

ARALIK'21 DECEMBER'21

Sayı/Number: 22/Volume: 11
Yıl/Year: 2021

Yayın Sahibi

TMMOB
Elektrik Mühendisleri Odası Adına
Bülent PALA

Sorumlu Yazı İşleri Müdürü

Can İNCEBİYİK

Yayın İdare Merkezi

Ihlamur Sokak No:10 Kat:3
Kızılay - Ankara
Tel: (312) 425 32 72
Faks: (312) 417 38 18
<http://bilimseldergi.emo.org.tr>
bilimseldergi@emo.org.tr
EMO Üyelerine parasız dağıtılır

Dergi Koordinatörü

Sevgi Kınacı

Sayfa Düzeni

TMMOB
Elektrik Mühendisleri Odası

Yayın Türü

Yerel Süreli Yayın
6 ayda bir yayınlanır

Basım Adedi

500

Basım Tarihi

Aralık 2021

EMO BİLİMSEL DERGİ

Elektrik, Elektronik, Bilgisayar, Biyomedikal
Mühendisliği Bilimsel Dergisi

The Journal of Electrical, Electronics, Computer and
Biomedical Engineering

YAYIN KURULU

BAŞ EDITÖR/EDITOR IN CHIEF

Prof. Dr. Timur Aydemir
Kadir Has Üniversitesi

EDİTÖRLER/EDITORIAL BOARD

Prof. Dr. Erhan Akın
Fırat Üniversitesi

Prof. Dr. H. Altay Güvenir
Bilkent Üniversitesi

Prof. Dr. Güven Önbilgin
Ondokuz Mayıs Üniversitesi

Prof. Dr. Arif Nacaroğlu
Gaziantep Üniversitesi

Prof. Dr. Özlem Özgün
Hacettepe Üniversitesi



TMMOB

Elektrik Mühendisleri Odası

UCTEA/Chamber of Electrical Engineers

EMO Bilimsel Dergi Danışma Kurulu

Prof. Dr. A. Hamit Serbest	Çukurova Üniversitesi
Prof. Dr. Altay Güvenir	Bilken Üniversitesi
Prof. Dr. Cengizhan Öztürk	Boğaziçi Üniversitesi
Prof. Dr. Erhan Akın	Fırat Üniversitesi
Prof. Dr. Güven Önbilgin	Ondokuz Mayıs Üniversitesi
Prof. Dr. Murat Eyüboğlu	Orta Doğu Teknik Üniversitesi
Prof. Dr. Timur Aydemir	Gazi Üniversitesi
Prof. Dr. Ali Hikmet Doğru	Orta Doğu Teknik Üniversitesi
Prof. Dr. Arif Nacaroğlu	Gaziantep Üniversitesi
Prof. Dr. Atilla Bir	İstanbul Teknik Üniversitesi
Prof. Dr. Aydın Köksal	Bilişim A.Ş.
Prof. Dr. Aydoğan Özdemir	İstanbul Teknik Üniversitesi
Prof. Dr. Aydın Baytan Ertüzün	Boğaziçi Üniversitesi
Prof. Dr. Belgin Türkay	İstanbul Teknik Üniversitesi
Prof. Dr. Bülent Sankur	Boğaziçi Üniversitesi
Prof. Dr. Cüneyt Güzeliş	Yaşar Üniversitesi
Prof. Dr. Erdal Panayırıcı	Kadir Has Üniversitesi
Prof. Dr. Erkan Afacan	Gazi Üniversitesi
Prof. Dr. Ferit Acar Savacı	İzmir İleri Teknoloji Enstitüsü
Prof. Dr. H. Bülent Ertan	Atılım Üniversitesi
Prof. Dr. Haldun Karaca	Dokuz Eylül Üniversitesi
Prof. Dr. İbrahim Eksin	İstanbul Teknik Üniversitesi
Prof. Dr. İnci Çilesiz	İstanbul Teknik Üniversitesi
Prof. Dr. İrfan Karagöz	Gazi Üniversitesi
Prof. Dr. İsmail Hakkı Altaş	Karadeniz Teknik Üniversitesi
Prof. Dr. İsmail Hakkı Çavdar	Karadeniz Teknik Üniversitesi
Prof. Dr. Kemal Leblebicioğlu	Orta Doğu Teknik Üniversitesi
Prof. Dr. Lale Tükenmez Ergene	İstanbul Teknik Üniversitesi
Prof. Dr. Mithat İdemen	İstanbul Teknik Üniversitesi
Prof. Dr. Muhittin Gökmen	MEF Üniversitesi
Prof. Dr. Murat Aşkar	İzmir Ekonomi Üniversitesi
Prof. Dr. Müjde Güzelkaya	İstanbul Teknik Üniversitesi
Prof. Dr. Osman Eroğul	TOBB Ekonomi ve Teknoloji Üniversitesi
Prof. Dr. Oya Kalıpsız	Yıldız Teknik Üniversitesi
Prof. Dr. Özlem Özgün	Hacettepe Üniversitesi
Prof. Dr. Sermin Onaygil	İstanbul Teknik Üniversitesi
Prof. Dr. Tayfun Akgül	İstanbul Teknik Üniversitesi
Doç. Dr. Ahmet Koltuksuz	Yaşar Üniversitesi
Doç. Dr. Osman Abul	TOBB Ekonomi ve Teknoloji Üniversitesi
Bora Güngören	Portakal Teknoloji
Fikret Küçükdeveci	TEPA
Hasan Ali Pazar	Siemens

İÇİNDEKİLER / CONTENTS

Klinik Arařtırmalarda Geliřtirilen Yapay Zekâ Modelleri için Yol Haritası
Roadmap for Artificial Intelligence Models Developed in Clinical Research
Alican Ařan, Duygu Sinanç Terzi
Makale Türü: Derleme

7

Yazılım Mühendisliğinde Çevik Eğitim Modeli (ÇEM)
Agile Education Model in Software Engineering
Erkan Sarıkaya, Selami Bağrıyanık, Mesut Gökalg
Makale Türü: Arařtırma Makalesi

15

Yapay Zeka Tabanlı Klinik Karar Destek Sistemi ile Tüp Bebek Tedavisi Gebelik Sonucu Tahmini
Predicting Pregnancy Outcome in In Vitro Fertilization Using an Artificial Intelligence Based Clinical Decision Support System
Zeynep Pacci, Yasemin Atılğan Şengül, Oya Alagöz, Rukset Attar, Aslı Uyar
Makale Türü: Arařtırma Makalesi

27

Güneş Enerjisinin Kısa-Dönem Tahmininde Farklı Makine Öğrenme Yöntemlerinin Karşılaştırılması
Comparison of different machine learning methods in short-term forecasting of solar energy
M. Fatih Bekçioğulları, Bünyamin Dikici, Hakan Açıkgöz, Ö. Fatih Keçecioglu

37

Türkiye'deki Yazılım Organizasyonlarının Mikroservis Tabanlı Mimaride Uyguladığı Analiz ve Tasarım Yöntemleri Üzerine Bir Arařtırma
A Survey on Analysis and Design Practices of Turkish Software Organizations for Microservice Based Architectures
Hüseyin Ünlü, Burak Bilgin, Onur Demirörs
Makale Türü: Arařtırma Makalesi

47

Otomotiv Endüstrisinde Zamanlama Mimarilerinin Değerlendirilmesi
Evaluation of Scheduling Architectures in Automotive Industry
Berkay Saydam, Tolga Ayav
Makale Türü: Arařtırma Makalesi

55

Klinik Araştırmalarda Geliştirilen Yapay Zekâ Modelleri için Yol Haritası

Roadmap for Artificial Intelligence Models Developed in Clinical Research

Alican Aşan¹, Duygu Sinanç Terzi²



¹Türkiye Cumhuriyeti Cumhurbaşkanlığı Dijital Dönüşüm Ofisi, Ankara, Türkiye

<https://orcid.org/0000-0002-9971-2678>, alican.asan@cbddo.gov.tr

²Amasya Üniversitesi, Mühendislik-Mimarlık Fakültesi, Bilgisayar Mühendisliği, Amasya, Türkiye
<https://orcid.org/0000-0002-3332-9414>, duygu.terzi@amasya.edu.tr

Öz

Yapay zekânın sağlık hizmetlerinin şeklini değiştirmeye başlamasıyla beraber gündeme gelen ve klinik araştırmalarda geliştirilen uygulamaların hayata geçirilmesinin önünde duran çeşitli zorluklar vardır. Bu zorluklar; veri seti hazırlamaktan önleme tekniklerine, model eğitiminden modelin çıktılarının yorumlanmasına kadar model geliştirme sürecinin neredeyse her fazında problem yaratmakta hatta hatalı ve yanlış modellerin geliştirilmesine sebep olmaktadır. Bu çalışmada, yapay zekânın potansiyelinin ortaya koyulmasını engelleyen zorlukların, yapay zekâ uzmanları ve sağlık çalışanları arasındaki iş birliği ile yani veri ve deneyimin birleştirilmesiyle mümkün olacağını vurgulayan bir yol haritası sunulmuştur. Bu yol haritasındaki her adımda karşılaşılan zorluklar ve önerilen çözümler, çeşitli sağlık uygulamaları ile örneklendirilmiştir. Bu yol haritasının, uygulayıcılara farklı bakış açıları sunması, geliştirilecek modellerin performansını iyileştirmesi ve gerçek dünya uygulamalarının artışında rol oynaması beklenmektedir.

Anahtar kelimeler: yapay zekâ, klinik araştırma, yol haritası

Abstract

There are various difficulties that stand in the way of the implementation of applications developed in clinical research, which came to the fore with artificial intelligence starting to change the shape of health services. These difficulties create problems in almost every phase of the model development process, from data set preparation to preprocessing techniques, from model training to interpretation of model outputs, and even lead to the development of faulty and biased models. In this study, a roadmap is presented emphasizing that the challenges that prevent the realization of the potential of artificial intelligence will be possible with the cooperation between artificial intelligence experts and healthcare professionals, that is, by combining data and experience. The

challenges and suggested solutions at each step in this roadmap are exemplified by various health practices. It is expected that this roadmap will offer different perspectives to practitioners, improve the performance of the models to be developed, and play a role in increasing real-world applications.

