

# Anomaly Diagnosis Using Autoencoder in Edge Computing Systems

Mert Kışlakçı <sup>a,1</sup>, Mahmut Durgun <sup>b</sup>

<sup>a</sup> Tokat Gaziosmanpaşa University, Graduate Education Institute, Mechatronics Engineering Department, Tokat, Turkey  
ORCID ID: 0000-0001-8506-7498

<sup>b</sup> Tokat Gaziosmanpaşa University, Graduate Education Institute, Mechatronics Engineering Department, Tokat, Turkey  
ORCID ID: 0000-0002-5010-687X

## Abstract

IoT systems operate in a traditional cloud-connected architecture. The data generated on IoT devices is transferred to the cloud, stored there, and then processed, trying to extract meaningful information. However, in this preferred structure, the disadvantages of being constantly dependent on the cloud are quite high. While transferring each piece of information raw to the cloud increases network traffic, it requires high hardware power to process data only at the cloud layer. UBISOKKAT (Anomaly Diagnosis Using Automatic Encoder in Edge Computing Systems) system has emerged as a solution to the problems mentioned above. The UBISOKKAT system acts as an intermediate layer between IoT systems and cloud systems. Every data generated at IoT points is first sent to the UBISOKKAT system in the middle layer, where it is transmitted to the cloud layer. The machine learning model is then placed in the cloud tier and begins to train itself using the data it receives from the middleware. The outputs of the model whose training process is completed are sent to the UBISOKKAT system and the automatic encoder is run in the middleware software, not in the cloud. The biggest advantage of this is that in real-time systems, data is not sent to the cloud but analyzed at local points, reducing network traffic and reducing latency. At the same time, since not every data is analyzed in the cloud, it is analyzed at local points, reducing the need for the cloud, reducing high costs and increasing the viability of the system. In this study, an automatic encoder model was operated in the last layer and it was seen that the UBISOKKAT system could diagnose anomalies at the extreme points based on the data obtained from the single-phase electric motor.

**Keywords:** “Edge computing, machine learning, anomaly diagnosis, cloud systems, IoT.”

## 1. Giriş

Endüstri 4.0 ile hayatımıza giren Nesnelerin İnterneti (IoT), elektronik cihazların çeşitli sensor ve donanımlarla çevresel durumları algılayabildiği ve aralarında haberleşerek veri üretebilen aygıtlar topluluğudur [1]. Bulut bilişim, sistemin geçmiş verilerinin izlenmesine ve depolanmasına ve uç cihazlara kıyasla daha yüksek bilgi işlem gücü gerektiren karmaşık işlemlerin gerçekleştirilmesine yardımcı olur [2]. Bununla birlikte, merkezi bir IoT veri analizi yaklaşımı, özellikle bu mevcut bulut bilişim çağında en iyi ve en uygun çözüm olmayabilir [3]. Uç sistemler, Endüstri 4.0'ın gelişimini büyük ölçüde destekleyen IIoT uygulamaları için gerçek zamanlı, esnek ve hızlı karar vermeyi sağlayan güçlü hesaplama kaynakları sağlamaktadır [4]. Geleneksel bulut tabanlı Nesnelerin İnterneti (IoT) mimarisi tasarımında, tüm veriler buluta gönderilerek öngörülemez gecikmeler, gizlilik endişeleri ve gerçek zamanlı olmayan özellikleri ortaya çıkarmaktadır. IoT sistemlerinde, ağ uç cihazlarının (ağ geçitleri, düğümler, vb.), anormallik tespiti gibi bazı amaçlar için verileri yeterince işleyecek kadar akıllı hale getirilmesi durumunda, bilgi işleminin daha gerçek zamanlı ve verimli olabileceği sıklıkla bulunur [5]. IoT alanında tahmin, sınıflandırma ve anormallik tespiti için derin öğrenme algoritmalarının uygulanması, teknolojinin durumunu iyileştirmektedir [3]. Arızalardan önce gelebilecek erken anormallikler tespit edilebilir ve aralıklarla önleyici bakım, ani arızaları önleyebilir ve bu IoT cihazlarının bazılarının kullanım ömrünün uzamasına yardımcı olarak operasyonel maliyetlerden tasarruf sağlanmasına yardımcı olmaktadır [6]. Son zamanlarda, elektrikli makinelerin arıza teşhisi konusundaki araştırmaları hararetli bir şekilde hız kazandı. Motorda kullanılan arıza tespit metodları hız, tork, gürültü, titreşim vb. gibi diğer sinyallerin frekans içerikleri ile yapılmaktadır [7].

