

Atanmış Bilgisayarlar ile Kuantum Ağların Dağıtık Benzetimi

Osman Semi CEYLAN, İhsan YILMAZ

Hava Trafik Yönetim Sistemleri Çekirdek Ağlarında Servis Kalitesi Temelli Ağ Trafik Yönetimi (Air-SDN)

Sertan PEKEL, Müge SAYIT

Derin Öğrenme ile Türkçede Adil Çözümleme

Mehmet TAZE, Senem KUMOVA METİN

Makine Öğrenmesi Yöntemleri ile Kan Sayımında Covid-19 Tanısı

Büşra ÇAKI, Ahmet EGESÖY, Yasemin TOPALOĞLU

Yazılım-tanımlı İHA Ağları için Deney Ortamı Tasarımı

Gökhan SEÇİNTİ

Güncel Düzmece Video Algılama Yöntemleri Üzerine Bir Kaynak Derlemesi

Suzan AYDIN, Zeki ÖZEN

Makine Öğrenmesi Temelli Obezite Durum Tahmini

Ercan ÖLÇER

Makine Öğrenmesi Yöntemleri Kullanılarak Kötü Amaçlı Yazılım Sınıflandırması: CIC-MamMem-2022 Veri Kümesi Üzerinde bir Başarım Karşılaştırması

Oğuzhan KIRLAR

Hedef Gen Çıkarımı için Makine Öğrenmesi Tabanlı Mikrodizi Verilerinin Analizi

Zerrin YILDIZ ÇAVDAR, Ahmet SERTBAŞ, Tolga ENSARİ

## EDİTÖR KURULU

### Eş-Baş Editörler

Prof. Dr. Eşref ADALI - İstanbul Teknik Üniversitesi

Prof. Dr. İbrahim SOĞUKPINAR - Gebze Teknik Üniversitesi

### Dil Editörü

Prof. Dr. Eşref ADALI - İstanbul Teknik Üniversitesi

### Alan Editörleri

Prof. Dr. Banu DİRİ - Yıldız Teknik Üniversitesi

Prof. Dr. Nevcihan DURU - Kocaeli Sağlık Üniversitesi

Prof. Dr. Sedat AKLEYLEK- Ondokuz Mayıs Üniversitesi

Prof. Dr. Tunga GÜNGÖR - Boğaziçi Üniversitesi

Doç. Dr. Mehmet KARAKÖSE - Fırat Üniversitesi

Doç.Dr. Hidayet TAKÇI- Cumhuriyet Üniversitesi

Dr. Burcu YILMAZ - Gebze Teknik Üniversitesi

### Adres:

Bilgisayar Bilimleri ve Mühendisliği Dergisi  
Darüşşafa Cad. 43L D:3 İstinye, Sarıyer-İstanbul  
E-posta: AkademikBilisimVakfi@gmail.com

## YAYIN KURULU

- Prof. Dr. İlyas Çiçekli - Hacettepe Üniversitesi  
Prof. Dr. Efendi Nasibođlu - Dokuz Eylül Üniversitesi  
Prof. Dr. Şule Gündüz Öğüdücü - İstanbul Teknik Üniversitesi  
Prof. Dr. Nizamettin Aydın - İstanbul Teknik Üniversitesi  
Prof. Dr. M. Halit S. Oğuztüzün - Ortadođu Teknik Üniversitesi  
Prof. Dr. Kemal Bıçakçı - İstanbul Teknik Üniversitesi  
Prof. Dr. Şeref Sađırođlu - Gazi Üniversitesi  
Prof. Dr. Erdal Kılıç - 19 Mayıs Üniversitesi  
Prof. Dr. N. Yasemin Topalođlu - Ege Üniversitesi  
Prof. Dr. Tolga Sakallı - Trakya Üniversitesi  
Doç. Dr. Serhan Yarkan - İstanbul Ticaret Üniversitesi  
Doç. Dr. Murat Yılmaz - Gazi Üniversitesi  
Doç. Dr. A. Hasan Koltuksuz - Yaşar Üniversitesi

## İÇİNDEKİLER

	Sayfa
Atanmış Bilgisayarlar ile Kuantum Ağların Dağıtık Benzetimi ( <i>Araştırma</i> ) Osman Semi CEYLAN, İhsan YILMAZ	90-94
Hava Trafik Yönetim Sistemleri Çekirdek Ağlarında Servis Kalitesi Temelli Ağ Trafik Yönetimi (Air-SDN) ( <i>Araştırma</i> ) Sertan PEKEL, Müge SAYIT	95-108
Derin Öğrenme ile Türkçede Adil Çözümleme ( <i>Araştırma</i> ) Mehmet TAZE, Senem KUMOVA METİN	109-119
Makine Öğrenmesi Yöntemleri ile Kan Sayımında Covid-19 Tanısı ( <i>Araştırma</i> ) Büşra ÇAKI, Ahmet EGESoy, Yasemin TOPALOĞLU	120-131
Yazılım-tanımlı İHA Ağları için Deney Ortamı Tasarımı ( <i>Araştırma</i> ) Gökhan SEÇİNTİ	132-141
Güncel Düzmece Video Algılama Yöntemleri Üzerine Bir Kaynak Derlemesi ( <i>Derleme</i> ) Suzan AYDIN, Zeki ÖZEN	142-155
Makine Öğrenmesi Temelli Obezite Durum Tahmini ( <i>Araştırma</i> ) Ercan ÖLÇER	156-164
Makine Öğrenmesi Yöntemleri Kullanılarak Kötü Amaçlı Yazılım Sınıflandırması: CIC-MamMem-2022 Veri Kümesi Üzerinde bir Başarım Karşılaştırması ( <i>Araştırma</i> ) Oğuzhan KIRLAR	165-173
Hedef Gen Çıkarımı için Makine Öğrenmesi Tabanlı Mikrodizi Verilerinin Analizi ( <i>Araştırma</i> ) Zerrin YILDIZ ÇAVDAR, Ahmet SERTBAŞ, Tolga ENSARİ	174-183



# Atanmış Bilgisayarlar ile Kuantum Ağların Dağıtık Benzetimi

## Distributed Simulation of Quantum Networks with Embedded System Devices

Osman Semi CEYLAN  
Çanakkale Onsekiz Mart Üniversitesi  
Bilgisayar Mühendisliği  
Çanakkale - Türkiye  
osman.semi.ceylan@gmail.com  
ORCID: 0000-0002-0442-0913

İhsan YILMAZ  
Çanakkale Onsekiz Mart Üniversitesi  
Bilgisayar Mühendisliği  
Çanakkale - Türkiye  
iyilmaz@comu.edu.tr  
ORCID: 0000-0002-0442-0913

### Öz

Günümüz kuantum ağ yapısı mimari bakımdan klasik ağ yapıları referans alarak kurulmuştur. Fakat kuantum ağ oluşturan aygıtların parçacık kontrol mekanizmaları nedeniyle klasik aygıtlara göre daha maliyetlidir. Deneysel bir kuantum ağın gerçekleştirme maliyetini arttıran bu sebepten dolayı bir kuantum ağ kurmadan önce olurluk benzetimleri yapılmaktadır. Fakat kuantum dolaşıklığın mevcut benzetim yöntemlerindeki üstel artan veri depolama kapasitesi gerektirdiğinden dolayı güçlükler yaşanmaktadır. Bu çalışmada bu probleme bir çözüm olarak kuantum ağların deneysel olarak birden fazla klasik cihaz yardımıyla dağıtık benzetim kullanarak daha az maliyetli olarak nasıl gerçekleştirilebileceği önerilmektedir. Bu amaç için geliştirilen eklenti yazılımı ile farklı ağ senaryolarında elde edilen sonuçlar dağıtık benzetim modelinin mevcut modellere göre daha etkin olduğunu göstermektedir.

**Anahtar sözcükler:** Kuantum Ağlar, Dağıtık Benzetim Modeli, Kuantum Benzetim

### Abstract

Today's quantum network structure was established architecturally by taking classical network structures as a reference. However, devices that create quantum networks are more costly than classical devices due to their particle control mechanisms. For this reason, which increases the cost of implementing an experimental quantum network, feasibility simulations are performed before establishing a quantum network. However, difficulties arise because quantum entanglement requires exponentially increasing

data storage capacity in existing simulation methods. In this study, as a

solution to this problem, it is suggested how quantum networks can be experimentally implemented in a less costly manner using distributed simulation with the help of multiple classical devices. The results obtained in different network scenarios with the plug-in software developed for this purpose show that the distributed simulation model is more effective than existing models.

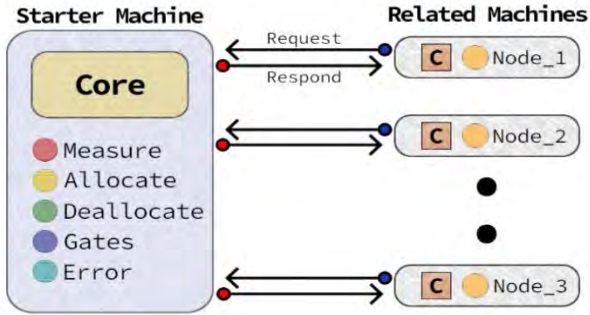
**Keywords:** Quantum Networks, Distributed Simulation Model, Quantum Simulation

### 1. Giriş

Yarı iletken teknolojilerinde yaşanan gelişmelere paralel olarak araştırmacılar mikro ölçekli sistemler için boyut olarak daha küçük, ölçüm ölçütlerinde daha hassas ve kararlı çalışan aygıtlar üretmişlerdir. Bu gelişmelerin kuantum teknolojilerdeki yansımalarına bir örnek olan kuantum belleğin [1] geliştirilmesi yeni kuantum algoritmalar ve aygıtlar geliştirilmesine yol açmıştır. Böylelikle kuantum ağların yapıtaşısı olan kuantum yönlendirici ve yineleyici aygıtlar için deneysel çalışmalara ve yeni yöntemlerin geliştirilmesine olanak sağlamıştır [2-7]. Yerel olmayan kuantum bellek içeren iki kuantum aygıtın fotonik kanallar gibi iletişim kanalları [8-9] kullanarak bağlanması sonucunda deneysel kuantum ağlar üzerinde çalışmaların yapılması mümkün olmuştur. Bu gelişmelerin yanı sıra yerel kuantum işleme [10-12] ve çeşitli kuantum hata önleme [13, 15] tekniklerindeki gelişmeler ardından kuantum internet [16] kavramı ortaya atılmıştır. Kuantum internet birbirleri arasında yerel olmayan düğümler içeren bir topolojiden oluşan ve kuantum ağ aygıtlarının yanında klasik kontrol mekanizmaları içeren bir kuantum ağ



gereken verilere erişmesine olanak tanır. Benzetim esnasında kuantum dalga fonksiyonu içeren veri-kuantum kaynaklar başka süreçlere ile bağımlılık gösterir ancak bu bağımlılık senaryoyu başlatan düğümlerin süreçleri ile sınırlıdır. Bu nedenle, bir kuantum ağı benzetiminde, merkezi benzetim modelindeki potansiyel darboğaz, birkaç yerel kuantum kaynağının aynı anda dağıtılmasıyla çözülür.



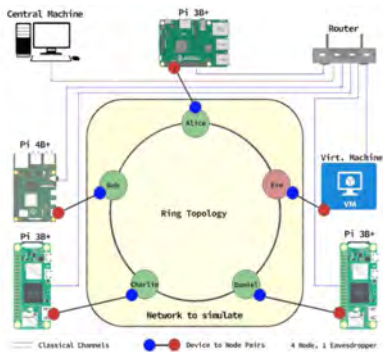
Şekil-2: Dağıtık bir benzetim senaryo esnasında eşleşmiş tarafların görevleri gösterilmektedir [29].

Bu çalışmada önerilen dağıtık benzetim modelini kuantum benzetimlerine entegre etmek için tarafımızdan geliştirilen QDNS kuantum simülatörü seçildi. Dağıtık benzetim geliştirilmesi, QDNS kuantum ağ simülatörünün bir uzantısı olarak uygulandı.

Bu dağıtık benzetimi gerçekleştirilmesini sağlayan eklenti QDNS'in yazılımsal çatı katmanında çalışmaktadır. Bu yöntemde her cihaz, kendine ait bir alıcısı olmasa bile, ağdaki ilgili düğümü taklit eden aygıtta gelen istekleri ve giden yanıtları iletme yeteneğine sahiptir.

#### 4. Sınamalar

Test ortamı 1 adet Raspberry 4B+, 1 adet Raspberry 3B ve 2 adet Raspberry Zero ve 1 adet ARM tabanlı sanal aygıttan oluşmaktadır. Ayrıca merkezi cihaz olarak bir masaüstü bilgisayar yapılandırılmıştır. Şekil-3'te test için kurulumu yapılan aygıtlar ve kurulan yerel ağ gösterilmektedir.



Şekil-3: Dağıtık benzetim için hazırlanan test aygıtlarını, ağ topolojisini ve sorumlu oldukları düğümleri göstermektedir [29].

Güvenli iletişim bağlamında kuantum mekaniği ilkelerini kullanan birçok kuantum iletişim modeli geliştirilmiştir. Bu iletişim modelleri kaynaklarda genelde BB84 [30] ve E91 [31] gibi iletişim protokolü olarak adlandırılmaktadır. Bir kuantum ağ senaryosu ise birbirini takip eden bir dizi protokollerden oluşmaktadır. Bu senaryolar, aşağıda belirtildiği gibi BB84 ve E91 kuantum ağ dağıtım protokollerinin çeşitli sıralarda

uygulanmasını içermektedir. Bu senaryolar BB84 ve E91 kuantum ağ dağıtım protokollerinin çeşitli sıralarda uygulanmasını içermektedir.

• **Ayrık senaryo:** Bu senaryoda benzetim başladığı anda Alice ile Daniel aralarında BB84 ve Bob ile Charlie aralarında E91 protokolü kullanarak anahtar üretimi yapacaktır. Tüm düğümler aynı anda en fazla bir çeşit senaryo gerçekleştirme yetkisine sahiptir [29].

• **Ardışık senaryo:** Benzetim sırasında Alice ve Bob, Bob ve Charlie, Charlie ve Daniel ve Daniel ve Alice adlı düğüm çiftleri BB84 veya E91 protokollerini rastgele yürütmektedir. Bu senaryoda, ağdaki tüm düğümler, ayrık senaryonun aksine birden fazla çeşit senaryo gerçekleştirme yetkisine sahiptir [29].

• **Bileşik senaryo:** Alice, benzetim sırasında Bob, Charlie ve Daniel'in de dahil olduğu BB84 ve E91 protokollerini eşzamanlı ve rastgele uygulamaktadır. Bu senaryo, kuantum kaynaklarının birleşik bir düğümde birleştirildiği durumu incelemektedir [29].

Test sırasında protokol tarafları her senaryoda 30 adet QKD protokolü çalıştırmaktadır. Böylelikle, benzetim sonucunda ayrık senaryonun toplam 60 adet tamamlanmış QKD protokolü, ardışık senaryonun 120 adet tamamlanmış QKD protokolü ve bileşik senaryonun 90 adet tamamlanmış QKD protokolü gerçekleştirmektedir. Eve düğümü ise her bir senaryoda saldırı görevini yürütmektedir [29].

Çizelge-1: Hazırlanan test düzeneğinde bulunan aygıtların bireysel performansı

Donanım	Kaynak Oluşumu	Operatör Uygulama	Durum Hazırlama	Ölçüm
Pi 4B+	1,000	0.6641	0.4912	0.1331
Pi 3B+	0.7406	0.4932	0.3793	0.0992
Pi Zero v2	0.5262	0.3514	0.2624	0.0702
Sanal Makine	0.4547	0.3026	0.2277	0.0651

Simüle edilen senaryolardaki işlem yükleri kaynak üretimi, durum hazırlığı, kapı uygulaması ve ölçüm olarak kategorize edilmektedir. Bu kategoriler, kuantum kaynak yönetimiyle bağlantılı performans ölçümleri üzerinde en büyük etkiye sahiptir. Bunların yanı sıra simüle edilen kuantum ağın iletişim kanal gürültü desenleri ve kuantum protokol sonlandırma eylemleri de etki eden diğer faktörlerdir. Elde edilen bulgular Tablo-1'de sunulmaktadır. Tüm sonuçlar sistemdeki en güçlü cihaz olan Pi 4B+ cihazının kaynak üretme performansına göre ölçeklendirilmiştir. Çizelge-1'den görülebileceği gibi aygıtların birçok parametrede işlem gücüyle doğrudan orantılı sonuçlar sergilediği görülmektedir.

Çizelge-2: Hazırlanan senaryoların çalışma zamanı (saniye)

Senaryo	Merkezi Model	Dağıtık Model	Dağıtık Model (Ağ gecikmesiz)
Ayrık Senaryo (60 QKD)	43.1	26.6	22.3
Ardışık Senaryo (120 QKD)	89.0	32.6	26.2
Bileşik Senaryo (90 QKD)	65.7	97.7	69.2

Test senaryolarına ait sonuçlar Çizelge-2 ile sunulmaktadır. Elde edilen test sonuçları dağıtık benzetim modelinde ayrı ve sıralı senaryolarının merkezi benzetim modeline kıyasla daha hızlı sonlandığını göstermektedir. Bileşik senaryo ise dağıtık benzetim modelinde merkezi benzetim modeline göre daha geç sonlanmaktadır. Aygıtlar arasındaki iletişim ağı gecikmeleri hariç tutulduğunda, bileşik senaryonun dağıtık ve merkezi benzetim modelinde birbirine yakın sürede sonlandığı görülmektedir [29].

Elde edilen sonuçlar, orta ve büyük ölçekli kuantum ağların benzetimi için bilgisayarların paralel işlem gerçekleştirme yeteneğini dağıtık benzetim modelinde kullanarak kuantum ağların klasik aygıtlar yardımıyla daha az maliyet ile kurgulanabileceğini göstermektedir.

Bu çalışma Ankara VIII. Uluslararası Bilimsel Araştırmalar Kongresi'nde sözlü sunumu yapılmıştır ve bildiri kitapçığında özeti yayınlanmıştır [32].

## Teşekkür

Hakemlere değerli görüşler ve önerilerinden dolayı teşekkür ederiz.

## Kaynakça:

- [1] Overtchayapong, P., Lee, K. W., Myers, B. A., ve Jayich, A. C. B., *Dynamic strain-mediated coupling of a single diamond spin to a mechanical resonator.*, Nature communications, 2014, 5(1).
- [2] Sangouard, N., Simon, C., De Riedmatten, H., ve Gisin, N., *Quantum repeaters based on atomic ensembles and linear optics.*, Reviews of Modern Physics, 2011, 83(1), 33-80.
- [3] Briegel, H. J., Dür, W., Cirac, J. I., ve Zoller, P., *Quantum repeaters: the role of imperfect local operations in quantum communication.*, Physical Review Letters, 1998, 81(26), 5932.
- [4] Zukowski, M., Zeilinger, A., Horne, M., ve Ekert, A., *"Event-ready-detectors" Bell experiment via entanglement swapping.*, Physical review letters, 1993, 71(26).
- [5] Dür, W., Briegel, H. J., Cirac, J. I., ve Zoller, P., *Quantum repeaters based on entanglement purification.*, Physical Review A, 1999, 59(1), 169.
- [6] Yuan, Z. S., Chen, Y. A., Zhao, B., Chen, S., Schmiedmayer, J., ve Pan, J. W., *Experimental demonstration of a BDCZ quantum repeater node.*, Nature, 2008, 454(7208), 1098-1101.
- [7] Zhao, B., Chen, Z. B., Chen, Y. A., Schmiedmayer, J., ve Pan, J. W., *Robust creation of entanglement between remote memory qubits.*, Physical review letters, 2007, 98(24), 240502.
- [8] Van Enk, S. J., Cirac, J. I., ve Zoller, P., *Photonic channels for quantum communication.*, Science, 1998, 279(5348), 205-208.
- [9] Gisin, N., ve Thew, R., *Quantum communication.*, Nature photonics, 2007, 1(3), 165-171.
- [10] Spiller, T. P., ve Munro, W. J., *Towards a quantum information technology industry.*, Journal of Physics: Condensed Matter, 2005, 18(1), V1.
- [11] Dowling, J. P., & Milburn, G. J., *Quantum technology: the second quantum revolution.*, Philosophical Transactions of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences, 2003, 361(1809), 1655-1674.
- [12] Lloyd, S., *Universal quantum simulators.* Science, 1996, 273(5278), 1073-1078.
- [13] Hartmann, L., Kraus, B., Briegel, H. J., ve Dür, W., *Role of memory errors in quantum repeaters.*, Physical Review A—Atomic, Molecular, and Optical Physics, 2007, 75(3), 032310.
- [14] Dür, W., ve Briegel, H. J., *Entanglement purification and quantum error correction.*, Reports on Progress in Physics, 2007, 70(8), 1381.
- [15] Lafiamme, R., Miquel, C., Paz, J. P., ve Zurek, W. H., *Perfect quantum error correction code.*, Physical Review Letters, 1996, 77(1), 198-201.
- [16] Kimble, H. J., *The quantum internet.*, Nature, 2023, 453(7198), 1023-1030.
- [17] Elliott, C., Colvin, A., Pearson, D., Pikalo, O., Schlafer, J., ve Yeh, H., *Current status of the DARPA quantum network.*, Quantum Information and computation III, 2005, Vol. 5815, pp. 138-149.
- [18] Peev, M., Pacher, C., Alléaume, R., Barreiro, C., Bouda, J., Boxleitner, W., ve Zeilinger, A., *The SECOQC quantum key distribution network in Vienna.*, New journal of physics, 2009, 11(7), 075001.
- [19] Sasaki, M., Fujiwara, M., Ishizuka, H., Klaus, W., Wakui, K., Takeoka, M., ve Zeilinger, A., *Field test of quantum key distribution in the Tokyo QKD Network.*, Optics express, 2011, 19(11), 10387-10409.
- [20] Chen, L., Xue, K., Li, J., Yu, N., Li, R., Sun, Q., ve Lu, J., *SimQN: A network-layer simulator for the quantum network investigation.*, IEEE Network, 2023, 37(5), 182-189.
- [21] Coopmans, T., Kneijens, R., Dahlberg, A., Maier, D., Nijsten, L., de Oliveira Filho, J., ve Wehner, S., *Netsquid, a network simulator for quantum information using discrete events.*, Communications Physics, 2021, 4(1), 164.
- [22] Satoh, R., Hajdušek, M., Benchasattabuse, N., Nagayama, S., Teramoto, K., Matsuo, T., ve Van Meter, R., *Quisp: a quantum internet simulation package.*, In 2022 IEEE International Conference on Quantum Computing and Engineering (QCE) (pp. 353-364). IEEE.
- [23] DiAdamo, S., Nötzel, J., Zanger, B., ve Beşe, M. M., *Qunetsim: A software framework for quantum networks.*, IEEE Transactions on Quantum Engineering, 2021, 2, 1-12.
- [24] Joubert, T., Hodson, D. D., ve Grimaila, M. R., *An Examination into SQUANCH and its Conversion to Julia.*, In 2023 Congress in Computer Science, Computer Engineering, & Applied Computing (CSCE) (pp. 772-779). IEEE.
- [25] Ceylan, O. S., ve Yılmaz, İ., *QDNS: Quantum Dynamic Network Simulator Based on Event Driving.*, In 2021 International Conference on Information Security and Cryptology (ISCTURKEY) (pp. 45-50). IEEE.
- [26] Smith, K. N., Perlin, M. A., Gokhale, P., Frederick, P., Owusu-Antwi, D., Rines, R., ve Chong, F., *Clifford-based circuit cutting for quantum simulation.*, In 2023 Proceedings of the 50th Annual International Symposium on Computer Architecture (pp. 1-13).
- [27] Dahlberg, A., ve Wehner, S., *SimulaQron—a simulator for developing quantum internet software.*, Quantum Science and Technology, 2018, 4(1), 015001.
- [28] Wu, X., Kolar, A., Chung, J., Jin, D., Zhong, T., Kettimuthu, R., ve Suchara, M., *SeQeNCE: a customizable discrete-event simulator of quantum networks.*, Quantum Science and Technology, 2021, 6(4), 045027.
- [29] Ceylan, O. S., ve Yılmaz, İ., *Kuantum ağ benzetimlerinin dağıtık yürütülmesi.*, Çanakkale Onsekiz Mart Üniversitesi, Lisansüstü Eğitim Enstitüsü, Çanakkale, Türkiye, 2024.



- [30] Shor, P. W., ve Preskill, J., *Simple proof of security of the BB84 quantum key distribution protocol*, Physical review letters, 1984, 85(2), 441.
- [31] Ekert, A. K., *Quantum cryptography based on Bell's theorem.*, Physical review letters, 1991, 67(6), 661.
- [32] Ceylan, O. S., ve Yılmaz, İ., *Gömülü Aygıtlar Kullanarak Kuantum Ağların Dağıtılmış Benzetimi.*, In 2023 Ankara International Congress On Scientific Research-VIII. (pp. 762-763).

# Hava Trafik Yönetim Sistemleri Çekirdek Ağlarında Servis Kalitesi Temelli Ağ Trafik Yönetimi (Air-SDN)

## Quality-of-Service Based Network Traffic Management in Air Traffic Management System's Core Network (Air-SDN)

Sertan PEKEL  
Ege Üniversitesi  
Uluslararası Bilgisayar Enstitüsü  
İzmir - Türkiye  
sertanpekel@gmail.com  
ORCID: 0000-0003-0994-3307

Müge SAYIT  
Ege Üniversitesi  
Uluslararası Bilgisayar Enstitüsü  
İzmir - Türkiye  
muge.sayit@ege.edu.tr  
ORCID: 0000-0002-7990-1463

### Öz

Hava Trafik Yönetim (ATM) sistemleri, ticari hava taşımacılığının yanı sıra hava sahasında gerçekleşen tüm uçuşların yönetimini sağlar. Bu sistemin gerçek zamanlı, sürekli ve kritik veri barındırması nedeniyle, sistemin çekirdek ağında gerçekleşen ağ trafiğinin bu parametrelere cevap verebilecek şekilde yönetilmesi, başka bir ifade ile bu ağ yapısında belirli bir servis kalitesi (QoS) yaklaşımının uygulanması gerekmektedir. Bu çalışmada önerilen Air-SDN yöntemi, ATM sistemlerindeki çekirdek ağlarda Yazılım Tanımlı Ağ (SDN) konsepti çerçevesinde belirli bir servis kalitesi (QoS) yöntemi uygulayarak ağ trafiğinin hızlı ve güvenli yönetilmesine olanak tanımaktadır. Air-SDN yöntemi bir test ortamında benzetilmiş ve önerilen yöntemin, TCP taşıma protokolü ile %153, UDP ve Burst-UDP taşıma protokolleri için de %63 oranında veri çıkış hacmi parametresinde performans artışı sağladığı gözlenmiştir.

**Anahtar sözcükler:** Hava Trafik Yönetimi, ATM Sistemlerinde Çekirdek Ağ Trafik Yönetimi, Yazılım Tanımlı Ağlar (SDN), Servis Kalitesi (QoS)

### Abstract

Air Traffic Management (ATM) Systems provide managing all flights in the airspace including commercial air transporting. Because of this system contains real-time, consistent and critical data, the network traffic which occurs in the core network of this system should be managed to satisfy these

parameters; in other terms, a quality of service (QoS) approach should

be applied to this network structure. Air-SDN method, which is proposed in this paper makes enable to manage network traffic fast and reliable within ATM System's core network using a specified quality of service (QoS) manner in Software Defined Networking (SDN) concept. Air-SDN method is emulated in a test-bed and it has been observed that this method provides performance gain as 153% for TCP transport protocol, 63% for UDP and Burst-UDP transport protocols.

**Keywords:** Air Traffic Management, Traffic Management in ATM System's Core Network, Software Defined Networks (SDN), Quality of Service (QoS)

### 1. Giriş

Günümüzde milyonlarca yolcunun zaman tasarrufu ve güvenlik sebepleri ile havayolu ulaşımını tercih etmesi, ülkelerin toprakları üzerinde yer alan hava sahalarının da aynı kara ve deniz yolları gibi yoğun bir şekilde kullanılmasına yol açmaktadır [1]. Bununla birlikte kara ve deniz ulaşımında kullanılan taşıtların görece düşük hızları, bu taşıtları idare eden kişilerin (şoför, kaptan vb.) diğer taşıtlar ile aralarındaki güvenli mesafeyi kendilerinin ayarlayabileceği seviyededir. Ancak hava sahasını kullanan hava araçlarına ait hızların diğer ulaşım araçlarına nazaran daha yüksek oluşu, hava sahasının spesifik gözetleme araçları ile gözetlenerek trafiğinin aktif olarak yönetilmesi gerekliliğini ortaya koymaktadır.

Hava sahasında gerçekleşen trafik akışı, sivil yolcu taşımacılığı ile birlikte askeri ve sivil amaçlı eğitim, taktik, savunma,

önleme, istihbarat, müdahale, intikal, iş, sportif, zirai, tıbbi, güvenlik ve turizm amaçlı yapılan tüm uçuşları da içine alır. Bu uçuşların güvenli, düzenli ve acil bir durumda müdahale edilebilecek şekilde gerçekleşmesi, Hava Trafik Yönetimi (*Air Traffic Management – ATM*) ile mümkün olur. ATM, devletler bazında, kamu hizmeti olarak yürütülür. Ülkemizde bu sorumluluk, ulusal hava ulaşım hizmet sağlayıcısı (*Air Navigation Service Provider – ANSP*) misyonu ile Ulaştırma ve Altyapı Bakanlığı'nın ilgili kuruluşu olan Devlet Hava Meydanları İşletmesi (DHMI) Genel Müdürlüğü tarafından üstlenilmiştir.

ATM kullanıcıları (hava sahasının yönetiminden sorumlu hava trafik kontrolörleri, uçuş planlamalarından sorumlu havacılık bilgi yönetimi uzmanları, kullanılan teknik altyapıdan sorumlu kişiler ve uçuş düzenleyen şahıs, kurum ve kuruluşlar vb.), coğrafi anlamda farklı konumlarda olmaları sebebiyle fiziksel ve mantıksal olarak birbirine bağlı bir bilgisayar ağ sistemi (ATM sistemi) ile iletişim kurarlar. Günümüzde pek çok ATM sistemi geleneksel ağ yönlendirme yapısı baz alınarak tasarlanmıştır. Bu sistemlerde ağ yönlendirme yapısını ve ağ mimarisini etkileyen köklü değişikliklere gitmek, taşınan verinin gerçek zamanlı, sürekli ve kritik yapısı nedeniyle ATM ağını kuran ve işleten devletler açısından çok tercih edilmemektedir. Böyle bir değişikliğin sistemleri tamamen devre dışı bırakarak çekirdek ağdan uç birimlere, duvarlara ve antenlere doğru yapılması gerekmektedir. ATM sisteminin -anlık da olsa- devre dışı kalması hava sahasının ve uçuşların güvenliğini doğrudan etkilediğinden, sistemin devre dışı kaldığı zaman aralığında yedek bir ATM sisteminin kullanılması elzemdir. Bu sebeple dünya üzerinde kullanılan hemen hemen tüm ATM sistemleri geleneksel ağ yapısı üzerine tasarlanmıştır ve bu sistemlerde ağ yönlendirme yapısı ya da ağ mimarisinde (bu çalışmada önerilen yöntem gibi) köklü bir değişiklik yapmak, aktif olarak çalışacak yedek bir ATM sistemi gerektirmektedir.

Yazılım Tanımlı Ağlar (*Software Defined Networks – SDN*), bilgisayar ağlarındaki kontrol ve veri düzlemlerini birbirinden ayırarak, kontrol düzleminin esnek ve ihtiyaca özgü tasarlanabilmesini sağlayan bir teknolojidir. Sahip olduğu avantajlar sebebiyle, ATM sistemlerinde kullanımı bu sistemlerin ihtiyacı olan gerçek zamanlı, kritik ve sürekli iletişimi mümkün kılabilir.

Mevcut ATM sistemlerinde servis kalitesi (*Quality of service – QoS*), genellikle veriyi taşıyan kuruluşa (çoğunlukla ulusal iletişim kuruluşu – *national carrier*) bırakılmış, bu kuruluşlar ile imzalanan hizmet seviyesi anlaşması (*Service Level Agreement – SLA*) ile belirli bir seviyede sağlanmaya çalışılmıştır. Böylelikle ATM sistemlerinde QoS sağlama genellikle uç ağda hava ulaşımı hizmet sağlayıcısının kontrolünde olmakta, çekirdek ağda ise veriyi taşıyan kuruluşa bağımlılık sebebi ile genellikle bir QoS sağlama çabası güdülmemektedir. Bu sebeple bu çalışmanın motivasyonu, SDN konseptinin 'programlanabilir' yetisini ve OpenFlow protokolünün güçlü akış kontrolü kabiliyetini çeşitli parametreler üzerinden kullanarak bir ATM sistemi çekirdek ağında QoS sağlama olarak açıklanabilir.

ATM Sistemlerinde SDN ve NFV (*Network Functions Virtualization – Ağ Fonksiyonları Sanallaştırma*) kavramları, ilk kez 2019 yılında Brezilya'nın hava ulaşım hizmet sağlayıcı (ANSP) kuruluşu olan (*Department of Airspace Control – DECEA*) tarafından ATM sistemlerine entegre edilmiş ve daha önce dengersiz seviyedeki servis kalitesinin arttığı ve stabil hale geldiği kuruluş tarafından açıklanmıştır [2].

Bu çalışmada bir ATM sistemi çekirdek ağında, geleneksel ağ yönlendirme yapısı yerine yazılım tanımlı ağ (SDN) konsepti uygulanarak çekirdek ağdaki veri trafiğinin uyarlamalı bir servis kalitesi temelli yönetilmesi ele alınacaktır. Önerilen yöntem, sanal bir ağ ortamında benzetimlenecektir. Bildiğimiz kadarıyla bu çalışma, ATM sistemleri için SDN teknolojisini kullanarak QoS sağlama amaçlı ilk akademik çalışmadır. Çalışmanın katkıları şu şekilde listelenebilir:

- SDN ve NFV teknolojilerini kullanarak ATM ağlarının karakteristiklerine uygun bir QoS çözümü geliştirilmiş ve yapılan deneyler ile bu çözümün uygulanabilirliği gösterilmiştir.
- ATM ağlarında QoS sağlamak amacıyla geliştirilen yaklaşım, her bir akış için dinamik kuyruk yönetimini sağlamaktadır. Bu özelliği ile literatürden farklılaşarak, mevcut çalışmalarda bulunan sınıf bazlı kuyruk atama yaklaşımından daha detaylı ve gerçek zamanlı bir yaklaşım sunmaktadır.
- Gerçekleştirilen deneylerde, önerilen yaklaşımın performansı TCP, UDP ve Burst-UDP trafiği ile ölçümlenerek, farklı iletim katmanlarına ait performansı irdelenmiştir.

Çalışmanın ilerleyen bölümlerinde teorik arka plan ve ilgili çalışmaların tanıtılması, Air-SDN yönteminin açıklanması, simülasyon ve performans detaylarının belirtilmesi ile sonuçların değerlendirilmesi ve tartışma kısımları yer almaktadır.

## 2. Teorik Arka Plan ve İlgili Çalışmalar

### 2.1 Hava Trafik Yönetimi (ATM) Sistemleri

Hava Trafik Yönetimi (ATM), hava sahası ve kullanıcılarının tamamına etkin hizmet verecek şekilde yapılandırılmış bir ATM sistemi ile donatılmış olmalıdır. Bir ATM sistemi, hizmet verdiği hava sahasındaki iletişim, ulaşım ve gözetleme (*Communication, Navigation, Surveillance – CNS*) hizmetlerini ve bu hizmetlerin arasındaki ilişkileri kapsar ve yönetir.

ATM sistemlerinin birden fazla hizmet ve altyapıyı kapsamaları ve yönetebilmesi ise üretilen gerçek zamanlı verinin vakit geçirilmeden ATM çekirdek ağındaki sunucularda işlenmesi, kaydedilmesi ve işlenen verinin gecikme olmaksızın muhataplarına aktarılabilmesi ile gerçekleşir. Bunun için de sağlam, sürekli, hızlı ve bütüncül veri akışı sağlayabilecek bir bilgisayar ağının, hava sahası kullanıcıları arasında tesis edilmiş olması gerekmektedir. Hava trafiğinin emniyetli ve hızlı yönetilebilmesi, uçuşların başladığı ve bittiği havaalanları (*aerodromes*), hava sahasının yönetildiği hava ulaşım üniteleri (kule, yaklaşma, saha kontrol birimleri ve uçuş bilgi merkezleri [*flight information center – FIC*]) ile uçuşların gerçekleştiği

hava yolu ve tanımlanmış hava sahalarındaki (*designated airspace*) CNS hizmetlerinin aksamadan sağlanması ve bu birimlerin birbirleri ile anlık ve doğru koordine kurabilmeleri ile mümkün olmaktadır. Bu da tüm uç birimlerden (havalimanları, bağımsız oluşturulan hava ulaşım ve komşu ATM üniteleri vb.), anten ve duyarılardan (radar ve meteorolojik gözetim aygıtları) ve hava araçları ile iki yönlü telsiz iletişimi için kullanılan hava-yer frekansı antenlerinden alınan uçuş bilgi (*Flight Information Data – FID*), uçuş gözetleme (*Flight Surveillance Data – FSD*), hava sahası yönetim mesajı (*Airspace Management Message – ASM*), frekans ve telefon konuşmaları (*Voice-over-IP – VoIP*) ile hava ulaşım merkezlerinde kullanılan gözetleme ekran görüntüsü verisinin gerçek zamanlı olarak çekirdek ağa iletilmesi ve çekirdek ağda işlenen ve kaydedilen verinin yine gerçek zamanlı olarak bu veriyi kullanacak uç birimlere, komşu ATM ünitelerine ve hava araçlarına gönderilmesi ile mümkün olmaktadır.

Bir ATM sisteminin çekirdek ağı, kendisine ulaşan her tür veriyi işleyebilecek, kaydedebilecek ve gerektiğinde işlenen veriyi vakit geçirmeden muhataplarına gönderebilecek komponentleri barındırmalıdır. Frekans verisi (hava araçları ve hava ulaşım ünitelerinin iki yönlü telsiz iletişimi) ve hava ulaşım ünitelerinin kendi aralarındaki koordineler için kullandıkları telefon haberleşmesi verisini işleyen ve kaydeden bir VoIP sistemi, gözetleme duyarılardan alınan ham veriyi işleyen ve hava aracının konumunun dört boyutlu düzlemde izlenmesini sağlayan bir gözetleme verisi işleme (*Surveillance Data Processing – SDP*) sistemi, uçuşla ilgili her türlü bilginin kayıt ve kontrol edilebilmesini sağlayan bir uçuş bilgi işleme (*Flight Data Processing – FDP*) sistemi, hava sahası ile ilgili tüm mesaj ve parametrelerin kayıt ve kontrol edilebilmesini sağlayan hava sahası yönetim (*Airspace Management – ASM*) sistemi, tüm hava ulaşım ünitelerinde takip edilen ve girdi yapılan anlık hava resminin akış videosu halindeki medyasını kaydedip arşivleyecek bir kayıt ve çalma (*Recording and Playback – RNP*) sistemi ile bu sistemleri birbirlerine ve uç birimlere bağlayacak ağ yönlendirme elemanları bu bileşenlere örnek verilebilir.

## 2.2 SDN ve QoS

SDN, klasik ağ sistemlerinde ağ trafiğini yönlendiren cihazlarda birlikte bulunan yönlendirme karar mekanizmasını yöneten kontrol düzlemi ve yönlendirme işlemini gerçekleştiren veri düzleminin birbirinden ayrılması ve kontrol düzleminin tek elden yazılımsal olarak yönetilebilmesini sağlayan yeni bir ağ konseptidir. Bu sayede klasik ağ yapısında üretici bağımlı, düşük seviyede ve esnek olmayan bir şekilde yönetilebilen yönlendirme cihazları basit birer yönlendirme cihazı haline gelmekte, yönlendirme karar mekanizması ise yazılımsal fonksiyon yürütebilme yetisi kazanmış şekilde tek cihazda (genellikle bir SDN denetleyicisi (*SDN Controller*)) toplanabilmektedir.

Kontrol düzleminin veri düzleminde ayrılması ile birlikte ağ zekası ve ağ durumu mantıksal olarak merkezi hale gelmekte, ayrıca ağ kullanıcıları ve uygulamalar ağ donanımından soyutlanmaktadır [3]. Dolayısıyla SDN, ağ yöneticilerinin paket yönlendirmesini çeşitli uygulamaların ihtiyaçlarına göre

donanımdan ve cihaz üreticisinden bağımsız olarak programlayabilmesi kolaylığı getirmekle birlikte, klasik ağ yönetiminde karşılaşılan zorlukları da maliyetsiz bir şekilde aşabilen bir paradigma olarak karşımıza çıkmaktadır.

SDN konseptinde kontrol düzlemi ve veri düzlemi, SDN teknolojisinin 'yazılımsal' ilkesini gerçekleştirmek için ortak bir protokolle buluşmak zorundadır. Bu sebeple bu düzlemlerin birbirleri ile iletişim kurduğu güney köprü arayüzü (*southbound interface – SBI*) ya da *control-data plane interface – CDPI*) için ölçünlü protokoller bulunmaktadır. Bu protokollerin en bilinen ve en yaygın kullanılanı OpenFlow'dur [4].

İnternet başta olmak üzere pek çok klasik ağ sistemi, en-iyi-çaba modelini temel alarak çalışmaktadır. En iyi çaba modeli, bir ağın kullanıcılarına sunabileceği ağ servisleri (bant genişliği, paket teslim garantisi, sıralı paket iletimi, zamanlı iletim vb. için herhangi bir garanti vermeden, o anki durumuna göre yapabileceğinin en iyisini yapmayı taahhüt etmesi olarak tanımlanabilir. Ancak, kullanıcıların ağdan beklediği performans düzeyi bu şekilde karşılanamayabilir. Bu sebeple bir ağın sunabileceği genel servisler için servis kalitesi (QoS) yaklaşımı benimsenmiştir. QoS, ağ kullanıcıların tükettikleri veri bazında, ilgili ağdan performans beklenti düzeyleri olarak tanımlanabilir.

Bilgisayar ağları için QoS, ağ yöneticisinin bir ağda taşınan verinin türüne, sınıfına, büyüklüğüne, kaynak ve varış noktalarına ya da kullanıcı isterlerine göre belirlediği veri iletim önceliklendirmesi olarak tanımlanabilir. Klasik ağ sistemlerinde QoS, yönlendirme cihazlarının yönetiminin esnek olmaması sebebi ile kısıtlı düzeyde uygulanabilmiştir. SDN konsepti ile birlikte yönlendirme cihazlarının ve dolayısı ile trafik akışının yazılımsal düzeyde esnek olarak yönetilebilmesi, SDN yetisine haiz olan ağlarda QoS uygulanabilmesini oldukça kolay ve derin keşiflere açık bir hale getirmiştir.

Klasik ağlarda ve özellikle İnternet'te zaman içerisinde QoS yaklaşımı sağlayan algoritmalar geliştirilmiş ve benimsenmiştir. Bunlardan en önemlileri IntServ (*Integrated Service*) ve DiffServ (*Differentiated Service*)'dir [5,6]. Ancak IntServ, çok karmaşık ve ölçeklenemeyen yapısı nedeniyle, DiffServ ise kaynak rezervasyonu yönetiminde yetersiz kalmasından dolayı stabil bir QoS yaklaşımı sunamadığından günümüz talepleri açısından yetersiz kalmaktadır [7,8].

## 2.3 SDN Teknolojisi Kullanılarak Gerçekleştirilen QoS Çalışmaları

SDN teknolojisinin getirdiği esneklik ve merkezilik imkanları ile birlikte QoS sağlanması konusunda kaynaklarda pek çok önemli çalışma sunulmuştur. Bu çalışmalarda, QoS sağlamak için kimi zaman uçtan uca yolların belirlenmesi, kimi zaman kuyruk yönetimi, kimi zaman da hem yol belirleme hem de kuyruk yönetimi yaklaşımları öne sürülmüştür.

[9] çalışmasında veri merkezi ağları için yük dengeleme ve servis kalitesini (QoS) gerçekleştirmeyi amaçlayan algoritma sunulmuştur ve algoritma, talep edilen bant genişliğini göz önüne alarak 'en iyi' iletim yolunu hesaplamaktadır. Gerçek zamanlı ve uçtan uca servisler için SDN ağında bir sistem

prototipi dizayn edilen [10] çalışmasında yazarlar, en iyi bir çevrimiçi yönlendirme algoritması önermişlerdir. [11] çalışmasında SDN ağlarında QoS gerçekleştirmesini sağlamak için, şayet talep karşılanacak ise ağ yolu talebi, talep eden uygulamanın türüne göre belirlenmektedir. [12] çalışmasında ise SDN konseptini temel alan bir Nesnelerin İnterneti (IoT) mimarisi üzerinde gri delik atakları ve Dağıtık Servis Reddi (DDoS) saldırılarını saptayan bir yöntem önerilmiştir. Bu çalışmaların tamamında SDN ile QoS sağlamak için yol belirleme yaklaşımlarından yararlanılmış, kuyruklar üzerinde bir çalışma yapılmamıştır.

Vanitchasatit ve Sanguankotchakorn tarafından yapılan çalışmada, SDN kavramını temel alan bir kablosuz yerel alan ağındaki (WLAN) trafiği VoIP, video akışı ve dosya transferi (*file transfer*) olarak sınıflandırmıştır [13]. Daha sonra sınıflanan her bir veri türü için hedeflenen QoS parametreleri belirlenerek her bir veri sınıfı türünde maksimum akış trafiğine izin verecek şekilde tam sayılı doğrusal programlama (*Integer Linear Programming – ILP*) optimizasyon tekniği uygulanmıştır. Elde edilen sonuçlara dayalı olarak her bir trafik türü için gerekli olan minimum bant genişliği belirlenmiştir.

Al-Haddad ve Ark., ağırlıklandırılmış adil kuyruklama (*Weighted Fair Queue – WFQ*) tekniğini kullanarak SDN tabanlı veri kesitlerini taşıyan ağlar için yeni bir trafik şekillendirme algoritması önermişlerdir [14]. Algoritma, kaynak cihazlardan gelen veriyi DiffServ protokolü ile sınıflandırarak ilk katman anahtarlayıcılarına göndermekte ve bu anahtarlayıcılarda bir paket ayrıştırma ve yönlendirme algoritması uygulayarak veri kesitinin türüne göre veriyi ikinci katman anahtarlayıcılarına göndermektedir. İkinci katman anahtarlayıcılardan üçüncü katman anahtarlayıcılarına gönderilen veri, üçüncü katman anahtarlayıcılarında WFQ mantığını temel alan bir kuyruklandırmaya sokulmakta ve ağırlık oranına göre belirlenmiş bant genişliği atanarak alıcı cihazlara gönderilmektedir.

Li ve Ark., çalışmalarında ağdaki QoS yönetimini sağlayabilmek için SDN denetleyicisine C4.5 karar ağacı (*C4.5 karar ağacı*) uygulamışlardır [15]. Bu karar ağacının budanması ile uygulama tipleri belirlenerek her bir uygulama tipine ait QoS seviyesi hızlandırılmış yönlendirme, teslim-garantili yönlendirme ve en iyi çaba atanmaktadır. Daha sonra paket akışları ilgili anahtarlayıcı iskelesi için kuyruklandırılarak bu kuyruklara QoS seviyesine göre öncelik atanmaktadır.

Baek ve Ark., SDN ağlarında bant genişliği atanmış uygulamaların güvenilir bir servis kalitesini yakalayabileceğini gösteren bir çalışma sunmuşlardır [16]. Çalışmada Mininet, Ryu Controller ve Open vSwitch yapısında oluşturulan bir topolojide belirlenmiş dört öncelik seviyesindeki uygulamalara hiyerarşik sınıflandırılmış jeton bazlı (*Hierarchical Token Based – HTB*) kuyruklama tekniği uygulayarak, uygulamaların atanmış bant genişliğini gerçekten alıp almadıklarını ölçmüşlerdir.

Krishna ve Ark., çalışmalarında QoS ve *best-effort* olarak iki sınıfa ayırdıkları trafik akışları için atanmış bant genişliğini aşan bir veri çıkış hacmi gerçekleştiğinde servis kalitesinin (QoS) nasıl etkilendiğini araştırmışlardır [7]. Bunun için yaptıkları ilk

denemede OpenFlow protokolünün sayaç işlevini kullanarak rezerve edilen bant genişliğini aşan hacimdeki akış paketlerini düşürme yöntemi ile servis kalitesinin (QoS) korunmasını sağlamışlardır. Yazarlar ikinci denemelerinde ise bant genişliği ödünç alma yöntemini uygulayarak daha önce atanan bant genişliğini aşan akışlara ait paket kayıp oranını ölçmüşlerdir. Yazarlar ayrıca test ortamında kullandıkları sanal anahtarlayıcıların temel aldığı kuyruklama yöntemi olan Hiyerarşik Sınıflandırılmış Jeton Kovası'nın (*Hierarchical Token Bucket – HTB*) da QoS paketleri için paket kayıp oranını %0 bularak önerdikleri yöntemin üretim ağlarında kullanılabileceğini belirtmişlerdir.

Thazin ve Ark., 3 farklı servis sınıfı oluşturarak akışların bağlı olduğu sınıfa göre kuyruklar atamış, bant genişliğinin yeterli gelmemesi durumunda yolun değiştirilmesini önermişlerdir [3].

Akella ve Xiong, bulut bilişim kullanıcıları için farklı düzeydeki servis kalitesi (QoS) ihtiyacına, uçtan uca bant genişliği garanti edecek bir çalışma gerçekleştirmişlerdir [17]. Bu çalışmada da QoS sağlamak için yol seçimi ön plana çıkmaktadır. Kuyruklama için Open vSwitch tarafından desteklenen HTB ve Sınıflandırılmış Adil Sıralı Eğri (*Hierarchical Fair Sequence Curve – HFSC*) kuyruklama teknikleri kullanılmış olup, akış başına kuyruk belirleme üzerinde bir çalışma gerçekleştirilmemiştir.

[3], [8], [9], [11] çalışmaları en iyi yolu bulma üzerine yoğunlaşmıştır. Bu çalışmada temel olarak ele alınan ağ sistemlerinin (ATM ağları) kapalı sistemler olması ve ağ yollarının statik olması nedeniyle, en iyi yolu bulma metodunun bu çalışmanın problemi için bir çözüm getirmediği saptanmıştır.

[8] çalışması statik kuyruk yapısı üzerine bir yöntem önermiştir. Air-SDN yönteminde statik kuyruk yapısı değil, veri tipleri için adaptif kuyruk yapısı uygulanmıştır. [12] çalışması ise ağ güvenliği üzerine yoğunlaştığı ve Air-SDN yönteminin uygulandığı ağ ortamının kapalı bir ağ olmasından dolayı bu çalışmanın çözümüne uymamaktadır.

Literatürde SDN ve QoS ile ilgili burada bahsedilmeyen daha pek çok çalışma bulunmaktadır. Bu çalışmaların güzel bir özeti [18] çalışmasında sunulmuştur.

Buna rağmen ATM ağlarında QoS sağlanması ile ilgili literatürde kayda değer nadir çalışmalar bulunmaktadır. Bunlardan birisi, Ongkasae ve Ark. uçtan uca QoS seviyesini artırma adına ağ sanallaştırma tekniği kullanılan SWIM (*System-Wide Information Management - Sistem Seviyesinde Bilgi Yönetimi*) modeli önerdikleri bir çalışmadır [19]. Bu çalışmada SDN ya da NFV teknolojileri kullanılmamış olup, geleneksel ağ altyapısının getirdiği kısıtlar sebebiyle akış değil sınıf bazlı bir QoS modeli önermişlerdir.

Diğer bir çalışma ise Izquierdo-Zaragoza ve Ark. bir çalışmasıdır [20]. Bu çalışmada ATM ağlarının geniş ağ yapısının (WAN - *Wide Area Network*) yazılım tanımlı ve hiyerarşik olarak nasıl tasarlanabileceği ve farklı QoS profillerinin bunlara nasıl bir temel sağlayabileceği tartışılmıştır. Çalışma daha çok tasarım üzerine olduğundan ve

trafik yönetimi veri bazında ele alınmadığından bu çalışma da önerilen yöntemin problemi için bir çözüm getirmemektedir

Sonuç olarak, mevcut literatürde QoS sağlamak için önerilen yol seçim yaklaşımları ATM ağları için anlamlı olmamaktadır. Akışlara kuyruk atama yaklaşımlarında ise akış bazlı değil, sınıf bazlı atamalar yapılmış, akışlar için dinamik olarak kuyruk değişimi göz önünde bulundurulmamıştır. Bu çalışmada önerilen yaklaşımın literatürdeki çalışmalardan farkı, akış bazında kuyruk atamanın yapılması ve önceliğe göre dinamik olarak bu kuyrukların değiştirilmesinin sağlanmasıdır.

### 3. Air-SDN

#### 3.1 ATM Sistemindeki Veri Tipleri

Bir önceki başlıkta da belirtildiği üzere ATM sistemlerinde üretilen ve işlenen her türlü veri 'en iyi çaba' yaklaşımı ile taşınmayacak kadar kritik, gerçek zamanlı ve sürekli karaktere sahiptir. Bu sebeple belirli bir servis kalitesi (QoS) sağlama adına önerilen yöntem, bu verinin iletilmesi esnasında yaşanabilecek paket kayıplarına ve gecikmelere karşı duyarlı, ayrıca iletilen verinin türüne göre belirli bir veri çıkış hacmi garanti edebilen bir yöntem olmalıdır. SDN konsepti ve OpenFlow protokolü çerçevesinde uyarlamalı kuyruklama ve yük dengeleme paradigmasını ilke alan ve bu çalışmada önerilen yönetime 'Air-SDN' adı verilmiştir.

Air-SDN yönteminde öncelikle bir ATM çekirdek ağında taşınan ve işlenen veri tipleri tespit edilmiştir. Daha önce açıklandığı üzere, bir ATM ağında genel olarak beş çeşit veri tipi bulunabilir: VoIP verisi, uçuş gözetim verisi (FSD), uçuş bilgi verisi (FID), hava sahası yönetimi verisi (ASM) ve video akış verisi (*Video Stream* - VS). Bu veri tipleri, simülasyonlar için tespit edilen ortalama bant genişliği ve minimum bant genişliği ihtiyacı ve trafik yönetimindeki öncelikleri Çizelge-1'de verilmiştir.

Çizelge-1: Veri Tipleri ve Önceliklendirilmeleri

Sıra	Veri Tipi			
	Tanımı	Ort. BW	Min. BW	Önceliği
1	VoIP	25 Mbit/s	10 Mbit/s	En Yüksek
2	FSD	20 Mbit/s	10 Mbit/s	Yüksek
3	FID	15 Mbit/s	5 Mbit/s	Orta
4	ASM	5 Mbit/s	2 Mbit/s	Düşük
5	VS	40 Mbit/s	10 Mbit/s	Yok

Tespit edilen veri tiplerinde en öncelikli veri tipi frekans ve telefon haberleşmesi (VoIP) olarak belirlenmiştir. Bunun sebebi hava trafik yönetiminin (ATM) en önemli parçası olan hava araçlarına verilen kontrol hizmetinin (*Air Traffic Control Service* – ATCS) hava-yer frekansları üzerinden gerçekleşmesi ve bunlarla ilgili tüm kritik koordinelerin hava ulaşım merkezleri arasında telefon üzerinden yapılmasıdır. Bu önceliğin verilmesinde hava trafik hizmetinin herhangi bir ATM sistemi olmasa dahi sadece hava-yer muhaberesi ile sunulabileceği kuralı da göz önüne alınmıştır. İkinci öncelikli veri tipi ise uçuş gözetim verisidir (FSD). Radar antenleri ve sensörlerden alınan ham gözetim verisi, hava sahasının ve hava araçlarının dört boyutlu takibi için azami düzeyde önem

arz etmektedir. Üçüncü ve dördüncü öneme haiz veri tipleri uçuş bilgi (FID) ve hava sahası yönetim mesajlarıdır (ASM). Hava ulaşım ünitelerinde takip edilen ve üzerinde çeşitli girdiler yapılabilen anlık hava resminin çekirdek ağa aktarımı için kullanılan video akış (VS) verisine ise öncelik verilmemiştir. Bu veri tipine öncelik verilmemesinin sebebi bu verinin çekirdek ağda sadece kaydedilmesi, aynı verinin eş zamanlı olarak hizmeti sunan hava ulaşım ünitesinde de saklanmasıdır.

#### 3.2 Yöntem Detayları

Air-SDN yönteminde kullanılan uyarlamalı kuyruklama ve yük dengeleme ilkesi, ilgili veri tipine ait akış ağ üzerinde bulunduğu, ihtiyaç duyacağı bant genişliğine ve orta anahtarlayıcıların yük durumlarına göre belirlenmiş kuyrukları adaptif olarak kullanma üzerine kurulmuştur. Sistemde video akış verisi (VS) dışında kalan veri tipleri, bu çalışmada 'öncelikli veri tipleri' olarak adlandırılmıştır. Ağda bulunan akışlar, önerilen yöntemin uygulanması esnasında Çizelge-1'deki öncelik sırasına göre öncelik almaktadır.

Önerilen yönetime göre, başlangıçta, teorik olarak en yüksek bant genişliğine ihtiyaç duyacağı öngörülen video akış (VS) veri tipi en yüksek bant genişliğine sahip kuyruklara atanmaktadır. Ancak, dinamik kuyruk yapısı ile, yüksek bant genişliği sağlayabilen kuyruklar, ihtiyaç olması halinde öncelikli veri tiplerine tahsis edilebilmektedir. Bu ihtiyaç ortadan kalktığında video akış verisi (VS) yine en yüksek bant genişliği sunan kuyruklara atanır.

Önerilen Air-SDN yönteminin motivasyonu, paket iletiminde veri tiplerinin önceliklendirme sırasına göre ele alınarak ortalama bant genişliği ihtiyacına uygun bir kuyruğa atanması ve orta anahtarlayıcılarda yükün dengelenmesidir. Bunun için sunuculara oluşturulan veriye ait trafik akış istatistiklerinin, kullanılan SDN denetleyicisi tarafından okunarak ilgili veri tipi için gerekli bant genişliğinin belirlenmesi, bu değer o veri tipi için belirlenmiş olan minimum bant genişliği değerinin altına düşmeyecek şekilde ilgili trafik akış yönlendirmesinin orta anahtarlayıcılar ile ana fonksiyon anahtarlayıcı (*core function switch*) arasındaki linke bağlı çıkış portuna tanımlanmış kuyruklara yönlendirilmesinin sağlanması, bu işlem yapılırken aynı zamanda yük dengelemesi yapılabilmesi için aynı veri tipine ait akışların farklı orta anahtarlayıcılara yönlendirilmesi hedeflenmiştir.

#### 3.3 Air-SDN Algoritması

Air-SDN mimarisinde kullanılan algoritma Algoritma-1'de verilmiştir. Algoritma, SDN denetleyicisi tarafından periyodik olarak çalıştırılmaktadır. Algoritmanın çalıştırılması için bir başka yöntem, herhangi bir akış başladığında çalıştırılması olabilir. Ancak bu yöntem, akış başlangıcının tespiti için, gelen paketin SDN denetleyicisine gönderilmesini (*packet-in*) gerektirmektedir. Bu da gerçek zamanlılığı ve arzu edilen performansı olumsuz yönde etkileyen ve önerilen yöntem için gerek duyulmayan bir işlem olduğundan, algoritmanın herhangi bir akış başladığında değil, belirli periyotlar ile çalıştırılması tercih edilmiştir.

### Algoritma-1: Air-SDN Algoritması

```
1: DT ← Veri tipi listesi
2: F ← Akış (flow) listesi
3: FN ← Akış (flow) listesi (F'ten türetilmiş, kuyruk seçilmemiş, akış önceliğine göre sıralanmış)
4: Q ← Kuyruk (queue) listesi (orta switch'ler için)
5: QN ← Kuyruk (queue) listesi (Q'dan türetilmiş, herhangi bir akış atanmamış, kuyruk max_rate'e göre sıralanmış)
6: ES ← Uç switch'lerin listesi
7: PS ← VS veri tipi haricindeki (öncelikli) veri tiplerine ait uç switch'lerin listesi
8: atanmış_queue_id ← Akışların kuyruklarına ait kuyruk_id'leri
9: veri_tipi_dpidd[] ← İlgili veri tipi akışlarına atanmış switch_id'leri
10: //veri akış istatistiklerinin okunması
11: for switch s in ES do
12:   | ff.demand ← FLOW_STATS(ff),  $\forall f \in F$ 
13: end for
14: for data d in DT do
15:   for flow f in F do
16:     for switch s in PS do
17:       for queue q in Q do
18:         //kuyruğa bir akış atanmışsa ve kuyruk switch_id'si, veri tipi akışlarının atandığı switch'ler listesinde ise döngüden çık
19:         if f için atanmış_queue_id değeri varsa ve sq.switch_id ∈ veri_tipi_dpidd[] then
20:           | break
21:         end if
22:         //mevcut kuyruğun en yüksek bant genişliği parametresi akışa ait ihtiyacın üzerinde ya da eşitse ve bu değer akış için belirlenmiş en düşük bant genişliğinden yüksek ya da eşitse akışı kuyruğa ata
23:         if sq.max_rate ≥ f.demand ve sq.max_rate ≥ f.min_rate then
24:           | atanmış_queue_id ← sq.id
25:           | f.queue_id ← sq.id
26:           //yük dengeleme için ilgili veri tipine ait diğer akışların atandığı switch_id'lerinin saklanması
27:           | veri_tipi_dpidd[] ← sq.id
28:         end if
29:       end for
30:     end for
31:   end for
32: end for
33: //kuyruk seçilmemiş akışlar için döngüyü başlat
34: for flow f in FN do
35:   | for queue q in QN do
36:     | f.queue_id ← q.id
37:   | end for
38: end for
39: wait (t saniye)
```

Algoritmanın genel akışı verilerin tanımlanması, veri akış istatistiklerinin uç anahtarlayıcı okunması, iç içe döngüler vasıtasıyla her bir veri tipi, her bir akış, her bir uç anahtarlayıcı ve her bir kuyruk için değerlerin kontrol edilerek ilgili akışların ilgili kuyruklara atanması, herhangi bir akış atanmamış kuyruklara ait liste üzerinden akışlara kuyruk atanması ve belirlenmiş süre kadar (*t*) beklenerek algoritmanın tekrar çalıştırılması şeklindedir.

Bu sebeple, algoritmada öncelikle algoritmanın kullanacağı değer ve listeleri saklayacak diziler tanımlanmıştır. Algoritmanın 11. ve 12. satırlarında tüm uç anahtarlayıcılardan akış istatistikleri talep edilmekte ve akışlara ait parametre olarak kaydedilmektedir. 19. ve 20. satırlarda ilgili veri tipi için daha önce kuyruklara atanmış akışların anahtarlayıcı kimlikleri kontrol edilerek, şayet ilgili veri tipine ait uç anahtarlayıcılardan herhangi birine ait akışlara daha önce kuyruk ataması yapılmışsa son döngüden çıkılmakta ve o veri tipine ait akışı farklı bir orta anahtarlayıcıdaki (kuyrukları barındıran ve asıl yönlendirmenin yapıldığı anahtarlayıcılar) kuyruğa atanması sağlanmaktadır. Böylece algoritma aynı zamanda yük dengeleme fonksiyonunu da yerine getirmektedir. 23.-27. arasındaki satırlarda ise asıl akış atama işlemi yapılmakta ve algoritmanın kullandığı diğer değerlere ilişkin atamalar gerçekleştirilmektedir. Algoritmanın son kısmındaki 34.-36. satır aralarında ise daha önce herhangi bir kuyruğa atanmayan akışlar (öncelikli olmayan veri tiplerine ilişkin akışlar ile bir önceki kısımda kuyruğa atanmayan akışlar) kuyruğa atanmaktadır. En son olarak 39. satırda bu algoritma ile yapılan kuyruklama ve yük dengeleme konfigürasyonu ile *t* saniye boyunca devam edilebilmesi için (bir sonraki çerçeveye kadar) *t* saniyelik bekleme yapılmaktadır.

Algoritmanın zaman karmaşıklığı  $O(d.s.q.f)$  şeklinde gösterilebilir. Burada *d* (veri türü), *s* (orta anahtarlayıcıların sayısı) ve *q* (kuyrukların sayısı) belirli sayıda sabit olup, bu çarpıma *c* dersek, karmaşıklık  $O(c.f)$  olarak hesaplanabilir. *d*'nin sabit değeri 5; bir anahtarlayıcı için tanımlanabilecek kuyruk sayısı da sabit olduğundan, *c* değeri burada *s* değerine bağımlı olarak artacaktır. Bu sebeple algoritma karmaşıklığı *f* (akış sayısı) ve orta anahtarlayıcıların sayısı *s* ile sınırlıdır. Bu sebeple algoritmanın zaman karmaşıklığı doğrusaldır.

Algoritmanın bu yapısı ile yalnızca test ortamında değil, çok daha büyük ölçekteki ATM sistemi çekirdek ağlarında rahatlıkla kullanılabilir, klasik ağ sistemlerindeki yönlendirme problemlerine çözüm olabilecek bir algoritma olduğu 'Sonuçların Değerlendirilmesi ve Tartışma' bölümünde ortaya konmuştur.

## 4. Benzetim ve Başarım

### 4.1 Benzetim Sınama Test Ortamı

#### 4.1.1 Kullanılan Topoloji

Air-SDN yönteminin benzetilebileceği sinama ortamı için gerçek bir ATM sistemi çekirdek ağ topolojisi model alınarak bir topoloji oluşturulmuştur. Bu topolojide öncelikle ağa giren verinin üretildiği aygıtlar (*host*) ve onlara bağlı anahtarlayıcılar tanımlanmıştır. Her bir veri tipi için bir sunucu çifti

belirlenmiştir. Bu sunucular aynı zamanda havalimanları, hava ulaşım merkezleri, duyargalar ya da antenler olarak da düşünülebilir. Daha sonra asıl yönlendirme işlevini gerçekleştirecek orta anahtarlayıcılar tanımlanarak tüm uç anahtarlayıcılara *mesh* topoloji yöntemi ile bağlanmıştır. Ham verilerin işlenmesini ve kaydedilmesini sağlayan fonksiyonel sunucuların (VoIP server, SDP server, FDP server, ASM server ve RNP server) orta anahtarlayıcılar ile bağlantısını sağlayacak bir ana fonksiyon anahtarlayıcısı topolojiye eklenmiştir. Topolojide tüm linkler 100 MBit/s veri taşıyacak şekilde belirlenmiştir. İlgili topoloji Şekil-1'de verilmiştir.

Bu topolojide öncelikli veri tiplerini orta anahtarlayıcıları taşıyacak olan anahtarlayıcıları sayısı 8, orta anahtarlayıcıların sayısı da 3 olarak belirlenmiştir. Üretim ortamında bu sayılar ihtiyaca göre kullanıcılar tarafından tekrar düzenlenebilir. Algoritmanın bu konuda herhangi bir kısıtı bulunmamaktadır.

#### 4.1.2 Belirlenmiş Kuyruklar ve Parametreleri

OpenFlow protokolünün kuyruk yönetimi işlevi olmadığından, bu işlem simülasyon ortamı için tercih edilen sanal anahtarlayıcılar vasıtası ile yapılmıştır. Bu amaçla her bir orta anahtarlayıcıda çıkış iskelesine tanımlanan kuyruklar ve parametreleri Çizelge-1'de belirlenen veri tipleri ve parametrelerine uygun olarak Çizelge-2'de verilmiştir.

**Çizelge-2: Kuyruklar ve Atanmış Veri Tipleri**

Sıra	Kuyruk ID	Min-Rate	Max-Rate	Atanan Veri Tipi
1	q0	10 Mbit/s	40 Mbit/s	VS
2	q1	10 Mbit/s	25 Mbit/s	VoIP
3	q2	10 Mbit/s	20 Mbit/s	FSD
4	q3	5 Mbit/s	15 Mbit/s	FID
5	q4	2 Mbit/s	5 Mbit/s	ASM

#### 4.1.3 Tercih Edilen Yazılım Bileşenleri

Simülasyon ortamında kullanılacak sanal anahtarlayıcılar için Open vSwitch (OVS) tercih edilmiştir. Bu tercihin sebebi, Open vSwitch'lerin üretim ortamı düzeyinde, açık kaynak kodlu ve çok katmanlı yapıda olması ve en önemlisi her bir OpenFlow protokolüne tam destek vermesidir [21].

**Çizelge-3: Simülasyon Ortamı Yazılım Bileşenleri**

Sıra	Yazılım Bileşeni	Marka	Versiyon
1	İşletim Sistemi	Ubuntu	20.04.6 LTS (x64)
2	SDN Denetleyicisi	Ryu	4.34
3	Sanal Switch'ler	Open vSwitch	2.13.8
4	Sanal Ağ Emülatörü	Mininet	2.3.0.dev6
5	SDN SBI <sup>1</sup>	OpenFlow	1.3
6	Programlama Dili	Python	3.8.10
7	Trafik Oluşturucusu	iPerf	2.0.13

1: SBI: Güney Köprü Arayüzü

Open vSwitch'ler (OVS), OpenFlow protokolüne bağlı kuyruk yönetimi için Linux işletim sistemi çekirdeğinde bulunan HTB algoritmasını kullanmaktadır. HTB algoritmasında iletilecek paketler için, belirlenmiş oran kadar jeton bulunması halinde paketin iletilmesi -şayet kuyruk için bir maksimum oran belirlenmiş ise bu limit dahilinde iletilmesi-, değilse kuyruğa beklemeye devam etmesi esas alınır [7].

SDN denetleyicisi için Ryu Controller tercih edilmiştir [22]. Bu tercihin sebebi, Ryu'nun komponent bazlı yapısı, tamamen Python programlama dili ortamında çalıştırılabilmesi ve OpenFlow protokolünün tüm versiyonlarını desteklemesidir [23]. Ryu Controller ayrıca Open vSwitch (OVS) kuyruk yapısını yönetebilen bir uygulama programlama arayüzü (API) barındırmaktadır. Kullanılan algoritmaya bağlı olarak Ryu Controller'ın çalıştırılabileceği bir Python kodu yazılarak önerilen Air-SDN yöntemi simülasyon ortamında test edilmiştir.

Önerilen Air-SDN yöntemini sanal bir ağ ortamında test edebilmek için Mininet Network Emulator kullanılmıştır [24]. Mininet, SDN konsepti ve OpenFlow protokolünün test ortamı olarak yaygın bir şekilde kullanılmaktadır. Simülasyonlar için Mininet'in tercih edilmesinin sebebi yüksek seviyede topoloji kullanımına izin vermesi, tüm fonksiyonlarının Python programlama dili ile çalıştırılabilmesi ve işletim sistemi düzeyinde (*OS-level*) sanallaştırma uygulayabilmesidir.

Mininet ortamında gerekli topoloji, sanal anahtarlayıcılar ve SDN denetleyicisi çalıştırdıktan sonra, benzetilecek ağ trafiğini oluşturmak adına trafik oluşturucu programı olan iPerf kullanılmıştır [25]. Trafik oluşturucusu olarak iPerf'in tercih edilme sebebi, TCP ve UDP kullanan istemci-sunucu bağlantıları için güçlü bir ölçüm aracı olması, çeşitli modülasyon parametreleri ile çalışabilmesi, UDP paketleri için belirlenmiş bant genişliğinde trafik gönderebilmesi ve UDP paketleri için paket kaybı ve seğirme ölçümleri yapabilesidir.