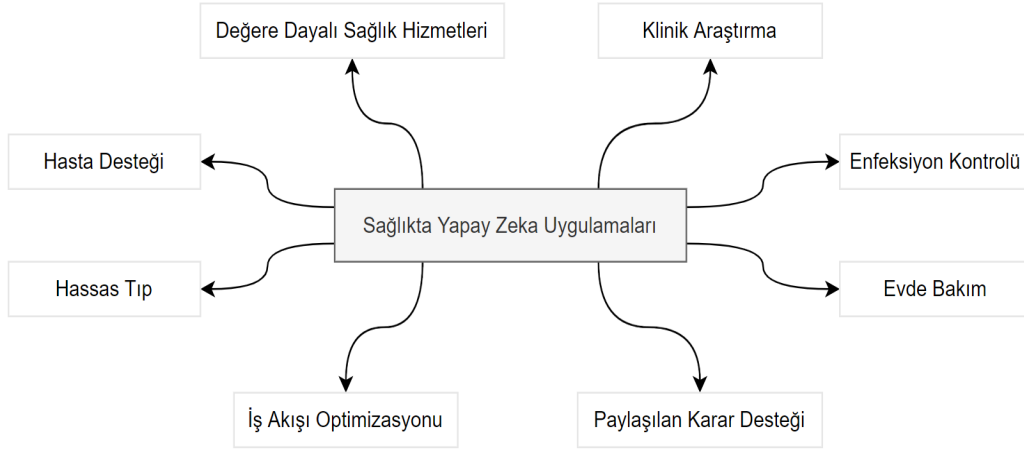
Key words: artificial intelligence, clinical research, road map

1. Giriş

Yapay zekâ; karmaşık makinelerde insan benzeri akıllı davranışı yapay olarak yaratmak ve simüle etmek için tasarlanmış bilgisayar bilimi dalıdır. Sağlık hizmetlerinde yapay zekâ ise teşhis, tedavi, tahmin ve önlem gibi amaçlar doğrultusunda sağlık verisindeki örüntüleri ortaya çıkarmayı hedefler [1]. Bu bağlamda yapay zekâ, hem hizmetlerin erişimini, kalitesini ve verimliliğini iyileştirmek hem de değere dayalı sağlık sistemleri oluşturmak için stratejik bir kaldıraç görevi taşımaktadır.

Sağlık verileri; hem hasta kayıtlarından, tanısal görüntülerden ve raporlardan gelen statik verileri hem de başucu monitörlerinden veya uzaktan hasta izleme cihazlarından gelen dinamik verileri kapsar ve doğası gereği büyük ölçüde yapılandırılmamıştır [2]. Bu tür karmaşık ve dinamik verileri işlemek, geleneksel analitik araçların kapasitesinin ötesine geçer. Bu verilerden anlamlı iç görü elde etmek ise, büyük veri analitiği ve yapay zekâ uygulamaları ile mümkündür.

Yapay zekâ uygulamalarından en çok yararlanan sağlık alanları Şekil 1'de görselleştirildiği üzere sekiz başlık altında ele alınmaktadır [3,4]. Değere dayalı sağlık hizmetleri; bakım yolları, tedaviler ve maliyetlerle ilgili verilerin analizi ile belirli performans ölçütlerini karşılayarak sağlık hizmetinin kalitesini ve etkililiğini iyileştirmek için sağlık hizmeti sağlayıcısına finansal teşvikler sunan bir modeldir. Hasta desteği; hastalar tarafından bildirilen sağlık sonuçlarının toplanmasına ek



Şekil 1. Sağlıkta yapay zekâ uygulamaları

olarak, hastaların sağlık okuryazarlığı düzeylerini güçlendirmek ve klinik araştırma çalışmalarına dâhil etmek için forum veya sohbet robotları gibi fırsatların oluşturulmasını ifade etmektedir. Hassas tıp; genetik verilerin hastalıklar, tedaviler ve sonuçlarla birlikte sistematik olarak toplanması ve analizi ile hastalara zarar vermektan kaçınarak en iyi tedavilerin seçilmesi veya oluşturulması sürecidir. İş akışı optimizasyonu; birden çok departmanı, personeli ve varlığı kapsayan dinamik hastane koşullarında üretkenliği artıran verimli kaynak kullanımı gereksinimlerini ifade etmektedir. Paylaşılan karar desteği; hastalara kendi sağlık verileri üzerinde kontrol ve iç gözü sağlayarak daha yüksek uyum oranları ve daha düşük ilaç maliyetleri elde etmeyi amaçlayan süreçtir. Evde bakım; akıllı ev teknolojilerini, giyilebilir cihazları, klinik verileri ve periyodik yaşamsal belirti ölçümlerini birleştirerek evde bakım veya ayakta tedavi için uzaktan destek sağlanmasıdır. Enfeksiyon Kontrolü; hastanede edinilen veya sağlık bakımı ile ilişkili olan ve yüksek oranda ölümlü sonuçlanan enfeksiyonun tahmini ve önlenmesiyle ilgili disiplindir. Son olarak, klinik araştırma; birçok farklı kaynaktan gelen büyük hacimli sağlık verilerindeki gizli örüntüleri ve ilişkileri keşfedecek modeller geliştirerek hastalıkları tahmin etmek, yeni biyobelirteçleri ortaya koymak ya da kişiselleştirilmiş ilaç üretmek gibi amaçların gerçekleştirilmesidir.

Klinik araştırmalar, sağlıkta yapay zekâ uygulamaları arasında oldukça büyük bir paya sahiptir. Panch ve arkadaşları, yapay zekâ teknolojilerinin, tıp uzmanlarını faaliyetlerinde destekleyeceği ve işlerini basitleştireceği görüşündedir [5]. Benzer şekilde, Jiang ve arkadaşlarına göre, yapay zekâ doktorların daha iyi klinik kararlar almasına ve hatta sağlık hizmetine özgü işlevsel alanlarda insan yargısının yerini almasına yardımcı olacaktır [6]. Çünkü, yapay zekâ ham verilerdeki anlamlı ilişkileri tanımlayabildiğinden, birçok tıbbi durumda teşhis, tedavi ve tahmin sonuçlarını destekleyebilir. Ek olarak, robotik teknolojileri ile güçlendirilen yapay zekâ, rehabilitasyon tedavisi ve cerrahide de fark yaratabilir.

Klinik araştırmalar için kullanılan verilerin, sağladıkları katma değeri en üst düzeye çıkarmak amacıyla çeşitli ilkelere uyumlu olmaları gerekmektedir. Bu ilkeler; bulunabilirlik, erişilebilirlik, birlikte çalışabilirlik ve yeniden kullanılabilirlik olarak özetlenebilir [7]. Bulunabilirlik; verinin kalıcı bir kaynaktan ve zengin meta verilerle açıklanmış olmasını,

erişilebilirlik; verinin standartlaştırılmış bir iletişim protokolü kullanılarak açık, ücretsiz ve evrensel olarak sunulmasını, birlikte çalışabilirlik; bilgi temsilinin nitelikli referanslar içeren resmi ve yalnız bir dille sunulmasını, yeniden kullanılabilirlik ise; verinin çok sayıda doğru ve alakalı özelliklerle zengin bir şekilde tanımlanmasını ve alanla ilgili standartları karşılamasını ifade etmektedir.

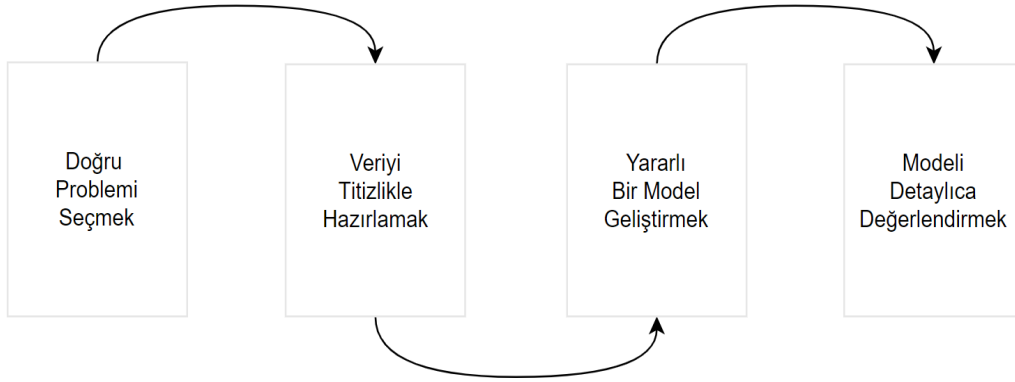
Bu ilkeler ve alan deneyimlerimiz doğrultusunda bu çalışmada, güvenli ve anlamlı ilerlemeyi hızlandırmaya yönelik bir bakış açısıyla, klinik araştırmalarda yeni yapay zekâ çözümleri tasarlarırken, test ederken ve uygularken dikkate alınması gereken kritik adımlar belirlenerek bir yol haritası olarak sunulmuştur. Bu yol haritası, başarının ancak yapay zekâ uzmanları ve sağlık çalışanları arasındaki iş birliği ile mümkün olacağını vurgulamaktadır.

2. Önerilen Yol Haritası

Yapay zekânın sağlık hizmetlerinde uygulanmasının, hasta yönetimi süreçlerini iyileştirme gibi açık faydalarının yanı sıra daha az sevk, maliyetleri düşürme ve zaman tasarrufu gibi potansiyel ikincil faydaları da bulunmaktadır. Bu faydaları ortaya çıkarmak veya daha belirgin kılmak için önerilen ve alan uzmanları, karar vericiler ve kullanıcılar dâhil olmak üzere disiplinler arası ekipler tarafından gerçekleştirilmesi beklenen klinik araştırmalarda geliştirilen yapay zekâ modelleri için yol haritası Şekil 2’de verildiği üzere dört temel adımdan oluşmaktadır.

2.1. Doğru problemi seçmek

Sağlık hizmetlerinin amacı daha kişisel, öngörücü, önleyici ve katılımcı olmaktır ve yapay zekâ bu yönlerde büyük katkılar sağlayabilir [8]. Doğru sorular sorulduğunda yapay zekâ, büyük verilerde gizli olan ve klinik kararlarda rolü olabilecek dikkate değer bilgileri ortaya çıkarma potansiyeline sahiptir. Ne yazık ki, sağlık için yapay zekâ uygulamalarının ilerlemesi, iyi tanımlanmış soruların olmaması ve açıklamalı veri kümelerinin eksikliği nedeniyle sınırlı kalmıştır [9]. Pek çok yapay zekâ araştırmacısı, problemlerin ve çözümlerinin klinik ilişkisini sorgulamadan, ek açıklamaların hazır olduğu sorulara odaklanmıştır. Verilerle ilgili yakın bilgi eksikliği çoğu zaman yanlış yönlendirilmiş problem formülasyonlarına ve



řekil 2. Klinik arařtırmalarda geliřtirilen yapay zekâ modelleri iin yol haritası

özmlerine yol aar. Bu sebeple, klinik faydayı gz nnde bulundurarak modelin kullanılacađı senaryoya gre dikkatli bir deęerlendirme yapılması gerekmektedir. Yeni projelere bařlarken, zmler geliřtirmeden nce ilgili paydařların katılımıyla bu tr sorunlardan kaınılabilir.

2.2. Veriyi titizlikle hazırlamak

Sorun formle edildikten ve veri kullanımına iliřkin izinler alındıktan sonra, sz konusu sorun ile uygunluđu aısından mevcut verinin detaylıca incelenmesi gerekmektedir. Bu srete dikkat edilmesi gereken drt unsur bulunmaktadır [10]. İlk unsur, belirli bir sonucu tahmin ederken, verilerin nasıl ve ne zaman toplandıđı ve verilerin hangi amaca hizmet ettiđinin belirlenmesidir. İkinci unsur, verilerin toplanma řekillerindeki farklılıkların anlaşılması ve hesaba katılmasıdır. nc unsur, eđitim verisinin modelin kullanılacađı ortamı kritik bir şekilde

temsil etmesi gerekliliđidir. Son unsur ise, model gvenilirliđini azaltan nyargıların giderilmesidir.

