

KARADENİZ TEKNİK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI

ENDÜSTRİYEL NESNELERİN İNTERNETİNDE DERİN ÖĞRENME VE
DEMPSTER SHAFER TEORİSİNE DAYALI KARAR VERME

DOKTORA TEZİ

Nader EBRAHİMPUR

EKİM 2022
TRABZON



**KARADENİZ TEKNİK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI

**ENDÜSTRİYEL NESNELERİN İNTERNETİNDE DERİN ÖĞRENME VE
DEMPSTER SHAFER TEORİSİNE DAYALI KARAR VERME**

Nader EBRAHIMPOUR

**Karadeniz Teknik Üniversitesi Fen Bilimleri Enstitüsünce
“DOKTOR (BİLGİSAYAR MÜHENDİSLİĞİ)”
Unvanı Verilmesi İçin Kabul Edilen Tezdir.**

**Tezin Enstitüye Verildiği Tarih : 08/09/2022
Tezin Savunma Tarihi : 13/10/2022**

Tez Danışmanı : Doç. Dr. Tuğrul ÇAVDAR

Trabzon 2022

ÖNSÖZ

Nesnelerin İnternetinin (Internet of Things, IoT) endüstrideki uygulaması, günümüzde endüstri alanında önemli konulardan biri haline gelen Endüstriyel Nesnelerin İnterneti (Industrial Internet of Things, IIoT) olarak adlandırılmaktadır. IIoT sensörleri tarafından toplanan büyük miktarda veri, analizleri verimli makine öğrenimi yöntemleri gerektirir. Bilgi toplamak için birkaç farklı sensörün kullanıldığı durumlar, kararlarda belirsizliğe neden olabilir ve bu durumlarda güvenilir kararlar vermek için karar füzyonu yöntemleri kullanılabilir. Bu nedenle, bu çalışmada, Makine Öğrenimi (Machine Learning, ML), Derin Öğrenme (Deep Learning, DL) ve Dempster-Shafer (DS) teorileri kullanılarak IIoT'de karar verme yöntemleri sunulmaktadır. Önerilen yöntemler, kararların doğruluğunu iyileştirmekte ve belirsizliklerini önemli ölçüde azaltmaktadır.

Doktora eğitimim boyunca her aşamasında belirleyici rol oynayan, yapıcı ve akılcı katkılarıyla bana yol gösteren ve desteğini benden esirgemeyen, çok değerli danışman hocam Sayın Doç. Dr. Tuğrul ÇAVDAR'a sonsuz teşekkürlerimi sunarım. Çalışma süresince bilgi ve tecrübelerini esirgemeyen Sayın Prof. Dr. Kadir TÜRK ve Sayın Dr. Öğr. Üyesi Selçuk CEVHER'e en içten teşekkürlerimi sunarım.

Ayrıca hayatım boyunca beni destekleyen ve bu günlere ulaşmamı sağlayan aileme özellikle çok değerli babam Haydar EBRAHIMPOUR'a, sevgili annem Mahzar EBRAHIMPOUR'a şükranlarımı sunarım. Bu çalışmam boyunca sabır, anlayış ve her türlü desteğini esirgemeyen, çok sevdiğim eşim Zehra'ya, güzel kızım Alin'e sonsuz teşekkürlerimi sunarım.

Nader EBRAHIMPOUR
Trabzon, 2022

TEZ ETİK BEYANNAMESİ

Doktora Tezi olarak sunduğum "Endüstriyel Nesnelerin İnternetinde Derin Öğrenme ve Dempster Shafer Teorisine Dayalı Karar Verme" başlıklı bu çalışmayı baştan sona kadar danışmanım Doç. Dr. Tuğrul ÇAVDAR'ın sorumluluğunda tamamladığımı, verileri/örnekleri kendim topladığımı, deneyleri/analizleri ilgili laboratuvarlarda yaptığımı/yaptırdığımı, başka kaynaklardan aldığım bilgileri metinde ve kaynakçada eksiksiz olarak gösterdiğimi, çalışma sürecinde bilimsel araştırma ve etik kurallara uygun olarak davrandığımı ve aksinin ortaya çıkması durumunda her türlü yasal sonucu kabul ettiğimi beyan ederim. 13/10/2022

Nader EBRAHIMPOUR

İÇİNDEKİLER

	<u>Sayfa No</u>
ÖNSÖZ.....	III
TEZ ETİK BEYANNAMESİ.....	IV
İÇİNDEKİLER.....	V
ÖZET	VII
SUMMARY	VIII
ŞEKİLLER DİZİNİ	IX
TABLolar DİZİNİ.....	XI
SEMBOLLER DİZİNİ	XIII
1. GENEL BİLGİLER.....	1
1.1. Giriş	1
1.2. İlgili Çalışmalar	6
1.3. Nesnelerin İnterneti	11
1.3.1. IoT nin Tarihçesi.....	14
1.3.2. Endüstriyel Nesnelerin İnterneti	16
1.4. Makine Öğrenmesi.....	18
1.4.1. Derin Öğrenme	20
1.4.1.1. Evrişimli Sinir Ağları	23
1.4.1.2. Optimizasyon Yöntemleri.....	25
1.5. Anomali Algılama	26
1.5.1. Anomali Algılama Yöntemleri	26
1.5.1.1. Sınıflandırmaya Dayalı Yöntemler	27
1.5.1.2. En Yakın Komşuya Dayalı Yöntemler	27
1.5.1.3. Kümelemeye Dayalı Yöntemler	28
1.6. Bilgi Füzyonu	28
1.6.1. Dempster-Shafer Yöntemi	31
2. YAPILAN ÇALIŞMALAR.....	34
2.1. Küçük Boyutlu IIoT'ler için DS Kullanarak Karar Verme	34
2.1.1. Veri Dönüştürme ve Özellik Çıkarma	35
2.1.2. Sınıflandırma	36
2.1.2.1. Dendrogram Tabanlı Destek Vektör Makinesi (DSVM).....	37
2.1.2.2. Yapay Sinir Ağları	39

2.1.3.	Veritabanı	40
2.1.4.	Bilgi Füzyonu	40
2.1.4.1.	Dempster-Shafer Teorisi.....	41
2.2.	Tek Boyutlu Evrişimli Sinir Ağları ve DS Teorinin Kullanarak IIoT'lerde Anormallik İçin Karar Verme (DS-1DCNN)	44
2.2.1.	Tek Boyutlu Evrişimli Sinir Ağları	46
2.2.1.1.	Evrişim Katmanı.....	47
2.2.1.2.	Havuzlama Katmanı	47
2.2.1.3.	Düzleştirme (flatten) ve Tamamen Bağlı Katman.....	48
2.2.1.4.	Dempster Shafer Birleştirme Kuralları.....	48
2.2.2.	Adam Optimize Edici	48
2.2.3.	Önerilen Yöntem Ayrıntıları.....	49
3.	BULGULAR.....	53
3.1.	Birinci Önerilen Yöntem	53
3.1.1.	Asenkron Motorun Veri Seti	53
3.1.2.	Çok Sensörlü Veri Füzyonuna Dayalı Aktivite Tanıma Veri Seti (AReM).....	60
3.2.	İkinci Önerilen Yöntem	63
3.2.1.	Mill Veri Seti	63
3.2.2.	Değerlendirme Kriterleri	64
3.2.3.	Önerilen Yöntemin Simülasyonu ve Sonuçları	65
4.	SONUÇLAR.....	71
5.	ÖNERİLER.....	74
6.	KAYNAKLAR	75
ÖZGEÇMİŞ		

Doktora Tezi

ÖZET

ENDÜSTRİYEL NESNELERİN İNTERNETİNDE DERİN ÖĞRENME VE DEMPSTER
SHAFFER TEORİSİNE DAYALI KARAR VERME

Nader EBRAHIMPOUR

Karadeniz Teknik Üniversitesi
Fen Bilimleri Enstitüsü
Bilgisayar Mühendisliği Anabilim Dalı
Danışman: Doç. Dr. Tuğrul ÇAVDAR
2022, 87 Sayfa

IIoT'nin ana hedeflerinden biri, endüstriyel ortamlardaki olayları, beklenmedik değişiklikleri ve anomalileri izlemek ve otomatik olarak tespit etmek için endüstriyel ortamlardan veri toplamaktır. Endüstriyel ortamlarda toplanan hacimli verilerin işlenmesine yönelik DL gibi verimli ML yöntemlerinin kullanılması gereklidir. Sensör teknolojisinin gelişmesine paralel olarak farklı sensör kaynaklarından elde edilen veriler kullanılarak amaca uygun çıkarımlar yapılabilir. Bu da tek ve güvenilir bir karar elde etmek için ortaya çıkan alt kararların birleştirilmesini gerektirir. Birçok kararı birleştirmek için çok kullanılan yöntemlerden biri DS'in birleştirme teorisidir. Bu nedenle, bu tezde IIoT anomalileri hakkında karar vermek için iki yöntem önerilmiştir. Birinci yöntem sınıflandırma için Yapay Sinir Ağları (Artificial Neural Network, ANN), Destek Vektör Makinesi (Support Vector Machine, SVM) ve kararları birleştirmek için DS teorisini kullanır. Bu yöntem nispeten az miktarda veriye sahip küçük ve orta ölçekli endüstriyel birimler için uygundur. Önerilen ikinci yöntemde, bilgi sınıflandırması için Tek Boyutlu Evrişimli Sinir Ağları (One Dimensional Convolutional Neural Network, 1DCNN) ve kararları birleştirmek için DS birleştirme teorisi kullanılır. Bu yöntem, DL ağlarının devasa verileri işleme gücü nedeniyle büyük endüstriyel birimlerde kullanıma uygundur.

Anahtar Kelimeler: Karar verme, karar füzyonu, Dempster-Shafer, Tek Boyutlu Evrişimli Sinir Ağları, Derin Öğrenme, Anomali

PhD. Thesis

SUMMARY

DEEP LEARNING AND DEMPSTER SHAFER THEORY BASED DECISION-MAKING IN
THE INDUSTRIAL INTERNET OF THINGS

Nader EBRAHIMPOUR

Karadeniz Technical University
Institute of Science and Technology
Department of Computer Engineering
Supervisor: Associate Professor Dr. Tuğrul ÇAVDAR
2022, 87 Pages

One of the main goals of IIoT is to collect data from industrial environments to monitor and automatically detect events and unexpected changes and anomalies in industrial environments. Because the data collected in industrial environments is voluminous, it is needed to use efficient machine learning methods such as deep learning (DL) to analyze this voluminous data. According to the development of sensor technology, different sensor sources can be used by IIoT to obtain proper inference, which requires combining the resulting decisions to get a single and reliable decision. One of the most used methods for the fusion of many decisions is Dempster Shafer's (DS) combining theory. Therefore, two methods are proposed in this thesis to make decisions about IIoT anomalies. The first uses artificial neural networks (ANN) and support vector machine (SVM) for classification and DS theory to combine decisions. This method is suitable for small and medium-sized industrial units with relatively small data. The second proposed method uses one-dimensional convolutional neural networks (1DCNNs) for information classification and DS combination theory to combine decisions. This method is suitable for large industrial units due to the massive data processing power of deep learning networks.

Key Words: Decision-making, decision fusion, Dempster–Shafer, 1D Convolutional Neural Networks, Deep Learning, Anomaly

ŞEKİLLER DİZİNİ

	<u>Sayfa No</u>
Şekil 1. IoT Ekosistemi	12
Şekil 2. IoT bağlantısının boyutu	13
Şekil 3. ITU tarafından IoT'nin Katmanlı mimarisi	14
Şekil 4. IoT ve IIoT ve Endüstri 4.0	17
Şekil 5. IIoT'inin yapısı.....	18
Şekil 6. AI, ML ve DL.....	19
Şekil 7. Derin öğrenme.....	20
Şekil 8. DL uygulama teknikleri	21
Şekil 9. Temel evrişimli sinir ağlarının genel şeması.....	23
Şekil 10. İnsan bilgi füzyon sisteminin basit gösterimi.....	29
Şekil 11. Sensör seviyesinde füzyonu	30
Şekil 12. Özellik seviyesinde füzyonu	30
Şekil 13. Karar seviyesinde füzyonu	30
Şekil 14. DS teorisinde kararların nasıl birleştirildiğini gösteren basit bir grafik.....	32
Şekil 15. Önerilen yöntemin mimarisi.....	35
Şekil 16. AHC yöntemi ile sınıfları hiyerarşik olarak gruplama.....	38
Şekil 17. Önerilen yöntemin akış şeması	41
Şekil 18. IIoT Mimarisi ve önerilen modelin konumu	45
Şekil 19. Karar verme için önerilen yöntemin yapısı	45
Şekil 20. Tek boyutlu evrişimli sinir ağı (1DCNN) mimarisi	46
Şekil 21. Asenkron motorun yapısı	53
Şekil 22. DSVM ve ANN sınıflandırıcıları kullanarak sınıflandırıcıların doğruluk yüzdesini karşılaştırma	56
Şekil 23. DSVM ve ANN sınıflandırıcı sonuçlarının karar füzyonu için doğruluk yüzdesinin karşılaştırılması.....	58
Şekil 24. DSVM ve ANN sınıflandırıcılarının ses ve titreşim üzerindeki doğruluklarının ve tüm sınıflandırıcıların sonuçlarına karar füzyonunun uygulanmasının grafiksel karşılaştırması.....	59
Şekil 25. DSVM ve ANN sınıflandırıcıları için birlikte karar füzyonu sonucu.....	61
Şekil 26. Önerilen, LI-ESN ve IDNNs yöntemleri için doğruluk karşılaştırılması.....	62
Şekil 27. Veri setinin 147. çalışması	63

Şekil 28.	1DCNN_CURRENT, 1DCNN_SOUND, 1DCNN_VIBRATION, 1DCNN_DLFUSION ve DS-1DCNN'nin duyarlılık karşılaştırması.....	69
Şekil 29.	1DCNN_CURRENT, 1DCNN_SOUND, 1DCNN_VIBRATION, 1DCNN_DLFUSION ve DS-1DCNN'nin kesinlik karşılaştırması	70



TABLolar DİZİNİ

Sayfa No

Tablo 1.	Birinci sanayi devriminden dördüncü sanayi devrimine ekonomik üretim gelişmelerinin genel süreci	2
Tablo 2.	Tahmine dayalı bakım ve anomali algılamada makine öğrenimi modellerini kullanan inceleme makalelerinin özetidir	10
Tablo 3.	Zamanla IoT	15
Tablo 4.	DL uygulama tekniklerinin karşılaştırılması.....	24
Tablo 5.	DL ile ilgili avantajlar ve zorluklar	24
Tablo 6.	Sınıflandırmaya göre yöntemin avantajlar ve dezavantajları ve güçlü yönleri.....	27
Tablo 7.	En yakın komşuya göre yöntemin avantajlar ve dezavantajları	28
Tablo 8.	Kümelemeye dayalı yöntemin avantajlar ve dezavantajları.....	28
Tablo 9.	Özellik çıkarımı için bazı önemli istatistiksel fonksiyonlar	36
Tablo 10.	Önerilen yöntem sözde kodunun giriş, çıkış ve başlatma parametreleri.....	50
Tablo 11.	Önerilen yöntemin sözde kodu.....	51
Tablo 12.	Asenkron motorun özellikleri.....	53
Tablo 13.	Sağlıklı ve dört arıza durumu sınıfları.....	54
Tablo 14.	Titreşim özelliklerine ANN sınıflandırıcı uygulamasının sonucu	54
Tablo 15.	Ses özelliklerine ANN sınıflandırıcı uygulamasının sonucu	55
Tablo 16.	Titreşim özelliklerine DSVM sınıflandırıcı uygulamasının sonucu	55
Tablo 17.	Ses özelliklerine DSVM sınıflandırıcı uygulamasının sonucu	56
Tablo 18.	Ses ve titreşim özellikleri üzerinde DSVM sınıflandırıcı için karar füzyonu uygulamasının sonucu	57
Tablo 19.	Ses ve titreşim özellikleri üzerinde ANN sınıflandırıcı için karar füzyonu uygulamasının sonucu	57
Tablo 20.	Ses için DSVM ve ANN sınıflandırıcılarından gelen karar füzyon uygulamasının sonucu	57
Tablo 21.	Titreşim için DSVM ve ANN sınıflandırıcılarından gelen Karar Füzyon uygulamasının sonucu	57
Tablo 22.	İki Sınıflandırıcının tüm sonuçlarına Karar Füzyon uygulamasının sonucu.....	58
Tablo 23.	DSVM ve ANN sınıflandırıcının ses ve titreşim üzerindeki doğruluğu ve tüm sınıflandırıcıların sonuçlarına karar füzyonu uygulaması	59
Tablo 24.	DSVM sınıflandırıcı için karar füzyonu sonucu	60
Tablo 25.	ANN sınıflandırıcı için karar füzyonu sonucu	60

Tablo 26.	DSVM ve ANN sınıflandırıcıları için birlikte karar füzyonu sonucu.....	61
Tablo 27.	Önerilen, LI-ESN ve IDNNs yöntemleri için doğruluk karşılaştırılması.....	62
Tablo 28.	Mill veri seti özellikleri	64
Tablo 29.	1DCNN kullanılarak sınıflandırılan akım verilerin karışıklık matrisi	65
Tablo 30.	1DCNN kullanılarak sınıflandırılan ses verilerin karışıklık matrisi	66
Tablo 31.	1DCNN kullanılarak sınıflandırılan titreşim verilerin karışıklık matrisi	66
Tablo 32.	Veri düzeyinde füzyon ve 1DCNN kullanılarak sınıflandırılan akım, ses ve titreşimin karışıklık matrisi.....	66
Tablo 33.	Veri düzeyinde füzyon ve LSTM kullanılarak sınıflandırılan akım, ses ve titreşimin karışıklık matrisi.....	67
Tablo 34.	Veri düzeyinde füzyon ve RF kullanılarak sınıflandırılan akım, ses ve titreşimin karışıklık matrisi	67
Tablo 35.	DS-1DCNN'nin karışıklık matrisi	68
Tablo 36.	1DCNN-CURRENT, 1DCNN-SOUND, 1DCNN-VIBRATION, 1DCNN-DIFFUSION, LSTM, RF ve DS-1DCNN'nin duyarlılık karşılaştırması.....	68
Tablo 37.	1DCNN_CURRENT, 1DCNN_SOUND, 1DCNN_VIBRATION, 1DCNN_DLFUSION ve DS-1DCNN'nin kesinlik karşılaştırması	69

SEMBOLLER DİZİNİ

DL	: Derin Öğrenme (Deep Learning)
ML	: Makine Öğrenme (Machine Learning)
AI	: Yapay Zeka (Artificial Intelligence)
DS	: Dempster Shafer
ANN	: Yapay Sinir Ağı (Artificial Neural Network)
DSVM	: Dendrogram-Tabanlı Destek Vektör Makinesi (Dendrogram-Based Support Vector Machine)
SVM	: Destek Vektör Makinesi (Support Vector Machine)
CNN	: Evrişimsel Sinir Ağı (Convolutional Neural Network)
1DCNN	: Bir Boyutlu Evrişimsel Sinir Ağı (One Dimensional Convolutional Neural Network)
IoT	: Nesnelerin İnterneti (Internet of Things)
IIoT	: Endüstriyel Nesnelerin İnterneti (Industrial Internet of Things)
FFO	: Meyve Sineği Optimizasyonu (Fruit Fly Optimization)
RF	: Rastgele Orman (Random Forest)
LSTM	: Uzun Kısa Süreli Bellek (Long Short-Term Memory)
ITU	: Uluslararası Telekomünikasyon Birliği (International Telecommunication Union)
M2M	: Makineden Makineye (Machine to Machine)

1. GENEL BİLGİLER

1.1. Giriş

İnternet yalnızca insanların bağlantı kurabileceği ve onun yeteneklerinden kişisel olarak sadece bireylerin yararlanabileceği bir platform olarak düşünülürdü, ancak IoT kavramı ile birlikte çevremizde gördüğümüz fiziksel nesneler de internetin bir parçası haline geldi. . Artık herhangi bir fiziksel nesnenin internete bağlanarak veya diğer iletişim araçlarını kullanarak diğer nesnelerle irtibat kurması IoT'nin ana fikridir. . IoT cihazları, internete bağlı birçok fiziksel nesnenin görevlerini ve bilgilerini birbirleriyle ve insanlarla paylaşması ve geri dönüşler almasıyla amaca uygun kararlar verir. IoT'nin en önemli faktörü çeşitli iletişim teknolojileri ve çözümlerinin entegrasyonudur. Tanımlama ve izleme teknolojileri, kablolu ve kablosuz sensörler, aktif ağlar ve akıllı nesneler IoT'nin en önemli uygulama alanlarıdır. İnternetin artan trendiyle birlikte, IoT teknolojisi kilit ve önemli bir rol oynayacaktır. Çünkü IoT teknolojisi her nesneye bu çağın en önemli unsuru olan internete erişim imkânı sağlamakta ve bulut bilişim platformları ile bu nesnelere daha ileri düzeyde yetenekler kazandırmaktadır. Böylece, her alanda nesneler düzeyinde iletişimin yaygınlaşması günlük hayatımızı kolaylaştırdığı gibi birçok alandaki bilimsel çalışmalara da büyük katkılar sunmaktadır. Organizasyonların hayatta kalmasında rekabetin ilk rolü oynadığı bir dünyada, IoT'nin önemi ve gerekliliği tartışılmazdır.

IoT kavramının ortaya çıkmasından bu yana, araştırmacılar tarafından birçok farklı uygulama alanında kullanılan ürünler geliştirilmiştir. IoT'nin en önemli uygulama alanlarından biri de endüstridir [1]. Güvenlik, üretim, işletme ve çevre koruma için ürün ve hizmetlere yönelik artan taleple birlikte, endüstriyel ortamlarda çok sayıda sensör ve kablosuz cihaz kullanımı yaygınlaşmıştır. Buna dayanarak, endüstriyel IoT çözümlerini ifade eden IIoT kavramı yaygın olarak kullanılmaya başlanmıştır [2, 3]. Son yıllarda, teknolojinin endüstriyel ortamlarda uygulanması dikkate değer bir hıza sahiptir. Bugün IIoT, endüstriyel ortamları kontrol etmek için yeni çözümler ortaya koymuştur. Üretim, ilaç, gıda, yakıtlar gibi çeşitli endüstriyel alanlarda, ürünlerin kalitesini en iyi şekilde korumak ve işgücünün güvenilirliğini ve verimliliğini artırmak için IIoT çözümleri kullanılmaktadır. IIoT, endüstri ortamlarından büyük miktarda veri toplayarak ve farklı cihazlardan farklı zamanlarda elde edilen bu büyük miktardaki veriyi analiz ederek, verimliliği artırmak ve

güvenliği sağlamak için uygun maliyetli çözümler sunmayı amaçlamaktadır. Yüksek verimliliğin yanı sıra çevre dostu endüstriyel ürün ve hizmet üretimini de hedeflemektedir [3]. IIoT'nin yetenekleri, endüstriyel ekipmanın ağa bağlanmasına ve sistemler ile ilgili verileri paylaşmasına bağlıdır; Böylece, veriler toplanabilir ve analiz edilebilir. Veri analizinin sonuçları, yöneticilerin yönettikleri grubun verimliliğini ve üretkenliğini önemli ölçüde etkileyen daha verimli kararlar almalarını sağlar [4].

Tablo 1. Birinci sanayi devriminden dördüncü sanayi devrimine ekonomik üretim gelişmelerinin genel süreci

Sanayi Devrimi	Zaman Dilimi	Ana Özellikleri	Önemli Kazanımları
Birinci	1760-1840	Buhar motoru - demiryolu	Mekanik üretim
İkinci	19. yüzyılın sonu - 20. yüzyılın başı	Elektrik akımı	Montaj hatları - seri üretim
Üçüncü	1960'lar - 20. yüzyılın sonları	Dijital devrim - yarı iletkenlerin gelişimi - ana bilgisayarlar - dayalı hesaplamalar	Kişisel bilgisayarlar - İnternet
Dördüncü	21. yüzyılın başlarından bugüne kadar	Mobil İnternet - daha küçük sensörler - ucuz ve gücünü sensörler	AI - ML, IoT - Siberfizik Sistemler

Bilimsel çalışmalar sonucundan elde edilen her sonuç veya bilimsel bir ilerleme, hizmetlerin optimizasyonuna yol açmış ve yaşamları farklı açılardan değiştiren önemli kazanımlar ve olanaklar sağlamıştır. Örneğin, “Sanayi Devrimi” veya “Endüstri Devrimi” olarak bilinen aşama ile birlikte endüstride meydana gelen değişim ve ilerleme, doğal olarak sağlanan imkânlar ve ilerlemeler doğrultusunda iletişim türü, iş ve meslek, tedavi yöntemi, eğitim gibi hayatın farklı yönlerinin de değişmesine neden olmuştur. Bu açıdan bakıldığında sanayi devrimi sadece sanayide meydana gelen ilerleme ve değişimden ibaret değildir. Sanayi devrimi, araştırmacıların kullandığı kapsamlı bir terimdir [3]. Genel olarak, insanlık şimdiye kadar üç sanayi devrimi yaşamıştır. İlk sanayi devrimi, 18. yüzyılın sonlarında üretim makinelerinin inşası ve buharlı makineler kullanan fabrikaların kurulmasıyla gerçekleştirilmiştir. İkinci sanayi devrimi, sanayinin elektrifikasyon çağı olarak adlandırılmıştır. 1970-2000 yılları arasında yaşanan üçüncü sanayi devrimi, bilgi ve internet devrimine dayanan evrim aşamalarını tamamlamıştır. Şimdi IoT ve siber-fiziksel sistemlere dayanan ve önceki fiziksel sistemler ile modern siber sistemlerin birleşimi olan

dördüncü sanayi devrimi büyüme aşamasındadır. Özellikle dördüncü sanayi devrimi, farklı bilimsel alanların birbiriyle hizalanması ve etkileşimidir. Dördüncü sanayi devriminin gelişim sürecinde, bilgi ve teknoloji yoluyla sanayide üretimin önemli ölçüde artması beklenmektedir. Aslında dördüncü sanayi devrimi verilere dayanmaktadır [5]. Tablo 1, üretimdeki değişikliklere odaklanarak sanayi devrimi gelişmelerinin genel sürecini göstermektedir. Tablo 1 de belirtildiği gibi, ML ve Yapay Zeka (Artificial Intelligence, AI), dördüncü sanayi devrimi ile bağlantılı olarak kilit rol oynayan teknolojiler arasındadır.

Endüstri 4.0, endüstrideki organizasyonel süreçleri değiştirmek isteyen firmalar için yeniliğin merkezde olduğu bir devrim yarattı. Bu durum endüstriyel ekipmanın esas olarak sensörler ve sensör verileri tarafından yönlendirildiği IIoT teknolojisinde patlamaya yol açmıştır. Herhangi bir sektör, üreticinin işlerini daha iyi anlamak ve yönetmek için IIoT den gelen verileri doğru karar verme amacı ile kullanabilir (örneğin, tahmine dayalı analitik bir yolla bir hatayı belirlemek için). Ayrıca veriler, endüstride kullanılan ekipmanın sağlığını kontrol etmek için hayati bir varlık olarak kabul edilebilir [6]. IIoT birçok karmaşık veri türünü içerir: sensör, radyo frekansı, video ve görüntü verileri, tanımlayıcı bilgiler, yerel, çevresel ve sensör ağı verileri. Bu durum IIoT'deki yönetim, analiz ve veri madenciliğinde önemli zorlukları da beraberinde getiriyor. Bu nedenle, akıllı üretim, verimli ağ oluşturma, kaza tespiti, sıcaklık izleme, hava kirliliği, anomali tespiti gibi gerçek dünya uygulamalarından sürekli veri akışını araştırmak için analitik veri sistemlerine hayati bir ihtiyaç vardır [2, 7].

İmalat sanayilerinin IIoT'de karşılaştığı en büyük sorunlardan biri, imalat süreçlerindeki anomalilerin yalnızca %20'sini önceden tespit edebilmesidir. Bu, çoğu zaman anomalilerin tespit edilmediği ve anomali tespiti ile ilgili verimsiz teknikler nedeniyle endüstrinin bunlarla başa çıkamadığı anlamına gelir. Anomali tespiti, aynı veri setinde beklenen bir kalıba ait olmayan ve normalde insan gözüyle görülemeyen değişkenleri veya öğeleri tanımlama işlemidir [8]. Arızanın erken belirtileri olarak kabul edilebilecek bu tür anomaliler, doğru zamanda tespit edilemediğinde genellikle ekipman arızasına yol açar. Anomalilerin teşhis edilmesi zordur çünkü:

- Mevcut bilgiler onları sınıflandırmak için yeterli değildir.
- Çoğu zaman, verilerde anomaliyi doğru bir şekilde tanımlayacak hiçbir örnek yoktur.

Endüstriyel sistemler tarafından toplanan büyük miktarda veri, endüstriyel üretim hattı boyunca meydana gelen süreçler, olaylar ve alarmlar hakkında bilgiler içerir. Ayrıca bu

veriler işlenip analiz edildiğinde üretim süreci ve sistem dinamikleri ile ilgili değerli bilgiler elde edilebilir. Veriye dayalı analitik yaklaşımlar kullanılarak stratejik karar verme için yorumlanabilir sonuçlar elde edilebilir. Bu analiz sonuçlarının faydalarından bazıları şunlardır; i) bakım ve onarım maliyetlerinin düşürülmesi, ii) cihaz hatalarının azaltılması, iii) onarım arıza sürelerinin azaltılması, iv) yedek parça envanterinin azaltılması, v) yedek parça ömrünün artırılması, vi) üretimi artırılması, vii) operatör güvenliğinin sağlanması ve viii) onarımın doğrulanmasıdır [9, 10]. Bahsedilen faydaların bakım yöntemiyle güçlü bir bağlantısı vardır. Endüstrilerde, ekipman bakımı ve onarımı, ekipmanın çalışma süresini ve üretkenliğini etkileyen önemli bir unsurdur. Bu nedenle, üretim süreçlerinin herhangi bir arıza sonucunda durmasını önlemek için ekipman arızasını doğru zamanda tespit etmek ve bu arızaya çözüm bulmak gerekir [11]. Bakım ve onarım yöntemleri üç genel kategoriye ayrılabilir: düzeltici, önleyici ve kestirimci bakım ve onarımlar. Genel olarak, uygun bir bakım ve onarım stratejisi, ekipmanın durumunu iyileştirmeli, ekipmanın arıza oranını düşürmeli ve bakım ve onarım maliyetlerini en aza indirmelidir, bu da sonuçta ekipmanın kullanım ömrünün maksimize edilmesini sağlar. Bu nedenle kestirimci bakım ve onarım stratejisi, diğer stratejilere göre daha fazla öne çıkan bir yöntemdir ve ekipman kullanımını optimize etme yeteneği nedeniyle farklı sektörlerdeki analistlerin ilgisini çekmektedir [12]. Arıza oluşmadan önce arızayı tahmin etmek, öngörücü bakım ve onarım tanımlanmaktadır. Jardine ve ark.'nın 2006 yılındaki araştırmasına göre [13], teşhis ve tahmin doğrultusunda ekipman durumunu izleyebilen bakım ve onarım yöntemleri üç ana kategoriye ayrılmaktadır: istatistiksel yaklaşımlar; Makine öğrenimi yaklaşımları ve model tabanlı yöntemler. Model tabanlı yaklaşımlar, izlemek için mekanik bilgi ve ekipman teorisi gerektirirken, istatistiksel yaklaşımlar matematiksel arka plan gerektirir. AI yaklaşımları ise, tahmine dayalı bakım ve onarım programları içerisinde giderek daha fazla uygulanmaktadır. Örneğin, Baptista ve diğerleri [14], gelecekte ekipman arızasını tahmin etmek için, destek vektör makinesi gibi bir dizi AI yaklaşımını istatistiksel yaklaşıma dayalı modellerle karşılaştırmıştır. Bu araştırmanın sonuçları, AI yaklaşımının istatistiksel yaklaşımlara göre daha doğru tahminler sağladığını göstermektedir. Son yıllarda, AI alanındaki makine öğrenimi, akıllı tahmin algoritmaları geliştirmek için mevcut en güçlü araçlardan biri olarak tanıtılmaktadır. Makine öğrenimi yaklaşımları, yüksek boyutlu ve çok değişkenli verileri kontrol edebilir ve karmaşık ve dinamik ortamlarda (endüstriyel ortamlar gibi) verilerdeki gizli ilişkileri çıkarabilir. Bu nedenle, makine öğrenimi yöntemleri, tahmine dayalı bakım ve

onarım alanında güçlü tahmine dayalı araçlar sağlar. Ancak bu yöntemlerin performansı, uygun ML yöntemlerinin seçimine bağlıdır [15].