Simülasyon ortamı için Asus marka R507U model (Intel® Core™ i7-7500U CPU @ 2,70GHZ x 4, 8 GB RAM, 1 TB M.2 NVMe SSD, NV118 / Mesa Intel® HD Graphics 620 [KBL GT2]) bir taşınabilir bilgisayar üzerine kurulu Ubuntu işletim sistemi kullanılmıştır. Simülasyon ortamında kullanılan yazılımsal bileşenler Çizelge-3'te verilmiştir.

#### 4.1.4 Sınama Parametreleri

Air-SDN yönteminin sınanması için farklı parametreler belirlenmiş ve yöntem, bu parametrelere göre sınanmıştır. Bu parametreler şu şekilde sıralanmıştır:

- 1- Air-SDN Yönteminin Kullanılmaması/Kullanılması:
  - a) No Air-SDN: Air-SDN yönteminde SDN denetleyicisi kullanmadan olmadan, orta switch'lerde veri tipleri için Çizelge-2'de verilmiş sabit kuyruklar kullanılarak),
  - b) Air-SDN: Önerilen Air-SDN yöntemini baz alan SDN denetleyicisi (*Ryu Controller*) kullanılarak.
- 2- Tekli/Çoklu Kullanıcı:
  - a) Tekli Kullanıcı: İlgili veri tipine ait tekli kullanıcıya (*host*) ait verinin ağda test edildiği,



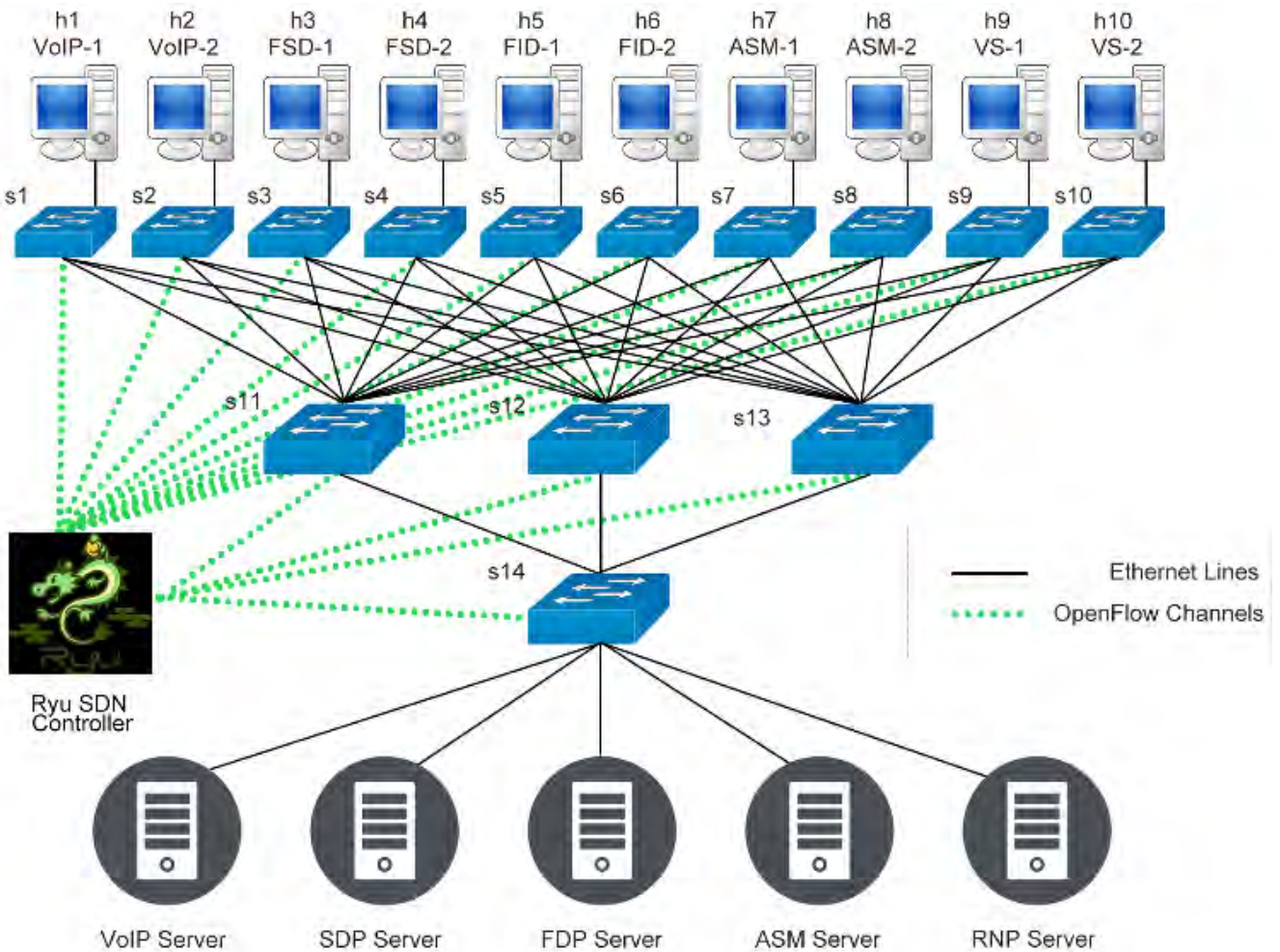
- b) Çoklu Kullanıcılı: İlgili veri tipine ait çoklu kullanıcıya (*host pair*) ait verinin ağda test edildiği.
- 3- Veri Tipi Yükü/Tam Yük:
- a) Veri Tipi Yükü: Çizelge-1’de verilmiş veri tipleri için yalnızca bir veri tipine ait verinin ağda test edildiği,
- b) Tam Yük: Çizelge-1’de verilmiş tüm veri tiplerine ait verinin aynı anda ağda test edildiği.
- 4- Taşıma Protokolü:
- a) TCP: TCP taşıma protokolü kullanılarak, 30 s süreli,
- b) UDP: UDP taşıma protokolü kullanılarak, her bir veri tipi için Çizelge-1’de verilen ortalama bant genişliği kadar trafik üretilerek, 10 s süreli,
- c) Burst-UDP: UDP taşıma protokolü kullanılarak, her bir veri tipi için 50 MBit/s kadar trafik üretilerek, 10 s süreli.

#### 4.2 Başarım Ölçümleri

Önerilen Air-SDN yöntemi ve karşılaştırıldığı No Air-SDN yöntemi tutarlı sonuçlar elde edilene kadar bir önceki başlıkta belirtilen sına ortamında ve parametrelerde benzetilmiş ve benzetime ilişkin ölçümler kaydedilmiştir. Bu benzetimlerde TCP taşıma protokolü için veri çıkış hacmi ve gecikme ölçümleri, UDP ve Burst-UDP taşıma protokolleri için de veri

çıkış hacmi, seçirme ve paket kaybı ölçümleri yapılmıştır. Bu ölçümler parametreler bazında şu şekilde sıralanmıştır:

- 1- Tekli/Çoklu Kullanıcılı, Veri Tipi Yükünde, TCP Veri Çıkış Hacmi ve Gecikme Ölçümü: Bu parametrelere ait simülasyonlar sonucunda Air-SDN yönteminin, No Air-SDN yöntemine göre öncelikli veri tiplerine daha fazla veri çıkış hacmi ve daha az gecikme sunduğu görülmektedir. Bu sonuçlar Şekil-2 ve Şekil-4’te verilmiştir.
- 2- Tekli/Çoklu Kullanıcılı, Veri Tipi Yükünde, UDP Veri Çıkış Hacmi, Seçirme ve Paket Kaybı Ölçümü: Bu parametrelere ait simülasyonlar sonucunda ölçülen veri çıkış hacminin, No Air-SDN yöntemine göre Air-SDN yönteminde tüm veri tipleri için çok küçük farklarla aynı kaldığı, seçirmenin öncelikli veri tipleri için daha az olduğu ve paket kaybının VoIP ve FSD veri tipleri için arttığı, diğer veri tiplerinde ise paket kaybı olmadığı görülmektedir. Bu artışın sebebi ‘Sonuçların Değerlendirilmesi ve Tartışma’ kısmında açıklanmıştır. Bu sonuçlar Şekil-2, Şekil-5 ve Şekil-7’de verilmiştir.
- 3- Tekli/Çoklu Kullanıcılı, Veri Tipi Yükünde, Burst-UDP Veri Çıkış Hacmi, Seçirme ve Paket Kaybı Ölçümü: Bu parametrelere ait simülasyonlar sonucunda Air-SDN yönteminin, No Air-SDN yöntemine göre tüm veri



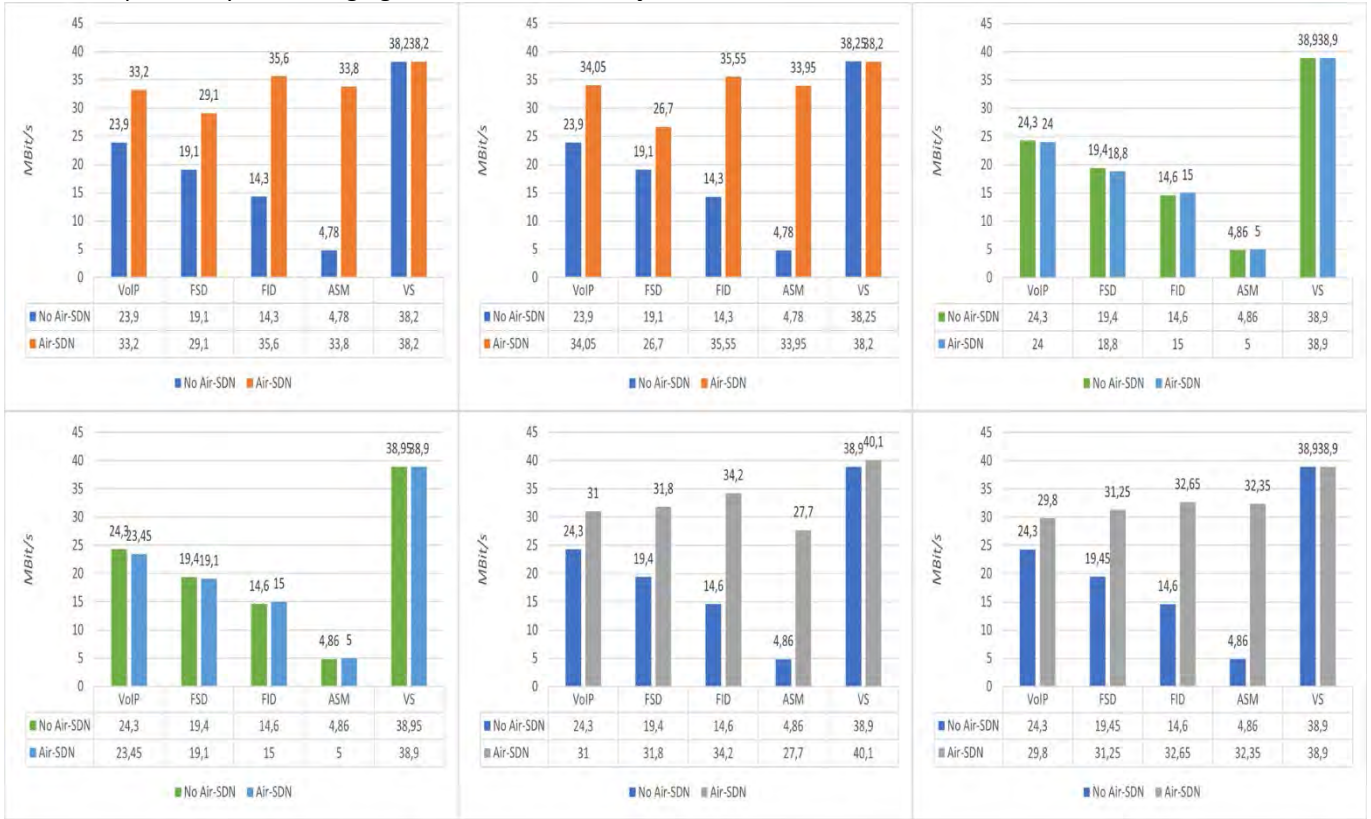
Şekil-1: Air-SDN Sınama ortamı topolojisi

tiplerine daha fazla veri çıkış hacmi sunduğu, seğirmenin VoIP veri tipi için artış gösterdiği, diğer veri tipleri için de azaldığı ve paket kaybının öncelikli veri tipleri için azaldığı görülmektedir. Bu sonuçlar Şekil-2, Şekil-5 ve Şekil-7’de verilmiştir.

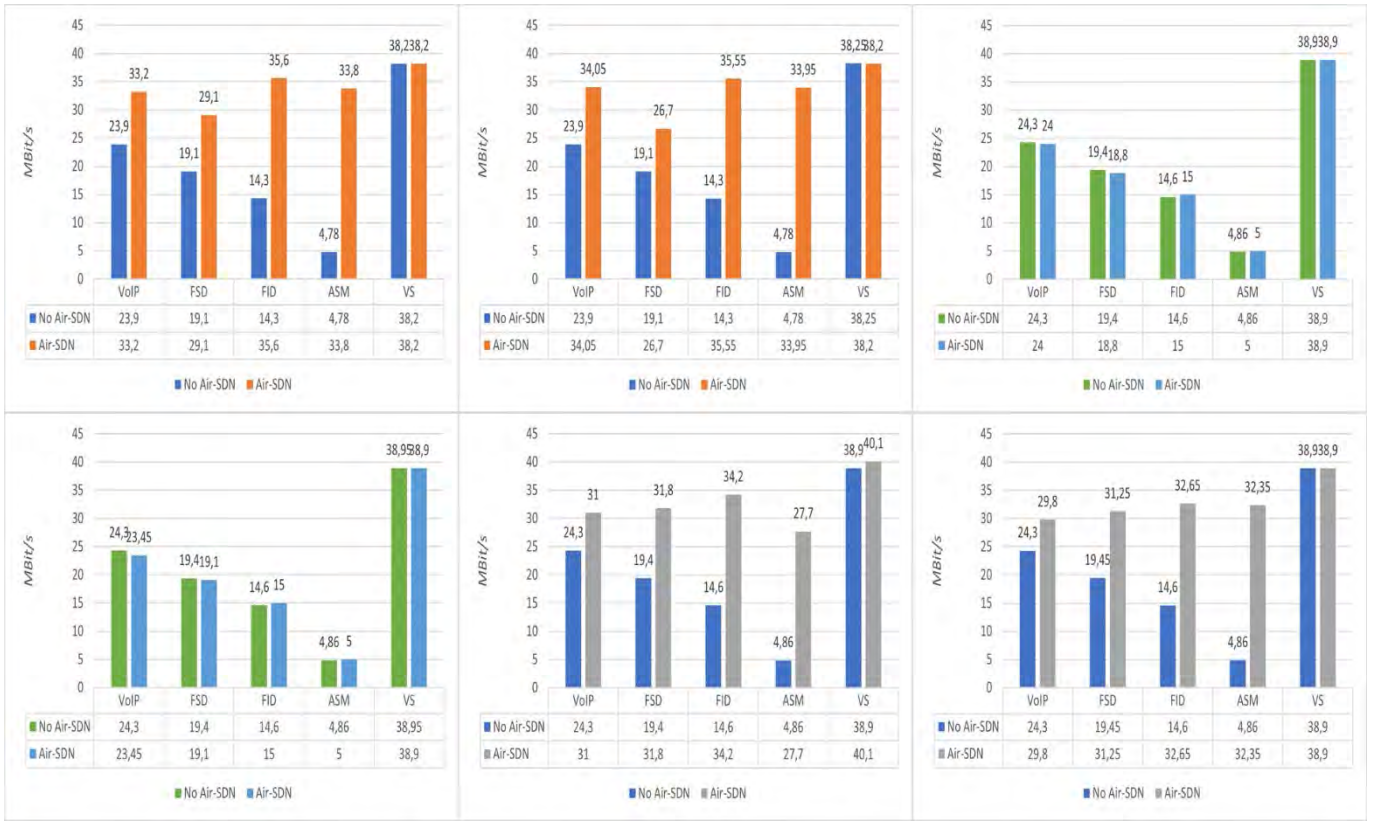
- 4- **Tekli/Çoklu Kullanıcı, Tam Yükte, TCP Veri Çıkış Hacmi ve Gecikme Ölçümü:** Bu parametrelere ait simülasyonlar sonucunda Air-SDN yönteminin, No Air-SDN yöntemine göre öncelikli veri tiplerine daha fazla veri çıkış hacmi ve daha az gecikme sunduğu görülmektedir. Bu sonuçlar Şekil-3 ve Şekil-4’te verilmiştir.
- 5- **Tekli/Çoklu Kullanıcı, Tam Yükte, UDP Veri Çıkış Hacmi, Seğirme ve Paket Kaybı Ölçümü:** Bu parametrelere ait simülasyonlar sonucunda ölçülen veri çıkış hacminin, No Air-SDN yöntemine göre Air-SDN yönteminde öncelikli veri tipleri için çok küçük farklarla aynı kaldığı, seğirmenin tekli kullanıcıda tüm veri tipleri için daha az olduğu ve çoklu kullanıcıda VoIP veri tipi hariç diğer öncelikli veri tipleri için daha az olduğu, paket kaybının ise tekli kullanıcıda VoIP ve FSD veri tipleri için, çoklu kullanıcıda ASM veri tipi haricindeki veri tipleri için arttığı, diğer veri tiplerinde ise paket kaybı olmadığı görülmektedir. Bu artışın

sebebi ‘Sonuçların Değerlendirilmesi ve Tartışma’ kısmında açıklanmıştır. Bu sonuçlar Şekil-3, Şekil-6 ve Şekil-8’de verilmiştir.

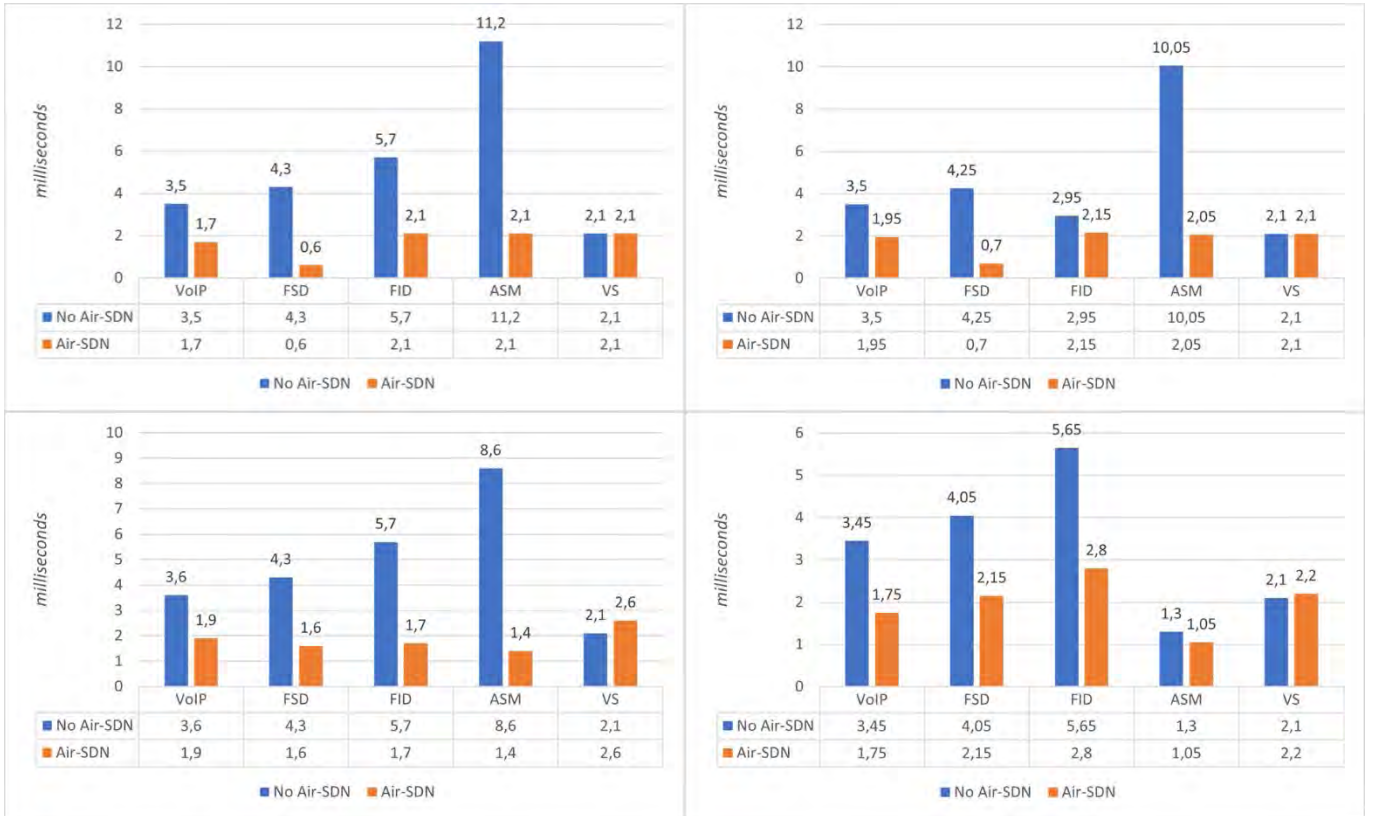
- 6- **Tekli/Çoklu Kullanıcı, Tam Yükte, Burst-UDP Veri Çıkış Hacmi, Seğirme ve Paket Kaybı Ölçümü:** Bu parametrelere ait simülasyonlar sonucunda Air-SDN yönteminin, No Air-SDN yöntemine göre öncelikli veri tiplerine daha fazla veri çıkış hacmi sunduğu, seğirmenin tekli kullanıcıda ASM veri tipi için, çoklu kullanıcıda VoIP ve FID veri tipleri için artış gösterdiği ve diğer öncelikli veri tipleri için azaldığı, paket kaybının ise tekli kullanıcıda öncelikli veri tipleri için, çoklu kullanıcıda ise FSD veri tipi haricinde öncelikli veri tipleri için azaldığı görülmektedir. Bu artışın sebebi ‘Sonuçların Değerlendirilmesi ve Tartışma’ kısmında açıklanmıştır. Bu sonuçlar Şekil-3, Şekil-6 ve Şekil-8’de verilmiştir.



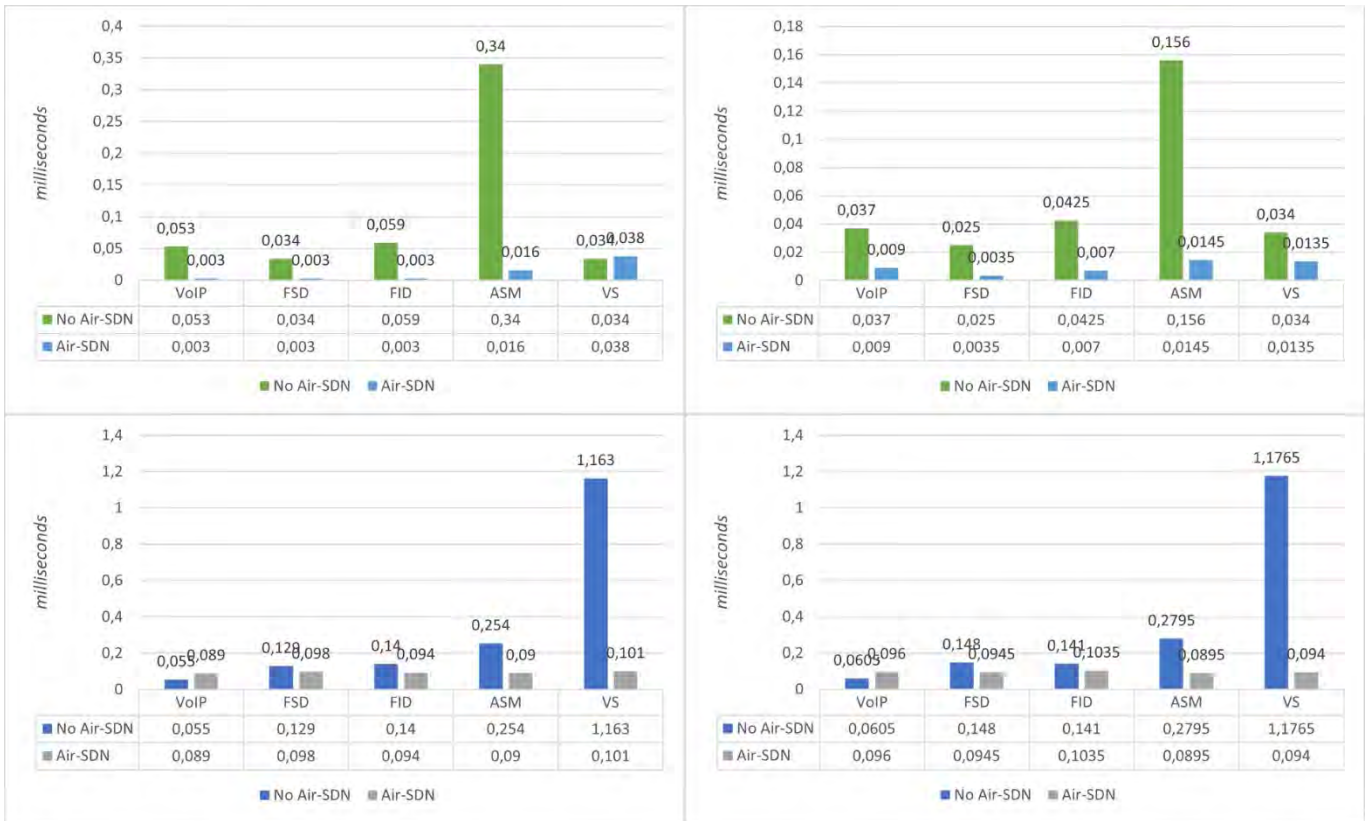
Şekil-2: Tekli/Çoklu Kullanıcı, Veri Tipi Yükünde, TCP, UDP ve Burst-UDP Veri Çıkış Hacmi



Şekil-3: Tekli/Çoklu Kullanıcı, Tam Yükte, TCP, UDP ve Burst-UDP Veri Çıkış Hacmi



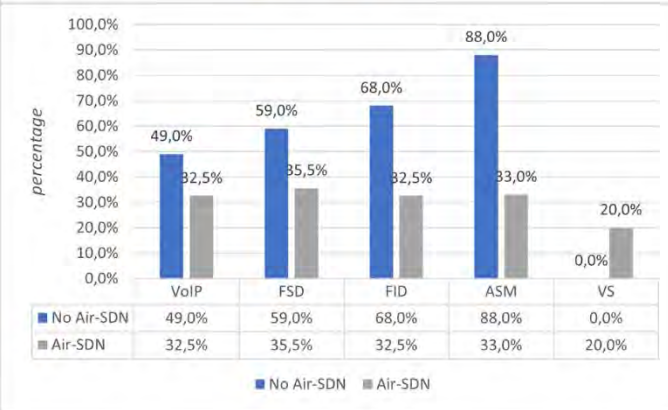
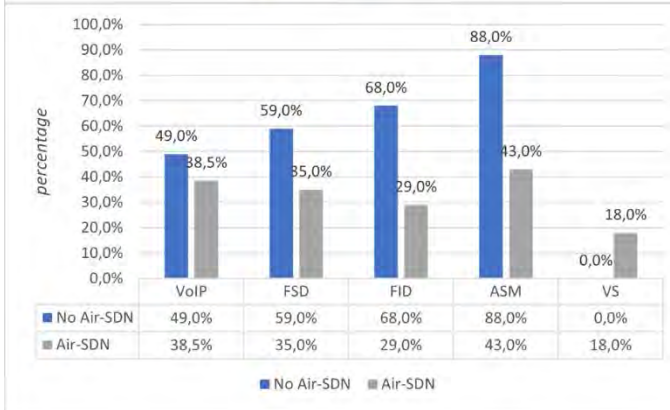
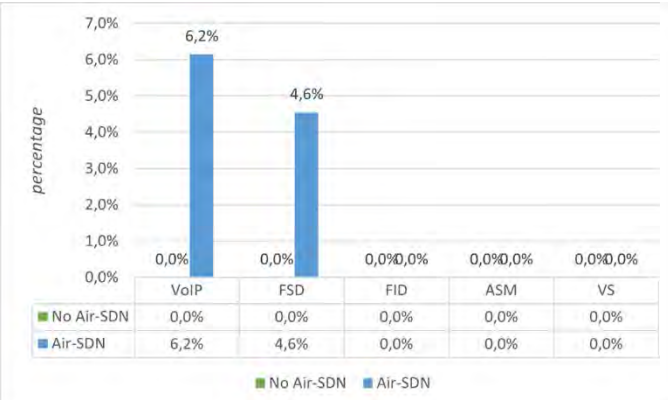
Şekil-4: Tekli/Çoklu Kullanıcı, Veri Tipi/Tam Yükte, TCP Gecikme



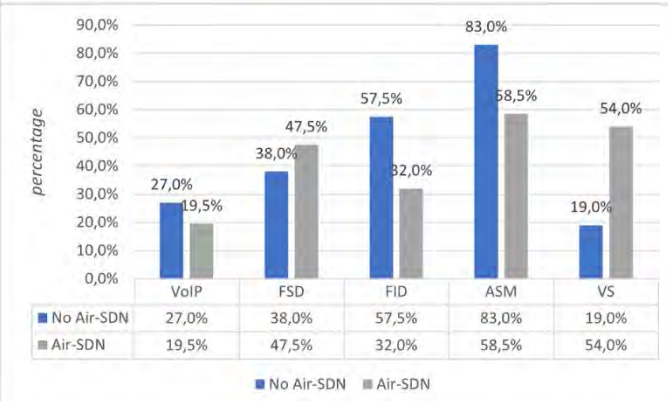
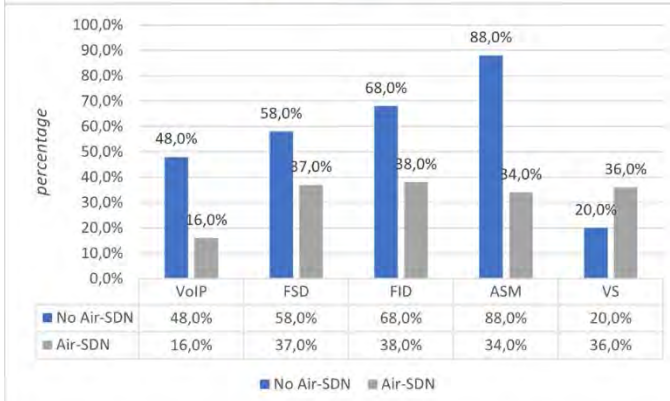
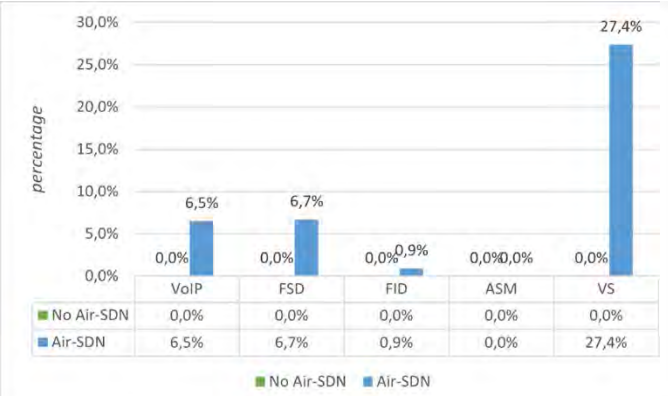
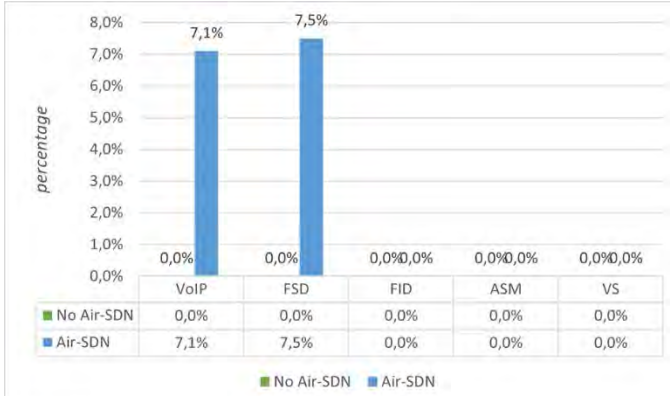
Şekil-5: Tekli/Çoklu Kullanıcı, Veri Tipi Yükünde, UDP ve Burst-UDP Seçirme



Şekil-6: Tekli/Çoklu Kullanıcı, Tam Yükte, UDP ve Burst-UDP Seçirme



Şekil-7: Tekli/Çoklu Kullanıcı, Tam Yükte, UDP ve Burst-UDP Paket Kaybı



Şekil-8: Tekli/Çoklu Kullanıcı, Veri Tipi Yükünde, UDP ve Burst-UDP Paket Kaybı

## 5. Sonuçların Değerlendirilmesi ve Tartışma

Bir önceki başlıkta verilen simülasyon sonuçları incelendiğinde, Air-SDN yönteminin TCP taşıma protokolünde tüm parametrelerde beklenen sonuçlara ulaştığı görülmektedir. Air-SDN yönteminin TCP taşıma protokolü ile, belirlenmiş kuyrukları uyarlamalı olarak kullandığı, No Air-SDN yöntemine göre öncelikli veri tiplerine daha fazla veri çıkış hacmi ve daha az gecikme sağladığı anlaşılmaktadır.

Air-SDN yöntemi UDP taşıma protokolü kullanıldığında No Air-SDN yöntemine göre veri çıkış hacmi ve seğirme ölçümlerinde gayet başarılı olsa da paket kaybı ölçümlerinde ulaşılan bazı sonuçlar dikkat çekicidir. Örneğin tekli kullanıcı, tam yükte VoIP ve FSD veri tipleri için paket kaybı %7'ler civarında olup, diğer veri tipleri için paket kaybı olmadığı saptanmıştır. Bu artışların nedeni şu şekilde açıklanabilir: Air-SDN algoritmasında tüm uç anahtarlayıcılardan akış istatistikleri belirli bir çerçeve içerisinde talep edilmekte, tüm anahtarlayıcıların cevap verebilmesi için bir saniyelik bekleme yapılmaktadır. Daha sonra değerlendirme fonksiyonu çalışarak akışlar için uyarlamalı kuyruklama ve yük dengeleme işlemi gerçekleştirilmekte ve kalan çerçeve süresi boyunca trafik akışına belirlenmiş konfigürasyonlar ile devam edilmektedir. Talep cevabı ile elde edilen değer, bir önceki çerçeve esnasında ölçülen bant genişliği ihtiyacı olup, bir önceki çerçeve içerisinde herhangi bir zamanda akışa dahil olan paketler için, daha önceden belirlenmiş kuyruğa ait bant genişliği, başlatılan akış bant genişliğinden daha az ise bu durum ilgili akış için paket kaybına yol açmaktadır. İkinci çerçeve başlangıcında akışın ihtiyaç duyduğu bant genişliğine göre kuyruk seçimi yapılmakta ve öncelikli veri tipleri için paket kaybı ortadan kalkmaktadır. Bu duruma örnek olarak VoIP trafiğine çoklu kullanıcı, veri tipi yükünde Burst-UDP taşıma protokolü parametresine ait akış-kuyruk atamaları, Çizelge-4'te zaman bazında ( $t$  zamanı 3 alınarak) verilmiştir.

**Çizelge-4: Çoklu Kullanıcı, Veri Tipi Yükünde Uyarlamalı Kuyruk Ataması (Burst-UDP)**

Host	Switch	0-3 s	3-6 s	6-9 s	9-12 s	12-15 s
h1	s11	q2	q0	q0	q0	q2
h2	s12	q2	q0	q0	q0	q2

Görüldüğü üzere başlangıçta q2'ye atanan ve 50 MBit/s bant genişliğinde verilen h1 sunucusuna ait akış 0-3 saniye arasında bir noktada başlamış, 3. saniyede hesaplanan bant genişliği ihtiyacı göz önünde bulundurularak ikinci çerçeveden itibaren bu akışa q0 atanmıştır. İlk çerçeve esnasındaki akış paketleri daha dar bir bant genişliği ile iletildiğinden paket kaybı ortaya çıkmıştır.

Burst-UDP taşıma protokolü ile yapılan simülasyonlarda No Air-SDN yöntemine göre Air-SDN yönteminin öncelikli veri tiplerine daha fazla veri çıkış hacmi sağladığı görülmüştür. Bu taşıma protokolü ile yapılan simülasyonlarda bazı veri tipleri için paket kaybı ölçümlerinin Air-SDN yönteminde No Air-SDN yöntemine nazaran yüksek çıkmasının sebebi önceki paragraflarda açıklanmıştır. Başka bir dikkat çeken ölçüm tekli/çoklu kullanıcı veri tipi yükünde Burst-UDP taşıma protokolü ile gözlenen seğirme değerleridir. Bu değerlerde

VoIP veri tipi için No Air-SDN yöntemi değerlerine nazaran bir artış gözlemlense de bu artışın yöntemden bağımsız olduğu, her iki yöntemde de tüm veri tipleri için ölçülen seğirmenin, veri tiplerin Çizelge-1'de verilen öncelik oranına göre arttığı ve kendi içerisinde tutarlı olduğu görülmüştür. Bir diğer dikkat edilmesi gereken ölçüm ise tekli/çoklu kullanıcı tam yükte ölçülen seğirme değerleridir. Bu parametrelerde yapılan tüm simülasyonlarda Air-SDN yöntemi ile bir ya da iki veri tipine ait seğirme değerlerinin sürekli No Air-SDN yöntemi değerlerine göre yüksek çıktığı, bazen ise doruk değerler elde edildiği görülmüştür. Parametrelerin sabit tutulduğu her simülasyonda bu durum için farklı veri tiplerinde yeni ve tutarsız değerler ölçülmüştür. Bu durumun trafik oluşturma programı iPerf'ün seğirme ölçme fonksiyonu kaynaklı rastgele bir davranışı olduğu başta düşünülse de Mininet emülatörünün kullanıldığı seğirme ölçümlerinde rastgele ve tutarsız sonuçlar verdiği Wang'ın çalışmasında ortaya konmuştur [26].

Her ne kadar simülasyonlarda karşılaşılan bu yöntemle ait paket kayıpları önceki paragraflarda belirtildiği şekilde açıklanabilir olsa da, paket kayıpları bir ATM sistemi için tolere edilebilir bir durum olmadığından, bu yöntemin UDP ve Burst-UDP taşıma protokolü ile kullanılması durumunda, belirli kuyruk parametrelerinde arzu edilen sonuçlara göre değişikliğe gidilmesi gerektiği saptanmıştır.

Sonuç olarak; önerilen Air-SDN yönteminin, SDN konsepti uygulanmış Hava Trafik Yönetimi (ATM) çekirdek ağlarında TCP taşıma protokolü ile başarılı bir şekilde kullanılabilir olduğu, UDP ve Burst-UDP taşıma protokolü parametrelerinde ise kuyruk parametrelerinin (*min-rate* ve *max-rate*), veri tipine ait parametrelerinin (*average/minimum bandwidth*), veri tipi bazında ağa verilen bant genişliği oranı ve *frame* süresi değerlerinin, üretim ortamındaki gerçek parametreler ile değiştirilerek rahatlıkla kullanılacağı görülmektedir. Test için kullanılan ağ topolojisinin de üretim ortamına uyumlu bir şekilde ölçeklenebilir olduğu, kapalı bir sistem olduğu için herhangi bir spesifik ağ güvenliği çözümü gerektirmediği de ulaşılan bulgular arasındadır.

Yapılan çalışma ve performans değerlendirmesi sonucunda, Hava Trafik Yönetimi (ATM) ağlarının gerçek zamanlı ve kritik veri yapısını taşıyabilecek kadar güvenilir ve sıkışıklık kontrolü sağlayabilen, sürekli veri yapısına da hızı ile cevap verebilecek ek protokollere ihtiyaç duyulduğu gözlenmiştir. TCP'nin paket iletim garantisi ve sıkışıklık kontrolü gibi özelliklerini taşıyan ancak UDP kadar hızlı olabilecek RTP (*Real Time Protocol*), MTCP (*Multipath TCP*) ve QUIC (*Quick UDP Internet Connections*) gibi bir protokolünün, ATM ağlarının veri yapısını, Air-SDN yöntemi ile sağlanan veri çıkış hacmi, seğirme ve paket kaybı ölçümlerindeki başarıyı ileri taşıyabilecek ve üstteki paragraflarda açıklanan tutarsız sonuçları da ortadan kaldırabileceği düşünülmektedir.

## Kaynakça

- [1] <https://www.eurocontrol.int/Economics/DailyTrafficVariation-States.html> (21.02.2023 tarihinde erişildi.)
- [2] <https://www.frequentis.com/en/pr/air-traffic-managements-first-software-defined-network-sdn-proves-operational-success-brazil> (17.03.2023 tarihinde erişildi.)
- [3] Thazin N., Nwe K. M., Ishibashi Y., *End-to-End Dynamic Bandwidth Resource Allocation Based on QoS Demand in SDN*, 2019, 25th Asia-Pacific Conference on Communications (APCC), Ho Chi Minh City, Vietnam, pp. 244-249.
- [4] Open Networking Foundation, *Software-defined networking: the new form for networks*, 2012, white paper.
- [5] Braden R., Clark D., Shenker S. *Integrated services in the internet architecture: an overview*, 1994, IETF RFC 1633.
- [6] Baker F., Black D., Blake S., Nichols K. *Definition of the differentiated services field (DS field) in the IPv3 and IPv6 headers*, 1998, RFC 2474.
- [7] Krishna H., van Adrichem L.M., Kuipers F.A. *Providing bandwidth guarantees with OpenFlow*, 2016, IEEE Symposium on Communications and Vehicular Technology in the Benelux, SCVT.
- [8] Xu C., Chen B., Qian H. *Quality of service guaranteed resource management dynamically in software defined network*, 2015, Journal of Communications, Vol.10 No.11.
- [9] Kadim U.N., Mohammed I.J. *SDN-RA: An optimized reschedule algorithm of SDN load balancer for data center networks based on QoS*, 2020, 2nd International Scientific Conference of Al-Ayen University (ISCAU-2020).
- [10] Gu Z., Wang Y., Lin X. *Achieving real-time quality of service in software defined networks*, 2018, 44th Annual Conference of the IEEE Industrial Electronics Society.
- [11] Al-Jawad A., Shah P., Gemikonaklı O., Trestian R. *Policy-based QoS management framework for software-defined networks*, 2018, International Symposium on Networks, Computers and Communications (ISNCC).
- [12] Guo X., Lin H., Li Z., Peng M., *Deep-Reinforcement-Learning-Based QoS-Aware Secure Routing for SDN-IoT*, 2020, IEEE Internet of Things Journal, Vol.7, No.7.
- [13] Vanitchasatit P, Sanguankotchakorn T. *A class-based adaptive QoS control scheme adopting optimization technique over WLAN SDN architecture*, 2022, International Journal of Computer Networks & Communications (IJCNC), vol. 14, no.3, pp.55-72
- [14] Al-Haddad R., Velazquez E.S., Fatima A., Winckles A. *A novel traffic shaping algorithm for SDN-slices networks using a new WFQ technique*, 2021, International Journal of Advanced Computer Science and Applications (IJACSA), Vol. 12, No.1.
- [15] Li F., Cao J., Wang X., Sun Y., Sahni Y. *Enabling software defined networking with QoS guarantee for cloud applications*, 2017, IEEE 10th International Conference on Cloud Computing.
- [16] Baek S.U., Park C.H., Kim E., Shin D. *Implementation and verification of QoS priority over software defined networking*, 2016, International Conference of Internet Computing and Internet of Things (ICOMP).
- [17] Akella A.V., Xiong K. *Quality of service (QoS) guaranteed network resource allocation via software defined networking (SDN)*, 2014, IEEE 12th International Conference on Dependable, Autonomic and Secure Computing.
- [18] Karakus M., Durrresi A. *Quality of Service (QoS) in Software Defined Networking (SDN): A Survey*, 2017, Journal of Network and Computer Applications, Vol.80, pp. 200-218.
- [19] Ongkasae K., Nakazato H., Koga T., Lu X. *QoS implementation in system wide information management (SWIM) network model*, 2015, IEEE Twelfth International Symposium on Autonomous Decentralized Systems, Taichung, Taiwan, pp. 181-184.
- [20] Izquierdo-Zaragoza J. -L., Lins W., Leydold P., Eier D. *Hierarchical software-defined networks for wide-area air traffic management networks*, 2019, Integrated Communications, Navigation and Surveillance Conference (ICNS), Herndon, VA, USA, pp. 1-11.
- [21] Open vSwitch *What is Open vSwitch?*, <https://openvswitch.org> (19.04.2023 tarihinde erişildi.)
- [22] Ryu *What's Ryu?*, <https://ryu-sdn.org> (20.04.2023 tarihinde erişildi.)
- [23] Python *Get Started*, <https://www.python.org/> (20.04.2023 tarihinde erişildi.)
- [24] Mininet *An Instant Virtual Network on your Laptop (or other PC)*, <https://mininet.org> (20.04.2023 tarihinde erişildi)
- [25] iPerf *The ultimate speed test tool for TCP, UDP and SCTP*, <https://iperf.fr> (20.04.2023 tarihinde erişildi.)
- [26] Wang S. *Comparison of SDN OpenFlow network simulator and emulators: EstiNet vs. Mininet*, 2014, IEEE Symposium on Computers and Communications (ISCC), Funchal, Portugal.

# Derin Öğrenme ile Türkçede Adıl Çözümleme

## Pronoun Resolution in Turkish with Deep Learning

Mehmet TAZE

İzmir Ekonomi Üniversitesi  
Lisansüstü Eğitim Enstitüsü  
İzmir - Türkiye  
mehmettaze89@gmail.com  
ORCID: 0000-0001-6835-302X

Senem KUMOVA METİN

İzmir Ekonomi Üniversitesi  
Mühendislik Fakültesi  
İzmir - Türkiye  
senem.kumova@ieu.edu.tr  
ORCID: 0000-0002-9606-3625

### Öz

Dilde, bir sözcüğün/sözcük öbeğinin sürekli tekrar eden kullanımını önlemek için, ilgili öncül sözcüğe/sözcük öbeğine atıfta bulunan adillerin kullanımına sık rastlanır. Bir adılın atıfta bulunduğu öncül ile eşleştirilmesi adıl çözümleme olarak adlandırılır. Bu çalışmada Türkçe metinlerde adillerin çözümlenmesinde derin öğrenme yöntemlerinin başarımı değerlendirilmiştir. Çalışma kapsamında 10 Türkçe çocuk hikayesi kullanılarak bir veri kümesi derlenmiş, deneylerde kullanılmak üzere 12 öznitelik belirlenmiştir. Çok katmanlı algılayıcı, evrimsel (konvüsyonel) ve tekrarlayan sinir ağları nöron ve katman sayılarının değiştiği bir dizi farklı konfigürasyonla uygulanarak F1 ölçüsü ile başarımlar ölçülmüştür. Sonuçlar, Türkçe adıl çözümlemesinde en yüksek başarımların, çok fazla nöron kullanan orta sayıda katmana sahip çok katmanlı algılayıcı sinir ağı tarafından elde edildiğini göstermiştir.

**Anahtar sözcükler:** Doğal dil işleme, Adıl çözümleme, Evrimsel sinir ağları, Çok katmanlı algılayıcı, Tekrarlayan sinir ağları

### Abstract

In language, to avoid repetitive use of a word/phrase, it is common to use pronouns that refer to the corresponding antecedent word/phrase. Matching a pronoun with the antecedent to which it refers is called pronoun resolution. This study evaluates the performance of deep learning methods in pronoun resolution in Turkish texts. Within the scope of the study, a dataset was compiled using 10 Turkish children's stories and 12 features were identified to be used in the

experiments. Multilayer perceptron, convolutional and recurrent neural networks were tested in several different

configurations with varying numbers of neurons and layers and the performance was measured with F1 score. The results show that the highest performance in Turkish pronoun resolution is achieved by a multilayer perceptron neural network with a medium number of layers using too many neurons.

**Keywords:** Natural language processing, Pronoun resolution, Convolutional neural networks, Multilayer Perceptron, Recurrent neural networks

### 1. Giriş

Günlük yaşamda dinleyici ve okuyucuların konuşma ve yazıları kolay takip edebilmesi ve akıcılığın sağlanması için aynı sözcük veya sözcük öbeklerinin tekrar tekrar kullanılmasından kaçınılır. Bu sebeple sıklıkla daha önce/sonra bahsedilen varlıklara atıfta bulunan veya onların yerini alan ifadeler kullanılır. Geriye/ileriye atıfta bulunan bu tür ifadeler gönderim ve işaret ettikleri varlıklar ise öncül olarak adlandırılır. Öncül-gönderim çiftlerini tanımlama görevi ise doğal dil işleme alanında önemli bir yere sahip olan gönderim çözümlemesi görevidir.

Alanyazında farklı dillerde gönderim kavramını tartışan ve farklı bakış açılarına odaklanan çeşitli çalışmalar bulunmaktadır. 1995 yılında Grosz gönderimin biçimsel tanımını yapmıştır [1]. Daha önce bir adıl ya da ad öbeği ile bahsedilen bir olay ya da nesneye atıfta bulunmanın gönderim/anafor olduğunu belirtmiştir. [2]'de ise gönderimin daha önce bahsedilmiş bir şeyi göstererek bağdaşıklık sağlama görevi olduğunu söylenmiştir. Valin ve LaPolla [3] metnin bağdaşıklığının mekân, zaman ve eylemin ayrınlığı/benzerliği ile şekillenebileceğini belirtmiştir. Bir



metindeki ya da konuşmadaki bağdaşıklık sayesinde insanlar bir metni ya da konuşmayı bütünleşik bir şekilde algırlarlar. Söylemdeki bağdaşıklık sayesinde ise gönderim ögesi ile atıfta bulunulan öğeler belirlenebilir. Bu sayede insanlar daha önce karşılaştıkları bir şey hakkında yeni bilgiler edinebilirler [4]. Örneğin, aşağıdaki cümlelerde gönderim ilişkisi "onlar" ve "çiçekler" kelimeleri arasındadır.

*Çiçekler solmuş. Onları sulamak gerek.*

Bu örnekte, "onlar" adılı "çiçekler" adına atıfta bulunmaktadır. Basitçe, ilk cümle "çiçekler" ögesinin mevcut durumunu vermekte ve ikinci tümce aynı öğeyle ilgili yeni bir bilgi (gerçekleştirilecek bir eylem) eklemektedir.

Gönderim-öncül ikililerinin tespiti bir metnin veya tümcenin anlaşılmasını geliştirebilir, ancak gönderimlerin belirlenmesi bile çeşitli göstergeler olmadığı zaman zordur [5]. Bazı durumlarda, sıklıkla gönderim olarak gözlenen sözcüklerin (örneğin adılar), farklı amaçlarla kullandığı gözlenebilir.

Gönderim çözümleme, makine çevirisi, otomatik özetleme, metin özetleme, bilgi çıkarımı ve soru yanıtlama sistemleri gibi birçok doğal dil anlama ve işleme uygulamasında pratik olarak kullanılmaktadır [6]. Gönderimsel ilişkilerin çözümlenmesini gerektiren bu geniş görev yelpazesi nedeniyle, bu problemi çözmek için yıllar içinde çeşitli çalışmalar sunulmuştur. Devam eden bu çalışmalar, sözdizimsel, anlamsal ve gerçek dünya bilgilerini yoğun olarak kullanan bilgi yoğun tekniklerden, metinden elde edilen bilgilere dayanan yaklaşımlara kadar çok çeşitli çözümleme yöntemlerini içermektedir [7]. Gönderimsel ilişkilerin sınıflandırılmasında ise ilgili ilişkinin yönü, gönderim ögesinin ses içeriğine dönüşebilme yetkinliği gibi çeşitli unsurlar söz konusu olur. Gönderimsel ilişkilerde yön kavramı gönderim ögesi ve öncülün metin içindeki konumları ile ilişkilidir. Gönderim yönlerinden ilki, önceki örnek tümcelerde olduğu gibi gönderim ögesinin daha önce bahsedilen bir öğeye atıfta bulunması anlamına gelen art gönderimdir. Diğeri ise ögenin daha sonra bahsedilecek bir öğeye atıfta bulunduğu ön gönderimdir. Örneğin aşağıda verilen ilk tümce de yer alan "onun" sözcüğü ikinci tümce deki "Ayşe" adına atıfta bulunan bir ön gönderimdir.

*Sabahtan beri onun hakkında konuşuyoruz. Ama Ayşe henüz gelmedi.*

Ses içeriği bakımından gönderimler değerlendirildiğinde gönderim ögesinin metin içinde açıkça yer alması veya gizli olarak sunulması şeklinde iki durum mevcuttur. Örneğin aşağıdaki tümcelerde üç gönderim yer almaktadır. İkinci tümcedeki "Onu" sözcüğü ses içeriğine sahiptir ancak Ø1 ve Ø2 sembolü ile gösterilen gönderim öğeleri ses içeriğine sahip değildir. Başka bir deyişle, "Öğretmen" ve "Mehmet" sözcüklerine gönderme yapan iki öğe ikinci tümce de açıkça verilmemiştir. Bu nedenle Ø1 ve Ø2 "gizli gönderim" olarak tanımlanır.

*Tümce 1. Öğretmen1, Mehmet2'e bir kalem3 verdi.*

*Tümce 2. Ø1 Ø2 Onu3 ders sonunda geri vermesini istedi.*

Gizli gönderimlerin kullanımı dile göre değişmektedir. Örneğin, Kornfilt [8] Türkçede gönderim öğelerinin (özellikle

adıların) mümkün olduğunda silinmesinin tercih edildiğinden bahsetmiştir. Öte yandan, İngilizcede gönderim öğelerinin sese dönüşmesi yani öğelerin metinde yer alması çok daha yaygındır [9]. Türkçede gizli gönderim öğelerinin sık kullanılması nedeniyle, çözümleme görevinin de İngilizceye kıyasla çok daha zor olduğu belirtilmektedir [10].

Bu çalışmada gönderim ögesi ifadesi aynı veya önceki tümcelerde yer alan bir öğeye atıfta bulunan Türkçe adılar ile sınırlandırılmıştır. Çalışma kapsamına yer alan adıl örnekleri Çizelge-1'de verilmiştir. Gönderim çözümlemede (bu çalışma özelinde adıl çözümlemede) 5 açık adıl türü göz önünde tutulmuştur. Deneylerimizde, söz konusu adıların olası tüm dilbilgisel durumları (örneğin, belirtme, yönelme, bulunma, ayrılma, yalın) dikkate alınmıştır.

**Çizelge-1: Çalışma kapsamında değerlendirilen adılar**

Adıl türü	Örnekler
Kişi adıları	ben, sen, o, biz, siz, onlar
Dönüşlülük adıları	kendim, kendin, kendi, kendimiz, kendiniz, kendileri
İşaret adıları	bu, şu, o, bunlar, şunlar, onlar, bura, şura, ora, buralar, şuralar, oralar, öbürü, ötekisi, berikisi, diğeri
Belgisiz (belirsiz) adılar	kimi, kimileri, kimse, biri, birisi, birileri, başkası, başkaları, bazısı, birçoğu, birkaçı, birazı, herkes, hepsi, hepimiz, hiçbirisi, herhangi biri, her biri, şey, çoğu
İşteş adıları	Birbiri, birbirimiz, birbirinizi, birbirleri

Deneylerimizde, adıl-öncül çözümlemesini ikili sınıflandırma görevi olarak kabul ederek, 593 olumlu ve en çok 17.790 olumsuz örnekten oluşan bir adıl-öncül veri kümesi oluşturulmuştur. Sınıflandırmada 12 öznitelik (örneğin öncülde büyük harf kullanımı, sözcük sayısı, karakter sayısı) kullanılmıştır.

Çalışmada temel amacımız yapay sinir ağlarında nöron ve katman sayısındaki değişimin ilgili sınıflandırma problemindeki etkisinin incelenmesi ve en başarılı kurulumla sahip sinir ağının tespit edilmesidir. Bu doğrultuda öznitelik vektörleri bir dizi farklı konfigürasyon ile kurulan üç farklı derin sinir ağına girdi olarak sunulmuştur. Başarım değerlendirmesi ağırlıklı F1 ölçüsü ile gerçekleştirilmiştir.

Çalışmanın geri kalanı şu şekilde düzenlenmiştir. Bölüm 2'de Türkçedeki adılar hakkında temel bilgiler verilmektedir. Adıl çözümlemesi ile ilgili çalışmalar 3. bölümde verilmiştir. Önerilen yöntem ve deneysel düzenek sırasıyla bölüm 4 ve 5'te verilmiştir. Sonuçlar bölüm 6'da sunulmuş ve makale bölüm 7'de sonlandırılmıştır.

## 2. Türkçede Adılar

Türkçe, sözcüklerin geleneksel olarak adlar, önadlar, belirteçler, adılar, ilgeç, bağlaçlar, ünlemler ve eylemler olarak sınıflandırıldığı sonda eklemeli bir dildir. Adılar sınıfı, genellikle daha önce bahsedilmiş veya daha sonra bahsedilecek bir ad veya ad öbeğine atıfta bulunmak için kullanılan küçük bir kelime kümesini belirtir. Türkçede adılar, atıf yapılan adların türüne ve/veya adların gönderme şekline

göre sınıflandırılabilir. Bu sınıflar ve kısa tanımları aşağıda verilmiştir.

1. *Kişi adilları*: Bu tür adiller tümce içinde özne olarak bir kişi ya da gruba atıfta bulunmak için kullanılır ve genellikle atlanırlar.
2. *Dönüşlülük adilları*: Türkçede "kendi" dönüşlülük adılı eylemi kimin yaptığını vurgulamak için kullanılır ve sıklıkla kişi adilları yerine kullanılır.
3. *İşaret adilları*: Kişi adları dışındaki tüm nesne/kavram adlarının yerini alan kapalı sözcük sınıfıdır.
4. *Belgisiz adilları*: Tanımlanmamış kişileri, yerleri, miktarları veya şeyleri gösterirler.
5. *Soru adilları*: İsimleri soru biçiminde temsil eden ve soru olarak ifade eden sözcükler soru adillarıdır. Soru adilları örnek olarak kim (kim), kaç tane (kaç tane) verilebilir.
6. *İyelik adilları*: İyelik adilları -im (örn. kalemim), -in (örn. kalemin) gibi bazı son eklerin kullanımıyla sahiplik gösterirler.
7. *İlgi -ki adilları*: İngilizcedeki "who (kim)", "whom (kimi)" gibi ilgi adillerinden farklı olarak Türkçe ilgi adilları -ki eki kullanılarak sahiplik ifade eder. Örnek olarak benimki (benim), onlarınki (onların) sözcükleri verilebilir.

Bu kategorilere ek olarak, işteş adiller olarak bilinen bir adıl türü daha vardır. İşteş adillara örnekler Çizelge-1'de verilmiştir.

### 3. Önceki Çalışmalar

Adıl ya da daha genel olarak gönderim çözümlemesindeki temel amaç, belirli bir metin ya da konuşmadaki gönderim ve öncülleri belirlemektir. Önerilen yaklaşımlar genellikle söylemdeki her bir gönderim/adıl için olası öncülleri bulur ve en uygun olanı en yüksek başarımla seçer. Çalışmalar, hedefe ulaşmak için izledikleri yol, çözümleme sürecinde kullanılan yöntem ve öznitelikler bakımından farklılık göstermektedir. Ayrıca deneylerin yapıldığı dil de çözümleme sisteminin yapısını değiştirebilecek önemli bir etkidir. Çözümleme çalışmaları bilgi tabanlı, öğrenme tabanlı ve melez yaklaşımlar olmak üzere üç ana grupta toplanabilir.

Bilgi tabanlı yaklaşımlar biçimsel, söz dizimsel, sözcüksel ve söylem kaynağı bilgilerine dayanır. Bilgi tabanlı yaklaşımların ilk örneklerinden ikisi [11]'de önerilmiştir. Bunlardan ilki, metindeki bir tümcenin ayrıştırma ağaçları üzerinde çalışan basit ve etkili bir yöntemdir. Algoritma, ağaçları soldan sağa doğru dolaşır ve adıl le birlikte doğru bir sayı ve cinsiyet içeren bir ad öbeği arar. Hobbs [11] deneylerinde Arthur Hailey'in bir romanını ve Newsweek dergisinin bir bölümünü kullanmış ve %88,3 başarı oranı elde etmiştir. Hobbs tarafından önerilen ikinci yaklaşım ise ilk yaklaşıma anlam analizinin eklendiği geliştirilmiş bir versiyonudur [11]. Bu yaklaşım ile %91,7 başarı oranı elde edildiği bildirilmektedir.

Lappin ve Leass [12], söz dizimsel yapı ve çarpıcı durum modelinden elde edilen dikkate değer ölçütleri kullanan anafora çözümleme sürecini (RAP) tanıtmıştır. RAP'da, çözümleme işlemi sırasında, bazı aday ad öbeklerini elemek için biçimsel ve söz dizimsel filtreler kullanılır. Daha sonra, kalan adaylara önceden tanımlanmış kriterler aracılığıyla belirli değerler atanır. Bu algoritmada kullanılan kriterler

yakınlık, özne önceliği, varlık önceliği ve ad öbeği önceliğinden oluşmaktadır. Tümceler arası için %72, tümce içi için ise %89 başarı oranı elde edildiği bildirilmiştir [12]. Kenndy ve Boguraev'in [13] çalışmasında RAP metnin derinlemesine bir şekilde sözdizimsel ayrıştırılmasına ihtiyaç duymayacak şekilde değiştirilmiştir. Değiştirilmiş algoritmada, aday öncüller dilbilgisel roller yerine sadece sözdizimsel bilgilere göre sıralanmaktadır. Kenndy ve Boguraev üçüncü şahıs adilları üzerinde %75 başarı oranı elde etmiştir [13].

Öğrenme tabanlı yaklaşımlar, çözümleme için makine öğrenimi tekniklerini kullanır. [14] çalışmasında karar ağacına dayalı denetimli bir öğrenme sistemi önerilmiştir. İlgili sistem sözlük, sözdizimi, anlambilim ve konum gibi 66 özellik ile eğitilmiştir. Sistem tarafından %90 başarı oranı elde edildiği rapor edilmiştir [14]. McCarthy ve Lehnert [15] tarafından sunulan RESOLVE sisteminde ise sekiz anlamsal özellik bir karar ağacına verilmiş ve deneyler MUC-5 metinlerinden elde edilen veri seti üzerinde gerçekleştirilmiştir. Başarı oranları budanmış ağaçlar için %87,6, budanmamış ağaçlar için ise %92,4 olarak sunulmuştur. Soon, Ng ve Lim'in [16] çalışmasında karar ağaçları 12 özellik (sözcüksel, söz dizimsel, anlamsal ve konum özellikleri) ile kullanılmış ve deneyler MUC-6 ve MUC-7 derlemleri üzerinde gerçekleştirilmiştir. Her bir özelliğin çözümleme görevine katkısı da [16]'da ölçülmüş ve sadece 3 özellik kullanılarak %68'lik bir başarı oranı elde edildiği raporlanmıştır.

Öğrenme ve bilgi tabanlı yaklaşımların bir arada kullanılmasıyla ise melez çözümleme yaklaşımları ortaya çıkmıştır. [17] ve [18] melez yaklaşımlara örnek olarak verilebilir. 2002 yılında Mitkov tam otomatik çözümleme sistemi MARS'ı (Mitkov's Anaphora Resolution System) tanıtmıştır [17]. MARS'a girdi olarak ham yazılı metin verilir. Ardından, sistem sözdizimsel ayrıştırma ile verilen metindeki ad öbeklerini belirler. Daha sonra makine öğrenmesi yöntemleri ile gönderimsel (anaforik) öğeler belirlenir. Tüm anaforik öğeler için sistem iki tümce ye kadar aday öncülleri toplar ve kişi, sayı ve cinsiyet entegrasyon filtreleri uygular. Filtrelenen aday öncüller çeşitli faktörler kullanılarak sıralanır ve son olarak gerçek öncül belirlenir. Mitkov, Evans ve Orasan, tam otomatik MARS ile %61,6'lık bir başarı oranı elde edildiğini bildirmiştir [17]. Preiss, algoritmayı makine öğrenmesi yöntemleriyle değiştirmiş ve bu şekilde bellek tabanlı yaklaşımın etkinliğini ölçmüştür. Deneylerde dört adet son teknoloji olasılıksal ayrıştırıcı kullanılmıştır. Düzeltmelerden sonra sistemin asıl kural tabanlı yaklaşımla yaklaşık olarak aynı başarımla gösterdiği bildirilmiştir [18].

Bildiğimiz kadarıyla, Türkçede gönderim/anafora çözümlemesi üzerine yapılan en eski çalışma [19]'da sunulmuştur ve burada Merkezleme Kuramına dayalı bir sistem önerilmiştir. İngilizceden farklı olarak, Türkçede sıralama dilbilgisi kurallarına dayalı sıralama yerine tematik rollere dayalı olmalıdır. [19]'un amacı, Türkçede konuya özel vurgu yaparak ifadeler arasındaki anaforik ilişkileri içeren söylem tutarlılığının bir yönünü tespit etmektir. Bu çalışmada, söylem anafora ile ilişkili birbirini tamamlayan üç soru bulunmaktadır [19]. İlk soru, sonraki gönderim ile ilişkili olabilecek mevcut ifadelerin incelenmesi üzerinedir. İkincisi, Türkçede, bir dizi potansiyel öncül arasında, gönderim için en belirgin olanı

gösteren faktörlerin belirlenmesidir. Sonucusu ise belirli bir göndergesel ifadenin işlevlerini tespit etmek üzerinedir.

[20]'de, yerel ve yerel olmayan gönderimleri modellemek için bağlama ilişkileri ve merkezleme teorisi için bir dizi kuralı birleştiren bir sistem önerilmiştir. Sistemde, bağlama kuralına uygun olarak, dönüşlü adılların öncülü yerel etki alanında olmalıdır. Önerilen sistem, anaför, adıllar ve gönderme yapan ifadelerin enterpolasyonu ile ilgilenen Bağlama Teorisine [21] ve Merkezleme Kuramına [1][20] dayanmaktadır.

[6] çalışmasında öncülleri tahmin etmek için makine öğrenmesi tabanlı bir yaklaşım sunulmuştur. Önerilen yöntem, topluluk öğrenme yöntemiyle birleştirilmiş bir karar ağacı sınıflandırma tekniğidir ve amaç şahıs adlarının öncüllerini tespit etmektir. [10]'da Hobbs'un Naive algoritması [11] Türkçe için kullanılmak üzere değiştirilmiştir. Sistem, verilen adılların olası öncüllerini toplamak için söz dizimsel bilgiyi kullanarak nominal seviye bilgisini işler.

Daha önceki çalışmalardan birinde [22], bir tümce deki açık veya boş adıl in nasıl ve nerede kullanılabileceğini açıklamak üzerine tartışmıştır. [23] çalışmasında, bağlamsal faktörlerin Türkçe anafora çözümlemesinde yüksek öneme sahip olduğundan bahsedilmiştir. Ayrıca, Türkçede adıl sel anaför çözümlemesinde kullanılan durum kuramsal bir hesaplama ortamı olan BABY-SIT'i tanımlamışlardır.

#### 4. Adıl Çözümlemesinde Derin Sinir Ağlarının Kullanımı

Bu bölümde Türkçe metinlerde adıl çözümlemesinde derin sinir ağlarının kullanımı için yürüttüğümüz çalışmamıza ilişkin ayrıntılar sunulacaktır. Sırasıyla yapay sinir ağlarına girdi olarak sunulan öznitelik vektörlerinin içeriği, deneylerde yer alan üç sinir ağı kısaca tanıtılacaktır.

##### 4.1. Öznitelikler

Yapay sinir ağları, verilen örnekleri sınıflandırmak için bir grup öznitelik kullanılması gerektirir. Bu çalışmada adıl-öncül çiftlerinin çözümlenmesinde 12 öznitelikten oluşan vektörler ağlara girdi olarak sunulacaktır. Bu öznitelikler aşağıda kısaca açıklanmıştır.

1. *Öncülde büyük harf kullanımı (CLU)*: CLU, öncülün büyük harfle başlaması durumunda 1, aksi takdirde 0 değerini alan ikili bir öznitelik olarak tanımlanmıştır. CLU 1 ise, adıl-öncül çiftinin doğru bir çift olma eğiliminde olması beklenir.
2. *Sözcük sayısı (NT)*: NT özelliği, öncül adayı ile adıl arasındaki sözcük sayısını (iki boşluk arasındaki herhangi bir dize) temsil eder. NT değerlerinin aralığı pencere boyutu ile sınırlıdır.
3. *Karakter sayısı (NC)*: NC, öncül adayı ile adıl arasındaki karakter sayısıdır (boşluklar hariç). Örneğin, aşağıdaki tümceler göz önüne alındığında  
*Ali eve geldi. Onu özlediğimi biliyordu.*  
(Onu, Ali) çifti için  $NC(\text{Onu}, \text{Ali})=9$  olarak hesaplanır.

4. *Benzersiz karakter sayısı (NUC)*: NUC özelliği, öncül adayı ile adıl arasındaki benzersiz (farklı) karakterlerin (boşluklar hariç) sayısını temsil eder. Örneğin, önceki örnek tümcelerdeki (Onu, Ali) çifti için  $NUC(\text{Onu}, \text{Ali})=6$ 'dır.

5. *Adıl ve öncül arasında büyük harf kullanımı (CLUB)*: CLUB özelliğinde, adıl ve öncül arasındaki büyük harf sayısı sayılır. CLUB değerinin düşük olması durumunda adıl ve öncül adayı arasında daha az sayıda özel ad kullanılması beklenir ki bu da ilgili öncülün gerçek öncül olduğunu düşündürür.

6. *Ad sayısı (NN)*: NN, öncül adayı ile adıl arasındaki ad etiketli sözcüklerin sayısıdır. Doğru adıl-öncül çiftleri arasında daha düşük NN değerleri gözlemlenmesi beklenir. Örneğin, aşağıdaki tümcelerde "Ali" öncül adayı ve "Onu" adıl i için  $NN(\text{Ali}, \text{Onu}) = 2$  ("Ali" ve "Onu" arasındaki adlar "eve" ve "çantasıyla" sözcükleridir).

*Ali eve çantasıyla geldi. Onu unuttuğunu bilmiyordu.*

7. *Adıl sayısı (NP)*: NP özniteliği adıl ile öncül adayı arasındaki adıl sayısını ifade eder. İlgili öncül ile adıl arasında çok sayıda adıl varsa, adıl-öncül çiftinin olumsuz/yanlış örnek olmaya daha yakın olduğu varsayılır.

8. *Noktalama sayısı (NPU)*: Bu çalışmada, verilen öncül adayı ve adıl arasındaki noktalama sayısı arttıkça, çiftin bileşenlerinin bağlanma şansını kaybetmesi beklenmektedir. Sonuç olarak NPU özelliğinde adıl ve öncül arasındaki noktalama sayısı sözcük türü etiketleyici çıktıları baz alınarak ölçülmektedir.