2.2.1. Aık veri ile gerek dnya verisi arasındaki farklılıklar

Sađlık alanında birok arařtırma aık veri setleri zerinden yapılmaktadır. Aık veriler zerinden uygulanan ve model karřılařtırması yapan alıřmalar ne kadar deđerli olsa da gerek dnyada kullanılacak bir modelin arařtırma ve geliřtirme ařamalarına ok benzememektedir. Dřnlenin aksine, rn olarak sunulması planlanan bir yapay zekâ modelinin arařtırma ve geliřtirme srecinin byk bir kısmını, veri problemleri ve bunlara retilen sonular oluřturmaktadır.

Burada dikkat edilmesi gereken ilk husus, literatrde gzlemlenen spesifik konu zerinden paylařılmıř aık ve hazır veri setleri ile gerek dnyada elde edilmiř veri setleri arasındaki farklılıkları ortaya koymak olacaktır. Arařtırmacıların kullandıkları verilerden anomalileri ayırmaları, ancak uzmanlık bilgisi elde edildiđinde gerekleřtirilebilmektedir. rnek olarak, BRATS 2018 [11] veri setinde kullanılan beyin manyetik rezonans (MR) grntleri bazı n iřlemelerden gemiř ve kafatası ıkarılmıř grntlerdir. Gerek hayatta ise ulařtıđımız verilerin hepsinin beyin olması bile mmkn olmayabilir. Arařtırmacı, elindeki veriyi istediđi hale getirmek iin alan bilgisini kullanarak nce ilgili organın ekimi olduđundan emin olmalı, aksenel/sađital/koronal ekimler arasındaki farkı incelemeli ve

seim yapmalı, bundan sonra ise T1/T2/flair gibi ekim tarzları zerinden bir analiz gerekleřtirmelidir.

Bu analizler sonucunda byk veri yığınmının ierisinden ilgili olan kısımların ıkarılması ile yapay zekâ modelinin daha dođru veri zerinden đrenmesi sađlanacaktır. Fakat bu ařamaya gelmek iin bile verinin teknik elementleri đrenilmeli, sađlık alıřanlarından gelen bilgi ile mhendislik bilgisi birleřtirilip, dođru seimin yapılması gerekmektedir.

Bir diđer konu ise, st verinin kullanılabilirliđidir. Grntleme cihazları veri ile birlikte kapsamlı bir st veri paylařmakta, bu paylařılan st veriden gerekli bilgiler alınabilmektedir. st veriyi okuma yeteneđi, alan bilgisinin artıřı ile geliřmektedir. st verinin kullanılması ise, verilerin temizlenmesi ařamasında byk avantaj sađlamaktadır.

Veriler temizlendikten sonra ise hangi tr verilerin hangi grevlerde ne řekilde kullanılacađı, doktorların alan bilgisi sayesinde daha gvenilir olarak gerekleřtirilmektedir. rneđin, eđer ekim aısı veya ila dozu deđiřiklikleri gibi konseptlerle oluřan farklı veri alt trleri, doktorlar tarafından farklı problemlerin saptanmasında kullanılıyorsa yapay zekâ modellerinin eđitiminde de bu bilgilerin gz nnde bulundurulması gerekmektedir.

2.2.2. Veri problemlerinin saptanması

Verideki anomaliler ayıklandıktan ve ilgili kısım seildikten sonra, verinin dođasından kaynaklanan problemlerin zlmesi gerekmektedir. Sađlık alanında tutulan herhangi bir veri insan faktr yznden hatalı tutulmuř olabilir. Bu hatalar; ekim hataları, teknisyen hataları, etiketleme hataları gibi farklı trden olabilir.

Hata olmasa da normalin dıřında ekilmiř bazı grntler de dikkate alınmalıdır. Bu gibi durumlarda yapılan ekim hatalarının veya teknik hataların yakalanabilmesi iin arařtırmacıların normal ekimleri yeterince anlamıř olması gerekmektedir. rnek olarak, rntgeni ekilmiř bir kiřinin zerindeki metallerden ameliyat geirmiř olduđu bilgisinin ıkarımı yapılabilir. MR makinesine giren bir hastanın, yařlılıđı veya rahatsızlıđı dolayısı ile kafasını daha rahat ve yan bir řekilde tutup MR'ı ekilebilmektedir. Bu tr problemler gz nne alındıđında, eđer gerek hayatta bu hastaların analizinin yapılması gerekmekte ise, retilen rnn de bu tarz verilerde alıřmasından emin olunmalıdır. Bunun iin ise eđitim veri setinin gerek veriyi kapsamlı olarak temsil

etmesi gerekmektedir. Hangi çekimlerin çekim hatası olduğu ve analizin yapılamayacağı ya da hangi çekimlerin gerçek hayatta karşılaşılabılır vakalar olduğu bilgisinin doktorlar tarafından paylaşılması, modelin eğitildiği veri setinin gerçek hayatta karşılaşılan vakaları temsil etme gücünü artıracak ve bu tür vakalarda başarılı olmasını sağlayacaktır.

Çekim hatası bulunan görüntülerde ise, modelin yanlış öğrenmemesi için tekrar bir eleme süreci gerçekleştirilmesi uygun olacaktır. Çekim hatasının görülmesi gereken kısımların görülmesini engellediği durumların eğitim veri setinde sıkça bulunması, öğrenme yeteneğini negatif etkilemektedir.

Bir diğer konu ise, teknisyenin üst veri girişinde yaptığı hataların, verilerin yanlış etiketlenmesiyle sonuçlanıp, modelin eğitim sürecindeki başarısını düşmesine yol açmasıdır. Bu problemler, doktorlar tarafından gelen alan bilgisi kullanılarak çözümlenmeli, modelin doğru ve geçerli veriler ile eğitildiğinden emin olunmalıdır.

Verideki kimi problemleri tanımlamak kolay olsa da bazı problemlerin ortaya çıkması gerçek hayatta uygulanmadığı sürece mümkün olmayabilir. Bu tür problemlerden en önemlisi, sağlık alanında kullanılan yapay zekâ tabanlı sistemlerin barındırdığı yanlışlıklar ve bu yanlışlıkların kaynaklarıdır. Sağlık alanında istatistiksel ve sosyal olmak üzere iki tür yanlışlık bulunmaktadır [12]. İstatistiksel yanlışlık; optimum altı örnekleme, değişkenlerde ölçüm hatası ve etkilerin heterojenliği gibi nedenlerle gerçek temel tahminden farklı bir sonuç üretilmesini ifade etmektedir. Sosyal yanlışlık ise; örtük veya açık olarak gerçekleştirilen ve belirli bir grup için sistematik olarak optimal altı sonuçlara yol açan eşitsizliği ifade etmektedir. Bu yanlışlıklar ne kadar ciddiyetle dikkate alınır, o kadar gürbüz modeller oluşturulabilecek ve böylece bu alanda yapılan çalışmaların genelleme kabiliyeti o kadar artacaktır.

Hem yanlışlıkları hem de hataları tespit edebilmek adına, gerçek hayat verisi üzerinde test edilen bir ürünün başarısız olduğu durumlar itina ile analiz edilmeli, doktorlar ve mühendisler bir araya gelerek hangi tür verilerde neden hata oluştuğu araştırılmalıdır. Örneğin; model eğitiminde orta yaş hastaların kullanılıp testin bebek verisiyle yapılmış olması ya da kadın hastalıkları için geliştirilecek bir modelde erkekler için verilerin bulunması gibi durumlar modelin karar mekanizmasını doğrudan etkileyecektir. Bu araştırmalar doğrultusunda, modelin daha gürbüz çalışması için, geriye dönülüp veri setinde veya eğitim sürecinde gerekli düzeltmelerin yapılması ya da ek doğrulama mekanizmalarının kurgulanması ve modelin tekrar eğitilmesi elzemdir [13].

Bunun yanında, eğitilen modelde kullanılan veri hacminin artmasının, yanlışlıkların etkisini azaltabileceği de göz önünde bulundurulmalıdır. Fakat gerçek dünyada modeli bütün yanlışlıklardan kurtaracak verilerin bulunması kolay değildir. Veri hacmi azaldıkça verideki yanlışlıkların, gürültülerin ve yapaylıkların eğitilen modele olan baskısı da artacaktır. Bu sebeple daha küçük gerçek dünya veri setleri üzerinde çalışırken araştırmacıların verideki yanlışlıklara proaktif bir şekilde hitap etmesi gerekmektedir. Bu da verinin çok ince detaylara kadar incelenmesi ve alan bilgisinin edinilmesi ile mümkündür.

Veri hacminin az olması, veri dağılımını daha sınırlı kalmaya zorlayacağı için, modelin gördüğü verilerin gerçek dünyada karşılaşılabılır veriler tarafından tam olarak temsil edilmemesine yol açabilecek ve modelin performansını büyük oranda etkileyecektir. Bu sebeple, verinin boyutu azaldıkça ön işleme tekniklerinin önemi de artmaktadır.

2.2.2. Veri Ön İşleme

Veri ön işleme genellikle, her biri farklı bir yapıyı düzelterek birkaç ayrı yöntemin belirli bir stratejide arka arkaya uygulanması ile gerçekleştirilmektedir. Nihai sonuçları değiştirebilme gücüne sahip bu süreçte; gürültü giderme, aykırı değer tespiti, özellik seçimi, ölçekleme ya da eksik değerlerle başa çıkma gibi yöntemler uygulanmaktadır [14].

Her tür veri, kendine özgü problemleri barındırdığı için bu problemlerin her birinin çözümü farklı stratejiler ile gerçekleştirilmektedir. Bu süreçte gerekli ön işleme türlerini ikiye ayırmak mümkündür. Birincisi, geleneksel ön işleme teknikleridir. Örnek olarak, doktorların bir MR görüntüsünü tersini almak gibi tümörü tespit etmelerini kolaylaştıran yöntemler verilebilir. Bu yöntemler hem gözle görülebilir bazı olguların daha kolay görülmesini sağlamakta hem de verinin doğasını kalıplaşmış bir şekilde araştırmacılara sunmaktadır. İkincisi ise derin öğrenme yöntemlerinin görme yeteneğinin spesifik veri üzerinde yaratabileceği çözümlerdir. Bunlar geleneksel ve hazır bir şekilde doktorlardan gelen olgular değil verideki gizli örüntülerdir. Bu sebeple, alan uzmanlığı sadece doktorlar tarafından değil aynı zamanda çekimi yapan teknisyenlerden de sağlanmaktadır. Örneğin, bir MR görüntüsünde beynin ve kafatasının dışındaki siyah alanlar, gözümüze ne kadar siyah görünse de bu alanlar aslında siyahı temsil eden sıfır değerine çok yakın değerlerde piksellerden oluşabilir. Veride topluca bulunan bir gürültü problemini ortaya koyan bu durum, verinin doğası ve derin öğrenmenin görme yeteneği sayesinde tespit edildiği için fark edilmesi daha zor ve çözülmesi daha kritik olan ön işleme türüdür. Farklı MR görüntüleri arasındaki çözünürlük ve oran farklılıklarının yarattığı problemlerin giderilmesi için, dolgu adı verilen ve belirli alanlara sıfır pikselleri konularak çözünürlük ve oran eşitlemeleri yapılan yöntem ile giderilmektedir. Dolgu yöntemi kullanıldığında ise, sıfıra yakın siyah gözükten pikseller yüzünden, belirli çözünürlükte ve belirli büyüklükte sıfırlardan oluşan örüntüler ortaya çıkaracaktır. Bu problemler saptanmadığı süreçte, geliştirilen modelin bu örüntüler üzerinden öznitelik çıkarıp yanlış bir sonuç vermesi kaçınılmaz olacaktır. Bu aşamada ise açıklanabilirlik haritalarının oluşturulması ve yorumlanması çözüm olarak kullanılabilir.