AI ve IoT arasındaki ilişki, insan beyni ve vücut arasındaki ilişkiye çok benzemektedir. Vücudumuz görme, ses ve dokunma gibi duyuşal girdileri toplar. Beynimiz bu verileri alır ve anlamlandırır, ışığı tanınabilir nesnelere ve sesleri anlaşılır konuşmaya dönüştürür. Beynimiz daha sonra bir karar verir ve bir nesneyi almak veya konuşmak gibi hareketleri yapmak için vücuda sinyaller gönderir. IoT'yi oluşturan tüm bağlı sensörler vücudumuza benzemektedir, dünyada ne olduğu hakkında ham bilgi toplar. AI beynimiz gibidir, verileri anlar ve ne yapacağına karar verir. AI, çevreyi anlamak ve karar vermek için IoT'de kullanılmaktadır [7, 16].

Çoklu sınıflandırmayı tek sınıflandırıcıya tercih etmenin birkaç temel faydası vardır. Genelde amaç sınıflandırıcının doğruluğunu ve verimliliğini artırmaktır. Çoklu sınıflandırıcı tercih edilmesinin üç önemli nedeni vardır: (A) Bir çok uygulamada, analiz edilmesi gereken veriler çok fazladır ve tek bir sınıflandırıcı ile analiz edilmesi çok zordur. Bu büyük veriyi bir sınıflandırıcıyla eğitmek oldukça zor ve bazen imkansızdır. Bu durumlarda, çoklu sınıflandırıcı yöntemleri daha etkin olabilir. Bu yaklaşımda, veri kümesi daha küçük veri kümelerine bölünür, çeşitli sınıflandırıcılarla sınıflandırılır ve nihai çıkarımı elde etmek için farklı sınıflandırıcılardan alınan kararlar birleştirilir. (B) Veri kaynakları türlerinin farklı olduğu uygulamalarda, tek bir sınıflandırıcı tüm veri türleri için iyi performans göstermeyebilir. Bu nedenle, bu veri türleri için çoklu sınıflandırıcı kullanmak daha iyi sonuç verebilir. (C) Tüm test verileri için iyi çalışabilecek tek bir sınıflandırıcı bulmak zordur [17].

Modern endüstride, izleme cihazlarının, büyük verileri işleme, düşük veri yoğunluğuna sahip olma, birden fazla sensör barındırma ve hızlı veri akışı sağlama gibi dört özelliği vardır [18]. Endüstride olan hacimli veriler, belirli endüstriyel gereksinimleri karşılamak için çok sensörlü füzyon ve akıllı algoritmalar gerektirir. Tek kanallı sinyal bilgilerini kullanan algoritmalar, akıllı algılama sistemleri için uygun değildir. Son zamanlarda çok sensörlü füzyon algılama sistemleri sıcak bir araştırma konusu haline gelmiştir [19].

Karar füzyonu, birden çoklu kaynaktan elde edilen verilerin katkısını kullanmak için geliştirilen ve kullanılan teorileri, teknikleri ve araçları içerir. Bu nedenle, bilgi füzyonundan elde edilen kararlar veya eylemler, her bir kaynaktan ayrı ayrı elde edilen kararlardan daha verimli sonuçlar üretir [20]. Çoklu sensörler endüstride bilgi toplayarak sistemlerin

doğruluğunu ve sağlamlığını artırır [21]. Karar seviyesi füzyon modeli, çok sayıda bilgi türünü birleştirir ve kesinlik avantajına sahip olan doğru kararlar elde edebilir [22]. Ancak, hata ve anomali tespiti için karar düzeyinde birleştirme yöntemleri nadiren DL modeliyle birleştirilmiştir.

1.2. İlgili Çalışmalar

IIoT'de anomali tespiti ve karar verme için önerilen birkaç yöntem vardır. [23]'te bir Evreşimsel Sinir Ağından (Artificial Neural Networks, ANN) yararlanarak anormal paternleri tespit etmek için bir yöntem önerilmiştir. Liang ve diğerleri Meyve Sineği Optimizasyonu (Fruit Fly Optimization, FFO) algoritmasına dayalı olarak doğru anomali kararları elde etmek için geçmiş verileri kullanarak eşik değerlerini optimize etmek için bir yöntem önerdi [24]. ANN [25, 26], SVM [27] ve Rastgele Orman (Random Forest, RF) [28] gibi diğer bazı AI yöntemleri de anomali tespiti için önerilmiştir.

Shah ve diğerleri [29], motor ekipmanından gelen IIoT verilerindeki anomalileri tespit etmek için birkaç makine öğrenimi algoritmasını gözden geçirmiştir. Belirli bir motorun ne zaman anormal davrandığını ve arızalanabileceğini belirlemek için yakıt sensörü, motor yükü ve yağ basıncı sensörü verilerini analiz etmişlerdir. Makine davranışlarındaki sapmaları tespit etmek için çok değişkenli doğrusal regresyon, Gauss karma modelleri ve zaman serisi veri analizi kullanmışlardır.

Thamilarasu ve diğerleri [30], IoT ortamına uyarlanmış akıllı bir saldırı tespit sistemi geliştirdiler. Özellikle, bu çalışma, IoT tabanlı ağlardaki anormal trafiği tespit etmek için bir DL algoritması kullandılar. Algılama çözümü, bir hizmet olarak güvenlik ve IoT ağ iletişim protokolleri arasında birlikte çalışabilirliği sağlar. Bu araştırma önerilen çerçeveyi gerçek bir ağ ortamı ile simülasyon ortamında değerlendirmiş, izinsiz girişleri ve anormal trafiği tespit etme yeteneği etkin bir şekilde kanıtlanmıştır.

Al-Garadi ve arkadaşları [31] IoT güvenliği için makine öğrenimi ve DL yöntemlerini kullanmayı denediler. Bu çalışmada, IoT güvenlik anomalileri ve tehditleri tanıtılmakta, ardından IoT güvenliğini sağlamak için ML ve DL yöntemlerinin avantaj ve dezavantajları araştırılmaktadır.

Yi ve arkadaşları [32], Evrişimli Sinir Ağı-Uzun Kısa Süreli Bellek (CNN-LSTM) kullanarak IIoT'de doğru anomali tespiti için bir yöntem önerdiler. Bu yöntem, ince taneli

özellikleri türetmek için CNN modelini ve zaman serisi verilerini tahmin etmek için Uzun Kısa Süreli Bellek (Long Short-Term Memory, LSTM) modelinin avantajlarını kullanır.

Ermal ve arkadaşları [33] Life-Log adı verilen kişilerin sağlık durumlarının izlenmesi için bir anomali tespit yöntemi önerdiler. Bu yöntemde, kişisel sağlık bakımındaki anomalileri tespit etmek için akıllı sensörlerden veriler toplanır. Life-Log, akıllı sensör verilerini analiz etmek ve sınıflandırmak için LSTM kullanır. Lifelog, kişilerin sağlık durumlarını analiz ederek ve bilgilendirerek sağlık risklerini azaltmaktadır.

Tang, Z. ve arkadaşları tarafından geniş bir kablolu köprü için özel bir anomali algılama yöntemi önerilmiştir [9]. Burada, CNN kullanılarak anomali paternlerinin tanımı üzerinde çalışılmıştır. Yazarlar, anomali sınıflandırması için yarı-denetimli bir DL yaklaşımı uygulamışlardır ve denetimsiz sınıflandırma konusu için daha fazla araştırmaya ihtiyaç duyulmaktadır. Li, X. ve diğerleri, makalelerinde anomali tespiti sırasında veri artırma tekniklerinin kullanımını vurgulamışlardır. Bu, endüstrilerde kesin olarak etiketlenmiş verileri elde etmek zor olduğundan, küçük ana eğitim veri kümeleri için bile yüksek doğruluk elde edilmesi için yapılmıştır [23]. Bireysel takviyeli teknikleri, kombinasyon halinde kullanıldıklarından dolayı daha yüksek doğrulukla çalışmışlardır.

A. Graß ve diğerleri [34] tarafından yedi yıldan fazla bir süre boyunca kaydedilen eski algılayıcı verileri kullanılarak yeniden akıtma fırını (reflow oven) için anomalileri tespit etmek için üretim hatları (GADPL) için yeni bir genel anomali algılama yöntemi önerilmiştir. Bununla birlikte, bu çalışma, çok değişkenli verilerdeki anomalilerin ilişkisini incelememiştir ve anomalileri bulmak için farklı boyutlardaki bilgilerin belirli bir farklı ölçümde nasıl ilişkilendirileceği konusunda daha fazla iyileştirmeye ihtiyaç duymaktadır. Żabiński, T. ve diğerleri, makalelerinde, CNC işleme aletlerinin bir örneğini göstererek, yeni algılama ve sınıflandırma için akıllı durum izleme sistemleri ve bir araç seti kullanan platform önermektedir [35]. Bununla birlikte, CNC takım tutucu yanlış hizalama tespiti aynı zamanda araştırılmamıştır.

Liang, Y.C. ve arkadaşları tarafından daha doğru anomali tespiti elde etmek ve geçmiş verilere dayalı olarak daha fazla eşik değeri elde etmek için FFO algoritmasının kullanılması önerilmiştir [24]. Ancak daha iyi tahmin için titreşim, kuvvet vb. farklı veri kaynakları birleştirilmemiş ve DL algoritmaları araştırılmamıştır.

Önerilen yöntemde Chuan-Jun ve arkadaşları, sabit disk arızasının tam zamanını tahmin etmeye çalıştılar [36]. Mevcut veriler üzerinde ön-işleme sürecini ve ardından rastgele orman (random forest) yaklaşımını kullanarak modelleme işlemini

gerçekleştirmişlerdir. Siddhartha Sharma ve arkadaşları, demiryolu hattı üzerinde yaptıkları çalışmada, çalışmalarının ilk aşamasında hasarı tahmin etmek için rastgele orman yaklaşımını kullanmışlardır, ikinci aşamada Markov zincirini çizmişler ve son olarak Markov karar sürecini kullanmışlardır. Rayın, demir yolu onarımı ve bakımı için en optimum koşulları belirlenmiştir [37].

Nandini Chakravorti ve arkadaşları, makinelerin arıza süresini tahmin etmek için önce arıza kayıtlarını gruplandırmış, ardından metin madenciliği yaklaşımını kullanarak, belirtilen grupların her birini makine kılavuzunda kayıtlı arıza durumlarına göre etiketlemişlerdir [38]. Riccardo Accorsi ve arkadaşları, üretim sistemi performansı ve arıza olayları için tahminleri iyileştirmeye çalıştıkları bir makale yayınladılar [39]. Yöntemlerini gerçekleştirmek için yüksek hızlı paketleme makinesinden istatistiksel ve geçmiş verileri kullandılar. Bu yöntem, kümeleme (cihazları benzer davranışlara göre gruplama), birliktelik kuralları (arızalar ve makine davranışı arasındaki ilişkiyi inceleme) ve sınıflandırma yaklaşımı (arıza tahmini) olmak üzere üç aşamayı içermektedir.

Biswal, S. ve diğerleri [40], önemli rüzgar türbini bileşenlerinin arızasını tespit etmek için yapay sinir ağı kullanmıştır. İyi ve kötü koşullarda titreşim verilerini topladılar ve hatalı bileşeni sağlıklı bir bileşenle değiştirdiler. Son olarak, sağlıklı ve hatalı bir durumun özelliklerini sınıflandırmak için yapay sinir ağı tahminleri yapılmıştır. Bu araştırmanın sonuçları %6,92'lik bir sınıflandırma doğruluğu göstermektedir. Kolokas, N., ve diğerleri [41], çalışma döngülerini kullanarak endüstriyel ekipmandaki hataları gerçek zamanlı olarak tespit etmek için yapay sinir ağını diğer ML algoritmalarıyla karşılaştırdılar.

Büyük ölçekli veri işleme motoru Spark'ı kullanarak motorun mevcut durumunu tahmin etmek için bir tür tekrarlayan sinir ağı olan kısa süreli bellek ağları önerdiler. Bu durumda, veriler üç çalışma ayarından ve 21 derece sıcaklık, motor basıncı ve yakıt sensörü ölçümünden oluşur. Praveenkumar, T., ve diğerleri [42] otomobil şanzımanlarında arıza tespiti için bir destek vektör makine modeli önerdi. Bu çalışmada, dört dişli kutusu iki farklı hız ve yük koşulunda test edilmiş ve daha sonra bir destek SVM eğitmek için elde edilen titreşim sinyallerinden kaynaklar çıkarılmıştır. Deneysel sonuçlar, destek vektör makine modelinin dişli kutusu arızalarını %90'ın üzerinde bir doğrulukla sınıflandırabildiğini göstermiştir.

Bukhsh, Z.A. ve diğerleri [43], rastgele orman karar ağacı ve gradyan artırma ağacı gibi ağaç tabanlı ML tekniklerinin kaldırıcı bir sınıflandırmasını önermek için bir demiryolu kurumundan elde edilen mevcut verilerden yararlanan tahmin modelleri

geliştirerek bakım ve onarım ihtiyaçları, faaliyet türü ve demiryolu makaslarının durumuna bakmışlardır. Bakım ve onarım ihtiyaçlarını tahmin etmek için, eğim artırma ağacı modelinin diğer yöntemlere kıyasla %86 doğrulukla en iyi performansı gösterdiğini buldular. Furian, N., ve diğerleri [44] gelecekte belirli bir zaman diliminde bir taşlama (freze) makinesi arızasının olasılığını tahmin etmek için veriye dayalı bir yaklaşım sunmuş ve gerçek dünyadaki bir vaka çalışması üzerinde geçerliliğini kanıtlamıştır. Araştırmalarının açıklanan aşamaları, veri işleme, özellik çıkarma, makine öğrenimi ve makine arızası olasılığı tahminini gösterir. Tahminler, karma bir tahmin yaklaşımıyla oluşturulur. Önerilen birleşik tahmin yönteminin performansının genellikle tek bir yöntemden daha kararlı olduğunu bulmuşlardır.

Taşlama makinelerinde sunulan uygulama, bu makinelerin büyük arızalarının gelecekte 168 saate kadar tahmin edilebileceğini gösteriyor. Bu nedenle birçok reaktif bakım işlemi hassas bir şekilde planlanabilmekte ve yedek parça gibi kaynaklar doğru zamanda ve doğru yerde sağlanabilmektedir, bu da bakım ve onarım maliyetlerini optimize etmektedir. Baptista, M., ve diğerleri [14], sistemin gelecekte ne zaman arıza riski altında olduğunu tahmin edebilen ve ayrıca bakım ve onarım eylemlerinin ne zaman yapılması gerektiğini öneren bir çerçeve önermiştir. Bu tür bir tahmini kolaylaştırmak için, veriye dayalı tekniklerle birlikte bir Otomatik Regresif Hareketli Ortalama (Auto Regressive Moving Average, ARMA) modeli kullandılar ve ayrıca birkaç veriye dayalı tekniğin performansını karşılaştırdılar. Çalışmalarının yeniliği, ARMA yönteminin arıza olaylarını tahmin etmek için veriye dayalı tekniklerle entegrasyonudur. Çalışmaları, bir uçak motorunun gerçek bir endüstriyel örneği üzerinde rapor edilmiştir. Sonuçlar, destek vektör regresyon modelinin standart sapma, ortalama hata, ortalama mutlak hata ve hata yüzdesi gibi değerlendirme ölçüleri açısından yaşam kullanım modelinden daha iyi performans gösterebileceğini göstermiştir. Kalan veriye dayalı modeller daha düşük genel performansa sahiptir. Ayrıca önerilen yöntemlerin tek ve bağımsız yöntemlerden daha iyi performansa sahip olduğu görülmüştür.

Tahmine dayalı bakım ve anomali tespitinde kullanılan farklı makine öğrenimi modellerinin kısa bir açıklaması Tablo 2'de rapor edilmiştir.

Tablo 2. Tahmine dayalı bakım ve anomali algılamada makine öğrenimi modellerini kullanan inceleme makalelerinin özetidir

Yazarlar	Tahmin Edilen Parametre	Kullanım Alanı	Model Türü	Kaynak
Lee ve arkadaşları	Bozukluk vakası	Kesici alet ve mil	Destek vektör makinesi ve yapay sinir ağı	[45]
Su ve arkadaşları	Cihaz bozuklukları	Hard disk sürücüler	Yapay sinir ağı, destek vektör makinesi, rastgele orman	[36]
Alexander ve arkadaşları	Bozukluk	Akıllı bakım sistemleri Rexnord şirket desteği	Veri eksenli yaklaşımlar	[46]
Butte ve arkadaşları	Bakım operasyonlarının ve onarımların zamanını tahmin etmek	Yarı iletken	Rastgele ormanlar, DL, gradyan güçlendirme, genelleştirilmiş doğrusal model	[47]
Jung ve arkadaşları	Komşu panellerin güç eğrilerine dayalı olarak bir fotovoltaik panelin günlük elektrik güç eğrisini tahmin etme	Fotovoltaik panoları	Anlamsal ANN	[48]
Massimiliano ve arkadaşları	Bozukluk	Gerçek fabrika aletleri	ANN	[49]
Lasisi ve arkadaşları	-	Demir yolları	Rastgele ormanlar, destek vektör makinesi	[50]
Luo ve arkadaşları	Bozukluk	Bilgisayar sayısal kontrol makinesi	DL	[51]
Bernard ve arkadaşları	Bozulma zamanı	Gerçek bilgisayar bakımı için bilgisayar yönetim sistemi	ML yöntemleri	[52]
Aydın ve arkadaşları	Büyük ölçekli motor durumu tahmini	Motor	Kısa süreli bellek ağı	[53]
Canizo ve arkadaşları	Bozulma tahmini	Rüzgar türbünü	Rastgele ormanlar	[54]
Santos ve arkadaşları	Stator kısa devre hatası	Sincap kafesli asenkron motorlar	Rastgele ormanlar	[55]

Tablo 2'nin devamı

Yazarlar	Tahmin Edilen Parametre	Kullanım Alanı	Model Türü	Kaynak
Pan ve arkadaşları	Bozukluk	-	Anlamsal ANN	[56]
Edson ve arkadaşları	Bozukluk	Brezilya otomotiv endüstrisi	Öngörülü ve olasılıklı modeller, Bayes ağı	[57]
Durbhaka ve arkadaşları	Bozukluk	Rulman	K-en yakın komşu, destek vektör makinesi	[58]
Susto ve arkadaşları	Bozukluk	Yeri iletken üretimi	Denetleyici kümülatif özelliğin çıkarılması	[59]
Langone ve arkadaşları	Depolama süresi ve Onarımlar	Sızdırmazlık cihazları	Doğrusal olmayan otoregresif modelin çekirdek spektral kümelemesi	[60]
Prytz ve arkadaşları	Faydalı ömür tahmini	Ticari araçlarda hava kompresörleri	Makine öğrenimi teknikleri (rastgele ormanlar)	[61]
Rioux ve arkadaşları	Bozulma tahmini	Havalandırma sistemi	yapay sinir ağı, Doğrusal regresyon, karar ağacı	[62]
Machado ve arkadaşları	Bozukluk oranı	Elektrik sistemleri	Yapay sinir ağı, destek vektör makinesi	[63]
Biswal ve arkadaşları	Başlangıç bozuklukları	Rüzgar türbünü	ANN	[40]

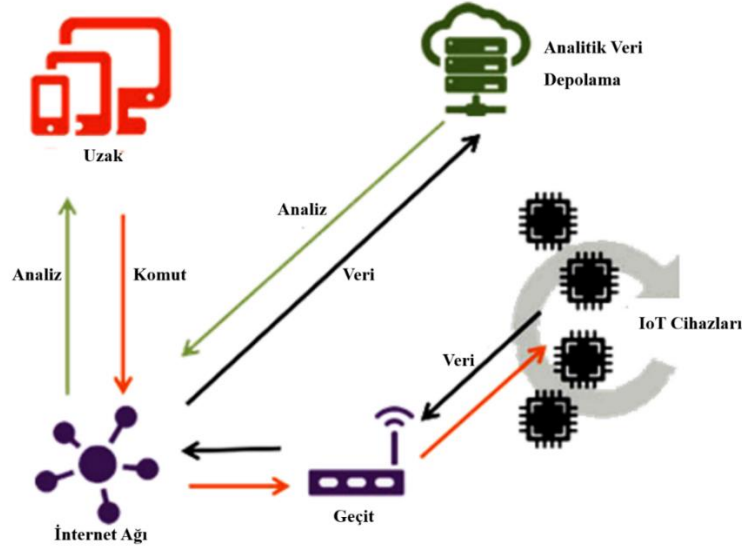
Bu bölümde geçmiş ilgili çalışmalar sunulmuştur. Bu çalışmalarda önerilen yöntemler çoklu sensörlü ortamlar için uygun değildir. Bu nedenle bu tezde çoklu sensörlü ortamlara uygun olan iki yöntem sunulmaktadır. İlk yöntem orta ve küçük veri hacmi olan IIoT'ler için uygundur. İkinci yöntem ise veri analizinde DS kullandığı için büyük veri hacmi olan IIoT'ler için uygundur.

1.3. Nesnelerin İnterneti

IoT, dünya çapında İnternet'e bağlı milyarlarca fiziksel cihazı ifade eder. Bilgi toplar ve kullanıcı ve diğer bağlı cihazlarla paylaşırlar. Daha basit bir ifadeyle, IoT, açık veya

kapalı olsun, İnternet'e bağlı herhangi bir cihazı içerir. Bu bağlı cihazlar, akıllı telefonlardan bina güvenlik sistemlerine ve hatta uçak motoru bakım sistemlerine kadar aklınıza gelebilecek her türlü cihazı kapsar. Hayvan vücutlarına gömülü kalp sağlığı izleme sistemleri veya biyokimyasal vericiler gibi tıbbi cihazlar, verileri bir ağ üzerinden iletebilir. IoT, İnternet'e bağlı geniş bir nesne ağını içerir. Cep telefonunuza bağlı bir kapı zili, IoT'nin en son uygulamalarına harika bir örnektir. Kapı zili çaldığında, kapının arkasındaki kişinin kim olduğunu her yerden bileceksiniz [64, 65].

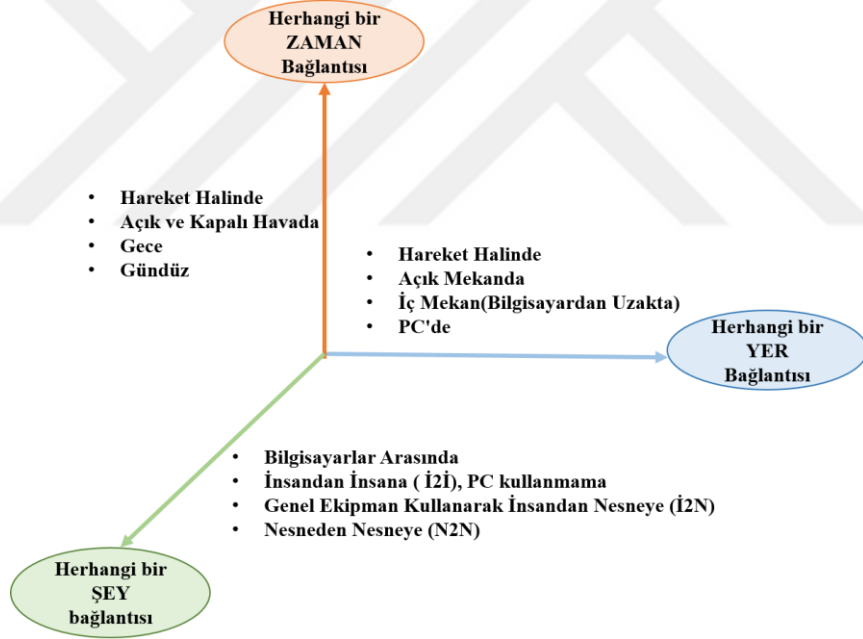
IoT ekosistemi, gömülü işlemciler, sensörler ve iletişim donanımı kullanarak ortamdaki alınan verileri toplayan, gönderen ve bu veriler üzerinde hareket eden, internete bağlı bir akıllı cihazlar topluluğudur. Sensörler tarafından toplanan veriler, bir IoT portuna veya uç cihaza bağlanarak analiz için buluta gönderilir veya yerel olarak analiz edilir. Bazen bu cihazlar diğer ilgili cihazlarla haberleşir ve birbirlerinden aldıkları bilgilere göre hareket ederler. Bu cihazlar, işin çoğunu insan müdahalesi olmadan yapar. Elbette insanlar cihazlarla etkileşime girebilir, örneğin, insanlar cihazları ayarlayabilir onlara talimat verip veya toplanan verilere erişebilirler. Bu web tabanlı cihazlarla kullandıkları bağlantı, ağ ve iletişim protokollerinin esas olarak belirli IoT uygulamalarına bağlı olmaktadır [65-67]. Şekil 1, IoT bileşenlerinin nasıl çalıştığını ve iletişim kurduğunu göstermektedir.



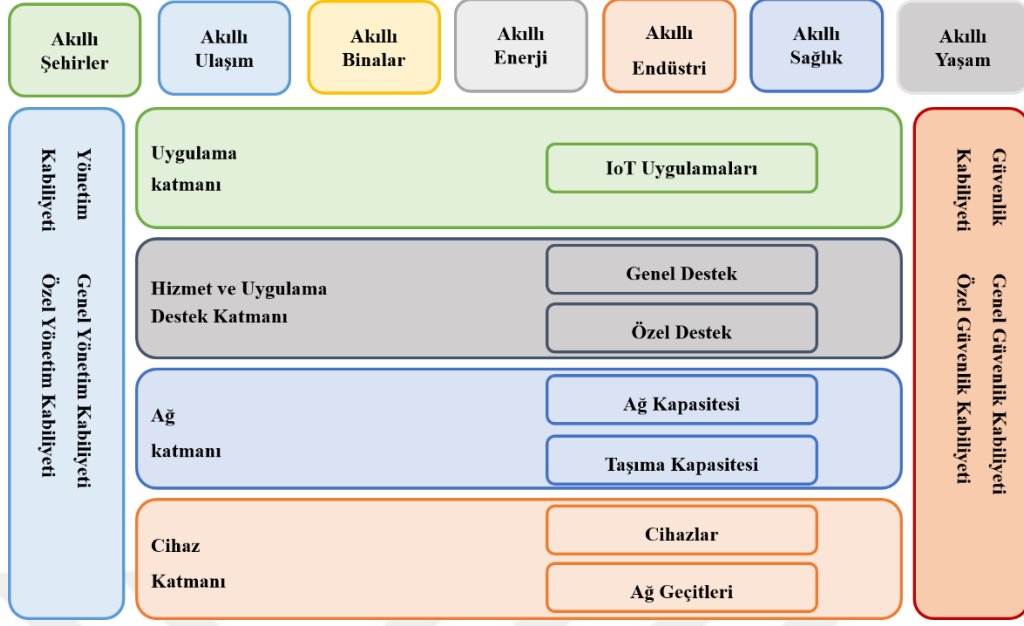
Şekil 1. IoT Ekosistemi [66]

Her yerde, her zaman ve her şeyde IoT'nin varlığı, endüstriyel ortamlar da dahil olmak üzere çeşitli sorunlara akıllı çözümler sunar [69]. Şekil 2, IoT bağlantısının boyutlarını göstermektedir [68].

Uluslararası Telekomünikasyon Birliği (ITU), Şekil 3'de gösterildiği gibi IoT'nin tasarlanması için dört katmanlı bir mimari önermiştir. Bu yapı, IoT uygulamalarının, akıllı şehir, akıllı bina, akıllı enerji, akıllı endüstri, akıllı sağlık ve akıllı yaşam gibi alanlarda kullanılmasını sağlar. Uygulama katmanı, IoT uygulamalarını içerir. Hizmet ve uygulama destek katmanı, veri işleme ve depolama gibi standart yetenekleri ve farklı uygulamalar için gereken özel nitelikleri içerir. Ağ katmanı, erişim ve taşıma kaynak kontrolü, hareketlilik yönetimi, kimlik doğrulama, yetkilendirme ve muhasebe gibi ağ bağlantısı kontrol işlevlerinden oluşur. Cihaz katmanında, cihazlar ve sensörler çevresel bilgileri toplar, iletişim ağına yükler ve verileri doğrudan iletişim ağından alır [2, 70].



Şekil 2. IoT bağlantısının boyutu [68]



Şekil 3. ITU tarafından IoT'nin Katmanlı mimarisi [70]

1.3.1. IoT nin Tarihçesi

IoT'nin ilginç bir başlangıç hikayesi var. Belki de IoT olarak adlandırılabilcek ilk örnek, 1980'lerde, Pensilvanya Pittsburgh'daki Carnegie Mellon Üniversitesi'nde, IoT adının verilmesinden on yıldan fazla bir süre önce kurulan Coca-Cola kutularını içeren bir otomatik makinedir. Bu üniversitenin öğrencilerinden biri olan David Nichols, bir içki almak için her seferinde ofisinden bu buzdolabına kadar uzun bir yol yürümek zorunda kalmaktan bıkmıştı. Nichols'un aklı uzun bir süre bu hikayeye meşguldü ki, aniden Stanford Üniversitesi'ndeki ilk bilgisayar kontrollü otomat makinesini hatırladı ve üniversitesinin Coca-Cola buzdolabı sorununu çözmek için aradığı teknolojinin zaten var olduğunu fark etti. Birkaç gün sonra, Nichols, bazı arkadaşlarıyla birlikte, otomat makinesine ARPANET (bugünkü İnternet'in öncüsü) aracılığıyla bağlanmak için bir sistem tasarladı. Bu sistem, buzdolabının durumunu uzaktan kontrol etmelerini sağlıyordu. Birçokları na göre bu sistem ilk gerçek IoT cihazıydı [71, 72].

Ancak 1999'da "Nesnelerin İnterneti" adı, MIT Üniversitesi'ndeki Auto-ID Labs araştırma grubunun kurucularından biri olan Kevin Ashton tarafından yapıldı. 1999 yılında yaptığı bir konuşmada bu yaklaşımı tanıttı: “Günümüzde bilgisayarlar ve ardından internet, insanların ihtiyaç duyduğu bilgilere ulaşmak için vazgeçilmez araçlardır. İnternetteki yaklaşık 50 petabayt (bir petabayt 1024 terabayta eşittir) bilgi yazma, ses kaydetme, fotoğraf

çekme veya barkod tarama gibi insan eylemleriyle ilgilidir. Asıl sorun, insanların az zaman, dikkat ve hassasiyete sahip olmasıdır. Bütün bunlar, gerçek dünyadaki nesnelerden bilgi alma konusunda pek iyi olmadıkları anlamına gelir. İnsan müdahalesi olmadan nesneler hakkında veri toplayabilen bilgisayarlarımız varsa, para harcamadan ve zaman kaybetmeden nesneler ve ihtiyaçlarımız hakkında çok fazla bilgi alabiliriz. Bu durumda etrafımızdaki eşyaların ne zaman değiştirilmesi gerektiğini, ne zaman tamir edilmesi gerektiğini ve son kullanma tarihlerinin bizim için ne zaman bittiğini kolaylıkla bileceğiz". Kevin Ashton, radyo frekansı tanımlamasının (RFID) IoT için bir ön koşul olduğuna inanıyordu. Tüm nesneler etiketlenirse bilgisayarların onları yönetip izleyebilmesi mümkün olabilecektir [73, 74].