Baghbanpourasl ve ark. yaptıkları modele dayalı bir makine öğrenimi yöntemi kullanarak tamamen veriye dayalı anormallik algılama çalışmasının erken aşamalarında arıza tespitini amaçlamışlardır. Hatta bağlanan 4 farklı sıcaklık sensöründen elde edilen veriler kullanılarak makine öğrenme modeli eğitilmiştir. Çalışma sonuçlarına göre veri setindeki tüm arızalı cihazları tespit edebilir ve ölçüm süresini en az %43 oranında azalttığı bulgusuna rastlanmıştır [8].

<sup>1</sup> Corresponding Author  
E-mail Address: mertkislakci@gmail.com



mekanik güce toplu güç dönüşümü için uygulanır. Küçük güç dönüşümü için çoğunlukla tek fazlı AC motorlar kullanılır. Tek fazlı AC motorun boyutu neredeyse küçüktür ve ev, ofis, iş kaygıları, fabrikalar vb. alanlarda çeşitli hizmetler sunar [13].



Şekil 2. AC motor.

### 2.3. Raspberry Pi

Raspberry Pi işletim sistemi Linux olan üzerinde mikroişlemci(ler), bellek, giriş/çıkış (G/Ç) ve işlevsel bir bilgisayar için gereken diğer özelliklere sahip tek bir devre kartı üzerine kurulmuş bilgisayardır [14]. Raspberry Pi 3 B+ kart üzerinde 1.4GHz'de çalışan, 4 çekirdekli 64-bit bir işlemci bulunur. Bu modelde 4.1 olan bluetooth desteği 4.2'ye çıkarılmıştır. Çift-Band 2.4GHz ve 5GHz kablosuz yerel ağ bağlantısına sahiptir. Ağ bağlantılarına önem veren bu model daha hızlı bir internete sahiptir. WI-FI ve PoE gibi konuların üzerinde duran model, PoE desteği ile artık adaptör olmadan Ethernet hattından hem data hem güç bağlantısı yapmaktadır[15].

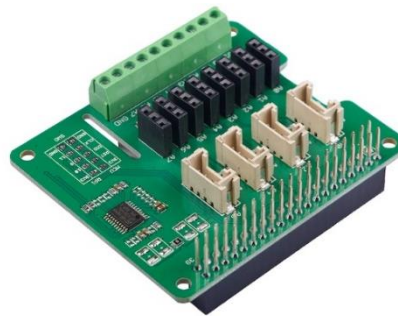


Şekil 3. Raspberry Pi.

### 2.4. STM32F030

Raspberry Pi bir donanım analogdan dijitale dönüştürücü içermez, bu nedenle harici ADC STM32F030 sürekli değişen analog sinyalleri ayrık zamanlı dijital sinyallere dönüştürmede kullanılır [16].

- Çalışma Besleme Gerilimi 3.3V
- ADC Çözünürlüğü 12 bit
- Maksimum Saat Frekansı 48 MHz



Şekil 4. STM32F030

## 2.5. Analog Piezoelektrik Seramik Titreşim Sensörü

Titreşim sensörü, piezoelektrik seramik çip analog titreşimine dayanmaktadır. Piezoelektrik seramik, titreşiminin elektrik sinyallerini üreteceği anti-dönüşüm sürecini kullanır. Piezoelektrik seramik çip titreştiğinde, sensörün sinyal terminali elektrik sinyalleri üretir [17].