Dolgu yöntemine alternatif bir bakış açısı ise, dolgu kısmında sıfır değerli pikseller yerine görüntünün doğasından gelen gürültüye yakın bir değer ile dolgu yapılması veya görüntüdeki bütün sıfıra yakın değerlerin sıfıra çekileceği bir şekilde gürültü elemesi yapılması olabilir. Fakat bu çözüm de beraberinde farklı problemleri getirmektedir. Örneğin, bu yöntem MR görüntüsündeki beynin içindeki sıfıra yakın değerlerin değişmesine yol açarak görüntüde bozukluklar yaratmakta ve beynin içindeki yapıları değiştirmektedir. Akılcı bir çözüm olarak, beynin sınırlarını belirleyip onun dışında kalan bütün piksellerin sıfır yapılması düşünülebilir. Bu çözümün, MR görüntüleri arasında çekim cihazı veya çekim tarzı kaynaklı farklılıkların bulunması durumunda,

istenilen sonucu üretemeyeceği göz önünde bulundurulmalıdır. Bu gibi karmaşık durumlarda, derin öğrenme modelinin problemlerini farklı derin öğrenme modelleri ile gidermek mümkündür. Tekrar beyin MR'ı üzerinden örneklendirecek olursak; beyinin kısımlarını ortaya koyan ayrı bir segmentasyon modelinin oluşturulması hem başarılı hem de genellenebilir bir gelişmiş ön işleme metodu olarak kullanılabilir. Böylece analizin yapılacağı kısmın dışındaki kısımlar analize dâhil edilmeyecek, bu kısımlardan oluşabilecek yanlışlıkların önüne geçilebilecektir.

Spesifik veri problemleri ise ancak alan uzmanlığının kullanılmasıyla tespit edilebilmekte, bu problemlere çözümler ise tekrar alan uzmanlıkları kullanılarak belirlenmekte ve gerekirse yeni bir yapay zekâ modeli kullanılarak çözülmektedir.

2.3. Yararlı bir model geliştirmek

Sağlık sistemlerinin etkili yönetimi, özünde bilgi işleme görevlerinin bir kafesidir ve iki temel bilgi işleme görevini içerir [15]. Birincisi; vakaların öykü, muayene ve tetkiklere göre sınıflandırılması olan tarama ve teşhis, ikincisi ise; gelecekteki bir sonuca ulaşmak için çok aşamalı bir sürecin planlanmasını, uygulanmasını ve izlenmesini içeren tedavi ve izlemedir.

Sağlık sistemi yönetimi ve bakım sağlama alanlarında bu süreçlerin temel biçimi, hipotez üretmeyi, hipotez testini ve eylemi içerir. Yapay zekâ, verilerdeki önceden gizlenmiş eğilimleri ortaya çıkararak bir sağlık sistemi içindeki hipotez oluşturma ve hipotez testi görevlerini geliştirme potansiyeline sahiptir ve bu nedenle hem bireysel hasta hem de sistem düzeyinde önemli etki potansiyeline sahiptir. Yapay zekâ, verilerin dağılımı ile ilgili ön varsayımlara dayanmayan ve verilerde hipotezleri ve hipotez testlerini formüle etmek için kullanılabilir modelleri bulabilen yöntemleri kullanarak mevcut istatistiksel teknikleri genişletir [16]. Bu nedenle, yapay zekâ modellerinin yorumlanması daha zor olsa da çok daha fazla değişkeni modellemeye dâhil ederek genelleştirme sağlar ve karmaşık durumlar için sonuçlar üretebilirler.

Bu başarı potansiyelinin açığa çıkarılmasının önündeki en önemli adım, veriyi anlamak ve ön işleme yapmaktır. Bu süreç verimli geçirildikten sonra, yararlı modellere ulaşmak için gerekli olan, modelleme sürecinde farklı yöntemlerin ve parametrelerin kullanılıp, çeşitli başarı ölçümlerinde sağladıkları performanslara göre karşılaştırmalı olarak değerlendirilerek en ideal seçeneğin belirlenmesidir. Geliştirilen modelin parametrelerinin ve yapısının değişikliği, modelin faydası üzerinde doğrudan etkiye sahiptir. Örneğin, MR üzerinde tümör tespiti probleminde seçilen filtre boyutu; aranan tümörün boyutu ve şekli gibi özelliklere göre ayarlandığında modelin başarısı artacaktır.

En ideal çözümün belirlenmesi, modelin tasarımında alan bilgisinin yol gösterici olarak kullanılmasıyla da sağlanabilir. Örnek olarak bir MR görüntüsünü inceleyen radyolog, belirli bilişsel süreçlerden geçerek bu MR görüntüsünün analizini gerçekleştirmektedir. Radyoloğun bilincinde yarattığı bu senaryo ile, spesifik çekimlerin spesifik bölgeleri spesifik durumlarda önem kazanmakta, farklı girdiler farklı şekillerde kullanılarak teşhis yapılmaktadır. Bu alan bilgisinin kullanılması ve derin öğrenme modelinin bu gerçek senaryoya en yakın bir şekilde tasarlanması modelin gerçek dünya

başarısını büyük oranda artırmaktadır. Dikkat mekanizmaları, çoklu girişler veya modellerin birleştirilmesi gibi teknikler kullanılarak gerçek hayatta doktorların karar verme süreçlerini taklit edecek sistemlerin tasarlanması mümkün olacaktır. Benzer şekilde, alan uzmanlarının, araştırma sürecinde yapay zekâ modellerinin çalışma şeklini daha iyi öğrenmesi ile eğitim veya ürün kısmına sağlayacakları katkılar da artacaktır. İki farklı alanın birleşmesiyle gelişen çıktılar, iki dünyanın toplamından daha fazla olacaktır.

Ek olarak, modelin çıkardığı öznitelikler kullanılarak müfredat öğrenmesi gibi yöntemler ile daha yüksek performans elde edilebilir. Müfredat öğrenmesi, modelin önce daha kolay verilerden öğrenerek, eğitildiği verilerin giderek zorlaştığı bir çerçeveye içerisinde modelin başarısını arttırmayı amaçlamaktadır [17]. Müfredat öğrenmesinin gerçekleştirilebilmesi için verilerin zorluk açısından derecelendirilmesi de alan uzmanları tarafından yapılmalıdır. Bu alan bilgisi kullanılarak modelin aynı veriler üzerinden daha başarılı sonuçlar elde etmesi sağlanmaktadır.

2.4. Modeli detaylıca değerlendirmek

Model değerlendirme, bir sistemin ürettiği sonuçların kalitesinin ölçüldüğü süreçtir. Model değerlendirme performans ölçütleri; modelin ne kadar iyi performans gösterdiğini, modelin yetersiz ya da aşırı öğrenmiş olma durumunu, modelin ürüne dönüştürülecek kadar iyi olup olmadığını ve daha farklı bir eğitim setinin model performansını iyileştirme potansiyelini ortaya koymada yardımcı olmaktadır.

Yapay zekâ sistemlerinin ve bileşenlerinin değerlendirilmesi, görev odaklı ve yetenek odaklı olmak üzere iki şekilde gerçekleştirilmektedir [18]. Görev odaklı değerlendirme; görevin kesin bir tanımı verildiğinde ondan bir performans kavramının üretilmesiyle gerçekleştirilir ve insan ayrıştırması, ölçütler ya da eş karşılaştırma gibi yollar kullanılır. Yetenek odaklı değerlendirme ise; muhakeme becerileri, tümevarımlı öğrenme yetenekleri, sözlü yetenekler ya da hareket yetenekleri olan ve görev odaklı değerlendirmeye uygun olmayan modellerin, bir dizi bilgi işleme testine verdiği yanıtların hız, kalite ya da psikometri açısından incelenmesidir.

Sağlık uygulamalarında yapılan işin kritikliği, bu modellerin çalışma şekillerinin ve başarılarının önemini artırmaktadır. Bu sebeple modelin başarısı farklı metriklerle ölçülmeli, yapılan hatalar analiz edilmeli, hataların neden ortaya çıktığı saptanmalı ve hatalar arasındaki benzerlikler incelenmelidir. Bu saptamalar yapıldıktan sonra veri toplama, veri hazırlama veya model eğitime adımlarına geri dönülerek bu problemlerin ortadan kaldırılması ya da azaltılması ile ilgili aksiyonlar alınabilir. Örneğin, belirli bir yaş üzerindeki hastalar yapılan yanlışların büyük bir kısmını oluşturmaktaysa; bu yaş grubundaki hastalardan alınan verilerin sayısı artırılabilir, bu yaş grubundaki hastalarda daha az hata yapacak modeller geliştirilebilir veya bu yaş grubundaki hastalardaki hatanın neden kaynaklandığı saptanarak bu problem ön işleme yöntemleri ile çözülebilir. Ayrıca, belirli bir test noktasında bir modelin ne kadar güvenilir olduğuna dair ölçümler elde etme yöntemleri, daha büyük belirsizlik bölgelerinde nitel değerlendirmeye rehberlik etmek için de kullanılabilir.

Göz önünde bulundurulması gereken bir diğer konu ise doğrulama, model ayarlama ve bağımsız test için kullanılan veri kümeleri arasında veri sızıntısı meydana gelmediğinden emin olunmasıdır. Örneğin, hasta başına çok sayıda röntgen görüntüsünün olduğu veri setlerinin hasta düzeyinde bölünmesi oldukça önemlidir, böylece hem eğitim hem de test setlerinde aynı hastadan hiçbir görüntü bulunmamış olacaktır.

Model değerlendirme aşamasında, modellerin açıklanabilir olması en büyük araçlardan biridir. Yapay zekâ açıklanabilirliği; yanlışlıkları tespit ederek karar vermede tarafsızlık sağlar, kararı değiştirebilecek potansiyel zayıflıkları vurgulayarak sağlıklı sağlar, anlamlı değişkenlerin çıktıya etkisini ortaya koyarak nedensellik sağlar [19]. Bu sayede, açıklanabilirlik haritalarından yapılan analizler, alan bilgisi ile birleştirildiğinde modellerin gürbüzlüğünü artıracak gelişmeleri mümkün kılar.

Yapay zekânın klinik pratikte oluşturduğu potansiyelin açığa çıkarılabilmesi için, çalışmaların aynı zamanda şeffaf şekilde yürütülmesi gerekmektedir. Şeffaflık; kullanılan verilerin kaynağının ve kalitesinin yanı sıra performans bileşenleri ile de ilişkilidir [20]. Şeffaflık düzeyini netleştirmek yapay zekâ uzmanları, klinik kullanıcılar ve düzenleyiciler için büyük önem arz etmesine rağmen, gereksinimlere, risk düzeyine ve kullanım amacına göre farklılık gösterir. Bu bağlamda şeffaflığın sağlanması ve düzeyinin belirlenmesi, yapay zekâ modelinin güvenilirliğini artıracaktır.