2000'lerin sonundan 2010'ların başlarına kadar, dünyanın dört bir yanındaki şirketler, şimdi AI ve makine öğrenimi konusunda olduğu kadar IoT konusunda da heyecanlıydılar. 2009 yılında IBM, ekonomik büyüme ve sürdürülebilir kalkınma hedefiyle Smarter Planet kampanyasını başlattı. Dünyanın en saygın yönetim danışmanlığı firması olan McKinsey, IoT teknolojisinin durumu hakkında raporlar yazmaya başladı. Ağ ekipmanı alanındaki en büyük şirket olan Cisco, 2011'de IoT'nin 2008 ve 2009 yılları arasında "doğduğunu" duyurdu; İnternete bağlı cihazların sayısı, gezegende yaşayan insan sayısını aştığında. Aynı yıl, bir Amerikan araştırma şirketi olan Gartner, bu yeni fenomeni ilk kez gelişen teknolojiler listesine ekledi [75-77].

Kısa bir süre sonra, IoT girişimleri her köşeden ortaya çıktı. 2010 yılında, Nest Labs girişimi, öğrenme yeteneklerine sahip sensörler ve Wi-Fi ile donatılmış akıllı termostatlar ve duman dedektörleri üretmeye başladı. 2014 yılında Google bu girişimi 3,2 milyar dolara satın aldı ve bu, IoT'nin genel halkın dikkatini çektiği an olduğu söylenebilir. Akıllı ses asistanları Alexa ve Google Home'un ortaya çıkmasıyla birlikte IoT cihazlarının insanların evlerinde bulunması her geçen gün daha yaygın hale geldi. Tablo 3, IoT tarihinin bir özetini göstermektedir.

Tablo 3. Zamanla IoT

Etkinlik	Date
David Nicholls, ilk IoT cihazını geliştirdi: Coca-Cola Otomat Makinesi [72].	1980
Kevin Ashton, bilgisayarın gözlerini ve kulaklarını tanımlamak için "Nesnelerin İnterneti" terimini kullandı [78].	1999
LG, ilk bağlantılı buzdolabını tanıttı [78].	2000
Dünyanın ilk IoT konferansı İsviçre'nin Zürih kentinde yapıldı [77].	2008

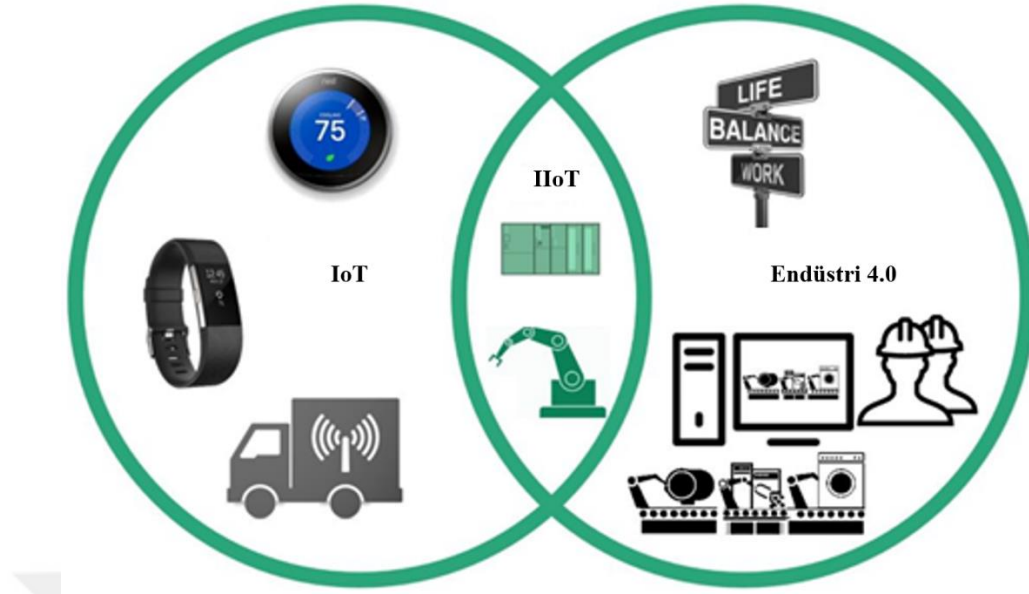
Tablo 3'ün devamı

Etkinlik	Date
Tony Fadel, akıllı bir termostat şirketi olan Nest'i kurdu [79].	2010
Oxford Dictionaries, "Nesnelerin İnterneti" terimini ekledi [80].	2013
Mirai botnet, 600.000'den fazla IoT cihazına kötü amaçlı yazılım bulaştırıyor [81].	2016
İnternete bağlı cihaz sayısı 20 milyar cihazı aşıyor [82].	2020
Bağlı cihazların sayısı yaklaşık 50 milyara ulaşacak [76].	2021
Cisco'ya göre 2030 yılına kadar IoT'ye 500 milyardan fazla cihaz bağlanacak [75].	2030

1.3.2. Endüstriyel Nesnelerin İnterneti

IIoT, IoT büyümesinin en önemli ve yaygın olarak kullanılan alanlarından biridir. IoT teknolojisinin endüstriyel alanda akıllı bir endüstriyel ağ olarak kullanılmasına IIoT denir. Endüstriyel birimlerde IIoT kullanılarak tüm nesneler bağlanabilir ve tüm bilgi alışverişi, kontrol ve izleme görevlerini yerine getirmek için entegre bir ağ oluşturulabilir. Bu teknoloji, endüstriyel otomasyonun geleceğini büyük ölçüde etkileyecek dört büyük teknoloji arasında yer alıyor [83].

Endüstri 4.0 olarak adlandırılan dördüncü sanayi devrimi [3] bağlamında IIoT, büyük veri analitiği yardımıyla fiziksel sistemlerin ve üretim süreçlerinin dönüştürülmesine yardımcı oluyor. Sensörlerden ve diğer bilgi kaynaklarından gelen gerçek zamanlı veriler, endüstriyel cihazların ve altyapıların kararlar almasına ve belirli öngörüler ve eylemler kazanmasına yardımcı olur. Makineler, önceki sanayi devrimlerinin gerçekleştiremediği görevleri de gerçekleştirebilir ve otomatikleştirebilir. Daha geniş bir bağlamda, şehirlerin nasıl akıllı şehirler haline geldiği ve fabrikaların nasıl akıllı fabrikalar haline geldiği gibi ekosistemler veya bağlantılı ortamlarla ilgili kullanım örnekleri için kritik öneme sahiptir [84]. Şekil 4. IoT, IIoT ve endüstri 4.0 arasında birbirleri ile olan bağlantısını göstermektedir.

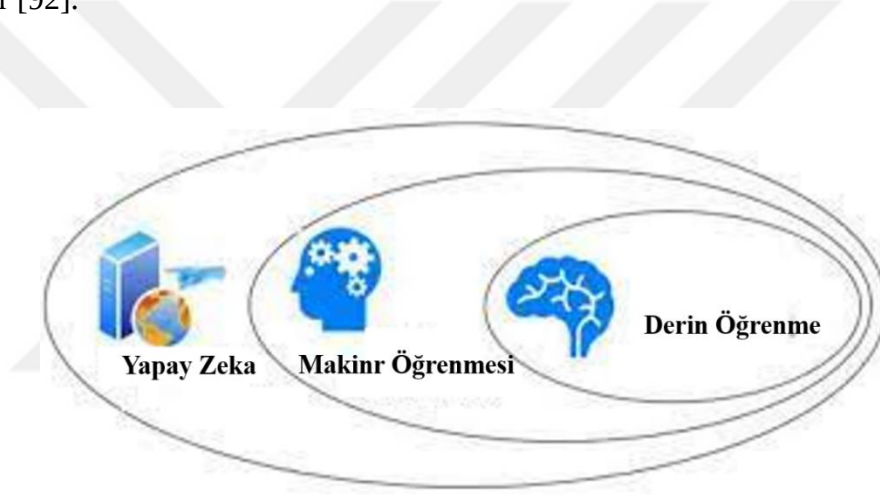


Şekil 4. IoT ve IIoT ve Endüstri 4.0 [85]

Akıllı cihazlar ve makineler arasında verilerin sürekli toplanması ve aktarılması, endüstriler ve şirketler için birçok büyüme fırsatı sunar. Veriler, endüstrilerin ve şirketlerin hataları veya verimsizlikleri tespit etmesine ve bunları hemen ele almasına olanak tanır. Bağlı ve akıllı cihazlarla işletmeler daha büyük miktarda veriyi daha hızlı toplayabilir ve analiz edebilir. Ölçeklenebilirliği ve performansı artırır ve üretim katları ile genel ofisler arasındaki boşluğu kapatır. IIoT entegrasyonu, endüstriyel birimlere ilerleyen operasyonları hakkında daha doğru bir görünüm verebilir ve bilinçli iş kararları vermelerine yardımcı olabilir [4, 86].

Makineden makineye (Machine-to-Machine, M2M) iletişim, büyük veri ve makine öğrenimine güçlü bir şekilde odaklanan IIoT, endüstrilerin ve şirketlerin operasyonlarında daha iyi verimlilik ve güvenilirliğe sahip olmalarını sağlar. IIoT, robotik, tıbbi cihazlar ve yazılım üretim süreçleri dahil olmak üzere endüstriyel uygulamaları içerir. Şekil 5 IIoT'inin genel yapısını göstermektedir.

Başından beri bilgisayarlar, insan yaşamının çeşitli problemlerini çözmek için bilgi işlem makineleri olarak tasarlanmıştır. Makine öğrenimi, problem çözme yaklaşımında bir atılım olarak görülüyor. Geleneksel programlama yönteminde bilgisayar programcısı, sonuca ulaşmak için tüm girdileri, koşulları, çözümler ve mümkün olduğunca gerekli adımlar, öngörerek programı tasarlar. Makine öğrenimi bu yaklaşımın tam tersidir. Bu paradigmada, programcı yalnızca makinede programın kalitesini denemek ve değerlendirmek için bir platform (öğrenme algoritması) uygular. Bu vesile ile makine, yeniden programlamaya gerek kalmadan yeni deneyimler alarak görevinin kalitesini artırır ve işlemleri gerçekleştirir. AI hem programlama hem de makine öğrenimi yaklaşımlarında elde edilebilse de, son yıllarda bu alandaki ilerlemeler kesinlikle makine öğrenimine çok şey borçludur [92].

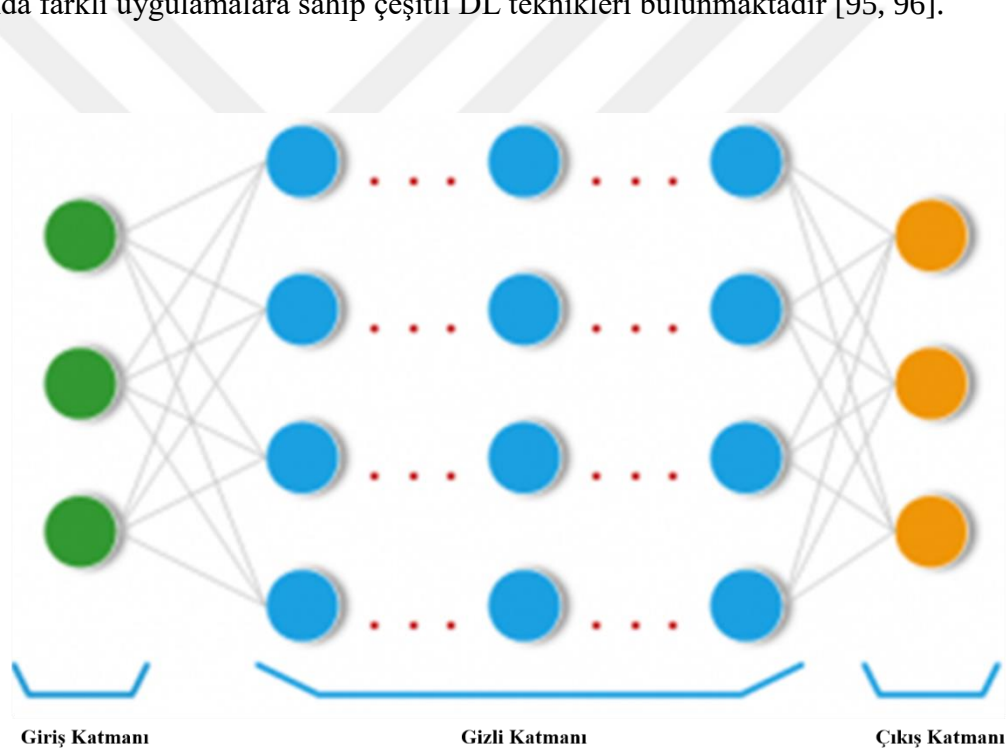


Şekil 6. AI, ML ve DL [93]

Makine öğrenimi, kendi kendine öğrenme kabiliyeti olan programlarının geliştirilmesine odaklanan bilgisayar bilimi ve AI in bir alt alanıdır. Bu programları geliştirmesinde, veri kalıplarını bulmak için örüntü tanıma ve hesaplamalı öğrenme teorisi kullanılır. Makine öğrenimi ve veri madenciliğinin, her ikisinin de verilerden kalıp bulmaya odaklanması anlamında örtüştüğü söylenebilir. Makine öğrenimi, performansı artırmak ve karar vermek için veri kalıplarını kullanırken, veri madenciliği, verileri insan tarafından anlaşılabilir bir forma dönüştürür [94]. Şekil 6 AI, ML ve DL arasında bağlantıyı göstermektedir.

1.4.1. Derin Öğrenme

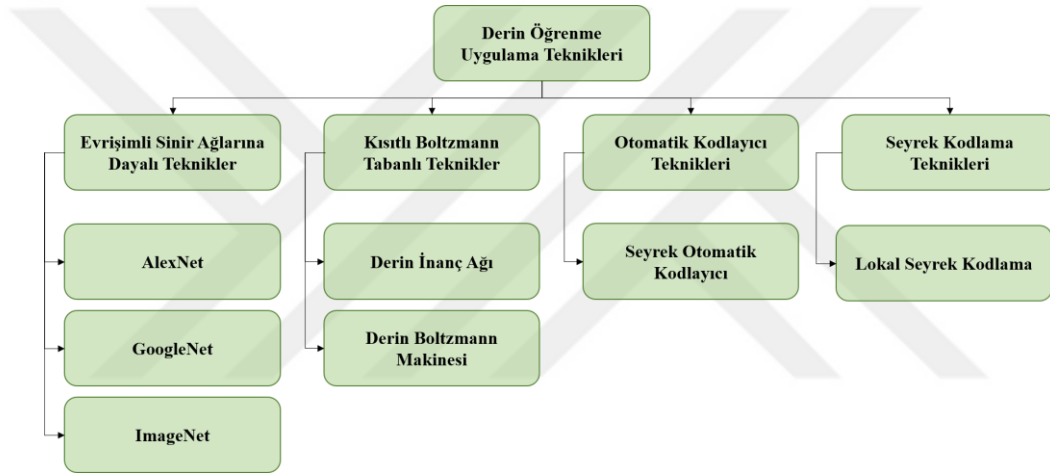
Geçmiş yıllarda derin olmayan yapıya sahip ML ve sinyal işleme teknikleri kullanılmıştır. Bu mimariler tipik olarak tek katmanlı doğrusal olmayan özellik dönüştürücü içerir ve çok katmanlı bir yapıya sahip değildir. Bu tür mimari basit ve yüzeysel problemlerin çözümünde etkilidir, ancak karmaşık problemler ve gerçek uygulamalar modellenirken bu teknikler sınırlamalarla karşılaşmaktadır. Örneğin, ses, görüntülerin ve doğal sahnelerin işlenmesi için yüzeysel mimarilerin sınırlamaları vardır. Karmaşık işleme, insanla ilgili bilgi işleme karmaşıklıklarını çıkarmak ve zengin bir temsil sağlamak için daha derin bir mimari gerektirmektedir. Bu nedenle karmaşık yapıları işlemek için DL yöntemleri oluşturulmuştur. Şu anda farklı uygulamalara sahip çeşitli DL teknikleri bulunmaktadır [95, 96].



Şekil 7. Derin öğrenme

Makine öğrenimi için çeşitli teknikler vardır ve DL bunlardan biridir. Evrişimli sinir ağı kavramı, DL alanında çok önemlidir. Birçok araştırmada, DL, evrişimli sinir ağı ile eşdeğer kabul edilirken, evrişimli sinir ağı, öğrenmenin hiyerarşik olarak yapıldığı, hiyerarşinin her adımında ve her katmanında, öğrenmenin bir parçasının yapıldığı sadece bir DL yaklaşımıdır. DL ile ilgili ilginç bir gerçek, tasarımının biyolojik sistemlerden ilham alması ve hiyerarşik bir model kullanmanın nedeninin aslında insan beyninin yapısını simüle

eden bir yapı elde etmektir [8]. Makine öğreniminin bir alt dalı olan sinir ağları ile birleştirilmiştir. Bu teknikler genellikle ANN'ne dayanır ve verilerdeki üst düzey soyutlamaları modellemeye çalışır. Bu modeller, veri temsili için bir yöntem geliştirmek için kullanışlıdır. DL ve diğer öğrenme için sinir ağı mimarisi tasarımının aslında doğruluk, genelleme ve hesaplama maliyeti arasında bir uzlaşmadır. Örneğin, öğrenme yüz tanıma için ise, ilk katmanlarda köşeler, çizgiler ve kenarların hiyerarşisi öğretilir. Şekil 7 kısaca DL sürecini göstermektedir. Unutulmamalıdır ki, yüz özelliklerinin her birini öğrenme için birer filtre olarak düşünülebilir [95, 97]. Bu bölümün geri kalanında, DL hakkında daha fazla ayrıntı verilecektir.



Şekil 8. DL uygulama teknikleri

Çeşitli çalışmalarda, DL'nin kullanıldığı alana bağlı çeşitli yöntemler önerilmiştir. Şekil 8, DL uygulama tekniklerinin dört grupta sınıflandırılmasını göstermektedir: evrişimli sinir ağı, Kısıtlanmış Boltzmann makinesi, otomatik kodlayıcı ve seyrek kodlama. Unutulmamalıdır ki, her bir tekniğe ek olarak, Şekil 8'de her birinin uygulanan örneğinden bahsedilmiştir. Evrişimli sinir ağları, hiyerarşik eğitimi nedeni ile en önemli ve çok güçlü DL yöntemlerinden biridir. Bu öğrenme türü, en yaygın kullanılan ve en ünlü DL yöntemlerinden biridir [97].

Kısıtlı bir Boltzmann makinesi aslında 1986'da tanıtılan ve DL'yi uygulamak için kullanılan üretken bir stokastik sinir ağıdır. Bu tür sinir ağı, görünür ve gizli katmanların, çift yönlü bir çizge oluşturmasını gerektiren bir sınırlamaya sahiptir. Bu sınırlama, daha optimal eğitim algoritmaları oluşturur [97].

Otomatik kodlayıcı tekniği, başka bir DL uygulama tekniğidir. Bu teknik aynı zamanda yapay sinir ağını da kullanır, ancak ağı eğitmek ve değerleri tahmin etmek yerine, girdiyi yeniden yapılandırmak için ağı eğitir. Bu nedenle çıkış vektörleri, giriş vektörü ile aynı boyutlara sahip olacaktır. Otomatik kodlayıcıya dayalı yöntemlerde öğrenme, derin inanç ağına benzer ve hatanın en aza indirilmesi için öğrenme beş aşamada yapılır [98].

Son yöntemde seyrek kodlamadır. Bu modelde amaç, ham verilerden seyrek özelliklerin çıkarmaktır. Seyrek gösterim, veri yorumlamayı basitleştirir [97]. Tablo 4, mevcut yöntemlerin karşıladıkları özelliklere göre birbirleri ile karşılaştırılmasını göstermektedir.

DL teknikleri için başka sınıflandırmalar da vardır; bu teknikler denetimsiz ve denetimli olarak sınıflandırılabilir [99].

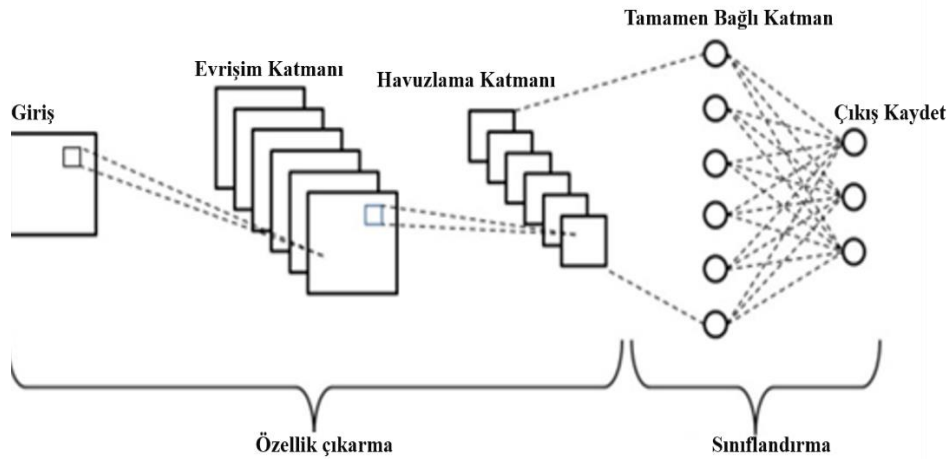
Denetimli öğrenmede denetçi, incelenen verileri açıkça etiketlemelidir. Denetimli öğrenme, girdi ve çıktının bilindiği bir tür ML dir. Bir denetimci, algoritmaya bilgi sağlar; böylece sistem girdiden çıktıya bir fonksiyon öğrenmeye çalışır. Denetimli öğrenme yöntemlerinin iki ana türü sınıflandırma ve regresyondur. Sınıflandırmada, öğrenme çıktısı sınırlı bir sınıflar kümesidir. Sınıflandırma, yalnızca iki tür dahil olmak üzere ikili, üçlü veya daha fazla sınıf içeren çok sınıflı olabilir. Sınıflandırmanın aksine, regresyon öğrenmesinin çıktısı bir veya daha fazla sürekli sayıdır. Regresyon analizi, bir kümenin her bir ögesinin ait olma olasılığının olduğu çoklu etiket sınıflandırmasına eşdeğer puanlanmış etiketler sağlamak için uygun bir mekanizmadır [100].

Denetimli öğrenmenin aksine, denetimsiz öğrenmenin etiketlenmiş verisi yoktur, amaç girdi ve çıktıyı iletmek değildir. Yine de, yalnızca bilginin Kümelenmesi önemlidir ve verilerde belirli bir yapı bulmak gerekir. Boyut azaltma, bir kümeleme uygulamasıdır. Çeşitli bileşen analizleri ve ayırım formları gibi farklı şekillerde yapılabilir. Örneğin, otomatik kodlayıcılar, veri sıkıştırmak veya bellek alanı azaltmak için girdi verilerini azaltılmış veya kodlanmış bir çıktıya dönüştürebilir. Kümeleme, küme analizi için bir DL uygulama yaklaşımıdır. Genellikle, dönüşümlü olarak küme merkezleri ve küme üyeleri seçilerek gerçekleştiriliyor [100].

Bu bölümde, avantajları, zorlukları ve uygulama alanındaki DL ile ilgili ek konular tartışılmaktadır. DL'nin bazı avantajları ve zorlukları Tablo 5'de gösterilmektedir. DL için Torch, TensorFlow, Theano ve Caffe gibi farklı uygulama araçları kullanılmaktadır [97, 99].

1.4.1.1. Evrişimli Sinir Ağları

Evrişimli sinir ağlarında, insan görsel sisteminden esinlenilmiştir. CNN, esas olarak görüntüler üzerinde çalışmak için kullanılan çok katmanlı bir derin sinir ağı türüdür. CNN'lerin görüntü ve video tanıma, öneri sistemleri, görüntü sınıflandırma, tıbbi görüntü analizi ve doğal dil işleme alanlarında geniş uygulamaları vardır. Geleneksel sınıflandırma algoritmaları ile karşılaştırıldığında, evrişimli sinir ağlarında daha az ön işlemeye ihtiyaç duyulur [101]. Bu ağlar, önceki yaklaşımlarda manuel olarak öğrenilen kriterleri öğrenir. CNN'ler, özellik çıkarma ve sınıflandırmaları bir araya getirdikleri için diğer görüntü tanıma algoritmalarından çok farklıdır. Şekil 9, CNN mimarisinin basit bir diyagramını göstermektedir. Basit bir ağ mimarisi beş farklı katmandan oluşur. Giriş katmanı, evrişim katmanı, kümülatif katman, tam bağlantılı katman ve çıkış katmanı. Öznitelik çıkarma ve sınıflandırma olmak üzere iki genel kısma ayrılır.



Şekil 9. Temel evrişimli sinir ağlarının genel şeması [105]

Giriş katmanı, giriş görüntüleri için sabit bir boyut belirtir. Bir sonraki seviyede, çeşitli çekirdekler/filtreler kullanarak görüntüden yeni özellikler çıkarabilen evrişim katmanı vardır. Bir sonraki adımda, havuz katmanı, mevcut bilgileri korurken görüntü boyutunu küçültür. Bu iki katman, özellik çıkarma bölümünü oluşturur. Daha sonra çıkarılan öznitelikler için orantısal ağırlık belirlenir ve CNN ağlarındaki sınıflandırma bölümüne eşdeğer olan tam bağlantılı katmanda birleştirilir. Son olarak, çıktı katmanında her nesne kategorisi için bir çıktı nöronu vardır.

Tablo 4. DL uygulama tekniklerinin karşılaştırılması [97]

Özellikler	evrişimli sinir ağları	Kısıtlanmış Boltzmann	otomatik kodlayıcı	seyrek kodlama
Genelleme yapabilme	Evet	Evet	Evet	Evet
denetimsiz öğrenme	Hayır	Evet	Evet	Evet
Özellik öğrenme	Evet	Evet	Evet	Hayır
Gerçek zamanlı eğitim	Hayır	Hayır	Evet	Evet
Küçük veri kümeleriyle eğitim	Evet	Evet	Evet	Evet

Tablo 5. DL ile ilgili avantajlar ve zorluklar

DL'nin avantajları	Yüzeysel öğrenme ile mümkün olmayan karmaşık uygulamaları mümkün kılar [102].
	Yüksek hesaplama gücüne sahiptir [98].
	Derin mimari, çok büyük boyutlarda gösterilmesi gereken durumlarda çok uygundur [103].
	Derin mimari, özellik çıkarımı için çok uygundur [97].
	DL, sensörden veri alınmasından çıktının üretilmesine kadar örüntü tanımanın tüm aşamalarını içerir [104].
	Görsel veriler için derin özelliklerin kullanılması önemli bir avantajdır [104].
	DL'nin en önemli avantajlarından biri, yerel değişikliklerden etkilenmeyen veri gösteriminin soyutlanmasıdır [103].
	DL, büyük verilerle çalışmanın zorluklarını büyük ölçüde çözebilir [103].
DL'nin zorlukları	DL ve ilgili yöntemlerin teorik kavramı hala iyi anlaşılmamıştır ve hatta mevcut mimariler arasında uygun bir karşılaştırma yoktur [97].
	Durağan olmayan verilerde artımlı öğrenme, DL için bir zorluktur [103].
	Eğitim verileri, derin öğrenme için bir zorluktur [97].
	Derin öğrenmede süreçlerin zaman karmaşıklığı zorlu bir iştir [104].

CNN ağının en önemli kısmı, hesaplamaların çoğunu gerçekleştiren evrişim katmanıdır. Evrişim katmanı, işlevi bu filtreler ile özellik haritaları oluşturmak için giriş katmanı arasındaki evrişim işlemi olan bir dizi filtreden oluşur. Havuzlama katmanının amacı, veri işleme için gereken bilgi işlem gücünü azaltmak için evrişim katmanından çıkan özellik alanının boyutunu küçültmektir. Bu katman, toplanan bilgileri basitleştirir ve aynı bilgilerin sıkıştırılmış bir versiyonunu oluşturur. Bu çalışma iki şekilde yapılır: maksimum ve ortalama. İlk yöntemde görüntünün çekirdeğin kapsadığı kısımdan maksimum değeri, ikinci yöntemde ise çekirdeğin kapsadığı kısımdan tüm değerlerin ortalamasını döndürür. CNN'nin son katmanları tamamen bağlantılıdır. Bu katmanların ana görevleri, bir dizi

evrişimsel katman ve havuz katmanı tarafından tespit edilen ve çıkarılan özellikleri sınıflandırmaktır. Özellik haritaları, tamamen bağlı katmanlara girmek için tek boyutlu bir vektöre dönüştürülmelidir [100, 106].

1.4.1.2. Optimizasyon Yöntemleri

Bir CNN'yi eğitmek, evrişimli katmanlarda uygun çekirdekler ve tam bağlantılı katmanlarda ağırlıklar bulmaya, eğitim veri kümesindeki çıktı tahminleri ve Gerçek Sınıflar arasındaki farklılıkları azaltmaya bağlıdır. Geri yayılım, eğitim durumunda çekirdekleri ve ağırlıkları güncellemek için kayıp işlevini kullanan bir yöntemdir [107].

DL'de kayıp kavramı, mevcut yinelemede modelin zayıf performans derecesini belirler. Bu konsept, ağı daha iyi çalışması doğrultusunda eğitmek için kullanılır. Genellikle, öğrenmede amaç, kaybı en aza indirmektir, çünkü düşük kayıp oranı daha iyi model performansını göstermektedir. Matematiksel olarak, minimize etme (veya maksimize etme) prosedürüne optimizasyon denilir. Aslında optimize ediciler, kayıpları azaltmak için ağırlık ve öğrenme oranı gibi DL tekniklerinin parametrelerini değiştirmek için kullanılan algoritmalar veya yöntemlerdir. Ayrıca verimlilik elde etmek ve model hatasını en aza indirmek için kullanılırlar [108]. En sık kullanılan optimizasyon algoritmalarından biri Adaptive Moment Estimation (Adam) algoritmasıdır. Son zamanlarda, bu algoritma araştırmacılar tarafından en fazla kullanılan algoritmalar arasında yer almıştır [109, 110]. Adam algoritmasını optimizasyon problemlerinde kullanmanın avantajları aşağıda listelenmiştir:

- Bu algoritma kolayca uygulanır.
- Adam algoritması, hesaplama açısından optimaldir.
- Adam algoritması, çok az belleğe ihtiyaç duyar.
- Adam, veri veya parametreler açısından büyük kabul edilen problemler için çok uygundur.
- Adam algoritması, durağan olmayan hedefler için uygundur.
- Adam, çok gürültülü veya seyrek eğimli problemler için iyi bir algoritmadır.

1.5. Anomali Algılama

Anomali tespiti, bir veri setinde beklendiği gibi normal şekilde davranmayan kalıpları bulmaktır. Bu beklenmedik davranışlara anomaliler veya aykırı değerler denir. Anomali, önceden bilinmeyen ve beklenmedik bir davranış olarak kabul edilebilir. Anomaliler zararlı veya zararsız olabilirler. Her iki durumda da anomalinin tespit edilmesi gerekir. Anomali tespiti, çeşitli uygulamalar hakkında hayati bilgiler sağlar [6].