9. *Öncülün Uzunluğu (LA)*: Zipf, dildeki en az çaba ilkesi nedeniyle insanların daha kısa sözcükleri daha sık kullanma eğiliminde olduğunu belirtmektedir [24]. Zipf'in iddiasından yola çıkarak, uzun sözcüklerin kısa biçimleriyle, başka bir deyişle adıllarla değiştirilmeye daha yatkın olduğu söylenebilir. Bu bakış açısından yola çıkarak, adıl çözümleme özelliği olarak öncül adaylarının uzunluğu kullanılmıştır. LA, öncül adayının karakter cinsinden uzunluğunu temsil etmektedir.

10. *Çoğul adıl (PP)*: Bu çalışmada, çok sözcüklü öncüller için, sınıflandırıcı çok sözcüklü bir öncülün kurucu sözcüklerinden birini ilgili adıl le eşleştirirse, eşleşme doğru pozitif / isabet olarak kabul edilir. Örneğin, aşağıdaki tümcelerde "onlar" adıl i "Ali ve Ayşe" çok sözcüklü öncülüne gönderme yapmaktadır.

*Ali ve Ayşe eve geldi. Onlar çok yorgundu.*

Sonuç olarak, (Onlar, Ali) (Onlar ve) ve (Onlar, Ayşe) çiftlerinden herhangi biri adıl-önad çifti olarak atanırsa, bu bir başarı (isabet) olarak kabul edilir. PP özelliği bu varsayıma dayanmaktadır. Eğer çift çoğul adıl içeriyorsa PP değeri 1 olarak atanır, çünkü çoğul adıllar için pozitif çiftlerin sayısı tekil adıllara kıyasla daha fazla olacaktır. Başka bir deyişle, çoğul adıllarla eşleştirilen öncül adaylarının adıl tarafından gerçekten ifade edilme şansı daha yüksektir.

11. *Sayı uyumu (NUMA)*: Sayı uyumu özelliğinde adil ve öncül adayının çoğulluk durumu açısından uyuşup uyuşmadığına bakılır. NUMA değeri NA1, NA2, ..., NA6 olmak üzere altı sayı uyumu göstergesinin bir kombinasyonudur. Bu parametreler şunlardır

NA1: Adil çoğulsa ve öncül adayı "-lar", "-ler" çoğul eklerini içeriyorsa veya öncül bir sayı ise NA1=1; aksi takdirde NA1=0 olur.

NA2: Adil çoğulsa ve öncül adayının ardından "ve", "ile" ve "," gibi bir bağlaç belirteci geliyorsa NA2=0,75; aksi takdirde NA2=0 olur.

NA3: Adil çoğulsa ancak çoğul bir öncülün söz dizimsel veya morfolojik kanıtı yoksa NA3=0,5; aksi takdirde NA3=0 olur.

NA4: Adil tekil ise ve çoğul bir öncül olduğuna dair kanıt yoksa NA4=0,75; aksi takdirde NA4=0 olur.

NA5: Adil tekil ise ve öncül adayı "-lar" veya "-ler" çoğul ekini içeriyorsa, NA5=0,25; aksi takdirde NA5=0 olur.

NA6: Adil tekilse ve öncül adayı "ve", "ile" ve "," gibi bir bağlaç belirteci izliyorsa veya öncül adayı bir sayı ise NA6=0,25; aksi takdirde NA6=0 olur.

NUMA, altı NA değerinin toplanması ve NUMA aralığını [0 1]'e yerleştirebilmek için toplama sonucunun 3,5'e bölünmesiyle elde edilir.

12. *Çoğul öncül (PA)*: PA, öncül adayının çoğul olup olmadığını veya çoğul bir ifadenin parçası olup olmadığını temsil eden bir özelliktir. Bu özellik iki durum göz önünde bulundurularak ölçülür:

Durum 1: Eğer öncül adayı "-lar" veya "-ler" çokluk eklerinden biriyle bitiyorsa PA1 değeri 1'e, bitmiyorsa 0'a atanır.

Durum 2: Eğer öncül adayından sonra gelen belirteç "ve" veya "ile" ise PA2= 1, aksi takdirde PA2=0 olarak atanır.

Bu durumlar göz önüne alındığında, PA1PA2 değerlerinin dört farklı olası kombinasyonu ("00", "01", "10" ve "11") gözlemlenebilir. Eğer kombinasyon "10", "01" veya "11" ise, aday öncül çoğul olarak kabul edilir ve karşılık gelen PA değerleri 1 olur; aksi takdirde PA=0 olarak atanır.

## 4.2. Çok Katmanlı Algılayıcı Ağlar

Derin ileri beslemeli ağlar olarak da bilinen çok katmanlı algılayıcı sinir ağları, giriş ve çıkış katmanı arasında bir veya daha fazla katmana sahip sinir ağlarıdır. Çok katmanlı algılayıcıların en basit şekli tek bir gizli katmana sahiptir [25]. Standart çok katmanlı algılayıcıda, sonlu sayıda nörona sahip birden fazla ardışık gizli katman vardır ve katmandaki her nöron, giriş katmanındakiler hariç doğrusal olmayan bir aktivasyon fonksiyonuna sahiptir. Çok katmanlı algılayıcı ağlar eğitimde geri yayılım algoritmasını kullanır.

Çok katmanlı algılayıcı ağlarda, bir katmandaki her bir nöron, bir sonraki ve bir önceki katmandaki her bir nörona bağlanır. Bu bağlantılar sinapsis olarak adlandırılır. Bir katmandan diğerine bilgi akışı bu sinapsisler tarafından sağlanır ve bu akış

ileri besleme olarak adlandırılır. Son olarak, ağıncı çıktısı çıkış katmanı tarafından aktarılır [26]. Eğitimde, hata geri yayılım algoritması çok katmanlı algılayıcı ağlarda yaygın olarak kullanılır. Basitçe, geri yayılım, ağıncı hatayı ortadan kaldırmak için ağıncı ağıncı ve eşikleri geriye doğru bir şekilde yeniden hesaplar. Geri yayılım sırasında hata örüntüleri sırayla yayılır. Sınıflandırma görevinde, eğitilen sinir ağıncı doğru çalışır. Ancak, öğrenme kuralları tarafından kullanılan ağıncı ayarları, gizli katmanlar aracılığıyla çıkış katmanından giriş katmanına geriye doğru yayılır [27]. Genel olarak, eşleme (giriş ve çıkış arasındaki) hatası kümülatiftir ve tüm eğitim seti üzerinden hesaplanır.

## 4.3. Evrimsel Sinir Ağları

Evrimsel sinir ağıncı, en az ön işlem gerektirecek şekilde tasarlanmış çok katmanlı algılayıcı modelinin bir çeşididir. Evrim katmanları, alt örneklem katmanları ve tam bağlı gizli katmanları içeren bir dizi çoklu gizli katman tarafından oluşturulur. Bu tür sinir ağlarının görüntü ve video tanıma, konuşma tanıma, sınıflandırma ve doğal dil işleme gibi birçok farklı alanda yaygın kullanımının olduğu belirtilmektedir [28]. Evrimsel sinir ağlarının belirli bir resimdeki nesnelere tanımlama veya yüz tanıma konusunda çok başarılı olduğu bilinmektedir ve el yazısı karakterlerinin tanınmasında, robot ve sürücüsüz araba teknolojilerinde vizyonu güçlendirmek için kullanılmıştır [29].

## 4.4. Tekrarlayan Sinir Ağları

Tekrarlayan sinir ağıncı, kısa ve/veya uzun vadeli bağımlılıkları keşfetmek için sıralı bilgilerin kullanıldığı ileri beslemeli sinir ağının değiştirilmiş bir versiyonudur [30]. Basitçe, geleneksel sinir ağlarında, her t zamanında, sınıflandırmada yalnızca mevcut girdi dikkate alınır. Başka bir deyişle, önceki girişlerin veya sonraki girişlerin sonuçlarından kaçınılır ve yalnızca mevcut ağa beslenen giriş değerlendirilir. Öte yandan, tekrarlayan sinir ağlarında, geçmiş çıktılarının girdi olarak kullanılmasına izin verilen gizli durumlar vardır.

## 5. Deneysel Düzenek

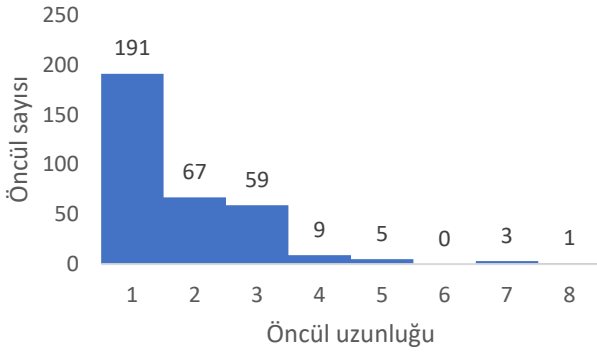
Bu bölümde deneylerimizde kullanılan veri seti, deney parametreleri ve değerlendirme ölçütleri sırasıyla sunulacaktır.

### 5.1. Veri Kümesi

Bu çalışmada, 10 Türkçe çocuk öyküsünden oluşan bir koleksiyon (toplam 1274 tümce -11613 sözcük) adil -öncül çiftleri veri kümesini oluşturmak için kullanılmıştır. Hikayeler web kaynaklarından elde edilmiş ve rastgele seçilmiştir. Çocuk hikayeleri diğer yazı türlerine göre daha anlaşılır ve kısa oldukları için tercih edilmiştir. Veri kümesinin oluşturulmasından önce lokmalama (tokenizasyon), sözcük türü (İng. Part of Speech-POS) belirleme ([30]'da sunulan POS etiketleyici ile) ve elle etiketleme görevleri gerçekleştirilmiştir.

Elle etiketlemede POS belirleyicisi tarafından adil olarak etiketlenen her sözcük (toplam 442 sözcük) incelenir. Bu inceleme aslında adil olarak etiketlenen sözcüğün uygun bir öncülü olup olmadığını kontrol etmek için yapılmaktadır.

Toplam 107 adıl etiketli sözcüğün metinde geçerli bir öncüle atıfta bulunmadığı gözlemlenmiştir. Bu atıfta bulunmayan sözcükler hariç tutulduğunda, geriye kalan adıl etiketli sözcükler (kısaca adılar) öncülleri ile eşleştirilerek pozitif adıl-öncül çift örnekleri elde edilmiştir. Bu eşleştirme işlemi sırasında bazı adıların tek bir sözcük yerine bir sözcük öbeğine gönderme yaptığı görülmüştür. Şekil 1'de öncüllerin sözcük sayısı cinsinden uzunluklarına ilişkin istatistikler verilmiştir. Yatay eksen öncüllerin uzunluğunu (içerdikleri sözcük sayısını), dikey eksen ise atıfta bulunan adıların toplam sayısını göstermektedir. Örneğin, 191 adıl in tek sözcüklük öncüllere sahip olduğu, 67 adıl in ise iki sözcüklük öncüllere atıfta bulunduğu görülmektedir.



Şekil-1: Öncül istatistikleri (öncülde bulunan sözcük sayısı cinsinden)

Adıl in birden fazla sözcükten oluşan bir öbeğe atıfta bulunduğu durumlarda, ilgili öbek sözcüklere bölünmüştür. İfadedeki her sözcük için, sözcük-adıl çiftinin bir isabet (doğru bir sınıflandırma/doğru çift) olduğu varsayılarak pozitif örnekler eklenmiştir.

Aşağıda, ikinci tümce deki "onu" adıl inin birinci tümce deki "güllerinin en güzeli" ifadesine gönderme yaptığı böyle bir duruma örnek verilmiştir.

*Güllerinin en güzeli solmamış. İyi yürekli peri, her gün onu evinin penceresinden seyrediyormuş.*

Yukarıdaki örnek için (onu, güllerinin), (onu, en) (onun, güzeli) çiftleri veri kümesine pozitif örnek olarak eklenmiştir.

Sonuç olarak veri kümesi, 227 benzersiz öncül ve 56 benzersiz adıl in bulunduğu 593 adıl -öncül çifti içermektedir. Gerçek öncülleri ile birlikte 593 adıl deneylerimizde pozitif örnekler olarak kullanılmaktadır. Negatif örnekler kümesi, farklı pencere boyutları kullanılarak deneylerde oluşturulmuştur. Buradaki pencere boyutu terimi, negatif bir çift örneği oluşturmak için adıl le eşleştirilecek sözcük sayısını (adıl ve gerçek öncülü arasında) temsil eder. Pencere boyutu (W) deneylerimizde 6 farklı değere (W= 5, 10, 15, 20, 25 ve 30) ayarlanmıştır. Kısaca, pencere boyutu W=n olduğunda, her adıl in önceki (en yakın) n sözcüğü negatif çiftlerde kullanılmıştır. Bu durumda gerçek öncül verilen pencere içindeyse, negatif örnek sayısı n-1 olmaktadır.

Çizelge 2'de verilen örnek metin göz önüne alındığında, W=n sütunları, Sözcük No=17 olan "onları" adıl i için pozitif ve negatif örnek çiftlerini göstermektedir. Örneğin W=5

olduğunda (onları, güllerini) pozitif, (onları, sabah), (onları, onu), (onları, sever), (onları, de) çiftleri ise negatif çiftler olarak etiketlenmiştir.

Çizelge 3, farklı pencere boyutlarıyla oluşturulan her bir veri kümesindeki pozitif/negatif örneklerin (çiftlerin) nihai sayısını vermektedir.

Çizelge 2. Pencere tabanlı veri seti oluşturma - Örnek metin

Sözcük No	Sözcük	W=5	W=10	W=15
1	Güller	-	-	-
2	öyle	-	-	Negatif
3	güzellermiş	-	-	Negatif
4	ki	-	-	Negatif
5	gören	-	-	Negatif
6	herkes	-	-	Negatif
7	perinin	-	Negatif	Negatif
8	güllerine	-	Negatif	Negatif
9	hayran	-	Negatif	Negatif
10	kalırmış	-	Negatif	Negatif
11	Peri	-	Negatif	Negatif
12	de	Negatif	Negatif	Negatif
13	güllerini	Pozitif	Pozitif	Pozitif
14	sever	Negatif	Negatif	Negatif
15	her	Negatif	Negatif	Negatif
16	sabah	Negatif	Negatif	Negatif
17	onları	Adıl	Adıl	Adıl
18	sularmış	-	-	-

Çizelge 3. Veri kümelerindeki pozitif/negatif örnek sayısı (çiftler)

Veri kümesi	Pencere Büyüklüğü	Pozitif Çift	Negatif Çift	Toplam
Küme-5	5	593	2881	3474
Küme-10	10	593	5714	6307
Küme-15	15	593	8575	9168
Küme-20	20	593	11468	12061
Küme-25	25	593	14369	14962
Küme-30	30	593	17298	17891

## 5.2. Deneysel Ayarlar

Bu çalışmada deneyler Tensorflow-Keras [32] ve DeepLearning4j araçları [33] kullanılarak gerçekleştirilmiştir. Tüm sınıflandırma deneylerinde, aşırı uyum sorununun üstesinden gelmek için n-kat çapraz doğrulama uygulanmıştır. Kısaca, n-kat çapraz doğrulamada veri kümesi (hem pozitif hem de negatif örnekler dahil) eşit büyüklükte n alt kümeye bölünür. n-1 alt küme eğitimde ve bir bölme testte kullanılır. Her denemede (katta), test alt kümesi daha önce test kümesi olarak kullanılmamış olan eğitim alt kümelerinden biriyle değiştirilir.

Her bir veri kümesi için (Küme-5 ... Küme-30), tüm deneyler farklı sayıda katmana ve/veya farklı sayıda nörona sahip üç farklı derin öğrenme makinesi için ayrı ayrı

gerçekleştirilmiştir. Çok katmanlı algılayıcı ağı özgü deneysel ayarlar aşağıda sunulmuştur:

- Çıktı sayısı=1
- Yineleme sayısı=1000
- Parti (İng. batch) büyüklüğü= Girdi büyüklüğünün %10'u
- Tohum değeri=6
- Algoritma öğrenme oranı =0.1

Çok katmanlı algılayıcı modeli için alanyazında yer alan çok sayıda farklı aktivasyon fonksiyonu arasından tanh aktivasyon fonksiyonu diğer aktivasyon fonksiyonlarına kıyasla daha hızlı yakınsadığı ve çok katmanlı algılayıcı sinir ağlarında daha iyi doğruluk sonuçları elde ettiği belirtildiği için deneylerimizde tercih edilmiştir [34]. Xavier yöntemi, ağırlıkların doğru olduğundan emin olmak için birçok katman boyunca sinyalleri makul bir değer aralığında tuttuğu ve çıkış aktivasyonlarının giriş aktivasyonları noktasındaki dağılımını dikkate aldığı için ağırlık ilk değere atamada kullanılmıştır [35].

Çok katmanlı algılayıcı derin sinir ağlarında iki ana parametre vardır. Birincisi ağdaki gizli katman sayısı, ikincisi ise her katmandaki nöron sayısıdır. Katman sayısındaki değişikliğe bağlı olarak performanstaki değişimi incelemek için aşağıdaki ağ türleri üzerinde deneyler yapılmıştır.

**Ağ FL.** Az katmanlı (1 girdi + 1 çıktı + 2 gizli katman)

**Ağ ML.** Orta sayıda katmanlı (1 girdi + 1 çıktı + 7 gizli katman)

**Ağ TL.** Çok katmanlı (1 girdi + 1 çıktı + 18 gizli katman)

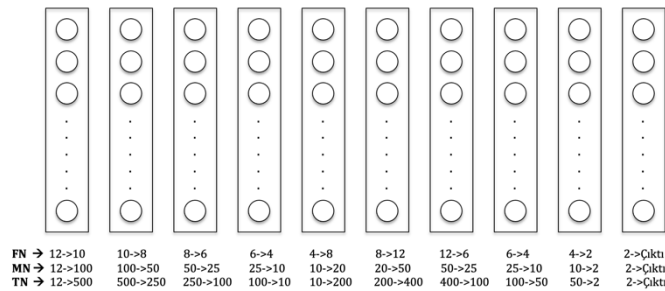
Şekil 2, nöron sayısını değiştirerek oluşturulan farklı ağ türlerini (Ağ FN, Ağ MN, Ağ TN) göstermektedir. Bu deneylerde ağdaki katman sayısı 10 olarak ayarlanmıştır. Farklı ağ türleri şu şekilde adlandırılmıştır:

**Ağ FN.** Az nöronlu

**Ağ MN.** Orta sayıda nöronlu

**Ağ TN.** Çok nöronlu

Her ağ/konfigürasyon için ilk katman girdiyi alır ve önce nöron boyutunu azaltır, sonra nöron boyutu artırılır ve daha sonra ikili sınıflandırma sonucunu döndürmek için önceden kararlaştırılan çıkış katmanı için nöron sayısı tekrar azaltılır. Örneğin, Ağ FN'de ilk katman 12 giriş değeri alır ve 10 tane üretir. İkinci katmanda 10 girdi 8 çıktı değerine düşürülür ve bu şekilde devam eder.



**Şekil-2:** Çok katmanlı algılayıcılar (Ağ FN, MN ve TN)

Çalışmamızdaki ikinci yöntem olan evrişimli sinir ağı, aşağıda verilen parametre ve deneylerimizde gerekli başlangıç değerleri ile kurulmuştur. Bunlar

- Girdi sayısı=12 (öznitelik sayısı)
- Çıktı sayısı=1
- Yineleme sayısı=1000
- Parti (İng batch) büyüklüğü= Giriş büyüklüğünün %10'u,
- Tohum değeri=6
- Algoritma öğrenme oranı =0.01

şeklinde listelenir.

Önceki çalışmalarda evrişimli sinir ağı modelleri için çok farklı aktivasyon fonksiyonları kullanılmasına rağmen, hangi aktivasyon fonksiyonunun hangi tip problemlerde daha verimli olduğu kesin değildir. [36]'da evrişimli sinir ağlarının genellikle çok katmanlı algılayıcı sinir ağlarına kıyasla daha yüksek zaman karmaşıklığına sahip olduğu, bu nedenle tasarımı daha düşük maliyetli bir fonksiyonun tercih edilmesinin gerektiği belirtilmektedir. Bu sebeple, çalışmamızda sigmoid ve tanh aktivasyon fonksiyonlarına kıyasla hesaplama açısından daha düşük maliyetli olduğu bilinen relu aktivasyon fonksiyonu kullanılmıştır. Çok katmanlı algılayıcı ağına benzer şekilde ilk ağırlık değeri atamada Xavier yöntemi kullanılmıştır. Bilişsel sinir ağı için optimizasyon algoritması olarak ise stokastik kademeli azalma (İng. stochastic gradient descent) tercih edilmiştir.

Çok katmanlı algılayıcı deneylerine benzer şekilde, evrişimli derin sinir ağı deneyleri için katman sayısı ve nöron sayısı ana parametreler olarak kabul edilmiş ve 6-kat çapraz doğrulama yaklaşımı kullanılmıştır. Deneylerimizde ilk olarak katman sayısı filtre değerleri üzerinden değiştirilmektedir. İkinci olarak, toplam katman sayısı sabitlenerek nöron sayıları değiştirilmiştir.

Evrişimli katman ve alt örnekleme katmanlı bir evrişimli sinir ağı oluşturulurken katman sayısının belirlenmesi için kernel ve stride değerlerinin önceden hesaplanması gerekmektedir çünkü her katmanın çıkış hacminin boyutu, kernel ve kaydırma adım (İng. stride) değeri ile hesaplandığında sistemdeki katman sayısını değiştirebilmektedir. Katman sayısındaki değişime bağlı olarak performanstaki değişimi incelemek için çeşitli ağ türleri üzerinde deneyler gerçekleştirilmiştir. Bu ağ türleri

**Ağ VFL.** Çok az katmanlı (1 girdi + 1 çıktı + 1 gizli katman)

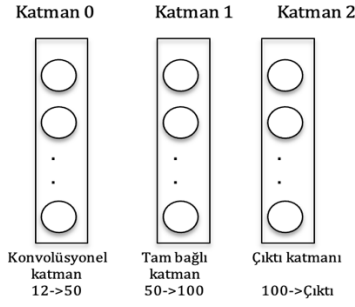
**Ağ FL.** Az katmanlı (1 girdi + 1 çıktı + 2 gizli katman)

**Ağ ML.** Orta sayıda katmanlı (1 girdi + 1 çıktı + 6 gizli katman)

**Ağ TL.** Çok katmanlı (1 girdi + 1 çıktı + 18 gizli katman)

şeklinde kurgulanmıştır.

Katman sayısındaki değişimin etkisinin incelenebilmesi için her katmandaki nöron sayısı tek bir değere sabitlenmiştir. Şekil 3'te Ağ VFL örnek olarak verilmiştir. Ağ VFL'de toplam 3 katman bulunmaktadır. Bilişsel katman olan 0. katman, çekirdek değeri 12 ve adım değeri 1 olan 12 girişe sahiptir. Bu katman 12 değer arasında herhangi bir korelasyon bulmaya çalışır ve daha sonra tam bağlantılı katmana 50 girdi üretir. Tam bağlı katman 50 girdi alır ve girdi hacmini 100'e çıkarır ve çıktı değerine karar vermek için çıktı katmanına gönderir.



**Şekil-3:** Ağ VFL – Çok az katmanlı evrişimli sinir ağı

Şekil 4, katman sayısının 7'ye sabitlendiği, nöron sayısının değiştirilmesiyle oluşturulan farklı evrişimli sinir ağı konfigürasyonlarını göstermektedir. Bu ağlar aşağıda listelenmiştir:

**Ağ FN.** Az nöronlu

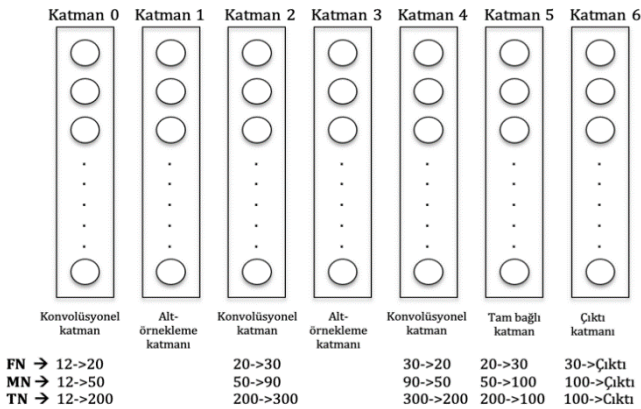
**Ağ MN.** Orta sayıda nöronlu

**Ağ TN.** Çok nöronlu

Tüm ağ türlerinde, sıfırıncı katman, çekirdek değeri 2 ve adım değeri 2'ye eşit olan 12 giriş değerini alan evrişimli katmandır. Sonraki tüm evrişimli ve havuzlama katmanlarında, çekirdek ve adım değeri sırasıyla 2 ve 1'e eşit olarak ayarlanır.

Çalışmamızda yer alan üçüncü makine tekrarlayan sinir ağıdır. Deneylerimizde tekrarlayan sinir ağının parametreleri ve gerekli başlangıç değerleri aşağıdaki gibi ayarlanmıştır:

- Girdi sayısı=12 (öznitelik sayısı)
- Çıktı sayısı=1
- Gizli katman(lar)daki birim sayısı =4
- Tohum değeri=6
- Çıktı dizisi= Doğru(çıktı dizisindeki son çıktı geri döndürülür)



**Şekil-4:** Evrişimli sinir ağları (Ağ FN, MN ve TN)

Tekrarlayan sinir ağı deneylerinde, çok katmanlı algılayıcıya kıyasla ağlarla ilgili karmaşık yapıları nedeniyle katman sayısı ana parametre olarak kabul edilmektedir. Çalışmamızda 4 kat çapraz doğrulama yaklaşımını kullanılmış; giriş katmanı bir gömme katmanı olarak ayarlanmış ve çıkış katmanı, çıkış sayısını bire düşüren yoğun bir katman olarak tasarlanmıştır. Tekrarlayan ağlarda gizli katman sayısındaki değişikliğe bağlı olarak performanstaki değişimi incelemek için aşağıdaki ağ türleri ile deneyler gerçekleştirilmiştir:

**Ağ FL.** Az katmanlı (1 giriş + 1 çıkış + 2 gizli katman)

**Ağ ML.** Orta sayıda katmanlı (1 giriş + 1 çıkış + 7 gizli katman)

Derin öğrenme ile adıl çözümlene performansının ölçümü için F1- ölçüsü kullanılmıştır. F1 ölçüsü aşağıdaki şekilde ifade edilir.

$$F1 = \frac{2TP}{2TP+FN+FP} \quad (1)$$

Burada TP doğru pozitiflerin sayısı (doğru çiftler olarak sınıflandırılan ve gözlemlenen çiftler), FP yanlış pozitiflerin sayısı (yanlış bir şekilde adıl -öncül çifti olarak sınıflandırılan çiftler) ve FN yanlış negatiflerin sayısıdır (aslında doğru çiftler olan ancak yanlış çiftler olarak sınıflandırılan çiftler).

## 6. Deney Sonuçları

Bu çalışmada n-kat çapraz doğrulama yöntemiyle toplam 15 farklı ağ türü ile sınıflandırma deneyleri gerçekleştirilmiştir. Aşağıda sırasıyla çok katmanlı algılayıcı, evrişimli ve tekrarlayan sinir ağlarının deneysel sonuçları ağırlıklandırılmış F1 ölçüsü ile verilecektir. Son olarak, üç derin sinir ağının sınıflandırma performansları karşılaştırılacaktır.

Çizelge 4'te farklı katman sayısına sahip çok katmanlı algılayıcı sinir ağlarının performans sonuçları verilmiştir. Çizelge 4'te W, veri kümelerindeki negatif örnek sayısını değiştiren pencere boyutunu temsil etmektedir ve sunulan performans değerleri 6-katlı çapraz doğrulamanın ağırlıklı F1 ortalamalarından oluşmaktadır. Her sütundaki altı çizili değerler ilgili ölçümde elde edilen en yüksek değerleri, son satır ise tüm pencere boyutları için ortalama değerleri vermektedir. Son satırdaki koyu değerler, üç ağ türü dikkate alındığında ilgili ölçütün en yüksek ortalama değerini göstermektedir.

**Çizelge-4:** Deney sonuçları – Farklı katman sayılarına sahip çok katmanlı algılayıcı sinir ağları

W	Ağ FL	Ağ ML	Ağ TL
5	0,936	0,943	0,939
10	0,868	0,881	0,879
15	0,824	0,830	0,827
20	0,784	0,798	0,785
25	0,759	0,765	0,797
30	0,745	0,748	0,731
Ortalama	0,819	0,828	0,826

Farklı sayıda katmana sahip çok katmanlı algılayıcı ağlar üzerinde yapılan deneyler göstermiştir ki

1. F1 ölçüsü dikkate alındığında, (tüm ağ türlerinde) derin ağ pencere boyutu 5 olarak ayarlandığında en yüksek başarı elde edilmiştir.
2. Ortalama değerler incelendiğinde (Çizelge 4'te son satır), en yüksek F1 ölçüsü sinir ağının orta katman sayısına sahip olduğu durumda (Ağ ML) elde edildiği görülmüştür.

Farklı nöron sayılarına sahip çok katmanlı algılayıcı ağının değerlendirme sonuçları Çizelge-4'e benzer şekilde Çizelge 5'te verilmiştir.

**Çizelge-5: Deney sonuçları – Farklı nöron sayılarına sahip çok katmanlı algılayıcı sinir ağları**

W	Ağ FN	Ağ MN	Ağ TN
5	0,937	0,942	0,946
10	0,868	0,883	0,888
15	0,822	0,834	0,842
20	0,788	0,798	0,804
25	0,749	0,775	0,785
30	0,659	0,738	0,749
Ortalama	0,804	0,828	0,836

Farklı nöron sayısına sahip çok katmanlı algılayıcı ağlar üzerinde yapılan deneyler aşağıdaki önemli sonuçları ortaya çıkarmaktadır:

1. Tüm ağ türlerinde, en yüksek F1 ölçüsü pencere boyutu (W=5) en küçük olduğunda elde edilmektedir.
2. F1 ölçüsünün ortalama değerleri göz önüne alındığında, çok fazla nöron kullanıldığında (Ağ TN) sınıflandırma sisteminin en yüksek başarıma ulaştığı görülmektedir.
3. F1 ortalama değerlerine bakıldığında, farklı ayarlarda doğruluk değerleri benzer olsa da çok fazla nöron kullanıldığında F1 değerinin (F1=0,836) çok daha yüksek olduğu görülmektedir.

Evrışimli sinir ağı deneylerinde çapraz doğrulamada her kat için tekrarlanacak toplam 7 (4 farklı katman sayısı + 3 farklı nöron sayısı) deney bulunmaktadır. Farklı katman sayılarıyla (Tip VFL, FL, ML, TL) yapılan deneylerin sonuçları Çizelge-6'da verilmiştir. Burada W pencere boyutunu, sütunlar 6 katın ağırlıklı F1 ortalamalarını göstermektedir. Çizelge-6'da altı çizili değerler ilgili sütun için en yüksek değerlerdir. Ortalama satırı sütunlardaki değerlerin ortalamalarını gösterir ve mean yüksek ortalama değer(ler) koyu olarak verilmiştir.

**Çizelge-6: Deney sonuçları – Farklı katman sayılarına sahip evrışimli sinir ağları**

W	Ağ VFL	Ağ FL	Ağ ML	Ağ TL
5	0,934	0,932	0,933	0,929
10	0,872	0,868	0,858	0,860
15	0,826	0,817	0,820	0,820
20	0,786	0,784	0,773	0,780
25	0,758	0,761	0,746	0,729
30	0,742	0,744	0,738	0,712
Ortalama	0,820	0,818	0,811	0,820

Farklı katman sayısına sahip evrışimli sinir ağları üzerinde yapılan deneyler, F1 ölçüsü dikkate alındığında ağın tüm türler için W=5 olduğunda en başarılı olduğunu göstermektedir. Ortalama performans değerleri göz önüne alındığında (Çizelge 6'da son satır), tek bir ağ türünün diğerlerinden anlamlı derecede daha başarılı olmadığı görülmüştür. Sonuç olarak, adil çözümleme probleminde evrışimli ağda çok fazla katmana ihtiyaç olmadığı söylenebilir.

Farklı nöron sayılarının etkisini gözlemlemek için evrışimli sinir ağı modeli üzerinde yapılan deneylerin sonuçları Çizelge- 6 ile aynı notasyon/yapı kullanılarak Çizelge-7'de gösterilmiştir. Çizelge-7'deki sonuçlar incelendiğinde, en yüksek F1 değerlendirme puanlarını elde etmek için W=5 penceresinin gerekli olduğu açıkça görülmektedir. Farklı nöron sayıları dikkate alındığında tüm farklı ağlar için neredeyse aynı maksimum performans sonuçları elde edilmiştir.

Çizelge-7'deki ortalama değerler incelendiğinde, evrışimli ağlarda az sayıda nöron yerine orta sayıda (Ağ MN) veya çok fazla sayıda nöron (Ağ TN) kullanılmasının tercih edilmesi gerektiği görülmektedir.

**Çizelge-7: Deney sonuçları – Farklı nöron sayılarına sahip evrışimli sinir ağları**

W	Ağ FN	Ağ MN	Ağ TN
5	0,933	0,935	0,935
10	0,857	0,860	0,862
15	0,809	0,820	0,821
20	0,767	0,769	0,773
25	0,738	0,743	0,746
30	0,735	0,737	0,736
Ortalama	0,807	0,811	0,812

Çizelge 8'de önceki çizelgelere benzer şekilde az ve orta sayıda katmana sahip tekrarlayan sinir ağı deneylerinin F1 sonuçları sunulmaktadır. Çizelge-8'de W pencere boyutunu ifade etmektedir. Ortalama satırı F1 değerlerinin ortalamalarını gösterir ve en yüksek ortalama değer koyu olarak verilmiştir.

**Çizelge-8: Deney sonuçları – Farklı katman sayılarına sahip tekrarlayan sinir ağları**

W	Ağ FL	Ağ ML
5	0,916	0,918
10	0,733	0,764
15	0,630	0,657
20	0,409	0,525
25	0,327	0,441
30	0,351	0,100
Ortalama	0,561	0,567

Farklı sayıda katmana sahip tekrarlayan sinir ağları üzerinde yapılan deneyler, W arttıkça F1 ölçüsünün önemli ölçüde azaldığını göstermektedir. Tekrarlayan sinir ağlarının birçok doğal dil işleme probleminde başarılı olduğu bilinmesine rağmen, çalışmamızda önceden belirlenmiş özniteliklerin bağlılık bilgisi içermediği bilindiği için deneysel sonuçlarımız şaşırtıcı değildir.

Bilişsel, tekrarlayan ve çok katmanlı algılayıcı sinir ağlarının adil çözümlemedeki performans karşılaştırmasında farklı ağ türleri için elde edilen F1 sonuçları dikkate alınmıştır. Bu karşılaştırmada kullanılacak olan F1 değerleri farklı büyüklükteki veri kümelerinden elde edilen puanların ortalamasıdır. Çizelge-9 ve Çizelge-10'da sırasıyla farklı katman sayısı ve farklı nöron sayısına sahip ağlar için F1



değerleri verilmiştir. Çizelge 9-10'daki ortalama satırları ilgili sütundaki değerlerin ortalamasını, koyu değerler ise ilgili ölçütte elde edilen en yüksek değerleri temsil etmektedir.

**Çizelge-9: Farklı katman sayısına sahip evrişimli, çok katmanlı algılayıcı ve tekrarlayan ağlar için ortalama F1 değerleri**

	Bilişsel	Çok katmanlı algılayıcı	Tekrarlayan
Ağ VFL	0,844	-	-
Ağ FL	0,843	0,839	0,561
Ağ ML	0,837	0,851	0,567
Ağ TL	0,832	0,844	-
Ortalama	0,839	0,845	0,564

**Çizelge-10: Farklı nöron sayısına sahip evrişimli ve çok katmanlı algılayıcı ağlar için ortalama F1 değerleri**

	Bilişsel	Çok katmanlı algılayıcı
Ağ FN	0,833	0,831
Ağ MN	0,837	0,849
Ağ TN	0,838	0,858
Ortalama	0,836	0,846

Çizelge 9'da yer alan

1. En yüksek F1 değeri orta sayıda katmanla oluşturulan çok katmanlı algılayıcı ile elde edilir
2. En yüksek ortalama F1 puanı çok katmanlı algılayıcı ile elde edilir

sonuçları değerlendirildiğinde çok katmanlı algılayıcı sinir ağının evrişimli ve tekrarlayan ağlardan daha yüksek başarımlar gösterdiği sonucu çıkarılabilir.

Çizelge 10'da verilen deneysel sonuçlara göre ise en yüksek başarımların çok fazla nörona sahip çok katmanlı algılayıcı ağ ile elde edildiği (Çizelge 10, Ağ TN'de değerlerin altı çizilmiştir) ve ortalama değerler açısından da bu ağın evrişimli sinir ağına göre daha başarılı olduğu görülmektedir. Çizelge 9 ve 10'daki sonuçlar göz önüne alındığında, çok fazla nörona (Ağ TN) ve orta katman sayısına (Ağ ML) sahip çok katmanlı algılayıcı sinir ağının en yüksek performansı elde ettiği söylenebilir. Sonuç olarak, daha uzun çalışma süresi gerektiren evrişimli sinir ağı yerine, çok katmanlı algılayıcı sinir ağının Türkçede daha önce bahsedilen özelliklerle adil çözümleme probleminde kullanılabileceği görülmüştür.

## 7. Sonuç

Bu makalede, adil çözümlemesi ikili bir sınıflandırma problemi olarak kabul edilerek Türkçe adil-öncül çifti adaylarının öznelik değerleri derin öğrenme makinelerine girdi olarak verilmiştir. Derin öğrenme makinelerinin ilgili problemi çözmedeki başarımlarını araştırılmıştır. Deneyleri gerçekleştirmek için 10 çocuk hikayesinden oluşan bir koleksiyon derlenmiş ve 593 pozitif adil-öncül çiftinden oluşan bir veri kümesi hazırlanmıştır. Veri kümesi, farklı boyutlarda veri kümeleri elde etmek için farklı sayıda negatif örneklerle büyütülmüştür. Derin öğrenmede kullanılmak üzere 12 öznelik tanımlanmıştır.

Çok katmanlı algılayıcı, evrişimli ve tekrarlayan sinir ağları olmak üzere üç derin öğrenme makinesinin performansı farklı konfigürasyonlar dikkate alınarak ölçülmüştür. İlk deney setinde, ağlardaki katman sayısı değiştirilmiştir. Performanslar (çok) az, orta ve yüksek sayıda katman için ölçülmüştür. İkinci sette, her katmandaki nöron sayısı az, orta veya yüksek olarak ayarlanmıştır. Çok fazla nöron kullanan orta sayıda (9) katmana sahip çok katmanlı algılayıcı sinir ağının en yüksek başarımlarını verdiği gözlemlenmiştir.

## Kaynakça

- [1] Grosz B. J., Weinstein S., Joshi A.K., *Centering: A Framework for Modelling the Local Coherence of Discourse Centering: A Framework for Modelling the Local Coherence of Discourse*, Computational Linguistics, 1995, 21: 203–225.
- [2] Halliday M.A., Hasan R., *Cohesion in English 1st ed.*, Routledge, <https://www.routledge.com/Cohesion-in-English/Halliday-Hasan/p/book/9780582550414> UK, 1976, 1-392.
- [3] Van Valin R.D., LaPolla R.J., *Syntax: Structure, Meaning, and Function 1st ed.*, Cambridge University Press, UK, 1997, 1-744.
- [4] Van Valin R.D., *A Summary of Role and reference Grammar*, Papers of the Summer Institute of Linguistics, University of North Dakota Session, 2005, 37(5):1–30.
- [5] Smeaton A.F., *Progress in the application of natural language processing to information retrieval tasks*, Computing Journal, 1992, 35(268):268-278.
- [6] Yıldırım S., Kılıcaslan Y., *A Machine Learning Approach to Personal Pronoun Resolution in Turkish*, FLAIRS Conference, USA:Florida, 2007, 269–270.
- [7] Kılıcaslan Y., Guner E.S., Yıldırım S., *Learning-based pronoun resolution for Turkish with a comparative evaluation*, Computers Speech and Language, 2009, 23: 311–331.
- [8] Kornfilt J., *Turkish 1st ed.*, Routledge, UK, 1997, 1-608.
- [9] Hamann J., Hamann J., *On the Syntax and Morphology of Double Agreement in Lavukaleve*, In agreement to Sebastian Bank, Doreen Georgi & Jochen Trommer (eds.) Linguistische Arbeits Berichte, 2010, 197–225.
- [10] Tufekci P., Kılıcaslan Y., *A Computational Model for Resolving Pronominal Anaphora in Turkish Using Hobbs' Naïve Algorithm*, Journal of Computing and Information Science in Engineering, 2007, 1: 854–858.
- [11] Hobbs J.R., *Resolving pronoun references*, Lingua, 1978, 44: 311–338.
- [12] Lappin S., Leass H.J., *An Algorithm for Pronominal Anaphora Resolution*, Computational Linguistics, 1994, 20: 535–561.
- [13] Kennedy C., Boguraev B., *Anaphora for Everyone : Pronominal Anaphora Resolution without a Parser*, Proceedings of the 16th International Conference on Computational Linguistics, 1996, 113–118.
- [14] Aone C., Bennett S., *Evaluating automated and manual acquisition of anaphora resolution strategies*, Proceedings of 33rd Annual Meeting of the Association for Computational Linguistics, Massachusetts, 1995, 122–129.
- [15] McCarthy J.F., Lehnert W.G., *Using Decision Trees for Coreference Resolution*, Proceedings of Fourteenth International Joint Conference of Artificial Intelligence, Montreal, Quebec, Canada, 1995, 1–5.

- [16] Soon W.M., Ng H.T., Lim D.C.Y., *A Machine Learning Approach to Coreference Resolution of Noun Phrases*, Computational Linguistics, 2001, 27: 521–544.
- [17] Mitkov R., Evans R., Orasan C., *A New, Fully Automatic Version of Mitkov's Knowledge-Poor Pronoun Resolution Method*, Proceedings of CICLing2002, Mexico City, Mexico, 2002, 2276: 168–186.
- [18] Preiss J., *Choosing a Parser for Anaphora Resolution*, Proceedings of 4th Discourse Anaphora Anaphora Resolution Colloq. (DAARC 2002), Lisbon, 2002, 175–180.
- [19] Turan U.D., *Null vs. Overt Subjects in Turkish Discourse: A Centering Analysis*, Phd. Thesis, University of Pennsylvania, Pennsylvania, 1995, 1-27.
- [20] Yuksel O., Bozsahin C., *Contextually appropriate reference generation*, 2002, Natural Language Engineering, 8: 69–89.
- [21] Chomsky N., *on Binding*, Linguist Inquires; 1980, 11: 1–46.
- [22] Erguvanli-Taylan E., *Pronominal versus zero representation of anaphora in Turkish*, Studies in Turkish Linguistics, 1986, 209–231.
- [23] Tin E., Akman V., *Situated Processing of Pronominal Anaphora*, Verarbeitung Natürlicher Spr, 1998, 369–378.
- [25] Elizondo D., *The Linear Separability Problem: Some Testing Methods*, IEEE Transactions Neural Network, 2006, 17: 330–344.
- [24] Zipf, G. K. *Human behavior and the principle of least effort*, Cambridge: Addison Wesley Press, 1949.
- [26] Pinkus A., *Approximation theory of the MLP model in neural networks*, Acta Numerica, 1999, 8(143): 143-195.
- [27] Zurada, J.M., *Introduction to Artificial Neural Systems* 1st ed., West Group, Minnesota:St. Paul,1992, 1-759.
- [28] Oord A.V.D., Dieleman S., Schrauwen B., *Deep content-based music recommendation*, Advances in Neural Information Processing Systems 26 (NIPS 2013), Nevada, 2013, 643-2651.
- [29] Sermanet P., Eigen D., Zhang X., et al., *OverFeat: Integrated Recognition, Localization and Detection using Convolutional Networks*, 2nd International Conference on Learning Representations (ICLR 2014), Banff, Canada, 2014, 1-17.
- [30] Marhon, S.A., Cameron, C.J.F., Kremer, S.C., *Recurrent Neural Networks*, Handbook on Neural Information Processing. Intelligent Systems Reference Library, Springer, Berlin, Heidelberg, 2013, 49:29-65.
- [31] Kisla T., Karaoglan B., *A hybrid Statistical Approach to Stemming in Turkish: An Agglutinative Language*, Anadolu University-Journal of Science and Technology, 2016, 17: 401–412.
- [32] Abadi, M., Agarwal, A., Barham, P., et.al., *Tensorflow: large-scale machine learning on heterogeneous distributed systems*, Arxiv:1603.04467, 2016.
- [33] <https://deeplearning4j.konduit.ai>. Access date: 29.12.2021.
- [34] Ruck D.W., Rogers S.K., Kabrisky M., et al., *Letters: The Multilayer Perceptron as an Approximation to a Bayes Optimal Discriminant Function*, IEEE Transactions on Neural Networks, 1990, 1: 296–298.
- [35] Glorot X., Bengio Y., *Understanding the difficulty of training deep feedforward neural networks*, Proceedings of 13th International Conference on Artificial Intelligence and Statistics (AISTATS), Italy, 2010, 9: 249–256.
- [36] Krizhevsky A., Sutskever I., Geoffrey E.H., *ImageNet Classification with Deep Convolutional Neural Networks*, Advances in Neural Information Processing Systems, 2012, 25:1–9.

# Makine Öğrenmesi Yöntemleri ile Kan Sayımında Covid-19 Tanısı

## Diagnosis of Covid-19 by Machine Learning from Blood Test

Büşra ÇAKI

Ege Üniversitesi

Bilgisayar Mühendisliği Bölümü

İzmir - Türkiye

govbusra@gmail.com

ORCID:0000-0002-2804-4051

Ahmet EGESÖY

Ege Üniversitesi

Bilgisayar Mühendisliği Bölümü

İzmir -Türkiye

ahmet.egesoy@ege.edu.tr

ORCID:0000-0002-5050-5547

Yasemin TOPALOĞLU

Ege Üniversitesi

Bilgisayar Mühendisliği Bölümü

İzmir - Türkiye

yasemin.topaloglu@ege.edu.tr

ORCID:0000-0003-28161984

### Öz

Sağlık alanında kullanılan yapay zekâ teknolojileri, makine öğrenmesi yöntemleri öncülüğünde; erken tanı, değerlendirme ve karar verme gibi pek çok alanda etkili olmaktadır. Bu teknolojiler küresel bir salgına neden olmuş olan Covid-19 hastalığına tanı koymak ve gelişiminin izlenmesinde önemli başarı göstermiştir. Bu çalışmada hastalığı tanılamak ile birlikte, şiddetini ve bulaşıcılık düzeyinin saptanmasında, makine öğrenme yöntemlerini deneyimleyen çalışmalar taranmıştır. Bu incelemede tüm deneyimlerin dikkate alınması ve anlamlı sonuçlara ulaşılması amacıyla bazı eksik veriler tamamlanmış benzer çalışmalar birleştirilmiş ve sonuçlar karşılaştırılabilir hale getirilmiştir. Ayrıca hata maliyet analizini temel alarak değerlendirme ölçülerine F-β ölçütleri de katılmıştır. Yapay sinir ağı yönteminin hastalığın tanılanması ve şiddetini belirlemede başarılı olduğu görülmektedir. Bulaşma hızının belirlenmesinde ise hangi yöntemin daha iyi olduğunu belirlemede kullanılabilecek güvenilir bir model henüz yoktur.

**Anahtar sözcükler:** Makine öğrenmesi, Covid-19, gerçek-zamanlı ters transkripsiyon polimeraz zincir reaksiyonu, kan sayımı.

### Abstract

Artificial intelligence technologies that are used in healthcare are led by machine learning methods and are effective in many areas such as early diagnosis, assessment and decision making. These technologies have shown significant success in diagnosing and monitoring the course of Covid-19 disease, which has caused a global epidemic. In this paper, studies that experimented with machine learning methods in diagnosing the disease, determining its severity and level of contagion

were reviewed. In the review process, in order to take all of the experiences into account and reach meaningful outcomes, some missing data were completed and similar studies were combined so as to make the results comparable. In addition, F-β criteria were included in the evaluation metrics based on error cost analysis. The artificial neural network method was found to be successful in diagnosing the disease and determining its severity. There is not yet a reliable model that can be used, to determine which method is better in determining the infection rate.

**Keywords:** Machine learning, Covid-19, real-time reverse transcription polymerase chain reaction, blood analysis.

### 1. Giriş

Koronavirüs Hastalığı (Covid-19), 31 Aralık 2019 tarihinde Çin'in Wuhan Eyaletinde ortaya çıkışının ardından kısa sürede tüm dünyaya yayılarak, Dünya Sağlık Örgütü (WHO) tarafından salgın hastalık unvanını kazanmıştır [1]. Aşı ve ilaçların keşfine kadar dünya çapında bu hastalıkla mücadelede en önemli dönüm noktası erken tanılama başarılması olmuştur [1,2]. Hastalığın varlığının ve şiddetinin saptanması, hastalığın tedavisi ve bulaşmanın önlenmesi açısından hayati önem taşımaktadır.

Hastalığın tanılanmasında RT-PCR Sinaması olarak da bilinen Tanı Sinaması ve halk arasında Kan Sinaması olarak anılan Antikor Sinaması kullanılmaktadır. Bir moleküler genetik yöntemi olan RT-PCR sinamasında enfeksiyonun varlığı, bizzat antijenin nükleik asit dizileri üzerinden saptanır. RT-PCR sinama verilerinin etkili biçimde yorumlanmasında ne yazık ki hala zorluklar yaşanmaktadır [3]. Diğer bir tanı yöntemi olan Antikor sinaması ise daha önce korona-virüs hastalığına yakalanıp yakalanmadığınızı gösteren bir kan sinamasıdır. Bu sinama genellikle virüs vücudu tamamen terk ettikten sonra yapılmaktadır ve kanda virüse karşı antikor gelişip

gelişmediğini saptamaktadır. Yapılan gözlemler, Antikor sinamalarının %100 bulma ve %99 tutturma oranlarına ulaştığını göstermektedir [4]. Hastalık tanısında antikor sinamalarının RT-PCR sinama sonuçlarına göre daha güvenilir ve etkili olduğuna ilişkin kanılar bulunmaktadır [5,6].

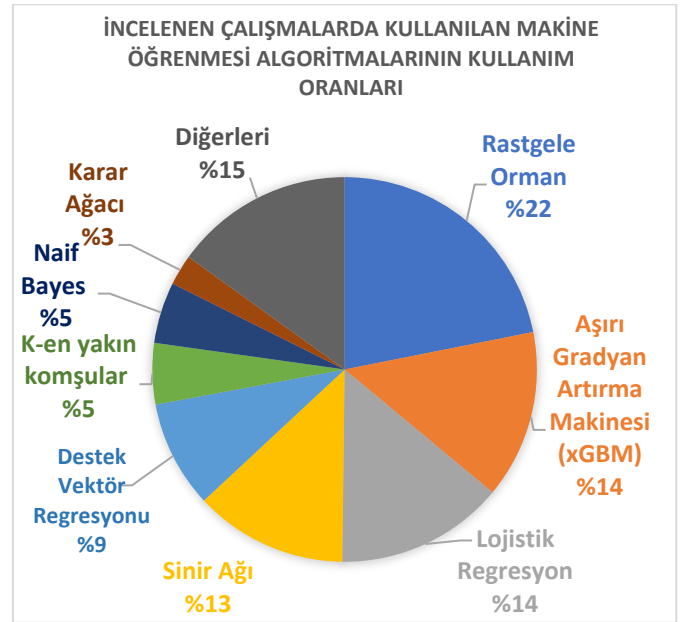
Kan sinamalarının Covid-19 tanılmasında ve hastalığın şiddetini belirlemedeki başarısı, bu tür sinamaların ön plana çıkmasını sağlamaktadır [6,7]. Karar verme aşamasında kan ölçümlerinin yanı sıra, yardımcı veriler olarak ulamsal değişkenlerin de (cinsiyet, yaş ve nefes darlığının varlığı gibi verilerin) hastalığın tanısında etkili olabileceği saptanmıştır [8,9].

Covid-19 tanısı, sadece tıbbi bir sorun olmanın ötesinde, felaketin ekonomik etkileri ve insan hayatının ciddi tehlikeye girmesi nedeniyle, güvenilir veri analizi, eşgüdüm, ölçütleme ve otomasyon gibi bilgi-işlem sorunlarını da beraberinde getirmektedir. Bu karmaşıklık, sadece tıp alanında değil, aynı zamanda bilgi teknolojileri ve veri yönetimi alanlarında da etkili çözümler gerektirmektedir.

Son zamanlarda geliştirilen yapay zekâ temelli makine öğrenmesi teknolojileri, çeşitli ölçütleri, özellikleri ve görüntüleri bir araya getirme yetenekleri sayesinde, hastalık tanısı ve sınıflandırılmasında önemli bir rol oynamıştır [10]. Bunun yanında salgının, ülkelerin mücadele kapasitesini zorlaması ve kontrolden çıkma riskinin oluşması nedeniyle, vaka sayılarındaki zaman içinde oluşan değişimlerin öngörülmesi alanında da çalışmalar yapılmıştır. Bu konuda başarılı bir çalışma [11]'de polinom regresyon yöntemiyle yapılmış olan modellemedir. Bu tür modellerin güvenilirlik kazanması durumunda, gelecekte ortaya çıkabilecek benzer salgınlarda da hastanelerde bulundurulması gereken aşı stoku veya boş yatak kapasitesi gibi kaynakların planlanması için önemli birer rehber olarak kullanılabilirliklerini belirtmek önemlidir.

Salgınla mücadele kapsamında, tanı amaçlı olarak bir dizi algoritmaya dayanan çok sayıda makine öğrenmesi biçimleri önerilmiştir [12,13]. Bu önerilen makine öğrenmesi biçimleri salgının etkilerini anlamak ve daha etkili teşhis yöntemleri geliştirmek amacıyla geniş bir veri kümesini değerlendirmektedir. Makine öğrenmesi yöntemlerinin girdisi olarak, RT-PCR sinama sonuçları ve kan sinaması gibi kritik sağlık verileri üzerine odaklanılmıştır. Bu veri tabanlı yaklaşımlar, hastalığın tanılanması için daha kesin ve hızlı sonuçlar elde etmeyi amaçlamaktadır. Kaynaklarda yer alan çalışmalarda en çok kullanılan makine öğrenme algoritmalarının başında %21,75 ile Rastgele Orman ve %14,10 oranıyla aşırı Gradyan Artırma makinesi olduğu gözlemlenmektedir.

Şekil-1'de hastalığın saptanması, şiddet derecesinin veya bulaşıcılık hızının belirlenmesi konusundaki incelemeye alınmış çalışmalarda farklı makine öğrenmesi algoritmalarının kullanım oranları görülmektedir.



Şekil 1: İncelenen Covid-19 çalışmalarında en çok kullanılan makine öğrenmesi algoritmalarının kullanım oranları.

Bu makalede, Covid-19 ile ilişkili incelemelerde makine öğrenmesinden yararlanan çalışmalar başarımları açısından incelenmiştir. İkinci bölümde hastalığın tanılanmasında kullanılan yöntemler karşılaştırılmıştır. Üçüncü bölümde makine öğrenmesi yöntemleri ile hastalığın şiddetinin belirlenmesine, dördüncü bölümde ise hastalığın bulaşma hızının belirlenmesine yönelik çalışmalar incelenmiştir. Beşinci bölümde elde edilen veriler değerlendirilmiştir. Altıncı bölüm ise sonuç bölümüdür.

## 2. Makine Öğrenmesi ile Covid-19 Tanılama

Makine öğrenmesi yöntemleri, hastalığın otomatik tanılanması için yapılan uygulamalarda hem en kritik hem de en başarılı olarak değerlendirilmektedir. Klinik özellikler, kan değişkenleri ve ölçülen değerlerin, kabul edilen normal aralıklardan yaptıkları sapmalar, tanılama amaçlı algoritmalar için önemli birer veri kaynağıdır [14,15,16]. Algoritma başarımları değerlendirilirken kullanılan veri kümelerindeki özneliklerin seçimi ve uygulanan filtreleme işlemlerinin de dikkate alınması gerekmektedir.

[17] çalışması, makine öğrenme yöntemlerinden rastgele orman, yapay sinir ağı ve eniyileme algoritması kullanarak hastalığın teşhisine odaklanmıştır. Veri kümesi 08 Mart – 05 Nisan 2020 tarihleri arasında Sheba Tıp Merkezi'nden hastaneye başvuran 162 hastanın verileri ile oluşturulmuştur. Değişkenler cinsiyet, yaş, komorbidite, obezite, yoğun sigara içme, diyabet, astım, kalp hızı, solunum hızı, beyaz kan hücre sayısı, belirtilerin gözlemlenmesinden hastaneye yatışa kadar geçen süre ve oksijen doyumu (satürasyonu) olarak belirlenmiştir. Çalışmada kullanılan veri kümesi %70 eğitim ve %30 sinama verisi olmak üzere ikiye ayrılmıştır. Başarım değerlendirmesinde hata matrisi, eğri altında kalan alan ve beklenen ölüm skoru göz önüne alınmıştır. 12 fizyolojik parametre, yaş ve önceki sağlık durumu bilgisine dayalı beklenen ölüm ölçüsü (mulbsta-II): 0,92 bulunmuştur. En başarılı model olarak belirlenen rastgele orman ile alıcı işlem

karakteristik eğrisi (ROC Eğrisi) altında kalan alan: 0,93, bulma: 0,88, tutturma: 0,92 ve doğruluk: 0,92 olmuştur. Bu sonuçlar, makine öğrenmesi yöntemlerinin hastalık teşhisi açısından başarılı ve güvenilir bir araç olabileceğini göstermektedir.

[8] ile, hastalığın teşhisinde rastgele orman, gradyan artırma makinesi, lightboost, catboost ve topluluk öğrenme algoritmalarından torbalama algoritmasını kullanarak bir çalışma gerçekleştirilmiştir. Veri kümesi, Brezilya'nın São Paulo kentindeki Israelta Albert Einstein Hastanesine 28 Mart 2020 ile 3 Nisan 2020 tarihleri arasında gelenler arasında rastgele seçilen 600 kişiye ait kan değerleri ve ulamsal değişkenlerden oluşturulmuştur. Bu verileri kararlı kılmak için sentetik azınlık aşırı örnekleme tekniği (Smote) kullanılmıştır. Veri kümesi üzerinde kan gruplarının etkisini görmek için normallik sınamai (Shapiro) ve parametrik olmayan istatistiksel sınama (Whitney-U) yapılmıştır. Başarım değerlendirmesinde ise doğruluk, bulma, tutturma, kesinlik, f1 puanı ve uyum (kappa) istatistikleri incelenmiştir. Elde edilen en iyi sonuçlar; 0,87 doğruluk, 0,88 bulma, 0,89 kesinlik, 0,87 f1 ölçüsü ve 0,36 uyum ölçütü ile rastgele orman algoritmasında görülmüştür.

[18] çalışmasında, 279 hastadan alınan kan sınamaları kullanılmış ve eksik veri kümesini Zincirleme Denklemlemlerle Çoklu Atama (Multivariate Imputation by Chained Equations – MICE) yöntemiyle tamamlanmıştır. Değişken olarak yaş, lökosit, trombosit, CRP, AST, ALT, GGT, LDH, nötrofiller, lenfositler, monositler, eozinofiller ve bazofil değerleri seçilmiştir. Tanı için çeşitli makine öğrenme algoritmaları kullanılmıştır. Bu algoritmalar arasında karar ağacı, aşırı rastgele ağaçlar, k en yakın komşu, lojistik regresyon, naif bayes, rastgele orman, destek vektör makinesi ve sınıflandırma yöntemleri bulunmaktadır. Hastalığı en iyi tahmin eden algoritma olarak rastgele orman belirlenmiştir. Ayrıca, transaminaz değerinin 25'ten düşük olması, Covid-19 pozitifliğinin göstergesi olarak belirlenmiştir. Yapılan başarım değerlendirmesinde, ROC eğrisi altındaki alan 0,84 olarak bulunmuştur. Hata matrisinde 0,82 doğruluk ve 0,65 tutturma elde edilmiştir [18]. Bu sonuçlar, rastgele orman algoritmasının, belirlenen parametreler üzerinden yapılan teşhislerde etkili bir başarım sergilediğini göstermektedir.

[19] çalışması farklı olarak; Covid-19 teşhisinde rastgele orman algoritmasının yaygın olarak kullanıldığı bir alan olan makine öğrenmesi yöntemlerini incelemek yerine, gradyan artırma karar ağaçları üzerinde çalışmıştır. Bu yöntemin rastgele orman algoritmasından farkı, zayıf eğitim verilerini kullanırken bağımsız hareket etmek yerine sıralı olarak değerlendirme yapmasıdır. Kullanılan veri kümesi, İsrail Sağlık Bakanlığı'nın kayıtlarında bulunan Covid-19 sınaması yaptırmış kişilerin sınama sonuçlarından oluşmaktadır. Veri kümesinin %75'i eğitim, %25'i sınama verisi olarak ayrılmıştır. Veri kümesine ulamsal değişkenler, gözlenen hastalıklar ve hastalarla temas hikayeleri dahil edilmiştir. Verileri filtrelemede her tahmin sonucunu özellik değerinin katkısına bölerek modeller arasında ayırım yapmak için özellik alt uzayını kullanan SHAP (SHapley Additive exPlanations) modeli kullanılmıştır. 0,90 olan ROC eğrisi altındaki alan, değişkenler normallik sınaması ile filtreledikten sonra 0,95 olarak ölçülmüştür. Diğer başarım ölçüleri, bulma: 0,87 ve tutturma:

0,71 bulunmuştur. Özellik değişikliklerinin etkisini araştırmak için modelde bulma sabitlenmiştir. Bu durumda ROC eğrisi altındaki alan 0,66 ölçülmüştür.

Covid-19 tanılmasına odaklanan makine öğrenmesi çalışmalarında k-en yakın komşu, destek vektör makinesi, karar ağaçları, rastgele orman, doğrusal regresyon ve lojistik regresyon yöntemlerinin sıkça kullanıldığı gözlemlenmiştir.

Asıl veri kümesinden rastgele seçilen satır ve sütunlarla çok sayıda karar ağacı oluşturup bunların verdiği sonuçları bir arada değerlendiren bir yöntem olan rastgele orman algoritması diğerlerine göre üstün başarı göstermiştir [8,17,18,20,21].

Çizelge-1'de hastalığın tanılması için makine öğrenme yöntemleriyle yapılan çalışmaların karışıklık matrisinden hesaplanan başarı ölçütlerine göre karşılaştırılması gösterilmiştir. Verinin doğru ve yanlış sınıflandırılmış örnek sayısını içeren karışıklık matrisi (Toplam vaka sayısı= DP + DN + YP + YN) ele alınarak Çizelge 1, 3 ve 5' te çalışmalarda verilmeyen eksik değerler, verilen değerler kullanılarak hesaplanmış ve tamamlanmıştır (Çizelgelerde tamamlanan değerler kalın yazı tipi ile altı çizili olarak gösterilmiştir.). Farklı çalışmalarda ön plana çıkan başarım ölçütlerinin çeşitliliği, yöntemlerin karşılaştırılmasında potansiyel sorunları ortaya çıkarmaktadır. Verilerin işlenmiş olmasının tüm algoritmalarda başarıyı artırdığı gözlemlenmiştir. Bu nedenle, incelenen çalışmaların sadece filtrelenmiş ve dengelenmiş veriler kullanarak yapılan denemeleri dikkate alınmıştır.

İstatistik terimlerini hastalık saptama alanına özgü bir biçimde yorumlayacak olursak: Çizelgede yer alan ölçütlerden doğruluk, tüm tespitler içindeki doğru tespitlerin oranıdır (hasta veya sağlıklı). Bulma, tüm gerçek hastalar içindeki tespit edilebilmiş hastaların oranıdır. Kesinlik, hasta teşhisi konmuş bireyler arasındaki gerçek hastaların oranını göstermektedir. Tuturma ise gerçek sağlıklı bireyler arasındaki sağlıklı tescil edilmiş bireylerin oranı için kullanılan terimdir.

Algoritmaların ayırıcı gücünü alandan bağımsız olarak ölçmenin iyi bir yolu ROC eğrisi altında kalan alanı hesaplamaktır. ROC eğrisi (Receiver Operating Characteristic Curve) potansiyel olarak birbiriyle çelişen öncelikleri olan bulma ve tuturma arasındaki dengeyi, değişken bir eşik değeri boyunca modellemek amacıyla kullanılan bir fonksiyondur. Fonksiyon özgüllüğün tam tersi (tümleyen) olan yanlış pozitif oranı ile bulma eksenlerini içeren bir grafik olarak temsil edilir. Bu grafiğin eşik değerinin belirlenmesi konusunda önemli bir işlevi vardır, zira bulma ve tuturma kaygılarının dengelendiği optimum bir nokta tayin etme görevini görselleştirmektedir. Bunun dışında grafiğin altında kalan alan da AUC (Area Under the Curve) olarak bilinen başarım ölçütüdür. Çizelge 1'de bu değer ROC Eğrisi Altında Kalan Alan olarak gösterilmiştir ve teşhis koyan bir algoritmanın toplam ayırıcı gücünü (kaba kuvvetini) ölçmek konusunda kullanışlı bir değerdir. Eşik değeri belirlendiğinde ise bu gücün hangi yöne evrileceği belirlenmiş olmaktadır.

Bir algoritmanın toplam etkinliğini sayısal olarak modellemenin bir başka yolu, uygun bir  $\beta$  değeri için F- $\beta$

oranını hesaplamaktır. Yaygın olarak F puanı veya F1 puanı olarak bilinen ölçüt aslında mümkün olan  $F-\beta$  değerlerinden sadece bir tanesidir ve özel bir durumu ifade etmektedir. Bu ölçütün hesaplanmasında esas olan bulma ve kesinlik değerlerinin ağırlıklı harmonik ortalamasının bulunmasıdır. Beta değeri ise ağırlık çarpanıdır. Dolayısıyla F1 kesinlik ve bulma arasında tam bir denge kurarak ( $\beta=1$ ) her ikisine eşit önem atfetmektedir.

Kesinlik veya bulma farklı bir biçimde ağırlıklandırılmak istendiğinde farklı bir  $F-\beta$  ölçütü uygulanır. Kesinlik değeri, yanlış pozitiflerin ne kadar azaltılabileceğini göstermekte bulma ise yanlış negatiflerin ne kadar azaltılabildiğini ifade etmektedir. Yanlış pozitiflerin daha zararlı olduğu düşünüldüğünde  $\beta < 1$  olacak biçimde bir  $\beta$  parametresine ihtiyaç olacaktır. Tersine yanlış negatiflerin giderilmesinin daha öncelikli olduğunu düşünürsek  $\beta > 1$  olacak biçimde bir  $\beta$  belirlenmesi gerekecektir [22].

$\beta$  değerini doğru belirleyebilmek için, bir yanlış negatif vakasının, bir yanlış pozitif değerlendirmeye nazaran kaç kat daha kötü olduğunu subjektif ve kabaca da olsa saptamak ve bunu bir sayı olarak ifade etmek gerekmektedir. Hastalık teşhisi söz konusu olduğunda bu sayının genellikle 1 olmayacağı açıktır.

Hastalık tanılama ve tedavisi kendi önceliklerini dayatan alanlardır ve gereksiz uygulanan tedavilerin göze alınamayan ek riskler yaratabileceği hayali aykırı senaryoları bir kenara bırakırsak, yanlış negatif sınıflandırmaların, yanlış pozitiflere nazaran çok daha kötü bir ulama işaret ettiğini güvenli bir biçimde iddia edebiliriz. Sağlam sanılarak tedaviden mahrum bırakılmış bir hasta, hasta sanılarak yanlışlıkla tedavi edilmiş bir kişiye göre çok daha tehlikeli bir durumdadır. Hele Covid-19 gibi oldukça bulaşıcı hastalıklarda yanlış negatif olarak sınıflandırılan kişilerin hastalığı yaymaya devam edecekleri düşünülürse, durumun asimetrisi (ve dolayısıyla F1 ölçütünün neden iyi bir ölçüt olmadığı) daha iyi anlaşılacaktır.

Hastalık tanılmasında  $\beta$  oranını 1'den büyük seçmek gerekmektedir. Bu alanda bulma, kesinlik karşısında önceliklidir. Çalışmamızda Çizelge 1'de görüldüğü gibi F-2 ve F-10 ölçütleri dikkate alınmıştır (sırasıyla  $\beta=2$  ve  $\beta=10$ ).

İncelenen birçok çalışmada çok sayıda algoritmanın karşılaştırıldığı ve içlerinden birinin en başarılı yöntem olarak işaret edildiği görülmektedir. Çizelge-1'de yer alan değerler her çalışmadaki en başarılı bulunan algoritmaları yansıtmaktadır. Bu sıralamalarda birinci gelme sıklığını eğer bir ölçüt olarak alırsak bu konuda en başarılı olan yöntemin rastgele orman algoritması olduğu görülmektedir. Fakat tüm çalışmalar genelinde F-2 ve F-10 değerlerinin en iyi başarıyı gösteren yöntemlerin yapay sinir ağları ve aşırı gradyan artırma olduğunu görmekteyiz [28,29].

Sonuçta her iki bakış açısı da sadece birinci (veya ikinci) gelen algoritmaları dikkate aldığından algoritmaların tercih edilme sıklığından bağımsız bir biçimde değerlendirmek ve sağlıklı bir sonuca varmak zordur. Söz gelimi rastgele orman algoritmasının birçok karşılaştırmada birincilikle çıkması, kısmen en sık kullanılan algoritma olmasından kaynaklanabilir.

Bu etkiyi önlemek için tüm çalışmalarda yapılan denemelerden elde edilen sonuçlar, kullanılan algoritmalar üzerinden gruplandırılarak geometrik ortalamaları alınmıştır. Bu yöntem, çalışmalardaki tekrar edilemeyen aykırı değerlerin ve istatistiksel gürültünün etkisini en aza indirmek için tercih edilmiştir. Elde edilen sonuçlar, Çizelge-2'de görülmektedir. Tüm denemelerin geometrik ortalaması penceresinden bakıldığında, yapay sinir ağlarının en başarılı algoritma olduğu ikinciliğin ise derin öğrenme algoritmasına ait olduğu görülmektedir. Rastgele orman algoritmasının ise ilginç bir şekilde geride gözlemlenmektedir.

Çizelge-1 üzerinde yapılan değerlendirmede, F-2 ve F-10 değerlerinin en üst düzeyde, özellikle yapay sinir ağı ve aşırı gradyan artırma yöntemlerinde bulunduğu gözlemlenmektedir. Bu yöntemler, diğer başarımların değerlendirme metriklerine bakıldığında bulma değerlerinin diğer yöntemlere kıyasla daha yüksek olduğunu ortaya koymaktadır. Yüksek bulma, yanlış negatiflerin oranının ne kadar düşük olduğunu ifade eder.

Geometrik ortalamalara odaklanıldığında, yapay sinir ağının en üstün başarımlar sergileyen algoritma olduğu görülmektedir. Aşırı gradyan artırma yöntemi ise bu başarı seviyesinden biraz geride kalmaktadır. Tüm değerlendirmelerin göz önüne alındığı bir açıdan, 0,96 doğruluk ve 0,98 kesinlik değerleriyle yapay sinir ağının, hastalığın tanılmasında en etkili algoritma olduğu sonucuna varılmaktadır.