Son teknoloji yapay zekâ uygulamaları, klinik iş akışlarına entegre edilmedikçe tam potansiyellerine ulaşamayacaklardır [21]. Bu nedenle entegrasyon sırasında ve sonrasında ortaya çıkan gereksinimler doğrultusunda da model değerlendirme sürecinin dinamik olarak devam etmesi gerekmektedir.

3. Değerlendirme ve Sonuç

Sağlık uygulamaları yapay zekâ algoritmalarının ve araçlarının güvenli ve etkili kullanımı için kritik bir kavşak noktasındadır. Sağlık hizmetlerini dönüştürmek için bu araçlardan yararlanacak teknik kapasite mevcut olmasına rağmen süreç çeşitli zorlukları barındırmaktadır. Bu zorluklar: gerçekçi olmayan beklentiler, önyargılı ve temsili olmayan veriler, eksik önceliklendirmeler, belirsiz düzenleyici ortamlar ve değerlendirme sürecindeki yetersizlikler olarak özetlenebilir. Bu ve benzeri zorluklar ise: yanlışlığa, yanıltıcı tahminlere, yanlış kararlara ve hatta ayrımcılığa neden olabilir.

Potansiyel zorlukları en aza indirmek ve hatta ortadan kaldırmak, yapay zekâ çalışmalarını yürütmek için gereken zamana, uzmanlığa ve kaynaklara yatırım yapmakla ve iş birliklerini güçlendirmekle mümkün olacaktır. Klinisyenlerin görev tanımlarına; bilgi entegratörlüğü ve tercümanlık gibi yeni roller eklenmesi gerekecektir. Bununla birlikte yapay zekâ uygulamasını kimin kontrol edeceği, onaylayacağı veya bundan faydalanacağı hâlâ üzerinde tartışılması gereken konulardır.

Bu çalışmada; klinik araştırmalar için yapay zekâ modelleri geliştirme sürecinde karşılaşılabilecek muhtemel zorlukları ortaya koyan ve yapay zekâ uzmanları ile sağlık çalışanları arasındaki iş birliği ile çözüm önerilerinde bulunan bir yol haritası sunulmuştur. Doğru problemi seçmek, veriyi titizlikle hazırlamak, yararlı bir model geliştirmek ve modeli detaylıca değerlendirmek olmak üzere dört adımdan oluşan bu yol

haritasının uygulayıcılara farklı bakış açıları sunması, geliştirilecek modellerin performansını iyileştirmesi ve gerçek dünya uygulamalarının artışıyla rol oynaması beklenmektedir.

Bu yol haritası, sağlık uygulamalarındaki girişimlere kritik katkılar sağlamanın yanında, geniş bir kitleye yöneliktir ve farklı uygulama alanları için de kullanılabilir. Yapay zekâ hangi uygulama alanında olursa olsun, tüm aktörleri ile ortaya çıkan zorlukların üstesinden gelinip doğru kullanımına elverişli bir ekosistem inşa edildiğinde, sektörü geliştirmek için büyük fırsatlar ortaya koyacaktır.

Kaynaklar

- [1] Salazar-Reyna, R., Gonzalez-Aleu, F., Granda-Gutierrez, E. M., Diaz-Ramirez, J., Garza-Reyes, J. A., Kumar, A. "A systematic literature review of data science, data analytics and machine learning applied to healthcare engineering systems", *Management Decision*, 1-20, 2020.
- [2] Mehta, N., Pandit, A., Shukla, S. "Transforming healthcare with big data analytics and artificial intelligence: A systematic mapping study", *Journal of Biomedical Informatics*, 100, 1-14, 2019.
- [3] Abedjan, Z., Boujemaa, N., Campbell, S., Casla, P., Chatterjea, S., Consoli, S., Costa-Soria, C., Czech, P., Despenic, M., Garattini, C. "Data Science in Healthcare: Benefits, Challenges and Opportunities". In: Consoli S., Reforgiato Recupero D., Petković M. *Data Science for Healthcare*. Springer, 3-38, 2019.
- [4] Noorbakhsh-Sabet, N., Zand, R., Zhang, Y., Abedi, V. "Artificial intelligence transforms the future of health care", *The American Journal of Medicine*, 132(7), 795-801, 2019.
- [5] Panch, T., Szolovits, P., Atun, R. "Artificial intelligence, machine learning and health systems", *Journal of Global Health*, 8(2), 1-8, 2018.
- [6] Jiang, F., Jiang, Y., Zhi, H., Dong, Y., Li, H., Ma, S., Wang, Y., Dong, Q., Shen, H., Wang, Y. "Artificial intelligence in healthcare: past, present and future", *Stroke and Vascular Neurology*, 2(4), 2017.
- [7] Wilkinson, M. D., Dumontier, M., Aalbersberg, I. J., Appleton, G., Axton, M., Baak, A., Blomberg, N., Boiten, J.W., Santos, L.B., Bourne, P. E. "The FAIR Guiding Principles for scientific data management and stewardship", *Scientific Data*, 3(1), 1-9, 2016.
- [8] Rong, G., Mendez, A., Assi, E. B., Zhao, B., Sawan, M. "Artificial intelligence in healthcare: review and prediction case studies", *Engineering*, 6(3), 291-301, 2020.
- [9] Ghassemi, M., Naumann, T., Schulam, P., Beam, A. L., Chen, I. Y., Ranganath, R. "Practical guidance on artificial intelligence for health-care data", *The Lancet Digital Health*, 1(4), e157-e159, 2019.
- [10] Maddox, T. M., Rumsfeld, J. S., Payne, P. R. "Questions for artificial intelligence in health care". *JAMA*, 321(1), 31-32, 2019.
- [11] Menze, B. H., Jakab, A., Bauer, S., Kalpathy-Cramer, J., Farahani, K., Kirby, Burren, Y., Porz, N., Slotboom, J., Wiest, R., Lanczi, L. "The multimodal brain tumor image segmentation benchmark (BRATS)", *IEEE Transactions on Medical Imaging*, 34(10), 1993-2024, 2014.

- [12] Parikh, R. B., Teeple, S., Navathe, A. S. “Addressing bias in artificial intelligence in health care”, *JAMA*, 322(24), 2377-2378, 2019.
- [13] Althubaiti, A. “Information bias in health research: definition, pitfalls, and adjustment methods”, *Journal of Multidisciplinary Healthcare*, 9, 211-2017, 2016.
- [14] Engel, J., Gerretzen, J., Szymańska, E., Jansen, J. J., Downey, G., Blanchet, L., Buydens, L. M. “Breaking with trends in pre-processing?”, *TrAC Trends in Analytical Chemistry*, 50, 96-106, 2013.
- [15] Secinaro, S., Calandra, D., Secinaro, A., Muthurangu, V., Biancone, P. “The role of artificial intelligence in healthcare: a structured literature review”, *BMC Medical Informatics and Decision Making*, 21(1), 1-23, 2021.
- [16] Beam, A. L., Kohane, I. S. “Big data and machine learning in health care”, *JAMA*, 319(13), 1317-1318, 2018.
- [17] Hacothen, G., Weinshall, D. “On the power of curriculum learning in training deep networks”, arXiv preprint arXiv:1904.03626, 2019.
- [18] Hernández-Orallo, J. “Evaluation in artificial intelligence: from task-oriented to ability-oriented measurement”, *Artificial Intelligence Review*, 48(3), 397-447, 2017.
- [19] Adadi, A., Berrada, M. “Peeking inside the black-box: A survey on Explainable Artificial Intelligence (XAI)”, *IEEE Access*, 6, 52138-52160, 2018.
- [20] Racine, E., Boehlen, W., Sample, M. “Healthcare uses of artificial intelligence: Challenges and opportunities for growth”, *Healthcare Management Forum*, 32(5), 272-275, 2019.
- [21] Yu, K. H., Beam, A. L., Kohane, I. S. “Artificial intelligence in healthcare”, *Nature Biomedical Engineering*, 2(10), 719-731, 2018.

Özgeçmişler

Alican Aşan, İstanbul Teknik Üniversitesi Kontrol ve Otomasyon Mühendisliği Bölümü'nden lisans derecesini almıştır. Türkiye Cumhuriyeti Cumhurbaşkanlığı Dijital Dönüşüm Ofisi Büyük Veri ve Yapay Zekâ Uygulamaları Birimi'nde araştırmacı olarak çalışmaktadır. Türk Beyin Projesi ve pek çok sağlık alanında yapay zekâ projelerinde görev almıştır. Çalışma alanları bilgisayarlı görü, yapay zekâ ve veri bilimi üzerinedir.

Duygu Sinanç Terzi, Selçuk Üniversitesi Bilgisayar Mühendisliği Bölümü'nden lisans derecesini, Gazi Üniversitesi Fen Bilimleri Enstitüsü Bilgisayar Mühendisliği Anabilim Dalı'ndan yüksek lisans ve doktora derecelerini almıştır. Amasya Üniversitesi Bilgisayar Mühendisliği Bölümü'nde araştırma görevlisi olarak çalışmaktadır. Çalışma alanları veri bilimi, büyük veri ve bilgi güvenliği üzerinedir.

Yazılım Mühendisliğinde Çevik Eğitim Modeli (ÇEM)

Agile Education Model in Software Engineering

Erkan Sarıkaya¹, Selami Bağriyanik², Mesut Gökalp³



¹Endüstri Mühendisliği Bölümü, Mühendislik Fakültesi

Özyeğin Üniversitesi

erkan.sarikaya@ozyegin.edu.tr

²Yazılım Mühendisliği Bölümü, Mühendislik Mimarlık Fakültesi

Nişantaşı Üniversitesi

selami.bagriyanik@nisantasi.edu.tr

³Çevik Ofis Bölümü

Eureko Sigorta

mesut.gokalp@eurekosigorta.com.tr

Özet

2020 yılında yaşanan covid-19 pandemisi gibi çevre kaynaklı büyük olayların etkisi ile teknolojinin önemi daha da artmaktadır. Üniversitelerin geleneksel eğitim yaklaşımları ve endüstrilerdeki hızlı dijital dönüşümün bir sonucu olarak Bilgisayar Bilimleri Mühendisliği, Yazılım Mühendisliği, Bilgisayar Mühendisliği ve diğer ilgili bölümlerden mezun olanların yetkinlikleri ile endüstri beklentileri arasında bir yetenek açığı olduğu gözlenmektedir. Bu çalışmada, özellikle son 15 sene içinde dünyada büyük bir yaygınlık kazanan çevik yöntemlerin, Yazılım Mühendisliği pratiklerinde sağladığı kazanımların bir eğitimi yöntemi olarak kullanıldığında da benzer fayda potansiyeline sahip olup olmadığı irdelenmiştir. Çalışmada, yeni bir Çevik Eğitim Modeli (ÇEM) önerilmiş ve modelin uygulandığı bir lisans dersi ile ilgili sonuçlar paylaşılmıştır. Türkiye’de bir üniversitede, Özyeğin Üniversitesi mühendislik fakültesinde gerçekleşen “Çevik Yazılım ve Ürün Geliştirme” dersi ile ilgili eğitim tasarımı, öğrenme performansları ve sonuçları paylaşılmıştır. Çalışmanın, üniversitelere, öğretim elemanlarına, çevik koçlara, öğrencilere ve Bilişim ve İletişim Teknolojileri sektöründeki profesyonellere faydalı bilgiler sağlayacağı düşünülmektedir.