Anomali tespit yöntemleri, konunun önemi nedeniyle son yıllarda dikkatle incelenmiş ve farklı uygulamalar için teknikler geliştirilmiştir. Veri madenciliği boyutundan anomali tespit yöntemleri denetimli, denetimsiz ve yarı denetimli olmak üzere üç kategoride sınıflandırılmaktadır [6, 111].

Anomaliler, veri kümesindeki beklenen veya normal davranıştan farklı olan noktalar veya kalıplardır. Bir anomalinin nedeni ve tezahürü verilere ve uygulamalara göre değişir. Anomaliler aşağıdaki gibi üç farklı gruba ayrılır [112].

Nokta anomalileri: Tek bir veri örneğinin tüm verilere göre anormal olması durumunda, bu örneğin nokta anomalisine sahip olduğu söylenir. Bu en basit anomali türüdür ve anomali tespiti alanındaki araştırmaların çoğu buna odaklanmıştır [112].

Bağlamsal Anomaliler: Bağlamsal bir anomali, belirli bir bağlamda anormal olarak kabul edilebilir ama başkalarında bu şekilde olmayabilir. Örnek olarak zaman serilerinde, zaman içinde birbirine yakın noktaların benzer şekilde davranması beklenir [112].

Toplu anomali: Bir dizi veri örneği, tüm veri kümesine kıyasla anormal ise, bu kümeye toplu anomali denir. Bu veri örnekleri kendi başlarına anormal olmayabilir, ancak bunların bir set olarak birlikte ortaya çıkmaları anormal olarak kabul edilir [112].

1.5.1. Anomali Algılama Yöntemleri

Bu bölümde, anomalileri tespit etmeye yönelik veri madenciliği yöntemleri ve bunların zayıf ile güçlü yönleri dikkatle incelenmektedir. Bu alandaki araştırmaların çoğu veri madenciliği yöntemlerini kullanmıştır. Veri madenciliği yöntemleri veri keşfi için uygun araçlar sağlar. Anomali tespiti için veri madenciliği yöntemleri aşağıdaki bölümlere ayrılabilir:

1.5.1.1. Sınıflandırmaya Dayalı Yöntemler

Sınıflandırmaya dayalı yöntemler, bir kaydı doğrudan normal veya anormal kategorilere ayırmaya çalışır. Sınıflandırıcı örnekleri arasında ANN, DL, ileri beslemeli algılayıcı ağları, Bayes ağları, destek vektör makineleri ve kural tabanlı sınıflandırıcılar bulunur. Sınıflandırıcılar denetimli tiptedir ve çoğu sınıflandırıcı, herhangi bir anomali olmaksızın tüm olası normal davranışları sunan bir veri kümesi üzerinde eğitilir. Yeni kayıtlar normal davranış sınıflarından birine girmiyorsa, anomali olarak sınıflandırılırlar. Bir sınıflandırıcı, bir özellik kaydını normal veya anormal olarak doğru ve doğrudan sınıflandırabilen bir sınıflandırıcı oluşturmak için hem normal hem de anormal kayıtları içeren bir veri kümesi kullanılarak da eğitilebilir. Bu yöntemlerin ana dezavantajı, kural tabanlı yöntem dışında hepsinin büyük etiketli veri kümeleri gerektirmesidir. Bu etiketli veri kümelerinin genellikle elle oluşturulması gerektiğinden, bu tür veri kümelerini oluşturmak zor olacaktır [6, 113]. Tablo 6 bu yöntemin avantaj ve dezavantajların göstermektedir.

Tablo 6. Sınıflandırmaya göre yöntemin avantajlar ve dezavantajları ve güçlü yönleri

Avantajlar	Dezavantajlar
• Hızlı test aşaması • Gerçek zamanlı olarak kullanılabilir • Çok güçlü algoritmalar	• Etiketlenmiş veri ihtiyacı • Hesaplamalı olarak ağır hazırlık adımı • Anomali tespiti ikincil bir hedeftir

1.5.1.2. En Yakın Komşuya Dayalı Yöntemler

En yakın komşu analizine dayalı anomali tespit yöntemleri, normal kayıtların yoğun komşularda gerçekleştiğini, anomalilerin ise komşularından uzakta gerçekleştiğini varsayar. Bu yöntemin temel bir zorluğu, uygun mesafe boyutunu belirlemektir. Bu analiz genellikle, en yakın k komşuya olan mesafenin veya test kaydından d gibi belirli bir mesafedeki kayıtların sayısının analiz edilmesini içerir. Bu genellikle hesaplama açısından ağır bir iştir ve performansı mesafe boyutuna bağlıdır. Özellikle veriler büyük veya karmaşık olduğunda, mesafe boyutunu tanımlamak da zordur [6, 114]. Tablo 7, bu yöntemin avantaj ve dezavantajlarını göstermektedir.

Tablo 7. En yakın komşuya göre yöntemin avantajlar ve dezavantajları

Avantajlar	Dezavantajlar
• Denetimsiz • Bilgi odaklı (puan dağılımına ilişkin herhangi bir varsayım olmaksızın) • Kolay uygulama	• Noktalar arasındaki mesafeyi belirlemek zor olabilir. • Hesaplama işlemleri için ağır hazırlık adımı • Anomali barındırmayan bir veri seti ihtiyacı

1.5.1.3. Kümelemeye Dayalı Yöntemler

Kümeleme, benzer veri örneklerini kümeler halinde gruplandırmak için kullanılır. Kümeleme denetimsiz bir yöntemdir, ancak yarı denetimli kümeleme de kullanılmıştır. Kümeleme ve anomali tespiti temelde farklı olmasına rağmen, birkaç kümeleme tabanlı anomali tespit yöntemi geliştirilmiştir. Tablo 8, bu yöntemin avantaj ve dezavantajları göstermektedir [115].

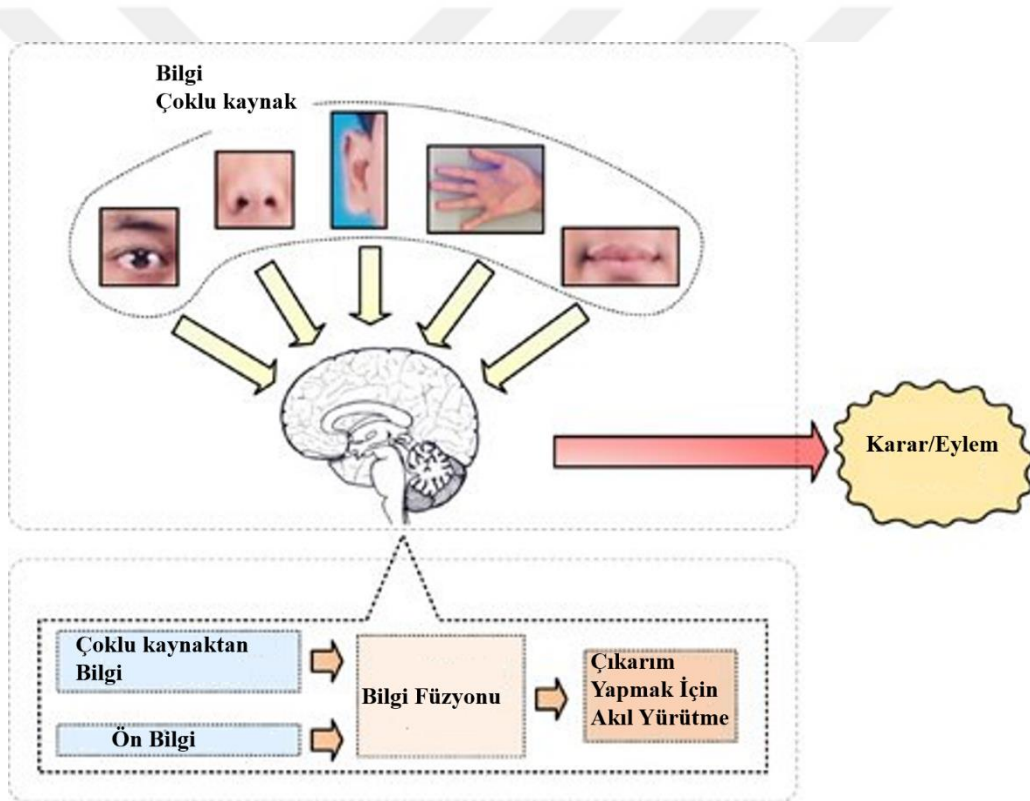
Tablo 8. Kümelemeye dayalı yöntemin avantajlar ve dezavantajları

Avantajlar	Dezavantajlar
• Denetimsiz • Hızlı test • Gerçek zamanlı olarak kullanılabilir	• Anomali tespiti için değil, kümeleri bulmak için optimize edilmiştir. • Hazırlanması hesaplama açısından ağırdır. • Önemli bir anomali kümesi algılanmadan kalabilir.

1.6. Bilgi Füzyonu

AI in son yıllarda kaydettiği önemli ilerlemeyle birlikte, çok sensörlü veri füzyonu konusu gündeme gelmiştir. Çok sensörlü veri füzyonu kavramı ve fikri, teori alanında çok yeni değildir ve tarihi 1960'lı yıllara ve hatta ondan öncesine dayanmaktadır. Diğer AI yöntemleri gibi, bu fikir de insan beyninin işleyişini ilham almaktadır. Beyin, gözler ve kulaklar gibi insanın biyolojik sensörlerinden veri alır ve çevreden doğru bir şekilde çıkarım yapmak için ses, ışık, ısı ve benzeri giriş sinyalleri birbiri ile birleştirir. Şekil 10 da beyinin ne şekilde füzyon yaptığını göstermektedir. Bu alandaki en iyi örneklerden biri, beyin TV izlerken ses ve görüntü verilerini birleştirmesidir. Ancak bu görevi yerine getirmek için pratik çerçeveleri ve matematiksel yöntemleri oluşturmak yirmi yıldan fazla sürmüştür [116, 117]. 1984 yılında, Amerika Birleşik Devletleri'nde veri füzyonu için genel bir çerçeve

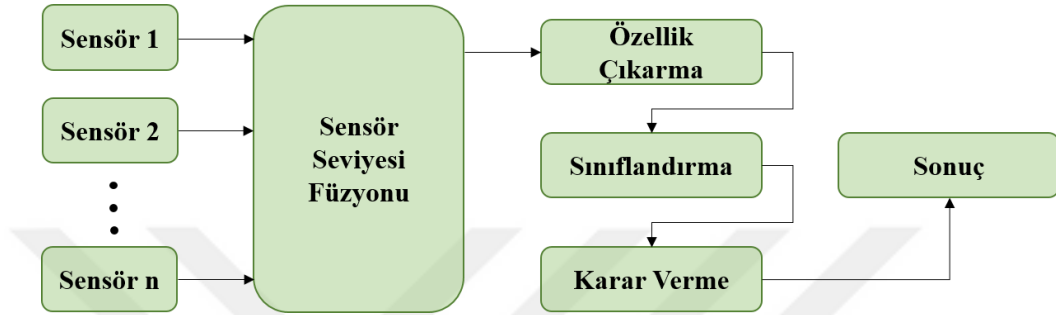
tanımlandı ve yayınlandı. İlerleyen yıllarda akademik çevreler için önemini daha şeffaf hale getiren ve büyük endüstrileri kullanımları ve avantajlarıyla tanıştıran birçok konferans ve kitap yayınlandı. Bu yıllarda "veri füzyonu" terimi pek çok bilim insanı için tanıdık bir terim değildi; ama, son yıllarda, birkaç sensörden gelen verilerin birleştirilmesi sorunu oldukça fazla dikkate alınmıştır [19, 118]. Günümüzde veri füzyonu, bilginin tüm alanlarında geniş uygulamalara sahiptir, bu nedenle endüstrinin, bugünün ve geleceğin bilgisinin ayrılmaz bir parçası haline gelmiştir. Askeri endüstrideki hayati uygulamaları arasında, askeri teçhizatın otomatik olarak yönlendirilmesi, savaş alanı yönetimi, füzelerin yerlerinin tahmin edilmesi ve benzerlerini belirtilebilir. Ayrıca, durum izleme, robotik, coğrafya ve ekonomi gibi birçok alanda uygulamaları da vardır [119, 120].



Şekil 10. İnsan bilgi füzyon sisteminin basit gösterimi [121]

Basit bir deyişle, veri füzyonu yöntemi, bir sistemin özelliklerini ve durumlarını daha doğru şekilde tahmin etmek için farklı sensörlerden gelen verileri birleştirir. Bu teknik, çeşitli kaynaklardan toplanan verileri birleştirir ve elde edilen bilgileri analiz edilen sistemin koşullarıyla ilişkilendirerek en iyi karara doğru hareket eder. Bu yöntem, birkaç bağımsız sensörden elde edilen verilere dayalı olarak gelişmiş ve tahmine dayalı bir sistem modeli

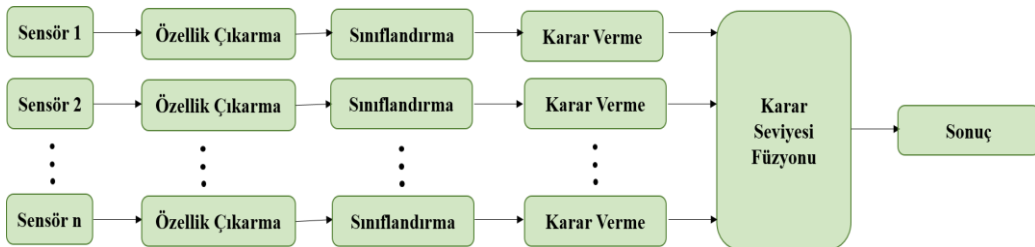
oluşturmayı amaçlar. Daha fazla sensör kullanmanın rastgelelik, araç, analiz vb. dahil olmak üzere tüm hataların olasılığını büyük ölçüde azalttığı bilimsel olarak kanıtlanmıştır. Ayrıca çok sensörlü bir veri füzyonu stratejisinin kullanılması sistem performansının kararlılığını artırabilir çünkü her sensör diğer sensörler hasar görmüş veya kullanılamıyor olsa bile, en azından kullanıcıya kendi bilgilerini sağlayabilir [116, 117, 122].



Şekil 11. Sensör seviyesinde füzyonu



Şekil 12. Özellik seviyesinde füzyonu



Şekil 13. Karar seviyesinde füzyonu

Bilgi füzyonu alanında gündeme getirilen konulardan biri, bilginin nasıl birleştirileceğinin belirlenmesidir. Genellikle birleştirme işlemi veri seviyesinde (düşük

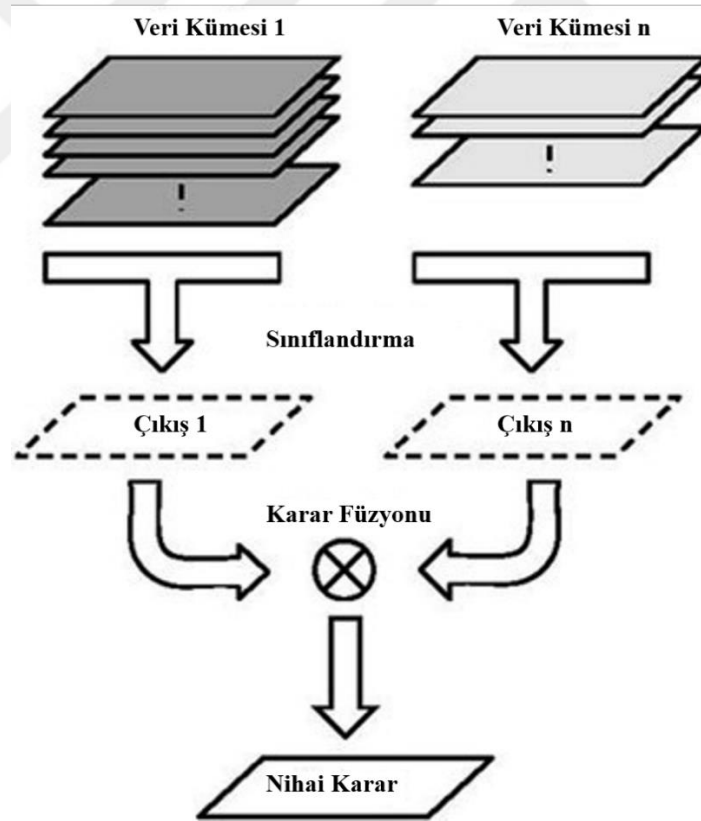
seviye), özellik seviyesinde (orta seviye) veya karar seviyesinde (yüksek seviye) yapılır [123]. Veri seviyesinde birleştirme mekanizması, doğrudan sensörlerin ham verileri üzerinde yapılır. Yani bu füzyon türünde herhangi bir analiz yapılmadan önce farklı sensörlerin verileri birleştirilir. Öznitelik çıkarma ve karar verme aşamaları, birleştirilmiş verilere dayalı olarak yapılır [21, 122]. Özellik düzeyinde, sensör verilerinden veya bilgi kaynaklarından çıkarılan özellikler üzerinde birleştirme gerçekleştirilir. Başka bir deyişle, bu tür bir füzyonda, sensörlerin her birinin bilgilerinin tanımlayıcı özelliklerinin çıkarılması işlemi ayrı ayrı yapılır. Bir sonraki adımda, sistemlerin durumunu analiz etmek için verilerin tanımlayıcı özellikleri birleştirilir. Bu durumda tespit süreci, bilgi tabanlı yöntemler (uzman sistemler ve bulanık mantık gibi) veya öğrenme tabanlı yöntemler (Bayes teorisi ve sinir ağları gibi) gibi teknikleri içerir [124]. Karar verme seviyesindeki füzyon, daha yüksek bir seviyede çalışır, farklı nesnelerden veya sensörlerden elde edilen yorumların sonuçlarının birleşimi ve entegrasyonu yapılır. Bu tür füzyonda, her bir sensörün bilgisiyle ilgili olarak tanımlayıcı özelliklerin çıkarılması ve nesnelerin algılanması işlemi ayrı ayrı yapılır ve bir sonraki aşamada Bayes teorisi veya DS yöntemi gibi tekniklerine dayalı olarak kararların birleştirilmesi yapılır. Her sensörün sonuçlarındaki üst düzey belirsizlik, karar verme füzyonu kullanılarak çözülür [2, 17]. Şekil 11, 12 ve 13 sıra ile sensör, özellik ve karar seviyesinde füzyonun aşamalarını göstermektedir.

1.6.1. Dempster-Shafer Yöntemi

DS'in inanç fonksiyonları hakkındaki teorisi, Dempster tarafından üst ve alt olasılıklara dayalı olarak kurulmuş ve daha sonra Shafer tarafından genişletilmiştir [125, 126]. Bu teori, kanıtlarla ilgili farklı görüş ve fikirlerden kümülatif kanıtları sentezlemenin güçlü bir yoludur. Olasılık teorisiyle karşılaştırıldığında, DS'in inanç işlevleri teorisi, bilinmeyen ve belirsiz kanıtları kullanarak karar vermeyi desteklemek için daha fazla bilgiyi dikkate almaktadır. Önceden bilgi ve olasılıklar olmadan belirsiz ve farklı kanıtlar için çözümler türeten bir mekanizma sunmaktadır. Bu teori, bilgi azaltma [127], hata tespiti [128], çok sınıflı sınıflandırmalar [129] ve tedarikçi seçimi [130] gibi birçok alanda etkili ve başarılı uygulamalara sahiptir. Bu teorinin uygulamalarından biri de teorinin diğer teori ve yöntemlerle birleştirilmesi ve geliştirilmesi ile karar verme problemleri alanında önerilen çalışmalardır [131, 132]. Şekil 14 bu teorinin karar füzyonu mekanizmasının basit bir şekli gösterilmektedir. Karar verme, olası seçenekler arasından en iyi seçeneği seçmek için karar

seeneklerini sıralama sürecidir. Genellikle, gerek durumlarda karar verme, belirsizlik ve yetersiz, kesin olmayan veya eliřkili bilgiler altında gerekleřir. Bu nedenle, llen verilerin doėruluėuna, zelliklerine ve mevcut kanıtlara gre, belirsizlik altında karar verme problemlerini zlemek iin farklı yntemler ve neriler nerilmiřtir.

Rastgelelik, yetersizlik, yanlıřlık ve eliřkiden kaynaklanan llebilir belirsizlik nedeniyle, karar verme sürecinde karar seeneklerini sıralamak ve en iyi zm bulmak iin eřitli ereveler ortaya konulmuřtur. Bulanık kme teorisi ve DS teorisi, belirsizlik kořulları altında problem zlemek iin iki nl teoridir [133]. Bu teorilerin matematiksel yapısının temel iliřkileri ve bazı zellikleri [126, 134-136]'de sunulmuřtur. Belirtildiėi gibi, eksik, kesin olmayan veya bilinmeyen kanıt, bilgi eksikliėi ve belirsizliėe sebep olabilir. Bu nedenle, veri ve bilgiyi artırıp, karar vermeyi iyileřtirerek en iyi seeneėi seerek bu belirsizliėi azaltmak mmkndr.



řekil 14. DS teorisinde kararların nasıl birleřtirildiėini gsteren basit bir grafik

Bulanık mantık çerçevesi genellikle kesin olmayan bilgilere ve DS'in yetersiz bilgi teorisine dayalı olarak önerilmektedir [133]. DS ve bulanık teorilerin bir kombinasyonuna dayanan birçok çalışma sunulmaktadır [137-139]. Bu alanda yapılan çalışmaların amacı, farklı bilgi kaynaklarından elde edilen mevcut kanıt ve inanç fonksiyonlarına dayalı olarak karar verme sürecinde en iyi seçeneğin seçilmesi için uygun bir matematiksel çerçeve sağlamaktır. İnanç fonksiyonu, bu yöntemde en iyi karar seçeneğini seçmek için yetersiz bilgiyi analiz etmede önemli bir rol oynamaktadır. İnanç fonksiyonları kavramlarını kullanarak karar verme problemlerini çözmek için önerilen kaynaklardan bazıları arasında, Yang'ın çok özellikli karar verme problemlerinin geliştirilmesi ve analizi için bir inanç yapısı tasarlama ve sunma çalışmasından bahsedilebilir [140]. Ayrıca Yang ve Sen, karar teorisi ve DS teorisine dayalı olarak belirsizlik altında nitel modeli değerlendirmek için kanıt teorisin yaklaşımını kullanarak geliştirilmiş bir model önermişlerdir [141].

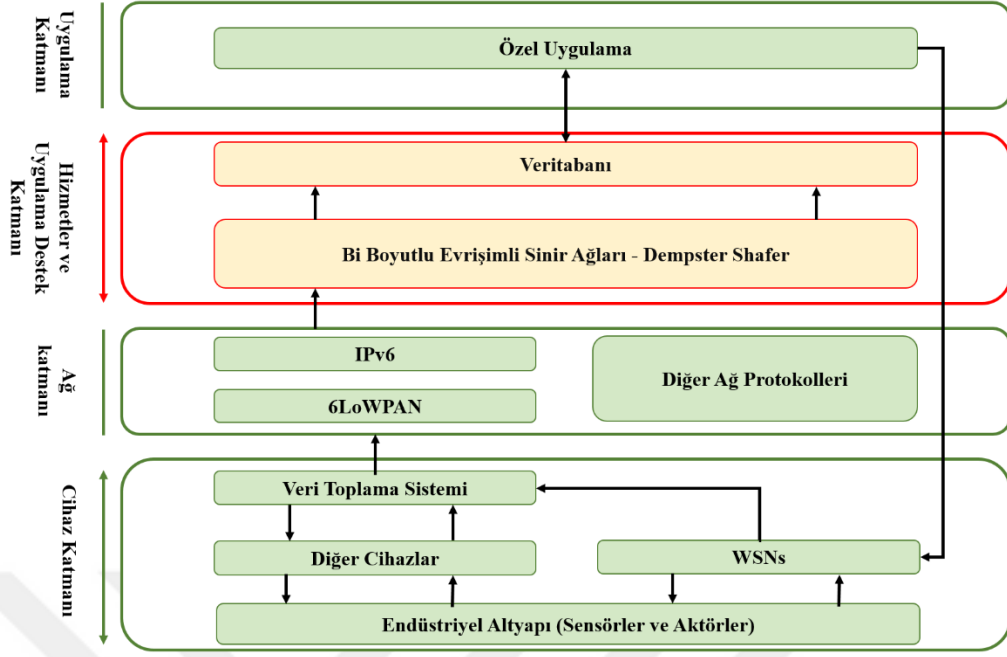
2. YAPILAN ÇALIŞMALAR

Bu bölümde, bu tezde önerilen iki yöntem açıklanacaktır. Her iki yöntem IIoT de karar verme amacıyla sunulmaktadır. İlk yöntem, ANN ve SVM sınıflandırmaları ve DS'in karar füzyonu teorisini karar füzyonunu gerçekleştirmek için kullanır. Bu yöntem orta ölçekli endüstriyel ortamlar için uygundur. İkinci yöntem, sınıflandırma için 1DCNN ve karar füzyonu için DS'in karar füzyonu teorisini kullanır. Bu yöntem, büyük ölçekli endüstriyel ortamlar için kullanılabilir. Bu iki yöntemin detayları bu bölümün devamında açıklanacaktır.

2.1. Küçük Boyutlu IIoT'ler için DS Kullanarak Karar Verme

Şekil 15, ITU tarafından önerilen modele göz önünde bulundurarak önerilen yöntemin mimarisini göstermektedir. Bu mimari, cihaz, ağ, hizmet ve uygulama destek ile uygulama katmanlarını içermektedir.

Önerilen yöntem, cihaz katmanından elde edilen verileri analiz etmek için bu mimarinin hizmet ve uygulama destek katmanında sunulmaktadır. Şekil 16, bu mimariye göre önerilen yöntemin akış şemasını göstermektedir. Cihaz katmanı, sensörler ve aktörlerden oluşur. Veriler, sensörler kullanılarak endüstriyel ortamdan toplanıp ağ katmanına iletilir, daha sonra ise hizmet ve uygulama destek katmanından gelen kararlara göre ortamda ilgili işlemleri aktörleri kullanarak gerçekleştirir. Ağ katmanı, uzun adresleme aralığı nedeniyle IIoT'de önemli bir protokol olan IPv6 protokolünü ve Kablosuz Algılayıcı Ağların (Wireless Sensor Network, WSN) IIoT'nin bir parçası olmasını sağlayan otomatik yapılandırılmış 6LoWPAN gibi verileri internet üzerinden iletmek için farklı protokolleri içerir. Hizmetler ve uygulama desteği katmanı, erken uyarı ve vaktinde davranmak için veri dönüştürme, sınıflandırma ve sistem doğruluğunu iyileştirme için çıkarımların birleştirilmesinden sorumludur. Uygulama katmanı, ortama özel herhangi bir uygulama içerir.



Şekil 15. Önerilen yöntemin mimarisi

Hizmetler ve uygulama destek katmanında önerilen yöntem, dört kısımdan yanı veri dönüştürme ve özellik çıkarma, veritabanı, sınıflandırma ve bilgi birleştirme bölümlerinden oluşmaktadır.

2.1.1. Veri Dönüştürme ve Özellik Çıkarma

Endüstriyel ortamlarda analiz için toplanan veriler büyük miktarda olup, gereksiz veriler içerebilmektedir. Bu nedenle büyük verileri bir dizi azaltılmış özelliğe dönüştürmek için bazı yöntemler geliştirmesi gerekmektedir. Bu yöntemler özellik çıkarma yöntemleri olarak tanınmaktadır. Özellik çıkarma, bazı fonksiyonları kullanarak verilerin fazlalığını azaltmak, verilerin önemli ve belirleyici özelliklerini seçmektir. Özellik çıkarmanın amacı, ham verileri analiz etmek için kullanılabilir hale getirmektir. Bu çalışmada, her bir örnek için geniş bir veri içeren ses ve titreşim verisinin örneklerinden özellik çıkarmak için bazı istatistiksel fonksiyonlar kullanılmalıdır. Tablo 9'de bu fonksiyonların önemli olanlar listelenmiştir [142].

2.1.2. Sınıflandırma

Sınıflandırma, etiketleri olan önceki verilere dayanarak yeni verilerin etiketini tahmin etmek için bir model oluşturan bir bilimdir. Sınıflandırma, ML ve veri madenciliğinin temel dallarından birisi olup geçmiş eylemlerden toplanan verilere dayanmaktadır, ki bu veriler uzmanların bilgisine dayalı olarak etiketlenmiştir. Bu amaçla örüntülü, ANN [143], ANFIS [144], ve DSVM [145] gibi birçok algoritma kullanılmaktadır. Bu blokta, ortamda hangi eylemin yapılacağına karar vermek için sinyallerden çıkarılan özellikler sınıflandırılır. Bu blokta sınıflandırma için farklı algoritmalar kullanılabildiğinden, farklı kanıt ve algoritmalar arasında net olmayan ve hatta çelişkili çıkarımlar yapılabilir. Bu nedenle, kararlar arasındaki tutarsızlıkları ve belirsizlikleri azaltmak için, bir sonraki bölümde açıklanacak olan bilgi füzyonu adına başka bir blok gereklidir. Bu bölümde bu çalışmada kullanılan DSVM ve ANN iki yaygın sınıflandırma algoritması açıklanacaktır.

Tablo 9. Özellik çıkarımı için bazı önemli istatistiksel fonksiyonlar

Özellik Çıkarma	Denklem	Kaynak
Ortalama Değer	$\bar{x} = \frac{\sum_{i=1}^n x_i}{n}$ (1)	[146]
Standart Sapma	$\sqrt{\frac{\sum_{i=1}^n (x_i - \bar{x})^2}{n}}$ (2)	[147]
Karekök Ortalama	$\sqrt{\frac{\sum_{i=1}^n (x_i)^2}{n}}$ (3)	[148]
Üçüncü Merkezi Moment	$\frac{\sum_{i=1}^n (x_i - \bar{x})^3}{n}$ (4)	[149]
Dördüncü Merkezi Moment	$\frac{\sum_{i=1}^n (x_i - \bar{x})^4}{n}$ (5)	[149]
Basıklık	$\frac{\frac{1}{n} \sum_{i=1}^n (x_i - \bar{x})^4}{\left(\frac{1}{n} \sum_{i=1}^n (x_i - \bar{x})^2\right)^2}$ (6)	[150]
Çarpıklık	$\frac{\frac{1}{n} \sum_{i=1}^n (x_i - \bar{x})^3}{\left(\frac{1}{n} \sum_{i=1}^n (x_i - \bar{x})^2\right)^{\frac{3}{2}}}$ (7)	[151]
FM4	$\frac{n \sum_{i=1}^n (x_i - \bar{x})^4}{(\sum_{i=1}^n (x_i - \bar{x})^2)^2}$ (8)	[152]

2.1.2.1. Dendrogram Tabanlı Destek Vektör Makinesi (DSVM)

DSVM, sınıflandırma için kullanılan denetimli öğrenme yöntemlerinden biridir. Birincil SVM Algoritması 1963'te Vladimir Vapnik tarafından önerilmiş ve 1995'te doğrusal olmayan model için Vapnik ve Corinna Cortes tarafından geliştirilmiştir [145].