### 3. Hastalık Şiddetinin Belirlenmesi

Literatürdeki çalışmalarda ele alınan diğer bir konu hastalığın şiddetinin belirlenmesi olmuştur [33,34]. Hastalığın şiddetinin sınıflandırılması, hastaneye yatış gerekliliğini belirlemek açısından teşhis kadar kritiktir. Hangi hastaların hastaneye yatacağını doğru bir şekilde belirlemek, hastane kaynaklarının (hastanenin personeli, donanımı, hasta yatak kapasitesi, hastalık sürecinde ihtiyaç duyulacak malzemeler ve yoğun bakım ünitelerinin hazırlanması) etkili bir şekilde kullanılması ve gerekli hazırlıkların yapılması için hayati önem taşımaktadır. Bu nedenle hastalık şiddeti ile ilgili çalışmalar, hastaneye yatmayı gerektiren ve gerektirmeyen vakaların birbirinden ayrılması amacına yöneliktir.

Bazı çalışmalar hastalığın şiddet sınıflandırmasında hangi değişkenlerin (özniteliklerin) belirleyici olduğunu ve bu değişkenlerin önem derecesini belirlemeye odaklanmaktadır [5,6]. Şiddet sınıflandırması yapılırken genel olarak kan değerleri, kategorik değişkenler (cinsiyet, daha önce hastalık geçirip geçirmediği vb.), yaş ve belirtilerin görülme durumu gibi faktörlerin önemli olduğu görülmüştür. Ancak, seçilen öznitelikler konusunda bir standardın bulunmaması, algoritmalar arasında sağlıklı bir karşılaştırma yapmayı zorlaştırmaktadır.

Çizelge 1: Covid-19 tanısı için makine öğrenmesi algoritmalarıyla yapılan çalışmaların karşılaştırılması

Hastalığın Teşhisinde Makine Öğrenme Yöntemlerinin Başarım Değerlendirmesi										
Kaynaklar	İstatiksel Analiz	Önerilen Model	ROC Eğrisi Altında Kalan Alan	Başarım Değerlendirme Ölçütleri						
				Doğruluk	Bulma	Kesinlik	Tutturma	Veri Sayısı	F-2 Oranı	F-10 Oranı
[17]	Student t Mann-Whitney U $\chi^2$	Rastgele Orman	0,79	0,92	0,88	<u>0,87</u>	0,93	162	<u>0,88</u>	<u>0,88</u>
[18]	Zincirli Denklemle Çok Değişkenli Tahmin (MICE)	Rastgele Orman	0,84	0,82	0,92	<u>0,99</u>	0,65	279	<u>0,93</u>	<u>0,92</u>
[20]	Smote	Rastgele Orman	0,99	0,95	0,95	<u>0,97</u>	0,96	287	<u>0,95</u>	<u>0,95</u>
[21]	Smote	Rastgele Orman	0,84	0,95	0,93	0,93	<u>0,93</u>	5644	<u>0,93</u>	<u>0,93</u>
[23]	Karar Ağacı Açıklayıcı (DTX)	Rastgele Orman	0,86	0,88	0,66	<u>0,89</u>	0,91	608	<u>0,70</u>	<u>0,66</u>
[24]	Smote	Rastgele Orman	0,99	0,95	<u>0,95</u>	<u>0,97</u>	0,96	169	<u>0,95</u>	<u>0,95</u>
[25]	Smote Shapley Katkı Açıklamaları (SHAP)	Rastgele Orman	0,92	0,92	<u>0,71</u>	<u>0,90</u>	0,96	5644	<u>0,74</u>	<u>0,71</u>
[26]	Çapraz Doğrulama	Aşırı Gradyan Artırma	0,97	<u>0,97</u>	0,81	<u>0,53</u>	<u>0,97</u>	5333	<u>0,73</u>	<u>0,81</u>
[27]	Normalizasyon Algoritması	Gradyan Artırma Ağaçları	0,85	0,79	0,76	<u>0,74</u>	0,80	3356	<u>0,76</u>	<u>0,76</u>
[28]	Smote	Aşırı Gradyan Artırma	0,99	0,99	0,99	<u>0,99</u>	0,99	5644	<u>0,99</u>	<u>0,99</u>
[19]	Shapley Katkı Açıklamaları (SHAP)	Güçlendirmeye Dayalı Algoritmalar (LightGBM)	0,90	0,95	0,87	<u>0,79</u>	0,72	4769	<u>0,85</u>	<u>0,87</u>
[29]	Korelasyon katsayısı	Yapay Sinir Ağı	0,95	<u>0,96</u>	1	<u>0,98</u>	0,86	151	<u>1</u>	<u>1</u>
[30]	Özellik Seçimi	Lojistik Regresyon	0,97	0,90	0,87	<u>0,86</u>	0,95	912	<u>0,87</u>	<u>0,87</u>
[8]	Normalizasyon	Aşırı Gradyan Artırma	0,67	0,87	0,88	0,89	<u>0,88</u>	600	<u>0,88</u>	<u>0,88</u>
[31]	Çapraz doğrulama	Lojistik Regresyon	0,78	0,84	0,81	0,80	<u>0,82</u>	114	<u>0,81</u>	<u>0,81</u>
[32]	Çapraz doğrulama	Derin Öğrenme	0,90	0,92	0,94	<u>0,92</u>	<u>0,98</u>	1087	<u>0,94</u>	<u>0,94</u>

Çizelge 2: Tanılama makine öğrenme modellerinin başarım değerlerinin geometrik ortalamaları

Tanılama Algoritmalarının Ortalama Başarımları							
Model	ROC Eğrisi Altında Kalan Alan	Ortalama Değerler					
		Doğruluk	Bulma	Kesinlik	Tutturma	F-2 Oranı	F-10 Oranı
Yapay Sinir Ağı	0,95	0,96	1	0,98	0,86	1	0,99
Derin Öğrenme	0,90	0,92	0,94	0,92	0,98	0,94	0,94
Güçlendirmeye Dayalı Algoritmalar (LightGBM)	0,90	0,95	0,87	0,79	0,72	0,85	0,87
Aşırı Gradyan Artırma	0,86	0,90	0,86	0,70	0,91	0,83	0,86
Rastgele Orman	0,88	0,91	0,85	0,93	0,89	0,86	0,85
Lojistik Regresyon	0,78	0,87	0,84	0,83	0,88	0,84	0,84





geometrik ortalama alınarak birleştirilmiştir. Sonuçlar Çizelge-4'te görülmektedir. En başarılı yöntem rastgele orman algoritması görülmektedir. Bu çalışmada 5 farklı istatistiksel analiz ile bu başarı sağlanmıştır. Makine öğrenme yöntemleri farklı istatistiksel analiz yöntemleri ile ayrı ayrı değerlendirilmiş; ek makine öğrenme yöntemleri ile %100 başarıya ulaşmıştır. Fakat değerlendirme 337 hasta verisi üzerinde gerçekleştirilmiştir. Modelin başarısı farklı veri setleri üzerinde değerlendirilmelidir. 337 hastanın cinsiyet dağılımı da orantısızdır ve veri kümesinin değişiklikler yapılarak yeniden değerlendirilmesine ihtiyaç vardır.

Lojistik regresyon, destek vektör makinesi, rastgele orman, k en yakın komşu, güçlendirmeye dayalı algoritma gibi özellikleri keskin bir şekilde ayıran algoritmalarla yapılan çalışmaların hastalığın şiddetini saptama konusunda daha

başarılı sonuçlara ulaşmaları beklenen bir sonuçtur. Bununla birlikte farklı öznelikler kullanan çalışmalar üzerinden algoritmalar ile ilgili genellemeler yapmak doğru olmaz.

Öznelik seçimi yapılmadığında ise algoritmaların başarısı düşmektedir. İki sınıfın daha net ayırım yapmasını sağlayan parametreler ayarlanması ve hastalığın şiddetinde önemli değişkenlerin belirlenmesi ile model geliştirilir. Özellik seçiminin ilk kuralı ise dengeli bir veri kümesidir [30, 41].

Hastalığın şiddetini belirleme odaklı yapılan çalışmaların karşılaştırılmasına yönelik olarak, Çizelge-3'ün incelenmesi sonucunda en üstün başarıyı sergileyen yöntemlerin rastgele orman, yapay sinir ağı ve aşırı gradyan artırma olduğu göze çarpmaktadır. Ancak, rastgele orman algoritmasının tercih edilmesinde çok sayıda filtreleme işlemi uygulanmasının sonuçları üzerinde etkili olduğu düşünülmektedir.

Çizelge 3: Hastalık şiddetinin belirlenmesinde başarımların ölçütleri.

Hastalığın Şiddet Sınıflandırmasında Başarımların Karşılaştırması										
Kaynaklar	İstatistiksel Analiz	Önerilen Model	ROC Eğrisi Altında Kalan Alan	Başarımların Değerlendirme Ölçütleri						
				Doğruluk	Bulma	Keskinlik	Tutturma	Veri Sayısı	F-2 Oranı	F-10 Oranı
[9]	Student t	Destek Vektör Makinesi	0,91	<b>0,81</b>	0,83	<b>0,83</b>	0,79	137	<b>0,83</b>	<b>0,83</b>
[40]	Normalizasyon	Destek Vektör Makinesi	0,85	<b>0,77</b>	<b>0,68</b>	<b>0,78</b>	0,85	235	<b>0,70</b>	<b>0,68</b>
[41]	Karşılıklı bilgi Ki-kare sınamai	Destek Vektör Makinesi	0,96	0,92	0,92	<b>0,71</b>	0,91	10237	<b>0,87</b>	<b>0,92</b>
[33]	Çapraz Doğrulama	Aşırı Gradyan Artırma	0,91	<b>0,92</b>	0,96	<b>0,95</b>	0,42	7842	<b>0,96</b>	<b>0,96</b>
[42]	Çapraz Doğrulama	Aşırı Gradyan Artırma	0,90	<b>0,89</b>	0,56	<b>0,33</b>	0,92	5148	<b>0,49</b>	<b>0,56</b>
[43]	Çapraz Doğrulama Shapley Katkı Açıklamaları (SHAP)	Aşırı Gradyan Artırma	0,92	0,85	<b>0,85</b>	<b>0,65</b>	0,85	3927	<b>0,80</b>	<b>0,85</b>
[9]	Student t	Aşırı Gradyan Artırma	0,91	0,91	0,96	0,97	<b>0,86</b>	137	<b>0,96</b>	<b>0,96</b>
[34]	Smote	Aşırı Rastgele Ağaçlar	0,96	0,96	0,95	<b>0,96</b>	0,95	55676	<b>0,95</b>	<b>0,95</b>
[38]	R normalleştirme	Naif Bayes	0,90	0,95	0,86	<b>0,99</b>	0,75	422	<b>0,88</b>	<b>0,86</b>
[39]	Chi 2 Karşılıklı bilgi F sınamai Anova PCA UMAP	Rastgele Orman	1	1	<b>1</b>	<b>1</b>	1	337	<b>1</b>	<b>1</b>
[44]	Kaplan-Meier Yöntemi	Lojistik Regresyon	0,89	<b>0,86</b>	<b>0,82</b>	<b>0,47</b>	0,81	2307	<b>0,71</b>	<b>0,81</b>
[45]	Eş değişken seçimi Özellik çıkarımı	Lojistik Regresyon	0,881	<b>0,82</b>	0,84	<b>0,80</b>	0,79	183	<b>0,83</b>	<b>0,84</b>
[46]	Özellik seçimi	Yapay Sinir Ağı	0,98	<b>0,96</b>	<b>0,96</b>	<b>0,91</b>	<b>0,96</b>	166	<b>0,95</b>	<b>0,96</b>

Çizelge 4: Hastalık şiddetinin belirlenmesinde başarımların geometrik ortalamaları.

Şiddet Sınıflandırması Algoritmalarının Ortalama Başarımları							
Model	ROC Eğrisi Altında Kalan Alan	Başarımların Değerlendirme Ölçüleri					
		Doğruluk	Bulma	Kesinlik	Tutturma	F-2 Oranı	F-10 Oranı
Rastgele Orman	1	1	1	1	1	1	1
Yapay Sinir Ağı	0,98	0,96	0,96	0,91	0,96	0,95	0,96
Aşırı Rastgele Ağaçlar	0,96	0,96	0,95	0,96	0,95	0,95	0,95
Naif Bayes	0,90	0,95	0,86	0,99	0,75	0,88	0,86
Lojistik Regresyon	0,89	0,84	0,83	0,61	0,80	0,77	0,82
Aşırı Gradyan Artırma	0,91	0,89	0,81	0,67	0,73	0,78	0,81
Destek Vektör Makinesi	0,91	0,83	0,80	0,77	0,85	0,80	0,80

#### 4. Hastalığın Bulaşma Hızının Belirlenmesi

Literatürdeki çalışmalarda makine öğrenmesi kullanılarak hastalığın bulaşma hızını hesaplamak amaçlanmıştır [34,47]. Hastalığın bulaşma hızını etkileyen faktörler arasında hastanın etkileşimde bulunduğu kişi sayısı, hastalığın teşhisi için geçen süre ve hastalığın şiddeti gibi birçok unsurdan bahsedilebilir. Bu bağlamda, farklı çalışmaların bu amaç doğrultusunda nasıl bir yol izlediği detaylı bir şekilde incelenmiştir [48, 49].

[50] çalışması, bulaşma hızı hesaplanmasında sadece kan değerlerinden oluşan veri setlerinin yetersiz kaldığı saptanmıştır. Bulaşma hızını belirlemek için iki kriter belirlemişlerdir. Bunlar, hastalığa sahip kişilerin, ilk olarak Wuhan'da seyahat etme; ikinci olarak, diğer şehirlerde iletişim kurabilecekleri tanıdıkları buluşma olasılıklarıdır. Çalışmada bulaşma hızının belirlenmesinde bu verilerin yetersiz olduğu sonucuna ulaşılmıştır.

[51] çalışması ise Hong Kong Üniversitesi-Shenzhen Hastanesine ateş, solunum semptomları ve akciğer grafilerinde pulmoner infiltrat bulguları/şikayetleri ile başvuran iki hastada hastalığın bulaşma oranını belirlemek istemiştir. Bu ailenin diğer beş üyesi de 11 Ocak 2020 ile 15 Ocak 2020 tarihleri arasında incelenmiştir. Değerlendirilen özellikler, semptomların başlangıcı ile Wuhan'a varış arasındaki süre, hastaneye yatış ile semptomların başlangıcı arasındaki süre ve semptom ve bulgular (ateş, öksürük). Hastalarda incelenen değişkenler hastalık belirtileri ve kan değerleridir.

Hastalara ishal enfeksiyon sınaması, gerçek zamanlı RT-PCR sınaması yapılmış; solunumları, biyolojik istatistiksel çalışmalarla incelenmiştir. Hasta yakınlarının hastalığı nerden kapıldığı ve hasta kişinin kaç kişiye hastalığı bulaştırdığıyla ilgili bir veri elde edilememiştir. Bu yüzden bulaşıcılık hızı belirlenememiştir [51].

Hastalığın bulaşma hızını etkileyebilecek faktörler arasında hastalık şiddeti ve hastalığın saptanmasında belirleyici özelliklerin etkisi de bulunmaktadır. Bu değişkenlerin önem

derecesinin belirlenmesi, hastalık tanısının ve şiddetinin belirlenmesinde daha hızlı ve etkili bir yol açacaktır [48, 49]. Bu bağlamda, değişkenlerin önem derecelerini belirlemeye yönelik yapılan çalışmalar Çizelge-5'te karşılaştırılmıştır.

Kullanılan veri kümesinde önemli değişkenlerin belirlenebilmesi için modellerin başarımlarına bakıldığında lojistik regresyon ve yapay sinir ağı ROC eğrisi altında kalan alan değeri ile en başarılı algoritma görülmüştür. Geometrik ortalaması alınması da sonucu değiştirmemiştir (Çizelge-6). Yapay sinir ağı hastalığın teşhisi, şiddeti ve hastalıkta önemli değişkenlerin belirlenmesinde en başarılı yöntem olmuştur.

#### 5. İncelenen Çalışmaların Değerlendirilmesi

İncelenen çalışmalarda en yaygın kullanılan makine öğrenmesi yöntemlerinin sırasıyla rastgele orman, aşırı gradyan artırma makinesi, regresyon karar ağacı ve yapay sinir ağları, olduğu görülmektedir.

Hastalığın teşhisine yönelik yapılan çalışmalarda bazı başarımların verilmemesi gözlemlenmiştir. İncelenen çalışmalar arasında doğruluk, bulma, kesinlik ve tutturma olarak sayabileceğimiz dört temel başarımların tamamını veren bir çalışmaya rastlanmamıştır. Seçilen ölçütler arasında bir benzerlik de gözlemlenmemektedir. Örneğin [24] ve [25]'de tıbbi teşhis alanındaki en önemli ölçütlerden biri olan bulma değeri verilmemiştir. Bu ve bu durumda olan diğer çalışmalarda eksik sonuçlar elde edilen veriler kullanılarak hesaplanmış ve böylece tamamlanmış bir tablo üzerinde tüm çalışmaların başarımlarının gösterilmesi sağlanmıştır.

Çizelge 5: İdeal bir modelde önemli değişkenlerin belirlenmesi için oluşturulan modellerin kıyaslanması.

Önemli Değişkenlerin Belirlenmesinde Başarım Karşılaştırması										
Kaynaklar	İstatistiksel Analiz	Önerilen Model	ROC Eğrisi Altında Kalan Alan	Başarım Değerlendirme Ölçüleri						
				Doğruluk	Bulma	Keskinlik	Tutturma	Veri Sayısı	F-2 Oranı	F-10 Oranı
[49]	Smote	Yapay Sinir Ağı	1	0,95	0,95	<u>0,95</u>	0,95	580	<u>0,95</u>	<u>0,95</u>
[48]	Minimum fazlalık (mRMR)	Lojistik Regresyon	1	0,95	0,98	0,95	<u>0,91</u>	110	<u>0,97</u>	<u>0,98</u>
[47]	Student-t Ki-kare sınama	Lojistik Regresyon	0,93	0,95	<u>0,95</u>	<u>0,82</u>	<u>0,95</u>	536	<u>0,92</u>	<u>0,95</u>
[34]	Ki-kare sınama Çapraz doğrulama	Karar Ağacı	0,92	0,89	0,84	0,94	<u>0,89</u>	55676	<u>0,86</u>	<u>0,84</u>

Çizelge 6: Önemli değişkenlerin belirlenmesi için oluşturulan modellerin geometrik ortalaması hesaplamıştır.

Değişken Belirleme Yöntemlerinin Ortalama Başarımları							
Model	ROC Eğrisi Altında Kalan Alan	Başarım Değerlendirme Ölçüleri					
		Doğruluk	Bulma	Keskinlik	Tutturma	F-2 Oranı	F-10 Oranı
Lojistik Regresyon	0,96	0,95	0,96	0,88	0,93	0,94	0,96
Yapay Sinir Ağı	1	0,95	0,95	0,95	0,95	0,95	0,95
Karar Ağacı	0,92	0,89	0,84	0,94	0,89	0,86	0,84

Yapay zekânın tıbbi teşhis amaçlı kullanımında bulma ölçütü özellikle önemli olduğu görülmektedir. Duyarlılığın ağırlığının istenildiği gibi artırılabilmesi F-β değerlerinin hesaplanması yöntemlerinin başarısını ölçmek için tercih edilmesi gereken gerçekçi bir kıstas sunmaktadır. Bir algoritmanın genel olarak başarılı gibi görünmesi, özellikle verilerin dengesiz olduğu tıp gibi alanlarda yanıltıcı olabilir. Beta değerinin 1'den büyük olduğu bir F-β ölçütü daha uygundur, çünkü yanlış negatif vakalar genellikle yanlış pozitif vakalara göre daha tehlikelidir. Bu iki hususa simetrik yaklaşan ölçütler, yanıltıcı olabilir.

Bu çalışmada F-2 ve F-10 değerleri hesaplanmıştır. Teşhis çalışmaları için yapılan karşılaştırmada ilk bakışta fark edilmesi zor olan bir başarı sıralaması ortaya çıkmış ve yapay sinir ağları diğer yöntemlerin önüne geçmiştir. ROC eğrisi altında kalan alan açısından yüksek puan aldığı halde F-2 skorundan düşük not alan çalışmaların varlığı, eşik değeri belirleme işlemlerinde hata yapılmış olabileceğini düşündürmektedir.

Verilerin ön işleme tabi tutulması (istatistiksel analiz/filtreleme işlemleri) tüm çalışmalarda ROC eğrisi altında kalan alan değerini kayda değer biçimde artırmıştır. Bu durum filtreleme ve analiz işlemlerinin yararını göstermektedir [51].

Tüm değerlendirmelerde veri kümesinin büyüklüğü ve içeriği

çok büyük değişkenlik göstermektedir. Çok sayıda değişkene rağmen algoritmalara yönelik genel bir yargıya varabilmek amacıyla aynı algoritmayı kullanan tüm denemelerin sonuçlarının geometrik ortalamasını içeren ayrı bir karşılaştırma yapılmıştır. Bu karşılaştırmada yapay sinir ağları en ön sıradaki yerini korumuş, rastgele orman algoritması ise biraz daha gerilere düşmüştür (Çizelge 2). Genelleme yapmanın çok daha zor olduğu hastalık şiddetinin ve bulaşıcılığının belirlenmesi alanlarında da aynı karşılaştırma yöntemi tekrar edilmiştir (Çizelge 4-6).

Makine öğrenmesi yöntemleri ile hastalığın şiddetinin belirlenmesi çalışmalarının değerlendirilmede iki temel sorun bulunmaktadır". Bunlardan birincisi çalışmalarda seçilen (veya elde edilebilen) öznelitliklerin birbirinden farklı olması; ikincisi ise, yapılan tahminlerin doğruluğunu sınama etmek için kullanılacak güvenilir bir çıktı verisinin olmamasıdır. Sonuçların doğruluğu, ölen veya yoğun bakıma yatan hastalar üzerinden değerlendirilmiştir. Gerçekte kimin ne kadar risk yaşadığını doğrudan gösteren bir ölçüm mümkün değildir.

Girdi verisi olarak ise beklenen ölüm oranı ölçütü [6]'da lenfopeni, bakteriyel koenfeksiyon, sigara içme, hipertansiyon ve yaş değişkenlerini kullanarak yapılan çalışmada başarıyla kullanılmıştır. İncelenen hastalardan ölen ve yoğun bakıma yatan hastaların değerlerindeki aralık

sapması ve ortak özellikler hastalığın şiddeti ile ilişkili bulunmuştur. Yaş, büyüme hormonu (GHS), CD3, çalışma durumu, diyabet, kanser varlığı, protein seviyesi, sitokin, oksijen seviyesi ve lenfosit sayısı hastalık şiddetinde belirleyici rol oynamıştır [15,30,41,44,45].

Metin etiketleme için kullanılan naif bayes ve özelliklerin birbirinden daha keskin ayrıldığı algoritmalar, yani lojistik regresyon, destek vektör makinesi, rastgele orman, k en yakın komşu ve yetkilendirme tabanlı algoritma, hastalığın şiddetini belirlemede oldukça başarılı bulunmuştur. Ölçütlere bakıldığında ise rastgele orman yönteminin 337 veri içeren bir çalışmada istatistiksel analiz işlemleri ile birlikte %100 doğruluğa ulaşması dikkat çekicidir [39]. Bunun yanında yapay sinir ağı yaklaşımı burada da başarısını sürdürmektedir [46].

Hastalığın bulaşma hızının belirlenmesi sürecinde, sinama öncesi ve sonrasında hastanın insanlarla olan etkileşiminin veri kümesinde temsil edilmesi gerektiği değerlendirilmektedir. Bu insani veriler olmadan, bulaşma oranının başarısı, ölçülemeyecektir [51].

Tüm deneylere ilişkin başarımların geometrik ortalamalarına bakıldığında hastalığın teşhisinde, şiddetinin belirlenmesinde ve değişkenlerin önem derecesinde yapay sinir ağı yönteminin tutarlı bir biçimde başarılı olduğu görülmektedir. Hastalık şiddetinin belirlenmesi alanında rastgele orman algoritmasının önemli bir çıkış yaptığı söylenebilir. Bununla birlikte bu sonuç tek bir çalışmanın ölçütlerinden kaynaklanmaktadır.

## 6. Sonuç

Bu çalışmada literatürde yer alan makine öğrenmesi algoritmaları ile yapılan Covid-19 çalışmaları hem sonuçları açısından hem de değerlendirme yöntemleri açısından sistematik bir biçimde incelenmiştir. İnceleme Covid-19'un tanısı, hastalığın şiddeti ve bulaşıcılık hızının hesaplanması konularını kapsamaktadır.

Çalışmalar veri büyüklüğü, öznitelik seçimi ve hasta popülasyonu açısından büyük farklılıklar göstermektedir. Ayrıca karşılaştırmaların dayandırıldığı başarı ölçütleri ile ilgili standartlar oluşturulmamıştır. Değişkenlerin fazlalığı, algoritmalar üzerinde güvenilir bir karşılaştırma yapmak konusunda bir engeldir.

Bu çalışmada karşılaştırmaları anlamlı kılmak amacıyla çeşitli yöntemler kullanılmıştır. Eksik olan başarımlar ölçütleri hesaplanmış ve alana özgü nedenlerle bu ölçütlerin tümünden daha önemli olan F-β değerleri (F-2 ve F-10 olmak üzere) hesaplanmıştır. Ayrıca aynı algoritmayı kullanan denemeler gruplanarak başarımlar ölçütlerinin geometrik ortalaması alınmıştır. Tüm bu hesaplamalar sonucunda yapay sinir ağı tabanlı yaklaşımların (derin öğrenme dahil edilebilir) hastalığın kan tahlilinden teşhisi konusunda en başarılı yöntemler olduğu görülmektedir.

Covid-19 şiddetinin belirlenmesinde farklı özniteliklerin kullanılması nedeniyle yöntemler arası karşılaştırma sonuçları yeterince güvenilir olmasa da makine öğrenmesi yöntemlerinin genel olarak hastalık şiddetinin belirlenmesinde de teşhis alanında olduğu kadar etkili

oldukları gözlemlenmektedir.

Hastalığın bulaşma hızının hesaplanmasında ise elimizde güvenilir bir yöntem yoktur. Eldeki yöntemlerle bazı etkili değişkenler belirlenebilmektedir; fakat bulaşma hızının tespiti için hem bulaşma olaylarının detaylı bir dökümü hem de fiziksel temas sıklığının temsil edilmesine ilişkin insan ilişkilerinin yüksek sadakatli bir modeline ihtiyaç vardır. Bu derece detaylı verilerin elde edilmesi günümüzdeki sosyal, kültürel ve yasal çerçevelerde çok zordur.

Çalışmanın odak noktası, başarımların değerlerinin analizi üzerinden salgın hastalıklarda hangi metriklerin önemli olduğunu belirlemek ve Covid-19 gibi günümüzde mutasyona uğrayarak ortaya çıkabilecek yeni hastalıkların değerlendirmesini sadece başarımlar üzerinden değil, aynı zamanda yanlış kararların performansa öznel bir etki yapmadığını vurgulamaktadır. Bu bağlamda, makine öğrenmesi yöntemleriyle incelenen bir hastalığın değerlendirilmesinde, farklı bakış açılarının da göz ardı edilmemesi önemlidir. Yani, başarımın ötesinde, kararların objektif bir şekilde değerlendirilmesi gerekliliği vurgulanarak, salgın hastalıkların etkili bir şekilde analiz edilebilmesi için çeşitli perspektiflere odaklanılmalıdır.

Sonuç olarak eksiksiz ve dengeli bir veri kümesinin (eğitim ve sinama kümesi) en doğru sonuçları vereceği araştırmalarla gösterilmiştir. Ancak gerçek hayat veri kümelerinin dengesiz olması, daha iddialı bir model elde etmek için daha fazla sayıda örneklem gerektirmektedir. Modellerin başarımını tam bir doğrulukla değerlendirebilmek için aynı veri kümesi üzerinde yöntemlerin tamamını denemek ve daha sonra bu deneyi farklı veri setleri ile tekrarlamak gerekmektedir.

Tüm eksikliklere ve belirsizliklere rağmen, makine öğrenmesi yöntemleri hastalığın teşhisi ve takibi konusunda etkili birer yardımcıdır. Bu teknolojileri kullanan yazılımlar Covid-19 ve benzer salgınlarda risk ve maliyetlerin azaltılması ve kaynakların daha etkili yönetilmesi gibi yararlarıyla hekimlerin güvenilir destekçileri olacaktır.

## Kaynakça

- [1] Lewnard JA, Lo NC. The scientific and ethical basis for social-distancing interventions against COVID-19. *Lancet Infect Dis.* 2020 Jun;20(6):631-633.
- [2] Koo JR, Cook AR, Park M, Sun Y, Sun H, Lim JT, Tam C, Dickens BL. Interventions to mitigate the early spread of SARS-CoV-2 in Singapore: a modeling study. *Lancet Infect Dis.* 3-2020 Jun;20(6):678-688.
- [3] Pandemi ilan edilmesi-Dünya Sağlık Örgütü. DSÖ Genel Direktörü'nün 11 Şubat 2020 tarihinde 2019-nCoV medya brifinginde yaptığı açıklamalar. <https://www.who.int/dg/speeches/detail/who-director-general-s-remarks-at-the-media-briefing-on-2019-ncov-on-11-february-2020> (2020).
- [4] Rasheed J, Jamil A, Hameed AA, Al-Turjman F, Rasheed A. COVID-19 in the Age of Artificial Intelligence: A Comprehensive Review. *Interdiscip Sci Comput Life Sci.* Published online April 22, 2021.
- [5] Guan WJ, Ni ZY, Hu Y, Liang WH, Ou CQ, He JX, Liu L, Shan H, Lei CL, Hui DSC, Du B, Li LJ, Zeng G, Yuen KY, Chen RC, Tang CL, Wang T, Chen PY, Xiang J, Li SY, Wang JL, Liang ZJ, Peng YX, Wei L, Liu Y, Hu YH, Peng P, Wang JM,

- Liu JY, Chen Z, Li G, Zheng ZJ, Qiu SQ, Luo J, Ye CJ, Zhu SY, Zhong NS, China Medical Treatment Expert Group for Covid-19. Clinical Characteristics of Coronavirus Disease 2019 in China. *N Engl J Med*. 2020 Apr 30;382(18):1708-1720.
- [6] Chen N, Zhou M, Dong X, Qu J, Gong F, Han Y, Qiu Y, Wang J, Liu Y, Wei Y, Xia J, Yu T, Zhang X, Zhang L. Epidemiological and clinical characteristics of 99 cases of 2019 novel coronavirus pneumonia in Wuhan, China: a descriptive study. *Lancet*. 2020 Feb 15;395(10223):507-513.
- [7] Rodriguez-Morales A, Cardona-Ospina J, Gutierrez-Ocampo E, Villamizar-Pe~na R, Holguin-Rivera Y, Escalera-Antezana J, Alvarado-Arnez L, Bonilla-Aldana D, Franco-Paredes C, Henao-Martinez A. Clinical, laboratory and imaging features of COVID-19: a systematic review and meta-analysis. *Trav Med Infect Dis* 2020;34: 101623.
- [8] Demirarslan M., Suner A. Rutin kan sınımalarıyla Covid-19 tanı tahmininde makine öğrenmesi yöntemleriyle mobil uygulama geliştirilmesi. *Ege Tıp Dergisi*. 2021, 60(4): 384-393.
- [9] Yao H, Zhang N, Zhang R, Duan M, Xie T, Pan J, Peng E, Huang J, Zhang Y, Xu X, Xu H, Zhou F, Wang G. Severity Detection for the Coronavirus Disease 2019 (COVID-19) Patients Using a Machine Learning Model Based on the Blood and Urine Sinamas. *Front Cell Dev Biol*. 2020 Jul 31;8:683.
- [10] Booth, A.L, Abels, E, McCaffrey, P, Development of a prognostic model for mortality in COVID-19 infection using machine learning. *Mod. Pathol*. 2021, 34, 522–531.
- [11] Özen N. S., Saraç S. and Koyuncu M., "COVID-19 Vakalarının Makine Öğrenmesi Algoritmaları ile Tahmini: Amerika Birleşik Devletleri Örneği", *Avrupa Bilim ve Teknoloji Dergisi*, no. 22, pp. 134-139, Jan. 2021.
- [12] WHO Gender and COVID-19, World Health Organization: Geneva, Switzerland, 2020.
- [13] Malom, Z., Rahman, M.M.S., Nasrin, S., Taha, T.M., Asari, V.K. COVID-MTNet: COVID-19 detection with multi-task deep learning approaches. *arXiv* 2020, arXiv:2004.03747.
- [14] Michael J. Loeffelholz & Yi-Wei Tang (2020) Laboratory diagnosis of emerging human coronavirus infections – the state of the art, *Emerging Microbes & Infections*, 9:1, 747-756.
- [15] Zhu N, Zhang D, Wang W, Li X, Yang B, Song J, Zhao X, Huang B, Shi W, Lu R, Niu P, Zhan F, Ma X, Wang D, Xu W, Wu G, Gao GF, Tan W, China Novel Coronavirus Investigating and Research Team. A Novel Coronavirus from Patients with Pneumonia in China, 2019. *N Engl J Med*. 2020 Feb 20;382(8):727-733.
- [16] Coronaviridae Study Group of the International Committee on Taxonomy of Viruses. The species Severe acute respiratory syndrome-related coronavirus: classifying 2019-nCoV and naming it SARS-CoV-2. *Nat Microbiol* 5, 536–544 (2020).
- [17] Assaf D, Gutman Y, Neuman Y, Segal G, Amit S, Gefen-Halevi S, Shilo N, Epstein A, Mor-Cohen R, Biber A, Rahav G, Levy I, Tirosch A. Utilization of machine-learning models to accurately predict the risk for critical COVID-19. *Intern Emerg Med*. 2020 Nov;15(8):1435-1443.
- [18] Brinati D, Campaigner A, Ferrari D, Locatelli M, Banfi G, Cabitza F. Detection of COVID-19 Infection from Routine Blood Exams with Machine Learning: A Feasibility Study. *J Med Syst*. 2020 Jul 1;44(8):135.
- [19] Zoabi Y, Deri-Rozov S, Shomron N. Machine learning-based prediction of COVID-19 diagnosis based on symptoms. *NPJ Digit Med*. 2021 Jan 4;4(1):3.
- [20] Sumayh S. Aljameel, Irfan Ullah Khan, Nida Aslam, Malak Aljabri, Eman S. Alsulmi, "Machine Learning-Based Model to Predict the Disease Severity and Outcome in COVID-19 Patients", *Scientific Programming*, vol. 2021, Article ID 5587188, 10 pages, 2021.
- [21] Freitas Barbosa, V.A., Gomes, J.C., de Santana, M.A. et al. Heg.IA: an intelligent system to support diagnosis of Covid-19 based on blood sinamas. *Res. Biomed. Eng*. 38, 99–116 (2022).
- [22] Scikit-Learn, F-Bta Skoru, Erişim adresi: [https://scikit-learn.org/stable/modules/generated/sklearn.metrics.fbeta\\_score.html](https://scikit-learn.org/stable/modules/generated/sklearn.metrics.fbeta_score.html), Erişim Tarihi:02.02.2023.
- [23] Alves MA, Castro GZ, Oliveira BAS, Ferreira LA, Ramirez JA, Silva R, Guimarães FG. Explaining machine learning based diagnosis of COVID-19 from routine blood sinamas with decision trees and criteria graphs. *Comput Biol Med*. 2021 May;132:104335.
- [24] Jiangpeng Wu, Pengyi Zhang, Liting Zhang, Wenbo Meng, Junfeng Li, Chongxiang Tong, Yonghong Li, Jing Cai, Zengwei Yang, Jinhong Zhu, Meie Zhao, Huirong Huang, Xiaodong Xie, Shuyan Li, Rapid and accurate identification of COVID-19 infection through machine learning based on clinical available blood sinama results *medRxiv* 2020.04.02.20051136.
- [25] Chadaga K, Chakraborty C, Prabhu S, Umakanth S, Bhat V, Sampathila N. Clinical and Laboratory Approach to Diagnose COVID-19 Using Machine Learning. *Interdiscip Sci*. 2022 Jun;14(2):452-470.
- [26] Kukar M, Gunčar G, Vovko T, Podnar S, Černelč P, Brvar M, Zalaznik M, Notar M, Moškon S, Notar M (2021) COVID-19 diagnosis by routine blood sinamas using machine learning. *Sci Rep* 11(1):1–9.
- [27] Yang HS, Hou Y, Vasovic LV, Steel PAD, Chadburn A, Racine-Brzostek SE, Velu P, Cushing MM, Loda M, Kaushal R, Zhao Z, Wang F. Routine Laboratory Blood Sinamas Predict SARS-CoV-2 Infection Using Machine Learning. *Clin Chem*. 2020 Nov 1;66(11):1396-1404.
- [28] AlJame M, Ahmad I, Imtiaz A, Mohammed A. Ensemble learning model for diagnosing COVID-19 from routine blood sinamas. *Inform Med Unlocked*. 2020;21:100449
- [29] Kang J, Chen T, Luo H, Luo Y, Du G, Jiming-Yang M. Machine learning predictive model for severe COVID-19. *Infect Genet Evol*. 2021 Jun;90:104737.
- [30] Sun N, Yang Y, Tang L, Li Z, Dai Y, Xu W, et al. (2021) A Prediction Model Based on Machine Learning for Diagnosing the Early COVID-19 patients. *J Antivir Antiretrovir*. S18:002.
- [31] Arpacı I, Huang S, Al-Emran M, Al-Kabi MN, Peng M (2021) Predicting the COVID-19 infection with fourteen clinical features using machine learning classification algorithms. *Multimedia Tools Appl* 80(8):11943–11957.
- [32] Zhang RK, Xiao Q, Zhu SL, Lin HY, Tang M. Using different machine learning models to classify patients into mild and severe cases of COVID-19 based on multivariate blood sinamaing. *J Med Virol*. 2022 Jan;94(1):357-365.
- [33] Plante TB, Blau AM, Berg AN, Weinberg AS, Jun IC, Tapson VF, Kanigan TS, Adib AB (2020) Development and external validation of a machine learning tool to rule out COVID-19 among adults in the emergency department using routine blood sinamas: a large, multicenter, real-world study. *J Med Internet Res* 22(12): e24048.
- [34] dos Santos Santana ÍV, da Silveira AC, Sobrinho Á, Silva LC, da Silva LD, Santos DF, Gurjão EC, Perkusich A (2021) Classification models for COVID-19 sinama prioritization in Brazil: machine learning approach. *J Med Internet Res* 23(4): e27293.
- [35] Luo C. L., Rong Y., Chen H., Zhang W., Wu L., Wei D., et al. (2019). A logistic regression model for noninvasive prediction of AFP-negative hepatocellular carcinoma. *Technol. Cancer Res. Treat*. 18:1533033819846632.

- [36] Heijnen B. J., Bohringer S., Speyer R. (2020). Prediction of aspiration in dysphagia using logistic regression: oral intake and self-evaluation. *Eur. Arch. Otorhinolaryngol.* 277 197–205.
- [37] Zhu N, Zhang D, Wang W, Li X, Yang B, Song J, Zhao X, Huang B, Shi W, Lu R, Niu P, Zhan F, Ma X, Wang D, Xu W, Wu G, Gao GF, Tan W, China Novel Coronavirus Investigating and Research Team. A Novel Coronavirus from Patients with Pneumonia in China, 2019. *N Engl J Med.* 2020 Feb 20,382(8):727-733.
- [38] Zhang RK, Xiao Q, Zhu SL, Lin HY, Tang M. Using different machine learning models to classify patients into mild and severe cases of COVID-19 based on multivariate blood sinamaing. *J Med Virol.* 2022 Jan,94(1):357-365.
- [39] Laatifi M, Douzi S, Bouklouz A, Ezzine H, Jaafari J, Zaid Y, El Ouahidi B, Naciri M. Machine learning approaches in Covid-19 severity risk prediction in Morocco. *J Big Data.* 2022,9(1):5.
- [40] A.F.de M. Batista, J.L. Miraglia, T.H.R. Donato, A.D.P. Chia vegatto Filho, COVID-19 diagnosis prediction in emergency care patients: a machine learning approach, 2020.
- [41] An C, Lim H, Kim DW, Chang JH, Choi YJ, Kim SW. Machine learning prediction for mortality of patients diagnosed with COVID-19: a nationwide Korean cohort study. *Sci Rep.* 2020 Oct 30,10(1):18716.
- [42] Du R, Tsougenis ED, Ho JWK, Chan JKY, Chiu KWH, Fang BXH, Ng MY, Leung ST, Lo CSY, Wong HF, Lam HS, Chiu LJ, So TY, Wong KT, Wong YCI, Yu K, Yeung YC, Chik T, Pang JWK, Wai AK, Kuo MD, Lam TPW, Khong PL, Cheung NT, Vardhanabhuti V. Machine learning application for the prediction of SARS-CoV-2 infection using blood snamas and chest radiograph. *Sci Rep.* 2021 Jul 9,11(1):14250
- [43] Bertsimas D, Lukin G, Mingardi L, Nohadani O, Orfanoudaki A, Stellato B, Wiberg H, Gonzalez-Garcia S, Parra-Calderón CL, Robinson K, Schneider M, Stein B, Estirado A, A Beccara L, Canino R, Dal Bello M, Pezzetti F, Pan A, Hellenic COVID-19 Study Group. COVID-19 mortality risk assessment: An international multi-center study. *PLoS One.* 2020 Dec 9,15(12): e0243262.
- [44] Sánchez-Montañés M, Rodríguez-Belenguer P, Serrano-López AJ, Soria-Olivas E, Alakhdar-Mohmara Y. Machine Learning for Mortality Analysis in Patients with COVID-19. *Int J Environ Res Public Health.* 2020 Nov 12,17(22):8386.
- [45] Hu C, Liu Z, Jiang Y, Shi O, Zhang X, Xu K, Suo C, Wang Q, Song Y, Yu K, Mao X, Wu X, Wu M, Shi T, Jiang W, Mu L, Tully DC, Xu L, Jin L, Li S, Tao X, Zhang T, Chen X. Early prediction of mortality risk among patients with severe COVID-19, using machine learning. *Int J Epidemiol.* 2021 Jan 23,49(6):1918-1929.
- [46] Kocadagli O, Baygul A, Gokmen N, Incir S, Aktan C. Clinical prognosis evaluation of COVID-19 patients: An interpretable hybrid machine learning approach. *Curr Res Transl Med.* 2022 Jan,70(1):103319.
- [47] Gangloff C, Rafi S, Bouzillé G, Soulat L, Cuggia M (2021) Machine learning is the key to diagnose COVID-19: a proof-of-concept study. *Sci Rep* 11(1):1–1.
- [48] Wu G, Zhou S, Wang Y, Lv W, Wang S, Wang T, Li X. A prediction model of outcome of SARS-CoV-2 pneumonia based on laboratory findings. *Sci Rep.* 2020 Aug 20,10(1):14042.
- [49] Göreke V, Sarı V, Kockanat S. A novel classifier architecture based on deep neural network for COVID-19 detection using laboratory findings. *Appl Soft Comput.* 2021 Jul,106:107329.
- [50] Sanche, S., Lin, Y., Xu, C., Romero-Severson, E., Hengartner, N., & Ke, R. (2020). Şiddetli Akut Solunum Sendromunun Yüksek Bulaşıcılığı ve Hızlı Yayılması Koronavirüs 2. Ortaya Çıkan Bulaşıcı Hastalıklar, 26(7), 1470-1477.
- [51] Chan JF, Yuan S, Kok KH, To KK, Chu H, Yang J, Xing F, Liu J, Yip CC, Poon RW, Tsoi HW, Lo SK, Chan KH, Poon VK, Chan WM, Ip JD, Cai JP, Cheng VC, Chen H, Hui CK, Yuen KY. A familial cluster of pneumonia associated with the 2019 novel coronavirus indicating person-to-person transmission: a study of a family cluster. *Lancet.* 2020 Feb 15,395(10223):514-523.

# Yazılım-tanımlı İHA Ağları için Deney Ortamı Tasarımı

## Test-bed Design for Software-defined UAV Networks

Gökhan SEÇİNTİ

*İstanbul Teknik Üniversitesi  
Bilgisayar ve Bilişim Fakültesi  
İstanbul, Türkiye  
secinti@itu.edu.tr  
ORCID: 0000-0003-0640-8368*

### Öz

İnsansız Hava Araçları (İHA) çeviklikleri ve esnek kişiselleştirilme seçenekleri ile geniş bir yelpazedeki problemlerin çözümlerinde kolay ve yaygın şekilde kullanılmaktadırlar. Fakat, bu çözümleri geliştirirken ve uygularken uygulama ortamının gerektirdiği birden fazla ve farklı telsiz haberleşme yöntemleri, benzer şekilde çeşitlilik gösteren hesaplama birimlerinin birbirleri ile uyum içerisinde işlemesi zorlu bir engel teşkil etmektedir. Bununla birlikte uçuş denetçisinin de İHA'ların görev sahasına uygun bir şekilde biçimlendirilmesi ve geliştirilmesi gerekmektedir. Bu çalışmada İHA'lar için esnek deney ortamı tanıtılmakta ve kurulum ve yürütülme basamakları anlatılmaktadır. Önerilen deney ortamı haberleşme için Wi-Fi, LoRa ve Yazılım-tanımlı Radyo erişim yöntemlerini desteklemekte ve ek olarak bu haberleşme teknolojilerini ağ katmanı seviyesinde, Yazılım-tanımlı Ağ yaklaşımı kullanılarak yatay düzlemde birleştirmekte ve karmaşık yönlendirme algoritmalarının kolayca yürütülmesine olanak sağlamaktadır. Dolayısı ile önerilen deney ortamı Fiziksel, Veri Bağlantı ve Ağ katmanlarının esnek şekilde değiştirilmesine ve bu katmanları kapsayan çözüm veya çözümlerin kolaylıkla geliştirilmesine olanak sağlamaktadır. Çalışmada son olarak bu önerilen deney ortamı kullanarak elde edilen deney detayları ve ilgili sonuçları paylaşılmaktadır.

**Anahtar sözcükler:** İnsansız Hava Araçları (İHA), Yazılım-tanımlı Radyo, Yazılım-tanımlı Ağlar (YTA)

### Abstract

Unmanned Aerial Vehicles (UAVs) can be used to solve a wide range of problems due to their high agility and flexible configuration capabilities. However, developing and implementing these solutions become a challenging obstacle

Gönderme, düzeltme ve kabul tarihi: 27.11.2023 - 10.04.2024 – 28.04.2024

Makale türü: Araştırma

as multiple communication technologies and computing units must be coordinated to work together seamlessly. Additionally, flight control must be provided appropriately for the task at hand. In this study, we introduce our flexible UAV-based test environment and explain the setup steps. Our test environment has Wi-Fi, LoRa, and Software-defined Radio capabilities for communication. Furthermore, these communication technologies can be adjusted as needed at the network layer level using the Software-Defined Networking paradigm. Therefore, our test environment facilitates the flexible coordination of the Physical, Data Link, and Network layers. Finally, we share the test results obtained using this proposed structure in our study.

**Keywords:** Unmanned Aerial Vehicles, Software Defined Radio, Software Defined Networks

### 1. Giriş

Günümüzde İHA'lar, teknolojideki ilerlemeyle birlikte, birçok farklı sorunun çözümünde temel araç olarak kullanılmaktadır. İHA'lar tarımdan enerji endüstrisine, acil durum müdahalelerinden çevresel izleme çalışmalarına kadar geniş bir ölçekte kritik roller üstlenmektedir. İHA'lar, tarımda bitki sağlığını sürdürebilmek ve toprak verimliliğini arttırmak için kullanılırken, enerji endüstrisinde ise güç hatlarının bakımını yapmak ve rüzgâr türbinlerini izlemek ve denetlemek için kullanılmaktadır [1]. Ayrıca afet hali gibi kritik ve hızlı müdahale gerektiren durumlarda kurtarma çalışmaları için destek sağlayabilir ve zarar tespitlerinde önemli bir rol oynayabilir. Bu alanlara ek olarak, bu araçlar savunma sanayisi başta olmak üzere ülkelerin doğal kaynaklarının ve varlıklarının korunması gibi askeri alanlarda da her geçen gün daha fazla kullanılmaktadır.

İHA'ların maliyet açısından diğer hava araçlarına göre daha erişilebilir olması ve daha kolay kişiselleştirilebilmesi kullanım

alanlarını benzer birim ve çözümlere kıyasla arttırmaktadır. Ek olarak bu hava araçların denetiminin uzaktan sağlanması ve sürdürülebilmesi, İHA'ların kritik ve tehlikeli görevleri üstlenmesini kolaylaştırmaktadır. Ayrıca, bu araçlar ile kullanılacak geniş duyurga çeşitliliği sayesinde bahsedilen kritik görevlerde ve alanlarda etkin şekilde veri toplanabilmekte ve farklı uygulama alanlarında da değerlendirilebilmektedir. Bu sayede afet durumları gibi insan gücünün kısıtlı olduğu ve mevcut altyapının yetersiz olduğu ortamlarda, görevli insanların (arama kurtarma ekipleri, itfaiye ekipleri vb.) ulaşamayacağı ve müdahale edemeyeceği alanlara kolaylıkla ve risksiz şekilde erişim sağlanır.

Yukarıda bir kısmı tanımlanan bu kullanım alanlarında ve ilgili çözümlerde, İHA'lar tekil olarak kullanılabilen ya da takım olarak görev yapabilmektedirler. Takım halinde kullanıldıklarında İHA'ların eş güdümlü şekilde hareket etmelerinin sağlanması ve yer denetçisinin İHA'lar ile sürekli ve güvenilir iletişim kurması görev başarısı açısından büyük önem taşımaktadır. İHA'ların sahada görev veya görevlerinin başarılı şekilde tamamlanmasını sağlamak için İHA'lar arası ve İHA-yer denetçisi arası bir veya birden fazla iletişim ağının kurulması ve eş zamanlı kullanılması gerekmektedir. Bu ağlarda kullanılacak haberleşme sistemleri için uydu bağlantıları, mobil ağlar ve baz istasyonları gibi seçenekler mevcuttur. Bu haberleşme sistemleri ve ağ altyapıları haberleşme altyapılarını güvenilir ve/veya iş birliği içerisinde olduğu ortamlarda gerekli hizmetleri ve işlevleri sağlayabilmektedir. Fakat mevcut iletişim altyapılarının bulunmadığı (askeri ortamlar, arama kurtarma çalışmaları vb.) ya da altyapı hizmetinin kalitesiz ve/veya yetersiz olduğu ve doğrudan insan müdahalesinin mümkün olmadığı durumlarda İHA'ların etkin şekilde yönetimini sağlayabilmek için özgün ağ yapılarına ihtiyaç duyulmaktadır. Tasarsız (*ad-hoc*) ağ yaklaşımı İHA kümelerinin, iletişim altyapısının zayıf veya güvenilir olmadığı bölgelerde kullanılmasına olanak tanımaktadır. Bu ağ yapısı, doğası gereği karmaşık tasarımlara olan ihtiyacı ortadan kaldırmakta ve esnek bir haberleşme ortamı sağlamaktadır. Bu bağlamda tasarsız ağların ilgi çekiciliği, benzer ağ altyapılarının işlevlerini merkezi bir denetçiden bağımsız şekilde yerine getirebilmelerinden kaynaklanmaktadır. Bu doğrultuda, bu çalışmada İHA'lar arası haberleşme için temel olarak Wi-Fi (IEEE 802.11) protokolü ve bu protokolün tasarsız ağ yaklaşımından (*IBSS – Independent Basic Service Set*) yararlanan bir İHA deney ortamı önerilmektedir. Ek olarak bu çalışma haberleşme altyapısının kısıtlı olduğu ortamlardaki iletişim zorluklarına çözüm getirmek için LoRa ve YTA'dan faydalanan yeni bir yaklaşım sunulmaktadır.

Bu çalışmada önerilen yazılım-tanımlı İHA ağı deney ortamı, ağ üzerindeki veri akışlarını yönetmek için yazılım-tanımlı bir denetçiden faydalanmaktadır. Bu kapsamda OpenFlow [24] iletişim protokolü, İHA'lar ve ilgili denetçi arasında dinamik ve programlanabilir bir ağ oluşturmak için kullanılmaktadır. Dolayısıyla OpenFlow yazılım-tanımlı iletişim kurallarının ağı yönetmesi ile esnek ve programlanabilir bir denetim elde edilmektedir. Böylelikle geleneksel ağ mimarileri ile karşılaştırıldığında yeni yöntemlerin daha hızlı ve kolay bir şekilde geliştirilmesine ve deneylerinin yapılmasına olanak

sağlanmaktadır. Yazılım-tanımlı denetçi, İHA'lar arasındaki tasarsız iletişimi düzenlemede ve hızla değişen ağ koşullarına uyum sağlamada çok önemli bir rol oynamaktadır. OpenFlow iletişim protokolü yalnızca hızlı ve son derece hareketli İHA'ların denetimini desteklemekle kalmamakta, aynı zamanda ağ trafiğinin de doğrudan denetlenmesine ve yönlendirilmesine olanak tanımaktadır. Bu yetenek, bilhassa İHA'lar arasında hızlı veri paylaşımının gerekli olduğu durumlarda çok kritik bir önem teşkil etmektedir. Denetim algoritmalarının anlık olarak ortamdaki değişikliklere göre yürütülmesi, İHA'ların zorlu koşullar altındaki durumlarda bile verimli şekilde uçmasını ve iletişim kurmasını sağlamaktadır. Yukarıda da belirtildiği gibi önerilen mimari, Wi-Fi kullanan tasarsız ağ yapısını (IEEE 802.11 IBSS) deney ortamına dâhil ederek, sınırlı altyapıya sahip alanlarda haberleşme koşullarını iyileştirmekte ve aynı zamanda pratik ve etkili bir çözüm sunmaktadır. Bu ağların denetimi ise Yazılım-tanımlı ağlar (YTA) ve OpenFlow iletişim protokolü kullanılarak yapılmaktadır. Böylelikle İHA'ların dinamik ve kolay öngörülemez ortamlarda bile başarılı bir şekilde konumlandırılması için esnek, kolay programlanabilen bir denetleme katmanı oluşturulmaktadır. Bu çalışmanın temel amacı, İHA teknolojilerini ve bunların kaynak kısıtlı ortamlardaki uygulamalarını daha hızlı geliştirmeyi ve farklı haberleşme katmanlarını ve hesaplama unsurlarını kapsayan karmaşık deneyleri kolaylaştıran bir ortam sunmaktır. Bu sayede de bu çalışma, ivmesi gün geçtikçe artan İHA araştırma ortamına destek olmayı hedeflemektedir.

Yukarıda verilen gerekçe ve çıkarımlara ek olarak, YTA ile denetlenen WiFi tabanlı tasarsız bir ağ yapısı esnek bir kullanım sağlasa da uçtan uca haberleşme sırasında ihtiyaç duyulan ara birim sayısındaki artış ile uçtan uca haberleşmedeki veri hızında ciddi bir düşüşe ve gecikme miktarında büyük bir artışa neden olmaktadır [25]. Bu kapsamda düşük enerji gereksinimi ile uzun menzilli haberleşmeye olanak tanıyan LoRa teknolojisi önemli bir potansiyel sağlamaktadır. Önerilen deney ortamında LoRa'nın kullanımı yalnızca haberleşme menzilini genişletmeyi değil, aynı zamanda birbirinden fiziksel olarak uzakta yer alan uç birimlerin haberleşmesi sırasında oluşabilecek yüksek gecikmeleri de denetim altında tutmayı amaçlamaktadır. Ancak LoRa teknolojisinin bu kazanımları (düşük enerji ve uzun menzil) sağlanan veri hızından feragat etmesi ile mümkündür. LoRa'nın bu kazanımlarının yanı sıra sağlayabildiği veri aktarım hızı, LoRa teknolojisinin İHA ağındaki tüm uçtan uca bağlantılar için kullanılmasını imkânsız hale getirmektedir. Bu kısıt göz önünde bulundurulduğunda LoRa'nın İHA ağı üzerindeki bilgi trafiği için değil, yalnızca yer denetçisi ve İHA'lar arasında gereken denetim trafiği için kullanılması uygun görülmektedir. LoRa'nın deney ortamına eklenmesi sayesinde, merkezi yer denetçisi ile İHA'lar arasında aktarılan ağ trafiği ve İHA denetim paketleri LoRa ağ bağlantısı üzerinden iletilebilmektedir. LoRa'nın bu kullanımı, denetim trafiğinin hatasız ve güvenilir olmasını ve enerji açısından verimli bir şekilde iletilmesini sağlamaktadır. Bu sayede hem menzilde etkili bir artış sağlanmakta hem de denetim trafiğinde oluşabilecek yüksek gecikmelerin önüne geçilmektedir. Böylelikle İHA'ların uçuş sırasında ağıdaki veri iletimini etkin bir şekilde yönlendirmesi ve sürdürmesi



desteklenir. LoRa'nın, uzun menzilli haberleşme ve/veya yüksek irtifa gerektiren ortamlarda uçtan uca gecikmeleri daha ölçülenebilir seviyelerde tutma yeteneği, önerilen deney ortamının genel performansını iyileştiren önemli unsurlardan biridir. LoRa ağ bağlantısının, yer denetçisi ve İHA'lar arasındaki denetim trafiğini desteklemesi ile birlikte İHA'lar yazılım-tanımlı anahtarlar (*switch*) olarak kullanılabilir ve yer denetçisi üzerinde yer alan *OpenDayLight* (ODL) YTA denetçisi sayesinde akış tabloları uygulama katmanında yer alan basit betikler kullanılarak güncellenebilmektedir.

Bunların yanı sıra, yazılım-tanımlı radyo (YTR) kümesinde maliyet etkin yaklaşımlarından biri olan LimeSDR kullanılarak İHA anahtarlar üzerinde veri trafiği için ek bir iskele tanımlanmıştır. Bu iskele, TCP/IP ağ mimarisinde ilk üç katmanda çalışarak karmaşık deneylere olanak ve esneklik sağlamaktadır. Anahtar üzerinde yer alan ve kolaylıkla programlanabilen bu iskele sayesinde farklı kipler (*modulation*) ve kodlama teknikleri, İHA üzerinde görev sırasında kullanılabilir ve tüm katmanları kapsayan bir en iyilemeye olanak tanımlanmaktadır. Bir başka deyişle, önerilen mimari ile, özgün uyarlanabilir kipler ve kodlama (*adaptive modulation and coding, AMC*) kullanan yöntemler kolaylıkla İHA ortamı üzerinde geliştirilebilir ve başarımları ölçülenebilir. Uyarlanabilir kipler ve kodlama ile birlikte, haberleşme birimlerinin özel ihtiyaçlarına göre haberleşme frekans bant genişliği dinamik olarak denetlenmekte ve ayarlanmaktadır. Bu sayede değişen ağ koşullarına uyum sağlayan gerçek zamanlı denetimlerle İHA ortamının hizmet niceliği iyileştirilmektedir. Sonuç olarak, LoRa, YTA ve YTR gibi teknoloji ve yaklaşımların birlikte kullanımı, dinamik ve güvenilir bir İHA iletişim ağı elde etmek için kapsamlı bir deney ortamı elde etmemizi sağlamaktadır. Gelişmiş kipler tekniklerinin kullanılması ise, önerilen ağın değişen çevresel koşullara dinamik olarak uyarlanabilir olmasını sağlayarak, haberleşme koşullarının öngörülemez olduğu İHA uygulamaları için sağlam bir çözüm sunmaktadır. Bu tümleşik yaklaşım, özellikle geleneksel iletişim altyapılarının yetersiz kaldığı durumlarda İHA teknolojilerinin ilerlemesine katkıda bulunmaktadır. Özet olarak, bu çalışmada sunulan temel katkılar aşağıdaki gibi sıralanmaktadır:

- Düşük enerji gereksinimi ve uzun menzilli haberleşmeye olanak tanıyan LoRa teknolojisinin İHA ağlarına bütünleştirilmesi ile özellikle yer denetçisi ve İHA'lar arasında sağlam ve enerji açısından verimli denetim trafiği yaklaşımı sunulmuştur. Bu, özellikle uzun menzilli görevlerde ve zorlu ortamlarda İHA'ların verimli iletişimini ve uçuşunu desteklerken, yüksek gecikmeleri azaltarak ağ performansını iyileştirmektedir.
- Makale, bir YTR yaklaşımı olan LimeSDR'in İHA ağlarında kullanılması ile haberleşme birimlerinin frekans bant genişliği dinamik olarak ayarlanabildiğini; kipler ve kodlama teknikleri değişen ağ koşullarına göre uyarlanabildiğini göstermektedir. Bu önerilen yaklaşım İHA ağ ortamının hizmet kalitesini artırırken, ağı çevresel koşullara daha dinamik bir şekilde uyum

sağlamasını ve haberleşme koşullarının öngörülemez olduğu durumlar için sağlam bir çözüm sunmasını hedefler.

- Makalenin önerdiği yazılım-tanımlı İHA ağı deney ortamı, veri akışlarını yönetmek için OpenFlow iletişim kurallarıyla etkileşimli ve dinamik bir ağ yapısı sunar. Bu sayede İHA'lar için gelişmiş denetleme mekanizması sağlanır; fiziksel, veri bağı ve ağ katmanındaki hızlı değişiklikler YTA altyapısında gerçekleştirilir.
- Önerilen deney ortamı, İHA teknolojilerinin ve kaynak kısıtlı ortamlardaki uygulamalarının daha hızlı geliştirilmesini ve deneylerinin yürütülmesini kolaylaştıran önemli bir akademik katkıdır. Ayrıca, İHA araştırma ortamına önemli bir destek sağlayarak, TA yapısına bütünleştirilmesi zor olan LoRa ve YTR tekniklerinin kurulumunu ve deney ortamının nasıl hazırlandığını adım adım paylaşır.

Makalenin geri kalan bölümlerine ait düzen şu şekildedir: Bir sonraki bölümde, literatürde var olan ilgili çalışmalar anlatılmıştır ve neden önerilen deney ortamına ihtiyaç olduğu gerekçelendirilmiştir. Üçüncü bölümde ise önerilen deney ortamı kapsamında kullandığımız donanımlarla birlikte deney ortamımızın nasıl çalıştığı ve hangi telsiz iletişim ve İHA denetim teknolojilerinin önerilen test ortamına eklendiği detaylı şekilde açıklanmıştır. Dördüncü bölümde deney ortamımızdan elde ettiğimiz sonuçlar gösterilip tartışılmıştır. Son bölümde ise önerilen yaklaşım ile ilgili önemli detaylar vurgulanıp makalenin çıkarımları sonuçlandırılmıştır.

## 2. Kaynak Araştırması

Kaynaklara baktığımızda İHA destekli başka ortamları da bulunmaktadır. Bunların çoğu özelleştirilmiş bir sorunun çözümüne odaklanmıştır. Örneğin [2], YTR kullanarak bir yazılım-tanımlı ağ destekli İHA ortamı önermiştir. Bu çalışma ağ işletmeni tarafından verilen denetimlerin en uygun şekilde yürütülmesi için onları dağıtık bir en iyileştirme sorununa çevirip, İHA'ların eş güdümlü bir şekilde çalışmasını sağlayarak yaklaşım sunmuşlardır. Bir diğer deyişle, SwarmControl ismini verdikleri bir çözümle İHA ağlarının denetimini basitleştirmeyi ve en iyileştirmeyi amaçlamışlardır. Yazarlar, SwarmControl'un benzer çözümlere göre daha etkili ve esnek olduğunu, ayrıca YTA tabanlı ön ürünün deneylerde de öne çıktığını belirterek, önerdikleri çözümün ağ denetimini basitleştirip insan müdahalesini en aza indirdiklerini vurgulamışlardır.

Önerilen bir başka çalışmada ise [3], yazarlar, yüksek hareketli özelliklere sahip uçan tasarsız ağlarda yönlendirme ve erişim iletişim kurallarını kullanarak ağ kapasitesini arttıran bir çözüm önermişlerdir. Bu amaçla, Wi-Fi, YTR ve Doğrudan Sıralı Yayılı Spektrum (DSYS) teknolojileri bir arada kullanılarak esnek bir yönlendirme ve kanal seçim yöntemi geliştirmişlerdir. Sonuç olarak, önerdikleri yöntemin, yalnızca IEEE 802.11 IBSS bağlantılarını kullanan bir temel duruma kıyasla %51,22 daha fazla net veri hızı ve %52,44 daha az gecikme sağladığını göstermişlerdir. Bu çalışmada ise [4] LoRa teknolojisinin Nesnelerin İnterneti erişim ağları için uygun olduğu kadar tasarsız ağlarda da etkili olduğu vurgulanmıştır.

**Çizelge-1: Kaynaklardaki sistemler ile karşılaştırma**

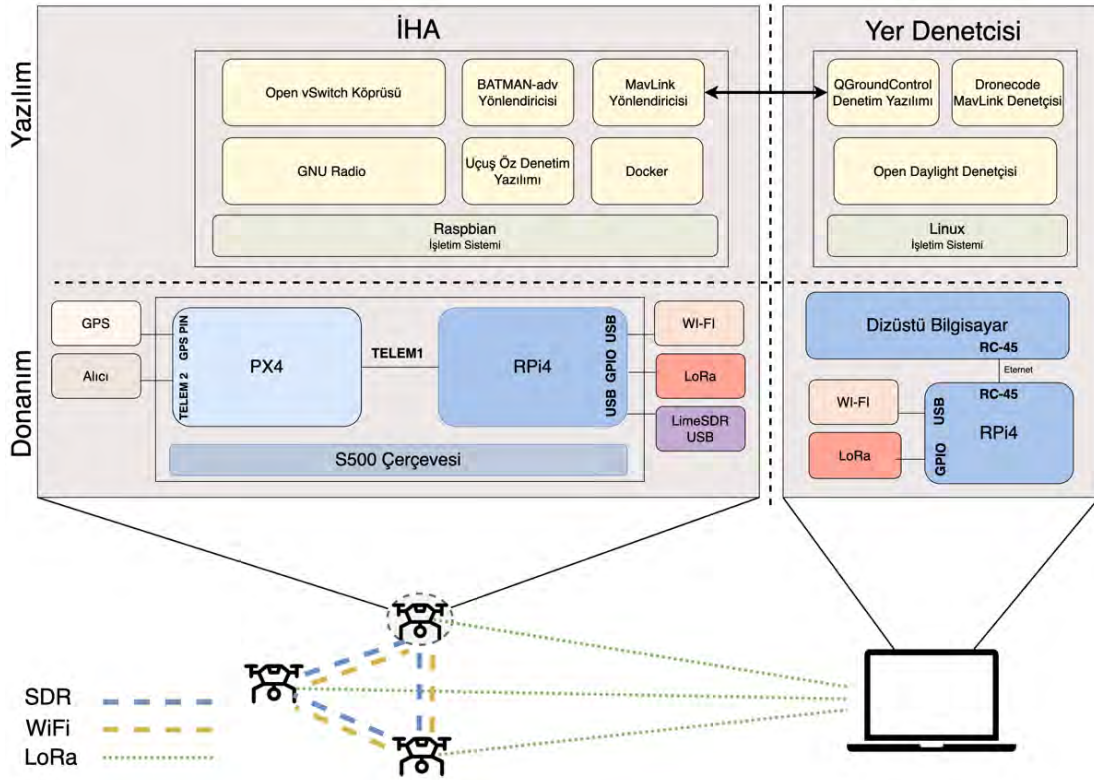
<i>Çalışma</i>	<i>Wi-Fi</i>	<i>LoRa</i>	<i>YTR</i>	<i>CPU</i>	<i>İHA</i>	<i>YTA</i>	<i>Gerçek Deney Ortamı</i>
[2]	✓		✓	✓	✓	✓	✓
[3]	✓		✓		✓		✓
[5]	✓	✓		✓	✓		✓
[6]		✓		✓	✓		
[8]	✓		✓		✓		✓
[12]			✓	✓	✓	✓	
[16]	✓			✓			✓
[19]	✓		✓	✓	✓		✓
[20]	✓		✓	✓	✓		✓
[21]	✓			✓			✓
[25]	✓			✓	✓	✓	✓
[26]	✓			✓	✓	✓	
Bu çalışma	✓	✓	✓	✓	✓	✓	✓

Tarım, ormancılık ve benzeri çevresel uygulamalar için yapılmış [5] bu çalışmada ise hücrel ağ kapsama alanı dışındaki kırsal ortamlarda İHA tabanlı düşük güçlü geniş alan ağı deneylerini desteklemek için geliştirilmiş bir deney ortamı sunulmuştur. Çalışmada, iki katmanlı İHA destekli dar bant Nesnelerin İnterneti ve LoRa kullanılarak geliştirilen ağı deney sonuçları paylaşılmış ve verimli sonuçlar verdiği paylaşılmıştır. Yine bir başka çalışmada [6], LoRa ağ geçitleri ile donatılmış İHA'ların, özellikle çeşitli değişkenleri algılayan çok sayıda yer duyargasıyla, tarım alanları gibi geniş alanların izlenmesinde önemli avantajlar sunduğu belirtilmiştir.

Bu çalışmada [7] önerilen mimaride ise yazılım-tanımlı İHA'lar için 'uçmaya hazır' bir sanal ortam oluşturulmuştur. Bu çerçevede farklı İHA ağ denetim sorunları çok etmenli Derin Pekiştirmeli Öğrenme (DPÖ) yaklaşımı ile modellenmiş ve performans verileri toplanarak büyük İHA ağları için verimli bir şekilde ölçeklendirilmiştir. Başka bir çalışmada [8], İHA ve YTR kullanılarak dağıtık mimaride havadan-yere hüzmeleme sistemi geliştirilmiştir. Bu yaklaşım sayesinde İHA sürüsü ile YTR arasında eş zamanlama sağlanmış ve yönlendirmedeki verimliliği arttırmak amacıyla da havadan-yere hüzmeleme sistemine ağırlıklar atanmıştır.

Daha birçok araştırmacı, İHA'ların iletişim yeteneklerini artırmak [9], uzun uçuş sürelerine ulaşmak [10,11] ve görev tamamlamada daha iyi çıktılar elde etmek [12] için İHA'lar üzerinde çalışmış ve çalışmaktadır. Ancak, geliştirilen yöntemlerin ve sistemlerin gerçek hayatta uygulanabilirliğini göstermek ve deneysel doğrulamaları yapmak için öz yönetimli, kapsamlı ve esnek özellikleri olan bir deney ortamının eksik olduğu açıkça görülmektedir.