Anahtar kelimeler: çevik, SCRUM, eğitimde çeviklik, yazılım mühendisliği eğitimi, öğrenme, takım performansı

Abstract

Considering the effect of major environmental events such as the covid-19 pandemic in 2020, technology has become even more influential. While the aforementioned phenomenon is being experienced, a talent gap is observed between the competencies of the graduates of Computer Science Engineering, Software Engineering and other related departments and industry expectations as a result of the traditional education approaches of the universities. In this study, it has been examined whether the agile methods, which have become widespread within software industry in the last

15 years, have similar potential when used as a training method. In this regard, a novel Agile Education Model is proposed and the results of its application have been presented. The study took place in the faculty of engineering at a university, the Ozyegin University in Turkey in “Agile Software and Product Development” course. The study is considered to provide useful information for universities, faculty members, agile coaches, students, and professionals in the Information and Communication Technologies industry.

Keywords: agile, SCRUM, agility in education, software engineering education, self-learning, team performance

1. Giriş

Son yıllarda çevik çerçeveler hızla yaygınlaşmakta ve yazılım geliştirme endüstrisinde neredeyse standart olarak uygulanmaktadır [1]. İş hayatındaki bu hızlı gelişmeler, çeviklik ile ilgili yöntem ve pratiklerin endüstride uygulanması için bilgi ve deneyime ihtiyacı arttırmıştır. Genelde üniversitelerde Çevik yazılım geliştirme süreçlerinin yeterince öğretilmediği tespit edilmiş, buna göre bilgi ve deneyimin artması için gerekli çalışmalar gerçekleştirmediği için de endüstri ihtiyaçları ile akademi dünyası arasındaki açık giderek büyümektedir [2]. Buna bağlı olarak çeviklik, manifesto, çeviklik prensipleri, farklı çevik çerçeveler, uygulama alanları, roller, çıktılar, etkinlikler gibi çeviklik bilgilerinin verildiği eğitimlere çok ihtiyaç duyulmaktadır [3]. Bu yeni yaklaşımın, kendi kendine öğrenmeyi destekleyerek sürekli gelişim ve geleceğe hazırlığı kolaylaştıracağı, buna bağlı olarak akademi ile endüstri arasındaki açığın giderilmesinde katkı sağlayacağı öngörülmektedir.

Özyeğin Üniversitesi’nde, Endüstri, Yazılım, İnşaat ve Elektrik/Elektronik mühendisliği öğrencilerinin katılımı ile gerçekleşen, “Agile Software and Product Development” dersinde edinilen tecrübeler ve sonuçları ele alınmıştır [4]. Bu çalışmada farklı olarak “Çevik Eğitim Modeli (ÇEM)” önerilmiş ve tasarımı genişletilmiştir. Son 2 senedir hem yüz yüze hem de pandemi dönemi ile birlikte tamamen uzaktan çevrimiçi olarak bu yaklaşım uygulanmaya devam edilmiştir.

Değerler ve prensipler etrafında, tüm öğrenciler ile takımlar oluşturarak gerçekleşen çalışmalar, deneysel yaklaşım, sonuçların gelişmesini ve pratiklerin sağladığı faydaların doğrulanmasını sağlayarak model geliştirilmiştir. Sınıf anlaşmasında kullanılan değerler, çevik eğitim süreci etkinlikleri, detaylandırılmış ve öğrenmeyi kolaylaştıran aktiviteler ile çalışmanın doğrulanması için zenginleştirilmiştir. Uygulanan farklı öğrenme teknikleri, öğrencilerin çevik yöntemleri deneyimlemesine ve uygulamasına olanak sağlayacak şekilde tasarlanan eğitim akışı, esnek, adaptif bir ders içeriği üzerinden işlenerek ders başarımlarının puanlanması ile tamamlanmış, edinilen deneyimler paylaşılmış ve gözlenen bulgular tartışılmıştır.

Üniversite lisans seviyesindeki bu dersin hedefleri aşağıdaki gibi belirlenmiştir;

1. Yazılım geliştirme yaşam döngüsü (SDLC: Software Development Life Cycle) hakkında temel bir anlayış sağlamak
2. Çevik çalışma yönteminin ürün geliştirmeye farklı bir perspektif olarak anlaşılmasını sağlamak
3. Motivasyon ve öğrenmenin takım boyutunda incelenmesi ile birlikte performansla farklı bir bakış açısı getirmek

Dünyada son 15 yıl içerisinde, çevik pratiklerle beslenen ders işleme, öğretme yaklaşım çalışmalarına rastlanmıştır. Bir sonraki bölümde paylaşılacak olan literatürden görüleceği gibi farklı pratikler ve yaklaşımlar uygulanmıştır. Literatürdeki bu yayınlardan farklı olarak bu çalışmada, kendi kendine organize olan öğrenciler tarafından seçilen projelerle takımların oluşturulduğu, bireysel çalışma ile takım çalışması arasındaki faydaların ve zorlukların nasıl ortaya çıktığı, belirlenen hedef gerçekleştirilirken yaşanan aksaklıklar ve bunların nasıl çözümlendiği, takım performansı ve ders notu ilişkisi, farklı performans ve ders notu belirleme yaklaşımları, genel olarak not yerine sürekli öğrenme isteği arasındaki memnuniyet farkı, tek yönlü bilgi aktarma yerine grup aktiviteleri ve oyunlaştırma ile öğrenmenin kalıcı hale getirilmesi, sürekli geri bildirim alınarak ders müfredatının dinamik olarak değiştirilmesi ve bu sayede ihtiyacın karşılanarak motivasyonla nasıl bir ilişki içinde olduğu irdelenmiştir.

Bu çalışmada, daha önceki çalışmanın [4] devamı niteliğinde ve bir bütün olarak 4 aşamalı Çevik Eğitim Modeli (ÇEM) tasarlanmıştır. Süreç detaylandırılarak Özyeğin Üniversitesi'nde toplamda 231 öğrenci ile oluşturulan 28 çevik proje takımının tecrübeleri dikkate alınarak model validasyonu yapılmıştır. Önceki çalışmada sadece bir dönemdeki 8 takım için yapılan örnek çalışmalar, sonuçları ile birlikte paylaşılmıştır. Benzer yaklaşımları hem endüstri içinde hem de üniversite ortamında uygulamak isteyen tüm eğitimciler, çalışmanın sonuçlarından faydalanabileceklerdir.

2. İlgili Çalışmalar

Tarihsel gelişimine bakıldığında yazılım geliştirme projelerinde farklı yaklaşımlar 1986 yılında ilk kez gündeme geldikten sonra, Scrum [5] ve Extreme Programming (XP) [6] çerçevelerinin 1993 ve 1999 da tanımlandıkları görülmektedir. 2001 yılında üniversite ve endüstriden araştırmacı ve pratisyen 17 kişi tarafından Çevik Manifesto [3] yayınlanmış ve 2011 'de Scrum Kılavuzu yazılarak kullanımı giderek yaygınlaşmaya başlamıştır [5]. Çevik çalışma yöntemlerinin tarihsesi TABLO 1'de gösterilmiştir [1, 5, 6, 7, 8].

Tablo 1: Çevik Yöntemlerin Tarihsesi

Sıra	Yıl	Çerçeve	Kaynak	Başlık
1	1986		[1]	The New New Product Development Game.
2	1993	SCRUM	[5]	Scrum Guide
3	1999	XP	[6]	eX-treme Programming Explained: Embrace Change
4	2001	AGILE	[3]	Manifesto for Agile Software Development
5	2011	SCRUM	[7]	The Scrum Guide
6	2015	eduScrum	[8]	The eduScrum Guide (No. 1.2) (p. 21).

Eğitimde ilk kez Çevik yaklaşımları üzerine Akademik araştırmalar 1990'ların ortasında Harvard Üniversitesinde başlarken üniversitelerde uygulamalar Avrupa, Kanada ve orta Asya'da 2002 yıllarında başlamıştır [9]. Yazılım mühendisliği bölümlerinde ilk çeviklik uygulamaları TABLO 2'de gösterilmiştir. 2003 yılında ilk kez XP denemeleri görülürken [10], 2005'te yazılım mühendisliğinde [11], 2010'da da K-12 eğitiminde çeviklik süreç modelleri dersleri işletilmiştir [12]. 2010 yıllarında ilk Scrum denemeleri eğitimde uygulanmaya başlanmış [13], ilerleyen yıllarda Test Güdümlü Geliştirme (TDD: Test Driven Development) [14], KANBAN [15], LEGO blokları ile Scrum gibi farklı çerçeveler uygulanırken [16], ilk kez 2015 yılında "eduScrum" adında eğitimde "Scrum Guide" yayınlanmıştır [8]. 2018 yılında da çevrim içi olarak çeviklik öğretimi uygulanmıştır [17].

Tablo 2: Yazılım Mühendisliği Eğitimlerinde İlk Çeviklik Uygulamaları

Sıra	Yıl	Yöntem	Kaynak	Ana Konu	Kitle
1	2003	XP	[10]	Öğrenilen Dersler	Yazılım Müh.
2	2005	Agile	[11]	Çevik Süreç Modeli	Yazılım Müh.
3	2010	Agile	[12]	Çevik Süreç Modeli	K-12 (İlk ve Orta Öğretim)
4	2010	Scrum	[13]	Scrum, ile öğrenme	Yazılım Müh.
5	2013	TDD	[14]	Dojo puanlama ile Çeviklik	Yazılım Müh.
6	2014	KANBAN	[15]	Kanban ile Öğrenme	Yazılım Müh.
7	2014	Scrum	[16]	LEGO Blokları kullanarak Scrum öğretimi	Yazılım Müh.
8	2015	eduScrum	[8]	"eduScrum" Kılavuzu (No. 1.2)	Yazılım Müh.
9	2018	Agile	[17]	Çevik Yöntemleri Öğrenme	"Online" Proje Yön..

3. Çevik Eğitim Modeli (ÇEM)

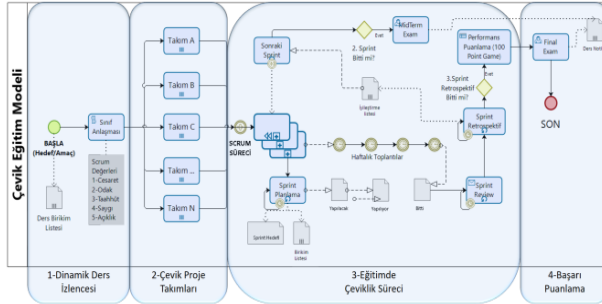
Çevik Eğitim Modeli öğrencilerin öğrenme etkinliğini yaratıcı bir şekilde arttırmalarını amaçlayan, kendi kendilerine yönetme ortamı içerisinde takım halinde yüksek değerli çıktılar üretmelerine olanak sağlayan bir eğitim modelidir. Özellikle ürün geliştirme amaçlı yazılım projelerinde kullanılan Scrum çerçevesinin üniversite öğrencileri tarafından etkili bir şekilde öğrenilebilmesi için bir eğitim içeriği ve uygulaması tasarlanmıştır. Bununla birlikte dönem boyunca öğrencilerin çevik yazılım ve ürün geliştirme, merak etme ve sorumluluk alarak hedefe ulaşma başarısı izlenmiştir. “Çevik Eğitim Modeli (ÇEM)”, çevik prensip ve değerleri çerçevesinde uygulanan iyi pratiklerle zenginleştirilerek aşağıdaki konu başlıklarında diğer “eğitimde çeviklik” uygulamalarından ayrılmaktadır.

