: State of the art, challenges and opportunities,» *Mechanical Systems and Signal Processing*, cilt 23, pp. 724-739, 2009.
- [17] C. Hu, B. D. Youn, P. Wang ve J. T. Yoon, «Ensemble of Data-Driven Prognostic Algorithms for Robust Prediction of Remaining Useful Life,» *Reliability Engineering & System Safety*, cilt 103, pp. 120-135, 2012.
- [18] B. YEGNANARAYANA, *ARTIFICIAL NEURAL NETWORKS*, New Delhi: PHI Learning Pvt. Ltd, 2009.

- [19] K. Fukushima, «Neocognitron: A Self-organizing Neural Network Model for a Mechanism of Pattern Recognition Unaffected by Shift in Position,» Biological Cybernetics, pp. 193-202, 1980.
- [20] S. Hochreiter ve J. Schmidhuber, «Long Short-term Memory,» Neural Computation, p. 1735–1780, 1997.
- [21] P. Malhotra, L. Vig, G. Shroff and P. Agarwal, "Long short term memory networks for anomaly detection in time series," in European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning., Bruges, 2015.
- [22] STMicroelectronics, «Life Augmented,» [Çevrimiçi]. Available: <https://www.st.com/>. [Erişim Tarihi: 04 01 2020].
- [23] A. Technology, «Driving Unconventional Growth Through the Industrial Internet of Things,» 2015. [Çevrimiçi]. Available: https://www.accenture.com/ph-en/_acnmedia/Accenture/next-gen/reassembling-industry/pdf/Accenture-Driving-Unconventional-Growth-through-IIoT.pdf. [Erişim Tarihi: 04 01 2020].
- [24] T. Kuzin and T. Borovicka, "Early Failure Detection for Predictive Maintenance of Sensor Parts," in Information Technologies - Applications and Theory, Tatranské Matliare, Slovakia, 2016.
- [25] C. Bishop, Pattern Recognition and Machine Learning, Springer, 2006.
- [26] A. Prokhorov, «Correlation (in statistics),» Encyclopedia of Mathematics, 11 02 2011. [Çevrimiçi]. Available: [http://www.encyclopediaofmath.org/index.php?title=Correlation_\(in_statistics\)&olidid=11629](http://www.encyclopediaofmath.org/index.php?title=Correlation_(in_statistics)&olidid=11629). [Erişim Tarihi: 04 01 2020].
- [27] W. J. Dixon, «Analysis of Extreme Values,» The Annals of Mathematical Statistics, cilt 21, no. 4, pp. 488-506, 1950.
- [28] H. Cramér, Mathematical methods of statistics, Princeton Univ. Press, 1946.

- [29] A. B. Sharma, L. Golubchi ve R. Govindan, «Sensor Faults: Detection Methods and Prevalence in Real-World Datasets,» ACM Transactions on Sensor Networks, cilt 6, no. 3, pp. 1-39, 2010.
- [30] G. YAMAN ve H. M. KARADAYI, «Titreşim Analizi ile Pompalarda Arıza Tespiti ve Kestirimci Bakım İçin Örnek Bir Çalışma,» Tesisat Mühendisliği, cilt 140, no. 140, pp. 37-51, 2014.
- [31] J. Cohen, C. P., S. West and L. Aiken, Applied multiple regression/correlation analysis for the behavioral sciences (3rd ed.), Mahwah, New Jersey: Psychology Press, 2002.
- [32] Hopkins, G. V. Glass and K. D., Statistical methods in education and psychology, Boston: Allyn and Bacon, 1996.
- [33] G. S. Maddala, Introduction to Econometrics, Wiley, 2009.
- [34] F. E. Grubbs, «Procedures for Detecting Outlying Observations in Samples,» Technometrics, cilt 11, no. 1, pp. 1-21, 1969.
- [35] W. Chauvenet, A Manual of Spherical and Practical Astronomy, London: J. B. Lippincott & Company, 1863.
- [36] B. PEIRCE ve L. D, «CRITERION FOR THE REJECTION OF DOUBTFUL OBSERVATIONS.,» ASTRONOMICAL JOURNAL, cilt 2, no. 45, pp. 161-163, 1852.
- [37] V. Chandola, A. Banerjee ve V. Kumar, «Anomaly detection: A survey,» ACM Computing Surveys (CSUR), cilt 41, no. 3, pp. 1-58, 2009.
- [38] P. Zhao, M. Kurihara, J. Tanaka, T. Noda, S.Chikuma and T. Suzuki, "Advanced correlation-based anomaly detection method for predictive maintenance," in IEEE International Conference on Prognostics and Health Management (ICPHM), Dallas, TX, USA, 2017.