Bir sistemin farklı davranışlarından elde edilen bilgilerin aşağıdaki gibi olduğunu varsayalım:

$$\{(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)\} \quad (9)$$

x_i girdi alanıdır ve y_i ikili sınıf etiketlerini ifade etmektedir:

$$y_i \in \{1, -1\} \quad (10)$$

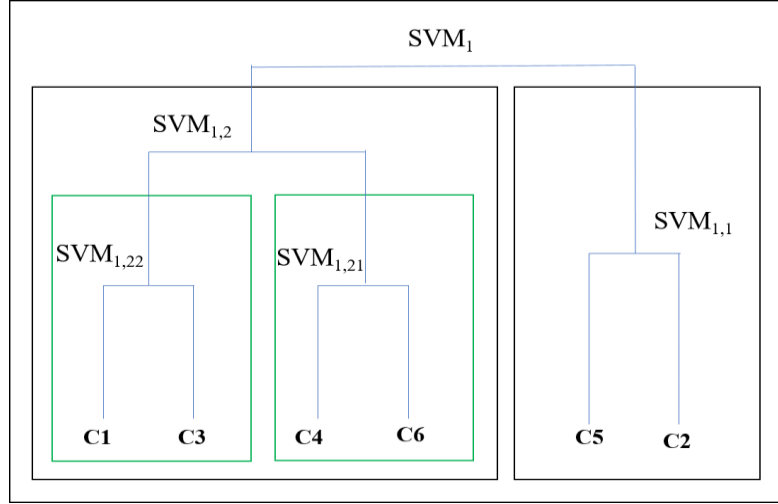
SVM nin amacı, verileri iki sınıftan maksimum marjla ayıran optimal hiper düzlemi (11) bulmaktır.

$$w \cdot x + b = 0 \quad (11)$$

Maksimum marjı garanti etmek için w ve b değişkenleri aşağıdaki eşitsizliği sağlamalıdır.

$$y_i(w \cdot x_i + b) \geq 1, \quad i = 1, 2, \dots, n \quad (12)$$

Hiper düzleme en yakın olan x_i 'ye destek vektörleri denir.



Şekil 16. AHC yöntemi ile sınıfları hiyerarşik olarak gruplama [145]

Bir x_i noktası ile bir hiperdüzlem (x, w) arasındaki mesafe geometriye göre hesaplanır:

$$d = \frac{w^T \cdot x + b}{\|w\|} \quad (13)$$

İki sınıf grubu arasındaki minimum mesafe aşağıdaki denklem ile hesaplanmaktadır:

$$d_{min} = \min_{\frac{x}{y}=+1} \frac{w^T \cdot x + b}{\|w\|} - \max_{\frac{x}{y}=-1} \frac{w^T \cdot x + b}{\|w\|} \quad (14)$$

kurallı hiperdüzlem için pay değeri bire eşittir. Böylece, denklem (14)'te yerine koyarsak aşağıdaki denklem elde edilmektedir:

$$d_{(w_0, x_0)} = \frac{2}{\|w\|} = \frac{2}{\sqrt{w_0 \cdot w_0}} \quad (15)$$

Son olarak, $d_{(w_0, x_0)}$ maksimize etme problemi, bazı kısıtlamalarla $L(w)$ fonksiyonunu minimize etme durumuna eşittir.

$$\min_{w, b} L(w) = \frac{1}{2} \|w\|^2 \quad \text{subject to } y_i(w^T \cdot x_i + b) \geq 1, \quad i = 1, 2, \dots, n \quad (16)$$

polinom gibi bazı çekirdek fonksiyonlar SVM'de kullanılır:

$$k(x, y) = (1 - x \cdot y)^d \quad (17)$$

Üstel RBF:

$$k(x, y) = \exp\left(\frac{-\|x-y\|}{2\sigma^2}\right) \quad (18)$$

ve Gauss RBF:

$$k(x, y) = \exp\left(\frac{-\|x-y\|^2}{2\sigma^2}\right) \quad (19)$$

Çok sınıflı sınıflandırma için, her biri $y_i \in \{c_1, c_2, \dots, c_k\}$ ile etiketlenmiş $x_1, x_2, \dots, x_n$ girdi örnekleri kümesini varsayılmakta, k sınıf sayısı olmaktadır ($k \leq n$). DSVM yönteminin ilk adımı, k sınıfı için k ağırlık merkezinin hesaplanmasını içerir. Daha sonra bu k merkezler için artan hiyerarşik kümeleme yöntemi (AHC) kullanılır (Şekil 16). Bir sonraki adımda, her SVM bir düğüme bağlanır ve bu düğümün iki alt ağacının elemanları ile eğitilir [145].

2.1.2.2. Yapay Sinir Ağları

ANN, üç giriş, çıkış ve işlem katmanından oluşur. Her katman, kullanıcı nöronlar arasındaki iletişimi kısıtlamadıkça, genellikle diğer katmanların tüm nöronlarına bağlı olan bir grup sinir hücresi (nöron) içerir; Ancak her katmanın nöronlarının aynı katmandaki diğer nöronlarla hiçbir bağlantısı yoktur. Nöron, sinir ağlarının temelini oluşturan en küçük bilgi işlem birimidir. Bir sinir ağı, farklı katmanlara yerleştirilerek farklı katmanlardaki nöronlar arasındaki bağlantılara dayalı belirli bir mimari oluşturan bir dizi nörondur. Bir nöron doğrusal olmayan bir matematiksel fonksiyon olabilir, sonuç olarak bu nöronların topluluğu tarafından oluşturulan bir sinir ağı da tamamen karmaşık ve doğrusal olmayan bir sistem olabilir. Sinir ağında, her nöron bağımsız olarak hareket eder ve ağın genel davranışı, birçok nöronun davranışının sonucudur. Başka bir deyişle, nöronlar işbirlikçi bir süreçte birbirlerini düzeltirler [153].

2.1.3. Veritabanı

IoT projeleri karmaşık olduğu için çeşitli veri depolama stratejileri kullanılması gerekmektedir. Veri depolama için kullanılan cihazlar fiziksel olarak güçlü olmalıdır. Akıllı endüstride verileri, işlenmiş bilgileri ve bilgileri depolamak için veritabanı sistemine ihtiyaç vardır. Veritabanı, şimdi ve gelecek için ihtiyaç olan bilgileri depolamakla tüm endüstriyel operasyonlarını koordine etmek için imkanlar sağlamaktadır [154].

2.1.4. Bilgi Füzyonu

Bu blokta, yüksek doğruluk elde etmek ve belirsizliği azaltmak için farklı sınıflandırıcıların sonuçları birleştirilir. Bu amaçla Bayesian, DS, bulanık mantık ve anlamsal bilgilerin birleştirilmesi gibi çeşitli çıkarımsal yöntemler kullanılmaktadır [9]. Bu çalışmada, Şekil 17'deki akış şemasına göre önerilen yöntemin son aşamasında karar füzyonu için DS ve Yager teorisi kullanılmıştır. DSVM ve ANN sınıflandırmalardan çıkan kararlar eğer birbirine yakın ise DS eğer bu kararların arasında çok fark var ise Yager birleştirme teorisi kullanılmıştır. Aşağıda bu iki birleştirme teorisinin detayları açıklanmaktadır.

Burada 2^Ω , Ω 'un alt kümelerinin kümesi ve ϕ , bir boş değerler kümesidir.

İnanç işlevi (Bel) ve inandırıcılık fonksiyonu (Pls), belirsizlik argümanındaki en önemli işlevlerdir. Ve inancın alt ve üst sınırlarını istenen gerçeklik değişkeni olarak temsil eder ve şu şekilde tanımlanır:

$$Bel(A) = \sum_{B \subseteq A} m(B), \forall A \subseteq \Omega \quad (21)$$

$$Pls(A) = 1 - \sum_{B \cap A = \phi} m(B) \quad (20)$$

Karar vermek için birden fazla bilgi kaynağı olduğunda, farklı bilgi kaynaklarından alınan kararlar birleştirilir ve nihai karar verilir. Bu kombinasyon için DS'in bileşim yasası kullanılır:

$$m_1 \oplus m_2(A) = \begin{cases} (1 - k)^{-1} \sum_{B \cap C = A} m_1(B) * m_2(C), & \text{if } A \neq \emptyset \\ 0, & \text{if } A = \emptyset \end{cases} \quad (24)$$

aşağıdakini varsayarak:

$$k = \sum_{B \cap C = \emptyset} m_1(B) * m_2(C) \quad (25)$$

K , istenen olayla ilgili çeşitli bilgi kaynaklarının çıkarımların arasındaki tutmamak derecesini temsil eder. eğer iki kaynak için k değeri 1'e yakınsa, iki kaynak tam bir çelişki içindedirler ve k değeri 0'a yakınsa iki kaynak birbirine daha uyumludur. DS'in birleştirme kuralı, farklı bilgi kaynaklarından alınan kararlar arasındaki uyumsuzlukları dikkate almaz. Bu nedenle, bu teori uyumsuz kanıt kararları için uygun değildir. Bu bağlamda Yager, θ kümesindeki uyumsuz bilgi kaynaklarını sınıflandırarak DS'in teorisini geliştirmiştir. Bu şekilde ki sınıfını hesaplamayabildiği verinin sınıfın [156] aşağıdaki denklem ile elde edilebilir:

$$\theta_i = 1 - \alpha_i \quad (26)$$

Burada α_i , i . kanıt veya bilgi kaynağının önem faktörüdür. Şekil 17'deki akış şemasında gösterildiği gibi, her sınıflandırıcının ortalama doğruluğu, bunun α_i değeri olarak kabul edilebilir.

Ayrıca, Yager, temel olasılık atama fonksiyonu (m) yerine zemin olasılık atamalarını (q) kullanmıştır bu parametre aşağıdaki denklemle hesaplanmaktadır:

$$q(A) = \sum_{\cap A_i = A} [m_1(A_1) \times m_2(A_2) \times m_3(A_3) \times \dots \times m_i(A_i)] \quad (27)$$

Yager yönteminde $q(\emptyset)$ sıfırdan büyük olmalıdır, bu da iki sınıflandırıcı arasında uyumsuzluk olduğu anlamına gelmektedir. Yeni BPA'lar aşağıdaki gibi hesaplanmaktadır:

$$\begin{cases} m_i = \alpha_i \times O_i \\ \text{where } \alpha_i = 1 - \alpha_i \end{cases} \quad (28)$$

O_i , i . bilgi kaynağının çıktı değeridir. yeni birleştirme kuralı aşağıdaki gibi hesaplanmaktadır:

$$m(A) = \frac{q(A)}{1 - q(\emptyset)} \quad (29)$$

Veri kaynakları ikiden fazla ise birleştirme kuralları aşağıdaki gibi hesaplanabilir:

$$m_1 \oplus m_2 \oplus \dots \oplus \dots = ((m_1 \oplus m_2) \oplus \dots) \oplus \dots \quad (30)$$

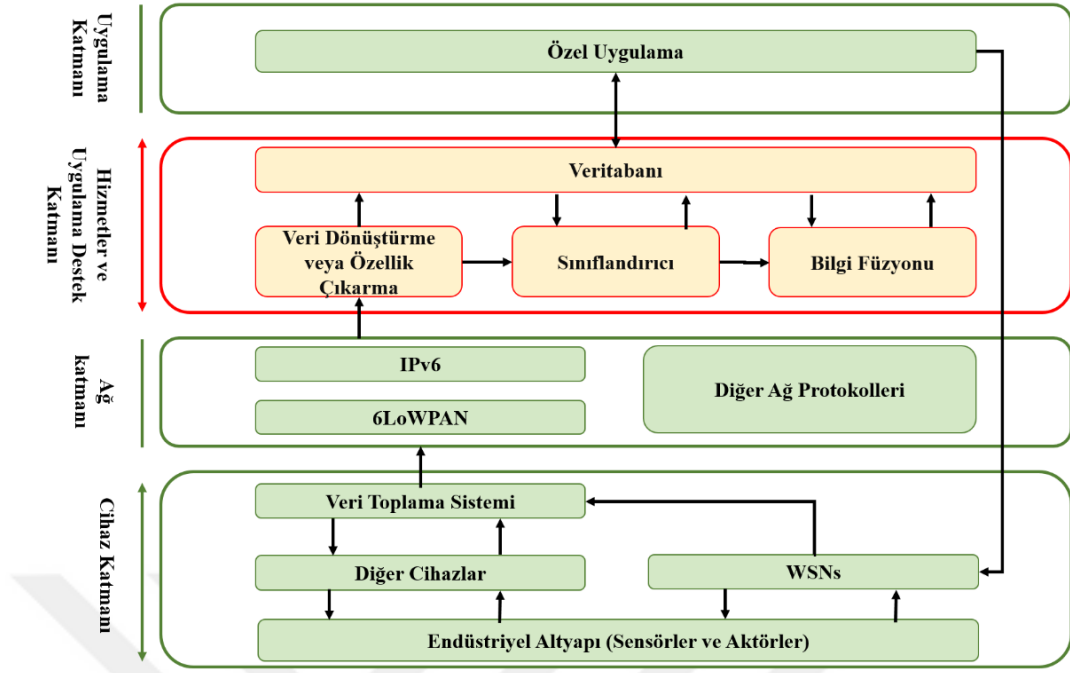
Bu bölümde, ML yöntemleri ve karar füzyonu teorisi kullanılarak yeni bir karar verme mekanizması önerilmiştir. Bu yöntemin akış şeması Şekil 17'de bulunmaktadır. Burada, farklı kaynaklardan gelen verilerin öz nitelikleri çıkarılmakta ve daha sonra farklı sınıflandırıcılar ile sınıflandırılmaktadır. Ortaya çıkan kararlar, nihai bir karara varmak için DS birleştirme teorisi kullanılarak birleştirilmektedir. Değerlendirme için önerilen yöntem Python programlama dili kullanılarak uygulanmış, iki veri seti üzerinde test edilmiş ve bu testlerin sonuçları bir sonraki bölümde sunulmuştur.

2.2. Tek Boyutlu Evrişimli Sinir Ağları ve DS Teorisin Kullanarak IIoT'lerde Anormallik İçin Karar Verme (DS-1DCNN)

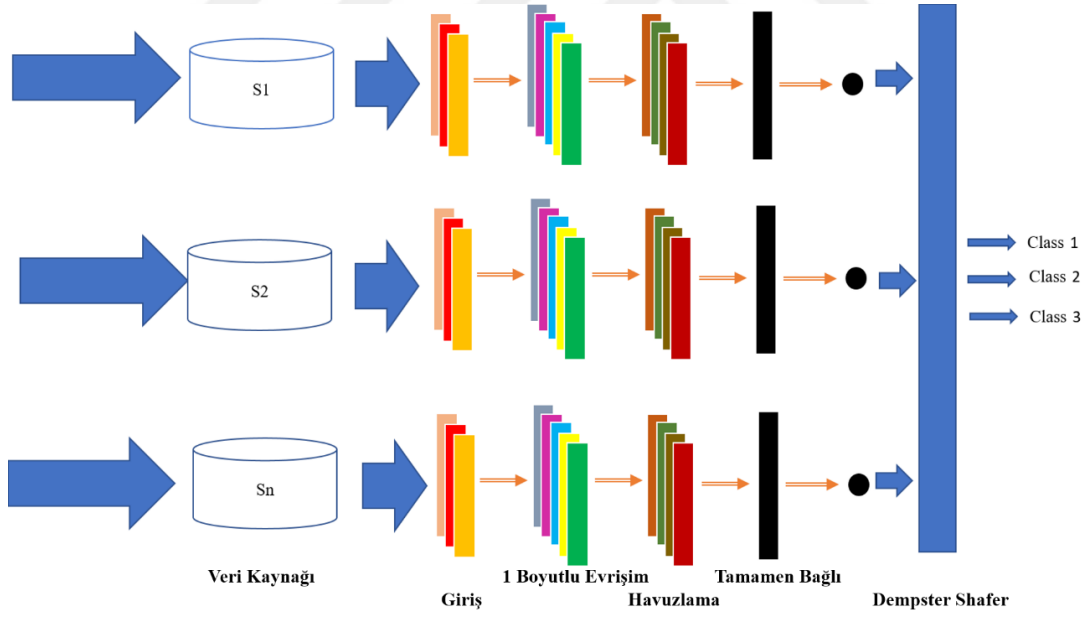
Bu çalışmada, IIoT'de karar vermek için [2]'de sunulan modelden ilham alınmıştır (Şekil 18). Bu çalışmanın ana katkısı, 1DCNN ve DS teorisini birleştirerek hizmet katmanında IIoT anormalliklerini daha verimli bir şekilde tespit etmektir. Bu yöntem, birden fazla sensör türü bulunduğu durumlarda etkili olabilir. Analiz edilecek birden fazla veri türünün olduğu durumlarda kararda belirsizlik olabileceği gibi, farklı veri türleri için alınan kararlar da tutarsızlık gösterebilir. Bu nedenle, bu çalışmada kararları birleştirmek, belirsizlik ve kararlar arasındaki farkı ortadan kaldırmak amacıyla DS'in kara füzyonu teorisi kullanılmıştır. Ayrıca, mevcut IIoT'de üretilen veriler çok büyük olduğundan dolayı veri analizi için DL algoritmaları uygun görünmektedir. İşlenecek büyük hacimde veri akışı olduğu durumda, tamamen bağlı ağlar çok karmaşıktır ve çok parametresi vardır. Bu gibi durumlarda, 1DCNN ağları, daha az parametreye sahip ve tam bağlı ağlardan daha hızlı olan daha yüksek seviyeli özellikleri çıkararak veri sırasını basitleştirmek için bir ön işleme adımı olarak kullanılabilir. Bu nedenle, bu çalışmada verilerin analizi için daha az karmaşıklığı ve daha az parametresi olan 1DCNN kullanılmıştır.

Önerilen yönteminin yapısı Şekil 19'de gösterilmektedir. Burada sensörlerden elde edilen veriler 1DCNN ağlarında paralel olarak işlenir. Son katman olan DS katmanında, 1DCNN ağlarından çıkan sonuçlar DS birleştirme kurallarını kullanılarak birleştirilir. Daha sonra her 1DCNN ağının ve DS katmanının hatası hesaplanır. Son olarak, bu elde edilen hatalar 1DCNN ağlarının ağırlıklarını ve parametrelerini güncellemek için kullanılır.

Önerilen yöntemde, modelin parametrelerin ve ağırlıkların güncellemek için Adam optimize algoritmasının bu yöntem için özelleştirilmiş 39 ve 40 denklemleri kullanılmaktadır. Bu kısımda önerilen yöntemde kullanılan algoritmalar ve yöntemin detayları açıklanacaktır.

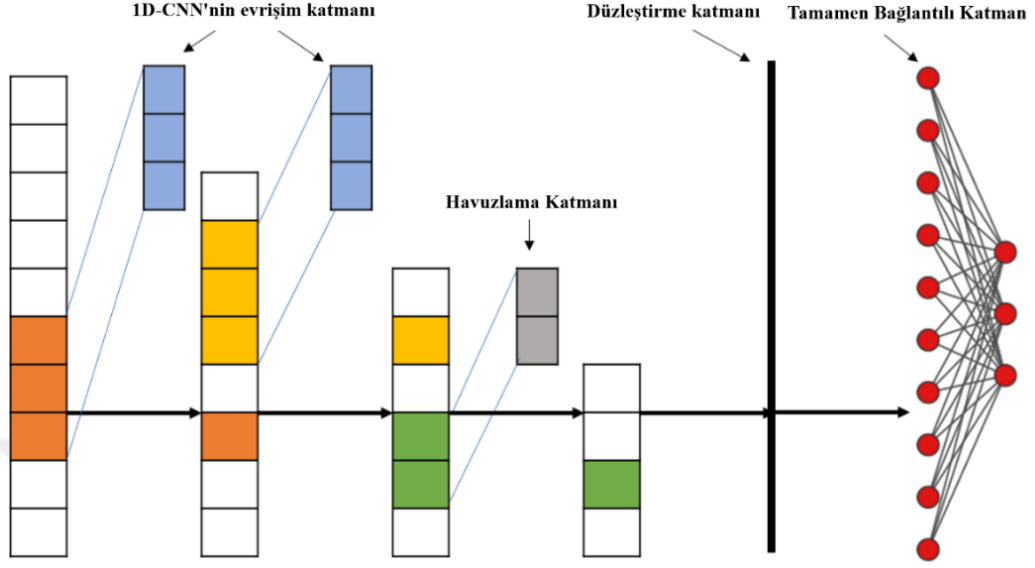


Şekil 18. IIoT Mimarisi ve önerilen modelin konumu



Şekil 19. Karar verme için önerilen yöntemin yapısı

2.2.1. Tek Boyutlu Evrişimli Sinir Ağları



Şekil 20. Tek boyutlu evrişimli sinir ağı (1DCNN) mimarisi [158]

Bir boyutlu sinyal verilerini analiz etmek için uygun yöntemlerden biri 1DCNN'dir. Son zamanlarda 1DCNN, kişisel tıbbi veri sınıflandırması, anormallik tespiti, arıza tespiti ve güç elektroniği tanımlaması gibi çeşitli uygulama alanlarında kullanılmıştır. Bu yöntemin önceki uygulamalara göre önemli avantajı, 1DCNN'nin her bir ağ katmanındaki tüm sinyal yerine yerel bilgileri dikkate alarak sinyal özelliklerini çıkarmasıdır. Bu, daha az öğrenilebilir parametre aracılığıyla daha hızlı ağ eğitimi sağlayarak daha az maliyet ve bilgi işlem gücü sağlar [157]. Şekil 20, 1DCNN'nin mimarisini göstermektedir.

Yukarıda bahsedilen son çalışmalarda, 1DCNN 'lar, metin ve sinyaller gibi tek boyutlu uygulamalarda etkilidir. 1DCNN 'da iki tür katman vardır: 1) bir boyutlu evrişimli katmanlar 2) Sınıflandırma için kullanılan tam bağlantılı katmanlar. 1DCNN aşağıdaki parametrelere sahiptir [159]:

- 1) Evrişim ve tam bağlı katmanların sayısı.
- 2) Her evrişim katmanındaki çekirdek boyutu
- 3) Havuzlama (pooling) ve aktivasyon parametreleri nasıl seçilir.

Bu parametrelerin hepsi Şekil 20'de gösterilmiştir.

2.2.1.1. Evrişim Katmanı

Evrişim katmanı, kaydırma ve hesaplama yöntemi olarak bilinir. x , evrişim katmanının n uzunluğundaki girdisi ve w , k uzunluğundaki çekirdeği olmaktadır. Ayrıca, her evrişim işleminden sonra çekirdek penceresinin konumu s (adım sayısı) ile değişmektedir. Adım s için x ve w arasındaki evrişim aşağıdaki gibi tanımlanır [160]:

$$y(n) = \begin{cases} \sum_{i=0}^k x(n+i)w(i), & \text{if } n = 0. \\ \sum_{i=0}^k x(n+i+(s-1))w(i), & \text{otherwise.} \end{cases} \quad (31)$$

n uzunluğundaki girdi, k uzunluğundaki çekirdek ve p dolgusu için çıktı uzunluğu (o), aşağıdaki denklemden [159] elde edilir.

$$o = \left\lfloor \frac{(n + 2p - k)}{s} \right\rfloor + 1 \quad (32)$$

2.2.1.2. Havuzlama Katmanı

Bir havuzlama katmanı genellikle bir evrişim katmanından sonra yerleştirilir ve özellik haritalarının ve ağ parametrelerinin boyutunu küçültmek için kullanılır ve ayrıca fazla uydurmayı da azaltır. Maksimum havuzlama, f boyutundaki bir pencerede bir maksimum değerin seçildiği ve ardından her havuzlamadan sonra s adımıyla girdi üzerinde kaydırıldığı en ünlü havuzlama tekniğidir [159]. Maksimum havuzlama şu şekilde formüle edilir:

$$y(n) = \begin{cases} \max(x(n+i)) \text{ for } i = 0, 1, 2, \dots, (n+i+f) & \text{if } n = 0 \\ \max(x(n+i+(s-1))) \text{ for } i = 0, 1, 2, \dots, (n+i+f+(s-1)) & \text{otherwise} \end{cases} \quad (33)$$

2.2.1.3. Düzleştirme (flatten) ve Tamamen Bağlı Katman

Evrişim katmanlarının çıktısının birden fazla derinliğe sahip olduğunu varsayalım. Bu durumda, düzleştirme katmanı, evrişimli katmanların çıktısını tam bağlı katmanlara girdi olarak beslenebilecek bir biçime dönüştürür. Tam bağlantılı katmanlar, ardışık katmanlardaki tüm düğümleri birbirine bağlar. Düzleştirilmiş matris, istenen sınıflara [159] sınıflandırılmak için tam bağlantılı bir katmandan geçer.

2.2.1.4. Dempster Shafer Birleştirme Kuralları

Kanıt teorisi veya Dempster-Schaefer teorisi, belirsizlikle tartışmak için genel bir çerçevedir. Arthur P. Dempster bu teoriyi ilk önce istatistiksel çıkarımda önermiş, daha sonra Glenn Shafer tarafından epistemik belirsizlik teorisini modellemek için ana çerçeve olarak geliştirilmiştir. Bu teori, kişinin çeşitli kaynaklardan elde edilen kanıtları birleştirmesine ve mevcut tüm kanıtları dikkate alan bir inanç düzeyine ulaşmasına izin verir [116, 156]. DS'in birleştirme teorisinin detayları ve denklemleri bölüm 2.1.4.1'de anlatılmıştır.

2.2.2. Adam Optimize Edici

Adam'ın optimize edicisi, RMSProp [161] optimize edicisinin geliştirilmiş bir versiyonudur. Bu optimizasyon algoritmasında gradyanların hem ikinci momentlerinin hem de gradyanların ortalamaları kullanılır. $w(t)$ parametrelerine ve t 'nin mevcut eğitim yinelemesine atıfta bulunduğu bir kayıp fonksiyonu $L(t)$ 'ye göre, Adam parametreleri şu şekilde güncellenir:

$$m_w(t+1) \leftarrow \beta_1 m_w(t) + (1 - \beta_1) \nabla_w L(t) \quad (34)$$

$$v_w(t+1) \leftarrow \beta_2 v_w(t) + (1 - \beta_2) (\nabla_w L(t))^2 \quad (35)$$

burada β_1 ve β_2 , algoritmanın hiper parametreleridir. $m_w(t)$ ve $v_w(t)$, sırasıyla gradyanların toplamı ve t zamanında geçmiş gradyanların karelerinin toplamıdır. Çoğunlukla ilk adımlarda ve bozunma oranlarının düşük olduğu durumlarda sıfıra

yakındırlar. Bunlara karşı koymak için birinci ve ikinci anın akıcı olduğu tahmin edilir [109, 162]:

$$\hat{m}_w = \frac{m_w(t+1)}{1 - \beta_1(t+1)} \quad (36)$$

$$\hat{v}_w = \frac{v_w(t+1)}{1 - \beta_2(t+1)} \quad (37)$$

Son olarak, parametreleri güncellemek için Adam güncelleme formülü aşağıdaki gibi kullanılır:

$$w(t+1) \leftarrow w(t) - \frac{\hat{m}_w}{\sqrt{\hat{v}_w + \epsilon}} \quad (38)$$

burada ϵ , sıfıra bölmeyi önlemek için kullanılan küçük bir değeri olup, 10^{-8} gibi değerler alabilmektedir.

2.2.3. Önerilen Yöntem Ayrıntıları

Önerilen yöntem sözde kodu Tablo 11'de, ilgili parametreler Tablo 10'da gösterilmekte olup önerilen modelin ana yapısını bu tablolarda açıklanmaktadır. Bu sözde kod, modelin eğitilme yöntemini ve ağ parametrelerinin ve ağırlıkların güncellenmelerini göstermektedir. Ayrıca önerilen algoritma bu çalışmada kullanılan veri setini göz önünde bulundurarak geliştirilmiştir.

Tablo 10'da yöntemin girdi ve çıktı parametreleri özetlenmiştir. Giriş parametreleri, veri seti, iterasyon sayısı ve erken eğitim sonlandırması için beklenen kayıp oranından oluşmaktadır. Bu çalışmada kullanılan veri seti ses, akım ve titreşim gibi üç veri tipini (d_{s_0} , d_{s_1} , d_{s_2}) içermektedir. Bu veriler ve etiketleri eğitim (x_{train_i} , y_{train_i}) ve test alt kümelerine (x_{test_i} , y_{test_i}) bölünmektedir, i üç veri türünün (d_{s_0} , d_{s_1} , d_{s_2}) bahsedilen veri setinde gösterimini ifade eder. $w(l)$, l inci evrişim ve havuzlama katmanının ağırlıklarıdır (Çekirdekler) ve $W(j, k)$, $b(j, k)$, tam bağlı katmanın ağırlıkları ve yanlılıklarıdır. Bu parametreler tümü ilk başta rastgele başlatılmaktadır. m_{it} , öğrenme aşamasının maksimum iterasyon sayısını ve e_{er} eğitim aşamasının sonlandırılması için

beklenen hatayı ifade etmektedir. Bu iki parametreyi yazılımcı deneme ve test yöntemi kullanılarak belirtmektedir. α_r , DS katmanında kullanılmak üzere r inci bilgi kaynağının önem faktörüdür, bu parametre uzmanlar tarafından belirlenebilir. Bu çalışmada, tüm sınıflandırıcılar aynı önem faktörüne sahiptirler. Çıktılar, her iterasyonun sonuçlarını ve karışıklık matrisini, her 1DCNN'den elde edilen hatayı ve DS katmanından elde edilen hatayı içermektedir. $y_i(t)$, her t iterasyonun da 1DCNN modellerinden elde edilen eğitiminin çıktısıdır. m_i , karışıklık matrisidir ve her i veri türü için $y_i(t)$ ve y_{train_i} kullanılarak hesaplanmaktadır. $L_i(t)$, her t iterasyonun da 1DCNN modellerinden elde edilen eğitiminin hatasıdır. $L^d(t)$, her t iterasyonun da DS katmanındaki elde edilen eğitiminin hatasıdır.

Tablo 10. Önerilen yöntem sözde kodunun giriş, çıkış ve başlatma parametreleri

DS-1DCNN yönteminin Giriş Parametreleri:
$data_types: [d_{s_0}, d_{s_1}, d_{s_2}]$ Veri seti, sırasıyla ses, akım ve titreşim olmak üzere üç tür veri içerir.
x_{train_i}, y_{train_i} : feVeri setindeki üç veri türü için $i: 0, 1, 2$ eğitim kümesinin özellikleri ve etiketleri.
x_{test_i}, y_{test_i} : Veri setindeki üç tip veri için test Seti $i: 0, 1, 2$ 'nin özellikleri ve etiketleri
$w(l)$: l Evrişim ve havuzlama katmanının ağırlıkları (Çekirdekler)
$W(j, k), b(j, k)$ Tam bağlı katmanın ağırlıkları ve önyargısı
m_{it} : Öğrenme adımının maksimum tekrar sayısı
e_{er} : Mevcut eğitim hatası beklenen hatadan az olduğunda eğitim durur.
α_r : DS katmanında kullanım için r inci bilgi kaynağının önem faktörü
DS-1DCNN yönteminin çıktıları:
$y_i(t)$: Her eğitim tekrarının çıktısı (t)
m_i : i inci veri türü için her ağırlık karışıklık matrisi
$L_i(t)$: Her t inci eğitim tekrarının hatası
$L^d(t)$: DS katmanındaki eğitimin her t inci tekrarının hatası.
Parametrenin başlatılması:
$w(l), W(j, k), b(j, k)$: Bu parametreler rastgele ayarlanır.
m_{it}, e_{er} : Deneme ve yanılma ile elde edilir.

Yukarıda açıklanan giriş parametrelerini başlattıktan sonra, eğitim bir while döngüsünde başlar, iterasyon ($t < m_{it}$) ve hata ($\rho L_i(t) + (1 - \rho)L^d(t) < e_{er}$) koşulları doğru olduğu sürece devam eder. ρ , 1DCNN ve DS katman hatalarının etkisini belirleyen ağırlık parametresidir. Bu parametre bire yakınsa 1DCNN'nin etkisi daha fazladır.

Burada, 1DCNN'ler her veri türü için paralel olarak eğitilir. Her iterasyondan sonra model sonucu ($y_i(t)$) ve model kaybı ($L_i(t) = \frac{1}{2} \sum_{n=1}^N (y_{train} - p_{train_i})^2$) hesaplanılır.