- Ders programının dinamik olarak gerektiğinde güncellenmesi
- Tamamen öğrenciler tarafından seçilen projeler
- Kendi kendine organize olacak şekilde, Ürün Sahibi ve Scrum Master seçilerek oluşturulan takımlar
- Takımların birbirini ve takım içi bireyleri değerlendirecek şekilde not değerlemesi

“Çevik Eğitim Modeli (ÇEM)” Şekil 1’de gösterildiği gibi 4 unsurdan oluşmaktadır;

- 1- Dinamik Ders İzlenesi
- 2- Çevik Proje Takımları
- 3- Eğitimde Çeviklik Süreci
- 4- Başarı Puanlama

Çevik Eğitim Modeli (ÇEM)



Şekil 1: Çevik Eğitim Modeli

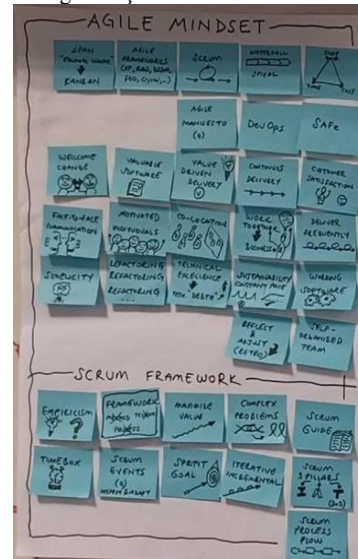
3.1. Dinamik Ders Programı (Syllabus)

Temel ders programı ve katılımcıların beklentileri doğrultusunda her hafta konu ekleme/değiřtirmelerinin dinamik olarak belirlendiđi “Çevik Yazılım ve Ürün Geliřtirme” ders içeriđi TABLO 3’te gösterilmiřtir. Dönem içinde öğrencilerden gelen geri bildirimler ve gözlemler sonucunda son hali oluşturulmuřtur. Ders planı, 15 haftalık, her hafta 3 ders řeklinde uygulanarak ilerletilmiř, öğrenciler kendi kendilerine proje takımları oluşturmuř ve 3 Sprint Scrum etkinlikleri iřletilerek ilerletilmiřtir. İlk Sprint’ten itibaren, proje takımları ve proje konuları belirlenmiřtir. Sprint 0 atlanarak Sprint 1 ile yařayarak deneme, yanılmalarla öğrenmeye olan kalıcı etkisi izlenmiřtir. Bu kapsamda, “bireyler ve takımlar uygulayarak, bizzat konu hakkında çalışarak daha hızlı ve kalıcı öğrenmeye ulařır” ilkesi benimsenmiřtir.

Tablo 3: Ders Programı

Hafta	Sprint	Konu
1	Sprint 1	Giriř- Yazılım ve Ürün Geliřtirme
2		Ürün Geliřtirme Döngüsü, Yalın ve Çeviklik Tarihçe
3		Agile Ürün Yönetimi, Manifesto
4		Takım Projeleri Sprint Gözden Geçirme, Retrospektif 1 Project Presentations
5	Sprint 2	Agile Çerçevesi – Kanban, Scrum, KATA Çalıştay
6		Scrum Etkinlikleri, Roller, Planlama, Günlük Toplantı, Gözden Geçirme, Retrospektif
7		Mid Term
8		Takım Projeleri Sprint Gözden Geçirme, Retrospektif 2
9	Sprint 3	Sprint Planlama, Tahminleme, Hikaye Puanlama ve Fibonacci Series
10		Takım oluřturma ve Duygusal Zeka
11		Büyük Ölçekte Scrum – “X-Chain” Tekniđi
12		Çevik Liderlik ve Marshmallow Challenge
13		Takım Projeleri Sprint Gözden Geçirme, Retrospektif 3
14		Ders Retrospektif
15	Release	Final

Ders içeriđi oluşturulmuř ve sınıfa asılarak Şekil 3’teki gibi görsel olması sağlanmıřtır. İçerikte tüm dönem iřlenecek konuların anahtar kelimeleri yer almaktadır. Bununla birlikte ders ana hedef/amacı ve kabul kriterleri, bunlar üzerinden rahatlıkla aktarılmıř ve řeffaf bir řekilde sunulmuřtur. Dönem boyunca içerik dinamik bir řekilde sınıftan gelen geri bildirimlere göre güncellenmiřtir. Kanban ile Scrum’ın farkı, “Hikaye puanı, SP (Story Point) ile tahminleme” konusu detaylandırılarak bu sayede müfredeta eklenebilmiřtir. Aynı zamanda yazılım projelerinde Hikaye Puanı’na göre daha objektif yazılım fonksiyonel büyüklük ölçümü yapılabilen Cosmic İşlev Puan (CFP: Cosmic Function Point) [18] gibi yöntemlere de deđinilmiřtir.



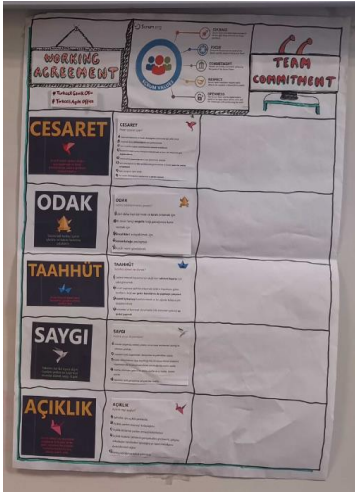
Şekil 2: Ders Birikim Listesi

3.1.1. Sınıf Anlaşması

Dersin ortak değerleri, Scrum değerleri (Cesaret, Odak, Taahhüt, Saygı, Açıklık) [19, 20] dikkate alınarak, tüm sınıf ile aşağıdaki sorular üzerinde değerlendirme yapılarak takım anlaşması olarak belirlenmiştir. Değerler üzerinden yapılan anlaşmaların ortak hedefe ve başarıya ulaşmadaki etkisinin anket sonuçlarına yansıtıldığı görülebilmektedir.

- Birlikte yaratmak istediğimiz kültür ve atmosfer ne?
- İşler zorlaştığında nasıl olmak istersiniz?
- Bu takımın büyümesine ne destek olur?
- Birbirinize ne taahhüt edersiniz?

Sınıf anlaşması, bir sonraki bölümde paylaşılan değerlerin detaylı anlamlandırıldığı maddelere göre öğrencilerle bir etkinlik ile seçilerek içeriği Şekil 3'teki gibi oluşturulmuş, benzer şekilde sınıfa asılarak ve çevrimiçi derslerde kullanılmak üzere dijital ortamda kayıt altına alınmıştır.



Şekil 3: Sınıf Anlaşması

3.1.1.1 Cesaret

“Scrum takım üyeleri doğru şeyi yapmaya ve zorlu problemlerin üstüne gitmeye cesaret ederler” [19];

1. Daha hızlı üretmemiz için baskı altındayken üretimimizle ilgili şeffaf olmak
2. Müşteriye henüz bitirmediğimiz işleri göstermemek
3. Takım hedefine katkıda bulunmayanları sorumlu tutabilmek
4. Müşterinin isteğini yanlış anladığımızı kabullenmek ve bunu fark ettiğimizde yön değiştirebilmek
5. Daha önce hiç yapmadığımız bir şeyi geliştirmeye çalışmak
6. Takım arkadaşlarımızla fikir ayrılıklarımızı gösterebilmek ve bunları yapıcı bir şekilde tartışabilmek
7. Hata yaptığımızı kabul etmek
8. Bazı şeyleri bilmediğimizi kabullenmek ve yardım istemek

3.1.1.2 Odak

“Takımdaki herkes Sprint işlerine ve takım hedefine odaklanır.” [19]

1. Sorumluluğu bir takım olarak paylaşmak
2. Büyük resmi görebilmek
3. İşleri daha hızlı bitirmek ve israfı önlemek
4. İlk önce hangi engelle boğuşacağımıza karar vermek
5. Öncelikleri daha iyi anlayabilmek

3.1.1.3 Taahhüt

“Kişiler bireysel olarak takım hedeflerini gerçekleştirmeyi taahhüt eder.” [19]

1. Sadece bireysel başarımız için değil tüm takımın başarısı için çaba göstermek
2. Scrum yapmayı taahhüt ediyorsak sadece hoşumuza giden taraflarını değil zor gelen kısımlarını da yapmaya çalışmak
3. Sürekli iyileşmeyi taahhüt etmek ve bu uğurda kolayca yön değiştirebilmek
4. Kompleks ve kaotik durumlarda bile elimizden gelenin en iyisini yapmak

3.1.1.4 Saygı

“Takımın her bir üyesi diğer üyelere yetkin ve bağımsız insanlar olarak saygı duyar” [19]

1. İnsanların farklı özgeçmişleri, deneyimleri ve yetkinlikleri vardır
2. Herkesin farklı görüşleri ve perspektifleri olabilir
3. İnsanlar doğasında üretken, yaratıcı ve karmaşık problemleri işbirliği içinde çözmeye yetkindir
4. İnsanlar ellerinden gelenin en iyisini yaparlar ve iyi niyetle hareket ederler
5. Kişiler otonomilerine saygı duyulduğunda, bir amaca hizmet ettiklerini hissettiklerinde ve uzmanlaşma fırsatı sunulduğunda motive olurlar

3.1.1.5 Açıklık

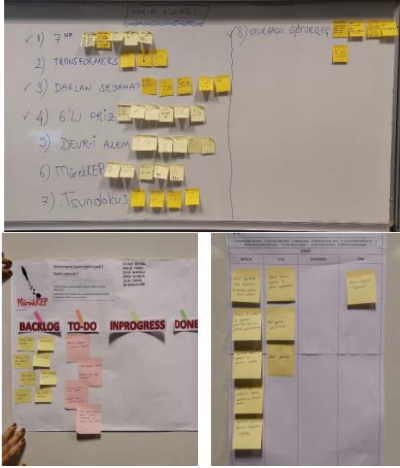
“Scrum takımı ve paydaşları yapılacak tüm işler ve bu işleri yaparken karşılaşılan zorluklar hakkında açık olmayı kabul eder” [19]

1. Birbirinden yardım isteyebilmek
2. Yardım edebilmek
3. Birbirinin perspektifini görmek
4. Yanlış yaptığımızı kabul edebilmek

3.2. Çevik Proje Takımları

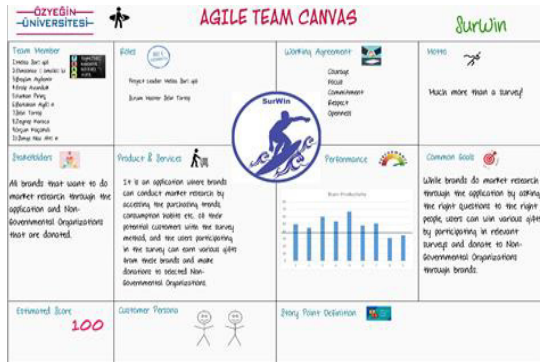
İlk ders başlar başlamaz öğrencilere dersin amacı, yapılacak örnek proje çalışmaları, beklentiler ve grup aktiviteleri aktarılmış ve sonrasında öğrencilerden kendi kendilerine takımlara bölünmeleri istenmiştir. Takımlarda kişi sayısının 10'u geçmemesi önerilir. [21] Tavsiye edilen ideal takım kişi sayısı 6 +3 'tür. Yani minimum 3, maksimum 9 kişilik takımlar önerilir. Öğrenciler buna göre, proje başlıklarından bazılarını seçerek birbirilerine teklifte bulunarak ve tanışarak 5-9 kişilik gruplara ayrılmış ve projelerini tüm sınıfa duyurmuşlardır. Her bir takım, seyahat deneyimi, enerji

tüketim verimliliği, okul servis optimizasyonu, akıllı kütüphane ve kitap okuma mobil uygulaması gibi projelerinin ana amaçlarını, hedeflerini ve çevik işletecekleri takım isim ve sloganlarını belirlemiş, takım Scrum panoları üzerinden Şekil 4'te gösterildiği gibi belirlemiştir.