- [39] S. Zhong, H. Luo, L. Lin and X. Fu, "An improved correlation-based anomaly detection approach for condition monitoring data of industrial equipment," *Prognostics and Health Management (ICPHM)*, pp. 1-5, 2016.
- [40] J. Ding, Y. Liu and L. Zhang, "LCAD: A Correlation Based Abnormal Pattern Detection Approach for Large Amount of Monitor Data," *Web Technologies and Applications*, vol. 8709, pp. 550-558, 2014.
- [41] Intel, «7.Nesil Intel® Core™ İşlemci Ailesi,» Intel, [Çevrimiçi]. Available: <https://www.intel.com.tr/content/www/tr-tr/processors/core/7th-gen-core-family-desktop-brief.html>. [Erişim Tarihi: 04 01 2020].
- [42] NVIDIA, «GeForce GTX 1070,» NVIDIA, [Çevrimiçi]. Available: <https://www.nvidia.com/tr-tr/geforce/products/10series/geforce-gtx-1070/>. [Erişim Tarihi: 04 01 2020].
- [43] Microsoft, «Visual Studio Code,» Microsoft, [Çevrimiçi]. Available: <https://code.visualstudio.com/>. [Erişim Tarihi: 04 01 2020].
- [44] P. S. Foundation, «Python,» Python Software Foundation, [Çevrimiçi]. Available: <https://www.python.org/>. [Erişim Tarihi: 04 01 2020].
- [45] GOOGLE, «TensorFlow,» GOOGLE, [Çevrimiçi]. Available: <https://www.tensorflow.org/>. [Erişim Tarihi: 04 01 2020].
- [46] «Keras: The Python Deep Learning library,» [Çevrimiçi]. Available: <https://keras.io/>. [Erişim Tarihi: 04 01 2020].
- [47] A. Saxena, K. Goebel, D. Simon and N. Eklund, "Damage propagation modeling for aircraft engine run-to-failure simulation," in *2008 International Conference on Prognostics and Health Management*, Denver, 2008.
- [48] X. Li and X. Wu, "Constructing Long Short-Term Memory based Deep Recurrent Neural Networks for Large Vocabulary Speech Recognition," in *IEEE International*

Conference on Acoustics, Speech and Signal Processing, Brisbane, QLD, Australia, 2015.

- [49] W. Zhang, "Shift-invariant pattern recognition neural network and its optical architecture," in Proceedings of annual conference of the Japan Society of Applied Physics, 1988.
- [50] V. Mathew, T. Toby, V. Singh, B. Rao and M. Kumar, "Prediction of Remaining Useful Lifetime (RUL) of turbofan engine using machine learning,," in IEEE International Conference on Circuits and Systems (ICCS), Thiruvananthapuram, India, 2017.
- [51] N. Gugulothu, V. TV, P. Malhotra, L. Vig, P. Agarwal and G. Shroff, "Predicting Remaining Useful Life using Time Series Embeddings based on Recurrent Neural Networks," in 2nd ML for PHM Workshop at SIGKDD, Halifax, Canada, 2017.
- [52] M. A. Kramer, «Nonlinear principal component analysis using autoassociative neural networks,» AICHE Journal, cilt 37, no. 2, pp. 233-243, 1991.
- [53] J. B. Diederik Kingma, «Adam: A Method for Stochastic Optimization,» %1 içinde 3rd International Conference for Learning Representations, San Diego, 2015.
- [54] P. Yu, X. Yong, L. Datong ve P. Xiyuan, «Sensor selection with grey correlation analysis for remaining useful life evaluation,» roceedings of the Annual Conference of the Prognostics and Health Management Society , pp. 343-352, 2012.
- [55] X. Shi, Z. Chen, H. Wang, D.-Y. Yeung, W.-K. Wong and W.-c. Woo, "Convolutional LSTM Network: A Machine Learning Approach for Precipitation Nowcasting," in Proceeding NIPS'15 Proceedings of the 28th International Conference on Neural Information Processing Systems, Montreal, Canada, 2015.
- [56] C. Spearman, «The Proof and Measurement of Association between Two Things,» The American Journal of Psychology, cilt 15, no. 1, pp. 72-101, 1904.

ÖZGEÇMİŞ

Kişisel Bilgiler

Adı Soyadı Anıl ELADAĞ
Doğum Yeri Yeni Mahalle / ANKARA
Doğum Tarihi 09.06.1992
Uyruğu ☒ T.C. ☐ Diğer:
Telefon 05383291278
E-Posta Adresi anleladag@gmail.com



Eğitim Bilgileri Lisans

Üniversite İstanbul Üniversitesi
Fakülte Mühendislik Fakültesi
Bölümü Bilgisayar Mühendisliği
Mezuniyet Yılı 2015

Yüksek Lisans

Üniversite Marmara Üniversitesi
Enstitü Adı Fen Bilimleri Enstitüsü
Anabilim Dalı Bilgisayar Mühendisliği Anabilim Dalı
Programı Bilgisayar Mühendisliği – Tezli Yüksek Lisans

Makale ve Bildiriler

Korelasyon ve Aykırılık Tespitinin Kestirimli Bakıma Etkisi, 5. ULUSLARARASI MUHENDİSLİK MİMARLIK VE TASARIM KONGRESİ, 2019