Tablo 11. Önerilen yöntemin sözde kodu

$w(l), W(j,k)b(j,k)$: Bu parametreler rastgele ayarlanır.
m_{it}, e_{er} : Deneme ve yanılma ile elde edilir.
Başla
it : Tekrar değişkenini başlat.
while $t < m_{it}$ and $(\rho L_i(t) + (1 - \rho)L^d(t)) < e_{er}$ do
for each d_s in $data_types$ do
for all x_{train_i} do
p_{train_i} (Eğitim adımının tahmini) x_{train_i} formül 11-13 kullanılarak elde edilir.
Endfor
$y_i(t)$ hesapla
Her d_s için m_i karışıklık matrisini hesaplanır
$L_i(t)$ 'yi her d_s için olarak hesaplanır.
Endfor
4-10 denklemlerini kullanarak DS katmanının p_{train_d} 'sini elde etmek için tüm 1DCNN'lerin m_i 'sini birleştirilir.
p_{train_d} ve y_{train} kullanarak $L^d(t)$ as $L^d(t) = \frac{1}{2} \sum_{n=1}^N (y_{train} - p_{train_d})^2$ hesapla.
34, 35 denklemlerine göre, 39, 40 denklemleriyle yanı gradyanların toplamı ve geçmiş gradyanların karelerinin toplamı hesaplanılır.
$w(l), W(j,k)$ ve $b(j,k)$ parametreleri denklem 38 kullanılarak güncellenir
$t += 1$
Endwhile
Bitir

Bir sonraki adımda, 1DCNN'lerin her birinin m matrisi hesaplanır ve daha sonra DS katmanı tahminini (p_{train_d}) elde etmek için DS katmanında (denklem 24-30) birleştirilen 1DCNN'lerden sonuçlar elde edilir. p_{train_d} , eğitim sonlandırılırsa DS-1DCNN modelinin çıktısı ve tahminidir. Algoritmanın bir sonraki aşamada, DS katmanının kaybı ($L^d(t) = \frac{1}{2} \sum_{n=1}^N (y_{train} - p_{train_d})^2$) hesaplanılır. Ardından, 1DCNN ve DS katman kaybı, Adam optimize edicinin gradyanların toplamını ve geçmiş gradyanların karelerinin toplamını hesaplamak için kullanılır.

$$m_w(t + 1) \leftarrow \beta_1 m_w(t) + (1 - \beta_1)(\rho \nabla_w L(t) + (1 - \rho) \nabla_w L^d(t)) \quad (39)$$

$$v_w(t+1) \leftarrow \beta_1 v_w(t) + (1 - \beta_2)((\rho \nabla_w L(t) + (1 - \rho) \nabla_w L^d(t)))^2 \quad (40)$$

Burada $L(t)$ ve $L^d(t)$ sırasıyla 1DCNN ve DS katmanının hatalarını göstermekte ve ρ , Adam parametrelerinin DS katmanının sonucundan etkilenen hızlı değişiklikleri önlemek için ayar parametresi olmaktadır. Son olarak, ağırlıklar ve parametreler 38 numaralı denklemlerle güncellenir ve 39, 40 ve 37 numaralı denklemler kullanılarak oluşturulur. p_{train_d} , eğitim sonlandırılırsa DS-1DCNN modelinin çıktısı ve tahminidir. Aksi takdirde, yukarıda belirtilen adımlar eğitim bitene kadar tekrarlanır.



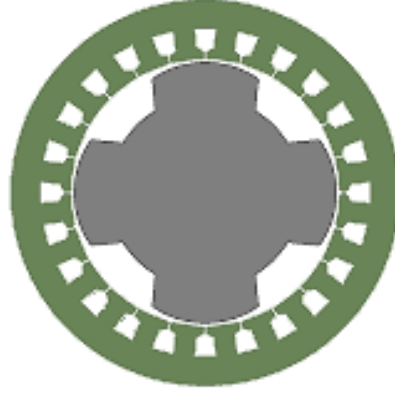
3. BULGULAR

3.1. Birinci Önerilen Yöntem

Bu bölümde, küçük boyutlu IIoT'ler için önerilen DS kullanarak karar verme yöntemin değerlendirmek için önerilen yöntem iki farklı veri seti üzerinde uygulanıp ve sonuçlar sunulmaktadır.

3.1.1. Asenkron Motorun Veri Seti

Bu çalışmada, önerilen yöntemin performansını göstermek için özellikleri Tablo 12'de verilen bir asenkron motor (Şekil 21) kullanılmıştır.



Şekil 21. Asenkron motorun yapısı

Tablo 12. Asenkron motorun özellikleri

Motor Özellikleri	Özellikler Değer
Dakikada Dönme	1800 RPM
Kutup Sayısı	4
Aşama Sayısı	3
Stator Yuvası Sayısı	24
Hava Boşluğu Uzunluğu	0.7 mm

Ses ve titreşim sinyalleri, yatak arızası, kütle dengesizliği, stator arızası ve rotor çubuğu kırık olmak üzere sağlıklı ve dört arıza koşulu altında toplanmıştır. Bu makalenin geri kalanında sırasıyla sağlık durumu ve dört arıza durumu için Class0, Class1, Class2, Class3 ve Class4 kullanılmıştır (tablo 13) [163].

Sonuçları elde etmek için önerilen yöntem, şekil 17'deki akış şemasına göre Matlab kullanılarak simüle edilmektedir. Yukarıda açıklandığı gibi, ses ve titreşim sinyalleri sistemin beş farklı durumuna göre (class0, class1,..., class4) toplanmıştır. Her sinyal türü için, verileri zaman-frekans alanına dönüştürmek için Dalgacık dönüşümü kullanılmaktadır.

Tablo 13. Sağlıklı ve dört arıza durumu sınıfları

Motor Durumları	Sınıf adı
Sağlıklı	Class0
Rulman Arızası	Class1
Kütle Dengesiz	Class2
Stator Arızaları	Class3
Kırık Rotor Çubuğu	Class4

Bu sinyaller her sınıf için geniş bir veri seti içerdiğinden, tablo 9 belirtilen bazı istatistiksel fonksiyonlardan yararlanarak verilerin özelliklerini çıkararak veriler azaltılıyor. Daha sonra bu verilerin %75'i bu çalışmada sınıflandırıcı olarak kullanılan ANN ve DSVM'yi eğitim için besler ve verilerin %25'i test için kullanılır. Yukarıda bahsedilen veri seti üzerinde ANN ve SVM sınıflandırıcı uygulamasının sonuçları tablo 14, 15, 16, ve 17'de gösterilmektedir ve şekil 22'de karşılaştırılmıştır.

Tablo 14. Titreşim özelliklerine ANN sınıflandırıcı uygulamasının sonucu

		Gerçek Sınıf				
		Class0	Class1	Class2	Class3	Class4
Tahmin Edilen Sınıf	Class0	96.6667	00.0000	03.3333	00.0000	00.0000
	Class1	00.0000	78.3333	00.0000	01.6667	20.0000
	Class2	01.6667	08.3333	61.6667	21.6667	06.6667
	Class3	00.0000	05.0000	18.3333	70.0000	06.6667
	Class4	00.0000	25.0000	05.0000	16.6667	53.3333

Tablo 15. Ses özelliklerine ANN sınıflandırıcı uygulamasının sonucu

		Gerçek Sınıf				
		Class0	Class1	Class2	Class3	Class4
Tahmin Edilen Sınıf	Class0	95.0000	00.0000	05.0000	00.0000	00.0000
	Class1	00.0000	88.3333	01.6667	00.0000	10.0000
	Class2	0s3.3333	06.6667	68.3333	10.0000	11.6667
	Class3	00.0000	01.6667	46.6667	45.0000	06.6667
	Class4	00.0000	36.6667	15.0000	00.0000	48.3333

Tablo 14, 15, 16, ve 17 ve şekil 22'de gözüktüğü gibi, bir bilgi kaynağı ve bir sınıflandırıcı kullanarak iyi bir kesinlik elde edilmemektedir. Elde edilen kesinlik seviyesine göre sistemin durumu hakkında doğru bir karar verilmesi zor ve belki imkansızdır. Bir bilgi kaynağı ve bir sınıflandırıcı kullanarak doğru karar verilememesi oldukça açıktır. Bu nedenle, karar doğruluğunun seviyesini artırmak için birden fazla bilgi kaynağı ve birden fazla sınıflandırıcı kullanılması gerekmektedir.

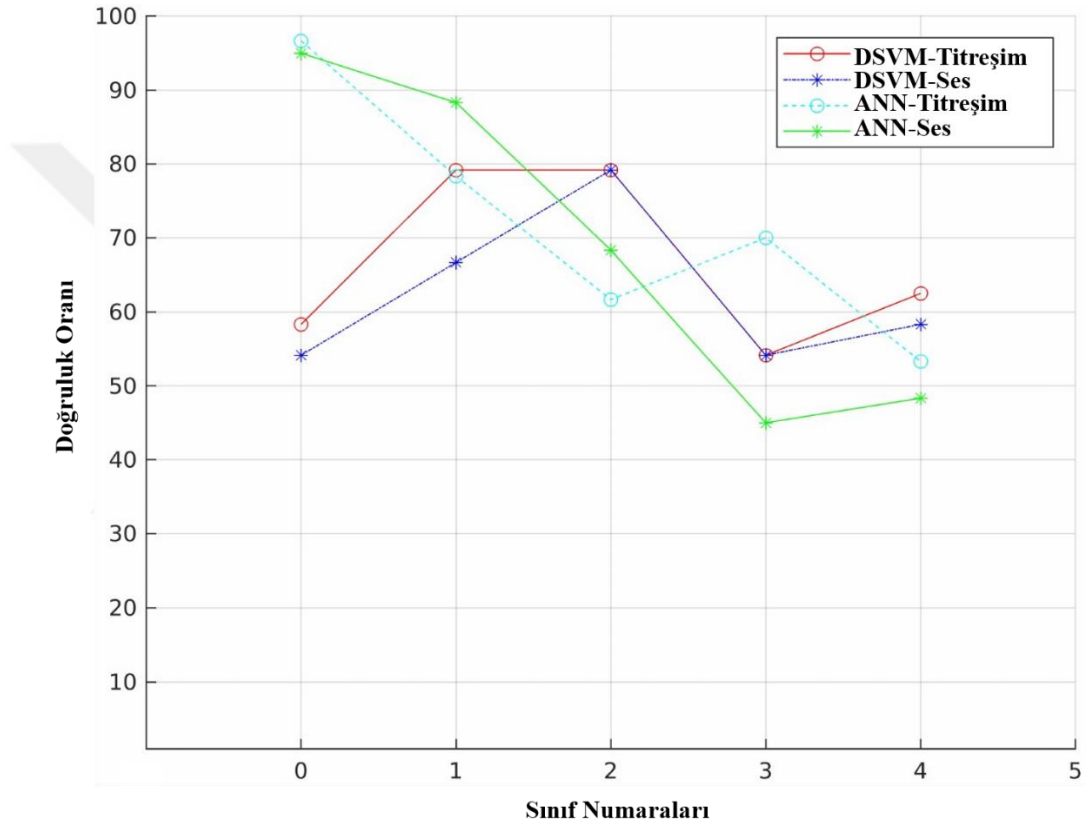
Tablo 16. Titreşim özelliklerine DSVM sınıflandırıcı uygulamasının sonucu

		Gerçek Sınıf				
		Class0	Class1	Class2	Class3	Class4
Tahmin Edilen Sınıf	Class0	58.3333	00.0000	41.6667	00.0000	00.0000
	Class1	08.3333	79.1667	04.1667	00.0000	8.3333
	Class2	04.1667	08.3333	79.1667	04.1667	04.1667
	Class3	00.0000	08.3333	33.3333	54.1667	04.1667
	Class4	00.0000	08.3333	20.8333	08.3333	62.5000

Yukarıda belirtildiği gibi, birden fazla bilgi kaynağı ve birden fazla sınıflandırıcı kullanıldığında DS veya Yager birleştirme teorisi, sınıflandırıcı sonuçlarını birleştirmek için kullanılır. Bu bölümde, önceki kısımda kullanılan ANN ve DSVM sınıflandırıcılarından elde edilen her bir sonuç çiftini birleştirmek için DS ve Yager kuralları uygulanmıştır. Sonuçlar tablo 18, 19, 20 ve 21'de sunulmakta ve bunların karşılaştırmaları Şekil 23'te gösterilmektedir.

Tablo 17. Ses özelliklerine DSVM sınıflandırıcı uygulamasının sonucu

		Gerçek Sınıf				
		Class0	Class1	Class2	Class3	Class4
Tahmin Edilen Sınıf	Class0	54.1666	00.0000	45.8333	00.0000	00.0000
	Class1	00.0000	66.6666	25.0000	00.0000	08.3333
	Class2	00.0000	08.3333	79.1667	12.5000	00.0000
	Class3	00.0000	00.0000	37.5000	54.1667	08.3333
	Class4	00.0000	20.8333	12.5000	08.3333	58.3333



Şekil 22. DSVM ve ANN sınıflandırıcıları kullanarak sınıflandırıcıların doğruluk yüzdesini karşılaştırma

Tablo 18, 19, 20 ve 21'de ve Şekil 23'de gösterildiği gibi, iki sınıflandırıcıdan gelen karar füzyonunu kullandığımızda doğruluk önceki sonuçlara kıyasla tek kaynaktan elde edilen karara göre daha iyidir. Ancak yine de kesinliğe yakın bir karar vermek için yetersizdir.

Tablo 18. Ses ve titreşim özellikleri üzerinde DSVM sınıflandırıcı için karar füzyonu uygulamasının sonucu

		Gerçek Sınıf				
		Class0	Class1	Class2	Class3	Class4
Tahmin Edilen Sınıf	Class0	62.3287	00.0000	37.6712	00.0000	00.0000
	Class1	00.0000	96.8152	01.9108	00.0000	01.2738
	Class2	00.0000	01.0869	98.0978	00.8152	00.0000
	Class3	00.0000	00.0000	29.6296	69.5473	00.8230
	Class4	00.0000	04.1841	06.2761	01.6736	87.8661

Tablo 19. Ses ve titreşim özellikleri üzerinde ANN sınıflandırıcı için karar füzyonu uygulamasının sonucu

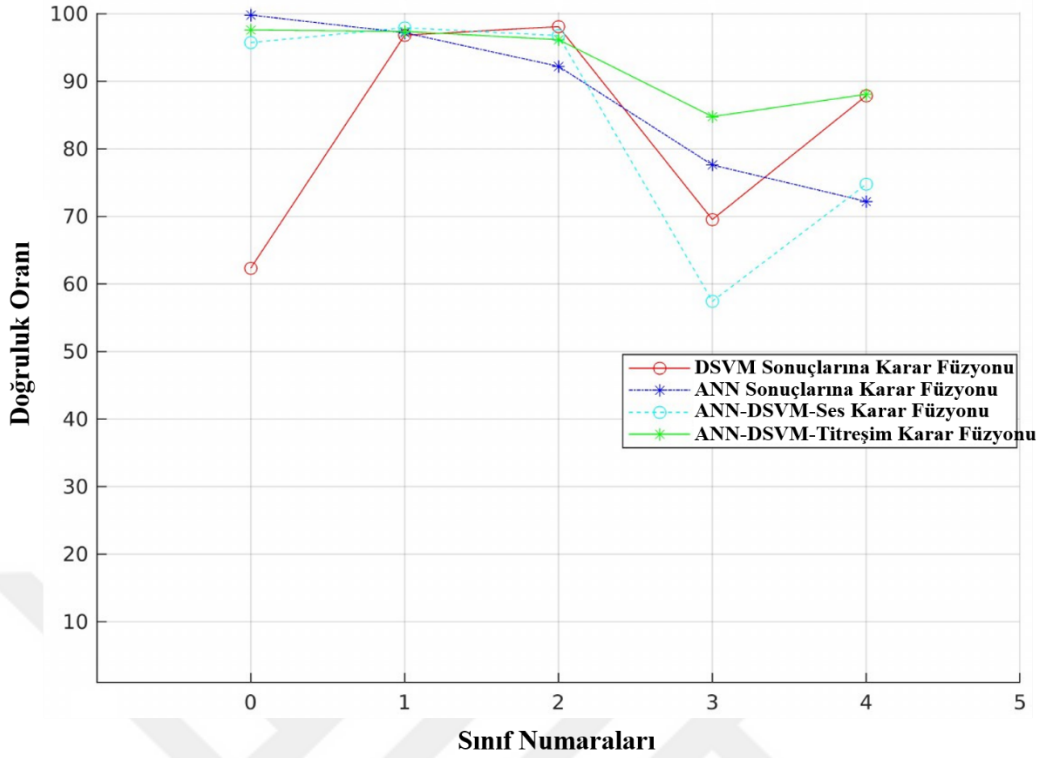
		Gerçek Sınıf				
		Class0	Class1	Class2	Class3	Class4
Tahmin Edilen Sınıf	Class0	99.8188	00.0000	00.1811	00.0000	00.0000
	Class1	00.0000	97.1907	00.0000	00.0000	02.8092
	Class2	00.1215	01.2158	92.2188	04.7416	01.7021
	Class3	00.0000	00.2053	21.0814	77.6180	01.0951
	Class4	00.0000	25.6809	2.1011	00.0000	72.2178

Tablo 20. Ses için DSVM ve ANN sınıflandırıcılarından gelen karar füzyon uygulamasının sonucu

		Gerçek Sınıf				
		Class0	Class1	Class2	Class3	Class4
Tahmin Edilen Sınıf	Class0	95.7364	00.0000	04.2635	00.0000	00.0000
	Class1	00.0000	97.9214	00.6928	00.0000	01.3856
	Class2	00.0000	00.9937	96.7701	02.2360	00.0000
	Class3	00.0000	00.0000	47.2795	51.2195	01.5009
	Class4	00.0000	20.2578	04.9723	00.0000	74.7697

Tablo 21. Titreşim için DSVM ve ANN sınıflandırıcılarından gelen Karar Füzyon uygulamasının sonucu

		Gerçek Sınıf				
		Class0	Class1	Class2	Class3	Class4
Tahmin Edilen Sınıf	Class0	97.5961	00.0000	02.4038	00.0000	00.0000
	Class1	00.0000	97.3827	00.0000	00.0000	02.6172
	Class2	00.1367	01.3679	96.1696	01.7783	00.5471
	Class3	00.0000	00.9316	13.6645	84.7826	00.6211
	Class4	00.0000	05.5045	02.7522	03.6697	88.0733



Şekil 23. DSVM ve ANN sınıflandırıcı sonuçlarının karar füzyonu için doğruluk yüzdesinin karşılaştırılması

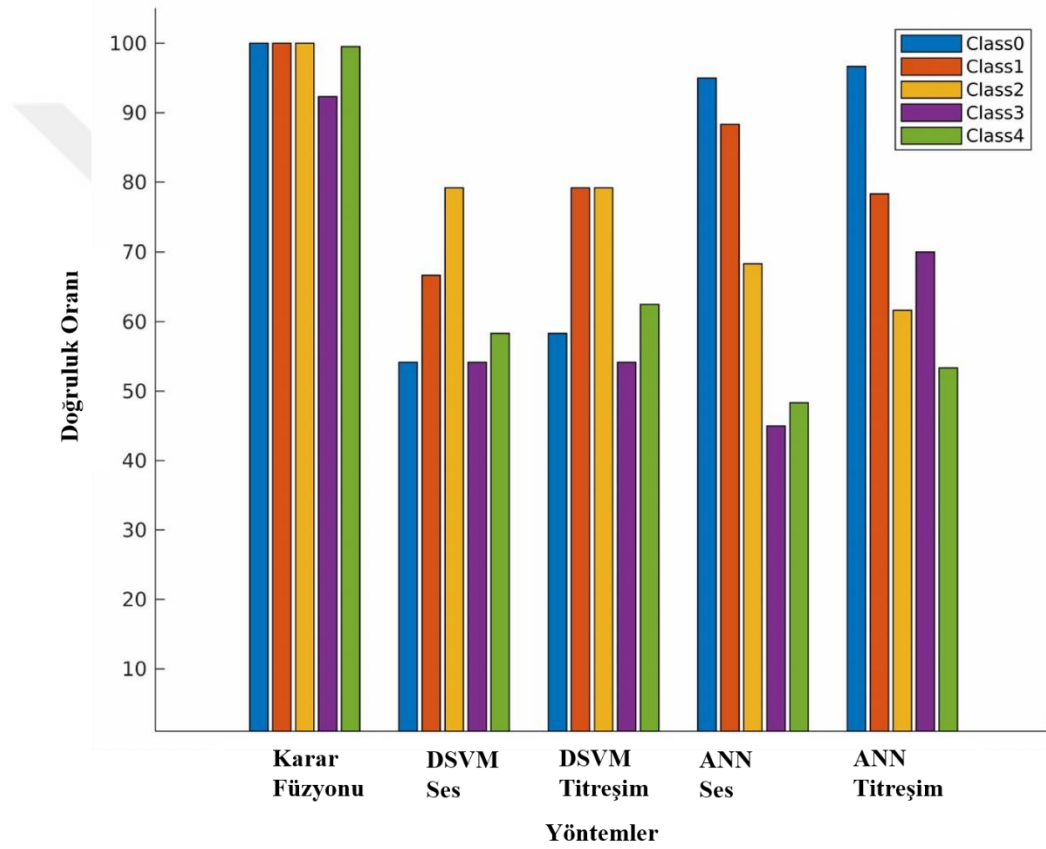
Bu nedenle, bir sonraki adımda, sınıflandırıcıların tüm sonuçlarına karar füzyonu uygulanmaktadır. Tablo 22 iki sınıflandırıcının tüm sonuçlarına karar füzyonu uygulanmasının sonuçlarını göstermektedir. Bu tabloda gözüktüğü gibi önerilen yöntem karar vermek için iyi bir doğruluk seviyesi sunmaktadır. Elde edilen sonuçlara göre, birden fazla veri kaynağı, birden fazla sınıflandırıcı ve kanıt kullanmanın, doğruluğu kabul edilebilir bir düzeye (Ortalama doğruluk değeri 98.3705) çıkardığı oldukça açıktır. Dolayısıyla önerilen yöntem, bir veri kaynağında yeterli bilginin olmadığı durumlarda kullanılabileceği gibi, birden fazla veri kaynağının olduğu durumlarda da kullanılabilir.

Tablo 22. İki Sınıflandırıcının tüm sonuçlarına Karar Füzyon uygulamasının sonucu

		Gerçek Sınıf				
		Class0	Class1	Class2	Class3	Class4
Tahmin Edilen Sınıf	Class0	99.9947	00.0000	00.0052	00.0000	00.0000
	Class1	00.0000	99.9078	00.0000	00.0000	00.0921
	Class2	00.0000	00.0186	99.6969	00.2843	00.0000
	Class3	00.0000	00.0000	05.9532	94.0217	00.0249
	Class4	00.0000	05.6155	00.0551	00.0000	94.3292

Tablo 23. DSVM ve ANN sınıflandırıcının ses ve titreşim üzerindeki doğruluğu ve tüm sınıflandırıcıların sonuçlarına karar füzyonu uygulaması

	Class0 doğruluk	Class1 doğruluk	Class2 doğruluk	Class3 doğruluk	Class4 doğruluk	Ortalama doğruluk
Karar füzyonu	99.9951	99.9994	99.9988	92.3292	99.5296	98.3705
DSVM ses	54.1666	66.6666	79.1666	54.1666	58.3333	62.5000
DSVM titreşim	58.3333	79.1666	79.1666	54.1666	62.5000	66.6667
ANN ses	95.0000	88.3333	68.3333	45.0000	48.3333	69.0000
ANN titreşim	96.6666	78.3333	61.6666	70.0000	53.3333	72.0000



Şekil 24. DSVM ve ANN sınıflandırıcılarının ses ve titreşim üzerindeki doğruluklarının ve tüm sınıflandırıcıların sonuçlarına karar füzyonunun uygulanmasının grafiksel karşılaştırması.

Önerilen yöntemin performansını vurgulamak için yukarıda belirtilen tüm durumlar önerilen yöntemle karşılaştırılmıştır. Bu karşılaştırma sonucu Tablo 23 ve şekil 24'de sunulmuştur. Görüldüğü gibi, karar füzyonu, beş sınıfın tümü için yüksek doğruluğa sahiptir.

3.1.2. Çok Sensörlü Veri Füzyonuna Dayalı Aktivite Tanıma Veri Seti (AReM)

Bu bölüm, önerilen yöntemin daha fazla performans değerlendirmesi için AReM veri seti [164] üzerinde uygulanıp sonuçları sunmaktadır. Bu veri seti, sistem aktörlerinden giyilebilir sensörler ile toplanan bisiklete binme, yatma, oturma, ayakta durma ve yürüme aktivitelerinden Alınan Sinyal Gücü (Received Signal Strength, RSS) verilerini içerir. Bu aktiviteler bundan sonra class1, class2, class3, class4, ve class5 olarak gösterilmiştir. Bu veri setindeki her aktivite için 15 zamansal dizi mevcuttur. Her dizi 480 örnek içerir ve sistem aktörleri tarafından giyilen üç çift sensör düğümü (göğüs-sağ ayak bileği, göğüs-sol ayak bileği, sağ ayak bileği-sol ayak bileği) tarafından toplanır. Bu bölümde, önerilen algoritma Şekil 17'de gösterilen akış şemasına göre AReM veri seti üzerinde uygulanmakta ve sonuçlar görüntülenmektedir. Tüm sensörlerin verileri ANN ve DSVM algoritmalarına verilir. Bu sınıflandırıcılardan elde edilen sonuçlar Dempster-Shafer veya Yager kuralı kullanılarak birleştirilir ve nihai çıkarım elde edilir.

Tablo 24 ve 25, DSVM ve ANN sınıflandırıcıları kullanılarak insan aktivitesi tespiti için önerilen yöntemin sonuçlarını göstermektedir. Bu sonuçları elde etmek için öncelikle beş aktörden giyilen algılayıcıdan elde edilen RSS verileri bahsedilen sınıflandırıcılara verilir ve ardından elde edilen sonuçlar birleştirilir.

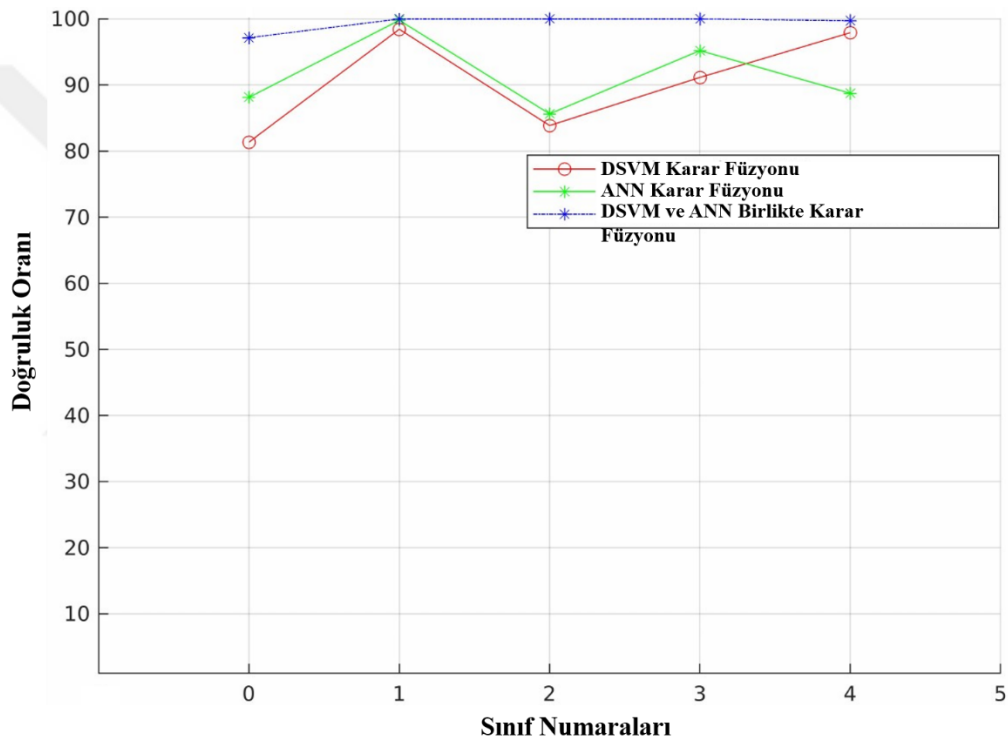
Tablo 24. DSVM sınıflandırıcı için karar füzyonu sonucu

		Gerçek Sınıf				
		Class0	Class1	Class2	Class3	Class4
Tahmin Edilen Sınıf	Class0	81.3541	00.0000	00.0000	00.0000	18.6458
	Class1	00.0000	98.4375	00.0000	00.0000	01.5625
	Class2	00.5208	00.0000	83.8541	00.0000	15.6250
	Class3	00.0000	00.0000	00.0000	91.1458	08.8541
	Class4	02.0833	00.0000	00.0000	00.0000	97.9166

Tablo 25. ANN sınıflandırıcı için karar füzyonu sonucu

		Gerçek Sınıf				
		Class0	Class1	Class2	Class3	Class4
Tahmin Edilen Sınıf	Class0	88.1250	00.0000	00.4166	00.2083	11.2500
	Class1	00.0000	99.7916	00.0000	00.0000	00.2083
	Class2	00.6250	00.0000	85.6250	13.7500	00.0000
	Class3	00.0000	00.0000	04.7916	95.2083	00.0000
	Class4	11.2500	00.0000	00.0000	00.0000	88.7500

Tablolarda görüldüğü gibi, DSVM ve ANN sınıflandırıcılardan ayrı ayrı sonuçların birleştirilmesiyle, ortalama doğruluklar sırasıyla 90.54179 ve 91.5000'dir ve bu, karar verme için kabul edilebilir bir doğruluk seviyesi değildir. Tablo 26, DSVM ve ANN sonuçlarının önerilen yöntemi ile karşılaştırmasını göstermektedir. Görüldüğü gibi, iki sınıflandırıcının sonuçları birleştirildiğinde, karar verme için kabul edilebilir 99.3761 ortalama doğruluk seviyesi elde edilmektedir. Bu tablodaki sonuçları vurgulamak için, Şekil 25'de karşılaştırılmıştır ve önerilen yöntemin doğruluğu ile kullanılan iki sınıflandırıcı oldukça yüksektir.

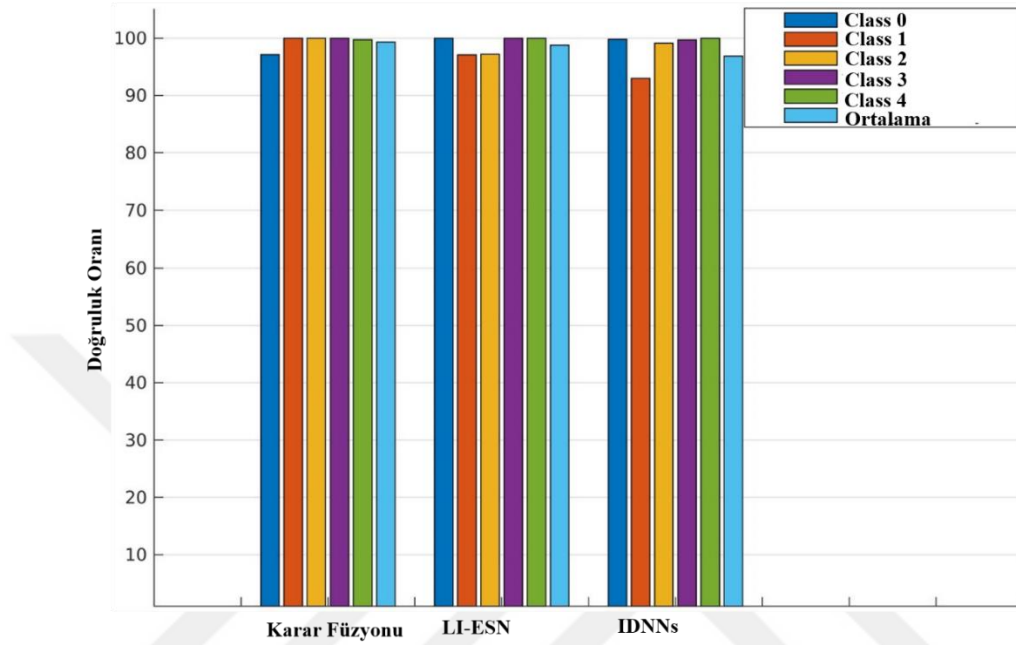


Şekil 25. DSVM ve ANN sınıflandırıcıları için birlikte karar füzyonu sonucu

Tablo 26. DSVM ve ANN sınıflandırıcıları için birlikte karar füzyonu sonucu

		Gerçek Sınıf				
		Class0	Class1	Class2	Class3	Class4
Tahmin Edilen Sınıf	Class0	97.1573	00.0000	00.0000	00.0000	02.8426
	Class1	00.0000	99.9966	00.0000	00.0000	00.0033
	Class2	00.0045	00.0000	99.9954	00.0000	00.0000
	Class3	00.0000	00.0000	00.0000	100.0000	00.0000
	Class4	00.2689	00.0000	00.0000	00.0000	99.7310

Tablo 27 ve Şekil 26, önerilen yöntem ile bu veri seti için [10]'da önerilen Leaky Integrator Echo State Networks (LI-ESN) ve Input Delay Neural Networks (IDNNs) yöntemlerinin karşılaştırmasını göstermektedir.