Şekil 5. Titreşim sensörü.

## 2.6. Bulut Bilişim

Bulut bilişim, daha hızlı yenilik, esnek kaynaklar ve ölçek ekonomileri sunmak için sunucular, depolama, veritabanları, ağ iletişimi, yazılım, analitik ve zeka dahil olmak üzere bilgi işlem hizmetlerinin İnternet üzerinden sunulmasıdır. Bulut kaynakları İnternet üzerinden, gelen veya giden istek üzerine kullandıkça ödeme fiyatlandırmasıyla sunulmaktadır [18].



Şekil 6. Bulut bilişim.

## 2.7. Uç Bilişim

Uç (sınır) bilişim mimarisi, verilerin buluttan uzak noktalarında işlendiği veya kullanıldığı her yerde uç bilgi işleme etkin olan tüm bileşenleri (tüm cihazlar, sensörler, sunucular, bulutlar vb.) kapsayan dağıtılmış bir bilgi işlem mimarisidir. Bir sistemin verilerine veya son kullanıcıya en yakın konumda, yani bilginin geldiği veya gittiği yerde yapılan hesaplamayı ifade eder. Kenar mimarisi, gecikmeyi ve gecikmeyi azaltarak işlemenin daha hızlı gerçekleşmesini sağlar. Uçta çalışan uygulamalar ve programlar daha hızlı ve verimli çalışarak daha iyi bir kullanıcı deneyimi ve daha iyi genel performans sağlar [19].



Şekil 7. Uç bilişim.

## 2.8. Makine Öğrenimi Modeli – Otomatik Kodlayıcı Sinir Ağı

Bir otomatik kodlayıcı, birkaç düğüm katmanından oluşur. İlk birkaç katman düğüm boyutunda küçülür (kodlayıcı) ve ikinci katman kümesi düğüm boyutunda artar (kod çözücü). Bir otomatik kodlayıcının arkasındaki temel fikir, giriş katmanında kendisine verileni yeniden oluşturmaya çalışmasıdır. Otomatik kodlayıcının ne kadar iyi performans gösterdiğini, girdi değerleri ile tahmin edilen (çıkış) değerler arasındaki ortalama karesel hatayı (MSE) hesaplayarak ölçülür [20].

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

Şekil 8. Otomatik kodlayıcı sinir ağı.

## 2.9. Colab

Colab, yapay zekâ ve derin öğrenme projeleri üzerinde çalışanlar için etkileşimli, tamamen bulut tabanlı, kullanımı kolay ve ortak çalışmaya dayalı bir programlama ortamıdır [21].



Şekil 9. Colab.

## 2.10. Keras

Keras, Python ile yazılmış, makine öğrenimi platformu TensorFlow üzerinde çalışan bir derin öğrenme uygulamasıdır. Hızlı denemeyi mümkün kılmaya odaklanılarak geliştirilmiştir. Fikirden sonuca mümkün olduğunca hızlı gidebilmek, iyi araştırma yapmanın anahtarıdır [22].



Şekil 10. Keras.

## 2.11. Tensorflow

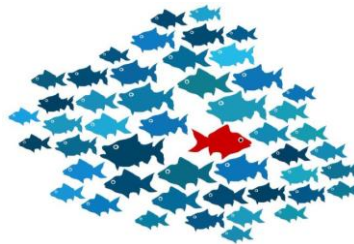
TensorFlow, makine öğrenimini ve sinir ağlarını geliştirmeyi daha hızlı ve daha kolay hale getiren sayısal hesaplama için Python dostu bir açık kaynak kitaplığıdır. Araştırmacıların makine öğrenimi alanındaki en son teknolojiyi kullanmalarına ve geliştiricilerin makine öğrenimi destekli uygulamaları kolayca oluşturup dağıtmalarına olanak tanıyan kapsamlı, esnek bir araç, kitaplık ve topluluk kaynakları ekosistemine sahiptir [23].