Öte yandan OpenFlow tabanlı YTA sisteminde ağ cihazları arasındaki veri yolu üzerinde esnek kontrol sağlayarak, ağ trafiğinin daha verimli yönetilmesine imkân tanımıştır [26]. Bu teknoloji, İHA ağlarının yönetiminde önemli bir potansiyel sunar; çünkü İHA ağlarının karşı karşıya olduğu değişken koşullar ve gereksinimler, OpenFlow'un sağladığı esnekliği gerektirir. Ancak, mevcut literatürde, OpenFlow ve benzeri YTA teknolojilerinin İHA ağları üzerindeki uygulamalarına yönelik detaylı deney ortamları ve bu teknolojilerin pratik etkilerini kapsamlı bir şekilde inceleyen çalışmaların eksikliği dikkat çekmektedir. Bir diğer detaylı inceleme yazısında ise ağ fonksiyonlarının sanallaştırılması (NFV) ve YTA'nın entegrasyonunu ele almıştır [27]. Bu entegrasyon, ağ hizmetlerinin maliyet etkin bir şekilde dağıtılmasını ve İHA ağlarının dinamik gereksinimlerine hızlı bir şekilde adapte olunmasını sağlar. Bu çalışma, NFV ve YTA'nın potansiyel avantajlarını belirlemekle birlikte, İHA ağları için spesifik olarak tasarlanmış ve bu teknolojilerin etkilerini detaylı bir şekilde analiz eden deney ortamlarının kurulması konusunda bir boşluk bırakmıştır. İHA tabanlı yeni uygulamaları inceleyen veya deneylerini yürüten birçok çalışma, genellikle benzetim ortamlarından faydalanmaktadır [9,13,14,15]. Ayrıca bir diğer çalışmada [16], açık havada deney ortamının kurulmasının zorluklarına ve her bölge için uçuş izninin alınmasının kolay olmadığına dikkat çekilerek kapalı ortamda uçuş yapabilecek boyutta minik İHA'larla deney ortamı oluşturmuşlardır. Özetle, İHA destekli deney ortamları genellikle belirli özelleştirilmiş sorunlara odaklanmış veya bu konudaki önceki çalışmalarda da [17] vurgulandığı gibi sadece kuramsal çözümleme ve/veya bilgisayar tabanlı benzetim yöntemlerine yönelmişlerdir. Fakat, bu mevcut yaklaşımlar genellikle belirli kullanım durumlarına odaklanmaktadır ve



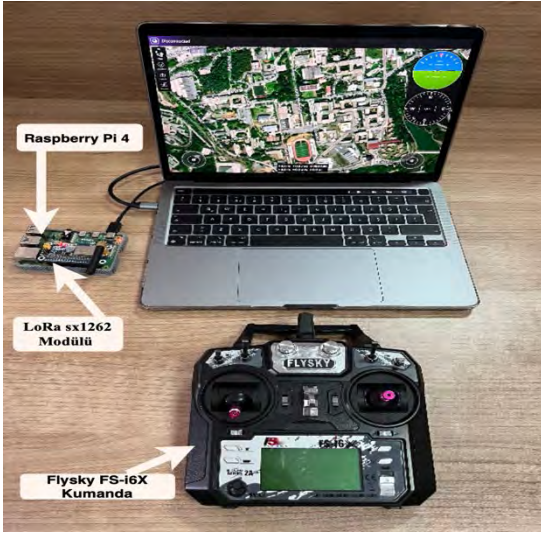
Şekil-1: Önerilen deney ortamına ait yazılım ve donanım mimarisi

genel bir bakış açısı sağlayacak şekilde çözüm önerememektedirler.

Kapsamlı deney ortamının eksikliği, önerilen çalışmaların tam anlamıyla doğrulanamamasına yol açmaktadır [18]. Bununla birlikte çeşitli kaynaklar, İHA tabanlı sistemlere dair detayları yetersiz bir şekilde sunmaktadır. Bazıları sadece kullanılan İHA'nın görsellerini paylaşmakla yetinirken; birçoğu İHA üzerinde önerilen değişikliklerin azami uçuş süresi gibi ilgili iyileştirmelerine olan katkısını detaylı bir şekilde incelememektedirler [19,20]. Suatlı telsiz iletişim sistemleri üzerine odaklanan bu araştırmada [21], geliştirilen iletişim sistemlerinin çıktısını ve güvenilirliğini, gerçek hayattaki çalışmalarda doğru bir şekilde değerlendirebilmek adına deney ortamının hayati bir önemi olduğu vurgulanmıştır. Ek olarak, istikrarlı, tekrarlanabilir ve esnek bir çalışma ortamının oluşturulması bu değerlendirmeleri sağlıklı bir şekilde yapabilmek için gerekli ve şarttır [22, 23]. Önerilen bu çalışmanın önceki çalışmalardan en önemli farkı, YTR, LoRa, WiFi veya YTA gibi birden fazla farklı haberleşme teknolojilerini ve yaklaşımlarını bir araya getirerek gerçek zamanlı kapsamlı bir deney ortamı oluşturmasıdır. Oluşturulan bu deney ortamı, farklı telsiz iletişim yaklaşımları ve etkileşimini durumlar ve ortamlar üzerinde gözlemlemek için kullanılabilir. Bu kapsamlı deney ortamı, ilgili teknolojilerin bütünleştirilmesi ile oluşturulduğundan, çeşitli telsiz iletişim unsurlarının karmaşıklığını ve birbiriyle olan etkileşimlerini gerçekçi bir şekilde benzetimini yapma kapasitesine sahip olacaktır. Son olarak Çizelge-1'de literatürdeki diğer sistemler ile kendi önerdiğimiz sistemin özelliklerini karşılaştırıyoruz.

### 3. Önerilen Yazılım-tanımlı İHA Ağı Deney Ortamı

Deney ortamı, İHA'lar arasındaki etkileşimi ve İHA'lar ile yer denetçisi arasındaki iletişimi destekleyecek şekilde gelişmiş bir iletişim altyapısı sunar. Bu altyapının temelini oluşturan üç ana iletişim kuralı, YTR, LoRa, ve Wi-Fi, birlikte bir düzen içinde haberleşme ağı sağlar. YTR yaklaşımı İHA'lar arasındaki iletişimi esnek ve uyarlanabilir hale getirirken, LoRa teknolojisi düşük güç tüketimi ve uzun menzil özellikleriyle uzaktaki İHA'larla bile güvenilir bir iletişim kurulmasını mümkün kılar. Wi-Fi ise yüksek hızlı veri transferi ve düşük gecikme süreleriyle İHA'lar arasında hızlı ve güvenilir eş güdümü sağlar. Bu üç temel iletişim protokolü, İHA'lar arasındaki eş güdümü artırırken, yer denetçisi ile sağlanan bütünleştirme sayesinde etkileşimi en iyi duruma getirir ve sistemin verimini güçlendirir. İHA'nın temelinde S500 gövde çerçevesi, Pixhawk 4 uçuş denetim birimi, Raspberry Pi 4 mini bilgisayar, GPS birimi, FlySky FS-X6B uzaktan kumanda, LoRa SX1262 ve LimeSDR USB birimi bulunmaktadır. Şekil-1'de verilen genel iletişim mimarisinde; iki ana düzlemde, İHA düzlemi ve yer düzlemi, iletişim sağlanmaktadır. Bu donanım ve iletişim mimarisi, İHA'nın dengeli uçuş ve görev yürütme kabiliyetini güçlendirmektedir. Deney ortamında Şekil-3' de görüldüğü üzere; güç kaynağı, bir 3S lityum-polimer (LiPo) pil ile sağlanır. RPi4, doğrudan PX4 uçuş denetim biriminin Telem1 bağlantı noktasına bağlanır ve 57600 baud hızında veri iletimi sağlar. İletişim, RPi4 ve PX4 uçuş denetim birimi arasında MAVLink iletişim kuralı kullanılarak gerçekleşir.



Şekil-2: Yer denetçisinde kullanılan donanım

Mavlink İHA haberleşmeleri için yaygın olarak kullanılan bir iletişim kuralıdır.

LoRa birimi olarak Waveshare sx1262 seçilmiştir, 915 Mhz sıklığında çalışır ve bu RPi4 ile uyumludur. RPi4'e bütünleşmesi doğrudan RPi çıkış yerleri üzerinden sağlanır ve LoRa iletişimi için gerekli çalışma ayarları yapılandırılmıştır.

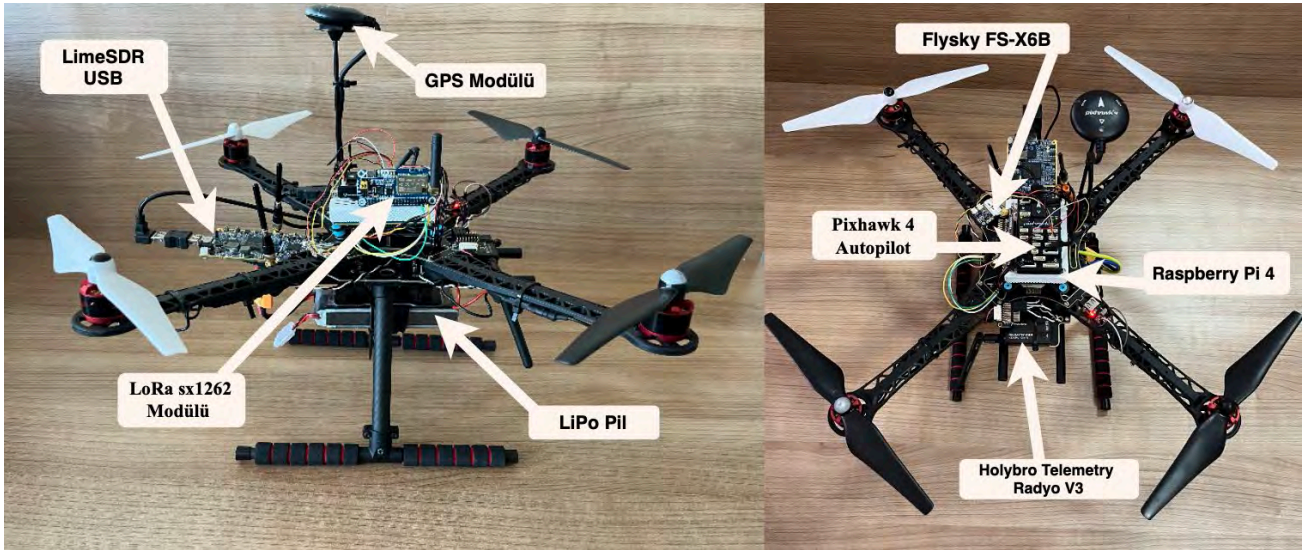
GPS birimi px4 uçuş denetim biriminin GPS bağlantı noktasına bağlanırken, FlySky alıcısı Telem2 bağlantı noktasıyla bütünleştirilir. Deney ortamının önemli bileşenlerinden biri de YTR'dir ve çeşitli radyo işaretlerini yakalayıp iletişim yeteneklerini daha da güçlendirir. Deney ortamımızda YTR özelliklerini YTA'ya uygulamak ve haberleşmeyi sağlamak için LimeSDR USB birimi, USB iskelesi üzerinden RPi4'e bağlanır. Yer denetçisi Şekil-2'de verilmiştir. QGroundControl (QGC), İHA'nın durumunu izlemek ve gerekli ayarlamaları yapmak için yer kontrol seviyesinde kullanıcı için bir arayüz sağlar. RPi4 ve Px4 uçuş denetçi birimi arasındaki MAVLink iletişimi, İHA'nın anlık uzaktan ölçüm verilerinin yer istasyonuna iletimini sağlarken aynı zamanda kullanıcı tarafından

gönderilen komutların da İHA'ya iletilmesini sağlar. Mavlink mesajlarının uzun mesafeleri desteklemesi ve düşük güç tüketimi sunması dolayısı ile LoRa üzerinden iletilir. İHA kumanda ile yönlendirilebileceği gibi QGC'e Wi-Fi bağlantısı ile de bağlanıp yönlendirilebilir. Tipik olarak uzaktan ölçüm yapan bir radyo birimi, iğine bağlantılarını kullanarak uçuş denetim biriminin Telem1 bağlantı noktasına bağlanır ve iletişim için QGC ile bir radyo bağlantısı kurar. Ancak deney ortamımızda RPi ile iletişimi kolaylaştırmak için uçuş denetim birimi üzerindeki Telem1 bağlantı noktasını genel bir MAVLink iletişim bağlantı noktası olarak kullanıyoruz. QGC ile bağlantıyı ise LoRa bağlantısı üzerinden gerçekleştiriyoruz. LoRa birimleri içinde bulunan LoRa SX1262 kütüphanesi kullanılarak kararlı bir bağlantı sağlanır ve İHA ile yer denetçisi arasında iletişim kurulur. Bu deney ortamı, uzun menzilli ve güçlü bir iletişim yeteneği sunar, böylece İHA işlemleri ve değişkenleri uzaktan hassas bir şekilde izlenebilir ve yönetilebilir hale gelir.

### 3.1 Önerilen Deney Ortamının Pratikte Kullanımı

Bu çalışmanın gerçek dünya senaryolarında nasıl uygulanabileceği ve pratik etkinliği, gelişmiş iletişim teknolojileri ve yazılım-tanımlı ağ (YTA) yaklaşımlarını entegre eden önerdiğimiz yenilikçi İHA ağı deney ortamımız ile çeşitli kullanım senaryoları ile sağlanabilir.

Wi-Fi ve LoRa'nın yanı sıra YTR kullanarak sağladığımız esnek haberleşme altyapısı, geniş bir uygulama yelpazesinde adaptasyon ve optimizasyon imkânı sunmaktadır. Örneğin, acil durum müdahaleleri sırasında, afet bölgesinden gerçek zamanlı veri toplama ve hızlı bir şekilde yer denetçisiyle iletişim kurma kapasitesi, arama-kurtarma ekiplerinin etkinliğini artırarak insan hayatını kurtarmada hayati bir rol oynayabilir. İHA'larımızın uzun mesafeli ve enerji verimli haberleşme yetenekleri, zorlu koşullar altında bile güvenilir veri iletimi sağlayarak hızlı ve etkili bir müdahale imkânı sunmaktadır. Benzer şekilde, çevresel izleme projelerinde, İHA'larımız aracılığıyla elde edilen hassas tarımsal verilerin analizi, tarım sektöründeki verimliliği ve sürdürülebilirliği artırabilir. Bu senaryolar, önerilen deney ortamının gerçek



Şekil-3: Deney ortamında kullanılan bir İHA'nın yandan ve üstten görünümü

zamanlı uygulamalarda nasıl uyarlanabileceğini ve karmaşık veri akışlarını yönetirken esneklik ve yüksek performans sağlayabileceğini göstermektedir. Ayrıca, bu çalışma, gerçek zamanlı trafik yönetimi ve akıllı şehir uygulamaları gibi daha geniş bir yelpazede uygulama alanları için İHA teknolojilerinin potansiyelini açığa çıkarmaktadır.

Sonuç olarak, bu çalışma, İHA tabanlı sistemlerin geliştirilmesi ve test edilmesi için esnek, kapsamlı ve gerçek dünya koşullarına uygun bir deney ortamı sunarak, İHA teknolojilerinin ilerlemesine ve çeşitli sektörlerdeki uygulamalarına önemli katkılarda bulunmaktadır.

#### 4. Performans Değerlendirmesi ve Deneysel Sonuçları

Önerdiğimiz deney ortamının başarımını sınamak için iki İHA'dan oluşan iki farklı deney ortamı hazırlanmıştır. Bunlardan ilkinde, İHA'ları birbirine 10 metre uzaklıkta uçurup, aralarındaki mesafeyi arttırarak bağlantı kalitesindeki değişim, veri iletim hızı ve gecikme süresi incelenmiştir. Ayrıca YTA yardımı ile birbirinden bağımsız bağlantıları farklı haberleşme kuralları kullanarak gönderdik. Kampüs kuralları gereği geniş ölçekli uçuşlar planlayamadığımız için, aygıtların anten güçlerini azaltarak daha yüksek uzaklık gereken tetiklemeleri gerçekleştirebildik. Deneylerde üç farklı yöntemi karşılaştırmaktayız: sadece Wi-Fi, sadece YTR ve birleşik veri gönderimi. YTR ile olan yöntemde DSYS kullandık. Birleşik metotta ise göndereceğimiz verinin yarısını YTR yarısını ise Wi-Fi ile gönderiyoruz.

Bu yöntemde artan veriyi ikiye ayırma ve sonradan birleştirmeden gelen artmış karmaşıklık karşılığında daha yüksek veri hızı hedefledik. İkinci deney ortamımızda ise bir tane İHA'yı uçururken etraftaki Wi-Fi erişim noktalarından gelen işaretlere göre, o erişim noktaları ile İHA arasındaki bağlantı kalitesinin değişimini İHA'nın konumuna göre inceledik. Bu erişim noktalarından gelen işaretlerin kalitesini ise İHA üzerindeki YTR ile ölçülmüştür.

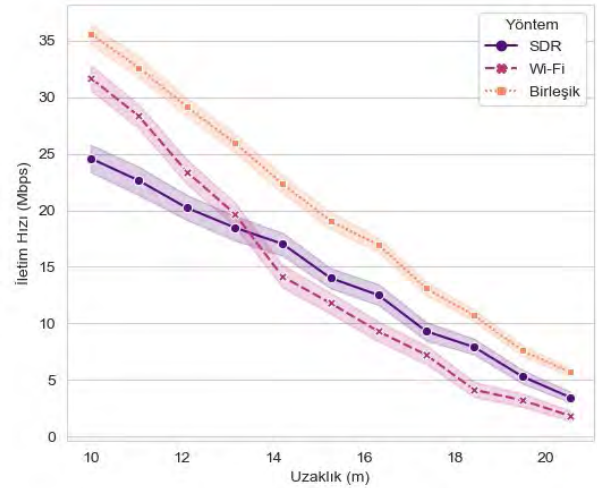
##### 4.1 Bağlantı Kalitesi Ölçümleri

İki İHA'nın birbirlerine göre uzaklıklarını arttırırken veri hızının değişimini Şekil-4'te göstermekteyiz. Burada, Wi-Fi uzaklık azken YTR'dan daha hızlı veri iletimi sağlasa da uzaklık arttıkça YTR'dan daha kötü bir çıktı sergilemiştir. Bunun nedeni DSYS'nin daha yüksek bir bant genişliğine sahip olması ve bu nedenle daha dayanıklı bir iletim sağlamasıdır.

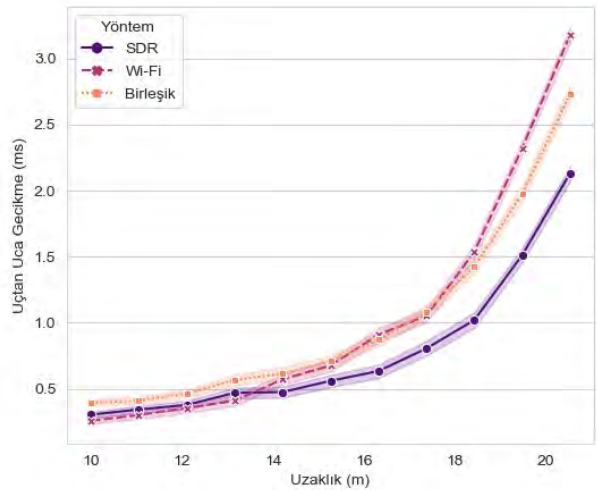
Ayrıca Wi-Fi ile YTR'yi birleştiren yöntemimiz daha yüksek veri iletim hızı sağlamıştır. Bunun asıl nedeni DSYS ile Wi-Fi haberleşme yöntemlerinin farklı kanalları kullanmasıdır.

Bundan sonraki ölçümümüzde ise uçtan uca bağlantı gecikmesini Şekil-5'te incelemekteyiz.

Burada ilk göze çarpan değişiklik düşük uzaklıkta birleşik yöntemin diğer yöntemlere kıyasla daha yüksek bir gecikmeye neden olmasıdır. Bunun asıl nedeni gönderilen paketleri ikiye bölüp tekrar birleştirmek için harcanan fazladan zamandır. Bu gecikme uzaklık arttıkça ve bağlantı kalitesi düştükçe, asıl iletim gelişmesine kıyasla daha az bir etkiye neden olmaktadır ve Wi-Fi yönteminden daha iyi bir sonuç vermektedir. Son olarak YTA kullanarak farklı veri akışlarını farklı haberleşme kuralları ile göndererek iki akış için de daha az bir gecikme



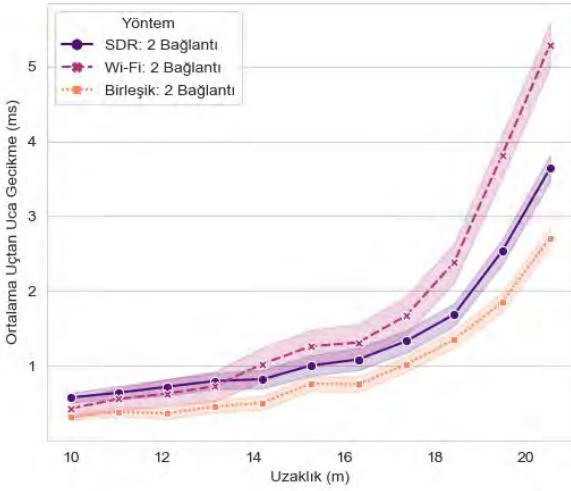
Şekil-4: Artan uzaklığa karşılık iletim hızının değişimi.



Şekil-5: Artan uzaklığa karşılık uçtan uca gecikme değişimi.

sağladı. Şekil-6'da bu deneyin sonuçları mevcuttur. Burada, farklı veri akışları için aynı haberleşme kaynağını kullanan yöntemler ortalama olarak daha fazla uçtan uca gecikmeye neden olmuşlardır.

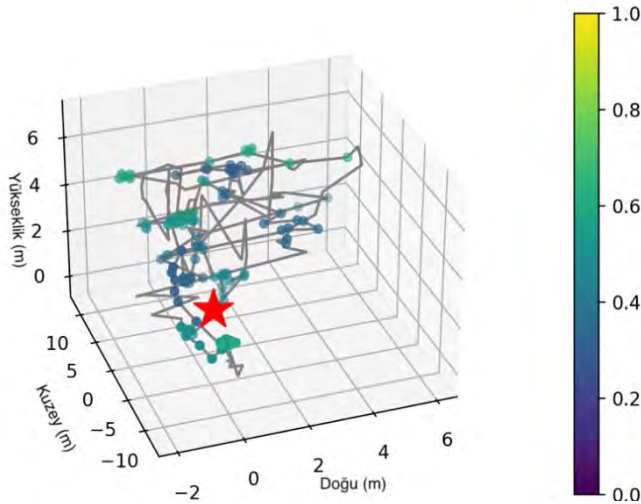
Öte yandan YTA denetimi sayesinde yerdeki görev denetçisi farklı akışlar için farklı iletişim kanalı numarası tanımlayıp bir akışı Wi-Fi ile, öteki akışı ise YTR ile göndermiştir. Bu sayede iki akış koşut olarak iletilmiş ve sonuç olarak ortalama gecikme de azalmıştır. Deney sonuçlarımız, İHA'ların birbirine olan mesafelerinin arttırılması durumunda bile, yazılım-tanımlı radyo ve Wi-Fi teknolojilerinin birleşik kullanımı ile iletişim istikrarının ve hızının korunabildiğini göstermektedir. Özellikle, düşük enerji tüketimi ve yüksek menzil özellikleriyle dikkat çeken LoRa teknolojisinin entegrasyonu, İHA'ların daha geniş alanlarda etkili bir şekilde görev yapabilmelerine olanak tanımıştır. Ayrıca, YTR ve Wi-Fi teknolojilerinin birleşik kullanımı, veri iletim hızını optimize ederken, uçtan uca gecikme sürelerinin minimumda tutulmasını sağlamıştır.



Şekil-6: Artan uzaklığa karşılık farklı veri akışlarının ortalama uçtan uca gecikme değişimi.

## 4.2 Erişim Noktası ile Bağlantının Konuma göre Değişimi

Son olarak, yapılan uçuşlarda GPS yardımı ile konumun değişimine göre bir Wi-Fi erişim noktasına olan bağlantının kalitesi YTR kullanılarak Şekil-7'de gösterilmiştir. Bu şekilde, yıldız erişim noktasının konumunu gösterirken İHA'nın hareket çizgisi üzerindeki noktalar ise işaret gücünün ölçümlerini göstermektedir. Şekildeki konumlar metre cinsindedir ve erişim noktası tarafımızdan yerleştirildiği için konumu bizim tarafımızdan belirlenmiştir.



Şekil-7: Artan uzaklığa karşılık farklı veri akışlarının ortalama uçtan uca gecikme değişimi

YTA denetimi ile gerçekleştirilen dinamik iletişim yöntemi sayesinde, farklı veri akışları için uygun haberleşme kurallarının atanması, kaynak kullanımının verimliliğini artırarak, sistem performansını iyileştirmiştir. İHA'ların farklı Wi-Fi erişim noktalarına olan bağlantı kalitesinin konuma göre değişimi analizi, gerçek zamanlı ortamlarda İHA'ların adaptasyon yeteneğini ve iletişim güvenilirliğini kanıtlar

niteliktedir. Bu kapsamlı değerlendirme, önerilen deney ortamının, karmaşık gerçek dünya senaryolarında karşılaşılabilecek iletişim zorluklarını üstesinden gelebilecek esnekliğe ve güvenilirliğe sahip olduğunu göstermektedir.

## 4.3 Önerilen Yaklaşımın Kullanılabilirlik Gözlemleri

Bu bölümde, önerilen İHA tabanlı esnek deney ortamının kullanılabilirliğini ve kullanıcı deneyimini değerlendirmek amacıyla gerçekleştirilen kullanılabilirlik testlerinin sonuçları sunulmaktadır. Çalışma kapsamında, çeşitli kullanıcı gruplarına yönelik olarak tasarlanan senaryolar üzerinden ölçüm tamamlama süreleri, Sistem Kullanılabilirlik Ölçeği (İngilizce: System Usability Scale (SUS)) puanları ve kullanıcı memnuniyet derecelerini içeren bir dizi test gerçekleştirilmiştir. Çizelge-2, test sürecinde elde edilen sonuçları özetlemektedir.

Çizelge-2: Kullanılabilirlik Sonuçları

Kullanıcı Grubu	Ortalama Ölçüm Tamamlama Süresi	SUS Puanı (0-100)	Memnuniyet Derecesi (1-5)
Yeni Başlayanlar (5 kişi)	41dk	81	4
Orta Seviye Kullanıcılar (5 kişi)	28dk	85	5
İleri Düzey Kullanıcılar (5 kişi)	15dk	88	5

Kullanıcı grupları, araştırma laboratuvarında çalışan kişiler arasından farklı tecrübe seviyelerine göre gruplama yapılarak oluşturulmuştur. Ortalama ölçüm tamamlama süresi, önerilen yaklaşım kullanıldığında 15 dakikalık bir uçuş görevi sonunda açığa çıkan ölçüm verilerinin başarıyla toplanması için harcanan süreyi göstermektedir. SUS puanı ise kullanıcıların bir ürün, sistem veya servisin kullanılabilirliğini değerlendirmek için verdiği yanıtlara dayanan, 1'den 5'e kadar bir Likert ölçeği üzerinden puanlama yapan bir anket yöntemidir [26]. SUS anketi, toplam 10 ifadeden oluşmaktadır ve bu ifadeler, sistemin çeşitli kullanılabilirlik yönlerini yansıtır. Bu ifadelerin yarısı olumlu, diğer yarısı ise olumsuz şekilde formüle edilmiştir. Ankette 1 'Kesinlikle Katılmıyorum' ve 5 'Kesinlikle Katılıyorum' anlamına gelmektedir. Puanlama elde edilirken olumlu ifadeler verilen yanıtlardan 1 çıkarılırken, olumsuz ifadelerin puanları 5'ten çıkarılıp ters çevrilir. Bu işlemlerden sonra, elde edilen değerlerin toplamı 2,5 ile çarpılarak 0 ile 100 arasında bir SUS puanı elde edilir [29]. Memnuniyet derecesi de 1 'Kesinlikle Katılmıyorum' ve 5 'Kesinlikle Katılıyorum' ölçeği üzerinden değerlendirilmiştir.

Sonuç olarak, gerçekleştirilen kullanılabilirlik testleri ve elde edilen kullanıcı geri bildirimleri, önerilen İHA tabanlı esnek deney ortamının, pratik kullanımlarda yüksek performans ve kullanıcı memnuniyeti sağladığını göstermektedir. Bu sonuçlar, sistemimizin ileriye dönük geliştirmeler için sağlam bir temel oluşturduğunu ve kullanıcı ihtiyaçlarına yönelik etkili çözümler sunabilecek potansiyele sahip olduğunu vurgulamaktadır.

## 5. Sonuç

Bu çalışmada İHA tabanlı esnek deney ortamımızı tanıttık. Deney ortamımız, Wi-Fi ve LoRa haberleşme teknolojilerinin yanı sıra YTR de kullanarak esnek haberleşme altyapısına sahiptir. Ayrıca, YTA desteği ile birbirlerinde farklı haberleşme teknolojileri aynı ağ içinde dinamik olarak kullanılabilir. Dahası, farklı akışlara farklı haberleşme teknolojileri atanarak koştur veri akışına imkân verilir ve daha yüksek bir ağ kalitesi sağlanmıştır. Ayrıca YTR yardımı ile etrafta var olan haberleşme ağlarının ölçümlerini incelemek mümkündür.

Bu çalışmada tanıtılan yazılım-tanımlı İHA ağı deney ortamı, mevcut alternatif çözümlerle karşılaştırıldığında, YTR ve LoRa teknolojilerinin entegrasyonunun, İHA ağlarında önemli performans iyileştirmeleri sağladığını ortaya koymuştur. Özellikle iletişim istikrarlılığı, hızı, güvenilirliği ve kaynak kullanımı gibi kritik metrikler üzerinden önerilen çözüm vurgulanmıştır. Ayrıca, önerilen YTA yaklaşımının dinamik iletişim yönetimi kapasitesi, esnek ve ölçeklenebilir bir sistem tasarımı sunarak, mevcut statik iletişim ağlarına kıyasla önemli bir avantaj sağlamıştır. Bununla birlikte, çalışmamız, yazılım-tanımlı teknolojilerin karmaşıklığı ve yüksek başlangıç maliyeti gibi faktörler nedeniyle mevcut çözümlere kıyasla dezavantaj da göstermektedir. Bu da geniş çaplı uygulamalarda dikkate alınması gereken önemli hususlardır.

Sonuç olarak, önerilen çözüm, İHA ağlarının yönetiminde esneklik, genişletilebilirlik ve dinamik adaptasyon sağlama konusunda önemli bir adım olarak değerlendirilmekte; ancak güvenlik ve gizlilik gibi kritik konularda daha derinlemesine çalışmalar gerekmektedir. Gelecek çalışmalar, bu alandaki eksiklikleri gidermeye yönelik olacak ve İHA tabanlı iletişim ortamlarının güvenlik ve gizlilik performansını artıracak yöntemler üzerine odaklanacaktır.

## Kaynakça

- [1] Fan, B., Li, Y., Zhang, R., & Fu, Q., *Review on the technological development and application of UAV systems*. Chinese Journal of Electronics, 29(2), 2020, 199-207.
- [2] Bertizzolo, L., D'oro, S., Ferranti, L., Bonati, L., Demirors, E., Guan, Z., Melodia, T., & Pudlewski, S., *Swarmcontrol: An automated distributed control framework for self-optimizing drone networks*, 2020, <https://arxiv.org/abs/2005.09781>
- [3] Çoğay, S., Sarı, T. T. & Seçinti, G. *SoNaR: Software-defined Network and Radio Framework for FANETs*, IEEE International Symposium on Dynamic Spectrum Access Networks (DySPAN), 2021, pp. 268-273.
- [4] Paredes, W.D., Kaushal, H., Vakili, I., Prodanoff, Z. *LoRa Technology in Flying Ad Hoc Networks: A Survey of Challenges and Open Issues*, Sensors 2023, 23, 2403. <https://doi.org/10.3390/s23052403>
- [5] Sobot, S., et al., *Two-Tier UAV-based Low Power Wide Area Networks: A Testbed and Experimentation Study*, 2023 6th Conference on Cloud and Internet of Things (CIoT), Lisbon, Portugal, 2023, pp. 85-90, doi: 10.1109/CIoT57267.2023.10084912.
- [6] De Rango, F., & Stumpo, D., *Supporting Path Planning in LoRa-based UAVs for dynamic Coverage for IoT devices*, 2023 IEEE 20th Consumer Communications & Networking Conference

- (CCNC), Las Vegas, NV, USA, 2023, pp. 337-340, doi: 10.1109/CCNC51644.2023.10060525.
- [7] Cheng, H., Bertizzolo, L., D'oro, S., Buczek, J., Melodia, T., & Bentley, E.S., "Learning to Fly: A Distributed Deep Reinforcement Learning Framework for Software-Defined UAV Network Control," in IEEE Open Journal of the Communications Society, vol. 2, pp. 1486-1504, 2021, doi: 10.1109/OJCOMS.2021.3092690.
- [8] Mohanti, S., et al, "AirBeam: Experimental Demonstration of Distributed Beamforming by a Swarm of UAVs," 2019 IEEE 16th International Conference on Mobile Ad Hoc and Sensor Systems (MASS), Monterey, CA, USA, 2019, pp. 162-170, doi: 10.1109/MASS.2019.00028.
- [9] Aydın, E. E., Kara, O., Cakir, F., Cansiz, B. S., Secinti, G., & Canberk, B., *Enabling Self-Organizing TDMA Scheduling for Aerial Swarms*. In Proceedings of the Eighth Workshop on Micro Aerial Vehicle Networks, Systems, and Applications, 2022, July, pp. 13-18.
- [10] Shukla, R.M., Sengupta, S., & Patra, A.N., "Software-defined network based resource allocation in distributed servers for unmanned aerial vehicles," in Proc. IEEE CCWC, 2018.
- [11] Zhao, Z., et al., "Software-defined unmanned aerial vehicles networking for video dissemination services," Ad Hoc Netw., vol. 83, 2019, pp. 68-77, [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S1570870518306231>
- [12] Xiong, F., Li, A., Wang, H., & Tang, L., "An SDN-MQTT Based Communication System for Battlefield UAV Swarms," in IEEE Communications Magazine, vol. 57, no. 8, August 2019, pp. 41-47, doi: 10.1109/MCOM.2019.1900291.
- [13] Shurrab, M., Mizouni, R., Singh, S., & Otrok, H., *Reinforcement learning framework for UAV-based target localization applications*. Internet of Things, 23, 100867, 2023.
- [14] Choi, H.-H., Oh, J., Kang, K.-M., & Lee, H., "Idle-Less Slotted ALOHA Protocol for Drone Swarm Identification," in IEEE Transactions on Vehicular Technology, vol. 72, no. 8, Aug. 2023, pp. 11080-11085, doi: 10.1109/TVT.2023.3261104.
- [15] Chang, H., Chen, Y., Zhang, B., & Doermann, D., "Multi-uav mobile edge computing and path planning platform based on reinforcement learning", IEEE Transactions on Emerging Topics in Computational Intelligence, 2021.
- [16] Liu, X., Lam, K., Alkouz, B., Shahzaad, B., & Bouguettaya, A., "Constraint-based Formation of Drone Swarms," 2022 IEEE International Conference on Pervasive Computing and Communications Workshops and other Affiliated Events (PerCom Workshops), Pisa, Italy, 2022, pp. 73-75, doi: 10.1109/PerComWorkshops53856.2022.9767410.
- [17] Song, Q., Zeng, Y., Xu, J. et al., *A survey of prototype and experiment for UAV communications*. Sci. China Inf. Sci. 64, 140301, 2021. <https://doi.org/10.1007/s11432-020-3030-2>
- [18] Douklias, A., Karagiannidis, L., Misichroni, F., & Amditis, A., Design and implementation of a UAV-based airborne computing platform for computer vision and machine learning applications, Sensors, 22(5), 2049, 2022.
- [19] Shi, Y., Wensowitch, J., Ward, A., Badi, M., & Camp, J., "Building UAV-Based Testbeds for Autonomous Mobility and Beamforming Experimentation," 2018 IEEE International Conference on Sensing, Communication and Networking (SECON Workshops), Hong Kong, China, 2018, pp. 1-5, doi: 10.1109/SECONW.2018.8396345.
- [20] Sommer, D., Irigireddy, A.S.C.R., Parkhurst, J., & Nastrucci, E.-R., "SDR- and UAV-Based Wireless Avionics Intra-Communication

- Testbed*," 2020 AIAA/IEEE 39th Digital Avionics Systems Conference (DASC), San Antonio, TX, USA, 2020, pp. 1-5, doi: 10.1109/DASC50938.2020.9256639.
- [21] Enhos, K., Unal, D., Turco, J., Demirors, E., & Melodia, T., *Marena: Sdr-based testbed for underwater wireless communication and networking research*. In Proceedings of The 17th ACM Workshop on Wireless Network Testbeds, Experimental evaluation & Characterization, 2023, October, pp. 80-87.
- [22] Baumgärtner, L., Bauer, M., & Bloessl, B., *SUN: A Simulated UAV Network Testbed with Hardware-in-the-Loop SDR Support*. In 2023 IEEE Wireless Communications and Networking Conference (WCNC), 2023, March, pp. 1-6.
- [23] Lahoud, C., Ehsanfar, S., Gabriel, M., Küffner, P., & Mößner, K., *Experimental Testbed Results on LTE/5G-V2I Communication using Software Defined Radio*. In ICC 2022-IEEE International Conference on Communications, 2022, May, pp. 2894-2899.
- [24] McKeown N., Anderson T., Balakrishnan H., Parulkar G., Peterson L., Rexford J., Shenker S. & Turner J.. 2008. OpenFlow: enabling innovation in campus networks. SIGCOMM Comput. Commun. Rev. 38, 2, April 2008, 69–74. <https://doi.org/10.1145/1355734.1355746>
- [25] Secinti G., Trotta A., Mohanti S., Di Felice M. & Chowdhury K. R., "FOCUS: Fog Computing in UAS Software-Defined Mesh Networks," in IEEE Transactions on Intelligent Transportation Systems, vol. 21, no. 6, pp. 2664-2674, June 2020, doi: 10.1109/TITS.2019.2960305.
- [26] Sharma, V., Song, F., You, I., & Chao, H. C. (2017). Efficient management and fast handovers in software defined wireless networks using UAVs. IEEE Network, 31(6), 78-85.
- [27] Oubbati, O. S., Atiquzzaman, M., Ahanger, T. A., & Ibrahim, A. (2020). Softwarization of UAV networks: A survey of applications and future trends. IEEE Access, 8, 98073-98125.
- [28] Brooke, J. (2013). SUS: a retrospective. Journal of usability studies, 8(2), 29-40.



# Güncel Düzmece Video Algılama Yöntemleri Üzerine Bir Kaynak Derlemesi

## A Literature Review on Current Deepfake Video Detection Methods

Suzan AYDIN  
Haliç Üniversitesi  
İşletme Fakültesi  
İstanbul, Türkiye  
demirsuzann@gmail.com  
ORCID: 0009-0005-7337-1851

Zeki ÖZEN  
İstanbul Üniversitesi  
İktisat Fakültesi  
İstanbul, Türkiye  
zekiozen@istanbul.edu.tr  
ORCID: 0000-0001-9298-3371

### Öz

Son yıllarda yapay zekâ ve derin öğrenme teknolojilerindeki hızlı gelişmeler, düzmece (Deepfake) gibi yeni ve yenilikçi uygulamaların ortaya çıkmasını sağlamıştır. Düzmece görsel ve işitsel içeriklerin düzenlenmesine olanak tanır ve özellikle bireylerin görüntü ve seslerini taklit etmek için kullanılır. Düzmece teknolojisi sağladığı olanak ve avantajların yanında kişisel bilginin güvenliği, mahremiyeti ve oluşturulan içeriklerin güvenilirliği gibi konularda ciddi endişelere yol açmaktadır. Bu endişeler, Düzmece içeriklerinin algılanması ve doğrulanması amacıyla yapılan araştırmalara ivme kazandırmıştır. Bu kaynak incelemesi, düzmece türlerini, düzmece video içerikleri algılayan algoritmaların eğitiminde kullanılan veri kümelerini ve güncel düzmece video algılama yöntemlerini ele almaktadır.

**Anahtar sözcükler:** Düzmece, Düzmece Türleri, Düzmece Görselleri Algılama Yöntemleri, Derin Öğrenme, Yapay Zekâ

### Abstract

"In recent years, rapid advancements in artificial intelligence and deep learning technologies have led to the emergence of new and innovative applications such as Deepfake. Deepfake allows for the manipulation of visual and auditory content and is particularly used to imitate individuals' images and voices. Alongside the possibilities and advantages provided by Deepfake technology, it raises serious concerns regarding the security of personal information, privacy, and the reliability of the created content. These concerns have accelerated

research aimed at the perception and verification of Deepfake content. This literature review addresses types of Deepfake, datasets used in training algorithms that detect Deepfake video content, and current methods for detecting Deepfake videos.

**Keywords:** Deepfake, Deepfake Types, Deepfake Video Detection Methods, Deep Learning, Artificial Intelligence

### 1. Giriş

2024 itibariyle dünya genelinde 5,35 milyar internet kullanıcısı bulunmaktadır [1]. Statista raporuna göre son bir yılda 97 milyon yeni kullanıcı sayesinde %1,8 oranındaki artış, internet kullanımının küresel ölçekte yayılmaya devam ettiğini göstermektedir. Bu devasa kullanıcı kitlesinin sosyal medya etkileşimleri ve sayısal içerik sunan platformlardaki paylaşımları kişisel verilerin kötü niyetli kullanımı veya sızdırılması gibi riskleri de beraberinde getirmektedir. ITRC (Identity Theft Resource Center) 2023 Veri İhlali Raporu'na göre 2023 yılında sadece Amerika Birleşik Devletleri'nde 353 milyondan fazla internet kullanıcısı veri ihlalden etkilenmiştir. Yine aynı rapora göre bu oran 2022 yılına kıyasla %16 artmıştır. Veri ihlali olayı sayısındaki bu artış bilişim güvenlik tehditlerinin ne kadar kritik bir duruma ulaştığını göstermektedir [2]. Özellikle veri ihlalleri sonucunda elde edilen kişisel bilgiler, daha gelişkin saldırılar için kullanılabilir. Bu bağlamda, veri güvenliği endişeleri arasına yeni nesil bir tehdit olarak ön plana çıkan "düzmece" teknolojisinin giderek artan kullanımı girmiştir. Düzmece sistemleri, insanların gerçeğinden ayırt edemediği sahte görüntüler oluşturabilir [3]. Veri ihlalleri sonucu sızdırılan kişisel veriler, özellikle ses ve görüntü kayıtları, düzmece içeriklerin oluşturulmasında

kullanılabilir. Bu durum, sadece bireylerin kimliklerinin kötüye kullanılmasına yol açmakla kalmaz, aynı zamanda gerçek olmayan bilginin yayılmasına ve toplumda güven kaybına neden olabilir. Dolayısıyla düzmece içeriklerin üretimi ve tespit edilmesi, teknikleri, yöntemleri ve mekanizmaları hakkında kapsamlı bir anlayış gerektirir [4].

## 2. Düzmece Tanımı ve Zararları

Düzmece (Deepfake), yapay zekâ ve derin öğrenme algoritmalarını kullanarak ses ve görüntü içeriklerini isteğe göre değiştirme teknolojisidir [5]. Bu teknolojinin özellikle sosyal medya platformlarında video, görüntü ve ses içerikleri üzerinde kullanılması siber zorbalık ve oynama riskini artırmaktadır. Bu teknoloji kullanılarak dolandırıcılar tarafından üretilen sahte videolar, toplumu kolayca aldatabilmekte ve endişeye yol açmaktadır [6, 7]. Bu sebeple düzmece teknolojisiyle oluşturulmuş içerikleri etkili bir şekilde algılayabilen yöntemlere duyulan artan ihtiyaç artmaktadır [8].

Sayısal medya ve platformlarda düzmece teknolojisinin ortaya çıkışı çift taraflı bir kılıç olarak belirmiştir [9]. Eğlence ve iletişimde yenilikler getirmiş olmasına karşın düzmece teknolojisinin kötüye kullanım potansiyeli önemli endişelere yol açmıştır. 2017 yılında 'deepfakes' adlı bir Reddit kullanıcısı tarafından bir ünlünün düzmece yöntemi ile oluşturulan pornografik videosu, bu teknolojinin kötüye kullanılmasının kaçınılmaz olduğunu gösteren ilk örnek olmuştur ve düzmece içerikleri algılama çalışmalarını tetiklemiştir [10]. Gerçek ile değiştirilmiş içerikleri ayırt etmede makine öğrenmesi ve derin öğrenme yöntemleri kullanılmaktadır. Düzmece teknolojileri ve özellikle düzmece video algılama yöntemleri hakkında bir kaynak taraması sunan bu araştırma bu alanda çalışan araştırmacılara ve siber güvenlik önlemlerinin sürekli gelişimine katkıda bulunmayı amaçlamaktadır.

## 3. Düzmece Türleri ve Veri Setleri

Düzmece içerikler; görüntüler ve video kliplerinin birleştirme, harmanlama, değiştirme ve üst üste bindirme gibi çeşitli teknikler kullanarak oluşturulur [10]. Bu klasik yöntemlerin dışında, ilk kez 2014'te Goodfellow'un tanıttığı Çekişmeli Üretken Ağlar (*Generative Adversarial Networks - GAN*) düzmece üretimini bir başka seviyeye taşımıştır. GAN, rekabet eden iki sinir ağından oluşur: üretici ağ rastgele gürültüden gerçekçi görünen veriler üretmeye çalışırken, ayırt edici ağ gerçek veriler ile üreticinin oluşturduğu sahte verileri ayırt etmeye odaklanır. GAN eğitim sürecinde, üretici daha inandırıcı veriler üretmeyi öğrenirken, ayırt edici de bu verileri gerçeklerden ayırt etme becerisini geliştirir. Bu karşılıklı rekabet, her iki ağın da gelişimini teşvik eder ve üreticinin giderek daha gerçekçi veriler oluşturmasını sağlar [11].

Düzmece içeriklerinin yaygınlığı, karmaşık ve gerçeğe çok yakın sonuçlar üreten birçok gelişmiş GAN tabanlı tekniğin gelişmesine olanak sağlamıştır. AttGAN [12], özellik farkındalığı, hassas ve yüksek kaliteli özellik değişikliklerine olanak sağlar ve yüz özelliklerini değiştirerek yüz değiştirme, yaşlandırma ya da gençleştirme gibi uygulamalarda etkilidir. Benzer şekilde StyleGAN [13], değişik yüz özelliklerinin değiştirilmesine izin vererek gerçeğe çok yakın ve ayrıntılı

görüntüler üretmede başarılı olan bir GAN mimarisidir. Aynı şekilde STGAN [14], özellik değişikliklerini doğru bir şekilde kontrol ederek kişinin sadece belirli yüz özelliklerini değiştirebilen etiketli ve etiketsiz verilerle çalışabilen bir araçtır. Başka bir teknik olan StarGANv2 [15], birçok görevi ayrı ayrı eğitmek yerine tek bir büyük modelle tüm bu dönüşümleri sağlayan yeteneğiyle aynı modelle hem saç rengini değiştirebilir hem kişiyi güldürebilir hem de canlandırma karakterini gerçek bir aktörmüş gibi gösterebilir.

Düzmece teknolojisinin görsel, sese ve hatta metin tabanlı uygulamaları bulunmaktadır. Metin tabanlı düzmece örnekleri, sahte veya yanıltıcı içerik oluşturan GAN veya Büyük Dil Modelleri yardımıyla üretilen metinleri ifade eder. Toplum ve insanları yanıltma amacıyla yayınlanan bu içerikler yanıltıcı bilgilendirmenin bir türüdür [16]. Ses düzmece örnekleri, belirli bir kişinin ses kayıtları üzerinde derin öğrenme modelleri kullanarak ses kopyalama ve ses değiştirme olanağı sağlamaktadır. Düzmece resimler bireylerin görünüşünü değiştirirken, düzmece videolar bireylerin görünüşünü veya eylemlerini değiştiren gerçekçi videolar yaratmak için en sık kullanılan medya formatıdır. Salman ve ark. [4]'e göre düzmece video oluşturulmasında dört ana tip bulunmaktadır. Birinci tip, Gerçek Sesli Yüz Değişimi (Fake Video/Face-Swapped With Real Audio - Type I), bir videodaki A Kişisinin yüzünü B Kişisinin yüzüyle değiştirmeyi ve A Kişisinin sesini korumayı hedefler. İkinci tip, Sentetik Konuşmalı Gerçek Video (Real Video With Synthetic Speech - Type II), videodaki kişinin yüzünü korurken sesini değiştirmeyi amaçlar. Üçüncü tip olan Ses Kopyalanmış Gerçek Video (Real Video With Voice-Cloned - Type III), hedef kişinin gerçek sesini taklit ederek söylemediği bir şeyi söylemiş gibi göstermeyi amaçlar. Dördüncü ve son tip, Sahte Sesli Sahte Video (Fake Audio With Fake Video - Type IV), hem yüzün hem de sesin tamamen oynanmış olduğu durumlardır [4].

Düzmece teknolojisi, farklı medya formatlarında çeşitli uygulamaları kapsadığı için her bir medya formatının benzersiz zorlukları bulunmaktadır [18]. Ancak sosyal medya ve sayısal ortamlarda özellikle video içerikler daha çok tüketildiği için bu kaynak incelemesinde düzmece video algılama çalışmaları değerlendirilecektir.

Tüm ortam formatlarındaki düzmece içeriklerin algılanabilmesini ve gerçek içeriklerden ayırt edilebilmesini sağlayan en önemli etken veri kümelerinin büyüklüğü ve verimliliğidir [17]. Düzmece içerik algılayan modellerin eğitim ve sinama aşamalarında kullanılan açık kaynak güncel veri kümelerinin bazıları şunlardır:

UADFV, YouTube'dan alınmış 49 özgün video ve bunların değiştirilmesiyle oluşturulan 49 video ile toplamda 98 videodan oluşan düzmece tespiti için ilk yayınlanmış veri kümesidir [19].

FaceForensics++ (FF++) veri kümesi, FaceForensics veri kümesinin bir uzantısı olarak araştırmacıların denetimli bir şekilde derin öğrenme yaklaşımları geliştirmelerine olanak tanıyan Face2Face, FaceSwap, Deepfakes ve NeuralTextures gibi dört alt veri kümesi içeren YouTube videoları kullanılarak oluşturulmuştur [20].

DeepfakeTIMIT veri kümesi, 32 öznenin toplam 620 videosunu içerir. Her konu için iki farklı kalitede 20 düzmece videosu bulunmaktadır; bunların 10 tanesi düşük kalite düzmece video içeren Deepfake-TIMIT-LQ alt veri kümesine, kalan 10 tanesi ise yüksek kalite düzmece video içeren Deepfake-TIMIT-HQ alt veri kümelerine aittir [21].

Facebook tarafından oluşturulan DFDC (*Deepfake Detection Challenge*) veri kümesi 100.000'den fazla video içerir ve Kaggle yarışmalarında kullanılmak üzere yayınlanmıştır. DFDC-preview veri kümesinin bir uzantısıdır [22].

Deepfake Detection (DFD) veri kümesi, 363 oyuncu kullanılarak çekilmiş 3068 sahte ve 363 gerçek video içeren düzmece tespit yöntemlerinin geliştirilmesine destek olmak amacıyla Google tarafından yayınlanmıştır [23].

Celeb-DF, YouTube'dan derlenen çeşitli yaş, etnik grup ve cinsiyetlerden kişilere ait gerçek videolar ve iki milyondan fazla kareye ilişkin bir sentez süreci kullanılarak oluşturulmuştur [24].

VoxCeleb2 veri kümesi, YouTube'a yüklenen videolardan çıkarılan, 6.000'den fazla ünlüye ait değişik görsel ve işitsel ortamlarda çekilmiş 150.480 video ve 1 milyondan fazla konuşma içermektedir [25].

FakeAVCeleb, VoxCeleb2 veri kümesinden çıkarılan 500 gerçek videonun çeşitli düzmece üretim yöntemleri ile oynanmış 20.000 düzmece video içeren veri kümesidir [26]. Aşağıda verilen Çizelge-1'de literatürde düzmece video tespit algoritmalarının sıklıkla kullandığı bazı veri kümeleri özet olarak verilmiştir [27, 28].

**Çizelge-1: Düzmece Video Tespit Çalışmalarında Kullanılan Bazı Veri Setleri**

Veri Kümesi	Yıl	İçerik	Toplam	Yöntem	İçerik Türü ve Sayısı	Kaynak
UADFV [19]	2018	Video	98	FakeApp	49 Sahte 49 Gerçek	YouTube
FaceForensics++ (FF++) [20]	2019	Video	4,000	Düzmece Bilgisayar Grafikleri	4,000 Sahte 1,000 Gerçek	YouTube
DeepfakeTIMIT [21]	2018	Video	640	GAN Faceswap	320 Düşük Kalite Sahte 320 Yüksek Kalite Sahte	32 Oyuncu
DFDC (Deepfake Detection Challenge) [22]	2020	Video	120,000	Bilinmiyor	100,000 Sahte 20,000 Gerçek	1,311 Oyuncu
DFDC- preview [22]	2020	Video	5,250	Bilinmiyor	4119 Sahte 1131 Gerçek	1,311 Oyuncu
Celeb-DF [24]	2020	Video	1,203	Düzmece	795 Sahte 408 Gerçek	YouTube
Deepfake Detection (DFD) [23]	2019	Video	3,431	Faceswap	3068 Sahte 363 Gerçek	363 Oyuncu
VoxCeleb2 [25]	2018	Video Ses	150,480 1,128.246	-	Gerçek	YouTube
FakeAVCeleb [26]	2021	Video	20,000	Faceswap FSGAN Wav2Lip	20,000 Sahte	VoxCeleb2
TIMIT-TTS [27]	2022	Ses	20,000	Dynamic Time Warping (DTW)	20,000 Sahte	VidTIMIT
DeeperForensics (DF) [27]	2020	Video	60,000	DF-VAE	10,000 Sahte 50,000 Gerçek	100 Oyuncu
WildDeepfakes (WDF) [27]	2020	Video	7,314	Bilinmiyor	3,509 Sahte 3,805 Gerçek	Bilinmiyor
Presidential Deepfakes Dataset [28]	2020	Video	32	Düzmece	16 Sahte 16 Gerçek	YouTube
World Politicians Deepfake Dataset (WPDD) [28]	2020	Video	135,251	Düzmece Faceswap	31,016 Sahte 104,235 Gerçek	YouTube

#### 4. Düzmece Video Algılama Yöntemleri

Rana ve ark. [29], düzmece algılama çalışmalarına yönelik yaptıkları kapsamlı literatür taramasında düzmece algılama

çalışmalarını Derin Öğrenme (*Deep Learning - DL*), Makine Öğrenimi (*Machine Learning - ML*), istatistiksel teknikler ve öbek zinciri (blockchain) tabanlı yöntemler olarak dört kategori altında değerlendirmiştir. Bu dört kategoriye insan

odaklı düzmece algılama çalışmaları [30, 31, 32] da dahil edilerek düzmece algılama yöntemleri beş ulam olarak sınıflandırılacaktır.

Ancak düzmece algılama çalışmalarında baskın ulam olan derin öğrenme tabanlı yöntemler, bu çalışmanın özellikle yöneldiği alan olmuştur. Derin öğrenme, makine öğrenmesinin bir alt dalı olarak verilerden özellik çıkarma ve dönüştürme işlemlerini gerçekleştirmek için çok katmanlı doğrusal olmayan işlem birimlerini kullanır. Bu katmanlar, birbirini takip eden şekilde düzenlenmiş olup, her bir katman önceki katmanın çıktısını girdi olarak alır [33]. Düzmece içerikleri algılamak için kullanılan üç derin öğrenme yaklaşımı dikkat çekmektedir: Evrişimli Sinir Ağı (*Convolutional Neural Networks - CNN*), Tekrarlayan Sinir Ağları (*Recurrent Neural Networks -RNN*) ve Transformatör (*Transformer*) modelleri çalışmalarda yoğunlukla yer almakta ve yüksek performans

sunmaktadır. Çizelge-2 çoğunlukla kullanılmakta olan CNN mimarilerini içermektedir.

Düzmece içerik algılama yöntemleri arasında uzamsal kalıntı (*spatial artefacts*) analizleri, biyolojik ve fizyolojik göstergelerin değerlendirilmesi, ses ve görüntü verileri arasındaki uyumsuzlukların incelenmesi, evrişimsel izlerin saptanması, kimlik doğrulama bilgilerinin analizi, zaman içindeki tutarsızlıkların izlenmesi, yüz ifadelerinin detaylı incelenmesi ve uzamsal-zamansal özelliklerin değerlendirilmesi yer almaktadır [34]. Çalışmanın devamında burada sayılan düzmece video algılama yöntemleri tek tek ele alınmış ve her bölümün sonunda ilgili yöntemle yapılan çalışmaları özetleyen tablolar verilmiştir. Bu çizelgelerde düzmece algılama çalışmalarında en çok kullanılan AUC (*Area Under Curve - Eğri Altı Alan*) ve ACC (*Accuracy – Doğruluk*) metriklerinden elde edilen en yüksek performans değeri dikkate alınmıştır.

Çizelge-2: CNN Mimarileri

Model	Yıl	Özellikler	Avantajlar	Dezavantajlar	Kullanım Alanları
AlexNet [35]	2012	Basit ve hızlı Katman: 8	Görüntü sınıflandırmada başarılı	Daha az doğruluk	Görüntü sınıflandırma, nesne algılama
VGGNet [36]	2014	Daha derin ve daha doğru Katman: 16/19	AlexNet [35]'e göre daha yüksek doğruluk	Daha fazla parametre ve işlem gücü	Görüntü sınıflandırma, nesne algılama
Inception [37]	2014	Daha az parametre ile yüksek doğruluk Katman: 22	Farklı boyutlarda filtreler kullanır	Hesaplama açısından daha karmaşık	Görüntü sınıflandırma, nesne algılama, görüntüyü metne çevirme
ResNet [38]	2015	En yüksek doğruluk Katman: 0/101/152	Kalan bağlantıları kullanır	Daha fazla parametre	Görüntü sınıflandırma, nesne algılama, insan pozlama tahmini
DenseNet [39]	2016	Yoğun bağlantı modeli	Daha az parametre ile yüksek doğruluk	Eğitmek daha zor	Görüntü sınıflandırma, nesne algılama, görüntü segmentasyonu
MobileNet [40]	2017	Mobil cihazlar için optimize edilmiş	Daha az parametre ve işlem gücü	Daha az doğruluk	Görüntü sınıflandırma, nesne algılama, yüz tanıma
EfficientNet [41]	2019	Otomatik model arama ve seçme	En yüksek doğruluk ve verimlilik dengesini sunar	Karmaşık mimari	Görüntü sınıflandırma, nesne algılama, görüntü segmentasyonu

#### 4.1. Uzamsal Kalıntılara Dayalı Tespit Çalışmaları

Düzmece görüntü üretilince resim veya videoda düzenlemelerin gerçekleşmesiyle sıklıkla uzamsal kalıntılar oluşur. Görüntüdeki sıkıştırma izleri, blokların bozulmaları, tekrar eden örüntü desenleri veya benekleşme gibi görsel bozulmaların varlığı videonun düzmece olduğunun anlaşılmasına dair ipuçları verir.

Afchar ve ark. [42] görüntülerdeki uzamsal detaylara ve ince farklılıklara odaklanarak düzmece algılama için mezoskopik özellikleri inceleyen MesolInception-4 adlı bir CNN modeli önerdiler. Bu model Inception [37] modüllerinin bir varyasyonunu kullanarak %98,4 doğruluk oranı ile ön plana çıkmıştır. Bu çalışma, düzmece algılamada derin ağların etkinliğini göstererek bu alanın genişlemesine katkıda bulunmuştur.

Bir yüz görüntüsü değiştirdikten sonra kafaya yerleştirildiğinde yüz ve baş açısının tutarsızlığına dikkat çeken Yang ve ark. [19] yüzdeki iki boyutta (2D) işaretlerden başın üç boyutlu (3D) pozisyonunu tahmin ederek sahte içeriği algılamayı hedeflemişlerdir. Çalışmada gerçek içerikle sahte içeriği ayırt etmek için Destek Vektör Makineleri (*Support Vector Machine - SVM*) kullanılmıştır. Bu teknik düzmece algılamada %89 AUC skoru verse de bulanık görüntülerde yüz işaretlerini tahmin etmekte zorlanmıştır.

H. Zhao ve ark. [43] düzmece algılamaya bir sınıflandırma problemi olarak yaklaşarak çoklu dikkat ağı (*Multiaattentional Network*) önerdiler. Bu yöntem, FF++ veri kümesinde %97,60 doğruluk sağlamış olsa da yüksek sıkıştırmaya sahip görüntülerde uzamsal alandaki faydalı bilgilerin çoğu bulanıklaştığı için düşük kaliteli görüntülerde iyi sonuçlar alınamamıştır. Aynı yıl Kohli ve Gupta [44], düzmece videolarını analiz etmek için frekans odaklı bir CNN

(Frequency Convolutional Neural Network - FCNN) kullanmıştır ve FF++ veri kümesinde %85,24 doğruluk oranı elde etmişlerdir. Luo ve ark. [45] ise mekânsal dikkat ve yüksek-düşük frekans etkileşimlerine odaklanan bir model kullanarak yaptıkları çalışmada %99,5 AUC skoru elde etmişlerdir. İsmail ve ark. [46] ise çeşitli modelleri birleştiren YOLO (You Only Look Once)-InceptionResNetV2-XGBoost (YIX) mimarisini geliştirerek düzmece algılamada %90,73 doğruluk elde etmişlerdir.

Das ve Sebastian [47] düzmece videoları algılamak için video karelerine yönelik bir yöntem geliştirdiler. Bu yöntemde, videodaki yüz algılanıp kırıldıktan sonra her bir video karesinin bir makine öğrenimi algoritmasıyla özellik çıkarımı ve sınıflandırması yapılmaktadır. Her karenin görüntü özellikleri üç farklı CNN modeli kullanılarak elde edilerek Temel Bileşen Analizi (Principal Component Analysis - PCA) ile seçilen ve boyutları küçültülen özellik vektörlerinde birleştirilmektedir. Ardından bir SVM, her kareyi gerçek veya sahte olarak sınıflandırmaktadır. Bu yöntem DFDC alt veri kümesinde %96,50 doğruluk elde etmiştir.

Güncel bir çalışma olarak Dhanaraj ve Sridevi [48] transfer öğrenimi kullanarak videolardaki yüz bozulma bölgelerini tespit eden bir yöntem geliştirdiler. ImageNet [35] veri kümesi ve bir CNN sınıflandırıcı kullanılarak geliştirilen önceden eğitilmiş bir model, bilgisini bu amaca özel yeni bir sınıflandırıcıya aktarmaktadır. İşlenmiş videoya uygulanan Xception [49] CNN sınıflandırıcısı yüz bozulma bölgelerini verimli bir şekilde tespit etmektedir. Önerilen model, bozulmuş yüz bölgelerini %89,25 doğrulukla tespit edebilmektedir.

Çizelge-3'te uzamsal kalıntılara dayalı düzmece algılama çalışmalarının özeti bulunmaktadır.

**Çizelge-3: Uzamsal Kalıntılara Dayalı Düzmece Video Tespit Çalışmaları**

Çalışma	Algoritma/ Önerilen Model	Veri Setleri	Başarım Oranı
[42]	CNN	VidTIMIT, FF++	%98,4 ACC
[19]	SVM	MFC, UADFV	%89 AUC
[43]	Multiattentional Framework	FF++, CelebDF, DFDC	%97,60 ACC
[44]	fCNN	FF++, Celeb-DF(v2)	%85,24 ACC
[45]	CNN	DeeperForensics, FF++, DFD, DFDC, Celeb-DF	%99,5 AUC
[46]	CNN	CelebDF, FF++, WiderFace	%90,73 ACC
[47]	CNN-SVM-PCA	DFDC	%96,50 ACC
[48]	CNN	Celeb-DF, Youtube videoları, ImageNet	%89,25 ACC

## 4.2. Biyolojik/Fizyolojik İşaretlere Dayalı Tespit Çalışmaları

Matern ve ark. [50] düzmece videolardaki göz rengi, yansıma eksiklikleri ve göz ile dişlerdeki bulanıklıklar gibi belirgin hataları incelediler ve bu biyolojik farklılıkların düzmece tespitinde yararlı olabileceğini belirtmişlerdir.

Göz kırpma, göz yüzeyini nemlendirmek ve tozlardan arındırmak için gerekli bir fizyolojik aktivitedir. GAN ile çalışan modellerin eğitim veri kümesi çoğunlukla açık gözlü insan görüntüleri içerdiği için bu modellerin ürettiği sahte yüzler, göz kırpma simülasyonunda genellikle başarısız olmaktadır [51]. Bu nedenle bu modellerin kapalı göz durumunu etkin bir şekilde taklit edemediğini belirten Li ve ark. [51], göz kırpma sayısını ölçmek için LRCN (Uzun Dönemli Yinelemeli CNN - Long-term Recurrent Convolutional Networks) modeli kullanmayı önerdiler. Önce gerçek videolarda göz kırpma oranlarını inceleyerek göz kırpmanın fark edilmesi gereken ortalama süreyi hesaplamışlardır. Ardından, düzmece videolardaki göz kırpma süreleri incelenmiş ve videolarda göz kırpma oranının, sentezlenmiş veya sahte bir videoyu tespit edebildiği sonucuna varılmıştır. Benzer bir yaklaşımda Jung ve ark. [52] göz kırpma düzenlerinin zamansal dinamiklerini analiz etmek için Uzun Kısa Süreli Bellek (Long Short-Term Memory - LSTM) ağlarından yararlanmışlardır. Bu yöntem göz kırpma periyodu, göz kırpma tekrarlama sayısı ve göz kırpma süresini değerlendirerek çoğu düzmece video türünü etkili bir şekilde tespit edebilmiştir. Ancak GAN tabanlı düzmece modelleri, kapalı göz resimleriyle eğitildiklerinde göz kırpma temelli düzmece algılama sistemlerini kandırabilecek gerçekçi göz kırpma hareketleri üretebilmektedir [53].

Kare kare üretilmiş düzmece videolar (özellikle her karenin tek başına manipüle edildiği türler) doğal kalp ritmi değişiminden gelen ipuçları içermemektedir [54]. Qi ark. [55] tarafından geliştirilen DeepRhythm, uzaktan görsel fotopletimografi (PPG) tekniği ile yüz derisindeki minik ve ritmik renk değişimlerini analiz ederek kan akışını ve dolayısıyla kalp atışını ölçmektedir. Gerçek yüzlerde ölçülebilen doğal kalp atış ritminin düzmece videolarda eksik olacağını varsayarak kalp atışı sinyalindeki bozulmaları hem uzamsal hem de zamansal olarak analiz etmişlerdir. Çalışma, yüksek doğruluk oranı ve JPEG bozulması gibi zorluklara karşı dayanıklılık iddialarıyla öne çıkmakta ve kalp atış ritmi bazlı ilk düzmece algılama yöntemi olma özelliğini taşımaktadır. Çiftci ve ark. [56] çalışmasında önerdikleri FakeCatcher yöntemi yüz bölümünden altı farklı biyolojik sinyali mekânsal ve zamansal olarak izleyerek, bu sinyaller arasındaki tutarlılığı gerçeklik belirtisi olarak değerlendirmiştir. Yine fizyolojik sinyallere odaklandıkları bir başka çalışmada Çiftci ve ark. [57] Düzmece algılama yöntemine ek olarak analiz edilen düzmece içeriğin hangi GAN modeliyle üretildiği sorusuna da cevap aramış ve bu amaçla PPG (fotopletimografi) temelli kalp atışı ölçümü yöntemini kullanmışlardır.

Hernandez-Ortega ve ark. [58] DeepfakesON-Phys uzaktan fotopletimografi (rPPG) kullanarak video dizilerindeki ince cilt rengi değişiklikleri aracılığıyla kalp atış hızı bilgilerini analiz

ettikleri çalışmada Celeb-DF ve DFDC veri kümelerinde %98'in üzerinde AUC oranı elde etmişlerdir.

Biyolojik sinyallerdeki farklılıklara odaklanan Wang ve ark. [59] Siamese ağ çerçevesi içinde Xception [49] ağını kullanarak UADFV veri kümesinde %99,94 oranında yüksek bir doğruluk elde etmişler ve oylama mekanizmaları aracılığıyla bu yöntemi daha da iyileştirmişlerdir.

Khurana ve ark. [60], GAN modellerinin her kişiye özel olan kalp atış hızını taklit edemeyeceğine dikkat çekerek düzmece algılamada kalp atış hızı analiz yöntemini kullanmışlardır. Önerilen metodoloji PPG hücreleri veya uzamsal-zamansal hücreler şeklinde bir biyolojik sinyal çıkarıcıyı farklı makine öğrenimi modelleri aracılığıyla işlemeyi içermektedir. MBConv Bloklarını kullanıldığı çalışmada yazarlar ResNet-18 üzerinde %95'e varan bir doğruluk artışı gözlemlemişlerdir.

He ve ark. [54], GazeForensics yöntemi ile özellikle kare kare manipülasyon teknikleriyle yapılmış olan videolardaki göz hareketlerinin doğal akıcılığındaki tutarsızlıkları (uzamsal tutarsızlık) ve göz bölgesindeki biyometrik özelliklerindeki (iris rengi, ışık yansımaları, göz şekli vb.) tutarsızlıkları görüntünün manipüle edildiğine dair işaret kabul etmiş ve yaptıkları çalışmada %99,64 doğruluk elde etmişlerdir.

Son olarak Liang ve ark. [61], düzmece algılama için yüzün değiştirilmiş ve değiştirilmemiş bölgelerinin özelliklerini ayırt etmek için yüz haritalarını ve frekans alanındaki korelasyonları analiz ettiler. Bu yöntemde göz çevresindeki farklılıkların geometrik özellikleri kullanılmıştır. Çalışma, FF++ veri kümesinde %99,6 doğruluk yakalamıştır.

Çizelge-4'te biyolojik ve fizyolojik işaretlere dayalı düzmece video algılama çalışmalarının özeti verilmiştir.

**Çizelge-4: Biyolojik/Fizyolojik İşaretlere Dayalı Düzmece Video Tespit Çalışmaları**

Çalışma	Önerilen Model	Veri Setleri	Başarım Oranı
[51]	LRCN	CEW Dataset, Özel Veri Kümesi	%99,00 ACC
[52]	CNN-LTSM	Özel Veri Kümesi	%87,5 ACC
[55]	CNN	FF++, DFDC	%100 ACC
[56]	CNN	FF++, Celeb-DF, FF, FakeCatcher	%96 ACC
[57]	CNN	FF++, Celeb-DF	%94,66 ACC
[58]	Convolutional Attention Network	Celeb-DF, DFDC-Preview	%98,7 AUC
[59]	CNN	UADFV, Celeb-DF (v2), FF++	%99,94 ACC
[60]	CNN-RNN	Celeb-DF	%95 ACC
[54]	CNN	FF++, Celeb-DF, WDF	%99,64 ACC
[61]	CNN-LTSM	CelebA, FF++, Celeb-DF, DFD	%99,6 ACC

### 4.3. Ses-Görsel Tutarsızlıklarına Dayalı Tespit Çalışmaları

Bu çok yönlü yaklaşım, yüz veya ses manipülasyonu gerçekleştirilmiş düzmece videolarda özellikle etkili bir metodoloji sunmaktadır [23].

Zhou ve Lim [62] konuşma sırasındaki hece ve dudak hareketleri arasında güçlü bir ilişkinin var olduğunu gözlemlediler. Bu ilişki, düzmece videolarda sıklıkla bozulmaktadır. Dil bağımsız bu modeli FF++ ve DFDC veri kümelerinde test ederek %81,96 doğruluk elde ettiler. Model, video ve ses akışlarındaki düşük ve yüksek seviye özellikler arasındaki ilişkiyi öğrenerek ses ve görsel bölgeler arasındaki bağlantıyı kurmaktadır.