Şekil 26. Önerilen, LI-ESN ve IDNNs yöntemleri için doğruluk karşılaştırılması

Görüldüğü gibi önerilen yöntem karşılaştırılan iki yöntemden daha iyi performans göstermektedir ve sonuçlar farklı sınıflar için birbirine yakındır. Önerilen yöntem ile bu veri seti için ortalama 99.3 doğruluk elde edilirken karşılaştırılan iki yöntem aynı veri seti için ortalama 98.8 ve 96.90 doğruluk elde etmişlerdir. ve buda önerilen yöntemin iyi olduğunu açıkça göstermektedir.

Tablo 27. Önerilen, LI-ESN ve IDNNs yöntemleri için doğruluk karşılaştırılması

	Doğruluk					
	Class0	Class1	Class2	Class3	Class4	Ortalama
Karar füzyonu	97.1573	99.9966	99.9954	100.0000	99.7310	99.3000
LI-ESN	100.0000	97.1000	97.2000	100.0000	100.0000	98.80
IDNNs	99.8000	93.0000	99.1000	99.7000	100.0000	96.90

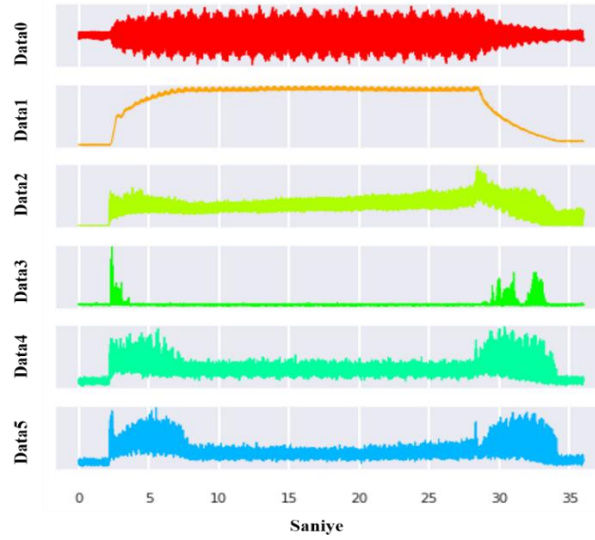
3.2. İkinci Önerilen Yöntem

Önerilen yöntemi değerlendirmek için bilgi birleştirme yöntemlerini değerlendirmeye uygun ve çeşitli çok sensörlü verileri içeren Mill veri seti kullanılmıştır. Bu veri seti, anomali tespit yöntemleri için uygundur.

3.2.1. Mill Veri Seti

Bu veri seti, bir CNC makinesinde farklı çalışma koşullarında gerçekleştirilen çeşitli testlerden elde edilen verileri içerir. Bu araçtaki aşınma anomalilerini araştırmak için veri seti derlenmiştir ve veriler akım, ses ve titreşim sensörleri kullanılarak kaydedilmiştir [8, 165].

CNC makinesi, hassas boyutlar ve pürüzsüz bir yüzey ile yüksek kaliteli bir işlem gerçekleştirmek için keskin bir alet kullanır. Takımın aşınması yüzeyi deforme edebilir ve ortaya çıkan sürtünme sıcaklığı, kusurlu bir takımda istenmeyen etkilere neden olabilir. Bu nedenle takım aşınması, hasar oluşmadan önce uygun zamanda tespit edilmesi ve kontrol edilmesi gereken anormal bir durumdur [8].



Şekil 27. Veri setinin 147. çalışması

Tablo 28. Mill veri seti özellikleri

Alan Adı	Açıklama	Sembol
<i>smcAC</i>	AC iş mili motor akımı	Data0
<i>smsDC</i>	DC iş mili motor akımı	Data1
<i>vib_spindle</i>	Mil titreşimi	Data2
<i>vib table</i>	Masa titreşimi	Data3
<i>AE spindle</i>	Milde ses emisyon	Data4
<i>AE table</i>	Masada ses emisyon	Data5

Bu veri seti, sağlıklı, hasarlı ve arıza olmak üzere üç duruma atıfta bulunan 0, 1 ve 2 olarak etiketlenmiş 167 çalıştırma içerir. Veriler rasgele eğitim, doğrulama ve test alt kümelerine bölünür. Tablo 28, veri setindeki veri özelliklerini göstermektedir. Şekil 27. veri setinin 147 inci çalışmasını göstermektedir.

3.2.2. Değerlendirme Kriterleri

Modelleri değerlendirmek için doğruluk (accuracy), kesinlik (precision) ve duyarlılık (recall) kriterleri kullanılmıştır. Bu kriterler (39) ile (41) arasındaki denklemler ile ölçülür [161].

$$\text{Accuracy} = \frac{TP+TN}{TP+FN+TN+FP} \quad (39)$$

$$\text{Precision} = \frac{TP}{TP+FP} \quad (40)$$

$$\text{Recall} = \frac{TP}{TP+FN} \quad (41)$$

Bu denklemlerde adı geçen TP, TN, FP ve FN değerleri sırasıyla doğru-pozitif, doğru-negatif, yanlış-pozitif ve yanlış-negatif anlamına gelmektedir. Doğruluk, test verileri için doğru tahminlerin yüzdesi olarak tanımlanır. Duyarlılık, model tarafından doğru olarak tanımlanan gerçek pozitif etiketlerinin oranını ölçer. Kesinlik ise bir makine öğrenimi modelinin yaptığı olumlu bir tahminin kalitesidir.

3.2.3. Önerilen Yöntemin Simülasyonu ve Sonuçları

DS-1DCNN'yi uygulamak için TensorFlow [166] açık kaynak platformunu temel alan Keras [167] kütüphanesi kullanılmıştır. Keras, derin öğrenme modellerinin hızlı bir şekilde uygulanmasını sağlayan açık kaynaklı bir Python kütüphanesidir. Ayrıca simülasyon için 8 çekirdekli Intel i7 CPU, 16 GB RAM, 256 GB SSD sabit disk ve NVIDIA GeForce GTX 1060 Ti ekran kartına sahip bir bilgisayar kullanılmıştır. Keras, işlevsel API'yi aracılığıyla birden çok girişi ve hatta birden çok çıkışı yönetebildiğinden, sınıflandırmaları GPU üzerinde paralel olarak uygulamak için Keras işlevsel API'yi kullanılmıştır. Modelin eğitimi, değerlendirilmesi ve test edilmesi için tüm veriler üç bölüme ayrılır: eğitim, değerlendirme ve test verileri. Tüm verilerden eğitim, değerlendirme ve test verilerinin seçimi, varsayımların olmaması ve sonuçların oldukça karşılaştırılabilir olması için tamamen rastgele yapıldı.

DS-1DCNN'yi değerlendirmek için akım, ses ve titreşim verileri 1DCNN ile ayrı ayrı sınıflandırılır. Daha sonra, yukarıda bahsedilen üç veri tipi veri seviyesinde birleştirilmiş ve 1DCNN kullanılarak sınıflandırılmıştır. Son olarak belirtilen üç veri tipi önerilen model kullanılarak sınıflandırılmış ve sonuçlar karşılaştırılmıştır. Bu deneyde farklı modeller tam olarak eşit koşullarda eğitilmektedir. Deneyde kullanılan parametrelerden bazıları 0.0001 eğitim oranı, 100 tekrar sayısı, Adam optimizasyon algoritması ve modellerin tüm katmanlarının eğitimidir.

Tablo 29, 30 ve 31 sırasıyla akım, ses ve titreşim verileri için karışıklık matrisi (Confusion matrix) sonuçlarını göstermektedir ve bu sinyallerin ortalama doğrulukları sırasıyla 0,82, 0,75 ve 0,80'dir. Bu verilerden herhangi biri karar vermek için yetersiz olduğundan kesin bir sonuç ve nihai karar için gerekli verimlilik elde edilememiştir.

Tablo 29. 1DCNN kullanılarak sınıflandırılan akım verilerin karışıklık matrisi

		Gerçek Sınıf		
		Sağlıklı	Hasarlı	Arızalı
Tahmin Edilen Sınıf	Sağlıklı	0.8366	0.1565	0.0069
	Hasarlı	0.1032	0.8259	0.0709
	Arızalı	0.0136	0.2458	0.7406

Bu sonuçlara göre, bazı durumlarda endüstriyel karar verme için bir sensörün verilerinin yetersiz olduğu ve doğru karar vermek için birkaç sensörün verilerinin aynı anda kullanılması gerektiği oldukça açıktır. Bu nedenle, bir sonraki adımda, bu üç tip sensörün verilerine, veri düzeyinde füzyon yapılır ve 1DCNN, LSTM ve RF yöntemleri kullanılarak sınıflandırılır. Daha sonra, elde edilen sonuçlar Tablo 32, 33 ve 34'de gösterilmiştir.

Tablo 30. 1DCNN kullanılarak sınıflandırılan ses verilerin karışıklık matrisi

		Gerçek Sınıf		
		Sağlıklı	Hasarlı	Arızalı
Tahmin Edilen Sınıf	Sağlıklı	0.7103	0.2682	0.0215
	Hasarlı	0.1177	0.7805	0.1018
	Arızalı	0.0254	0.2836	0.6910

Tablo 31. 1DCNN kullanılarak sınıflandırılan titreşim verilerin karışıklık matrisi

		Gerçek Sınıf		
		Sağlıklı	Hasarlı	Arızalı
Tahmin Edilen Sınıf	Sağlıklı	0.8406	0.1452	0.0141
	Hasarlı	0.1136	0.7902	0.0962
	Arızalı	0.0278	0.2896	0.6827

Tablo 32. Veri düzeyinde füzyon ve 1DCNN kullanılarak sınıflandırılan akım, ses ve titreşimin karışıklık matrisi

		Gerçek Sınıf		
		Sağlıklı	Hasarlı	Arızalı
Tahmin Edilen Sınıf	Sağlıklı	0.9051	0.0907	0.0042
	Hasarlı	0.0544	0.8965	0.0491
	Arızalı	0.0100	0.1890	0.8010

1DCNN, LSTM ve RF yöntemleri için 0.892, 0.908 ve 0.871 ortalama doğrulukları bu adımda elde edilir. Tablo 29-34'e göre, veri düzeyinde füzyon kullanılan sonuçların tek tek sensörlerin sonuçlarından daha iyi olduğu ancak yine de kesin ve doğru bir karar için yeterli olmadığı anlaşılabılır. Öte yandan, farklı sensörlerin verileri uyumlu değilse, veri düzeyinde füzyon olasılığı olmayacaktır.

Tablo 33. Veri düzeyinde füzyon ve LSTM kullanılarak sınıflandırılan akım, ses ve titreşimin karışıklık matrisi

		Gerçek Sınıf		
		Sağlıklı	Hasarlı	Arızalı
Tahmin Edilen Sınıf	Sağlıklı	0.9270	0.0665	0.0063
	Hasarlı	0.0530	0.9083	0.0386
	Arızalı	0.0085	0.1778	0.8136

Son olarak, önceki bölümde açıklanan önerilen yöntem veri setine uygulanmıştır. DS-1DCNN'nin karışıklık matrisi Tablo 35'da gösterilmektedir. Elde edilen sonuçlara göre, DS-1DCNN endüstriyel anomalide karar verme için gerekli doğruluğa ve ortalama %99 doğruluk oranına sahiptir. Tablo 29-35, her üç veri tipine ayrı ayrı 1DCNN uygulayarak ve tüm verilerin veri seviyesi kombinasyonuna 1DCNN, LSTM ve RF uygulanarak önerilen yöntemin sonuçlarını özetlemektedir. Bu sonuçlara göre DS-1DCNN diğer tekniklere göre daha verimlidir. Şekil 28 ve Tablo 36, i) yukarıda bahsedilen tüm sinyaller için 1DCNN, 1DCNN_CURRENT, 1DCNN_SOUND ve 1DCNN_VIBRATION olarak gösterilen duyarlılık karşılaştırma parametresini göstermektedir, ii) tüm sinyallerin veri seviyesi füzyonu için 1DCNN, LSTM ve RF, 1DCNN_DLFUSION, LSTM ve RF olarak gösterilen, iii) DS-1DCNN. Geri çağırma, veri setindeki tüm pozitif örneklerden yapılan pozitif sınıf tahminlerinin sayısını gösterir. Örneğin, bu metrik sağlık durumundaki toplam örnek sayısını temsil eder ve gerçekte kaç örnek sağlık durumunda olduğu sorusunu yanıtlar. Şekil 28 ve Tablo 36'da görüldüğü gibi önerilen yöntemin diğerlerinden daha yüksek duyarlılık puanına sahip olduğu açıktır. DS-1DCNN, uygun ve kabul edilebilir bir sonuç olarak ortalama 0.9764 geri çağırma elde etmiştir.

Tablo 34. Veri düzeyinde füzyon ve RF kullanılarak sınıflandırılan akım, ses ve titreşimin karışıklık matrisi

		Gerçek Sınıf		
		Sağlıklı	Hasarlı	Arızalı
Tahmin Edilen Sınıf	Sağlıklı	0.9087	0.0906	0.0007
	Hasarlı	0.0891	0.8684	0.0424
	Arızalı	0.0130	0.2943	0.6927

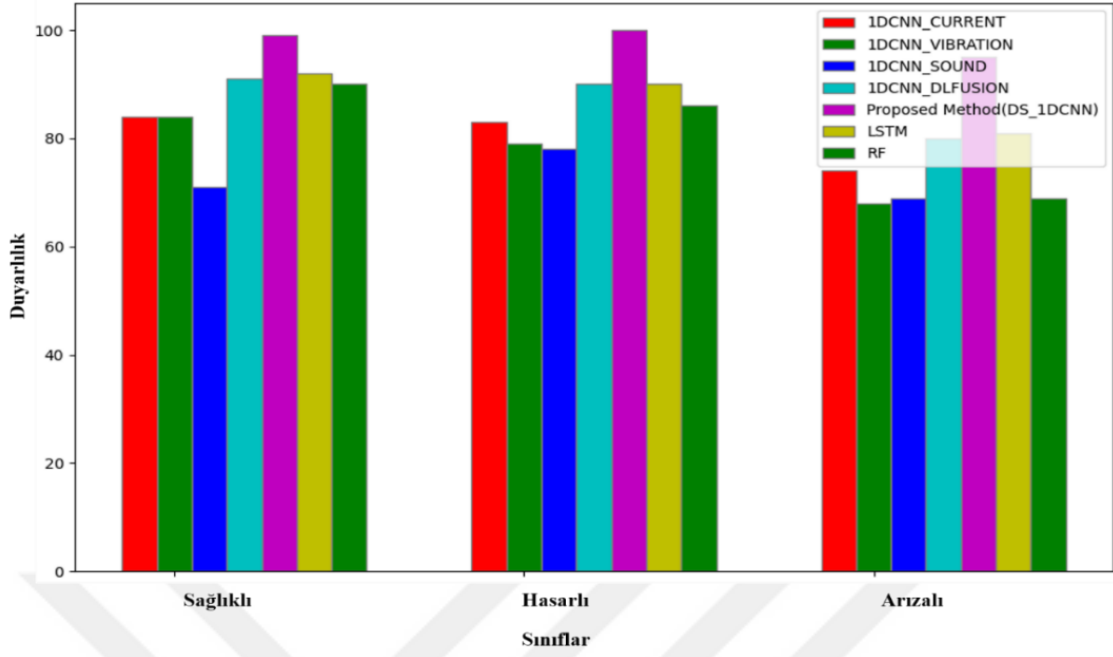
Tablo 35. DS-1DCNN'nin karışıklık matrisi

		Gerçek Sınıf		
		Sağlıklı	Hasarlı	Arızalı
Tahmin Edilen Sınıf	Sağlıklı	0.9879	0.0121	0.0000
	Hasarlı	0.0027	0.9959	0.0014
	Arızalı	0.0000	0.0546	0.9453

Şekil 29 ve Tablo 37, i) yukarıda bahsedilen tüm sinyaller için 1DCNN'nin, ii) tüm sinyallerin veri seviyesi füzyonu için 1DCNN, LSTM ve RF'nin ve iii) DS-1DCNN'nin kesinlik karşılaştırmasını göstermektedir. Kesinlik parametresi, pozitif sınıf örneklerinin toplam sayısından pozitif sınıf tahminlerinin sayısını ifade eder. Örneğin, bu metrik, sağlıklı olarak etiketlenen toplam numune sayısını gösterir ve gerçekte kaç numunenin sağlık durumunda olduğu sorusuna yanıt verir. Yüksek kesinlik, düşük yanlış pozitif oranı ile ilgilidir. Şekil 29 ve tablo 37'de görüldüğü gibi, önerilen yöntemin diğerlerinden daha yüksek bir kesinlik puanına sahip olduğu açıktır. DS-1DCNN, endüstriyel anomali tespit sistemleri için uygun ve kabul edilebilir bir puan olan 0.9895 ortalama kesinlik puanına sahiptir.

Tablo 36. 1DCNN-CURRENT, 1DCNN-SOUND, 1DCNN-VIBRATION, 1DCNN-DIFFUSION, LSTM, RF ve DS-1DCNN'nin duyarlılık karşılaştırması

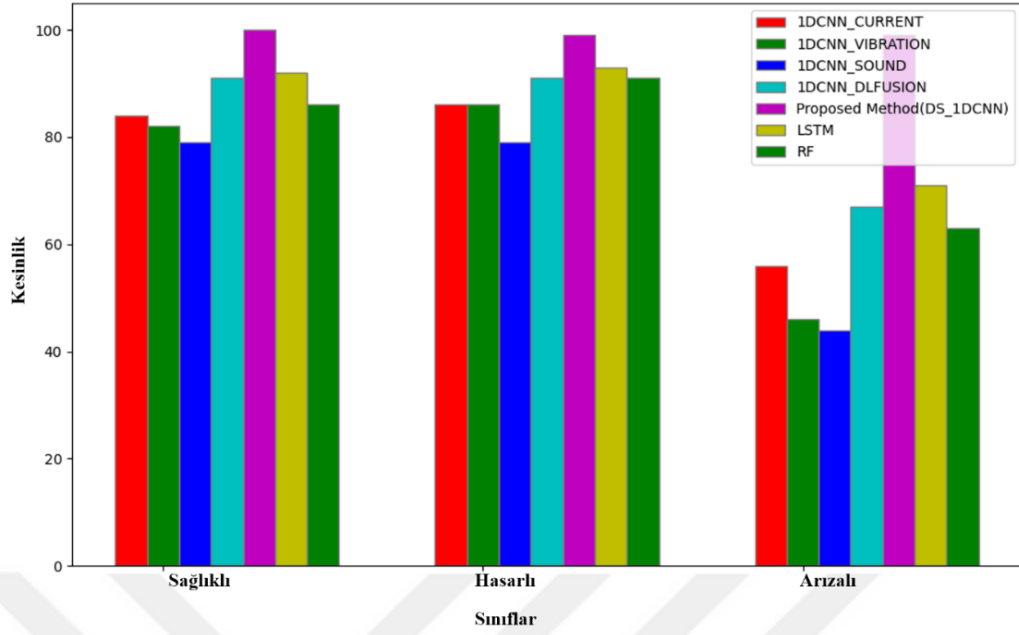
		Sınıflar (Etiketler)		
		Sağlıklı	Hasarlı	Arızalı
Yöntemler	1DCNN_CURRENT	0.8366	0.8259	0.7406
	1DCNN_SOUND	0.7103	0.7805	0.6910
	1DCNN_VIBRATION	0.8406	0.7902	0.6827
	1DCNN_DLFUSION	0.9051	0.8965	0.8010
	Proposed Method (DS-1DCNN)	0.9879	0.9959	0.9454
	LSTM	0.9271	0.9083	0.8136
	RF	0.9087	0.8684	0.6927



Şekil 28. 1DCNN_CURRENT, 1DCNN_SOUND, 1DCNN_VIBRATION, 1DCNN_DLFUSION ve DS-1DCNN'nin duyarlılık karşılaştırması

Tablo 37. 1DCNN_CURRENT, 1DCNN_SOUND, 1DCNN_VIBRATION, 1DCNN_DLFUSION ve DS-1DCNN'nin kesinlik karşılaştırması

		Sınıflar (Etiketler)		
		Sağlıklı	Hasarlı	Arızalı
Yöntemler	1DCNN_CURRENT	0.8383	0.8607	0.5599
	1DCNN_SOUND	0.7925	0.787	0.436
	1DCNN_VIBRATION	0.8235	0.8569	0.4564
	1DCNN_DLFUSION	0.9135	0.9148	0.6669
	Proposed Method (DS-1DCNN)	0.9958	0.9852	0.9887
	LSTM	0.9177	0.9319	0.7111
	RF	0.8554	0.9110	0.6341



Şekil 29. 1DCNN_CURRENT, 1DCNN_SOUND, 1DCNN_VIBRATION, 1DCNN_DLFUSION ve DS-1DCNN'nin kesinlik karşılaştırması

Her titreşim, ses ve akım verilerinin yeterli bilgiye sahip olmaması nedeniyle bu verilerden ayrı ayrı elde edilen sonuçlar doğru karar vermek için yeterli olmamaktadır. Sunulan sonuçlarda bu oldukça açıktır ve yapılan tahminler karar vermek için yeterli doğrulukta değildir. Bununla birlikte, DS-1DCNN, sonuçların belirsizliğini ortadan kaldırmak için verimli bir modeldir ve endüstriyel uygulamalarda karar verme için güvenilir ve kabul edilebilir sonuçlar sağlar.

4. SONUÇLAR

Nesnelerin İnterneti'nin endüstrideki uygulamasına Endüstriyel Nesnelerin İnterneti (Industrial Internet of Things, IIoT) denir. IIoT, çok daha büyük miktarda veriye çok daha yüksek hız ve üretkenlikle erişim sağlayarak üretimde süreçlerinde devrim yaratıyor. IIoT, endüstriyel ortamlardan büyük miktarda bilgi toplayan ve paylaşan akıllı bilgisayarlar, cihazlar ve nesnelerin ağıdır. Toplanan veriler merkezi bir sisteme gönderilir ve karar vermek için analiz edilir ve diğer verilerle birleştirildikten sonra son kullanıcı ile paylaşılır.

Bu tezde IIoT'de karar vermek için iki yöntem geliştirilmiştir. Bu yöntemler çoklu sensörlü ortamlar için önerilmişlerdir. Bu tür ortamlarda genelde birden fazla sınıflandırıcı farklı sensörlerden gelen veriler için kullanılır. Farklı sınıflandırıcıdan gelen kararları birleştirme için karar füzyonu kullanılıyor. Bu tezde her iki yöntemde de karar füzyonu için DS birleştirme kuralları kullanılmıştır. Sınıflandırıcı olarak ANN ve DSVM kullanılan ilk yöntem orta ve küçük boyutlu IIoT ler için uygundur. İkinci yöntemde büyük verileri analiz edilebilmesi için DL yöntemleri kullanılmıştır. Bu yöntemde karmaşıklığın az olması ve hızın yüksek olması için parametresi sayısı az olan 1DCNN modeli kullanılmıştır. CNN güçlü DL yöntemlerinden biridir, ki bu yöntemin iki boyutlusu genelde görüntü işleme gibi daha kompleks işlemler için kullanılmaktadır. Ama bir boyutlu CNN son zamanlarda sinyal işleme ve tek boyutlu verileri analiz etmek için etkin ve hızlı bir yöntem olarak kullanılmaktadır. O yüzden ikinci yöntemde verilerin analizi ve sınıflandırması için 1DCNN modelinin kullanılması tercih edilmiştir

Birinci önerilen yöntemde, küçük Endüstriyel Nesnelerin İnterneti (IIoT) için Uluslararası Telekomünikasyon Birliği (ITU) modeline dayalı bir mimari tanıtılmaktadır. Servis ve uygulama destek katmanında, veri analizi ve sağlıklı karar vermek için Dempster Shafer adlı karar füzyonu yöntemlerinden biri kullanılmaktadır. Karar füzyonu kullanmadan önce, verileri sınıflandırmak için farklı veri sınıflandırıcıları kullanılmıştır. Bu tezde, bahsi geçen amaçları gerçekleştirmek için DSVM ve ANN kullanılmıştır. Asenkron motordan toplanan veriler, yöntemin verimliliğini kanıtlamak için kullanılır. Bir önceki bölümde gösterildiği gibi, tek bir bilgi kaynağı ve hatta tek bir sınıflandırıcı ile kesin bir karar için amaçlanan doğruluk elde edilemez. Bu nedenle, bu çalışmada iki bilgi kaynağı ve iki sınıflandırıcı kullanılmıştır. Bir önceki bölümde yapılan karşılaştırmalara göre, birçok kaynaktan elde edilen bilgiler birkaç sınıflandırıcı kullanılarak değerlendirildiğinde ve bu

sınıflandırıcıların sonuçları DS karar füzyonu yöntemini kullanarak birleştirildiğinde karar vermede yüksek doğruluk elde edildiği görülmüştür. Bu algoritmanın etkinliği iki kullanım durumu göz önünde bulundurularak incelenmiştir. Sonuçlar bölümünde görüldüğü gibi, önerilen yöntem kullanılarak birinci ve ikinci vaka çalışmaları için sırasıyla yüzde 98.3705 ve yüzde 99.3761 ortalama doğruluk değerleri elde edilmiştir. Bu yöntemin operasyonel ve gerçek dünya uygulamalarında kullanılabilmesi için bu algoritmanın MATLAB yerine gerçek zamanlı programlamaya uygun bir programlama dili kullanılarak gerçekleştirilmesi gerekmektedir. Bu gereklilik hizmetlerde karar vermeye ve küçük IIOT'nin uygulama destek katmanında kesin kararlar vermek ve olası hataları ve endüstriyel birimlerin yüksek kayıplarını önlemek için doğru eylemi uygulamak için yardımcı olabilir, çünkü bu tür birimlerde bir parçanın arızalanması, sistemin bütününe birçok kayıplar getirebilir.

İkinci yöntemde, büyük hacimde verisi olan IIoT sistemlerindeki anormalliklere karar vermek için etkili bir yöntem önerilmektedir. Bu bağlamda, 1DCNN modeli endüstriyel bir cihazın sağlıklı, hasarlı ve arıza durumlarını sınıflandırmak için kullanılmıştır. Özet olarak bu tezde IIoT karar verme sürecindeki belirsizliği çözmek için 1DCNN algoritmasını ve Dempster Shafer birleştirme kurallarını (DS-1DCNN) kullanan yeni bir yaklaşım önerilmiştir. Ayrıca, DS-1DCNN'nin değerlendirilmesi için, 1DCNN algoritması, akım, ses ve titreşimin her üç veri tipini ayrı ayrı sınıflandırmak için kullanılmış, daha sonra bu üç veri tipinin füzyonunu sınıflandırmak için kullanılmış ve sonuç olarak DS-1DCNN ile karşılaştırılmıştır. Sonuçlara göre her bir sensörün verilerinden tek tek elde edilen kararların IIoT'de karar vermek için yeterli olmadığı görülmüştür. Ancak, bu üç veri türünün veri düzeyinde birleştirilmesiyle eğitilen modelin iyi bir karar doğruluğuna sahip olduğu görülmüştür. .

Ancak, sistem endüstri ortamında karar vermek için hâlâ yeterli kesinliğe sahip değildir. Bu nedenle karar verme sürecindeki mevcut belirsizliği ortadan kaldırmak için DS-1DCNN yöntemi önerilmiştir. Önerilen yöntemde, verileri sınıflandırmak için 1DCNN ve verilen kararın kesin ve kabul edilebilir doğruluğa ulaşması ve sınıflandırıcılardan alınan kararları birleştirmek için DS birleştirme kuralları kullanılmıştır. Önerilen yöntem ve diğer ilgili algoritmalar, GPU üzerinde Python'un Keras kütüphanesi kullanılarak simüle edilmiştir. DS-1DCNN, 0.9763 ortalama duyarlılık ve 0.9899 ortalama kesinlik değerlerine ulaşmıştır ve bu sonuçlar karar verme için kabul edilebilir ve güvenilir bir sonuçtur. Simülasyon sonuçlarına göre, DS-1DCNN, önceki yöntemlere göre daha iyi bir karar doğruluğuna sahiptir. Bu nedenle DS-1DCNN, anormalliklere karar vermek için çoklu

sensör verilerinin kullanılmasının gerekli olduğu durumlarda pratik bir modeldir. Sonuç olarak, DS-1DCNN, kararın doğruluğunu verimli bir düzeye çıkarır ve IIoT'deki belirsizliği azaltır.

Bu tezde iki karar verme yöntemi önerilmiş ve bu yöntemlerin verimliliği farklı veri setleri kullanılarak incelenmiştir. İlk yöntem orta ve küçük veri hacmi olan IIoT'ler için uygundur. İkinci yöntem ise veri analizinde DS kullandığı için büyük veri hacmi olan IIoT'ler için uygundur. Sonuç olarak, çoklu veri kaynağı olan uygulamalarda DL yöntemleri DS ile beraber kullanıldığında iyi karar verme doğruluğu elde edilmektedir.



5. ÖNERİLER

Mobil uygulamalar ve gömülü sistemlerin son zamanlarda endüstriyel uygulamalarda oldukça ilgi görmeye başladı. Bu nedenle bu tezde önerilen modellerin mobil ve gömülü sistemlerde kullanılması amacıyla için gerekli olan iyileştirmeler ve değişiklikler üzerinde ileride çalışma yapılabilir. Otomatik kodlayıcılar, yarı denetimli durumlarda model öğrenimi için kullanılabilir. Ayrıca, bu yöntemleri dinamik yapmak için, gelecekte karar füzyonu ve derin pekiştirmeli öğrenme kombinasyonu üzerinde çalışılabilir. Bu sayede model, çok sensörlü ve sınıflı sistemler için öğrenmesini dinamik olarak iyileştirebilir ve dinamik IIoT uygulamalarının da performansları artırılabilir.