Şekil 11. Tensorflow.

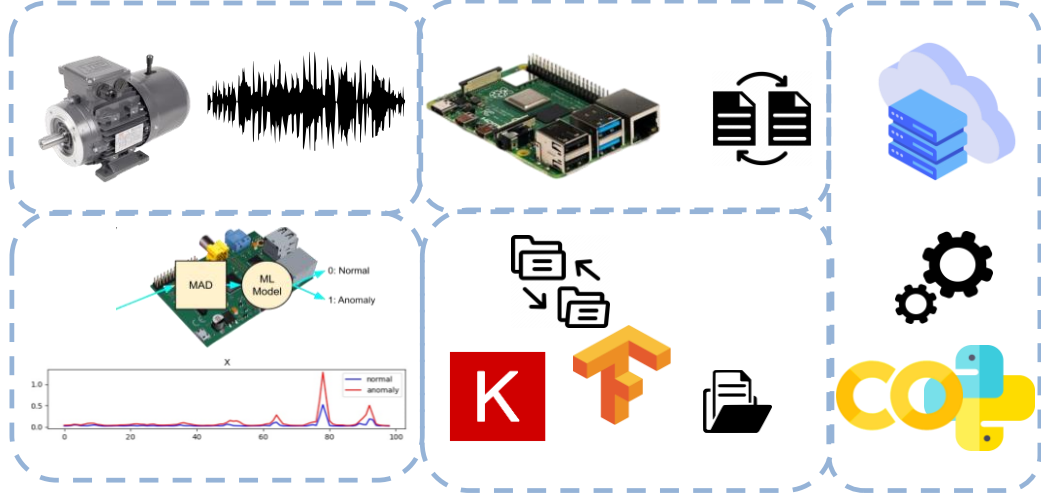
## 2.12. Anomali Teşhisi

Anomali Teşhisi, belirli bir veri kümesindeki aykırı değerleri bulma işlemidir. Aykırı değerler, veri kümesindeki diğer nesnelere arasında öne çıkan ve bir veri kümesindeki normal davranışa uymayan veri nesnelere dir. Anomali algılama, sınıflandırma, regresyon ve kümeleme gibi birden çok veri bilimi görevini birleştiren bir veri bilimi uygulamasıdır. Tahmin edilecek hedef değişken, bir işlemin aykırı değer olup olmadığıdır. Kümeleme görevleri aykırı değerleri bir küme olarak tanımladığından, anomali algılama görevlerinde mesafe tabanlı ve yoğunluk tabanlı kümeleme teknikleri kullanılabilir [24].



Şekil 12. Anomali teşhisi.



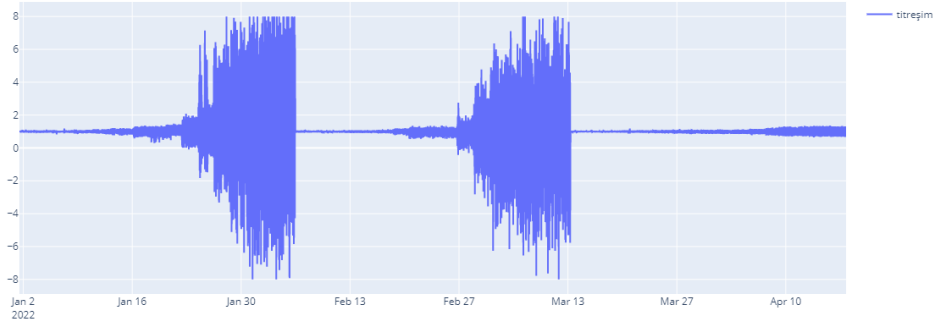


Şekil 14. Sistem tasarımı.