Cai ve ark. [63] geliştirdiği Sınır Algılamalı Zamansal Sahtecilik Tespiti (*Boundary Aware Temporal Forgery Detection*) tekniği, iki boyutlu evrişimsel sinir ağı (*2DCNN*) ile ses verilerinden bilgi çıkartarak çerçeve seviyesindeki uzamsal-zamansal bilgileri bir dizi kare olarak öğrenmek için de üç boyutlu evrişimsel sinir ağını (*3DCNN*) kullanmaktadır. Bu tekniğin tespit doğruluk oranı en yüksek %99 bulunmuştur.

Ilyas ve ark. [64] tarafından sunulan AVFakeNet tekniğinde özellik çıkarmak için Transformer modeli olan Yoğun Swin Dönüştürücü Ağı (*Dense Swin Transformer Net - DST-Net*) kullanılmıştır. Model çeşitli veri setlerinde test edilmiş %93,40 ACC performans değeri elde edilmiştir.

Anas Raza ve Mahmood Malik [65] tarafından geliştirilen Multimodaltrace tekniği, ses ve görüntü düzmece içeriği algılayan için yeni bir metodolojidir. Modelde, IntraModality Mixer katmanları aynı türdeki veri kanallarını (ses veya görüntü gibi) işlerken, InterModality Mixer katmanları ise farklı türdeki veri kanallarını (hem ses hem de görüntü) birleştirir. Böylece videodaki ses uyumsuzluğu incelenebilmektedir. FakeAVCeleb veri kümesi ve diğerleri üzerinde test edilen Multimodaltrace tekniğinin elde ettiği %92,9 doğruluk oranının mevcut yöntemlerden daha iyi olduğu ifade edilmiştir.

Hashmi ve ark. [66] ses ve görüntüyü birlikte analiz eden Transformer tabanlı bir ensemble modeli (toplu öğrenme) önermişlerdir. Mevcut yöntemlerin aksine bu yöntem hem ses hem de görsel manipülasyonları tespit edebilmektedir. Transformer modelinin güçlü modelleme, paralel işleme ve dikkat mekanizması sayesinde videodaki uzun vadeli bağımlılıklar ve küresel bilgi yakalanabilmektedir. Ayrıca ensemble öğrenme ile birden fazla modelin tahminleri birleştirilerek daha sağlam bir sonuç elde edilmektedir. FakeAVCeleb veri kümesi üzerinde yapılan analizlerde bu yöntemin mevcut tüm yöntemlere göre performansının daha yüksek olduğunu belirtmişlerdir.

Aşağıda verilen Çizelge-5'te ses ve görsel tutarsızlıklara dayalı düzmece video algılama çalışmalarının özeti verilmiştir.

**Çizelge-5: Ses-Görsel Tutarsızlıklarına Dayalı Düzmece Video Tespit Çalışmaları**

Çalışma	Önerilen Model	Veri Setleri	Başarım Oranı
[62]	ResCNN+GRU	FF++, DFDC	%81,96 ACC
[63]	3DCNN	DFDC, Celeb-DF	%99 ACC
[64]	AVFakeNet	FakeAVCeleb, Celeb-DF, ASVSpooft2019 LA, WLD, PDD	%93,40 ACC
[65]	Multimodaltrace	FakeAVCeleb, PDD, WLD	%92,9 ACC
[66]	AVTENetf	FakeAVCeleb	%99 ACC

#### 4.4. Evrişimsel İzlerin Algılanmasına Dayalı Çalışmalar

Düzmece üretiminden kaynaklanan bir kalıntı türü olan ve insan gözü ile fark edilmesi zor olan Evrişimsel izler (*convolutional traces*) aynı zamanda düzmece algılama için kullanılmaktadır. Bu yöntemin temel prensibi düzmece oluşturma işleminin görüntünün frekans spektrumunda belirli izler bırakmasıdır.

Li ve Lyu [67], çalışmasında düzmece video yapılırken yeniden boyutlandırma ve değiştirme işlemlerinin yarattığı yapay izlerin sebep olduğu yüzdeki bozulma kalıntılarını dikkate almaktadır. MesoNet [42]'i temel alan yöntemin tanıdık olmayan kaynaklardan gelen videolarda daha az etkili olduğu belirtilmiştir.

Frank ve ark. [68], farklı GAN mimarileri ve veri kümeleri üzerinde frekans alanında ortaya çıkan kalıntıları kapsamlı bir şekilde incelediler. GAN modellerinin örnekleme yükseltme teknikleri nedeniyle ciddi kalıntıların ortaya çıktığını gözlemlemişlerdir. Yapılan analizler, basit bir doğrusal model ve bir CNN tabanlı model içeren bir sınıflandırıcının, tüm frekans spektrumu üzerinde yüksek sonuçlar elde edebileceğini göstermektedir. Younus ve Hasan [69], Düzmece videolarını algılamak için Haar Dalga Dönüşümü odaklı bir yöntem geliştirdiler. Bu yöntem, videodaki yüzlerin boyut ve çözünürlüğündeki tutarsızlıkları ve farklılıkları analiz ederek sahte yüzlerin videolara eklenirken oluşturduğu belirgin bulanıklıkları tespit etmektedir. Sahte yüz sonradan yerleştirildiği için arka planla (yani gerçek vücut) arasında belli noktalarda bir bulanıklık oluşur. Haar Dalga Dönüşümü hem bu bulanıklığı ayırt edebilmektedir hem de dönüştürülen görüntünün "kenarlarını" daha iyi belirginleştirerek örneğin yüz çenesindeki geçişin keskinliğinin orijinale göre fark edilmesini sağlamaktadır. Yazarlar UADFV veri kümesi %90,5 oranında tespit başarısı elde etmişlerdir. Huang ve ark. [70], GAN üretimi sırasında video karesinin değiştirilmiş alanlarında renk, doku gibi özelliklerde anormallikleri tespit etmek için FakeLocator adında sahte alanları doğru bir şekilde bulan özel bir ağ ve gri tonlamalı bir harita kullanmıştır.

Görüntülerde yapılan değişiklikler gradyan bilgisinde (renklerin geçişlerinde, bir anlamda doku üzerindeki dalgalanma) içeriğin düzmece olduğunu belli edecek özellikler bırakır. Gri-Tonlama işlemi kullanılarak bu gradyan bilgisinin "görünürlüğü" artırılır [71]. Gri tonlamalı

görüntülerin içinde gizlenmiş gradyan verilerini kullanan Xiao ve ark. [71] geliştirdikleri yöntemle DeeperForensics veri kümesinde %98,17 oranında doğruluk elde etmişlerdir.

Lin ve Sun [72], az sayıda eğitim örneğiyle görüntülerdeki sahte alanları tespit edebilen GAN tabanlı bir yöntem geliştirdiler. Bu yöntem, gözetimsiz ve yarı gözetimli öğrenme kullanarak sahte yüzleri belirleyerek %93 doğrulukla sınıflandırma yapmıştır.

Çizelge-6'da evrişimsel izlere dayalı düzmece video algılama çalışmalarının özeti yer almaktadır.

**Çizelge-6: Evrişimsel İzlerin Algılanmasına Dayalı Düzmece Video Tespit Çalışmaları**

Çalışma	Önerilen Model	Veri Setleri	Başarım Oranı
[67]	CNN	UADFV, Deepfake-TIMIT	%99,9 AUC
[68]	CNN	Özel Veri Kümesi	%99,91 ACC
[69]	GAN	UADFV	%90,5 ACC
[70]	GAN	FF++, DFFD	%99,95 ACC
[71]	GAN	DeeperForensics	%98,17 ACC
[72]	GAN	FF++	%93 ACC

#### 4.5. Kimlik Bilgilerine Dayalı Çalışmalar

Agarwal ve ark. [73] yaptığı çalışma, düzmece videoları algılamak için statik yüz tanıma, yüz ifadelerinde gözlemlenen zamansal davranışlar ve baş hareketleri gibi çeşitli biyometrik özellikleri kullanmaktadır. Bu özelliklerin bütünleşik bir şekilde analiz edilmesi düzmece içeriklerin tespitinde kritik bir rol oynamaktadır. Çalışmada kullanılan CNN metrik-öğrenme amaç fonksiyonu aracılığıyla bu davranışsal özelliklerin entegrasyonunu öğrenmektedir.

Bu araştırmanın devamı niteliğindeki başka bir çalışmada Agarwal ve ark. [74] özellikle dudak senkronizasyonuna dayalı düzmece algılamaya odaklanmışlardır. Bu teknik, ağız şeklinin dinamiklerinin bazen söylenen seslerle (fonem) uyumsuz olabileceği gerçeğini göz önünde bulundurarak bu tür uyumsuzlukları tespit etmeye yöneliktir.

Cozzolino ve ark. [75], gerçek videolardan öğrenerek düzmece videoları algılayan bir yöntem geliştirdiler. Yüzdeki değişimleri ve hareketleri inceleyerek düzmece görüntüleri algılayan bu yüz okuma yöntemi, düşük kaliteli videolarda bile %81,8 doğruluk sağlayarak çalışmada kıyaslanan diğer yöntemlerden daha başarılı olmuştur.

Dong ve ark. [76], düzmece algılama için yüzdeki kimlik bilgilerini kullanan ve farklı veri kümelerindeki zorluklara rağmen kimlik tutarlılığını vurgulayan Kimlik Tutarlılığı Dönüştürücüsü (*Identity Consistency Transformer*) adında yeni bir model sundular. Bu çalışma benzerlik ölçüleri için bir eşik belirleme zorluğuna ve genellemeyi etkileyebilecek farklı veri kümelerinde değişen benzerlik dağılımları sorununa dikkat çekmiştir.

Shen ve ark. [77], gerçek ve sahte kimlikler arasında ayırım yapma becerisini geliştirmek için Dong ve ark. [76] yaptığı çalışmadan esinlenerek oynanmış ve edilmemiş yüzlerin farklarını öğrenmeye dayalı bir yöntem önerdiler. CNN temelli algılama yönteminin görüntünün merkezine odaklanmasına çözüm olarak giriş görüntülerinde rastgele maskeler kullandılar. Bu yaklaşım, zıt öğrenme görevlerinin aksine aynı kişiden kimliğe göre pozitif ve negatif örnekler almaktadır. Shen ve ark. [77] yaptıkları çalışma Celeb-DF veri kümesinde %91,76 doğruluğa ulaşmıştır.

Liu ve ark. [78], bir video içeriğinde yer alan bir kişinin yüzünde farklı kareler arasındaki farklılıkları tespit eden bir model geliştirdiler. Bu model, tüm karelerden elde ettiği kimlik bilgilerini kimlik vektörlerine dönüştürmekte ve ardından vektörlerden zamansal özellikler öğrenerek tutarsızlıkları tespit etmektedir.

Çizelge-7'de kimlik bilgilerine dayalı düzmece video algılama çalışmalarının özetine yer verilmiştir.

**Çizelge-7: Kimlik Bilgilerine Dayalı Düzmece Video Tespit Çalışmaları**

Çalışma	Önerilen Model	Veri Setleri	Başarım Oranı
[73]	CNN	FF++, DFDC, Celeb-DF, WLDR, DFD	%98,9 ACC
[74]	CNN	Özel Veri Kümesi	%99,6 ACC
[75]	CNN	FF++, DFDC, DFD	%96 AUC
[76]	Transformer	FF++	%94,43 AUC
[77]	CNN	FF++, Celeb-DF	%91,76 ACC
[78]	Encoder-RNN	FF++, DFD, DeeperForensics, Celeb-DF	%99,95 AUC

#### 4.6. Zaman Uyumsuzluğuna Dayalı Çalışmalar

Bu yöntemin amacı, videonun kareleri arasındaki tutarsızlıkları analiz edilerek düzmece görüntüleri algılamaktır.

Düzmece teknolojisinin hızla gelişmesiyle birlikte bu alandaki algılama metodolojileri de sürekli evrim geçirmektedir. Bu süreçte Guera ve Delp [79] RNN kullanarak düzmece algılamada önemli bir adım atmışlardır. Bu çalışmanın yaklaşımı, CNN ile çerçeve düzeyinde özellikler çıkararak RNN modeli eğitmeye dayanmaktadır. Bu yöntem sadece kısa süreli videolardaki zamansal bilgileri yakalamada %94'ün üzerinde bir başarı elde ederek zamansal dinamiklerin düzmece algılamadaki kritik rolünü vurgulamıştır.

Montserrat ve ark. [80] videolardaki görsel değişiklikleri tespit etmek için Otomatik Yüz Ağırlığı (*Automatic Face Weighting - AFW*) mekanizması ile CNN ve RNN kullanarak videonun karelerindeki sahte yüz olasılıklarını ağırlıklandırarak ve karelerin özellik vektörlerini zamansal (temporal) öğreniminde kullanılan bir GRU katmanı içermektedir. Geliştirilen model DFDC veri kümesinde %91,88 doğruluk elde etmişse de ses içeriği için bir analiz verilmemiştir.

Zheng ve ark. [81] gerçek ve sahte videolar arasındaki uzun vadeli bağımlılıkları yakalamak için zamansal transformatör kullanmayı önermişlerdir. Zamansal transformatör videonun her bir karesini ve kareler arasındaki ilişkileri dikkate alan bir evrişimsel ağ türüdür. Yazarlar, transformatörlerin küresel bağımlılıkları yakalamaya çok daha uygun olduğunu ancak zamana dayalı evrişimsel bir ağ kullanarak genelleme yeteneğinin geliştirilebileceğine dikkat çekmişlerdir. Çalışmada önerilen yöntemin %94,2 doğruluk oranıyla gerçek ve sahte videoları ayırt edebildiği gösterilmiştir.

Saikia ve ark. [82] videodaki nesnelere ve kameranın hareketini takip eden "optik akış" tekniği ile görüntülerden anlamlı özellikler çıkarmak için CNN ve zamansal sıralardaki bilgileri işleme ve öğrenme için RNN bir karışımı bir model kullanmışlardır. Model çeşitli veri kümelerinde test edilmiş ve FF++ veri kümesinde en yüksek %91,21 doğruluk oranı elde etmiştir.

Rahman ve ark. [83] düzmece videoları algılayabilecek zamana duyarlı bir çerçeve geliştirmişlerdir. Araştırmacılar InceptionResNetV2, MobileNet ve DenseNet modelleriyle basit CNN kullanmış ve analizlerde MobileNet en iyi performansı göstermiştir. Kullanılan CNN modeli özellikle düşük çözünürlüklü ve kısa süreli videolarda DFDC ve FF++ veri kümeleri üzerinde sınıdığında yüksek doğruluk oranları elde etmiştir. Model DFDC veri kümesinde %94,93 ve FF++ veri kümesinde %93,2 doğruluk oranına ulaşmıştır.

Kolagati ve ark. [84], CNN ve Çok Katmanlı Algılayıcı (*Multi-Layer Perceptron - MLP*)'yi birleştirerek düzmece videoları sınıflandıran hibrit bir model geliştirmişlerdir. Model CNN ile videodaki yüzün belirgin noktalarını analiz ederek özellikler çıkarmakta ve MLP ile ilk sınıflandırmayı gerçekleştirmektedir. Son karar ise CNN ve MLP'nin ortak bilgisine dayanmaktadır. 199 sahte ve 119 gerçek videodan oluşan test veri kümesinde model %87 doğruluk elde ederek sadece CNN kullanan modelleri geride bırakmıştır. Modelin aynı zamanda aşırı öğrenmeyi (*overfitting*) azaltarak daha hızlı eğitim aldığı belirtilmiştir.

Thing [85] çalışmasında düzmece algılamada zamansal tutarsızlıklara ve çoklu-veri kümesi değerlendirmesine odaklanmıştır. Araştırmada CNN ve Transformer modellerinin düzmece tespitindeki etkinlikleri incelenmiş ve veri kümesi bağımlılığı, mimari seçimi ve veri dengesizliği gibi zorluklar ele alınmıştır. Çalışma kapsamında FF++, Google DFD, Celeb-DF, DeeperForensics ve DFDC gibi köklü veri kümeleri üzerinde düzmece tespit modelleri test edilmiştir. DeeperForensics veri kümesinde %99,73 doğruluk elde edilmiştir.

Gu ve ark. [53], CNN ve kareler arası analiz için LSTM ağlarını kullandıkları yöntemde FF++ veri kümesinde %92,4 algılama doğruluğu elde edilmiştir.

Çizelge-8'de zaman uyumsuzluğuna dayalı düzmece video algılama çalışmaları verilmiştir.



**Çizelge-8: Zaman Uyumsuzluğuna Dayalı Düzmece Video Tespit Çalışmaları**

Çalışma	Önerilen Model	Veri Setleri	Başarım Oranı
[79]	CNN-RNN	İnternet Videoları	%94,00 ACC
[80]	CNN-GRU	DFDC	%91,88 ACC
[81]	Transformatör	FF++, FaceShifter, DeeperForensics, DFDC	%94,2 ACC
[82]	CNN-RNN	DFDC, FF++, Celeb-DF	%91,21 ACC
[83]	CNN	DFDC, FF++	%94,93 ACC
[84]	CNN ve MLP	DFDC, Youtube	%87 ACC
[85]	CNN ve Transformer	FF++, DFD, Celeb-DF, DeeperForensics, DFDC	%99,73 ACC
[53]	CNN-LSTM	FF++	%92,4 ACC

#### 4.7. Yüz İfadelerine Dayalı Çalışmalar

Mittal ve ark. [86], yüz ifadeleri ve sesteki duygusal ifadelerden yola çıkarak Siyam (*Siamese*) sinir ağıları ve modelin benzer örnekleri ayırt etmeyi ve farklı örnekleri birbirinden ayırmayı öğrenmesini sağlayan triplet kayıp fonksiyonundan esinlenen bir derin öğrenme yaklaşımı ortaya koydular. Özgünlükleri, bu modelde hem ses hem görüntü içeriklerinin yanında kişinin bu içeriklerdeki duygularını da dahil etmeleridir. Bu yöntem, DFDC veri kümesinde %84,4 ve DF-TIMIT veri kümesinde %96,6 AUC skoru sağlamıştır. Ancak yazarlar, insan duygularının karmaşıklığı nedeniyle oluşabilecek algılama başarısızlıklarına da dikkat çekmektedir.

Hosler ve ark. [87] düzmece tespiti için sesteki ve görüntüdeki duygusal ifadeler arasındaki tutarsızlıklara odaklanan bir yöntem geliştirdiler. Yöntem, zaman içinde duyguları temsil etmek için kişinin konuşmasından ve yüz ifadelerinden temel özellikler çıkarmayı ve ardından bu özellikleri eğitilmiş bir RNN modeli ile analiz etmeyi içermektedir. RNN modelinde, konuşmadaki temel duygusal özellikleri incelemek için LSTM ağıları ve tahmin edilen duygu bilgisine dayanarak düzmece videoları ayırt etmek için bir gözetimli sınıflandırıcı kullanılmıştır. Çalışmada, düzmece videolarda yüz ifadelerinin taklit edilmesine odaklanarak sahteciliği yakalayabilmekten ziyade düzmece ile sesteki duyguların taklit edilmesinin daha başarısız olduğu vurgulanmıştır. Öte yandan daha uzun video örneklerinde bu yöntemin başarısı %99,5 seviyesine kadar çıkabilmektedir.

Pei ve ark. [88] video içeriklerindeki düzmece yüzlerin tespiti için zamansal özelliklere odaklanan Çift Yönlü LSTM yöntemi önermişlerdir. Bu yöntem, yüz ifadelerindeki zamansal değişimleri analiz ederek kaş kaldırma, göz kırpması ve gülümseme gibi ince yüz ifadelerini dikkate alarak sahte ve gerçek yüzlerin ayırt edilmesi sağlamaktadır. Çift Yönlü LSTM yöntemi daha az eğitim süresi gerektirerek DFDC veri kümesi üzerinde %82,65 doğruluk oranı elde etmiştir. Bu yöntemin özellikle de gürültülü sıkıştırılmış videoları tespit etmede kayda değer başarılı olduğu belirtilmiştir.

Güncel ve önemli düzmece tespit çalışmalarından biri olarak Haq ve ark. [89], psikolojik bilginin ve sembolik muhakeme yeteneğinin birleştirildiği psikolojik analiz temelinde yeni bir biyolojik sinyal geliştirdiler. Bu sinyal, kişinin duygusal geçişlerindeki tutarlılığa ve farklı ifade araçları (mimik, ses, vücut dili) arasındaki uyuma odaklanmaktadır. Analiz sonuçlarında PDD veri kümesinde %93,7 doğruluk oranı, WLDR veri kümesinde ise %75,34 doğruluk oranı elde edilmiştir. Psikoloji ve duygu bilimi alanındaki temel bilgileri kullanan bu yöntemin daha genellenebilir olması oldukça yüksektir.

Çizelge-9'da yüz ifadelerine dayalı düzmece video algılama çalışmaları özetlenmiştir.

**Çizelge-9: Yüz İfadelerine Dayalı Düzmece Video Tespit Çalışmaları**

Çalışma	Önerilen Model	Veri Setleri	Başarım Oranı
[86]	IQIEA-FS	DFDC, DF-TIMIT	%96,6 AUC
[87]	RNN	Celeb-DF	%99,5 ACC
[88]	LSTM	DFDC	%82,65 ACC
[89]	ResNet50	WLDR, PDD	%93,7 ACC

#### 4.8. Uzamsal-Zamansal Özelliklere Dayalı Tespit

Uzamsal-zamansal özelliklerden yararlanma yöntemi, düzmece algılamada yaygın kullanılan çok modlu (farklı veri kaynağı kullanan) bir yaklaşımdır. Burada video kareleri içindeki görsel düzensizlikler (*intra-frame inceleme*) ve video akışları boyunca zamansal özellikler (*inter-frame inceleme*) analiz edilir [28].

Nguyen ve ark. [90] çoklu görev metodu CNN tabanlı Y-shaped Autoencoder (*Y-şekilli otokodlayıcı*) modeli, videolardan oynanmış içeriğin hem uzamsal (*spatial*) hem de zamansal (*temporal*) özelliklerini analiz ederek tespit ve yerelleştirme işlemleri için tasarlanmıştır. Bu model düzmece algılamada FF+ veri kümesinde %92,77 doğruluk oranı sağlasa da görülmemiş senaryolarda değerlendirme doğruluğunun düştüğü belirtilmiştir.

De Lima ve ark. [91] düzmece videolardaki karelerden uzamsal ve zamansal bilgileri (*3D input*) öğrenerek manipülasyonları tespit etmeyi hedeflemişlerdir. Çalışmada CNN modeli VGG-11 ile videolardan görsel özellik çıkarılmış ardından bu özelliklerin zaman içindeki sırasını analiz etmek için LSTM kullanılmıştır. R3D, ResNet, I3D gibi farklı CNN modelleri LSTM'in ürettiği zaman sıralı verilerle eğitilerek düzmece tespiti için zor olan Celeb-DF veri kümesine üzerinde test edilmiştir. Bu çalışmada R3D modeli en iyi sonuç vererek düzmece videolarını %98,26 doğruluk oranıyla sınıflandırabilmiştir. Ancak bu yaklaşım yüksek hesaplama maliyeti gerektirmektedir.

Li ve ark. [92], birden fazla video karesini tek seferde işledikleri çalışmada zaman içerisinde yüzdeki kısmi değişiklikleri algılamayı amaçlamışlardır. Model FF++ veri kümesinde %99,82 doğruluk elde etmiştir.

Hubalovsky ve ark. [93] CNN tabanlı YOLO dedektör ve yerel ikili desen histogramı (*local binary pattern histogram*) kullanarak, uzamsal ve zamansal özellikleri bir araya getirdiler. Bu yöntemle öncelikle video karelerinde veya görüntülerde yüz tespiti yapılmakta ve ardından uzamsal özellikler çıkarılmaktadır. Çalışmada %98,12 doğruluk elde edilmiştir.

Lu ve ark. [94], iCapsNet-TSF olarak adlandırdıkları gelişmiş bir Kapsül Ağı ile zamansal-uzamsal özellikleri yakalayan optik akış algoritmasını birleştiren bir yöntem geliştirmişlerdir. Ayrıca zamansal-uzamsal verileri analiz etmek için Dinamik Yönlendirme Algoritması kullanılmıştır. Kapsül ağı içindeki ağırlık başlatma ve güncelleme gibi iyileştirmeler sayesinde bu yaklaşım düzmece algılama doğruluğunu önemli ölçüde artırarak %98,83 doğruluk yakalamıştır. Ek olarak, yöntem farklı veri kümeleri arasında çalışırken de doğruluğu koruma zorluğunun üstesinden gelmiştir.

Dolla ve ark. [95], düzmece algılamayı iyileştirmek için uzamsal-zamansal özellik piramit ağı geliştirdiler. Bu ağ, video karelerinden gelen uzamsal ve zamansal özellikleri güçlendirerek çalışmaktadır. Model, doku bazlı özniteliklerle yüksek seviye görsel detayları bir araya getirerek yüzleri doğru bir şekilde temsil etmeye odaklanır. CNN ve RNN tabanlı modellerin kullanıldığı ve FF++, DeepForensics ve CelebDF gibi farklı veri kümelerinin sınındığı analizlerde FF++ veri kümesinde %99,99 AUC değeri elde edilmiştir.

Kaddar ve ark. [96] düzmece video algılama için uzamsal-zamansal özelliklere odaklanarak HCiT adını verdikleri mimariyi kullanmışlardır. HCiT modelinde bir CNN katmanı ilk olarak girdi görüntüsündeki kenar ve köşe gibi alt seviye görsel özellikleri çıkarmaktadır. Bu öznitelikler daha sonra ViT (*Vision Transformer*) mimarisine beslenir ve dikkat mekanizmasını kullanarak bu detaylar arasındaki ilişkiyi analiz etmektedir. Yöntem, özellikle yüz değiştirme teknikleriyle oynanmış görüntülerde yüksek değerler sağlarken FaceSwap ve Face2Face alt veri kümelerinde %95,85'in üzerinde doğruluk göstermiştir.

Çizelge-10'da uzamsal-zamansal özelliklere dayalı düzmece video algılama çalışmalarının özeti yer almaktadır.

**Çizelge-10: Uzamsal-Zamansal Özelliklere Dayalı Düzmece Video Tespit Çalışmaları**

Çalışma	Önerilen Model	Veri Setleri	Başarım Oranı
[90]	CNN	FF++	%92,77 ACC
[91]	CNN	Celeb-DF	%98,26 ACC
[92]	S-MIL	Celeb-DF, FF++, DFDC	%100 ACC
[93]	CNN-YOLO	CelebDF-FF++, DFFD, CASIA-Web Face Dataset	%98,12 ACC
[94]	iCapsNet-TSF	Celeb-DF, FaceSwap, Deepfakes	%98,83 ACC
[95]	CNN-RNN	DeepForensics, Celeb-DF, FaceForensics++	%99,99 AUC
[96]	ViT	FF++, DFDC preview, Celeb -DF	%95,85 ACC

## 5. Sonuç

Bu literatür incelemesi, düzmece video algılama amacıyla yapılan son yıllarda yapılan çalışmalarını ele almaktadır. Bu alanda yapılan çalışmalar kayda değer başarılar sergilese de düzmece algılama hala birçok açıdan geliştirilmesi gereken önemli bir konudur.

Düzmece üretimiyle ilgili endişe verici bir nokta, GAN modellerinin hızla gelişmesiyle inandırıcılığı yüksek video görüntülerinin ortaya çıkmasıdır. Bu eğilim, düzmece tespitini daha da zorlaştırdığı için mevcut yöntemlerden daha etkili algoritmaların ortaya çıkması veya mevcut algoritmaların evrimleşmesi ile bu sorunun çözülmesi sağlanabilecektir [34]. Çalışmadan da anlaşılacağı üzere günümüz tespit yöntemleri görülmemiş veri kümelerinde %100 algılama doğruluk oranı sağlayamamaktadır.

Veri kümeleri arasındaki tutarsızlıklar ve modelin hiç karşılaşmadığı veri türleri mevcut düzmece algılama modellerinin başarısını etkilemektedir. Düzmece algılama modellerinin etkinliği, eğitim için çeşitli ve çok sayıda örnek içeren veri kümelerine büyük ölçüde bağlıdır. Modeller, bilinmeyen manipülasyonlara sahip ortamlarla karşılaştığında bu tür manipülasyonları doğru bir şekilde tanımlama yetenekleri düşmektedir. Düzmece algılama sistemlerini aldatmak amacıyla video üzerinde bulanıklaştırma, yumuşatma, kırpma gibi işlemler uygulanır. Manipülasyon tekniklerindeki bu çeşitlilik kapsamlı veri kümelerinin eksikliğiyle birleştiğinde etkili tespit modelleri tasarlamada önemli bir zorluk oluşturmaktadır [23]. Bu nedenle veri kümelerinin geliştirilmesi ve bu konuda gerekli desteklerin belirli kuruluşlar tarafından sağlanması önemlidir.

Bir başka düzmece algılama kısıtı olan hesaplama kısıtları hem düzmece üretme hem de algılamayı önemli ölçüde etkilemektedir. Genellikle GAN modelleri gibi gelişmiş derin öğrenme modelleri kullanan düzmece oluşturma süreci yüksek hesaplama gücü gerektirmektedir. Siber suç analizi veya içerik denetimi gibi uygulamalar için gerçek zamanlı analiz gerektiren düzmece algılama algoritmaları yüksek çözünürlüklü videoları işlemek için yüksek işlem gücü gerektirdiği gibi çok miktarda enerji tüketerek yüksek maliyetlere yol açmaktadır. Bu durum sınırlı kaynaklara sahip olanlar için bu teknolojinin kullanımını zorlaştırmaktadır. Sonuç olarak, bu hesaplama sorunlarını gidermek için daha verimli yapay zekâ modellerine ve özel donanım geliştirmeye artan bir ihtiyaç vardır [27].

## Kaynakça

- [1] Statista. Global internet user penetration 2024, <https://0311b0kku-y-https-www-statista-com.halic.proxy.deepknowledge.io/statistics/325706/global-internet-user-penetration/>, Erişim tarihi:15.02.2024.
- [2] ITRC. 2023 Annual Data Breach Report, <https://www.idtheftcenter.org/publication/2023-data-breach-report/>, Erişim tarihi:15.02.2024.

- [3] Heidari, A., Jafari Navimipour, N., Dag, H., & Unal, M. Deepfake detection using deep learning methods: A systematic and comprehensive review, *WIREs Data Mining and Knowledge Discovery*, 2023. <https://doi.org/10.1002/widm.1520>
- [4] Salman, S., Shamsi, J. A., & Qureshi, R., Deep Fake Generation and Detection: Issues, Challenges, and Solutions. *IT Professional*, 2023, 25(1), 52-59. <https://doi.org/10.1109/MITP.2022.3230353>
- [5] Kietzmann, J., Lee, L. W., McCarthy, I. P., & Kietzmann, T. C., Deepfakes: Trick or treat? *Business Horizons*, 2020, 63(2), 135-146. <https://doi.org/10.1016/j.bushor.2019.11.006>
- [6] Ajao, O., Bhowmik, D., & Zargari, S., Sentiment Aware Fake News Detection on Online Social Networks, *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, 2507-2511. <https://doi.org/10.1109/ICASSP.2019.8683170>
- [7] Caldelli, R., Galteri, L., Amerini, I., & Bimbo, A., Optical Flow based CNN for detection of unlearned Deepfake manipulations, *Pattern Recognition Letters*, 2021, 146. <https://doi.org/10.1016/j.patrec.2021.03.005>
- [8] Van Der Sloot, B., & Wagenveld, Y., Düzmece: Regulatory challenges for the synthetic society, *Computer Law & Security Review*, 2022, 46, 105716. <https://doi.org/10.1016/j.clsr.2022.105716>
- [9] Neethirajan, S., Is Seeing Still Believing? Leveraging Deepfake Technology for Livestock Farming, *Frontiers in Veterinary Science*, 2021, 8. <https://api.semanticscholar.org/CorpusID:244715980>
- [10] Yu, P., Xia, Z., Fei, J., & Lu, Y., A Survey on Deepfake Video Detection, *IET Biometrics*, 2021, 10(6), 607-624. <https://doi.org/10.1049/bme2.12031>
- [11] Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., & Bengio, Y., *Generative Adversarial Networks*, 2014. <https://doi.org/10.48550/ARXIV.1406.2661>
- [12] He, Z., Zuo, W., Kan, M., Shan, S., & Chen, X., AttGAN: Facial Attribute Editing by Only Changing What You Want, *IEEE Transactions on Image Processing*, 2019, 28(11), 5464-5478. <https://doi.org/10.1109/TIP.2019.2916751>
- [13] Karras, T., Laine, S., & Aila, T., A Style-Based Generator Architecture for Generative Adversarial Networks, 2018. <https://doi.org/10.48550/ARXIV.1812.04948>
- [14] Liu, M., Ding, Y., Xia, M., Liu, X., Ding, E., Zuo, W., & Wen, S., STGAN: A Unified Selective Transfer Network for Arbitrary Image Attribute Editing, 2019. <https://doi.org/10.48550/ARXIV.1904.09709>
- [15] Choi, Y., Uh, Y., Yoo, J., & Ha, J.-W., StarGAN v2: Diverse Image Synthesis for Multiple Domains, *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, 8185-8194. <https://doi.org/10.1109/CVPR42600.2020.00821>
- [16] Temnikova, I., Marinova, I., Looking for Traces of Textual Deepfakes in Bulgarian on Social Media, *Proceedings of the Conference Recent Advances in Natural Language Processing - Large Language Models for Natural Language Processings*, 2023, 1151-1161. [https://doi.org/10.26615/978-954-452-092-2\\_122](https://doi.org/10.26615/978-954-452-092-2_122)
- [17] Conti, E., Salvi, D., Borrelli, C., Hosler, B., Bestagini, P., Antonacci, F., Sarti, A., Stamm, M. C., & Tubaro, S., Deepfake Speech Detection Through Emotion Recognition: A Semantic Approach, *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2022, 8962-8966. <https://doi.org/10.1109/ICASSP43922.2022.9747186>
- [18] Patel, Y., Tanwar, S., Bhattacharya, P., Gupta, R., Alsuwian, T., Davidson, I. E., & Mazibuko, T. F., An Improved Dense CNN Architecture for Deepfake Image Detectio, *IEEE Access*, 2023, 11, 22081-22095. <https://doi.org/10.1109/ACCESS.2023.3251417>
- [19] Yang, X., Li, Y., & Lyu, S., Exposing Deep Fakes Using Inconsistent Head Poses, 2018. <https://doi.org/10.48550/ARXIV.1811.00661>
- [20] Li, L., Bao, J., Zhang, T., Yang, H., Chen, D., Wen, F., & Guo, B., Face X-ray for More General Face Forgery Detection, 2019. <https://doi.org/10.48550/ARXIV.1912.13458>
- [21] Tolosana, R., Vera-Rodriguez, R., Fierrez, J., Morales, A., & Ortega-Garcia, J., Deepfakes and Beyond: A Survey of Face Manipulation and Fake Detection, 2020. <https://doi.org/10.48550/ARXIV.2001.00179>
- [22] Zi, B., Chang, M., Chen, J., Ma, X., & Jiang, Y.-G., WildDeepfake: A Challenging Real-World Dataset for Deepfake Detection, *Proceedings of the 28th ACM International Conference on Multimedia*, 2020.
- [23] Masood, M., Nawaz, M., Malik, K. M., Javed, A., & Irtaza, A., Deepfakes Generation and Detection: State-of-the-art, open challenges, countermeasures, and way forward, 2021. <https://doi.org/10.48550/ARXIV.2103.00484>
- [24] Waseem, S., Abu-Bakar, S., Omar, Z., Ahmed, B., Baloch, S., & Hafeezallah, A., Multi-attention-based approach for Deepfake face and expression swap detection and localization, *EURASIP Journal on Image and Video Processing*, 2023. <https://doi.org/10.1186/s13640-023-00614-z>
- [25] Chung, J. S., Nagrani, A., & Zisserman, A., VoxCeleb2: Deep Speaker Recognition, 2018. <https://doi.org/10.48550/ARXIV.1806.05622>
- [26] Salvi, D., Liu, H., Mandelli, S., Bestagini, P., Zhou, W., Zhang, W., & Tubaro, S., A Robust Approach to Multimodal Deepfake Detection, *Journal of Imaging*, 2023, 9(6), 122. <https://doi.org/10.3390/jimaging9060122>
- [27] Gupta, G., Raja, K., Gupta, M., Jan, T., Whiteside, S. T., & Prasad, M., A Comprehensive Review of Deepfake Detection Using Advanced Machine Learning and Fusion Methods, *Electronics*, 2023, 13(1), 95. <https://doi.org/10.3390/electronics13010095>
- [28] Juefei-Xu, F., Wang, R., Huang, Y., Guo, Q., Ma, L., & Liu, Y., Countering Malicious Deepfakes: Survey, Battleground, and Horizon, *International Journal of Computer Vision*, 2022, 130, 1-57. <https://doi.org/10.1007/s11263-022-01606-8>
- [29] Rana, M. S., Nobi, M. N., Murali, B., & Sung, A. H., Deepfake Detection: A Systematic Literature Review, *IEEE Access*, 2022, 10, 25494-25513. <https://doi.org/10.1109/ACCESS.2022.3154404>
- [30] Groh, M., Epstein, Z., Firestone, C., & Picard, R., Deepfake detection by human crowds, machines, and machine-informed crowds, *Proceedings of the National Academy of Sciences*, 2022, 119(1), e2110013119. <https://doi.org/10.1073/pnas.2110013119>
- [31] Köbis, N. C., Doležalová, B., & Soraperra, I., Fooled twice: People cannot detect Deepfakes but think they can, *iScience*, 2021, 24(11), 103364. <https://doi.org/10.1016/j.isci.2021.103364>
- [32] Somoray, K., & Miller, D. J., Providing detection strategies to improve human detection of Deepfakes: An experimental study, *Computers in Human Behavior*, 2023, 149, 107917. <https://doi.org/10.1016/j.chb.2023.107917>
- [33] Deng, L., *Deep Learning: Methods and Applications*, Foundations and Trends® in Signal Processing, 2014, 7(3-4), 197-387. <https://doi.org/10.1561/20000000039>

- [34] Naitali, A., Ridouani, M., Salahdine, F., & Kaabouch, N., Deepfake Attacks: Generation, Detection, Datasets, Challenges, and Research Directions, *Computers*, 2023, 12(10), 216. <https://doi.org/10.3390/computers12100216>
- [35] Krizhevsky, A., Sutskever, I., & Hinton, G. E., ImageNet classification with deep convolutional neural networks, *Communications of the ACM*, 2017, 60(6), 84-90. <https://doi.org/10.1145/3065386>
- [36] Simonyan, K., & Zisserman, A., Very Deep Convolutional Networks for Large-Scale Image Recognition, 2014. <https://doi.org/10.48550/ARXIV.1409.1556>
- [37] Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., & Rabinovich, A., Going Deeper with Convolutions, 2014. <https://doi.org/10.48550/ARXIV.1409.4842>
- [38] He, K., Zhang, X., Ren, S., & Sun, J., Deep Residual Learning for Image Recognition, 2015. <https://doi.org/10.48550/ARXIV.1512.03385>
- [39] Huang, G., Liu, Z., Van Der Maaten, L., & Weinberger, K. Q., Densely Connected Convolutional Networks, 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, 2261-2269. <https://doi.org/10.1109/CVPR.2017.243>
- [40] Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., & Adam, H., MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications, 2017. <https://doi.org/10.48550/ARXIV.1704.04861>
- [41] Tan, M., & Le, Q. V., EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks, 2019. <https://doi.org/10.48550/ARXIV.1905.11946>
- [42] Afchar, D., Nozick, V., Yamagishi, J., & Echizen, I., MesoNet: A Compact Facial Video Forgery Detection Network, 2018. <https://doi.org/10.48550/ARXIV.1809.00888>
- [43] Zhao, H., Wei, T., Zhou, W., Zhang, W., Chen, D., & Yu, N., Multi-attentional Deepfake Detection, 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2021, 2185-2194. <https://doi.org/10.1109/CVPR46437.2021.00222>
- [44] Kohli, A., & Gupta, A., Detecting Deepfake, FaceSwap and Face2Face facial forgeries using frequency CNN, *Multimedia Tools and Applications*, 2021, 80(12), 18461-18478. <https://doi.org/10.1007/s11042-020-10420-8>
- [45] Luo, Y., Zhang, Y., Yan, J., & Liu, W., Generalizing Face Forgery Detection with High-frequency Features, 2021. <https://doi.org/10.48550/ARXIV.2103.12376>
- [46] Ismail, A. A., Elpeltagy, M. S., Zaki, M. S., & Eldahshan, K. A., A New Deep Learning-Based Methodology for Video Deepfake Detection Using XGBoost, *Sensors*, 2021, 21.
- [47] Das, A., & Sebastian, L., A Comparative Analysis and Study of a Fast Parallel CNN Based Deepfake Video Detection Model with Feature Selection (FPC-DFM), 2023 *Advanced Computing and Communication Technologies for High Performance Applications (ACCTHPA)*, 2023, 1-9. <https://doi.org/10.1109/ACCTHPA57160.2023.10083340>
- [48] Dhanaraj, R., & SriDevi, M., Face Warping Deepfake Detection and Localization in a Digital Video using Transfer Learning Approach, *Journal of Metaverse*, 2023, 4(1), 11-20. <https://doi.org/10.57019/jmv.1338907>
- [49] Chollet, F., Xception: Deep Learning with Depthwise Separable Convolutions, 2017. <http://arxiv.org/abs/1610.02357>
- [50] Matern, F., Riess, C., & Stamminger, M., Exploiting Visual Artifacts to Expose Deepfakes and Face Manipulations, 2019 *IEEE Winter Applications of Computer Vision Workshops (WACVW)*, 2019, 83-92. <https://doi.org/10.1109/WACVW.2019.00020>
- [51] Li, Y., Chang, M.-C., & Lyu, S., In Ictu Oculi: Exposing AI Created Fake Videos by Detecting Eye Blinking, 2018, 1-7. <https://doi.org/10.1109/WIFS.2018.8630787>
- [52] Jung, T., Kim, S., & Kim, K., DeepVision: Derin Deepfakes Detection Using Human Eye Blinking Pattern, *IEEE Access*, 2020, 8, 83144-83154. <https://doi.org/10.1109/ACCESS.2020.2988660>
- [53] Gu, J., Xu, Y., Sun, J., & Liu, W., Exploiting Deepfakes by Analyzing Temporal Feature Inconsistency, *International Journal of Advanced Computer Science and Applications*, 2023, 14(12). <https://doi.org/10.14569/IJACSA.2023.0141291>
- [54] He, Q., Peng, C., Liu, D., Wang, N., & Gao, X., GazeForensics: Deepfake Detection via Gaze-guided Spatial Inconsistency Learning, 2023. <https://doi.org/10.48550/ARXIV.2311.07075>
- [55] Qi, H., Guo, Q., Juefei-Xu, F., Xie, X., Ma, L., Feng, W., Liu, Y., & Zhao, J., DeepRhythm: Exposing Deepfakes with Attentional Visual Heartbeat Rhythms, 2020. <https://doi.org/10.48550/ARXIV.2006.07634>
- [56] Ciftci, U., Demir, I., & Yin, L., FakeCatcher: Detection of Synthetic Portrait Videos using Biological Signals, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020, PP, 1-1. <https://doi.org/10.1109/TPAMI.2020.3009287>
- [57] Ciftci, U. A., Demir, I., & Yin, L., How Do the Hearts of Deep Fakes Beat? Deep Fake Source Detection via Interpreting Residuals with Biological Signals, 2020. <https://doi.org/10.48550/ARXIV.2008.11363>
- [58] Hernandez-Ortega, J., Tolosana, R., Fierrez, J., & Morales, A., DeepfakesON-Phys: Deepfakes Detection based on Heart Rate Estimation, 2020. <https://doi.org/10.48550/ARXIV.2010.00400>
- [59] Wang, B., Li, Y., Wu, X., Ma, Y., Song, Z., & Wu, M., Face Forgery Detection Based on the Improved Siamese Network, *Security and Communication Networks*, 2022, 1-13. <https://doi.org/10.1155/2022/5169873>
- [60] Khurana, P. S., Sudarshan, T. B., Natarajan, S., Nagesh, V., Lakshminarayanan, V., Bhat, N., & Vinay, A., AFMB-Net: Deepfake Detection Network Using Heart Rate Analysis, *Tehnički glasnik*, 2022, 16(4), 503-508. <https://doi.org/10.31803/tg-20220403080215>
- [61] Liang, P., Liu, G., Xiong, Z., Fan, H., Zhu, H., & Zhang, X., A facial geometry based detection model for face manipulation using CNN-LSTM architecture, *Information Sciences*, 2023, 633, 370-383. <https://doi.org/10.1016/j.ins.2023.03.079>
- [62] Zhou, Y., & Lim, S.-N., Joint Audio-Visual Deepfake Detection, 2021 *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021, 14780-14789. <https://doi.org/10.1109/ICCV48922.2021.01453>
- [63] Cai, Z., Stefanov, K., Dhall, A., & Hayat, M., Do You Really Mean That? Content Driven Audio-Visual Deepfake Dataset and Multimodal Method for Temporal Forgery Localization, 2022. <https://doi.org/10.48550/ARXIV.2204.06228>
- [64] Ilyas, H., Javed, A., & Malik, K. M., AVFakeNet: A unified end-to-end Dense Swin Transformer deep learning model for audio-visual Deepfakes detection, *Applied Soft Computing*, 2023, 136, 110124. <https://doi.org/10.1016/j.asoc.2023.110124>
- [65] Anas Raza, M., & Mahmood Malik, K., Multimodaltrace: Deepfake Detection using Audiovisual Representation Learning, 2023 *IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2023, 993-1000. <https://doi.org/10.1109/CVPRW59228.2023.00106>

- [66] Hashmi, A., Shahzad, S. A., Lin, C.-W., Tsao, Y., & Wang, H.-M., AVTENet: Audio-Visual Transformer-based Ensemble Network Exploiting Multiple Experts for Video Deepfake Detection, 2023. <https://doi.org/10.48550/ARXIV.2310.13103>
- [67] Li, Y., & Lyu, S., Exposing Deepfake Videos By Detecting Face Warping Artifacts, 2018. <https://doi.org/10.48550/ARXIV.1811.00656>
- [68] Frank, J., Eisenhofer, T., Schönherr, L., Fischer, A., Kolossa, D., & Holz, T., Leveraging Frequency Analysis for Deep Fake Image Recognition, 2020. <https://doi.org/10.48550/ARXIV.2003.08685>
- [69] Younus, M. A., & Hasan, T. M., Effective and Fast Deepfake Detection Method Based on Haar Wavelet Transform, 2020 International Conference on Computer Science and Software Engineering (CSASE), 2020, 186-190. <https://doi.org/10.1109/CSASE48920.2020.9142077>
- [70] Huang, Y., Juefei-Xu, F., Guo, Q., Liu, Y., & Pu, G., FakeLocator: Robust Localization of GAN-Based Face Manipulations, 2020. <https://doi.org/10.48550/ARXIV.2001.09598>
- [71] Xiao, S., Yang, J., & Lv, Z., Protecting the trust and credibility of data by tracking forgery trace based on GANs, Digital Communications and Networks, 2022, 8(6), 877-884. <https://doi.org/10.1016/j.dcan.2022.07.010>
- [72] Lin, Y.-K., & Sun, H.-L., Few-Shot Training GAN for Face Forgery Classification and Segmentation Based on the Fine-Tune Approach, Electronics, 2023, 12(6), 1417. <https://doi.org/10.3390/electronics12061417>
- [73] Agarwal S, Farid H, El-Gaaly T, Lim SN., Detecting deep-fake videos from appearance and behavior, In: 2020 IEEE International Workshop on Information Forensics and Security (WIFS), IEEE, 2020, pp 1–6.
- [74] Agarwal, S., Farid, H., Fried, O., & Agrawala, M., Detecting Deep-Fake Videos from Phoneme-Viseme Mismatches, 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 2020 2814-2822. <https://doi.org/10.1109/CVPRW50498.2020.00338>
- [75] Cozzolino, D., Rössler, A., Thies, J., Nießner, M., & Verdoliva, L., ID-Reveal: Identity-aware Deepfake Video Detection, 2020. <https://doi.org/10.48550/ARXIV.2012.02512>
- [76] Dong, X., Bao, J., Chen, D., Zhang, T., Zhang, W., Yu, N., Chen, D., Wen, F., & Guo, B., Protecting Celebrities from Deepfake with Identity Consistency Transformer, 2022. <https://doi.org/10.48550/ARXIV.2203.01318>
- [77] Shen, D., Zhao, Y., & Quan, C., Identity-Referenced Deepfake Detection with Contrastive Learning, Proceedings of the 2022 ACM Workshop on Information Hiding and Multimedia Security, 2022, 27-32. <https://doi.org/10.1145/3531536.3532964>
- [78] Liu, B., Liu, B., Ding, M., Zhu, T., & Yu, X., TI 2 Net: Temporal Identity Inconsistency Network for Deepfake Detection, 2023 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV), 2023, 4680-4689. <https://doi.org/10.1109/WACV56688.2023.00467>
- [79] Guera, D., & Delp, E., Deepfake Video Detection Using Recurrent Neural Networks, 2018, 1-6. <https://doi.org/10.1109/AVSS.2018.8639163>
- [80] Montserrat, D. M., Hao, H., Yarlagadda, S. K., Baireddy, S., Shao, R., Horváth, J., Bartusiak, E., Yang, J., Güera, D., Zhu, F., & Delp, E. J., Deepfakes Detection with Automatic Face Weighting, 2020. <https://doi.org/10.48550/ARXIV.2004.12027>
- [81] Zheng, Y., Bao, J., Chen, D., Zeng, M., & Wen, F., Exploring Temporal Coherence for More General Video Face Forgery Detection, 2021. <https://doi.org/10.48550/ARXIV.2108.06693>
- [82] Saikia, P., Dholaria, D., Yadav, P., Patel, V., & Roy, M., A Hybrid CNN-LSTM model for Video Deepfake Detection by Leveraging Optical Flow Features, 2022. <https://doi.org/10.48550/ARXIV.2208.00788>
- [83] Rahman, A., Siddique, N., Moon, M. J., Tasnim, T., Islam, M., Shahiduzzaman, Md., & Ahmed, S., Short And Low Resolution Deepfake Video Detection Using CNN, 2022 IEEE 10th Region 10 Humanitarian Technology Conference (R10-HTC), 2022, 259-264. <https://doi.org/10.1109/R10-HTC54060.2022.9929719>
- [84] Kolagati, S., Priyadarshini, T., & Mary Anita Rajam, V., Exposing Deepfake using a deep multilayer perceptron – convolutional neural network model, International Journal of Information Management Data Insights, 2022, 2(1), 100054. <https://doi.org/10.1016/j.ijimei.2021.100054>
- [85] Thing, V. L. L., Deepfake Detection with Deep Learning: Convolutional Neural Networks versus Transformers, 2023. <https://doi.org/10.48550/ARXIV.2304.03698>
- [86] Mittal, T., Bhattacharya, U., Chandra, R., Bera, A., & Manocha, D., Emotions Don't Lie: An Audio-Visual Deepfake Detection Method using Affective Cues, Proceedings of the 28th ACM International Conference on Multimedia, 2022, 2823-2832. <https://doi.org/10.1145/3394171.3413570>
- [87] Hosler, B., Salvi, D., Murray, A., Antonacci, F., Bestagini, P., Tubaro, S., & Stamm, M. C., Do Deepfakes Feel Emotions? A Semantic Approach to Detecting Deepfakes Via Emotional Inconsistencies, 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 2021, 1013-1022. <https://doi.org/10.1109/CVPRW53098.2021.00112>
- [88] Pei, S., Wang, Y., Xiao, B., Pei, S., Xu, Y., Gao, Y., & Zheng, J., A bidirectional-LSTM method based on temporal features for deep fake face detection in videos, 2nd International Conference on Information Technology and Intelligent Control, 2022, 28. <https://doi.org/10.1117/12.2653461>
- [89] Haq, I. U., Malik, K. M., & Muhammad, K., Multimodal Neurosymbolic Approach for Explainable Deepfake Detection, ACM Transactions on Multimedia Computing, Communications, and Applications, 2023, 3624748. <https://doi.org/10.1145/3624748>
- [90] Nguyen, H. H., Fang, F., Yamagishi, J., & Echizen, I., Multi-task Learning For Detecting and Segmenting Manipulated Facial Images and Videos, 2019. <https://doi.org/10.48550/ARXIV.1906.06876>
- [91] de Lima, O., Franklin, S., Basu, S., Karwoski, B., & George, A., Deepfake Detection using Spatiotemporal Convolutional Networks, 2020. <https://doi.org/10.48550/ARXIV.2006.14749>
- [92] Li, X., Lang, Y., Chen, Y., Mao, X., He, Y., Wang, S., Xue, H., & Lu, Q., Sharp Multiple Instance Learning for Deepfake Video Detection, 2020. <https://doi.org/10.48550/ARXIV.2008.04585>
- [93] Hubálovský, Š., Trojovský, P., Bacanin, N., & K, V., Evaluation of deepfake detection using YOLO with local binary pattern histogram, PeerJ Computer Science, 2022, 8, e1086. <https://doi.org/10.7717/peerj-cs.1086>
- [94] Lu, T., Bao, Y., & Li, L., Deepfake Video Detection Based on Improved CapsNet and Temporal-Spatial Features, Computers, Materials & Continua, 2023, 75(1), 715-740. <https://doi.org/10.32604/cmc.2023.034963>
- [95] Dolla, M. S., Ruan, L., Zhu, K., & Xiao, L., Spatio-Temporal Feature Pyramid Network for Deepfake Detection, SSRN, 2023. <https://doi.org/10.2139/ssrn.4507991>

- [96] Kaddar, B., Fezza, S. A., Akhtar, Z., Hamidouche, W., Hadid, A., & Serra-Sagristà, J., Deepfake Detection Using Spatiotemporal Transformer, *ACM Transactions on Multimedia Computing, Communications, and Applications*, 2024, 3643030. <https://doi.org/10.1145/3643030>

# Makine Öğrenmesi Temelli Obezite Durum Tahmini

## Machine Learning Based Obesity Status Prediction

Ercan ÖLÇER

Kocaeli Sağlık ve Teknoloji Üniversitesi, Doğa Bilimleri ve Mühendislik Fakültesi  
Bilgisayar Mühendisliği Bölümü  
Kocaeli - Türkiye  
ercan.olcer@kocaelisaglik.edu.tr  
ORCID: 0000-0003-3786-6230

### Öz

Obezite ciddi bir halk sağlığı sorunudur ve Dünya üzerinde gittikçe artış göstermektedir. Biyolojik, fizyolojik, psikolojik ve çevresel faktörlerden etkilenen karmaşık bir konudur. Yaşam kalitesini olumsuz etkileyen bir hastalık olarak kabul edilmektedir. Yüksek tansiyon, koroner arter hastalığı, kalp krizi, uyku apnesi, nefes alma zorluğu, eklem ağrısı ve osteoartrit eklem hastalıklarının oluşmasına neden olabilir. Ayrıca çeşitli kanser türlerinin görülme riski obez bireylerde daha yüksektir. Yüksek tansiyon, yüksek kan şekeri, yüksek trigliserid seviyeleri ve düşük HDL kolesterol seviyeleri gibi faktörlerin bir araya gelmesiyle oluşan metabolik sendrom riskini de artırır. Çalışma, makine öğrenimi sınıflandırıcıları kullanarak obezite tahmini için risk faktörlerini belirlenmesini amaçlamaktadır. Makine öğrenimi yöntemleri özellikle büyük veri kümelerinin analiz edilmesi ve bu verilerden obezitenin ana belirleyici değişkenlerini saptanmasını kolaylaştırır. Bu yöntemlerin uygulanması ile risk faktörlerinin öncelikle belirlenerek takibinin kolaylaşmasını sağlayabilir. Makine öğrenimi, obeziteyle ilgili sonuçların anlaşılması ve tahmin edilmesi için umut verici bir yol sunmaktadır. Araştırmacılar, büyük veri kümelerinden ve karmaşık algoritmalarından yararlanarak obezitenin temel belirleyicilerini ve risk faktörlerini belirleyebilir ve bu da önleme ve müdahale stratejilerine bilgi sağlayabilir. Geliştirilmiş algoritmalar, gelişmiş tahmin başarımını ortaya koyar ve çeşitli veri kaynaklarının entegrasyonu, obezite tahmin modellerini daha da geliştirebilir. Bu bilgiler, küresel obezite problemine yönelik müdahalelerin geliştirilmesine rehberlik edebilir. Bu yöntemlerin uygulanması ile elde edilecek sonuçların kullanımı ile hastalık tanılama uzmanlara yardımcı olacaktır ve karar vermelerinde destek sağlayacaktır. Çalı

şmada "kaggle" ortamında temin edilen ve içeriğinde on yedi parametreyi barındıran veri seti kullanılarak gerçekleştirilen makine öğrenmesi yöntemleri kullanılmıştır. Elde edilen sınıflandırma sonuçları hastalık riski olabilecek hastaların verilerinin diyabet risklerini belirlemede kullanılabilir. Bu veri kümesi üzerinde on bir farklı makine öğrenme algoritması kullanılmıştır. Bu makalede bu yöntemler karşılaştırılarak tahminde en başarılı yöntemler belirlenmiştir. Örneklemelerde veri seti içinden eğitim ve test seti oluşturulmuştur. Algoritmaların başarımı çeşitli metriklerle karşılaştırılmıştır. Ayrıca en başarılı birkaç algoritma değişkenlerin bazılarının ince ayar yapılarak başarımı artırılmıştır. Uygulanan sınıflandırıcı algoritmalarından en başarılı başarımlar, Gradient Boost ve XGBoost kullanan modeller olmuştur. Bu modeller test verileri üzerinde %97 doğruluk değerini elde etmiştir. Literatür taramasında bu çalışmada elde edilen sonucun en iyi sonuç olduğu görülmektedir. Kısıtlı özelliklerle obezite konusunun çalışıldığı makalelere göre farklı özellikleri de dikkate alan bir veri kümesi olması ve on bir farklı modelleme ile obezite sınıflandırmalarının makine öğrenmesinin yapılması açısından da anlamlı bir çalışma olduğu düşünülmektedir.

**Anahtar sözcükler:** Makine Öğrenmesi, Obezite, Tanı

### Abstract

Obesity is a serious public health problem and is increasing worldwide. It is a complex issue affected by biological, physiological, psychological and environmental factors. It is recognized as a disease that negatively affects quality of life. It can cause high blood pressure, coronary artery disease, heart attack, sleep apnea, breathing difficulties, joint pain and osteoarthritis joint diseases. In addition, the risk of various types of cancer is higher in obese individuals. It also increases the risk of metabolic syndrome, a combination of factors such as high blood pressure, high blood sugar, high triglyceride levels and low HDL cholesterol levels. The study aims to

*identify risk factors for obesity prediction using machine learning classifiers. Machine learning methods facilitate the analysis of large datasets and the identification of key determinants of obesity from these data. By applying these methods, risk factors can be identified first and follow-up can be facilitated. Machine learning offers a promising way to understand and predict obesity-related outcomes. By leveraging large datasets and complex algorithms, researchers can identify key determinants and risk factors of obesity, which can inform prevention and intervention strategies. Improved algorithms reveal improved prediction performance, and the integration of various data sources can further enhance obesity prediction models. This information can guide the development of interventions to address the global obesity problem. Using the results obtained by applying these methods will help experts in disease diagnosis and support decision-making. It is considered to be a meaningful study in terms of being a dataset that takes into account different features compared to the articles where obesity is studied with limited features and machine learning of obesity classifications with eleven different modeling.*

**Keywords:** Machine Learning, Obesity, Diagnosis

## 1. Giriş

Obezite, sağlığı olumsuz etkilere neden olabilecek derecede yüksek oranda vücuttaki yağın birikmesi olarak tanımlanabilir. Kişi ağırlığının boyunun karesine bölünmesiyle vücut kitle endeksi elde edilir. Bu endeks "BMI" olarak tanımlanır. BMI istatistiklerde aşırı kilo ve obezitenin bir göstergesi olarak tanımlanır [1]. Yetişkinlerde 30 kg/m<sup>2</sup> aşıldığında kişi obez olarak ifade edilir. Bazı Asya ülkelerinde daha düşük değerler görülmektedir [2].

Obezite, hayatı zorlaştıran önemli bir nedendir. Tip 2 diyabet, kardiyovasküler hastalıklar, uyku apnesi ve kanser tipleri gibi hastalıkların gelişmesine sebep olmaktadır [3]. Obezite oranları Dünya'da 1975 yılından beri üç kat arttığı ifade edilmektedir. 2022 yılında dünyadaki her 8 kişiden 1'i obez olduğu kaydedilmiştir. Dünya çapında yetişkin obezitesi 1990'dan bu yana iki kattan fazla, ergen obezitesi ise dört kat artmıştır. 2,5 milyar yetişkin (18 yaş ve üzeri) fazla kilolu olduğu ifade edilmiştir. Bunlardan 890 milyonu obeziteyle yaşadığı raporlanmıştır. 2022 yılında 18 yaş ve üzeri yetişkinlerin %43'ü aşırı kilolu, %16'sı ise obezdir. 5 yaş altı 37 milyon çocuk aşırı kilolu olduğu, 5-19 yaş arası 390 milyondan fazla çocuk ve ergen aşırı kilolu ve bunların 160 milyonu obez olduğu ifade edilmiştir [4].

Dünya genelinde obezitenin, önümüzdeki yıllarda artmaya devam edeceği ifade edilmektedir. Dünya Obezite Federasyonu'nun yayımladığı rapora göre, 2030 yılına kadar obez insan sayısının 1 milyar artması beklenmektedir. Bu durum, sağlık sistemleri ve bireyler için önemli bir sorun oluşturmaktadır. Avrupa ülkelerinde yetişkinlerin yüzde 30'unun obez olacağı öngörülmektedir. Aynı zamanda 2010 yılına göre 2030 yılındaki obez insan sayısının dünya genelinde 2 katına çıkması beklenmektedir [5].

Son zamanlarda makine öğrenme yöntem ve modelleri, sağlıkta, özellikle hastalıkların sınıflandırılmasında yaygın olarak kullanılmaya başlanmıştır. Sağlık hizmetlerinin yaygınlaşması ve giyilebilir teknolojiler aracılığı ile artık yüksek miktarda sağlık verisi toplanabilmektedir. Ancak yüksek veri yoğunluğu nedeniyle önemli miktarda verinin değerlendirilebilmesi için makine öğrenmesi, derin öğrenme gibi yeni yöntem ve akıllı sistemlere ihtiyaç duyulmaktadır. Hastalıkların değerlendirilmesinde makine öğrenme algoritmaları kullanılabilir. Obezitenin hem erken tespitinde hem de izlenmesinde bu algoritmalar kullanılabilir. Ayrıca makine öğrenmesi yöntemleri insan nüfusunun obezite tahmin etme yöntemi, başka hastalıkların da önemli sebeplerinden biri olarak kabul edilen obezitenin önüne geçilmesi için uygulanabilir.

Bu makalede "Kaggle" obezite veri kümesi kullanılarak on bir farklı makine öğrenmesi algoritması uygulanmıştır. Test verilerinin obezite sınıflarının tahmin edilmesinde kullanılması ile doğruluk, kesinlik, duyarlılık ve F1 ölçüsü başarımları hesaplamaları ile algoritmaların başarımları hesaplanmıştır. Başarımlar karşılaştırma tablosunda yer alan en iyi üç skor üzerinde hiper parametre iyileştirmesi yapılarak daha yüksek başarımlara ulaşılmıştır. İlerleyen kısımlarda ise, önceki yıllarda literatürde geçen benzer çalışmaların başarımlarını karşılaştırma yapılmıştır.

Kısıtlı özelliklerle obezite konusunun çalışıldığı makalelere göre farklı özellikleri de dikkate alan bir veri kümesi olması ve on bir farklı modelleme ile obezite sınıflandırmalarının makine öğrenmesinin yapılması açısından da anlamlı bir çalışma olduğu düşünülmektedir.

## 2. Veri Kümesi ve Yöntem

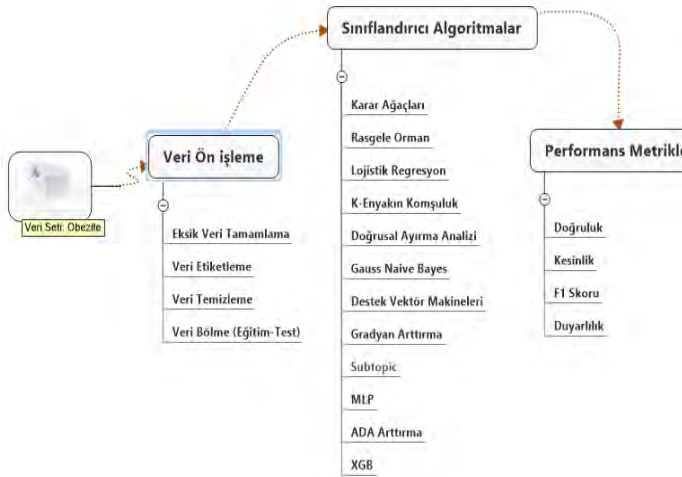
### 2.1 Önerilen Model

Şekil-1'de bu makaleye konu olan çalışmada izlenen yol özetlenmiştir. "csv" Formatında Intel i7-1260p işlemcili 64GB RAM'li lokal makineye indirilen veri kümesi Python dilinde yazılan bir yazılım aracılığı ile öncelikle veri ön işleme yöntemleri uygulanarak elden geçirilmiştir. Veri ön işleme aşamasında verilerin tekrar eden satırların temizlenmesi ve eksik verilerin tamamlanması işlemlerine tabi tutulmuştur. Ayrıca verilerin sınıflandırıcı algoritmalar tarafından işlenebilmesi için gerekli etiketleme işlemleri gerçekleştirilmiştir. Sonraki aşamada ise makine öğrenmesi süreçlerinde kullanılmak üzere veri eğitim ve test verileri olarak (0,8/0,2) oranı ile ikiye ayrılmıştır. Sınıflandırıcı algoritmalarına uygulanarak başarımları doğruluk, tutturma (precision), bulma (recall) ve F1-ölçüsü (F1Score) sınıflarında karşılaştırmaya tabi tutulmuştur.

### 2.2. Veri Kümesi

Kaggle web sitesinden elde edilen veri kümesi [6], bireylerin demografik özellikleri, fiziksel özellikleri ve yaşam tarzı alışkanlıkları hakkında kapsamlı bilgi sağlayarak obezite durum analizini ve tahminini kolaylaştırmayı amaçlamaktadır. Yaş, cinsiyet, boy, kilo, fiziksel aktivite düzeyi ve obezite kategorisi gibi temel değişkenleri içerir ve obezite sonuçlarını etkileyen faktörlere ilişkin değerli bilgiler sunar.



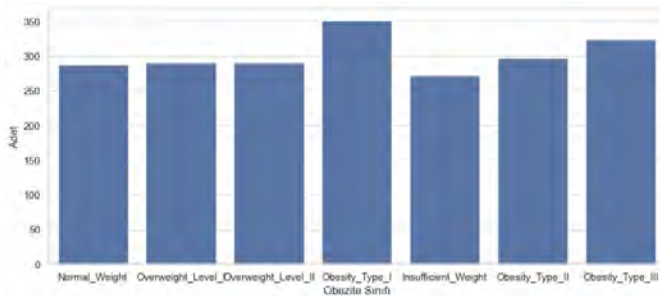


Şekil-1. Önerilen Model

Çizelge-1. Obezite Sınıfları

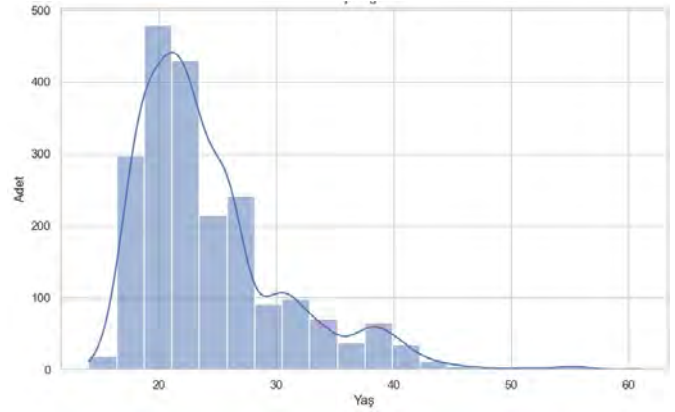
Obezite ("NObeyesdad")
Yetersiz kilo (insufficient wight)
Normal kilo (normal wight)
Aşırı kilolu Düzey 1 (overweight wight level 1)
Aşırı kilolu Düzey 2 (overweight wight level 2)
Obezite Tip 1 (obesity type 1)
Obezite Tip 2 (obesity type 2)
Obezite Tip 3 (obesity type 3)

Kullanılan veri kümesi farklı yaş ve cinsiyetten 17 özelliği içeren 2111 kaydı barındırmaktadır. Veri kümesinde obezite durumunu gösteren ve bağımlı değişken olarak tanımlanan "NObeyesdad" isimli veri alanında "yetersiz kilo, normal kilo, aşırı kilolu düzey I, fazla kilolu düzey II, obezite tip I, obezite tip II ve obezite tip III" olarak tanımlanmış ve Çizelge-1'de gösterilmiştir. Aynı şekilde bağımsız değişkenlerin listesi de Çizelge-2'de açıklamaları ile birlikte görülebilir.



Şekil-2. Obezite Sınıf Dağılımı

Obezite sınıflarının dağılımı Şekil-2'de yer almaktadır. Genel olarak eşit bir dağılım olduğu görülmektedir.

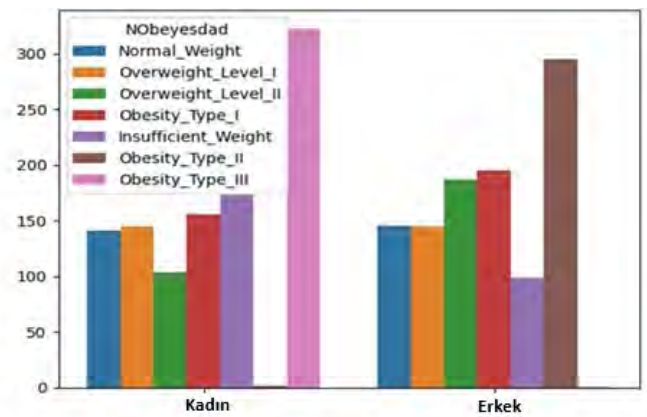


Şekil-3. Yaş Dağılımı

Katılımcıların yaş dağılımları Şekil-3'te görüldüğü gibidir. Ağırlıklı olarak katılımcılar 18-22 yaşları arasında yer almaktadır.