6. KAYNAKLAR

1. Gubbi, J., et al., Internet of Things (IoT): A vision, architectural elements, and future directions. *Future generation computer systems*, 2013. 29(7): p. 1645-1660.
2. ÇAVDAR, T. and N. Ebrahimpour, Decision-making for small industrial Internet of Things using decision fusion. *Turkish Journal of Electrical Engineering & Computer Sciences*, 2019. 27(6): p. 4134-4150.
3. Gilchrist, A., *Industry 4.0: the industrial internet of things*. 2016: Apress.
4. Beyerer, J. and T. Uslander, Industrial Internet of Things supporting Factory Automation. *At-Automatisierungstechnik*, 2016. 64(9): p. 697-698.
5. Nagy, J., et al., The role and impact of Industry 4.0 and the internet of things on the business strategy of the value chain—the case of Hungary. *Sustainability*, 2018. 10(10): p. 3491.
6. Chandola, V., A. Banerjee, and V. Kumar, Anomaly detection: A survey. *ACM computing surveys (CSUR)*, 2009. 41(3): p. 1-58.
7. Khalil, R.A., et al., Deep learning in the industrial internet of things: Potentials, challenges, and emerging applications. *IEEE Internet of Things Journal*, 2021. 8(14): p. 11016-11040.
8. Kamat, P. and R. Sugandhi. Anomaly detection for predictive maintenance in industry 4.0-A survey. in *E3S web of conferences*. 2020. EDP Sciences.
9. Peres, R.S., et al., IDARTS—Towards intelligent data analysis and real-time supervision for industry 4.0. *Computers in industry*, 2018. 101: p. 138-146.
10. Sezer, E., et al. An industry 4.0-enabled low cost predictive maintenance approach for smes. in *2018 IEEE International Conference on Engineering, Technology and Innovation (ICE/ITMC)*. 2018. IEEE.
11. Wan, J., et al., A Manufacturing Big Data Solution for Active Preventive Maintenance. *IEEE Transactions on Industrial Informatics*, 2017. 13(4): p. 2039-2047.
12. Jezzini, A., et al. Effects of predictive maintenance (PdM), Proactive maintenance (PoM) & Preventive maintenance (PM) on minimizing the faults in medical instruments. in *2013 2nd International conference on advances in biomedical engineering*. 2013. IEEE.

13. Jardine, A.K., D. Lin, and D. Banjevic, A review on machinery diagnostics and prognostics implementing condition-based maintenance. *Mechanical systems and signal processing*, 2006. 20(7): p. 1483-1510.
14. Baptista, M., et al., Forecasting fault events for predictive maintenance using data-driven techniques and ARMA modeling. *Computers & Industrial Engineering*, 2018. 115: p. 41-53.
15. Wuest, T., et al., Machine learning in manufacturing: advantages, challenges, and applications. *Production & Manufacturing Research*, 2016. 4(1): p. 23-45.
16. Liang, F., et al., Toward edge-based deep learning in industrial Internet of Things. *IEEE Internet of Things Journal*, 2020. 7(5): p. 4329-4341.
17. Mangai, U.G., et al., A survey of decision fusion and feature fusion strategies for pattern classification. *IETE Technical review*, 2010. 27(4): p. 293-307.
18. Lei, Y., et al., Applications of machine learning to machine fault diagnosis: A review and roadmap. *Mechanical Systems and Signal Processing*, 2020. 138: p. 106587.
19. Diez-Olivan, A., et al., Data fusion and machine learning for industrial prognosis: Trends and perspectives towards Industry 4.0. *Information Fusion*, 2019. 50: p. 92-111.
20. Dasarathy, B.V. Industrial applications of multi-sensor multi-source information fusion. in *Proceedings of IEEE International Conference on Industrial Technology 2000* (IEEE Cat. No. 00TH8482). 2000. IEEE.
21. Tong, Y., J. Bai, and X. Chen. Research on Multi-sensor Data Fusion Technology. in *Journal of Physics: Conference Series*. 2020. IOP Publishing.
22. Meng, T., et al., A survey on machine learning for data fusion. *Information Fusion*, 2020. 57: p. 115-129.
23. Li, X., et al., Intelligent rotating machinery fault diagnosis based on deep learning using data augmentation. *Journal of Intelligent Manufacturing*, 2020. 31(2): p. 433-452.
24. Liang, Y., et al., Data-driven anomaly diagnosis for machining processes. *Engineering*, 2019. 5(4): p. 646-652.
25. Bin, G., et al., Early fault diagnosis of rotating machinery based on wavelet packets—Empirical mode decomposition feature extraction and neural network. *Mechanical Systems and Signal Processing*, 2012. 27: p. 696-711.
26. Tian, Z., An artificial neural network method for remaining useful life prediction of equipment subject to condition monitoring. *Journal of intelligent Manufacturing*, 2012. 23(2): p. 227-237.

27. Zhang, X., Y. Liang, and J. Zhou, A novel bearing fault diagnosis model integrated permutation entropy, ensemble empirical mode decomposition and optimized SVM. *Measurement*, 2015. 69: p. 164-179.
28. Yang, B.-S., X. Di, and T. Han, Random forests classifier for machine fault diagnosis. *Journal of mechanical science and technology*, 2008. 22(9): p. 1716-1725.
29. Shah, G. and A. Tiwari. Anomaly detection in iiot: A case study using machine learning. in *Proceedings of the ACM India Joint International Conference on Data Science and Management of Data*. 2018.
30. Marín, G., P. Casas, and G. Capdehourat. Rawpower: Deep learning based anomaly detection from raw network traffic measurements. in *Proceedings of the ACM SIGCOMM 2018 Conference on Posters and Demos*. 2018.
31. Al-Garadi, M.A., et al., A survey of machine and deep learning methods for internet of things (IoT) security. *IEEE Communications Surveys & Tutorials*, 2020. 22(3): p. 1646-1685.
32. Liu, Y., et al. Communication-efficient federated learning for anomaly detection in industrial internet of things. in *GLOBECOM 2020-2020 IEEE Global Communications Conference*. 2020. IEEE.
33. Elbasani, E. and J.-D. Kim, LLAD: Life-Log Anomaly Detection Based on Recurrent Neural Network LSTM. *Journal of Healthcare Engineering*, 2021. 2021.
34. Graß, A., C. Beecks, and J.A.C. Soto, Unsupervised anomaly detection in production lines, in *Machine Learning for Cyber Physical Systems*. 2019, Springer. p. 18-25.
35. Żabiński, T., et al., Condition monitoring in Industry 4.0 production systems-the idea of computational intelligence methods application. *Procedia CIRP*, 2019. 79: p. 63-67.
36. Su, C.-J. and S.-F. Huang, Real-time big data analytics for hard disk drive predictive maintenance. *Computers & Electrical Engineering*, 2018. 71: p. 93-101.
37. Sharma, S., et al., Data-driven optimization of railway maintenance for track geometry. *Transportation Research Part C: Emerging Technologies*, 2018. 90: p. 34-58.
38. Chakravorti, N., et al., Validation of PERFoRM reference architecture demonstrating an application of data mining for predicting machine failure. *Procedia CIRP*, 2018. 72: p. 1339-1344.
39. Accorsi, R., et al., Data mining and machine learning for condition-based maintenance. *Procedia Manufacturing*, 2017. 11: p. 1153-1161.

40. Biswal, S. and G. Sabareesh. Design and development of a wind turbine test rig for condition monitoring studies. in 2015 international conference on industrial instrumentation and control (icic). 2015. IEEE.
41. Kolokas, N., et al. Forecasting faults of industrial equipment using machine learning classifiers. in 2018 Innovations in Intelligent Systems and Applications (INISTA). 2018. IEEE.
42. Praveenkumar, T., et al., Fault diagnosis of automobile gearbox based on machine learning techniques. *Procedia Engineering*, 2014. 97: p. 2092-2098.
43. Bukhsh, Z.A., et al., Predictive maintenance using tree-based classification techniques: A case of railway switches. *Transportation Research Part C: Emerging Technologies*, 2019. 101: p. 35-54.
44. Gutsch, C., et al., Log-based predictive maintenance in discrete parts manufacturing. *Procedia CIRP*, 2019. 79: p. 528-533.
45. Lee, W.J., et al., Predictive maintenance of machine tool systems using artificial intelligence techniques applied to machine condition data. *Procedia Cirp*, 2019. 80: p. 506-511.
46. Von Birgelen, A., et al., Self-organizing maps for anomaly localization and predictive maintenance in cyber-physical production systems. *Procedia cirp*, 2018. 72: p. 480-485.
47. Butte, S., A. Prashanth, and S. Patil. Machine learning based predictive maintenance strategy: a super learning approach with deep neural networks. in 2018 IEEE Workshop on Microelectronics and Electron Devices (WMED). 2018. IEEE.
48. Huuhtanen, T. and A. Jung. Predictive maintenance of photovoltaic panels via deep learning. in 2018 IEEE Data Science Workshop (DSW). 2018. IEEE.
49. De Benedetti, M., et al., Anomaly detection and predictive maintenance for photovoltaic systems. *Neurocomputing*, 2018. 310: p. 59-68.
50. Lasisi, A. and N. Attah-Okine, Principal components analysis and track quality index: A machine learning approach. *Transportation Research Part C: Emerging Technologies*, 2018. 91: p. 230-248.
51. Luo, B., et al., Early fault detection of machine tools based on deep learning and dynamic identification. *IEEE Transactions on Industrial Electronics*, 2018. 66(1): p. 509-518.
52. Schmidt, B. and L. Wang, Predictive maintenance of machine tool linear axes: A case from manufacturing industry. *Procedia manufacturing*, 2018. 17: p. 118-125.

53. Aydin, O. and S. Guldamlasioglu. Using LSTM networks to predict engine condition on large scale data processing framework. in 2017 4th International Conference on Electrical and Electronic Engineering (ICEEE). 2017. IEEE.
54. Canizo, M., et al. Real-time predictive maintenance for wind turbines using Big Data frameworks. in 2017 IEEE international conference on prognostics and health management (icphm). 2017. IEEE.
55. Dos Santos, T., et al. Stator winding short-circuit fault diagnosis in induction motors using random forest. in 2017 IEEE International Electric Machines and Drives Conference (IEMDC). 2017. IEEE.
56. Pan, Z., et al. Cognitive acoustic analytics service for Internet of Things. in 2017 IEEE International Conference on Cognitive Computing (ICCC). 2017. IEEE.
57. Ruschel, E., E.A.P. Santos, and E.d.F.R. Loures, Mining shop-floor data for preventive maintenance management: integrating probabilistic and predictive models. *Procedia Manufacturing*, 2017. 11: p. 1127-1134.
58. Durbhaka, G.K. and B. Selvaraj. Predictive maintenance for wind turbine diagnostics using vibration signal analysis based on collaborative recommendation approach. in 2016 International Conference on Advances in Computing, Communications and Informatics (ICACCI). 2016. IEEE.
59. Susto, G.A. and A. Beghi. Dealing with time-series data in predictive maintenance problems. in 2016 IEEE 21st International Conference on Emerging Technologies and Factory Automation (ETFA). 2016. IEEE.
60. Langone, R., et al., LS-SVM based spectral clustering and regression for predicting maintenance of industrial machines. *Engineering Applications of Artificial Intelligence*, 2015. 37: p. 268-278.
61. Prytz, R., et al., Predicting the need for vehicle compressor repairs using maintenance records and logged vehicle data. *Engineering applications of artificial intelligence*, 2015. 41: p. 139-150.
62. Tehrani, M.M., et al., A predictive preference model for maintenance of a heating ventilating and air conditioning system. *IFAC-PapersOnLine*, 2015. 48(3): p. 130-135.
63. Machado, R.G.V. and H. de Oliveira Mota. Simple self-scalable grid classifier for signal denoising in digital processing systems. in 2015 IEEE 25th International Workshop on Machine Learning for Signal Processing (MLSP). 2015. IEEE.
64. Fox, G.C., S. Kamburugamuve, and R.D. Hartman. Architecture and measured characteristics of a cloud based internet of things. in 2012 International Conference on Collaboration Technologies and Systems (CTS). 2012.

65. Peng, S.-L., S. Pal, and L. Huang, Principles of internet of things (IoT) ecosystem: Insight paradigm. 2020: Springer.
66. Florian, U.D.H.M.M., Architecting the Internet of Things. 1 ed. 2011: Springer-Verlag Berlin Heidelberg.
67. Costigan, S.S. and G. Lindstrom, Policy and the Internet of Things. *Connections*, 2016. 15(2): p. 9-18.
68. Kaliczyńska, M. and P. Dąbek, Value of the Internet of Things for the Industry—An Overview, in *Mechatronics-Ideas for Industrial Application*. 2015, Springer. p. 51-63.
69. Lu, T. and W. Neng. Future internet: The Internet of Things. in 2010 3rd International Conference on Advanced Computer Theory and Engineering(ICACTE). 2010.
70. Overview of the Internet of Things. 2012, Recommendation ITU-T Y.2060. : ITU-T Study Group 13.
71. Korte, A., V. Tiberius, and A. Brem, Internet of Things (IoT) technology research in business and management literature: results from a co-citation analysis. *Journal of Theoretical and Applied Electronic Commerce Research*, 2021. 16(6): p. 2073-2090.
72. Khurana, R. and M. Bhatia, Prevalence of Internet of Things in Pandemic. *Enabling Healthcare 4.0 for Pandemics: A Roadmap Using AI, Machine Learning, IoT and Cognitive Technologies*, 2021: p. 251-274.
73. Suresh, P., et al. A state of the art review on the Internet of Things (IoT) history, technology and fields of deployment. in 2014 International conference on science engineering and management research (ICSEMR). 2014. IEEE.
74. Ashton, K., That ‘internet of things’ thing. *RFiD Journal*, 2011. 22(7).
75. Khanh, Q.V., et al., Wireless Communication Technologies for IoT in 5G: Vision, Applications, and Challenges. *Wireless Communications and Mobile Computing*, 2022. 2022: p. 3229294.
76. Salih, K.O.M., et al., A comprehensive survey on the Internet of Things with the industrial marketplace. *Sensors*, 2022. 22(3): p. 730.
77. Floerkemeier, C., et al., The Internet of Things: First International Conference, IOT 2008, Zurich, Switzerland, March 26-28, 2008, Proceedings. Vol. 4952. 2008: springer.
78. Maghfur, H., A state of the art review on the Internet of Things (IoT). *Buletin Inovasi ICT & Ilmu Komputer*, 2015. 2(1).
79. Man, X., et al. The vision of design-driven innovation in China’s smart home industry. in *International Cognitive Cities Conference*. 2019. Springer.

80. Porras, J., et al., Security challenges and solutions in the internet of things. *Nordic and Baltic Journal of Information and Communications Technologies*, 2018. 2018(1): p. 177-206.
81. Alani, M.M., BotStop: Packet-based efficient and explainable IoT botnet detection using machine learning. *Computer Communications*, 2022.
82. Muteba, F., K. Djouani, and T.O. Olwal, Challenges and solutions of spectrum allocation in NB-IoT technology. *SATNAC*, Ballito, KwaZulu-Natal, South Africa, 2019.
83. Jeschke, S., et al., *Industrial Internet of Things*. Cham, Switzerland: Springer, 2017.
84. Shrouf, F., J. Ordieres, and G. Miragliotta. Smart factories in Industry 4.0: A review of the concept and of energy management approached in production based on the Internet of Things paradigm. in *2014 IEEE International Conference on Industrial Engineering and Engineering Management*. 2014.
85. Pretorius, B. and B. van Niekerk. IIoT Security: Do I Really Need a Firewall for my Train? in *Proceedings of the International Conference on Cyber Warfare & Security*. 2019.
86. Chehri, A. and G. Jeon, The industrial internet of things: examining how the IIoT will improve the predictive maintenance, in *Innovation in medicine and healthcare systems, and multimedia*. 2019, Springer. p. 517-527.
87. Sharma, A., et al., *Industrial Internet of Things: technologies and research directions*. 2022.
88. Qin, J., et al., Industrial Internet of Learning (IIoL): IIoT based pervasive knowledge network for LPWAN—concept, framework and case studies. *CCF Transactions on Pervasive Computing and Interaction*, 2021. 3(1): p. 25-39.
89. Molano, J.I.R., V.H. Medina, and J.F.M. Sanchez, Industrial Internet of Things: An Architecture Prototype for Monitoring in Confined Spaces Using a Raspberry Pi. *Data Mining and Big Data, Dmbd 2016*, 2016. 9714: p. 521-528.
90. Koç, E., ed. *Internet of Things (IoT) Applications for Enterprise Productivity*. 2020, IGI Global: Hershey, PA, USA. 1-357.
91. Goodfellow, I., Y. Bengio, and A. Courville, *Deep learning*. 2016: MIT press.
92. Alpaydin, E., *Machine learning*. 2021: MIT Press.
93. CANBAY, Y., A. İSMETOĞLU, and P. CANBAY, COVID-19 HASTALIĞININ TEŞHİSİNDE DERİN ÖĞRENME VE VERİ MAHREMİYETİ. *Mühendislik Bilimleri ve Tasarım Dergisi*, 2021. 9(2): p. 701-715.

94. Malke, W., Machine Listening with Very Small Training Datasets. 2017, Kutztown University of Pennsylvania.
95. Deng, L., A tutorial survey of architectures, algorithms, and applications for deep learning. APSIPA transactions on Signal and Information Processing, 2014. 3.
96. Druzhkov, P. and V. Kustikova, A survey of deep learning methods and software tools for image classification and object detection. Pattern Recognition and Image Analysis, 2016. 26(1): p. 9-15.
97. Yanming Guo, Y.L., Deep learning for visual understanding: A review. Neurocomputing, 2015.
98. Bengio, Y., Learning deep architectures for AI. Foundations and trends® in Machine Learning, 2009. 2(1): p. 1-127.
99. Ahmad, J., H. Farman, and Z. Jan, Deep learning methods and applications, in Deep learning: convergence to big data analytics. 2019, Springer. p. 31-42.
100. Kelleher, J.D., Deep learning. 2019: MIT press.
101. Hosseini, M.-P., et al., Optimized deep learning for EEG big data and seizure prediction BCI via internet of things. IEEE Transactions on Big Data, 2017. 3(4): p. 392-404.
102. Phung, V.H. and E.J. Rhee, A deep learning approach for classification of cloud image patches on small datasets. Journal of information and communication convergence engineering, 2018. 16(3): p. 173-178.
103. Nielsen, M.A., Neural networks and deep learning. Vol. 25. 2015: Determination press San Francisco, CA, USA.
104. Najafabadi, M.M., et al., Deep learning techniques in big data analytics, in Big Data Technologies and Applications. 2016, Springer. p. 133-156.
105. Islam, S.S., et al. Application of deep learning to computer vision: A comprehensive study. in 2016 5th international conference on informatics, electronics and vision (ICIEV). 2016. IEEE.
106. Hertel, L., et al. Deep convolutional neural networks as generic feature extractors. in 2015 International Joint Conference on Neural Networks (IJCNN). 2015. IEEE.
107. Véstias, M.P., Convolutional neural network, in Encyclopedia of Information Science and Technology, Fifth Edition. 2021, IGI Global. p. 12-26.
108. Sun, S., et al., A survey of optimization methods from a machine learning perspective. IEEE transactions on cybernetics, 2019. 50(8): p. 3668-3681.

109. Zaheer, R. and H. Shaziya. A study of the optimization algorithms in deep learning. in 2019 Third International Conference on Inventive Systems and Control (ICISC). 2019. IEEE.
110. Kingma, D.P. and J. Ba, Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980, 2014.
111. Villa-Pérez, M.E., et al., Semi-supervised anomaly detection algorithms: A comparative summary and future research directions. Knowledge-Based Systems, 2021. 218: p. 106878.
112. Lindemann, B., et al., A survey on anomaly detection for technical systems using LSTM networks. Computers in Industry, 2021. 131: p. 103498.
113. Bergman, L. and Y. Hoshen, Classification-based anomaly detection for general data. arXiv preprint arXiv:2005.02359, 2020.
114. Bergman, L., N. Cohen, and Y. Hoshen, Deep nearest neighbor anomaly detection. arXiv preprint arXiv:2002.10445, 2020.
115. Pecht, M.G. and M. Kang, Machine learning: Anomaly detection. 2019.
116. Liggins II, M., D. Hall, and J. Llinas, Handbook of multisensor data fusion: theory and practice. 2017: CRC press.
117. Raol, J.R., Multi-sensor data fusion with MATLAB®. 2009: CRC press.
118. Wald, L., Data fusion: definitions and architectures: fusion of images of different spatial resolutions. 2002: Presses des MINES.
119. TREATY, N.A., Military Data and Information Fusion.
120. Borges, V., Survey of context information fusion for ubiquitous Internet-of-Things (IoT) systems. Open Computer Science, 2016. 6(1): p. 64-78.
121. Sumari, A.D.W. and A.S. Ahmad. A new model of information processing based on human brain mechanism: toward a cognitive intelligent system. in Proceedings of the 1st Conference on Information Technology, Computer, and Electrical Engineering (CITACEE '13). 2013.
122. Alam, F., et al., Data Fusion and IoT for Smart Ubiquitous Environments: A Survey. Ieee Access, 2017. 5: p. 9533-9554.
123. Hall, D. and J. Llinas, Multisensor data fusion. 2001: CRC press.
124. Calhoun, V.D. and T. Adali, Feature-based fusion of medical imaging data. IEEE Transactions on Information Technology in Biomedicine, 2008. 13(5): p. 711-720.

125. Dempster, A.P., Upper and lower probabilities induced by a multivalued mapping, in *Classic works of the Dempster-Shafer theory of belief functions*. 2008, Springer. p. 57-72.
126. Shafer, G., *A mathematical theory of evidence*. Vol. 1. 1976: Princeton university press Princeton.
127. Wu, W.-Z., et al., Knowledge reduction in random information systems via Dempster–Shafer theory of evidence. *Information Sciences*, 2005. 174(3-4): p. 143-164.
128. Yang, B.-S. and K.J. Kim, Application of Dempster–Shafer theory in fault diagnosis of induction motors using vibration and current signals. *Mechanical Systems and Signal Processing*, 2006. 20(2): p. 403-420.
129. Liu, Y.-Z., et al., CSMC: a combination strategy for multi-class classification based on multiple association rules. *Knowledge-based systems*, 2008. 21(8): p. 786-793.
130. Xiao, Z., et al., A new evaluation method based on D–S generalized fuzzy soft sets and its application in medical diagnosis problem. *Applied Mathematical Modelling*, 2012. 36(10): p. 4592-4604.
131. Li, P. and S.-f. Liu. Intuitionistic fuzzy numbers decision-making methods based on grey incidence analysis and DS theory of evidence. in *Proceedings of 2011 IEEE International Conference on Grey Systems and Intelligent Services*. 2011. IEEE.
132. Wu, D.D., Supplier selection in a fuzzy group setting: A method using grey related analysis and Dempster–Shafer theory. *Expert Systems with Applications*, 2009. 36(5): p. 8892-8899.
133. Verbert, K., R. Babuška, and B. De Schutter, Bayesian and Dempster–Shafer reasoning for knowledge-based fault diagnosis—A comparative study. *Engineering Applications of Artificial Intelligence*, 2017. 60: p. 136-150.
134. Yager, R., M. Fedrizzi, and J. Kacprzyk, *Advances in the Dempster-Shafer theory of evidence*. 1994.
135. Zadeh, L.A., Fuzzy sets as a basis for a theory of possibility. *Fuzzy sets and systems*, 1978. 1(1): p. 3-28.
136. Pearl, J., *Probabilistic reasoning in intelligent systems*. Vol. 88. 2014: Elsevier.
137. Wang, X., et al., Combination of unreliable evidence sources in intuitionistic fuzzy MCDM framework. *Knowledge-Based Systems*, 2016. 97: p. 24-39.
138. Hajiagha, S.H.R., et al., Fuzzy belief structure based VIKOR method: an application for ranking delay causes of Tehran metro system by FMEA criteria. *Transport*, 2016. 31(1): p. 108-118.

139. Zhou, H., et al., Linguistic hesitant fuzzy multi-criteria decision-making method based on evidential reasoning. *International Journal of Systems Science*, 2016. 47(2): p. 314-327.
140. Yang, J.-B., Rule and utility based evidential reasoning approach for multiattribute decision analysis under uncertainties. *European journal of operational research*, 2001. 131(1): p. 31-61.
141. Yang, J.-B. and P. Sen, A general multi-level evaluation process for hybrid MADM with uncertainty. *IEEE Transactions on systems, man, and cybernetics*, 1994. 24(10): p. 1458-1473.
142. Guyon, I., et al., Feature extraction: foundations and applications. Vol. 207. 2008: Springer.
143. Kim, T.-h., Pattern Recognition Using Artificial Neural Network: A Review, in *Information Security and Assurance: 4th International Conference, ISA 2010, Miyazaki, Japan, June 23-25, 2010. Proceedings*, S.K. Bandyopadhyay, et al., Editors. 2010, Springer Berlin Heidelberg: Berlin, Heidelberg. p. 138-148.
144. Wang, J.-S. and C.G. Lee, Self-adaptive neuro-fuzzy inference systems for classification applications. *IEEE Transactions on Fuzzy Systems*, 2002. 10(6): p. 790-802.
145. Bennani, Y. and K. Benabdeslem, Dendogram-based SVM for multi-class classification. *CIT. Journal of computing and information technology*, 2006. 14(4): p. 283-289.
146. Mutlag, W.K., et al. Feature extraction methods: a review. in *Journal of Physics: Conference Series*. 2020. IOP Publishing.
147. Yousefpour, A., et al. Feature reduction using standard deviation with different subsets selection in sentiment analysis. in *Asian Conference on Intelligent Information and Database Systems*. 2014. Springer.
148. Díez-Pastor, J.F., et al., Experimental assessment of feature extraction techniques applied to the identification of properties of common objects, using a radar system. *Applied Sciences*, 2021. 11(15): p. 6745.
149. Pisaroni, M., S. Krumscheid, and F. Nobile, MATHICSE Technical Report: Quantifying uncertain system outputs via the multilevel Monte Carlo method–Part I: Central moment estimation. 2017, MATHICSE.
150. Qin, Y., J. Xing, and Y. Mao, Weak transient fault feature extraction based on an optimized Morlet wavelet and kurtosis. *Measurement Science and Technology*, 2016. 27(8): p. 085003.

151. Attallah, B., A. Serir, and Y. Chahir, Feature extraction in palmprint recognition using spiral of moment skewness and kurtosis algorithm. *Pattern Analysis and Applications*, 2019. 22(3): p. 1197-1205.
152. Oh, Y.R., et al. Deep-Learning Based Automatic Spontaneous Speech Assessment in a Data-Driven Approach for the 2017 SLATE CALL Shared Challenge. in *SLaTE*. 2017.
153. Yegnanarayana, B., *Artificial neural networks*. 2009: PHI Learning Pvt. Ltd.
154. Di Martino, S., et al. Industrial internet of things: persistence for time series with NoSQL databases. in *2019 IEEE 28th International Conference on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE)*. 2019. IEEE.
155. Huynh, V.-N. Discounting and Combination Scheme in Evidence Theory for Dealing with Conflict in Information Fusion. in *MDAI*. 2009. Springer.
156. Sentz, K. and S. Ferson, *Combination of evidence in Dempster-Shafer theory*. Vol. 4015. 2002: Citeseer.
157. Hinton, G.E. and R.R. Salakhutdinov, Reducing the dimensionality of data with neural networks. *science*, 2006. 313(5786): p. 504-507.
158. Mozaffari, M.H. and L.-L. Tay, A Review of 1D Convolutional Neural Networks toward Unknown Substance Identification in Portable Raman Spectrometer. *arXiv preprint arXiv:2006.10575*, 2020.
159. Kiranyaz, S., et al., 1D convolutional neural networks and applications: A survey. *Mechanical systems and signal processing*, 2021. 151: p. 107398.
160. Srinivasamurthy, R.S., *Understanding 1D Convolutional Neural Networks Using Multiclass Time-Varying Signalss*. 2018, Clemson University.
161. Olson, D.L. and D. Delen, *Advanced data mining techniques*. 2008: Springer Science & Business Media.
162. Zhang, Z. Improved adam optimizer for deep neural networks. in *2018 IEEE/ACM 26th International Symposium on Quality of Service (IWQoS)*. 2018. Ieee.
163. Karmakar, S., et al., Induction Motor and Faults, in *Induction Motor Fault Diagnosis : Approach through Current Signature Analysis*. 2016, Springer Singapore: Singapore. p. 7-28.
164. Palumbo, F., et al., Human activity recognition using multisensor data fusion based on reservoir computing. *Journal of Ambient Intelligence and Smart Environments*, 2016. 8(2): p. 87-107.

165. Tnani, M.-A., M. Feil, and K. Diepold, Smart Data Collection System for Brownfield CNC Milling Machines: A New Benchmark Dataset for Data-Driven Machine Monitoring. *Procedia CIRP*, 2022. 107: p. 131-136.
166. Joseph, F.J.J., S. Nonsiri, and A. Monsakul, Keras and TensorFlow: A hands-on experience, in *Advanced deep learning for engineers and scientists*. 2021, Springer. p. 85-111.
167. Vasilev, I., et al., *Python Deep Learning: Exploring deep learning techniques and neural network architectures with Pytorch, Keras, and TensorFlow*. 2019: Packt Publishing Ltd.?



ÖZGEÇMİŞ

Lise öğrenimini Miyandoab Froghi lisesinde tamamladı, 2006 yılında Tabriz Uygulamalı Bilim ve Teknoloji Üniversitesi Bilgisayar Mühendisliği Bölümünden Bilgisayar Yazılım Mühendisi ünvanı ile mezun oldu, 2009 yılında Qazvin Azad Üniversitesi Bilgi Teknolojileri Mühendisliği-Bilgisayar Ağı Bölümü'nden Bilgi Teknolojileri Yüksek Mühendisi ünvanı ile mezun oldu. 2014 yılında KTÜ Bilgisayar Mühendisliği Anabilim Dalı'nda Doktora öğrenimine başladı. Nader EBRAHIMPOUR iyi derecede İngilizce, Türkçe, Farsça ve Azerbaycan Türkçesi bilmektedir. Tez süresi boyunca yayınlanan eserleri aşağıda verilmiştir:

SCI / SCI-Expanded İndeksli Dergi Yayınları

- Çavdar, T., Ebrahimpour, N., Kakız, M.T. and Günay, F.B., 2022. Decision-making for the anomalies in IIoTs based on 1D convolutional neural networks and Dempster–Shafer theory (DS-1DCNN). The Journal of Supercomputing, pp.1-22.
- Çavdar, T. and Ebrahimpour, N., 2019. Decision-making for small industrial Internet of Things using decision fusion. Turkish Journal of Electrical Engineering and Computer Sciences, 27(6), pp.4134-4150.
- Çavdar, T., Gänay, F.B., Ebrahimpour, N. and Kakáz, M.T., 2022. An optimal anchor placement method for localization in large-scale wireless sensor networks. Intell. Autom. Soft Comput., 31(2), pp.1197-1222.
- Çavdar, T., Kakız, M.T., Günay, F.B. and Ebrahimpour, N., 2021. SERDP: Signature-based and energy-efficient relay discovery protocol for Internet of Things in cellular networks. International Journal of Communication Systems, 34(12), p.e4898.

Diğer Dergilerde Yayınlanan Makaleler

- Ebrahimpour, N., Öztürk, E. and Çavdar, T., Applications of Intermittently Connected Delay Tolerant Wireless Sensor Networks (ICDT-WSNs)–A Survey. International Journal of 3D Printing Technologies and Digital Industry, 3(3), pp.254-260.
- Ebrahimpour, N., Ayden, M.A. and Altay, B., 2022. Liveness control in face recognition with deep learning methods. The European Journal of Research and Development, 2(2), pp.92-101.

Uluslararası Konferans Yayınları

Ebrahimpour, N., Ayden, M.A. and Altay, B., Derin öğrenme yöntemleriyle yüz tanımda canlılık kontrolü International Conference On Emerging Sources In Science, May 26-27, 2022.

Ebrahimpour, N., Öztürk, E. and Çavdar, T., Applications of Intermittently Connected Delay Tolerant Wireless Sensor Networks (ICDT-WSNs)—A Survey. 4th International Congress On 3D Printing (Additive Manufacturing) Technologies and Digital Industry, Antalya, Türkiye, 11 - 14 Nisan 2019.