#### 4. Bulgular

Yapılan çalışmada tek fazlı elektrik motorunun titreşim ölçümü tek eksende yapılmıştır. Zaman serilerine bağlı toplamda 153000 adet örnek toplanmıştır. Toplanan bu örneklerin 20% test (30599 adet), kalan 80% eğitim (122401) verisi olarak ayrılmıştır.

Uç Bilgi İşlem Sistemlerinde Otomatik Kodlayıcı Kullanarak Anomali Teşhisi



Şekil 15. Tek fazlı motora ait titreşim grafiği.

Sensörden elde edilen değerlerin grafikte düzgün görüntülenebilmesi ve modelin iyi yorumlayabilmesi amacıyla 200% sabitiyle oranlama yapılmıştır. Bu oran neticesinde Şekil 15'te yer alan titreşim grafiğe bakıldığında değerler 8 ile -8 arasında olduğu anlaşılmaktadır. İki katmanlı olarak tasarlanan otomatik kodlayıcı totalde 612007 adet parametre ile eğitime başlanmıştır.

```

encoding_dim = 2
model = models.Sequential([
    layers.InputLayer(input_shape=sample_shape),
    layers.Dense(encoding_dim, activation='relu'),
    layers.Dropout(0.2),
    layers.Dense(*sample_shape, activation='relu')
])

model.summary()

Model: "sequential_3"
-----
Layer (type)                Output Shape              Param #
-----
dense_7 (Dense)              (None, 2)                 244804
dropout_4 (Dropout)          (None, 2)                 0
dense_8 (Dense)              (None, 122401)           367203
-----
Total params: 612,007
Trainable params: 612,007
Non-trainable params: 0

```

Şekil 16. Model katmanı.

10 adım şeklinde eğitim başlatılmış olup her bir eğitim dizisinin uzunluğu ise 3825 elemandan oluşmaktadır.

```

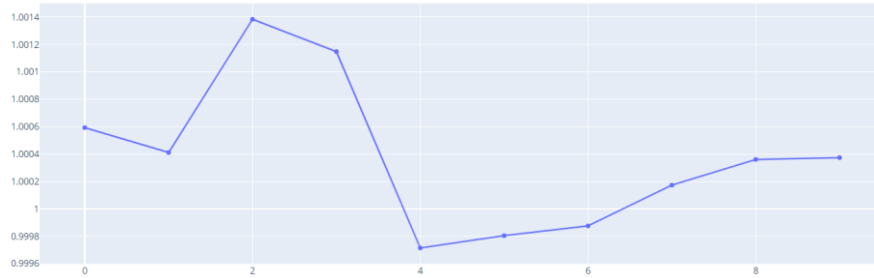
history = model.fit(X_train, y_train, epochs=10, verbose= 1)

Epoch 1/10
3825/3825 [=====] - 286s 74ms/step - loss: 1.0006
Epoch 2/10
3825/3825 [=====] - 281s 73ms/step - loss: 1.0004
Epoch 3/10
3825/3825 [=====] - 281s 74ms/step - loss: 1.0014
Epoch 4/10
3825/3825 [=====] - 288s 75ms/step - loss: 1.0011
Epoch 5/10
3137/3825 [=====>.....] - ETA: 50s - loss: 1.0064

```

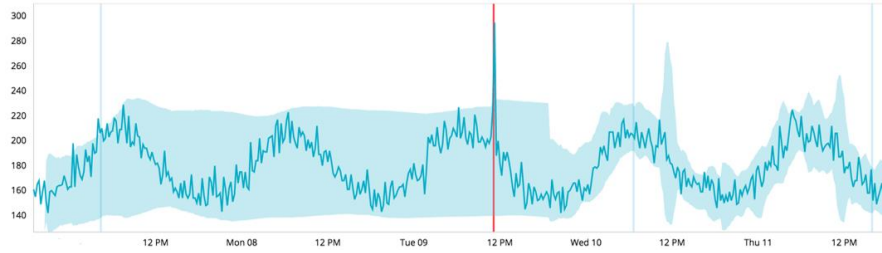
Şekil 17. Model eğitimi.