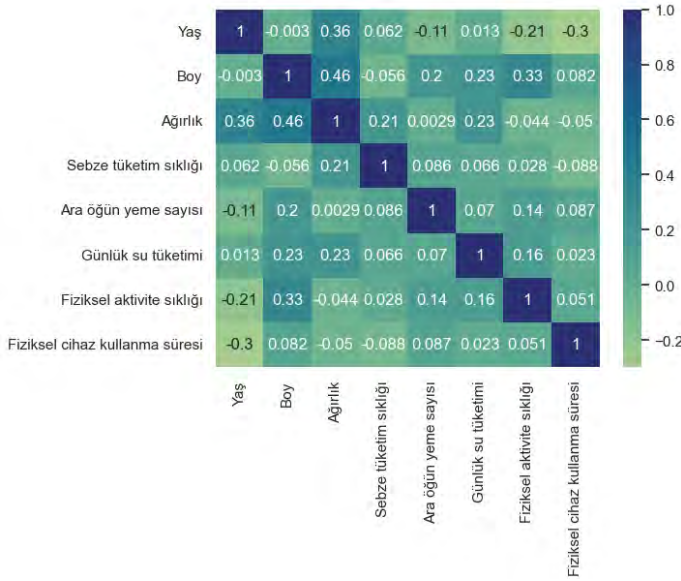
Çizelge-2. Veri Kümesi

Değişken	Açıklama
Gender	Cinsiyet
Age	Yaş
Height	Boy
Weight	Kilo
Family history overweight	Ailede fazla kilo öyküsü
FAVC	Yüksek kalorili gıda tüketimi
FCVC	Sebze tüketim sıklığı
NCP	Ara öğün yeme sayısı
CAED	Öğünler arası yeme sıklığı
CH20	Günlük su tüketimi
CALC	Alkol tüketimi
SMOKE	Sigara tüketimi
SCC	Kalori tüketimi
FAF	Fiziksel aktivite sıklığı
TUE	Fiziksel cihaz kullanma süresi
MTRANS	Kullanılan ulaşım türü



Şekil-4. Obezite Sınıfları ve Cinsiyet Dağılımı

Obezite ile cinsiyet arasındaki dağılım Şekil-4'te görülmektedir. Buna göre obezite 3 tipi kadınlarda, obezite 2 tipi erkeklerde yüksek sayıda olduğu anlaşılmaktadır.



Çizelge-3. Isı Haritası

Çizelge-3'te yer alan ısı haritasında boy ve ağırlık arasında 0,46 oranında yüksek bir ilişki olduğu anlaşılmaktadır. Ardından yaş ve ağırlık arasında 0,36, boy ile fiziksel aktivite

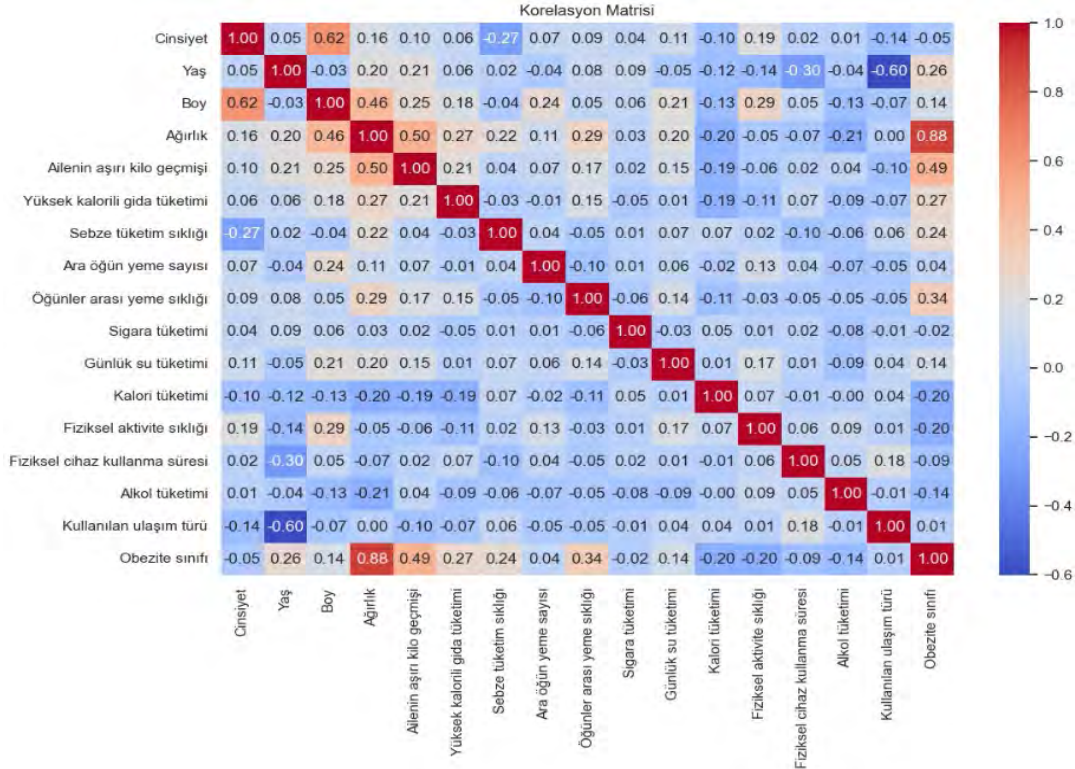
sıklığı değeri (FAF) 0.33 oranı ile doğrusal ilişkili olduğu tespit edilmektedir.

Çizelge-4'te yer alan korelasyon matrisinde bağımlı ve bağımsız değişkenlerin korelasyonları takip edilebilir. Buna göre obezite sınıflarını gösteren ve "NOBeyesdad" isimli bağımlı değişkenin ağırlık ve ailede fazla kilo öyküsü ile yüksek ilişkili olduğu gözlemlenmiştir.



Şekil 5.Bağımlı Değişken("NOBeyesdad") ile ilişki

Çizelge-4.Korelasyon Matrisi



Şekil-5'te bağımlı değişken olan obezite sınıflarının (NOBeyesdad) diğer bağımsız değişkenlerle olan ilişkisi grafik olarak sıralanmaktadır. Ağırlık değişkeni en yüksek korelasyona sahipken kalori tüketimi en az ve ters korelasyona sahiptir.

#### 2.4.Veri Kümesinin Eğitim ve Sınama Veri Kümesine Bölünmesi ve Ölçünlü Biçime Getirilmesi

Makine öğrenmesi modellerinin yapılabilmesi ve öğrenme sürecinin gerçekleştirilebilmesi için veri kümesi rasgele seçim yapılarak %80 eğitim seti-%20 sınama kümesi olmak üzere ikiye bölünmüştür. Ayrıca verilerin

değerlendirilmesinde ölçünleme yöntemi kullanılarak veriler minimum ve maksimum değerlerine göre oranlanıp ölçünlü verilere dönüştürülmüştür. Bunun için numpy kütüphanesi kullanılmış ve oranlama gerçekleştirilerek verilerin [0-1] aralığında dağılımı gerçekleştirilmiştir.

## 2.5. Model Değerlendirme

Makine öğrenmesi sürecinde on bir farklı sınıflandırma algoritması kullanılmıştır. Modellerin karşılaştırılması ve sınıflandırmaların değerlendirilmesi için eğitim keskinlik ölçüsü, doğruluk ölçüsü, karışıklık matrisi sınıflandırma raporu ölçüleri kullanılmıştır. Karışıklık matrisi, sınıflandırma modelinin başarımını özetleyen bir çizelgedir ve modelin farklı sınıflar genelinde hedef değişkeni ne kadar iyi tahmin ettiğine dair kapsamlı bir genel bakış sağlar. Doğruluk ile modelimizin doğru sınıflandırma yüzdesini verir. Ancak tek başına kullanılmaz. Tutturma ile tüm olumlu tahminlerden hangilerinin olumlu olduğuna dair bilgi verir. Bulma ile tüm gerçek değerlerden kaçının pozitif olmasını doğru olarak tahmin edildiği gösterilir. F1 ölçüsü ise bize tutturma ve bulma değerlerinin harmonik ortalamasını göstermektedir

$$\text{Doğruluk} = (\text{TP} + \text{TN}) / (\text{TP} + \text{FP} + \text{FN} + \text{TN})$$

$$\text{Tutturma} = \text{TP} / (\text{TP} + \text{FP})$$

$$\text{Bulma} = \text{TP} / (\text{TP} + \text{FN})$$

$$\text{F1 Değeri} = 2 * \text{keskinlik} * \text{duyarlılık} / (\text{keskinlik} + \text{duyarlılık})$$

Obezite sınıflarının her biri için sınıflandırma raporu üretilerek en başarılı olan model belirlenmiştir.

## 2.6. Modeller

Bu çalışmada on bir farklı sınıflandırma algoritması kullanılmıştır. Bunlar sırasıyla Decision Tree (DTC), Random Forest (RF), Logistic Regression (LR), K-Nearest Neighbors (KNN), Linear Discriminant Analysis (LDA), Gaussian NB (GNB), SVC, Gradient Boosting (GBC), MLP, Ada Boost (ADA), Extreme Gradient Boost (XGB)'dir.

Karar Ağaçları Sınıflandırma Algoritması (Decision Tree-DTC), denetimli makine öğrenimi uygulamalarında kullanılan ağaç yapısına sahip bir algoritmadır. Karar ağaçları, düğümler, dallar ve yapraklardan oluşur [7]. Basit yapısı sebebiyle karar ağaçları sıkça tercih edilmektedir.

Rasgele Orman Sınıflandırma Algoritması (Random Forest - RF), denetimli makine öğreniminde kullanılan algoritmalarından biri olup verileri sınıflandırmak veya regresyon yapmada kullanılır. Bu algoritma, veri kümesindeki özellikleri kullanarak ağaç yapısını oluşturur ve sınıflandırma veya regresyon görevlerini gerçekleştirir [8].

Lojistik Regresyon Sınıflandırma Algoritması (Logistic Regression-LR) bir lojistik modelin değişkenlerinin tahmin edilmesidir [9]. İkili lojistik regresyonda, bir gösterge değişkeni tarafından kodlanan tek bir ikili bağımlı değişken vardır ve "0" ve "1" olarak etiketlenir, bağımsız değişkenlerin her biri ikili veya sürekli değişken olabilir. Etiketleme, log-oranları olasılığa dönüştüren fonksiyon lojistik fonksiyondur [10].

K-En Yakın Komşuluk (K-Nearest Neighbors - KNN), 1951'de Evelyn Fix ve Joseph Hodges tarafından geliştirilen [11] ve Thomas Cover tarafından ise genişletilen parametrik olmayan denetimli bir öğrenme algoritmasıdır [12]. Sınıflandırma ve regresyon için kullanılır. Her iki durumda da girdi, bir veri kümesindeki en yakın k eğitim örneğinden oluşur. Çıktı, k-NN'nin sınıflandırma veya regresyon için kullanılıp kullanılmamasına bağlıdır.

Doğrusal Ayırma Analizi (Linear Discriminant Analysis - LDA), özneliklerin doğrusal birleşimini bularak veriyi sınıflara ayırmaya yarayan bir sınıflandırma algoritmasıdır [13]. Doğrusal ayırma analizi, değişkenlerin, veriyi en iyi açıklayan doğrusal birleşiminin incelenmesi bakımından temel bileşen analizi (TBA) ve faktör analizi ile yakından ilişkilidir [14]. 1936 yılında R. A. Fisher tarafından geliştirilen bir sınıflama yöntemidir. Basit olmasına rağmen kompleks problemlerde iyi sonuçlar üreten bir modeldir.

Gauss Naive Bayes Sınıflandırıcı Algoritması (Gaussian NB - GNB), hedef sınıfa göre özelliklerin koşullu olarak bağımsız olduğunu varsayan bir sınıflandırıcıdır. Bu sınıflandırıcılar en basit Bayes ağ modelleri arasındadır. İstatistik literatüründe bu sınıflandırıcı modeli, basit Bayes ve bağımsız Bayes gibi çeşitli isimler altında bilinmektedir [15]. 2006 yılında diğer sınıflandırma algoritmalarıyla yapılan kapsamlı bir karşılaştırma, Bayes sınıflandırmasının diğer yaklaşımlardan daha iyi başarımlar gösterdiğini ifade etmiştir [16]. Naive Bayes'in bir avantajı, sınıflandırma için gerekli parametreleri tahmin etmek için yalnızca küçük miktarda eğitim verisine ihtiyaç duymasındır. Gaussian NB'de ise her bir sınıfla ilişkili sürekli değerlerin Gauss dağılımına göre olmasıdır.

Destek Vektör Makineleri Sınıflandırma Algoritması (Suport Vector Machine - SVM), Makine öğreniminde vektör ağlarını da destekleyen destek vektör makineleri, sınıflandırma ve regresyon analizi için verileri analiz eden ilişkili öğrenme algoritmalarına sahip ve denetlenen bir modeldir. AT&T Bell Laboratuvarlarında Vladimir Vapnik ve arkadaşları geliştirilmiştir [17].

Gradyan Artırma Sınıflandırma Modeli (Gradient Boosting - GBC), denetimli öğrenme modelleri için yüksek başarımlar sağlayan algoritmalarından biridir. Tipik olarak basit karar ağaçları olan, veriler hakkında çok az varsayımda bulunan modeller gibi zayıf tahmin modellerin birleşimi şeklinde tahmin modeli önerir [18]. Bir karar ağacı zayıf öğrenen olduğunda, ortaya çıkan algoritmaya gradyan destekli ağaçlar adı verilir; genellikle rastgele ormandan daha iyi başarımlar gösterir [19].

Çok Katmanlı Algılayıcı Algoritması (Multi Layer Perceptron - MLP), doğrusal olarak ayıramayan verileri ayırt edebilmesiyle dikkat çeken, en az üç katman halinde organize edilmiş, doğrusal olmayan bir tür aktivasyon işlevine sahip tamamen bağlı nöronlardan oluşan modern ileri beslemeli yapay sinir ağının adıdır [20].

Adaptif Arttırma Algoritması (Adaptive Boosting - ADA) Yoav Freund ve Robert Schapire tarafından 1995 yılında formüle edilen istatistiksel bir sınıflandırma algoritmasıdır [21].

Gradyan Arttırma Algoritması (Extreme Gradient Boost – XGB) Derin Makine Öğrenimi Topluluğu (DMMLC) grubunun bir parçası olarak Tianqi Chen tarafından geliştirilmiştir. XGBoost, özellikle büyük veri setleri ve karmaşık veri yapıları üzerinde çalışırken yüksek başarımlar ve hız sunar. XGBoost, karar ağaçlarına göre daha yüksek doğruluk sağlaması nedeniyle tercih edilmektedir.

### 3. Deneysel Sonuçlar ve Tartışma

Yapılan modelleme çalışmasından elde edilen doğruluk değerleri aşağıdaki tabloda görülmektedir:

**Çizelge 5. Modellerin Doğruluk Değerleri**

Model Adı	Eğitim ve Sınama Süresi (ms)	Doğruluk
DTC	19	0.53
<b>RF</b>	<b>446</b>	<b>0.96</b>
LR	25	0.50
KNN	331	0.89
LDA	23	0.90
GNB	19	0.64
SVC	792	0.57
GBC	2798	0.94
MLP	3218	0.63
ADA	184	0.27
<b>XGB</b>	<b>561</b>	<b>0.96</b>

Çizelge-5'e göre RF ve XGB algoritmaları 0,96 ile en yüksek doğruluk değerini alırken GBC algoritması 0,94 değeri ile üçüncü en yüksek değer olarak sıralanmaktadır. Ayrıca eğitim ve test verileri üzerinde geçen süreler yine çizelgede görülmektedir. Doğruluğu yüksek olan modellere ait süreler 0,5 saniyenin altında gerçekleşmiştir.

RF algoritmasına göre elde edilen sınıflandırma raporu Çizelge-6'da verilmiştir: Test verileri tahmin değerleri %96 doğruluk değerini almıştır. Tabloda ayrıca her obezite sınıfı için elde edilen diğer başarımlar verileri görülmektedir.

**Çizelge-6. RF Algoritması Göre Başarımlar Değerleri**

Obezite Sınıfı	tutturma	bulma	f1-değeri	destek
Yetersiz kilo	0.90	0.97	0.93	62
Normal kilo	1.00	0.96	0.98	56
Aşırı kilolu Düzey 1	0.93	0.89	0.91	56
Aşırı kilolu Düzey 2	0.98	0.96	0.97	50
Obezite Tip 1	0.99	0.97	0.98	78
Obezite Tip 2	0.97	0.98	0.97	58
Obezite Tip 3	1.00	1.00	1.00	63
accuracy			0.96	423
macro avg	0.96	0.96	0.96	423
weighted avg	0.97	0.96	0.96	423

XGB algoritmasına göre elde edilen sınıflandırma raporu aşağıdaki Çizelge-7'de verilmiştir: Çizelgeye göre, obezite sınıflarına ait test verilerinin tahmin değerleri %96 doğruluğa ulaşmıştır. Tabloda ayrıca her obezite sınıfı için elde ettiği diğer başarımlar verileri görülmektedir.

**Çizelge-7. XGB Algoritmasına Göre Başarımlar Değerleri**

Obezite Sınıfı	tutturma	bulma	f1-değeri	destek
Yetersiz kilo	0.95	0.89	0.92	62
Normal kilo	0.93	1.00	0.97	56
Aşırı kilolu Düzey 1	0.91	0.95	0.93	56
Aşırı kilolu Düzey 2	1.00	0.98	0.99	50
Obezite Tip 1	0.97	0.96	0.97	78
Obezite Tip 2	0.97	0.97	0.97	58
Obezite Tip 3	1.00	1.00	1.00	63
accuracy			0.96	423
macro avg	0.96	0.96	0.96	423
weighted avg	0.96	0.96	0.96	423

GBC algoritmasına göre elde edilen sınıflandırma raporu aşağıdaki Çizelge-8'de verilmiştir: Çizelgeye göre, obezite sınıflarına ait test verileri için doğruluk değeri %96'ya ulaşmıştır. Tabloda ayrıca her obezite sınıfı için elde ettiği diğer başarımlar verileri görülmektedir.

**Çizelge-8. GBC Algoritmasına Göre Başarımlar Değerleri**

Obezite Sınıfı	tutturma	bulma	f1-değeri	destek
Yetersiz kilo	0.91	0.85	0.88	62
Normal kilo	0.93	0.96	0.95	56
Aşırı kilolu Düzey 1	0.88	0.91	0.89	56
Aşırı kilolu Düzey 2	0.94	0.96	0.95	50
Obezite Tip 1	0.97	0.92	0.95	78
Obezite Tip 2	0.93	0.98	0.96	58
Obezite Tip 3	1.00	1.00	1.00	63
accuracy			0.94	423
macro avg	0.94	0.94	0.94	423
weighted avg	0.94	0.94	0.94	423

Ancak doğruluk değerlerinin artırılması amacıyla algoritma giriş değerlerinde (Hiper Parametre) ince ayar yapılarak başarımlar iyileştirme çalışması yapılmıştır. Makine öğrenimi modellerinde hiper parametreler bulunur. Hiper parametreler, bir makine öğrenimi modelinin belirli bir görev veya veri kümesi için özelleştirilmesine olanak tanıyan seçim veya yapılandırma noktalarıdır.

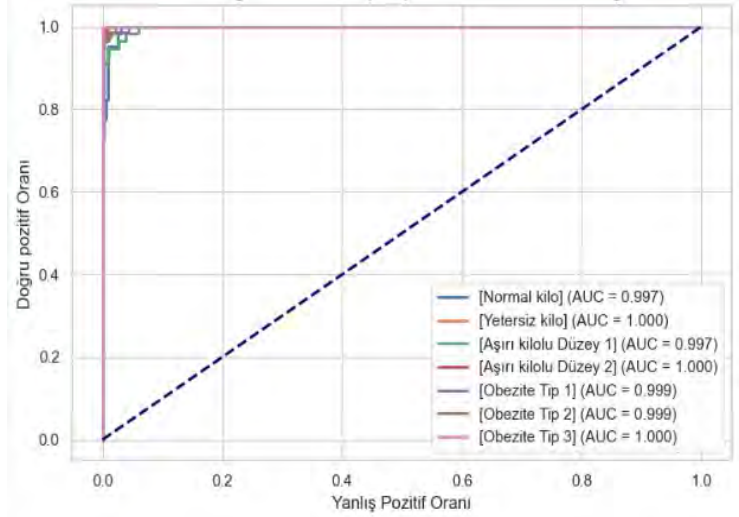
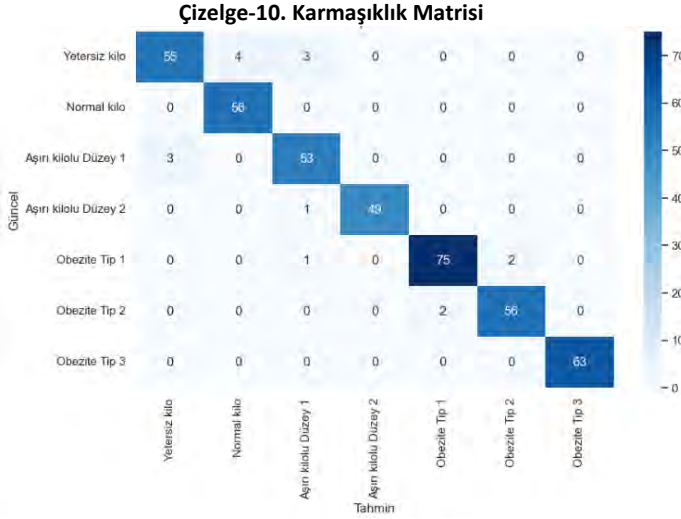
En iyi parametreleri belirlemek için her tekrarda farklı rasgelelik ile Katmanlı K-Katlama yöntemi uygulanmıştır. Katlama sayısı 5 tekrar sayısı 2 olarak seçilmiştir. Daha sonra verilen hiper parametre rasgele arama algoritması ile taranarak en iyi hiper parametreleri tespit edilmiştir.

Elde edilen yeni doğruluk değerleri ve ilgili parametreleri aşağıdaki Çizelge-9'da yer almaktadır:

**Çizelge-9. En İyi Hiper Parametre Değerleri**

Model	Random Forest	Gradient Boost	XG Boost
<b>N Estimator</b>	1000	500	100
<b>Max features</b>	Log2	3	6
<b>Learning rate</b>	0.1	0.1	0.1
<b>Önceki Doğruluk</b>	0.96	0.94	0.96
<b>İyileştirilmiş Doğruluk</b>	0.96	0.97	0.97
<b>Geçen Süre (sn)</b>	29	1466	152

Çizelgeye göre en iyi hiper parametre değerleri hesaplanarak iyileştirilmiş başarımlar doğruluk değerlerine ulaşmıştır. Gradient Boost ve XGBoost test verileri üzerinde %97 doğruluk oranını yakalamıştır. Buna göre hesaplama süreleri Çizelge-9'a eklenmiştir. En iyi başarımlar modeline göre (XGBoost) Çizelge-10. Karmaşıklık Matrisi Çizelge-10'da karmaşıklık matrisi gösterilmektedir. Güncel (gerçek) değerlere göre tahmin değerlerin durumu bu tablodan görülebilir. Küçük hata ile yüksek sayıda sınıf tahmini yapıldığı anlaşılmaktadır.



Şekil-6. XGBoost ROC Eğrisi

Farklı modellerde elde edilen başarımlar verilerine göre bu çalışmada Gradient Boost ve XGBoost sınıflandırıcı kullanan iki modelin daha iyi sonuçlar ürettiği saptanmıştır.

#### 4.Sonuç

Bu çalışmada, Kaggle web sitesinden açık kaynak veri kümesi olarak erişilen obezite veri kümesi kullanılmıştır. Veri kümesi çeşitli sınıflandırma algoritmalarıyla eğitim ve test süreçlerinden geçirilmiştir. Veri kümesi rasgele seçilerek ikiye bölünmüş ve elde edilen iki gruptan biri eğitim seti olarak makine öğrenmesinde kullanılmıştır. Diğer grup algoritmanın doğruluğunu görebilmek için test veri kümesi olarak kullanılmıştır. Test veri kümesi çapraz doğrulama modeli ile doğrulanmıştır. On bir farklı sınıflandırma algoritması kullanılarak elde edilen modellerde eğitim ve testler gerçekleştirilmiş ve elde edilen başarımlar verileri bir tabloda sunularak karşılaştırılmıştır. Tabloda verilen ve başarımlar en yüksek üç model tekrar ele alınarak en iyi doğruluk değerini sağlayan parametreler saptanmıştır. İyileştirme sonrası iki model ile elde edilen başarımlar sonucu literatür taramasında elde edilen diğer başarımlar sonuçları ile karşılaştırılmıştır. Sonuç olarak çalışmada oluşturulan ve Gradient Boost ve XGBoost sınıflandırıcı kullanan iki modelin test verileri üzerinde %97 doğruluk oranını yakalayarak en yüksek skorları elde ettiği görülmüştür. Bu çalışmayla aynı zamanda birçok çalışmada kısıtlı özelliklerle obezite ilişkisinin çalışıldığı makalelere göre farklı özellikleri de dikkate alan bir veri kümesi olması nedeniyle anlamlı olduğu düşünülmektedir. Farklı özellikler ile obezite sınıf ilişkisine bakılmıştır. Aynı zamanda bu çalışma on bir farklı modelleme ile obezite sınıflandırmalarının makine öğrenmesinin yapılması açısından da anlamlı olduğu düşünülmektedir.

Aynı şekilde kullanılan modele ait ROC eğrisi Şekil-6'da verilmiştir. Bu eğride çoklu sınıflandırma eğrilerinin hızlı bir şekilde yükselerek "1" değerine yakınsadığı ve yanlış pozitif oranının tüm sınıflar içinde oldukça küçük olduğu görülmektedir.

Çizelge 11'de literatür taramalarında yer alan ve obezite verilerinin makine öğrenmesi modelleri ile bu makalede yer alan model karşılaştırmaları görülmektedir. Tabloda 2111 kayıt ve 17 özellik olarak gösterilen veri kümeleri makaleler incelendiğinde farklı kaynaklardan temin edilse de bu çalışmada kullanılan veri kümesi ile aynı kaynaktan olduğu anlaşılmaktadır.

Tabloya baktığımızda, Turan [22], RF ve KNN modellerini deneyerek %94'lük doğruluk oranına erişse de veri kümesinde korelasyonu düşük değişkenlerin çıkarılması ve kullanılan optimizasyon parametrelerin yeterli düzeyde olmaması bu çalışmaya göre doğruluğun nispeten düşük kalmasına neden olmuştur. Yine, Cuhadar ve ark. [23], yaptığı çalışma yüksek doğruluk oranı elde etse bile günlük kişisel alışkanlıkları kullanan veri kümesinden farklı olarak kişisel kan değerlerine dayalı olması nedeniyle obezite tahmin etme konusuna farklı bir yaklaşım olarak değerlendirilebilir.

**Çizelge 11.Kaynak Araştırma Çalışmaları ile Başarım Karşılaştırması**

Çalışma Adı	Veri kümesi	Model	Doğruluk
Five Machine Learning Supervised Algorithms for The Analysis and the Prediction of Obesity [24]	2111 kayıt 17 özellik	RF	91
Estimation of Obesity Levels Based on Decision Trees [25]	2111 kayıt 17 özellik	XGBoost	86
Estimation of Obesity Levels with a Trained Neural Network Approach optimized by the Bayesian Technique [26]	2111 kayıt 17 özellik	DL	93
OBESYE: Interpretable Diet Recommender for Obesity Management using Machine Learning and Explainable AI [27]	19-95 yaş arası 146 hasta verileri	LightGBM	86
Using machine learning to predict obesity in high school students [28]	Tennessee eyaletinde 2015 anket verileri	KNN	89
Machine Learning Approach for the Early Prediction of the Risk of Overweight and Obesity in Young People [29]	İngiltere Milenyum Kohort Çalışması verileri	MLP	90
Machine learning approaches for the prediction of obesity using publicly available genetic profiles [30]	164 adet genetik profil veri	SVM	90
Optimize Edilmiş Denetimli Öğrenme Algoritmaları ile Obezite Analizi ve Tahmini [22]	2111 kayıt 17 özellik	RF	94
Deep Learning-Based Prediction Of Obesity Levels According To Eating Habits And Physical Condition [31]	2111 kayıt 17 özellik	CNN	82
Predictive Analytics: The Application of J48 Algorithm on Grocery Data to Predict Obesity [32]	35 hane halkından elde edilen 8273 adet veri	J48 decision tree	89
Obesity level prediction based on data mining techniques [33]	2111 kayıt 17 özellik	MLP	95
Obesity Prediction Using Ensemble Machine Learning Approaches. [34]	Belirtilmemiş	RF	89
Detection of Obesity Stages Using Machine Learning Algorithms [35]	2111 kayıt 17 özellik	RF	96
<b>Yapılan Çalışma</b>	<b>2111 kayıt 17 özellik</b>	<b>GB, XGB</b>	<b>97</b>

## Kaynakça

- [1] Khanna D., Peltzer C., Kahar P. ve Parmar, M. S. "Body mass index (BMI): a screening tool analysis," *Cureus*, cilt 14, 2022.
- [2] Tan K. C. B. ve Ark., "Appropriate body-mass index for Asian populations and its implications for policy and intervention strategies," *The lancet*, 2004.
- [3] Haslam D. W. ve James W. P. T., "Obesity," *The Lancet*, cilt 366, pp. 1197-1209, 2005.
- [4] *World Health Organization, World Health Organization*, 2024.
- [5] *World obesity day atlases: Obesity Atlas 2024 (no date) World Obesity Federation Global Obesity Observatory*, 2024.
- [6] Palechor F. M, Manotas A. *Estimation of obesity levels UCI dataset*, Kaggle, 2021.
- [7] Quinlan J. R., "Induction of Decision Trees," *Machine Learning*, cilt 1, pp. 81-106, 1986.
- [8] Breiman L., "Random Forests," *Machine Learning*, vol. 45, pp. 5-32, 2001.
- [9] Tolles J. ve Meurer W. J., "Logistic Regression: Relating Patient Characteristics to Outcomes," *JAMA*, cilt 316, p. 533, August 2016.
- [10] Hosmer D. W., Lemeshow S. ve Sturdivant R. X., "Applied Logistic Regression: Hosmer/Applied Logistic Regression," 2005.
- [11] Silverman B. W. ve Jones M. C., "E. Fix and J.L. Hodges (1951): An Important Contribution to Nonparametric Discriminant Analysis and Density Estimation: Commentary on Fix and Hodges (1951)," *International Statistical Review / Revue Internationale de Statistique*, cilt 57, p. 233, December 1989.
- [12] Cover T. M. ve Hart P. E., "Nearest neighbor pattern classification," *IEEE Trans. Inf. Theory*, cilt 13, pp. 21-27, 1967.
- [13] Doğan M., Orman A., Örcü M. ve Örcü H., "Çok gruplu sınıflandırma problemlerine regresyon analizi ve matematiksel programlama tabanlı yeni bir yaklaşım," *Gazi Üniversitesi Mühendislik-Mimarlık Fakültesi Dergisi*, June 2019.
- [14] Martinez A. M. ve Kak A. C., "PCA versus LDA," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, cilt 23, p. 228-233, 2001.
- [15] Hand D. J. ve Yu K., "Idiot's Bayes: Not So Stupid after All?," *International Statistical Review / Revue Internationale de Statistique*, cilt 69, p. 385, December 2001.
- [16] Caruana R. ve Niculescu-Mizil A., "An empirical comparison of supervised learning algorithms," %1 içinde *Proceedings of the 23rd International Conference on Machine Learning*, New York, NY, USA, 2006.
- [17] Cortes C. ve Vapnik V., "Support-vector networks," *Machine Learning*, cilt 20, p. 273-297, September 1995.
- [18] Madeh Piryonesi S. ve El-Diraby T. E., "Using Machine Learning to Examine Impact of Type of Performance Indicator on Flexible Pavement Deterioration Modeling," *Journal of Infrastructure Systems*, cilt 27, June 2021.

- [19] Piryonesi S. M. ve El-Diraby T. E., "Data Analytics in Asset Management: Cost-Effective Prediction of the Pavement Condition Index," *Journal of Infrastructure Systems*, cilt 26, p. 04019036, 2020.
- [20] Cybenko G., "Approximation by superpositions of a sigmoidal function," *Mathematics of Control, Signals, and Systems*, cilt 2, p. 303–314, December 1989.
- [21] Freund Y. ve Schapire R. E., "A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting," *Journal of Computer and System Sciences*, cilt 55, pp. 119-139, 1997.
- [22] Turan T., "Optimize Edilmiş Denetimli Öğrenme Algoritmaları ile Obezite Analizi ve Tahmini," *Mehmet Akif Ersoy Üniversitesi Fen Bilimleri Enstitüsü Dergisi*, cilt 14, p. 301–312, 2023.
- [23] Cuhadar S. N., Karaduman G., Uyanik A. ve Durmaz H., "Performance Analysis Of Machine Learning-Based Models For Early Diagnosis Of Obesity Using Blood Test Parameters," *International Journal of Engineering Science and Application*, cilt 7, p. 117–128, 2023.
- [24] Kabongo J., Luzolo M., CLEM'S ve D. R. Congo, "Five Machine Learning Supervised Algorithms for The Analysis and the Prediction of Obesity".
- [25] Cui T., Chen Y., Wang J., Deng H. ve Huang Y., "Estimation of Obesity Levels Based on Decision Trees," %1 içinde *2021 International Symposium on Artificial Intelligence and its Application on Media (ISAIAM)*, 2021.
- [26] Yagin F. H., Gülü M., Gormez Y., Castañeda-Babarro A., Colak C., Greco G., Fischetti F. ve Cataldi S., "Estimation of Obesity Levels with a Trained Neural Network Approach optimized by the Bayesian Technique," *Applied Sciences*, cilt 13, p. 3875, March 2023.
- [27] Roy M., Das S. ve Protity A. T., *OBESYE: Interpretable Diet Recommender for Obesity Management using Machine Learning and Explainable AI*, 2023.
- [28] Zheng Z. ve Ruggiero K., "Using machine learning to predict obesity in high school students," %1 içinde *2017 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, 2017.
- [29] Singh B. ve Tawfik H., "Machine Learning Approach for the Early Prediction of the Risk of Overweight and Obesity in Young People," %1 içinde *Computational Science – ICCS 2020*, Cham, 2020.
- [30] Montanez C. A. C., Fergus P., Hussain A., Al-Jumeily D., Abdulaimma B., Hind J. ve Radi N., "Machine learning approaches for the prediction of obesity using publicly available genetic profiles," %1 içinde *2017 International Joint Conference on Neural Networks (IJCNN)*, 2017.
- [31] Kivrak M., "Deep Learning-Based Prediction Of Obesity Levels According To Eating Habits And Physical Condition," *The Journal of Cognitive Systems*, cilt 6, p. 24–27, June 2021.
- [32] Daud N., Mohd Noor N. L., Aljunid S. A., Noordin N. ve Fahmi Teng N. I. M., "Predictive Analytics: The Application of J48 Algorithm on Grocery Data to Predict Obesity," %1 içinde *2018 IEEE Conference on Big Data and Analytics (ICBDA)*, 2018.
- [33] Alqahtani A., Albuainin F., Alrayes R. ve Ark., "Obesity level prediction based on data mining techniques," *International journal of computer science and network security: IJCSNS*, cilt 21, p. 103–111, 2021.
- [34] Jindal K., Baliyan N. ve Rana P. S., "Obesity Prediction Using Ensemble Machine Learning Approaches," 2018.
- [35] Kitis S. ve Goker H., "Detection of Obesity Stages Using Machine Learning Algorithms," *Anbar Journal of Engineering Sciences*, cilt 14, p. 80–88, April 2023.
- [36] Sançar B. ve Özcanarslan F., "Akademisyenlerin Obesite Farkındalıklarının Belirlenmesi: Toros Üniversitesi Örneği," *Uluslararası İktisadi ve İdari Bilimler Dergisi*, cilt 7, December 2021.
- [37] Alebna P. L., Mehta A., Yehya A., daSilva-deAbreu A., Lavie C. J. ve Carbone S., "Update on obesity, the obesity paradox, and obesity management in heart failure," *Progress in Cardiovascular Diseases*, cilt 82, pp. 34-42, 2024.

# Makine Öğrenmesi Yöntemleri Kullanılarak Kötü Amaçlı Yazılım Sınıflandırması: CIC-MamMem-2022 Veri Kümesi Üzerinde bir Başarım Karşılaştırması

## Malware Classification Using Machine Learning Methods: A Performance Benchmark on CIC-MamMem-2022 Dataset

Oğuzhan KIRLAR  
Kocaeli Üniversitesi  
Bilgisayar Mühendisliği Bölümü  
Kocaeli, Türkiye  
oguzhankirlar@gmail.com  
ORCID: 0009-0006-2023-1457

Gamze PEKSÖZ AKIN  
Kocaeli Üniversitesi  
Bilgisayar Mühendisliği Bölümü  
Kocaeli, Türkiye  
gamzepeksoz@gmail.com  
ORCID: 0009-0003-5239-1955

Meltem KURT PEHLİVANOĞLU  
Kocaeli Üniversitesi  
Bilgisayar Mühendisliği Bölümü  
Kocaeli, Türkiye  
meltem.kurt@kocaeli.edu.tr  
ORCID:0000-0002-7581-9390

### Öz

Zararlı yazılım veya kötü amaçlı yazılım; bilgisayar ve mobil cihazların işlevlerini bozmak, kritik bilgileri toplamak, özel bilgisayar sistemlerine erişim sağlamak ve istenmeyen reklamları göstermek amacı ile kullanılan yazılımdır. Kötü amaçlı yazılımların güvenlik ve antivirüs sistemlerinde tespit edilebilmesi ya da engellenmesi için makine öğrenmesi tabanlı saldırı tespit/önleme sistemleri kullanılmaktadır. Bu çalışmada CIC-MamMem-2022 veri kümesi üzerinde, makine öğrenmesi yöntemleriyle kötü amaçlı yazılımların sınıflandırılması amaçlanmıştır. Bu veri kümesi üzerinde zorlu bir problem olan on altı sınıf sınıflandırma için literatürde bilinen en iyi F1 ölçüsü, kesinlik, hassasiyet ve doğruluk değerleri sırasıyla %69,46, %70,94, %69,48 ve %69,48 iken; bu çalışmada özellikle on altı sınıf sınıflandırma problemi üzerine odaklanılmış ve literatürde bilinen en iyi sonuçlardan daha iyi sonuçlar elde edilmiştir. Yapılan deneysel çalışmalar sonucunda XGBoost ile F1 ölçüsü, tutturma, bulma ve doğruluk değerleri sırasıyla %75,53, %75,43, %75,65 ve %75,53 olarak elde edilmiştir.

**Anahtar sözcükler:** Zararlı Yazılım Sınıflandırma, Zararlı Yazılım Tespiti, Makine öğrenmesi, Saldırı Tespit Sistemi

### Abstract

Malware or malicious software is software used to disrupt the functioning of computers and mobile devices, collect critical information, gain access to private computer systems, and display unwanted advertisements. Machine learning-based intrusion detection/prevention systems are used to detect or block malware in security and antivirus systems. This study aims to classify malware using machine learning methods on the CIC-MamMem-2022 dataset. For the challenging problem of sixteen-class classification on this dataset, the best-known F1 score, precision, recall, and accuracy values in the literature are 69.46%, 70.94%, 69.48%, and 69.48%, respectively. In this study, a particular focus was placed on the sixteen-class classification problem, and better results than the best-known results in the literature were achieved. As a result of the experimental studies, the F1 score, precision, recall, and accuracy values obtained with XGBoost were 75.53%, 75.43%, 75.65%, and 75.53%, respectively.

**Keywords:** Malware classification, Malware detection, Machine learning, Intrusion Detection System



## 1. Giriş

Saldırı tespit ve saldırı önleme sistemleri, sistemlere yapılan saldırıları tespit etmeyi ve engellemeyi amaçlayan makine öğrenmesi tabanlı sistemlerdir. Özellikle pandemi döneminde uzaktan çalışma, uzaktan eğitim gibi internetin iş, eğitim ve sosyal amaçlarla daha yoğun kullanılmasıyla birlikte sistemlere ve cihazlara yapılan saldırılar da artış göstermiştir. Yapay zekanın ilerlemesi ve dijitalleşmenin her geçen gün her alanda artmasıyla saldırıların da artacağı öngörülmektedir. Bu nedenle zararlı yazılımların yüksek oranda tespit edilmesi ile ilgili çalışmaların yapılması büyük öneme sahiptir.

### 1.1 Motivasyon ve Katkı

Bu çalışmada, zararsız ve kötü amaçlı yazılımları içeren CIC-MalMem-2022 veri kümesi [1] üzerinde, yüksek başarımla kötü amaçlı yazılımları tespit eden makine öğrenmesi tabanlı yeni bir saldırı tespit sistemi geliştirilmesi amaçlanmıştır. Veri kümesi üzerinde farklı ön işleme ve özellik çıkarımı adımları gerçekleştirildikten sonra, Logistic Regression (LR), Random Forest (RF), Support Vector Machine (SVM), Gradient Boosting (GB), Deep Neural Network (DNN), K-Nearest Neighbors (KNN), Decision Tree (DT), XGBoost algoritmaları kullanılarak ikili ve çoklu sınıflandırma problemleri üzerinde çalışılmıştır.

İkili sınıflandırma en başarılı model DNN olup, bu model için doğruluk, F1-skor, kesinlik ve hassasiyet metrikleri sırasıyla; %100, %100, %100, %99,99 elde edilmiştir. Literatürde yer alan ikili sınıflandırma modelleri ile elde edilen sonuçlar değerlendirildiğinde bu değer literatürde yer alan çalışmalar gibi en yüksek başarıma sahiptir. Bu çalışmanın literatüre sağladığı önemli katkı on altı-sınıf sınıflandırma modelleri içindir. Özellikle alt kategori sınıflarına göre yapılan on altı-sınıf sınıflandırmada; en başarılı model XGBoost için doğruluk, F1-skor, kesinlik ve hassasiyet skorları sırasıyla; %75,53, %75,53, %75,43, %75,65 elde edilmiş olup, bu skorlar

literatürde bilinen en iyi skorlardan (%69,46, %70,94, %69,48 ve %69,48) daha başarılıdır.

### 1.2 Organizasyon

Çalışmanın ikinci bölümünde literatürde CIC-MalMem-2022 veri kümesi üzerinde yapılan çalışmalar özetlenmiştir. Üçüncü bölümde ise bu çalışmada önerilen saldırı tespit modelleri detaylandırılmıştır. Dördüncü bölümde gerçekleştirilen deneyler kapsamlı olarak sunulmuştur. Son bölümde ise elde edilen bulgular tartışılarak ileriki çalışmalardan bahsedilmiştir.

## 2. İlgili Çalışmalar

Literatürde, CIC-MamMem-2022 veri kümesi üzerinde, makine öğrenmesi ve/veya derin öğrenme yöntemleriyle kötü amaçlı yazılım tespitini hedefleyen birçok çalışma yer almaktadır.

CIC-MalMem-2022 veri kümesi üzerinde yapılan bu çalışmalar kapsamlı olarak incelenmiş olup; Çizelge 1, Çizelge 2 ve Çizelge 3'de sırasıyla ikili, dört-sınıf, on altı-sınıf sınıflandırma yapan çalışmalarda kullanılan en başarılı model, elde edilen başarımların bilgileri ayrıntılı olarak sunulmuştur. Çizelge 1'de ikili sınıflandırma için verilen sonuçlar değerlendirildiğinde; en başarılı modellerin RobustCBL [3], Decision Tree [6] ve Random Forest [17] olduğu görülmektedir. Çizelge 2'de dört-sınıf sınıflandırma için verilen sonuçlar değerlendirildiğinde; XGBoost [4], modeli ile en yüksek F1-ölçüsü, tutturma, bulma değerleri sırasıyla %88,13, %88,12 ve %88,15 elde edilirken, en yüksek doğruluk değeri ise %89,74 değeriyle Decision Tree [8] algoritmasıyla elde edilmiştir. Çizelge 3'te on altı-sınıf sınıflandırma için verilen sonuçlar değerlendirildiğinde; ExtraTrees [21] modeli ile en yüksek F1-Ölçüsü, tutturma, bulma ve doğruluk değerlerinin sırasıyla %69,46, %70,94, %69,48, %69,48 olarak elde edildiği gözlemlenmiştir.

Çizelge-1: CIC-MalMem-2022 Veri kümesi üzerinde ikili sınıflandırma yapan ilgili çalışmalar

Yıl	Kaynak	Kullanılan En İyi Sınıflandırma Modeli	Başarımların Ölçüleri			
			F-Ölçüsü (F1)	Tutturma	Bulma	Doğruluk
2024	[2]	KNN	%99,97	%99,97	%99,97	%99,97
2023	[3]	RobustCBL (Robust CNN-BiLSTM)	%100,00	%100,00	%100,00	%99,96
2023	[4]	XGBoost	%99,99	%99,99	%99,99	%99,99
2023	[5]	Logistic Regression	%99,99	%99,99	%99,99	%99,99
2023	[6]	Decision Tree	%100,00	%100,00	%100,00	%99,00
2023	[7]	Random Forest	%99,99	%99,99	%99,99	%99,98
2023	[8]	Decision Tree	%99,99	%100,00	%99,98	%99,99
2022	[9]	Logistic Regression	%99,97	%99,98	%99,97	%99,97
2022	[10]	Random Forest	%99,99	%99,99	%99,99	%99,98
2023	[11]	XGBoost	-	-	-	%99,99
2023	[12]	CNN	-	-	-	%99,80
2022	[13]	Logistic Regression	-	-	-	%99,97
2024	[14]	Random Forest	-	-	%99,99	%99,99
2023	[15]	OCC-PCA	%99,00	%99,00	%99,00	%99,40
2023	[16]	Recursive Feature Elimination (RFE)	-	-	-	%99,80
2022	[17]	Random Forest	%100,00	%100,00	%99,99	%100,00
2023	[18]	K-Means	-	-	-	%99,00
2024	[19]	EnsAdp_CIDS algorithm	-	-	-	%99,85
2024	[20]	CNN	%98,72	-	-	%98,82
2023	[21]	Extremely Randomized Trees Classifier (ExtraTrees)	%99,83	%99,78	%99,86	%99,82

2024	[22]	DNN	%99,70	%99,60	%99,99	%99,70
2022	[23]	Decision Tree	-	%99,97	-	%99,98
2023	[24]	SVM	-	-	-	%96,20
2023	[25]	LSTM	-	-	-	%97,69
2024	[26]	VolMemLyzer	%95,45	-	-	%91,25

**Çizelge-2: CIC-MalMem-2022 Veri kümesi üzerinde dört-sınıf sınıflandırma yapan ilgili çalışmalar**

Yıl	Kaynak	Kullanılan En İyi Sınıflandırma Modeli	Başarım Ölçüleri			
			F-Ölçüsü (F1)	Tutturma	Bulma	Doğruluk
2024	[2]	KNN	%82,24	%82,39	%82,21	%82,21
2023	[3]	RobustCBL (Robust CNN-BiLSTM)	%84,00	%85,00	%85,00	%84,56
2023	[4]	XGBoost	<b>%88,13</b>	<b>%88,12</b>	<b>%88,15</b>	%87,21
2023	[7]	Gradient Boosted Tree	-	-	-	%85,12
2023	[8]	Decision Tree	-	-	-	<b>%89,74</b>
2022	[10]	CNN	%75,13	%75,79	%75,18	%83,53
2023	[21]	Extremely Randomized Trees Classifier (ExtraTrees)	%83,02	%83,11	%83,05	%83,05

**Çizelge-3: CIC-MalMem-2022 Veri kümesi üzerinde on altı-sınıf sınıflandırma yapan ilgili çalışmalar**

Yıl	Kaynak	Kullanılan En İyi Sınıflandırma Modeli	Başarım Ölçüleri			
			F-Ölçüsü (F1)	Tutturma	Bulma	Doğruluk
2024	[2]	KNN	%67,32	%68,88	%66,93	%66,93
2023	[21]	Extremely Randomized Trees Classifier (ExtraTrees)	<b>%69,46</b>	<b>%70,94</b>	<b>%69,48</b>	<b>%69,48</b>

### 3. Önerilen Saldırı Tespit Modelleri

#### 3.1 Veri Kümesi

Bu çalışmada kullanılan CIC-MamMem-2022 veri kümesi 29.298 zararsız, 29.298 kötü amaçlı olmak üzere toplam 58.596 farklı gözlem içermektedir. Her bir gözlem için 57 farklı öznitelik bilgisi verilmiştir. Veri kümesi Trojan Horse, Ransomware ve Spyware olmak üzere üç farklı kötü amaçlı yazılım kategorisi içermektedir. Her bir kategori kendi içerisinde farklı kötü amaçlı yazılım ailelerine ayrılmıştır. Trojan Horse kategorisi Zeus, Emotet, Refroso, scar, Reconyc alt yazılım ailelerinden oluşurken; Spyware kategorisi 180Solutions, Coolwebsearch, Gator, Transponder, TIBS ailelerinden oluşur. Ransomware kategorisi ise Conti, MAZE, Pysa, Ako, Shade alt ailelerini içerir. Veri kümesinde yer alan zararsız ve kötü amaçlı yazılım kategorisi ve ilgili ailelere ait gözlem sayıları ile öznitelik bilgileri sırasıyla Çizelge-4 ve Çizelge-5'de sunulmuştur.

**Çizelge-4: CIC-MamMem-2022 veri kümesinde yer alan yazılım sınıf ve alt sınıf gözlem bilgileri**

Yazılım Türü	Yazılım Kategorisi	Yazılım Ailesi	Gözlem Sayıları
Kötü Amaçlı Yazılım	Trojan Horse	Zeus	1950
		Emotet	1967
		Refroso	2000
		scar	2000
		Reconyc	1570
	Spyware	180Solutions	2000
		Coolwebseah	2000
		Gator	2200
		Transponder	2410
		TIBS	1410

Ransomware	Conti	1988
	MAZE	1958
	Pysa	1717
	Ako	2000
	Shade	2128
Zararsız Yazılım	-	29298

**Çizelge-5: CIC-MamMem-2022 Veri kümesinde yer alan öznitelikler**

No	Öznitelik	No	Öznitelik
1	Category	30	malfind.protection
2	pslist.nproc	31	malfind.uniqueInjectio ns
3	pslist.nppid	32	psxview.not_in_pslis t
4	pslist.avg_threads	33	psxview.not_in_eproce ss_pool
5	pslist.nprocs64bit	34	psxview.not_in_ethrea d_pool
6	pslist.avg_handlers	35	psxview.not_in_pspcid _list
7	dlllist.ndlls	36	psxview.not_in_csrs s_handles
8	dlllist.avg_dlls_per_pro c	37	psxview.not_in_sessio n
9	handles.nhandles	38	psxview.not_in_deskth rd
10	handles.avg_handles_p er_proc	39	psxview.not_in_pslis t_false_avg
11	handles.nport	40	psxview.not_in_eproce ss_pool_false_avg
12	handles.nfile	41	psxview.not_in_ethrea d_pool_false_avg
13	handles.nevent	42	psxview.not_in_pspcid _list_false_avg

14	handles.ndesktop	43	psxview.not_in_csrss_handles_false_avg
15	handles.nkey	44	psxview.not_in_session_false_avg
16	handles.nthread	45	psxview.not_in_deskthr_false_avg
17	handles.ndirectory	46	modules.nmodules
18	handles.nsemaphore	47	svcsan.nservices
19	handles.ntimer	48	svcsan.kernel_drivers
20	handles.nsection	49	svcsan.fs_drivers
21	handles.nmutant	50	svcsan.process_services
22	ldrmodules.not_in_load	51	svcsan.shared_process_services
23	ldrmodules.not_in_init	52	svcsan.interactive_process_services
24	ldrmodules.not_in_memory	53	svcsan.nactive
25	ldrmodules.not_in_load_avg	54	callbacks.ncallbacks
26	ldrmodules.not_in_init_avg	55	callbacks.nanonymous
27	ldrmodules.not_in_memory_avg	56	callbacks.ngeneric
28	malfind.ninjections	57	Class
29	malfind.commitCharge		

### 3.2 Veri Önışleme

Çalışmada kullanılan CIC-MamMem-2022 veri kümesi toplamda 58.596 farklı gözlem ve 57 farklı öznelikten oluşmaktadır. Veri kümesi üzerinde farklı ön işleme yöntemleri ile veri kümesini temsil eden öznelıklar 33'e, gözlem sayısı ise 58.062'ye indirilmiştir. Aşğıdaki alt başlıklarda önışleme adımları kapsamlı olarak verilmiştir.

#### 3.2.1 Öznelik Seçimi

Veri kümesi üzerinde öncelikle her bir öznelik için gözlem değerleri aynı olan öznelikler kontrol edilmiştir. "pslist.nprocs64bit", "handles.nport", "svcsan.interactive\_process\_services" özneliklerinde yer alan gözlem değerlerinin tamamının aynı olduğu tespit edilmiştir. Bu üç farklı özneliğın, model üzerinde bir etkisi olmayacağı düşünülerek veri kümesinden çıkarılmıştır.

Ayrıca veri kümesi üzerinde eksik değer kontrolü yapılmış, ancak herhangi bir eksik değer gözlemlenmemiştir.

Son olarak öznelik değerleri birbiri ile tamamen aynı olan gözlem sayısı kontrolü yapılmış ve 534 gözlemin birbiriyle aynı olduğu tespit edilerek, bu gözlemler veri kümesinden çıkarılmıştır.

İkili sınıflandırma yapabilmek için "Class" özneliğine "One hot encoding" işlemi uygulanmıştır. Bunun sonucunda da verinin zararlı ya da zararsız olma durumu boolean değerine indirgenmiştir.

Veri kümesi içerisinde anlam olarak birbirine benzeyen öznelikler bulunduğu için bazılarının toplamı ya da ortalamasını almanın modellere daha fazla katkısı olduğu fark edilmiştir. Aynı ayrı işlem sayısını almak yerine toplam işlem sayısını belirlemek için "pslist.nproc" ve

"pslist.nprocs64bit" öznelikleri toplanıp "total\_procs" sayısı belirlendi. Ancak "pslist.nprocs64bit" değeri tüm verilerde aynı olduğu ve veri kümesinden kaldırdığı için "pslist.nproc" özneliğinin ismi "total\_procs" olarak değerlendirilmiştir.

Ortalama iş parçacığı ve işleyici sayısının toplamını belirlemek için "pslist.avg\_threads" ve "pslist.avg\_handlers" öznelikleri toplanarak "avg\_thread\_handlers" özneliği oluşturulmuştur.

Toplam handle sayısını belirlemek için "handles.nport", "handles.nfile", "handles.nevent", "handles.ndesktop", "handles.nkey", "handles.nthread", "handles.ndirectory", "handles.nsemaphore", "handles.ntimer", "handles.nsection", "handles.nmutant" öznelikleri toplanarak ve "total\_handles" özneliği oluşturulmuştur. Ancak "handles.nport" özneliği için veri kümesindeki tüm değerler aynı olduğundan bu öznelik kaldırıldığından ve bu toplama bu öznelik dahil edilmemiştir.

Ayrıca ortalama yüklenmeyen modül hesaplaması yapılarak, "ldrmodules.not\_in\_load", "ldrmodules.not\_in\_init", "ldrmodules.not\_in\_memory" özneliklerinin ortalaması alınıp "avg\_not\_loaded\_modules" özneliği oluşturulmuştur.

Son olarak, toplam görünmeyen işlem sayısı hesaplanarak; "psxview.not\_in\_pslist", "psxview.not\_in\_eprocess\_pool", "psxview.not\_in\_ethread\_pool", "psxview.not\_in\_pspcid\_list", "psxview.not\_in\_csrss\_handles", "psxview.not\_in\_session", "psxview.not\_in\_deskthrd" öznelikleri toplanarak "total\_hidden\_processes" özneliği oluşturulmuştur.

Yukarıda verilen hesaplamalarda kullanılan öznelikler veri kümesinden çıkarılarak, yeni oluşturulan öznelikler veri kümesine eklenmiştir. Ayrıca çoklu sınıflandırmada kullanılmak üzere "Category" özneliğinden "category\_name" ve "subcategory\_name" isimli iki yeni öznelik oluşturulmuştur. Bu iki öznelik dört sınıf ve onaltı sınıf sınıflandırmada kullanılmıştır, sınıf özneliğini temsil ettiğinden model eğitim özneliklerine dahil edilmemiştir.

Yapılan işlemler sonucunda veri kümesinin son hali Çizelge 6'da sunulmuştur.

**Çizelge-6: Önışleme ve öznelik mühendisliğı uygulandıktan sonra veri kümesinde yer alan öznelikler**

No	Öznelik	No	Öznelik
1	pslist.nppid	18	psxview.not_in_session_false_avg
2	dlllist.ndlls	19	psxview.not_in_deskthrd_false_avg
3	dlllist.avg_dlls_per_process	20	modules.nmodules
4	handles.nhandles	21	svcsan.nservices
5	handles.avg_handles_per_proc	22	svcsan.kernel_drivers
6	ldrmodules.not_in_load_avg	23	svcsan.fs_drivers
7	ldrmodules.not_in_init_avg	24	svcsan.process_services
8	ldrmodules.not_in_memory_avg	25	svcsan.shared_process_services
9	malfind.ninjections	26	svcsan.nactive
10	malfind.commitCharge	27	callbacks.ncallbacks
11	malfind.protection	28	callbacks.nanonymous

12	malfind.uniqueInjection_s	29	callbacks.ngeneric
13	psxview.not_in_pslis_false_avg	30	avg_threads_handlers
14	psxview.not_in_eproce_ss_pool_false_avg	31	total_handles
15	psxview.not_in_ethrea_d_pool_false_avg	32	average_not_loaded_modules
16	psxview.not_in_pspcid_list_false_avg	33	total_hidden_processes
17	psxview.not_in_csrss_handles_false_avg		

### 3.2.2. Özellik/Öznitelik Öneminin Uygulanması

Dengesiz ve dengeli veri kümeleri üzerinde en ayırt edici özniteliklerin belirlenmesi için öznitelik önemi kullanılmıştır. En önemli öznitelikler çıkarılırken XGBoost modeli üzerinde 'feature\_importance\_' yöntemi kullanılmış olup, önem grafiği Şekil 1'de sunulmuştur. Özniteliklerin önem oran grafiğine göre ilk 20 öznitelik çoklu sınıflandırma (dörtlü ve on altılı sınıflandırma) için modellerde kullanılmıştır. Dengesiz veri kümesi üzerinde kullanılan 20 öznitelik Çizelge 7'de sunulmuştur.

Dengeli veri kümesi için elde edilen önem grafiği ve özniteliklerin isimleri sırasıyla Şekil 2 ve Çizelge 8'de verilmiştir.

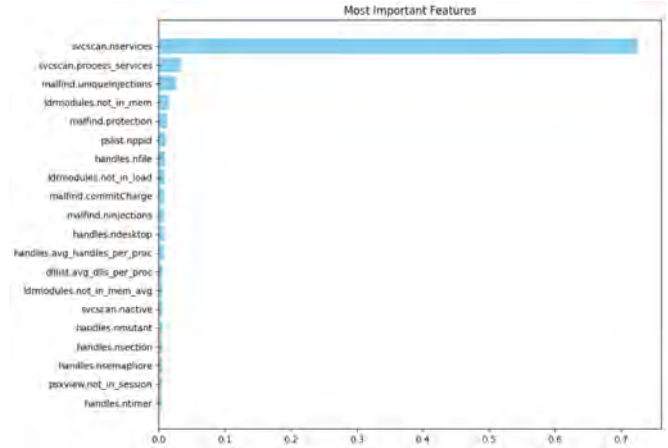


Şekil-1: Dengesiz veri kümesi üzerinde en önemli özniteliklerin belirlenmesi

Çizelge-7: Dengesiz veri kümesi üzerinde model eğitiminde kullanılan en önemli 20 öznitelik

No	Öznitelik	No	Öznitelik
1	svcscan.nservices	11	malfind.commitCharge
2	svcscan.shared_process_services	12	callbacks.ngeneric
3	svcscan.process_service_s	13	svcscan.nactive
4	malfind.uniqueInjection_s	14	psxview.not_in_csrss_handles_false_avg
5	dlllist.avg_dlls_per_proc	15	ldrmodules.not_in_load_avg
6	handles.avg_handles_per_proc	16	dlllist.ndlls
7	ldrmodules.not_in_mem_avg	17	pslist.nppid
8	average_not_loaded_modules	18	callbacks.nanonymous
9	malfind.ninjections	19	psxview.not_in_session_false_avg

10	malfind.protection	20	total_handles
----	--------------------	----	---------------



Şekil-2: Dengeli veri kümesi için en önemli özniteliklerin belirlenmesi

Çizelge-8: Dengeli veri kümesi üzerinde model eğitiminde kullanılan en önemli 20 öznitelik

No	Öznitelik	No	Öznitelik
1	svcscan.nservices	11	handles.ndesktop
2	svcscan.process_service_s	12	handles.avg_handles_per_proc
3	malfind.uniqueInjection_s	13	dlllist.avg_dlls_per_proc
4	ldrmodules.not_in_mem	14	ldrmodules.not_in_mem_avg
5	malfind.protection	15	svcscan.nactive
6	pslist.nppid	16	handles.nmutant
7	handles.nfile	17	handles.nsection
8	ldrmodules.not_in_load	18	handles.nsemaphore
9	malfind.commitCharge	19	psxview.not_in_session
10	malfind.ninjections	20	handles.ntimer

### 3.2.3. Hiper Parametre Ayarlaması

Daha etkili ve başarılı sonuçlar elde etmek için hiper parametre ayarlaması yapılmıştır. Bu işlemde GridSearchCV fonksiyonundan yararlanılmıştır. Bu fonksiyon, kendisine verilen hiper parametre değerlerinin tüm olasılıklarını deneyerek, çapraz doğrulama ile en yüksek başarıyı sağlayan parametre kombinasyonlarını belirler.

## 4. Deneysel Sonuçlar

Çalışma kapsamında veri kümesi üzerinde ikili sınıflandırma, dört-sınıf sınıflandırma ve on altı-sınıf sınıflandırma için modeller eğitilmiş ve test edilmiştir. Tüm sınıflandırma problemlerinde ön işleme adımlarından ve öznitelik seçimi sonucundan oluşan veriler kullanılmıştır. Modeller üzerinde hiper parametre ayarlaması yapılarak başarı oranları artırılmıştır.

Sınıflandırma başarımları değerlendirilirken; 58,062 gözlemin %60'ı eğitim, %20'si test verisi, %20'si ise validasyon verisi olarak ayrılmıştır.

Her bir sınıflandırma için gerçekleştirilen modellerin performansını değerlendirmek ve genelleme yeteneğini ölçmek amaçlı 5-katmanlı (fold) çapraz doğrulama yöntemi kullanılmıştır.

Bunun yanı sıra özellikle dört-sınıf ve on altı-sınıf sınıflandırma probleminde oluşacak dengesiz veri problemini çözmek amaçlı "zararsız yazılım" sınıfından (Bknz. Çizelge 4) diğer kötü amaçlı yazılımların ortalaması olacak şekilde rastgele gözlem seçilerek dengeli veri kümeleri oluşturulmuştur. Bu sayede dengesiz veri problemi çözülmüştür. Dengeli veri kümesi kullanarak kurulan modellerde XGBoost modeli üzerinde 'feature\_importance\_' uygulanmış önem sırasına göre ilk 20 öznitelik modellerde kullanılmıştır.

Dört-sınıf sınıflandırmada için oluşturulan dengeli veri kümesi; Çizelge 4'de ayrıntılı verilen 3 sınıfa ait 28.831 adet kötü amaçlı yazılım ile 9610 adet zararsız yazılım sınıfı olmak üzere, toplam 38.441 gözlem içerir.

Onaltı-sınıf sınıflandırma için oluşturulan dengeli veri kümesi; Çizelge 4'de ayrıntılı verilen 15 sınıfa ait 28.831 adet kötü amaçlı yazılım ile 1.949 adet zararsız yazılım sınıfı olmak üzere, toplam 30.780 gözlem içerir.

İki-sınıf sınıflandırmada dengesiz veri problemi oluşmadığından bu deneyler için gözlem sayısında

herhangi bir değişiklik yapılmamış olup Çizelge 4'de verilen sayıda gözlem kullanılmıştır.

Yukarıda verilen bilgiler doğrultusunda aşağıdaki alt başlıklarda ikili, dördü ve onaltılı sınıflandırma modelleri için sınıflandırma modelleri kurulmuş ve her bir sınıflandırma problemi için elde edilen sonuçlar detaylandırılmıştır.

#### 4.1. İkili Sınıflandırma

İkili sınıflandırma için "one hot encoding" ile oluşturulan öznitelik ("Class" özniteliği kullanılarak boolean değerinde zararlı ve zararsız özelliğine temsilen) sınıflandırma için kullanılmıştır. Veri kümesinde yer alan gözlemler zararlı (toplam 28.831 gözlem) ve zararsız (toplam 29.298 gözlem) olmak üzere iki sınıfa ayrılmıştır. İkili sınıflandırma için LR, RF, SVM, GB, DNN algoritmaları kullanılarak sınıflandırma modelleri oluşturulmuştur. Yapılan deneysel sonuçlar Çizelge 9'da sunulmuş olup en başarılı model DNN olmuştur. Literatürde yer alan ikili sınıflandırma ile ilgili çalışmalar (Bknz. Çizelge 1) ile elde edilen sonuçlar değerlendirildiğinde en başarılı metriklerin elde edildiği açıktır.

Çizelge-9: İkili Sınıflandırma İçin Modellerin Başarımı

Algoritma	Hiperparametre	Kullanılan Öznitelik Sayısı	Başarım Ölçüleri			
			F-Ölçüsü (F1)	Tutturma	Bulma	Doğruluk
LR	max_iter=1000	33	%99,62	%99,69	%99,69	%99,69
RF	n_estimators=100, random_state=42	33	%99,99	%99,98	<b>%100,00</b>	%99,99
SVM	kernel='linear', random_state=42	33	%99,82	%99,93	%99,72	%99,82
GB	n_estimators=100, learning_rate=1.0, max_depth=1, random_state=42	33	%99,97	%99,98	%99,96	%99,97
DNN	epochs=20, batch_size=32	33	<b>%100,00</b>	<b>%100,00</b>	%99,99	<b>%100,00</b>

#### 4.2. Dört Sınıf Sınıflandırma

Zararsız (Benign), Trojan Horse, Ransomware ve Spyware sınıflandırılması için öznitelik önem sırası dikkate alınarak, en önemli 20 öznitelik kullanılmıştır. Yapılan deneyler sonucunda dört-sınıf sınıflandırma için DT, XGBoost, RF, KNN modelleri 5-katmanlı çapraz doğrulama kullanılarak elde edilen başarımler Çizelge 10'da verilmiştir. Çizelgeden de görüleceği gibi dört-sınıf sınıflandırma için en başarılı model XGBoost olup, literatürde dört-sınıf sınıflandırma içeren çalışmalar [4,8] ile karşılaştırıldığında, başarı oranı daha düşük kalmıştır.

Literatürde dört-sınıf sınıflandırma problemi üzerine çalışan çalışmalarda [2-4, 7, 8, 21] veri dengesizliği problemi ele alınmamıştır. Dengesiz veri problemini çözmek amaçlı oluşturulan dengeli veri kümesi üzerinde elde edilen dört-sınıf sınıflandırma sonuçları ise Çizelge 11'de verilmiştir. Çizelgeden de görüleceği gibi dengeli veri

kümesi üzerinde modellerin sınıflandırma başarımları düşmüştür, ancak modeller arasında en başarılı algoritma XGBoost'tur.

**Çizelge-10: Dört Sınıf Sınıflandırma İçin Modellerin Başarımı**

Algoritma	Hiperparametre	Kullanılan Öznitelik Sayısı	Başarım Ölçüleri			
			F-Ölçüsü (F1)	Tutturma	Bulma	Doğruluk
DT	'criterion': ['gini', 'log_loss'], 'splitter': ['best', 'random'], 'min_samples_leaf': [100, 200, 300], 'max_depth': [3, 5, 8]	20	%77,83	%77,83	%77,83	%77,83
XGBoost	'objective': ['binary:logistic', 'multi:softmax'], 'n_estimators': [100, 200, 300], 'max_depth': [3, 5, 8], 'tree_method': ['hist']	20	<b>%85,64</b>	<b>%85,87</b>	<b>%85,42</b>	<b>%85,84</b>
RF	'n_estimators': [100, 200, 300], 'max_depth': [3, 5, 8, 10], 'min_samples_split': [2, 5, 10], 'min_samples_leaf': [1, 2, 4], 'bootstrap': [True, False]	20	%81,60	%81,58	%81,62	%81,75
KNN	'n_neighbors': [3, 5, 7, 9, 11], 'weights': ['uniform', 'distance'], 'metric': ['euclidean', 'manhattan']	20	%80,72	%80,85	%80,60	%80,72

**Çizelge-11: Dengeli Veri Kümesi Üzerinde Dört Sınıf Sınıflandırma İçin Modellerin Başarımı**

Algoritma	Hiperparametre	Kullanılan Öznitelik Sayısı	Başarım Ölçüleri			
			F-Ölçüsü (F1)	Tutturma	Bulma	Doğruluk
DT	'criterion': ['gini', 'log_loss'], 'splitter': ['best', 'random'], 'min_samples_leaf': [100, 200, 300], 'max_depth': [3, 5, 8]	20	%72,61	%71,46	%73,81	%71,46
XGBoost	'objective': ['binary:logistic', 'multi:softmax'], 'n_estimators': [100, 200, 300], 'max_depth': [3, 5, 8], 'tree_method': ['hist']	20	<b>%78,00</b>	<b>%77,98</b>	<b>%78,04</b>	<b>%77,98</b>
RF	'n_estimators': [100, 200, 300], 'max_depth': [3, 5, 8, 10], 'min_samples_split': [2, 5, 10], 'min_samples_leaf': [1, 2, 4], 'bootstrap': [True, False]	20	%74,73	%74,66	%74,81	%74,91
KNN	'n_neighbors': [3, 5, 7, 9, 11], 'weights': ['uniform', 'distance'], 'metric': ['euclidean', 'manhattan']	20	%73,50	%73,64	%73,38	%73,22

### 4.3. On altı Sınıf Sınıflandırma

Çalışmada ayrıca veri kümesinde zararlı yazılımlara ait alt kategoriler kullanılarak on altı-sınıf sınıflandırma yapılmıştır. Bu sınıflandırma için dört-sınıf sınıflandırmadaki gibi en önemli 20 öznitelik kullanılarak DT, XGBoost, RF, KNN modelleri test edilmiştir. Yapılan deneyler sonucunda onaltı-sınıf sınıflandırma ile elde edilen en iyi F1-skor, kesinlik, hassasiyet ve doğruluk skorları (sırasıyla %75,53, %75,43, %75,65, %75,53) Çizelge 12'de verilmiştir. Elde edilen deneysel sonuçlar değerlendirildiğinde en başarılı algoritma XGBoost olurken, elde edilen başarımların değeri literatürdeki en iyi sonuçlardır (Bknz. Çizelge 3).

Literatürde on altı-sınıf sınıflandırma problemi üzerine çalışan çalışmalarda [2,21] veri dengesizliği problemi ele alınmamıştır. Bu çalışmada dengesiz veri problemini çözmek amaçlı oluşturulan dengeli veri kümesi üzerinde elde edilen on altı-sınıf sınıflandırma sonuçları ise Çizelge 13'de verilmiştir. Çizelgede verilen sonuçlar değerlendirildiğinde modellerin başarımlarının önemli derecede düştüğü gözlemlenmiştir. Alt sınıflar birbirleri arasında benzerlik gösterdiğinden modellerin sınıfları ayırması daha da zorlaşmıştır.