Modele ait doyunluk grafiği şekil 18’de verilmiştir. Grafik yorumlandığında model doyunluğa ulaşmadığı yani yeterli derecede veriler üzerinden kendini eğitmediği sonucu çıkmaktadır. Modelin istenilen doyunluğa ulaşması için adım sayısı yükseltilmeli ve katman sayısı artırılmalıdır. Katman sayısını arttırmak modelin bir adımda bulundurduğu dizi sayısını kısaltmak anlamına gelmektedir. Böylelikle modelin öğrenme başarısı artış gösterecektir.



Şekil 18. Model doyunluğu.

Model ağırlığı test veri kümesi ile test edilmiştir. Çıkan sonuçlar şekil 19 da yer verilmiştir. Grafik yorumlandığında koyu mavi ile gösterilen çizgiler test verisini içermektedir. Açık mavi ile taratılan alan modelin %15 kadar hata toleransı katmıştır bunun yapılmasındaki neden modelin başarısını yükseltecek bir filtreleme aralığı elde etmektir. Kırmızı çizgi ise titreşim verilerinde algılanan anormallikleri temsil eder. Bakılan aralıkta 12:00’de bir adet anormallik tespit edilmiştir.



Şekil 19. Model sonucu.

## 5. Tartışma

Elde edilen deneysel veriler ve tasarım sırasında oluşacak muhtemel problemlerin çözümüne ait detaylı raporlama yapılarak akademik ve ticari projelere referans olması amaçlanmaktadır. Uç bilişim yeni bir teknoloji değildir lakin bu konuda Türkiye’deki kaynaklar referans alındığında kullanımı henüz yaygınlaşmamıştır. Bu bağlamda UBISOKKAT çalışması yurtiçinde muadil projelere benzememekte olup yeni araştırmalar yapılmasına öncü olmayı hedeflemektedir. Ayrıca sağlık, eğitim, ulaşım, savunma, endüstri, elektronik gibi alanlarda Ar-Ge bazlı veya ticari asıllı projeler başlatma potansiyeli oldukça fazladır. Nesnelerin İnterneti (IoT) görevlerinin esnasında oluşan donanım ihtiyacı gibi yatırım harcamalarını (CAPEX) ve operasyonel giderleri (OPEX) maliyetini düşürme çabası içindedir.

## 6. Sonuç

“Uç Bilgi İşlem Sistemlerinde Otomatik Kodlayıcı Kullanarak Anomali Teşhisi” çalışmasının sonuçları gösteriyor ki; Otomatik Kodlayıcılar kullanılarak elektrik motorlarından elde edilen verisetleriyle anormallik tespiti yapmak mümkün hale gelmektedir.



Makine öğrenme modellerinin uç katmanda çalıştırılması bulut sistemleri kullanmanın maliyetini düşürerek, uygun maliyetli arıza tespit sistemlerinin geliştirilmesine olanak sağlayacaktır. Ayrıca sistemin farklı arıza senaryolarında ve motor tiplerinde testleri devam etmektedir. Çalışmaların tamamlanmasından sonra beklenen kazanımlar şunlardır;

- UBISOKKAT (Uç Bilgi İşlem Sistemlerinde Otomatik Kodlayıcı Kullanarak Anomali Teşhisi) sistemi arıza senaryolarına uyarlanabilecek çok yönlü bir proje olma niteliği taşımaktadır.
- Nesnelerin İnterneti (IoT) platformuna dahil edilen sistemler hakkında daha fazla bilgi edinilmesine katkı sunacaktır.
- Nesnelerin İnterneti platformunda çalışan her sistemin verimli çalışabilmesi sağlanırken beklenmeyen arızaların önüne geçilecektir.
- Bulut sistemlerine duyulan ihtiyaçlar azalacak IoT arıza teşhis daha uygun maliyetli tasarlanmasına katkı sağlayacaktır.
- Gerçek zamanlı uygulamalarda arızanın bulut mimarisine oranla daha hızlı tespit edilmesine imkânı sunacaktır.
- Veri kirliliğinin önüne geçilerek sadece işe yarayacak veriler depo edilecektir. Bu sayede büyük veri sorununa da çözüm sunacaktır
- Makine öğrenme modellerinin artık uç sistemlerde çalıştırılabilmesi için teknolojik gelişmelere rehberlik edecektir.

## Kaynakça

- [1] Q. Xu, J. Zhang, ve B. Togookhuu, “Support mobile fog computing test in piFogBedII”, *Sensors (Switzerland)*, 2020, doi: 10.3390/s20071900.
- [2] Aydemir, F., “IoT Based Indoor Disinfection Coordinating System Against the New Coronavirus”, *International Scientific and Vocational Studies Journal*, 4(2),81 - 85. doi: 10.47897/bilmes.751995.
- [3] O. Kayode, D. Gupta, ve A. S. Tosun, “Towards a Distributed Estimator in Smart Home Environment”, *IEEE World Forum Internet Things, WF-IoT 2020 - Symp. Proc.*, Haz. 2020, doi: 10.1109/WF-IOT48130.2020.9221083.
- [4] Y. Liu vd., “Deep Anomaly Detection for Time-Series Data in Industrial IoT: A Communication-Efficient On-Device Federated Learning Approach”, *IEEE Internet Things J.*, c. 8, sayı 8, ss. 6348–6358, Nis. 2021, doi: 10.1109/JIOT.2020.3011726.
- [5] D. Utomo ve P. A. Hsiung, “Anomaly Detection at the IoT Edge using Deep Learning”, *2019 IEEE Int. Conf. Consum. Electron. - Taiwan, ICCE-TW 2019, May. 2019*, doi: 10.1109/ICCE-TW46550.2019.8991929.
- [6] O. Kayode ve A. S. Tosun, “LIRUL: A Lightweight LSTM based model for Remaining Useful Life Estimation at the Edge”, *Proc. - Int. Comput. Softw. Appl. Conf.*, c. 2, ss. 177–182, Tem. 2019, doi: 10.1109/COMPSAC.2019.10203.
- [7] S. Nandi, H. A. Toliyat, ve X. Li, “Condition monitoring and fault diagnosis of electrical motors - A review”, *IEEE Trans. Energy Convers.*, c. 20, sayı 4, ss. 719–729, Ara. 2005, doi: 10.1109/TEC.2005.847955.
- [8] A. Baghbanpourasl, D. Kirchberger, ve C. Eitzinger, “Failure prediction through a model-driven machine learning method”, *2021 IEEE Int. Work. Metrol. Ind. 4.0 IoT, MetroInd 4.0 IoT 2021 - Proc.*, ss. 527–531, Haz. 2021, doi: 10.1109/METROIND4.0IOT51437.2021.9488550.
- [9] M. Adkisson, J. C. Kimmell, M. Gupta, ve M. Abdelsalam, “Autoencoder-based Anomaly Detection in Smart Farming Ecosystem”, *Proc. - 2021 IEEE Int. Conf. Big Data, Big Data 2021*, ss. 3390–3399, 2021, doi: 10.1109/BIGDATA52589.2021.9671613.
- [10] Y. H. Park ve M. J. Kim, “Design of Cost-Effective Auto-Encoder for Electric Motor Anomaly Detection in Resource Constrained Edge Device”, *Proc. 3rd IEEE Eurasia Conf. IOT, Commun. Eng. 2021, ECICE 2021*, ss. 241–246, 2021, doi: 10.1109/ECICE52819.2021.9645739.
- [11] L. Njilla, L. Pearlstein, X. W. Wu, A. Lutz, ve S. Ezekiel, “Internet of Things Anomaly Detection using Machine Learning”, *Proc. - Appl. Imag. Pattern Recognit. Work.*, c. 2019-October, Eki. 2019, doi: 10.1109/AIPR47015.2019.9174569.
- [12] “Intelligent Systems with Internet of Things (IoT) Part 1 & 2 & 3 | by Mert Kışlakçı | Medium”. <https://medium.com/@mertkslk/intelligent-systems-with-internet-of-things-iot-part-1-2-3-7353dec3cbf8> (erişim Haz. 12, 2022).
- [13] “What is AC Motor? Definition & Types - Circuit Globe”. <https://circuitglobe.com/ac-motor.html> (erişim Haz. 12, 2022).