**Çizelge-12: On altı Sınıf Sınıflandırma için Modellerin Başarımı**

Algoritma	Hiperparametre	Kullanılan Öznitelik Sayısı	Başarım Ölçüleri			
			F-Ölçüsü (F1)	Tutturma	Bulma	Doğruluk
DT	'criterion': ['gini', 'log_loss'], 'splitter': ['best', 'random'], 'min_samples_leaf': [100, 200, 300], 'max_depth': [3, 5, 8]	20	%62,28	%62,28	%62,28	%62,28
XGBoost	'objective': ['binary:logistic', 'multi:softmax'], 'n_estimators': [100, 200, 300], 'max_depth': [3, 5, 8], 'tree_method': ['hist']	20	<b>%75,53</b>	<b>%75,43</b>	<b>%75,65</b>	<b>%75,53</b>
RF	'n_estimators': [100, 200, 300], 'max_depth': [3, 5, 8, 10], 'min_samples_split': [2, 5, 10], 'min_samples_leaf': [1, 2, 4], 'bootstrap': [True, False]	20	%70,72	%72,12	%69,38	%69,42
KNN	'n_neighbors': [3, 5, 7, 9, 11], 'weights': ['uniform', 'distance'], 'metric': ['euclidean', 'manhattan']	20	%67,49	%67,44	%67,70	%67,70

**Çizelge-13: Dengeli Veri Kümesi Üzerinde On altı Sınıf Sınıflandırma için Modellerin Başarımı**

Algoritma	Hiperparametre	Kullanılan Öznitelik Sayısı	Başarım Ölçüleri			
			F-Ölçüsü (F1)	Tutturma	Bulma	Doğruluk
DT	'criterion': ['gini', 'log_loss'], 'splitter': ['best', 'random'], 'min_samples_leaf': [100, 200, 300], 'max_depth': [3, 5, 8]	20	%55,64	%55,64	%55,64	%55,64
XGBoost	'objective': ['binary:logistic', 'multi:softmax'], 'n_estimators': [100, 200, 300], 'max_depth': [3, 5, 8], 'tree_method': ['hist']	20	<b>%67,60</b>	<b>%67,67</b>	<b>%67,52</b>	<b>%67,75</b>
RF	'n_estimators': [100, 200, 300], 'max_depth': [3, 5, 8, 10], 'min_samples_split': [2, 5, 10], 'min_samples_leaf': [1, 2, 4], 'bootstrap': [True, False]	20	%64,73	%64,62	%64,84	%64,80
KNN	'n_neighbors': [3, 5, 7, 9, 11], 'weights': ['uniform', 'distance'], 'metric': ['euclidean', 'manhattan']	20	%60,38	%60,26	%60,50	%60,48

Bu çalışma kapsamında yapılan deneysel sonuçlar literatürdeki çalışmalar ile ikili ve çoklu sınıflandırma başarımları açısından karşılaştırıldığında, diğer çalışmalara kıyasla önemli başarılar elde edildiği açıktır. Özellikle ikili sınıflandırma için literatürdeki en yüksek başarımlar elde edilirken, on altı-sınıf sınıflandırma ise; literatürde en iyi sonuçları elde eden çalışmaya [21] oranla çok daha yüksek başarımlar sağlanmıştır. [21]'de verilen çalışmada F1-skor, kesinlik, hassasiyet ve doğruluk skorları sırasıyla %69,46, %70,94, %69,48, %69,48 elde edilirken, bu çalışmada aynı skorlar XGBoost algoritmasıyla sırasıyla %75,53, %75,43, %75,65, %75,53 olarak elde edilmiştir.

Bu çalışmada ayrıca literatürdeki diğer çalışmalardan farklı olarak dengesiz veri problemini çözmek için 'zararsız yazılım' örneklerinin sayısı diğer sınıfların ortalaması olacak şekilde azaltılmıştır. Bu da problemi daha zorlu hale

getirmiştir. Çalışmada bu modellerin başarımlarının artırılması ileriki çalışma olarak bırakılmıştır.

Ayrıca bu çalışmada yapılan tüm deneysel çalışmaları içeren kaynak kodları, GitHub üzerinde [27] erişime açık olarak paylaşılmıştır.

## 5. Sonuç ve İleriki Çalışmalar

Bu çalışmada CIC-MalMeM-2022 veri kümesi kullanılarak zararlı ve zararsız olarak ikili sınıflandırma, zararlı yazılımların kategorileri kullanılarak dört-sınıf sınıflandırma ve alt kategorileri kullanılarak on altı-sınıf sınıflandırma probleminin çözülmesi üzerine çalışılmıştır. Çalışma kapsamında LR, RF, SVM, DT, GB, XGBoost, DNN ve KNN modelleri kullanılarak modeller oluşturulmuştur.

Deneysel sonuçlar incelendiğinde en önemli özniteliklerin belirlenmesi ve hiper parametre ayarlaması yapılarak başarımların her modelde arttığı gözlemlenmiştir.

Literatürde yer alan diğer çalışmalar incelendiğinde; veri CIC-MalMeM-2022 kümesi özelinde, özellikle on altı-sınıf sınıflandırma için en yüksek başarımlar elde edilmiştir.

İleriki çalışmalarda özellikle dört-sınıf ve on altı-sınıf sınıflandırma için dengeli veri kümesi üzerinde modellerin başarımının artırılması amaçlı farklı veri artırma tekniklerinin kullanılması, farklı modellerin denenmesi gibi çalışmalar hedeflenmektedir.

## Kaynakça

- [1] Carrier, T., Victor, P., Tekeoglu, A., & Lashkari, A. H. (2022, February). Detecting Obfuscated Malware using Memory Feature Engineering. In *Icissp* (pp. 177-188).
- [2] Abualhaj, M., Abu-Shareha, A., Shambour, Q., Alsaaidah, A., Al-Khatib, S., & Anbar, M. (2024). Customized K-nearest neighbors' algorithm for malware detection. *International Journal of Data and Network Science*, 8(1), 431-438.
- [3] Shafin, S. S., Karmakar, G., & Mareels, I. (2023). Obfuscated memory malware detection in resource-constrained IoT devices for smart city applications. *Sensors*, 23(11), 5348.
- [4] Hasan, S. R., & Dhakal, A. (2023, December). Obfuscated Malware Detection: Investigating Real-World Scenarios Through Memory Analysis. In *2023 IEEE International Conference on Telecommunications and Photonics (ICTP)* (pp. 01-05). IEEE.
- [5] Jiang, Q., Zhao, X., & Huang, K. (2011, June). A feature selection method for malware detection. In *2011 IEEE International Conference on Information and Automation* (pp. 890-895). IEEE.
- [6] Smith, D., Khorsandroo, S., & Roy, K. (2023, February). Supervised and unsupervised learning techniques utilizing malware datasets. In *2023 IEEE 2nd International Conference on AI in Cybersecurity (ICAIC)* (pp. 1-7). IEEE.
- [7] Benkerroum, S., & Chougali, K. (2023, December). Enhancing Forensic Analysis Using a Machine Learning-based Approach. In *2023 6th International Conference on Advanced Communication Technologies and Networking (CommNet)* (pp. 1-6). IEEE.
- [8] Balasubramanian, K. M., Vasudevan, S. V., Thangavel, S. K., Kumar, G., Srinivasan, K., Tibrewal, A., & Vajipayajula, S. (2023, July). Obfuscated Malware detection using Machine Learning models. In *2023 14th International Conference on Computing Communication and Networking Technologies (ICCCNT)* (pp. 1-8). IEEE.
- [9] Dener, M., Ok, G., & Orman, A. (2022). Malware detection using memory analysis data in big data environment. *Applied Sciences*, 12(17), 8604.
- [10] Mezina, A., & Burget, R. (2022, October). Obfuscated malware detection using dilated convolutional network. In *2022 14th international congress on ultra modern telecommunications and control systems and workshops (ICUMT)* (pp. 110-115). IEEE
- [11] Talukder, M. A., Hasan, K. F., Islam, M. M., Uddin, M. A., Akhter, A., Yousuf, M. A., ... & Moni, M. A. (2023). A dependable hybrid machine learning model for network intrusion detection. *Journal of Information Security and Applications*, 72, 103405
- [12] Naeem, H., Dong, S., Falana, O. J., & Ullah, F. (2023). Development of a deep stacked ensemble with process based volatile memory forensics for platform independent malware detection and classification. *Expert Systems with Applications*, 223, 119952.
- [13] Dener, M., Ok, G., & Orman, A. (2022). Malware detection using memory analysis data in big data environment. *Applied Sciences*, 12(17), 8604.
- [14] Smmarwar, S. K., Gupta, G. P., & Kumar, S. (2024). Android Malware Detection and Identification Frameworks by Leveraging the Machine and Deep Learning Techniques: A Comprehensive Review. *Telematics and Informatics Reports*, 100130.
- [15] Al-Qudah, M., Ashi, Z., Alnabhan, M., & Abu Al-Haija, Q. (2023). Effective one-class classifier model for memory dump malware detection. *Journal of Sensor and Actuator Networks*, 12(1), 5.
- [16] Alani, M. M., Mashatan, A., & Miri, A. (2023). XMal: A lightweight memory-based explainable obfuscated-malware detector. *Computers & Security*, 133, 103409.
- [17] Louk, M. H. L., & Tama, B. A. (2022). Tree-based classifier ensembles for PE malware analysis: A performance revisit. *Algorithms*, 15(9), 332.
- [18] Smith, D., Khorsandroo, S., & Roy, K. (2023, February). Supervised and unsupervised learning techniques utilizing malware datasets. In *2023 IEEE 2nd International Conference on AI in Cybersecurity (ICAIC)* (pp. 1-7). IEEE.
- [19] Roshan, K., & Zafar, A. (2024). Ensemble adaptive online machine learning in data stream: a case study in cyber intrusion detection system. *International Journal of Information Technology*, 1-14.
- [20] Maniriho, P., Mahmood, A. N., & Chowdhury, M. J. M. (2024). MeMalDet: A memory analysis-based malware detection framework using deep autoencoders and stacked ensemble under temporal evaluations. *Computers & Security*, 142, 103864.
- [21] Roy, K. S., Ahmed, T., Udas, P. B., Karim, M. E., & Majumdar, S. (2023). Malhystack: A hybrid stacked ensemble learning framework with feature engineering schemes for obfuscated malware analysis. *Intelligent Systems with Applications*, 20, 200283.
- [22] Cevallos-Salas, D., Grijalva, F., Estrada-Jiménez, J., Benítez, D., & Andrade, R. (2024). Obfuscated Privacy Malware Classifiers based on Memory Dumping Analysis. *IEEE Access*.
- [23] Nugraha, A., & Zeniarja, J. (2022). Malware Detection Using Decision Tree Algorithm Based on Memory Features Engineering. *Journal of Applied Intelligent System*, 7(3), 206-210b
- [24] Noor, B., & Qadir, S. (2023). Machine Learning and Deep Learning Based Model for the Detection of Rootkits Using Memory Analysis. *Applied Sciences*, 13(19), 10730.
- [25] Özkam, Y. (2023). Malware Detection in Forensic Memory Dumps: The Use of Deep Meta-Learning Models. *Acta Infologica*, 7(1), 165-172
- [26] Yogesh, K. M., Arpitha, S., Stephan, T., Praksha, M., & Raghu, V. (2023, December). Unravelling Obfuscated Malware Through Memory Feature Engineering and Ensemble Learning. In *International Conference on Information and Communication Technology for Competitive Strategies* (pp. 323-332). Singapore: Springer Nature Singapore.
- [27] MalMem-Classification, <https://github.com/oguzhankirlar/MalMem-Classification>, Erişim Tarihi: 24.06.2024.





# Hedef Gen Çıkarımı için Makine Öğrenmesi Tabanlı Mikrodizi Verilerinin Analizi

## Machine Learning Based Microarray Data Analysis For Target Gene Inference

Zerrin YILDIZ ÇAVDAR  
İstanbul Üniversitesi Cerrahpaşa  
Bilgisayar Mühendisliği Bölümü  
İstanbul, Türkiye  
zerrintt@gmail.com  
ORCID: 0000-0003-4299-5344

Ahmet SERTBAŞ  
İstanbul Üniversitesi Cerrahpaşa  
Bilgisayar Mühendisliği Bölümü  
İstanbul, Türkiye  
asertbas@iuc.edu.tr  
ORCID: 0000-0001-8166-1211

Tolga ENSARİ  
Arkansas Tech University  
Department of Engineering and  
Computing Sciences  
Russellville, USA  
tensari@atu.edu  
ORCID: 0000-0003-0896-3058

### Öz

Mikrodizi teknolojisi gen ifadesindeki farklılıkların tespit edilmesinde kullanılır. Bu teknoloji ilaç geliştirme süreçlerinden tedavi süreçlerinin iyileştirilmesine birçok alanda katkı sağlamaktadır. Bu çalışmada, kronik hipoksi tedavisinin fare beyni üzerindeki etkisi ve oksidatif strese maruz kalan fare nöronlarının gen üzerindeki etkisi ile ilgili veri kümeleri üzerinde mikrodizi analizi yapılmıştır. Çalışmada açık erişim sağlanabilen iki farklı mikrodizi veri kümesi kullanılmıştır ve makine öğrenmesi yöntemleri ile çalışılmıştır. İlk adım olarak veri kümelerinin indirilmesi, ardından ön işleme tabi tutularak normalizasyon uygulanması sağlanmıştır. Bu aşamalar ile veri kümeleri gen çıkarımı için çalışmaya uygun hale getirilmiştir. Hazırlanan bu gen ekspresyon ifadeleri üzerinden istatistiksel ve makine öğrenmesi yöntemleri kullanılarak analizler gerçekleştirilmiş ve hedef gen çıkarımı sağlanmıştır.

**Anahtar sözcükler:** Mikrodizi, makine öğrenmesi, hedef gen çıkarımı, gen ekspresyon analizi

### Abstract

Microarray technology is used to detect differences in gene expression. This technology contributes to many fields, from drug development processes to the improvement of treatment procedures. In this study, microarray analysis was performed on datasets related to the effect of chronic hypoxia treatment on the mouse brain and the effect of oxidative stress on the genes of mouse neurons. Two

different publicly accessible microarray datasets were used in the study, and machine learning methods were employed. The first step involved downloading the datasets, followed by preprocessing and normalization. These steps made the datasets suitable for gene extraction. Statistical and machine learning methods were applied to the prepared gene expression profiles, leading to the inference of target genes.

**Keywords:** Microarray, machine learning, target gene inference, analyzing gene expression

### 1. Giriş

Mikrodizin, biyoçip, gençip olarak da adlandırılan mikrodizi teknolojisi, gen ekspresyonunun incelenmesinde yaygın olarak kullanılan bir teknolojidir. Mikrodizi veri analizi, kanser araştırmaları, tedavi süreçlerinin iyileştirilmesi veya ilaç geliştirme gibi birçok alanda önemli bir araç olmuştur. Bu yayın çalışmasında mikrodizi teknolojisi yardımıyla, hastalık tedavi süreçlerinin etkili olduğu gen ifadesi farklılıkları incelenmiştir.

Mikrodiziler genomun (bir canlıya ait tüm DNA bilgisi) bir çip üzerinde görüntülenmesini sağlayarak gen ifadesindeki farklılıkları ortaya çıkarmayı sağlayan bir teknolojidir [1]. Değişkenlerin sayısından daha az deney replikasyonu yapılması veya gen düzenlemesi hakkında sınırlı bilgi sahibi olunması gibi faktörler mikrodizi analizini karmaşık hale getirmektedir [2]. Ancak kullanılan yazılımların hızla gelişmesi ve yeni teknikler ile bu olumsuzluklar her geçen gün daha da aşılmakta ve başarılı sonuçlar elde edilmektedir. Illumina BeadStudio [3], rastgele düzenlenmiş boncuklara dayanan mikrodizi teknolojisi, prob replikasyonu ve düşük hibritleşme gibi avantajları ile mikrodizi analizlerinde tercih edilen bir

platformudur. Gunderson ve arkadaşları [4] çalışmalarında bu platformun, geliştirdiği rastgele düzenlenmiş boncuklara dayanan mikrodizi teknolojisi; prob replikasyonu, düşük hibritleşme maliyeti gibi özellikleri sayesinde her geçen gün daha popüler olduğunu ifade etmişlerdir. Wang ve arkadaşları [5] ise Illumina BeadStudio'nun rastgele düzenlenmiş boncuk tabanlı mikrodizi teknolojisi ile gen ekspresyon analizi ve genotipleme çalışmalarında yaygın kullanıldığını belirtmişlerdir. Platform, her bir boncuğun yüzeyinde rastgele ancak yüksek yoğunlukta ve çoğaltılmış prob dizilerinin bulunmasıyla, sinyal-gürültü oranını artırarak yüksek hassasiyetli sonuçlar elde edilmesini sağlar [6]. Böylece geleneksel sabit prob dizilerine göre daha hassas veri elde edilmesine olanak sağlar. Illumina BeadStudio, prob replikasyonu ve düşük hibritleşme maliyetleri gibi avantajlara sahip olmasının yanında, genetik varyasyonların ve gen ekspresyon profillerinin güvenilir ve tekrarlanabilir analizini mümkün kılmaktadır [7]. Illumina BeadStudio'da kullanılan rastgele düzenlenmiş boncuk yapısı, yüzeye yüksek yoğunlukta yerleştirilmiş mikroskobik boncuklar üzerine kuruludur. Her boncuk, belirli bir prob dizisini taşır. Bu dizilerin başka kopyaları da yüzeyde bulunur. Bu durum sinyal-gürültü oranını artırarak daha doğru sonuçlar elde edilmesine yardımcı olur [8]. Bu yaklaşım, geleneksel mikrodizi platformlarına göre daha düşük maliyetli ve daha güvenilir bir analiz seçeneği sağlar. Bu çalışmada Illumina BeadStudio ile hazırlanmış mikrodizi verileri kullanılmıştır.

NCBI (National Center of Biotechnology Information – Ulusal Biyoteknoloji Bilgi Merkezi) geniş bir veri tabanı koleksiyonuna sahip bir platformdur. <https://ncbi.nlm.nih.gov> adresinden bu platforma erişim sağlanabilmektedir. NCBI altındaki GEO (Gene Expression Omnibus – Gen İfade Dizini) açık erişimli bir genomik veri tabanıdır. Bu çalışmada GEO altından erişilebilen GSE8262 ve GSE10470 nolu veri kümeleri kullanılmıştır [9]. Bu veri kümeleri fareler üzerinde yapılan çalışmalardan oluşmaktadır. GSE8262 veri kümesi kronik hipoksi tedavisinin fare beyni üzerindeki etkisini incelemek için hazırlanmıştır. GSE10470 veri kümesi ise oksidatif strese maruz kalan fare nöronlarının gen üzerindeki etkisini incelemek için hazırlanmıştır.

Mikrodizi verilerinin hazırlanması için birçok farklı üreticinin hazırladığı yazılımlar kullanılmaktadır. Yayın çalışmasına konu olan GSE8262 ve GSE10470 veri kümeleri Illumina firmasının BeadStudio yazılımı ile hazırlanmıştır. Veri kümelerinin incelenmesinde ise R [10] programlama dilinin Bioconductor aracı kullanılmıştır [11]. Veri kümeleri GEO'dan elde edildikten sonra ön işleme tabi tutularak normalizasyon uygulanması ile gen tahmini için çalışmaya uygun hale getirilmesi sağlanmıştır.

## 2. Kaynak Taraması

### 2.1 Veri Kümesi

Çalışmada Illumina BeadStudio veri kümeleri kullanılmıştır. Üzerinde çalışma yapılan ilk veri kümesi; GSE8262 erişim numarası ile GEO platformundan paylaşım açılan, kronik hipoksi tedavisinden sonra gelişen fare beynindeki gen ekspresyon profilini içermektedir. Veri kümesi; gelişmekte

olan fare beyninin kortikal ve hipokampal bölgelerinde CCH (Chronic Consistent Hypoxia – Kronik Tutarlı Hipoksi) ve CIH (Chronic Intermittent Hypoxia – Kronik Aralıklı Hipoksi)'nin etkisini incelemek için hazırlanmıştır. Veri kümelerinde 18 örnek bulunmaktadır: Kortikal tipi için 3, hipokampüs tipi için 3 olmak üzere 6 kontrol grubu, CCH tedavisi uygulanan 6, CIH tedavisi uygulanan 6 olmak üzere 12 deney grubu bulunmaktadır. Veri kümesine ve detay bilgiye [ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE8262](https://ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE8262) [12]'den erişim sağlanabilmektedir.

İkinci veri kümesi GSE10470 erişim numarası ile GEO platformundan paylaşım açılan, oksidatif strese maruz kalan WT ve ATF4 KO embriyonik nöronlarının gen ekspresyon profilini içermektedir. Veri kümesinde 24 örnek bulunmaktadır: WT tipi için 8, ATF4 tipi için 8 olmak üzere 16 kontrol grubu (tedavi edilmemiş nöronlar), WT tipi için 4, ATF4 tipi için 4 olmak üzere 8 deney grubu (glutamat homolog homostein ile oksidatif strese maruz bırakılan nöronlar) bulunmaktadır. Veri kümesine ve detay bilgiye [ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE19470](https://ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE19470) [13]'den erişim sağlanabilmektedir.

### 2.2 Kaynak İnceleme

Seçilen veri kümelerini kullanan çalışmalar incelemiş ve aşağıdaki çalışmalara dahil edildiği görülmüştür. Uygulanan yöntemler ile bu verileri farklı açılardan değerlendirerek tedavi ve koruma süreçlerine katkı sağlayacak çıktılar elde edilmeye çalışılmıştır.

Zhou ve arkadaşlarının yaptığı çalışmada [14], kronik hipoksinin (oksijen yetmezliği) fare beynindeki gen ifadelerini nasıl değiştirdiği araştırılmıştır. Özellikle sarcospan (sspn) geninin hipoksi altındaki glial hücre ölümündeki rolü incelenmiştir. Çalışma kapsamında; denek farelerinin seçimi, hipoksi tedavisi ve incelenecek dokuların hazırlanması, RNA (RiboNükleikAsid)'ların çıkarılarak mikrodizi analizlerinin yapılması adımları izlenmiştir. CIH etkisi ile 80, CCH etkisi 137 genin etkilendiğine dikkat çekilmiştir. Hipoksi ile ilişkili nörolojik bozuklukların tedavisine olumlu katkılar sağlayabilecek çıktılar olan bu çalışma, kronik hipoksi koşullarının kortikal ve hipokampüs bölgelerinde gen ifadelerinde önemli değişikliklere sebep olduğunu göstermiştir. Bu deneysel çalışma sırasında hazırlanan veri, GSE8262 koduyla GEO [9] platformundan erişime açılmıştır.

GSE8262 veri kümesi ile GSE2385 ve GSE1357 veri kümelerinin kullanıldığı, Wang ve arkadaşlarının yaptığı çalışmada [15], aralıklı hipoksi ile tetiklenen nöroenflamasyon ve bilişsel gerilemede SENP1 (SUMP spesifik proteaz 1) proteininin etkisini incelemiştir. Mikroglia da meydana gelen değişiklikler hem laboratuvar ortamında (in vitro) hem de canlı organizma içinde (in vivo) incelenmiştir. Yaptıkları çalışmada SENP1 proteininin, NEMO proteininin SUMOlaşmasını (SUMO: 'Small Ubiquitin like MOdifiers' proteininin hücrelerdeki diğer proteinlere bağlanarak işlevlerini değiştirmesi) modüle ederek aralıklı hipoksiye bağlı nöroenflamasyonu düzenlediğini ve NF-κB yolunu aktive ettiğini ifade etmişlerdir. Bu bulguların mikroglia da SENP1'in modüle edilmesinin aralıklı hipoksinin neden olduğu bilişsel

bozuklukların tedavisinde potansiyel bir hedef olabileceğini gösterdiğini belirtmişlerdir.

Wang ve arkadaşlarının yaptığı çalışmada [16], SENP1'in OSAS (Obstructive Sleep Apnea Syndrome – Obstrüktif Uyku apnesi Sendromu) ile ilişkili CIH'in neden olduğu nöroiltihaplanma ve nöronal hasara etkisini araştırmışlardır. Çalışma için SENP1 aşırı eksprese edilmiş mikroglia ve SENP1 eksik fareler kullanılarak, CIH mikroglia ve fare modelleri oluşturulmuştur. Çalışmada GSE8262, GSE2385 ve GSE1357 veri kümeleri kullanılmıştır. Yapılan çalışma sonucunda CIH'in SENP1 ve TOM1 seviyelerini azalttığını, TOM1 proteininin SUMOlaşmasını artırdığını ve mikroglial göç, nöroiltihaplanma, A $\beta$ 42 (nöronal beta-amiloid42) birikimi ve apoptozu teşvik ettiğini belirtmişlerdir. SENP1'in TOM1 üzerindeki etkisinin, CIH'in neden olduğu bilişsel işlev bozukluklarının tedavisinde yeni yaklaşımlar sunabileceğini ifade etmişlerdir.

Lange ve arkadaşlarının yaptığı çalışmada [17], ATF4 (Activating Transcription Factor 4 – Aktive Edici Transkripsiyon Faktörü 4) proteininin oksidatif stres altında nöronlarda nasıl bir transkripsiyon faktörü olduğu incelenmiştir. Birçok nörodejeneratif hastalığın temelinde yatan, vücutta serbest radikallerin artması sonucu meydana gelen oksidatif strese tepki olarak glutasyon gibi antioksidanların tükenmesiyle ATF4 oluştuğuna dikkat çekilen çalışmada, ATF4'ün nöronlarda hücre ölümünü tetikleyen genlerin ifadesini artırdığı öne sürülmüştür. Çalışma kapsamında denek farelerinin seçimi, fare beyinlerinden elde edilen nöron kültürlerinin (oksidatif stres koşullarını sağlamak için) glutasyon tükenmesi gibi çeşitli stres faktörlerine maruz bırakılması hem canlı organizmada hem de laboratuvar ortamında incelemelerin yapılması, RNA'ların çıkarılarak mikrodizi analizlerinin yapılması süreçleri uygulanmıştır. Çalışma çıktısı olarak, ATF4'ün sinir sistemi hücrelerinde oksidatif stres ile ilişkili hücre ölümü süreçlerini başlattığını ve oksidatif stresin rol oynadığı hastalıkların tedavisinde hedeflenebileceğini belirtmişlerdir. Bu deneysel çalışma sırasında hazırlanan veri, GSE10470 koduyla GEO platformundan erişime açılmıştır.

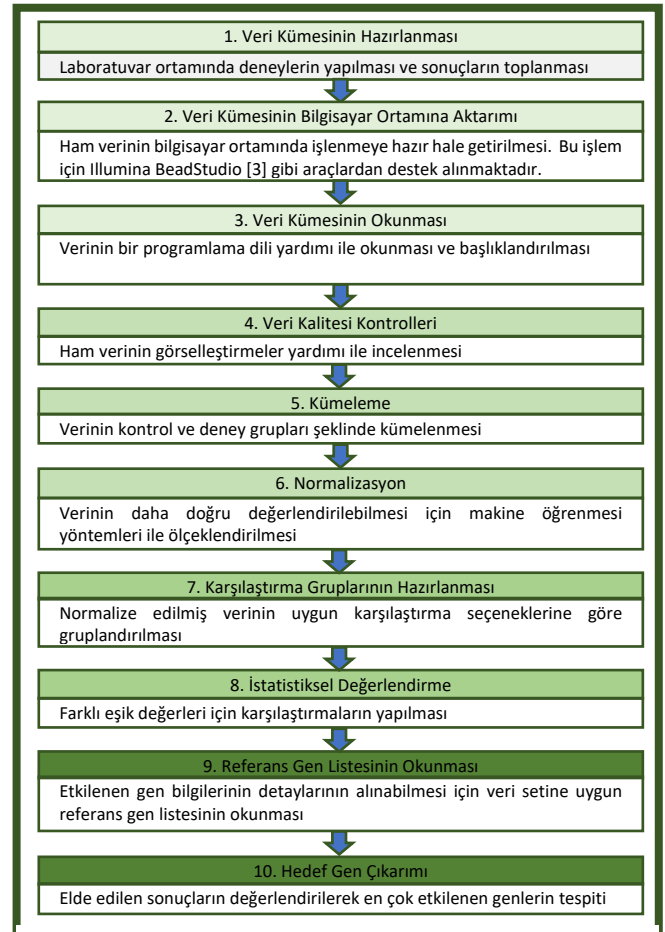
Woo ve arkadaşlarının yaptığı çalışmada [18], merkezi sinir sisteminde meydana gelen iltihaplanmada GRM8 (Glutamate Receptor, Metabotropic 8 – Metabotropik Glutamat Reseptörü 8)'in nöroprotektif rolü incelenmiştir. Bu çalışmada GSE10470 veri kümesi dışında birçok veri kümesi kullanılmıştır. MS, Alzheimer, Parkinson gibi nörodejeneratif hastalıklarda glutamat seviyesinin artması, nöronların aşırı uyarılmasına ve ölümüne neden olmaktadır. Çalışma, GRM8'in nörodejeneratif hastalıklarda glutamat seviyesini düzenleyerek nöronları koruduğunu öne sürmektedir. Çalışma çıktısı olarak elde ettikleri bulgular ile GRM8'in nörodejeneratif hastalıkların tedavisinde hedef olarak değerlendirilebileceğini belirtmişlerdir.

Bloomingdale ve arkadaşlarının yaptığı çalışmada [19], bortezomib isimli kemoterapi ilacının neden olduğu periferik nöropatiyi (el ve ayaklarda karıncalanma, uyuşma, ağrı gibi belirtiler) iyileştirmek için yeni bir strateji geliştirilmesi hedeflenmiştir. Kemoterapi kaynaklı nöropatik ağrının tedavi

edilmesi için ilaç geliştirme amaçlı çalışmada; GSE10470 veri kümesi ile bortezomib ile tetiklenen nöropatik ağrı yaşayan farelerde, sinir iletim hızları ve ağrı davranışları ölçülmüştür. Ayrıca laboratuvar ortamında, sinir hücreleri üzerinde bortezomib ve dexanabinol tedavisi sonrası çıktılar değerlendirilmiştir. Çalışmanın çıktısı olarak, dexanabinol'ün kemoterapiye bağlı nöropatik tedavide iyileştirici etkisi olabileceği belirtilmiştir.

### 3. Araçlar ve Yöntem

Mikrodizi verilerinin makine öğrenmesi yöntemleri ile analiz edilmesinde bu çalışmada da kullanılan Şekil-1'deki adımlar izlenebilmektedir. Şekil-1'deki 1. ve 2.adımlar [14] ve [17] yayınlarında detaylandırıldığı şekilde hazırlanmıştır ve çalışmada bu veri kümeleri kullanılmıştır. Diğer adımlar bu bölüm altındaki alt başlıklarda detaylandırıldığı şekilde uygulanmıştır. Veri kümesinin okunması süreci 3.1 başlığı altında açıklandığı şekilde GEO platformundan verinin elde edilmesi ve R programlama dili ile çalışmaya hazır hale getirilmesi şeklinde uygulanmıştır. Veri kalitesi kontrolleri için görselleştirme desteği de kullanılarak kontroller sağlanmıştır, çalışma detaylarına 3.2 başlığı altında değinilmiştir. Kümeleme çalışması 3.3 başlığı altında detaylandırıldığı gibi pearson korelasyona bağlı kümeleme kullanılarak yapılmıştır. Kümeleme işleminin ardından uygulanan niceliksel normalizasyon süreci 3.4 başlığı altında açıklanmıştır. Normalizasyon işlemi tamamlandıktan sonra 3.5 başlığında



Şekil-1: Mikrodizi verilerinin makine öğrenmesi yöntemleri ile analiz aşamaları

değindildiği gibi karşılaştırma grupları hazırlanmıştır. Hazırlanan karşılaştırma grupları üzerinde istatistiksel değerlendirmeler yapılmıştır. Yapılan değerlendirmeler 3.6 başlığı altında görselleştirme desteği ile detaylandırılmıştır. Değerlendirmelerin tamamlanması sonrasında sonuç verilerinin anlamlandırılabilmesi amacıyla veri kümelerine uygun referans gen listeleri okunmuştur (3.7 başlığı altında detay bilgiler bulunmaktadır). Son adım olarak hedef gen listeleri çıkarılarak 3.8 başlığı altında en çok etkilenen 20 gen listelenmiştir.

### 3.1 Veri Kümesinin Okunması

Veri kümesinin analizler için hazırlanması için kullanılacak yöntem belirlenirken veri kümesi dosyasının hangi platformda hazırlandığı, biçimi ve veri kümesinin açık erişim platformunda olup olmaması gibi bilgiler dikkate alınır. Çalışmaya konu [12] ve [13] veri kümeleri açık erişim platformunda olup, NCBI - GEO platformundan elde edilmiştir. Veri kümeleri üzerinde analizlerin yapılabilmesi için gerekli olan başlıklandırma işlemi R programlama dilinin Bioconductor aracı kullanılarak gerçekleştirilerek veri çerçevesi (dataframe) oluşturulmuştur.

Bu veri çerçevesinde;

- Sıra başlığı her bir deneyin sıra numarasını göstermektedir.
- Grup başlığı deneylerin gerçekleştirildiği grup bilgisidir. Veri kümeleri gruplar halinde alınarak işlenmektedir. Buradaki bilgi ilgili deneyin hangi grupta olduğu bilgisini gösterir.
- GenTipi başlığı deneylere konu gen tiplerini göstermektedir.
- Replika başlığı aynı koşullar altında tekrarlanan deneylerin numarasıdır. Birden çok replika kullanılarak verinin daha doğru değerlendirilmesi hedeflenmektedir.
- İlaç başlığı deney gruplarına uygulanan işlemi temsil eder. Bu işlem bir ilacın veya tedavinin uygulanması şeklinde olabilmektedir. Buradaki “ctrl” kontrol gruplarını ifade eder.
- Dizi kodu başlığı her bir deneyin ayırt edici kimlik bilgisidir. Her bir deney için benzersiz/tanımlayıcı bilgi niteliğindedir.

GSE8262 veri kümesi gelişen fare beyninin kortikal ve hipokampal bölgesinde CCH (kronik tutarlı hipoksi) ve CIH (kronik aralıklı hipoksi) tedavisi uygulanan deneye ait veri kümesi içermektedir. Veri kümesi 18 örnek içerip 145 kolon ve 46120 satırdan oluşmaktadır.

Bu veri kümesi için hazırlanan veri çerçevesi ise Çizelge-1’deki gibidir. Bu çerçeve her bir deney için veri çerçevesi bilgisini gösterir. Bu veri kümesindeki gen tipleri “Cortex” fare beynin kortikal bölgesine ait örnekleri, “HCampus” ise fare beyninin hipokampal bölgesine ait örnekleri ifade etmektedir. İlaç kısmındaki “ctrl” kontrol gruplarını, “cih” kronik aralıklı

hipoksi tedavisi uygulanan deneyleri, “cch” ise kronik tutarlı hipoksi tedavisi uygulanan örnekleri ifade etmektedir.

Çizelge-1: Veri Çerçevesi İçeriği – GSE8262

Sıra	Grup	GenTipi	Replika	İlaç	DiziKodu
1	1	Cortex	1	ctrl	1412023004_A
2	1	Cortex	1	cih	1412023004_B
3	1	Cortex	1	cch	1412023004_C
4	1	HCampus	1	ctrl	1412023004_D
5	1	HCampus	1	cih	1412023004_E
6	1	HCampus	1	cch	1412023004_F
7	2	Cortex	2	ctrl	1412023005_A
8	2	Cortex	2	cih	1412023005_B
9	2	Cortex	2	cch	1412023005_C
10	2	HCampus	2	ctrl	1412023005_D
11	2	HCampus	2	cih	1412023005_E
12	2	HCampus	2	cch	1412023005_F
13	3	Cortex	3	ctrl	1412023006_A
14	3	Cortex	3	cih	1412023006_B
15	3	Cortex	3	cch	1412023006_C
16	3	HCampus	3	ctrl	1412023006_D
17	3	HCampus	3	cih	1412023006_E
18	3	HCampus	3	cch	1412023006_F

GSE10470 veri kümesi ise ATF4 ve WT genotiplerine HCA uygulanan deneye ait veri kümesini içermektedir. Veri kümesi 24 örnek içerip 193 kolon ve 24048 satırdan oluşmaktadır.

Bu veri kümesi için hazırlanan veri çerçevesi ise Çizelge-2’deki gibidir. Bu veri kümesindeki gen tipleri “wt (wild type)” normal farelere ait örnekleri, “ko (knockout)” ise ATF4 eksikliği olan farelere ait örnekleri ifade etmektedir. İlaç kısmındaki “ctrl” kontrol gruplarını, “hca” ise glutamat homolog homostein ile oksidatif strese maruz bırakılan örnekleri ifade etmektedir.

Çizelge-2: Veri Çerçevesi İçeriği – GSE10470

Sıra	Grup	GenTipi	Replika	İlaç	DiziKodu
1	1	wt	1	ctrl	1412066061_A
2	1	wt	2	ctrl	1412066061_B
3	1	wt	3	ctrl	1412066061_C
4	1	wt	4	ctrl	1412066061_D
5	1	ko	1	ctrl	1412066061_E
6	1	ko	2	ctrl	1412066061_F
7	1	ko	3	ctrl	1412066061_G
8	1	ko	4	ctrl	1412066061_H
9	2	ko	5	ctrl	1541554116_A
10	2	wt	1	hca	1541554116_B
11	2	ko	1	hca	1541554116_C
12	2	wt	5	ctrl	1541554116_D
13	2	ko	6	ctrl	1541554116_E
14	2	wt	2	hca	1541554116_F
15	2	ko	2	hca	1541554116_G
16	2	wt	6	ctrl	1541554116_H
17	3	wt	7	ctrl	1541554117_A
18	3	ko	3	hca	1541554117_B
19	3	wt	3	hca	1541554117_C
20	3	ko	7	ctrl	1541554117_D
21	3	wt	8	ctrl	1541554117_E
22	3	ko	4	hca	1541554117_F
23	3	wt	4	hca	1541554117_G
24	3	ko	8	ctrl	1541554117_H

### 3.2 Veri Kalitesi Kontrolleri

Okunan veri kümesinin hazırlanan veri çerçevesi ile eşleştirilmesi sonrası veri gruplaması Çizelge-3 ve Çizelge-4'teki gibi olmuştur.

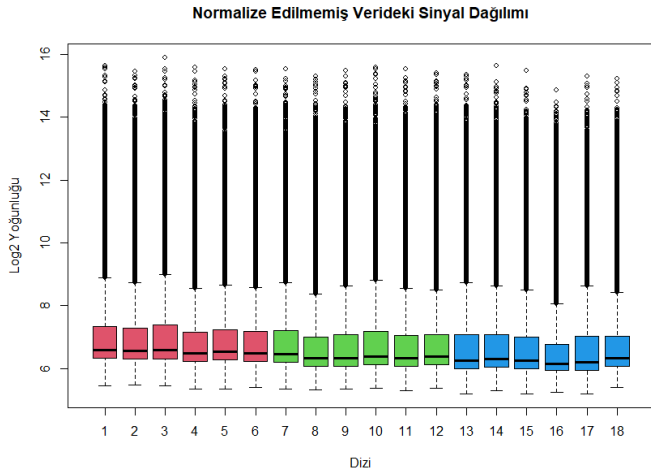
Çizelge-3: Veri Gruplaması – GSE8262

"Cortex.ctrl.1"	"Cortex.ctrl.7"	"Cortex.ctrl.13"
"Cortex.cih.2"	"Cortex.cih.8"	"Cortex.cih.14"
"Cortex.cch.3"	"Cortex.cch.9"	"Cortex.cch.15"
"HCampus.ctrl.4"	"HCampus.ctrl.10"	"HCampus.ctrl.16"
"HCampus.cih.5"	"HCampus.cih.11"	"HCampus.cih.17"
"HCampus.cch.6"	"HCampus.cch.12"	"HCampus.cch.18"

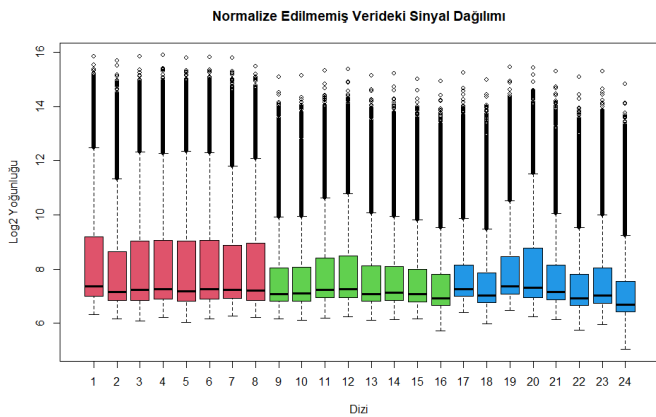
Çizelge-4: Veri Gruplaması – GSE10470

"wt.ctrl.1"	"ko.ctrl.7"	"ko.ctrl.13"	"wt.hca.19"
"wt.ctrl.2"	"ko.ctrl.8"	"wt.hca.14"	"ko.ctrl.20"
"wt.ctrl.3"	"ko.ctrl.9"	"ko.hca.15"	"wt.ctrl.21"
"wt.ctrl.4"	"wt.hca.10"	"wt.ctrl.16"	"ko.hca.22"
"ko.ctrl.5"	"ko.hca.11"	"wt.ctrl.17"	"wt.hca.23"
"ko.ctrl.6"	"wt.ctrl.12"	"ko.hca.18"	"ko.ctrl.24"

Sinyal dağılımı kontrol edildiğinde Şekil-2 ve Şekil-3'teki sonuçlara ulaşılmıştır.

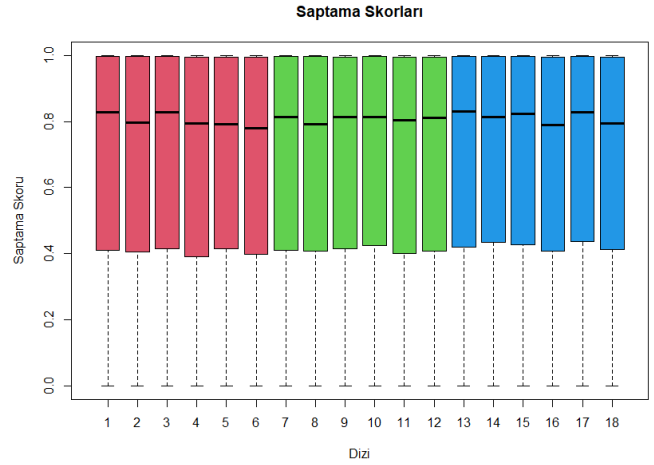


Şekil-2: Normalize Edilmemiş Verideki Sinyal Dağılımı – GSE8262

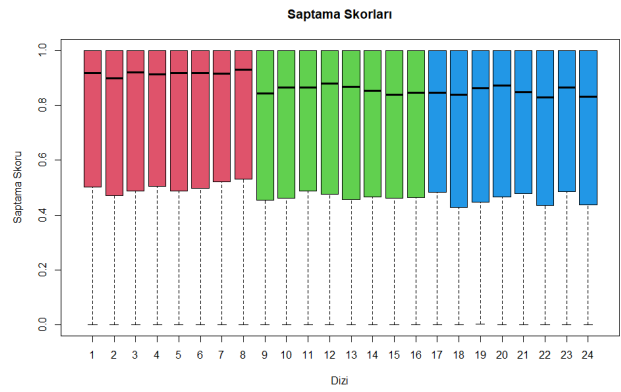


Şekil-3: Normalize Edilmemiş Verideki Sinyal Dağılımı – GSE10470

Veri kümelerinin saptama skorları ise Şekil-4 ve Şekil-5'teki gibidir.



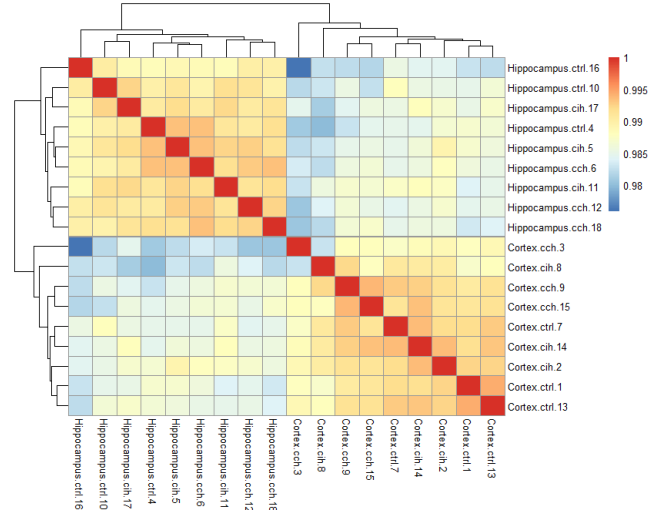
Şekil-4: Saptama Skorları – GSE8262



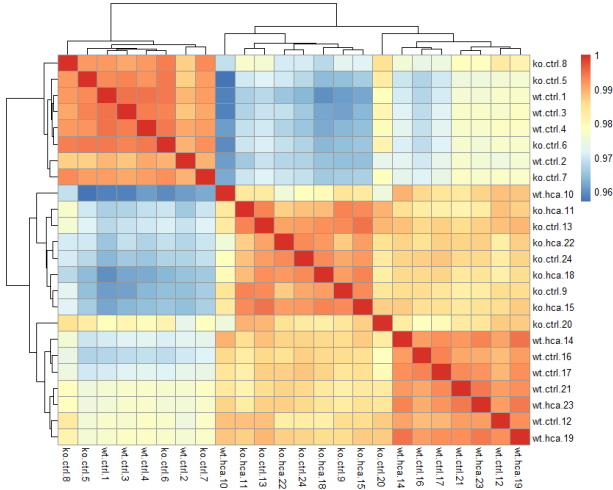
Şekil-5: Saptama Skorları – GSE10470

### 3.3 Kümeleme

Makine öğrenmesinde benzer veri gruplarını bir araya getirme işlemi olan kümeleme(clustering); verinin keşfi, anomali tespiti, özellik seçimi, segmentasyon gibi birçok süreçte kullanılmaktadır. Hiyerarşik kümeleme, yoğunluk tabanlı kümeleme (DBSCAN), K-Means kümeleme, spektral kümeleme, kavramsal kümeleme gibi birçok kümeleme tekniği mevcuttur. Bu çalışmada veri kümeleri üzerinde hiyerarşik kümeleme grubunda olan "pearson korelasyona bağlı kümeleme" kullanılmıştır.



Şekil-6: Pearson Korelasyona Bağlı Kümeleme – GSE8262



Şekil-7: Pearson Korelasyona Bağlı Kümeleme – GSE10470

### 3.4 Normalizasyon

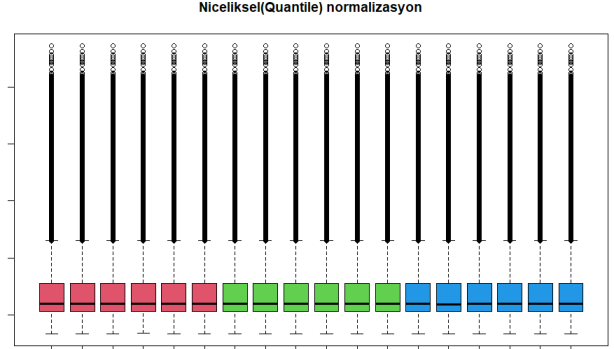
Normalizasyon veri kümesindeki özelliklerin belli bir aralığa veya ölçeğe dönüştürülmesi işlemidir. Veri kümesindeki büyük değer farklılıkları olan özelliklerde değerler 0-1 arasına ölçeklendirilerek daha doğru ve kolay değerlendirilebilir. Böylece farklı ölçeklerde olan verilerin karşılaştırılması mümkün olur. Min-Max normalizasyonu, Z-Score normalizasyonu, maksimum mutlak değer normalizasyonu (maks abs), niceliksel (quantile) normalizasyon gibi birçok farklı normalizasyon yöntemi mevcuttur.

Çalışmada kullanılan Bioconductor paketi altındaki Limma (linear models for microarray data) kütüphanesi gen ekspresyon analizlerinde tercih edilen bir araçtır. Bu kütüphane veri dağılımını standart hale getirme için çeşitli normalizasyon yöntemleri sağlamaktadır. Limma kütüphanesi niceliksel, ölçek, medyan, loess normalizasyon yöntemlerini sunmaktadır. Niceliksel normalizasyon örnekler arasındaki dağılımları eşitleyerek verileri standart hale getirirken, ölçek normalizasyon verilerin ortalama ve varyansını standartlaştırarak tutarlılığı artırır. Medyan normalizasyon her örneği medyan değerine göre ayarlayarak aşırı değer sorununu gidermeye yardımcı olurken, loess normalizasyon iki renkli mikrodizelerde yerel regresyon kullanarak dağılımı dengelemeye yardımcı olur.

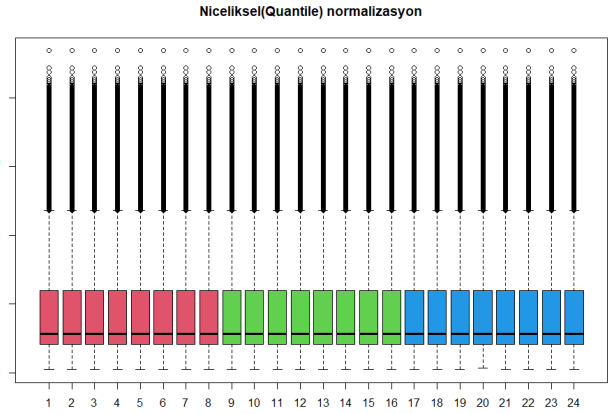
Niceliksel normalizasyon, yüksek boyutlu veri analizinde yaygın olarak kullanılan önemli bir normalizasyon tekniğidir [20]. Biyoformatikte gen ekspresyon verilerinin sıralanarak, tüm veri kümesinin aynı dağılımda olmasını sağlayan bir yöntem olarak sıkça kullanılmaktadır. Niceliksel normalizasyon, laboratuvar koşulları veya farklı örnek hazırlama teknikleri gibi sebeplerle mikrodizi deneylerinde oluşan teknik varyasyonları azaltarak verinin standart dağılımını sağlar. Bu yaklaşım, örnekler arasındaki teknik varyasyonları azaltarak değişimin daha doğru şekilde tespit edilmesini sağlar [21]. Ayrıca deneylerdeki gen ekspresyon dağılımını eşitleyerek, verilerin karşılaştırılabilirliğini artırır. Böylece farklı koşullar altında elde edilen verilerin daha anlamlı bir şekilde karşılaştırılması mümkün olur [22]. Niceliksel normalizasyon, biyolojik gürültüyü azaltarak daha tutarlı sonuçlar elde edilmesini sağlar. Biyolojik gürültünün

azaltılması özellikle düşük sinyalli genlerin analizinde homojen dağılım sağlayarak yanlış pozitif sonuçları azaltır [23]. Çalışmada biyolojik gürültü, teknik varyasyonu azaltması, veri karşılaştırma ve ölçeklenme sağlaması gibi avantajları sebebiyle niceliksel normalizasyon tercih edilmiştir.

Normalizasyon işlemi sonrasında sinyal dağılımları Şekil-8 ve Şekil-9'daki gibi değişiklik göstermiştir.

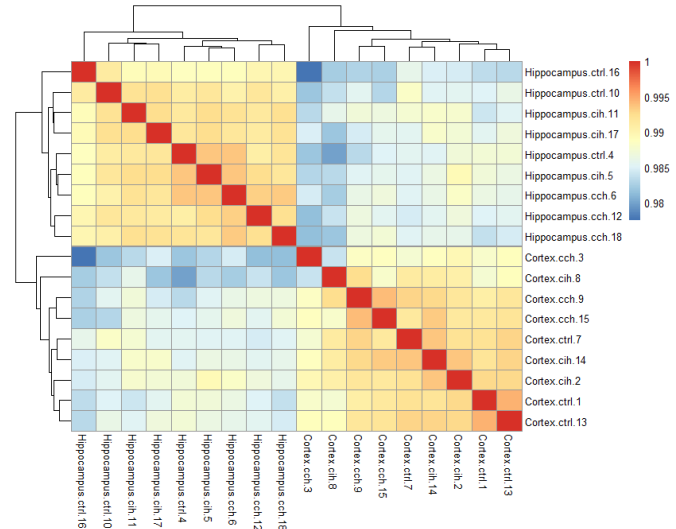


Şekil-8: Normalizasyon Sonrası Sinyal Dağılımı – GSE8262



Şekil-9: Normalizasyon Sonrası Sinyal Dağılımı – GSE10470

Normalizasyon sonrası sıcaklık haritası değişimleri ise Şekil-10 ve Şekil-11'deki gibidir.



Şekil-10: Normalizasyon Sonrası Kümelmiş Veri Sıcaklık Haritası – GSE8262







```

1 Illumina Inc. BeadStudio version 1.5.1.3
2 Normalization = none
3 Array Content = Mouse-6_V1.xml
4 Error Model = none
5 Date/Time = 12/15/2005 3:15 PM
6 Local Settings = en-US
7
8 TargetID MIN_Signal-1412023004 A AVG_Signal-1412023004 A MAX_Signal-141
9 10181072_239_rc-S 78.4 78.4 78.4 1 NaN 5.1 42 0.3539604 9
10 10181072_290_S 94 94 94 1 NaN 4 48 0.79022277 86.1 86.1 8
11 10181072_290_rc-S 99.1 99.1 99.1 1 NaN 6.7 37 0.84777228 9
12 10181072_311-S 71.9 71.9 71.9 1 NaN 2.8 48 0.1615099 73.3

```

Şekil-20: Referans Gen Listesi Bilgisinin Tespiti – GSE8262

```

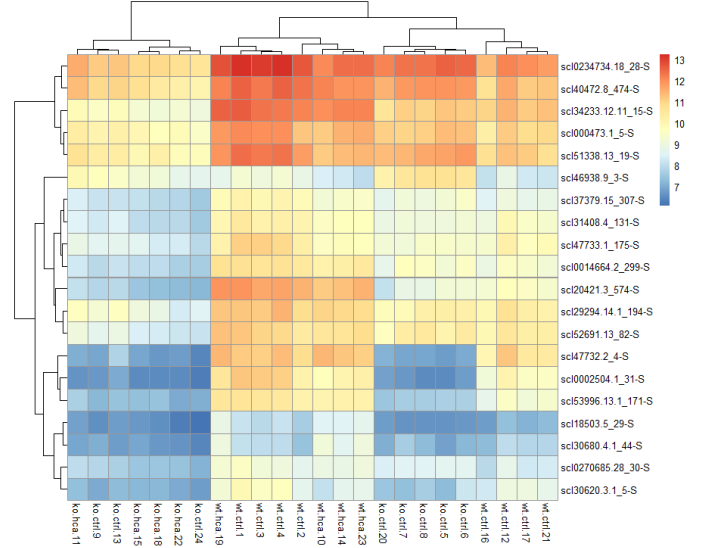
1 Illumina Inc. BeadStudio version 1.5.1.3
2 Normalization = none
3 Array Content = Mouse_Ref-8_V1.xml
4 Error Model = none
5 Date/Time = 3/3/2006 12:51 PM
6 Local Settings = en-US
7
8 TargetID MIN_Signal-1412066061 A AVG_Signal-1412066061 A MAX_Signal-141206
9 101_1RNA_X00604_523-S 6771.3 6771.3 6771.3 1 NaN 100.7 43 0.2179
10 GI_11079452-S 128.7 128.7 128.7 1 NaN 7.3 35 0.50542823 141.6 1
11 GI_21070949-S 17559.8 17559.8 17559.8 1 NaN 501.7 32 1 14079.4 14079
12 GI_21746160-S 14426.5 14426.5 14426.5 1 NaN 587.6 35 1 14459.9 14459

```

Şekil-21: Referans Gen Listesi Bilgisinin Tespiti – GSE10470

### 3.8 Hedef Gen Çıkarımı

Referans gen listesinin okunması ile ID'lere karşılık gen bilgilerinin alınabilmesi mümkün olmaktadır. Elde edilen diferansiyel ekspresyon sonuçları istatistiksel öneme göre sıralanmış ve bir ifade matrisi hazırlanarak en çok etkilenen 20 gen bilgisi seçilmiştir.



Şekil-24: Belirlenen Kriterlere Göre En Çok Etkilenen 20 Genin Matris Halinde Gösterimi – GSE10470

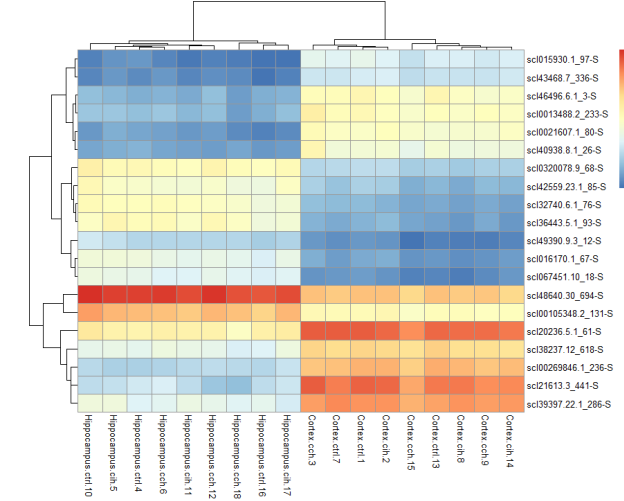
Transkript ID	Symbol	Definition
sci0234734.18_28-S	Aars	Mus musculus alanyl-tRNA synthetase (Aars) mRNA.
sci40472.8_474-S	Slc1a4	Mus musculus solute carrier family 1 (glutamate/neutral amino acid transporter) member 4 (Slc1a4) mRNA.
sci34233.12.11_15-S	Slc7a5	Mus musculus solute carrier family 7 (cationic amino acid transporter/neutral amino acid transporter) member 5 (Slc7a5) mRNA.
sci000473.1_5-S	Slc3a2	Mus musculus solute carrier family 3 (activators of dibasic and neutral amino acid transport) member 2 (Slc3a2) mRNA.
sci51338.13_19-S	Nars	Mus musculus asparaginyl-tRNA synthetase (Nars) mRNA.
sci46938.9_3-S	Slc25a17	Mus musculus solute carrier family 25 (mitochondrial carrier; peroxisomal membrane protein) member 17 (Slc25a17) mRNA.
sci37379.15_307-S	Shm2	Mus musculus serine hydroxymethyl transferase 2 (mitochondrial) (Shm2) mRNA.
sci31408.4_131-S	Atf5	Mus musculus activating transcription factor 5 (Atf5) mRNA.
sci47732.1_175-S	Mga3	Mus musculus mannoside acetylglucosaminyltransferase 3 (Mga3) mRNA.
sci0014664.2_299-S	Slc6a9	Mus musculus solute carrier family 6 (neurotransmitter transporter glycine) member 9 (Slc6a9) mRNA.
sci20421.3_574-S	1810008K03Rik	Mus musculus RIKEN cDNA 1810008K03 gene (1810008K03Rik) mRNA.
sci29294.14.1_194-S	Aens	Mus musculus asparagine synthetase (Aens) mRNA.
sci52691.13_82-S	Psa1	Mus musculus phosphoserine aminotransferase 1 (Psa1) mRNA.
sci47732.2_4-S	Atf4	Mus musculus activating transcription factor 4 (Atf4) mRNA.
sci0002504.1_31-S	Atf4	Mus musculus activating transcription factor 4 (Atf4) mRNA.
sci53996.13.1_171-S	Slc7a3	Mus musculus solute carrier family 7 (cationic amino acid transporter/neutral amino acid transporter) member 3 (Slc7a3) mRNA.
sci18503.5_29-S	Trib3	Mus musculus tribbles domain containing 3 (Trib3) mRNA.
sci30680.4.1_44-S	Nupr1	Mus musculus nuclear protein 1 (Nupr1) mRNA.
sci0270685.28_30-S	Fhfsd1	Mus musculus formyltetrahydrofolate synthetase domain containing 1 (Fhfsd1) mRNA.
sci30620.3.1_5-S	Cox6a2	Mus musculus cytochrome c oxidase subunit VI a polypeptide 2 (Cox6a2) mRNA.

Şekil-25: En Çok Etkilenen Genler Sembol ve Tanımları – GSE10470. Mouse\_Ref-8\_V1 referans gen listesindeki bilgilerden oluşturulmuştur.

## 4. Sonuç ve Tartışma

Bu çalışmada mikrodizi veri kümelerinde gen ekspresyon analizi çalışması yapılmıştır. Illumina BeadStudio ile hazırlanmış olan, kronik hipoksi tedavisi ve oksidatif stresin fareler üzerindeki etkisini inceleyen iki farklı veri kümesi kullanılmıştır. "Mikrodizi verilerinin makine öğrenmesi yöntemleri ile analizi" işlem hattı her iki veri kümesinde de başarıyla uygulanarak sonuçlar elde edilmiştir. Bu işlem hattı ile verinin okunmasından gen listelerinin elde edilmesine kadar tüm süreç adım adım incelenmiştir. Doğru yöntem ve değerlendirme metodolojileri belirlenerek uygulanmıştır. Görselleştirmeler kullanılarak sürecin anlaşılabilirliği artırılmıştır. Çalışmamız mikrodizi analizinde kullanılabilir yöntemler ve izlenmesi gereken adımları göstermesi açısından benzer çalışmalara katkı sağlayabilir.

Mikrodizi teknolojisi ile gen ekspresyon analizi yaygın bir yöntem olup, kanser gibi hastalıkların tedavi süreçlerinin iyileştirilmesi, hastalığın ilerlemesinin yavaşlatılması/durdurulması, hastalığın önlenmesi, etkin ilaç geliştirilmesi veya ilaçlara bağlı yan etkilerin azaltılması/giderilmesi gibi birçok süreçte fayda



Şekil-22: Belirlenen Kriterlere Göre En Çok Etkilenen 20 Genin Matris Halinde Gösterimi – GSE8262

Transkript ID	Symbol	Definition
sci015930.1_97-S	Indo	Mus musculus indoleamine-pyrrole 2
sci43468.7_336-S	Ist1	Mus musculus ISL1 transcription factor
sci46496.6.1_3-S	Tnnc1	Mus musculus troponin C
sci0013488.2_233-S	Drd1a	Mus musculus dopamine receptor DIA (Drd1a)
sci0021607.1_80-S	Tcrb-V8.2	
sci40938.8.1_26-S	Ppp1r1b	Mus musculus protein phosphatase 1
sci0320078.9_68-S	Olfm2b	Mus musculus olfactomedin-like 2B (Olfm2b)
sci42559.23.1_85-S	Slc26a4	Mus musculus solute carrier family 26
sci32740.6.1_76-S	Klk8	Mus musculus kallikrein related-peptidase 8 (Klk8)
sci36443.5.1_93-S	Spink8	Mus musculus serine peptidase inhibitor
sci49390.9.3_12-S	Pkp2	Mus musculus plakophilin 2 (Pkp2)
sci016170.1_67-S	Il16	Mus musculus interleukin 16 (Il16)
sci067451.10_18-S	Pkp2	Mus musculus plakophilin 2 (Pkp2), mRNA.
sci48640.30_694-S	Dgkg	
sci00105348.2_131-S	Golph2	Mus musculus golgi phosphoprotein 2 (Golp2)
sci20236.5.1_61-S	6330527006Rik	
sci38237.12_618-S	A130090K04Rik	
sci00269846.1_236-S	LOC665506	PREDICTED: Mus musculus similar to T-cell receptor beta 2 chain C region (LOC665506)
sci21613.3_441-S	Gpr88	Mus musculus G-protein coupled receptor 88 (Gpr88)
sci39397.22.1_286-S	Rgs9	Mus musculus regulator of G-protein signaling 9 (Rgs9)

Şekil-23: En Çok Etkilenen Genler Sembol ve Tanımları – GSE8262. Mouse-6.V1 referans gen listesindeki bilgilerden oluşturulmuştur.

sağlayabilmektedir. Schena ve arkadaşlarının [27], Lockhart ve Winzelerin [28], Alizadeh ve arkadaşlarının [29], Irizarry ve arkadaşlarının [30], Golub ve arkadaşlarının [31], Van 't Veer ve arkadaşlarının [32] yaptığı çalışmalar gibi birçok çalışmada mikrodizi gen ekspresyon analizinin bu yöndeki etkilerine değinilmiştir. Çalışmamızda elde ettiğimiz sonuçların da tanı, tedavi ve ilaç geliştirme süreçlerine olumlu katkı sağlayabileceği düşünülmektedir.

## Kaynakça

- [1] Yıldız Çavdar Z., Ensari T., Türker Şener L., Sertbaş A. Makine Öğrenmesi Tabanlı Mikrodizi Tekniği ile MikroRNA Hedef Tahmini: Araştırma Çalışması, Avrupa Bilim ve Teknoloji Dergisi, 2022, pp. 39-45.
- [2] Cordero F., Botta M., Calogero R.A. Microarray data analysis and mining approaches, Briefings in Functional Genomics and Proteomics, 2008, pp. 265-281.
- [3] Illumina BeadStudio, <https://www.illumina.com> (Erişim Tarihi: 22.05.2024).
- [4] Gunderson K.L., Kruglyak S., Graige M.S., Garcia F., Kermani B.G., Zhao C., Che D., Dickinson T., Wickham E., Bierle J., Doucet D., Milewski M., Yang R., Siegmund C., Haas J., Zhou L., Oliphant A., Fan J.B., Barnard S., Chee M.S. Decoding Randomly Ordered DNA Arrays, Genome Research, 2004, pp. 870-877.
- [5] Wang Y., Mooradian A., Cheng Y. High-throughput SNP genotyping using the BeadArray technology, microarray data analysis and mining approaches, Genomics, 2008, pp.99-110.
- [6] Bibikova M., Fan J.B. Genome-wide DNA methylation profiling using Infinium assay, Epigenomics, 2006, pp.177-200.
- [7] Steemers F.J., Chang W., Lee G. Whole-genome genotyping with the single-base extension assay, Nature Methods, pp.851-856.
- [8] Gunderson K.L., Steemers F.J., Lee G. A genome-wide scalable SNP genotyping assay using microarray technology, Genomics, 2004, pp.549-554.
- [9] National Center of Biotechnology Information, <https://ncb.nlm.nih.gov> (Erişim Tarihi: 22.05.2024).
- [10] R Programlama Dili, <https://www.r-project.org/> (Erişim Tarihi: 22.05.2024).
- [11] Bioconductor, <https://www.bioconductor.org/> (Erişim Tarihi: 22.05.2024).
- [12] GEO GSE8262 Veri Kümesi, <https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE8262> (Erişim Tarihi: 22.05.2024).
- [13] GEO GSE10470 Veri Kümesi, <https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE10470> (Erişim Tarihi: 22.05.2024).
- [14] Zhou D., Wang J., Zapala M.A., Xue J., Schork N.J., Haddad G.G. Gene expression in mouse brain following chronic hypoxia: role of sarcospan in glial cell death, Physiol Genomics, 2008, pp. 370-379.
- [15] Wang H., Yang T., Sun J., Zhang S., Liu S. SENP1 modulates microglia-mediated neuroinflammation toward intermittent hypoxia-induced cognitive decline through the de-SUMOylation of NEMO, Journal of Cellular and Molecular Medicine, 2021, pp. 6463-7099.
- [16] Wang H., Wang X., Shen Y., Wang Y., Yang T., Sun J., Liu S. SENP1 modulates chronic intermittent hypoxia-induced inflammation of microglia and neuronal injury by inhibiting TOM1 pathway, International Immunopharmacology, 2023, pp. 1-13.
- [17] Lange P.S., Chavez J.C., Pinto J.T., Coppola G., Sun C.W., Townes T.M., Geschwind D.H., Ratan R.R. ATF4 is an oxidative stress-inducible, prodeath transcription factor in neurons in vitro and in vivo, The Journal of Experimental Medicine, 2008, pp. 1227-1242.
- [18] Woo M.S., Ufer F., Rothhammer N., Liberto G.D., Binkle L., Haferkamp U., Sonner J.K., Engler J.B., Hornig S., Bauer S., Wagner I., Egervari K., Raber J., Duvoisin R.M., Pless O., Merkler D., Friese M.A. Neuronal metabotropic glutamate receptor 8 protects against neurodegeneration in CNS inflammation, The Journal of Experimental Medicine, 2021, pp. 1-30.
- [19] Bloomingdale P., Meregalli C., Pollard K., Canta A., Chiorazzi A., Fumagalli G., Monza L., Pozzi E., Alberti P., Ballarini E., Oggioni N., Carlson L., Liu W., Ghandili M., Ignatowski T.A., Lee K.P., Moore M.J., Cavaletti G., Mager D.E. Systems Pharmacology Modeling Identifies a Novel Treatment Strategy for Bortezomib-Induced Neuropathic Pain, Frontiers in Pharmacology, 2022, pp. 1-16.
- [20] Zhao Y., Wong L., Goh W.W.B. How to do quantile normalization correctly for gene expression data analyses. Scientific Reports, 2020, pp. 1-11.
- [21] Bolstad B.M., Irizarry R.A., Astrand M., Speed T.P. A comparison of normalization methods for high-density oligonucleotide array data based on variance and bias. Bioinformatics, 2003, pp. 185-193.
- [22] Irizarry R.A., Hobbs B., Collin F., Beazer-Barclay Y.D., Antonellis K.J., Scherf U., Speed T.P. Exploration, normalization, and summaries of high-density oligonucleotide array probe level data. Biostatistics, 2003, pp. 249-264.
- [23] Smyth G.K., Speed T., W.W.B. Normalization of cDNA microarray data. Methods, 2003, pp. 265-273.
- [24] Phipson B., Lee S., Majewski I.J., Alexander W.S., Smyth G.K. Robust Hyperparameter Estimation Protects Against Hypervariable Genes and Improves Power to Detect Differential Expression. The Annals of Applied Statistics, 2016, pp. 946-963.
- [25] Aubert J., Bar-Hen A., Daudin J.J., Robin S. Determination of the differentially expressed genes in microarray experiments using local FDR. BMC Bioinformatics, 2004.
- [26] Pawitan Y., Michiels S., Koscielny S., Gusnanto A., Ploner A. False discovery rate, sensitivity and sample size for microarray studies. Bioinformatics, 2005, pp. 3017-3024.
- [27] Schena M., Shalon D., Davis R.W., Brown P.O. Quantitative monitoring of gene expression patterns with a complementary DNA microarray. Science, 1995, pp. 467-470.
- [28] Lockhart D.J., Winzeler E.A. Genomics, gene expression and DNA arrays. Nature, 2000, pp. 827-836.
- [29] Alizaden A.A., Eisen M.B., Davis R.E., Ma C., Lossos I.S., Rosenwald A., Staudt L.M. Distinct types of diffuse large B-cell lymphoma identified by gene expression profiling. Nature, 2000, pp. 503-511.
- [30] Irizarry R.A., Hobbs B., Collin F., Beazer Barclay Y.D., Antonellis K.J., Scherf U., Speed T.P. Exploration, normalization, and summaries of high-density oligonucleotide array probe level data. Biostatistics, 2003, pp. 249-264.
- [31] Golub T.R., Slonim D.K., Tamayo P., Huard C., Gaasenbeek M., Mesirov J.P., Lander E.S. Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. Science, 1999, pp. 531-537.
- [32] Van't Veer L.J., Dai H., Van de Vijver M.J., He Y.D., Hart A.A., Mao M., Friend S.H. Gene expression profiling predicts clinical outcome of breast cancer. Nature, 2002, pp. 530-536.