- [14] “What is a Raspberry Pi and How Does it Work? | Pi Day”. <https://www.piday.org/whats-a-raspberry-pi-and-how-does-it-work/> (erişim Haz. 11, 2022).
- [15] “Alışverişe Devam Edin”. <https://market.samm.com/raspberry-pi-3-b-plus> (erişim Haz. 11, 2022).
- [16] “8-Channel 12-Bit ADC for Raspberry Pi (STM32F030) - Seeed Studio”. <https://www.seeedstudio.com/8-Channel-12-Bit-ADC-for-Raspberry-Pi-STM32F030.html> (erişim Haz. 12, 2022).
- [17] “Analog Piezoelektrik Seramik Titreşim Sensörü Satın Al | Robotelektronik.com”. <https://www.robotelektronik.com/urun/analog-piezoelektrik-seramik-titresim-sensoru> (erişim Haz. 15, 2022).
- [18] “What is cloud computing? Everything you need to know about the cloud explained | ZDNet”. <https://www.zdnet.com/article/what-is-cloud-computing-everything-you-need-to-know-about-the-cloud/> (erişim Haz. 12, 2022).
- [19] “What is edge architecture?” [https://www.redhat.com/en/topics/edge-computing/what-is-edge-architecture?sc\\_cid=7013a000002pu3zAAA&gclid=CjwKCAjwnZaVBhA6EiwAVVyv9E3Yr7ZeN8b6zMO9bT\\_IRpEGCm8UpVe6hmw5oWRnMNvApOyoDOAqUxoCN0YQAvD\\_BwE&gclsrc=aw.ds](https://www.redhat.com/en/topics/edge-computing/what-is-edge-architecture?sc_cid=7013a000002pu3zAAA&gclid=CjwKCAjwnZaVBhA6EiwAVVyv9E3Yr7ZeN8b6zMO9bT_IRpEGCm8UpVe6hmw5oWRnMNvApOyoDOAqUxoCN0YQAvD_BwE&gclsrc=aw.ds) (erişim Haz. 12, 2022).
- [20] “Edge AI Anomali Algılama Bölüm 2 - Özellik Çıkarma ve Model Eğitimi”. <https://www.digikey.com/en/maker/projects/edge-ai-anomaly-detection-part-2-feature-extraction-and-model-training/70927a6e439b49bea7305953a3c9bfff> (erişim Haz. 11, 2022).
- [21] “Google Colab Nedir ve Nasıl Kullanılır? – Global AI Hub”. <https://globalaihub.com/blog/google-colab-nedir-ve-nasil-kullanilir/> (erişim Haz. 12, 2022).
- [22] “About Keras”. <https://keras.io/about/> (erişim Haz. 12, 2022).
- [23] “TensorFlow”. <https://www.tensorflow.org/> (erişim Haz. 12, 2022).
- [24] V. Kotu ve B. Deshpande, “Anomaly Detection”, Data Sci., ss. 447–465, Oca. 2019, doi: 10.1016/B978-0-12-814761-0.00013-7.