Şekil 4: IE396 Dersi Çevik Proje Takımları ve Pano Örnekleri

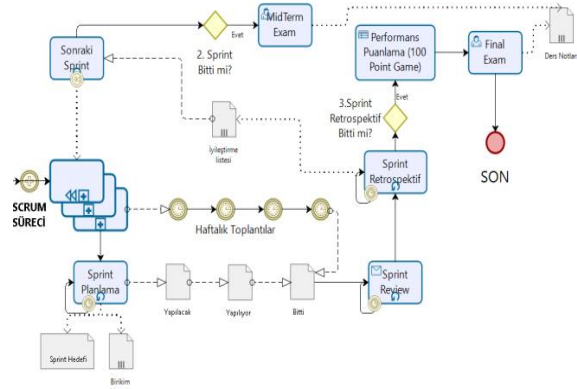
Daha sonra Şekil 5'te örnek olarak görüldüğü gibi, her bir proje takımı "Çevik Takım Kanvası" bilgilerini sunarak takım künyesini tamamlamıştır. Buradaki model ile amaç, takım ve proje ile ilgili temel bilgilerin şeffaflığının sağlanmasıdır. Kanvas'ta takım üyeleri, Scrum çerçevesi için ürün hedefi, proje lideri (ürün sahibi), takım tarafından seçilen Scrum Master, takım anlaşması, motto, paydaşlar, değerli ürün ve servisler, Hikaye Puanı (SP) üzerinden takım performans hız şeması ve müşteri persona bilgileri yer almıştır.



Şekil 5: Örnek Çevik Takım Kanvası

3.3. Eğitimde Çeviklik Süreci

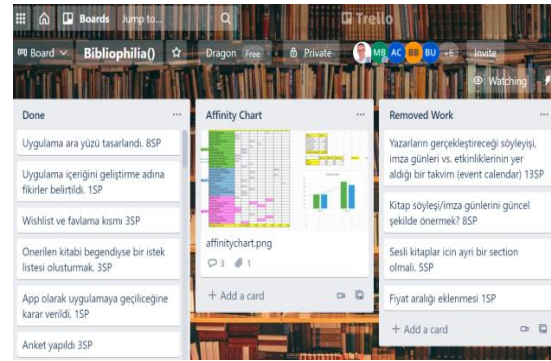
Sınıf mevcuduna göre 5-9'ar kişilik çevik proje takımları ile 15 hafta, 4 haftada bir Sprint koşarak, toplamda 3 Sprint oluşturulan değer akış süreci Şekil 6'da gösterilmiştir. Her Sprint içinde bir sonraki bölümde paylaşılacak Planlama, Haftalık Değerlendirme, Gözden Geçirme ve Retrospektif etkinlikleri gerçekleştirilmiştir.



Şekil 6: Eğitimde Çeviklik Süreci

3.3.1. Sprint Planlama

Proje takımları belirledikleri ürün hedeflerini, Şekil 7'de gösterilen bir dijital görsel pano örneğinde görüldüğü gibi şeffaf bir şekilde izlenebilirliğini sağlamıştır. Sprint içerisinde odaklanacakları tüm görevleri belirlemiş, taahhütlerini sınıf içi ve dışı tüm ilgili paydaşlara ilan etmiştir.



Şekil 7: Bir Takımın Dijital Görsel Pano Örneği

3.3.2. Haftalık Değerlendirme

Öğrencilerin takım içi iletişim problemlerini düzeltmek üzere ilk retrospektif etkinliğinde gelen bir iyileştirme önerisi sonucunda, her hafta ders başında 15 dk. haftalık çevik proje takım değerlendirme etkinliği yapılmasına karar verilmiştir. Sonraki dönemlerde takımlar kendi içinde haftalık toplanarak gözlem ve adaptasyon yetkinliklerini arttıracak yönde takımın sprint hedeflerine birlikte odaklanma konusunda pratik geliştirdiği gözlemlenmiştir.

3.3.3. Sprint Gözden Geçirme

Her 4 haftada 1 olmak üzere çevik proje takımları, seçtikleri projelerde katettikleri çalışmalarını ve çıktılarını yine tüm sınıfın önünde Şekil 8 ve 9'daki gibi şeffaf bir şekilde aktarmış geri bildirimlerle bir sonraki Sprint'i planlamak üzere ilerlemişlerdir.



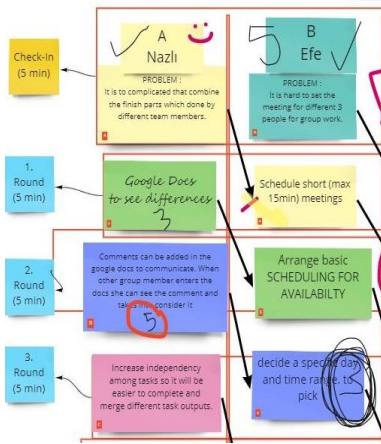
Şekil 8: İlk Sprint Gözden Geçirme



Şekil 9: Final Gözden Geçirme

3.3.4 Sprint Retrospektif

Her Sprint Gözden Geçirme sonrası takımlarla birlikte farklı teknikler öğrenerek iyileştirme önerileri oluşturmak için Retrospektif etkinliği gerçekleştirilmiştir. 3 Sprint sonunda tüm sınıf olarak ortak bir retrospektif gerçekleştirilerek genel anlamda ders ve projelerin değerlendirilmesi yapılmıştır. Pandemi döneminde dijital ortamda kullanılan Retrospektif uygulamalarından bir örnek sonuç Şekil 10'da gösterilmiştir.

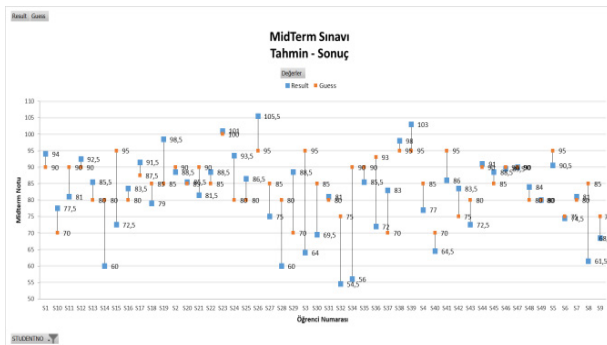


Şekil 10: Bir Retrospektif Uygulama Sonucu Örneği

3.4. Notlandırma

3.4.1. Vize ve Final Sınavları

Vize sınavında öğrencilerin sınavdan alacağı notu tahminlemesi istenmiştir. Tahminin ± 5 puan içinde tutması halinde ekstra bir 5 puan alacakları iletilerek, ders hakkında kendi öz değerlendirmeleri konusunda farkındalıkları ölçülmüş, Şekil 11'deki gibi sonuçlanmıştır. Genel sınav ortalamaları ve grafiğe göre Ara sınav sonuç formülünde TABLO 4'te verilmiştir. Tahminleme ile gerçekleşen not arasındaki korelasyon katsayısı 0,56 olarak bulunmuştur. Bu değer, 0,40-0,59 arasında kaldığı için istatistiksel olarak pozitif ilişkiye işaret etmektedir. Olasılık değeri olan p [22] de 0,064 olarak hesaplanmıştır. Buna göre istatistiksel olarak anlamlılık eğilimi olduğu ve öğrencilerin öğrenme farkındalığı pozitif olduğu düşünülmektedir.



Şekil 11: Vize Not Tahminleri ve Sonuç İlişkisi

Tablo 4: Sınav Not Sonuç Tablosu

Sınav Tipi	Not Ort.	Median	Formül
Vize	7837	82.25	$y = 0,8x, R^2 = 0.3$
Final	90.17	92.30	

3.4.2. Ödevler

Kendi kendine öğrenmeyi besleyecek bir çalışma için öğrencilere Şekil 12'de görülen örnekler gibi, birçok konuda video, kitap, araştırma ödevleri verilmiştir. Ödevlerde aşağıdaki gibi kabul kriterleri net bir şekilde belirlenmiştir. Önemli kriterlerden bazıları şunlardır: yorumlama, eleştirme, bir gözlemci ya da arkadaşı ile değerlendirme yapma, eleştirel düşünce, kendi kendine öğrenme.

Kabul Kriterleri;

- Konu ile ilgili en az 3 makale/teknik yayın okunması
- Tek sayfada kendi yorumuyla aktarılması
- En az 1 gözlemci ile değerlendirme yapılması
- Kendi anlatımı ile organizasyon içinde paylaşım yapılması, geri bildirim alması
- Bu konuda problemli yer tespit ederek iyileştirme önerileri ile birlikte geri bildirim alması

Ödev Konusu	Kabul Kriterleri
Agile Frameworks	1. DSDM, RAD, XP, Crystal, DevOps, Scrum ve Kanban süreçlerini araştırmak, anlamak. → 10 KP 2. Tüm çevrelerle ilgili bir sayfa/kitap tasarımı yapmak → 10 KP 3. Farklı kendi anlatımı ile organizasyon içinde paylaşım yapmak. (Open Space) Yapılan etkinlik geri bildirim notu → 10 KP
Video - Resource Utilization - Henric https://www.youtube.com/watch?v=CostXs2p6r0	1. İzlemek → 1 KP 2. Resource Utilization ile ilgili makale/paper okumak, araştırmak. → 3 KP 3. Yorumlamak → 5 KP 4. Organizasyonda bu konuda problemi yer tespit etmek ve iyileştirme önerileri hazırlamak. En az 1 gözlemci ile değerlendirme yapmak. Geri bildirim sonucu not. → 10 KP
Video - Spotify Eng. Culture 1&2 https://www.youtube.com/watch?v=4GK1NDTWbky https://www.youtube.com/watch?v=Ot48bWlWQw	1. İzlemek → 2 KP 2. Organizasyon ihtiyaçlarını yorumlamak → 5 KP 4. Organizasyonda bu konuda problemi yer tespit etmek ve iyileştirme önerileri hazırlamak. En az 1 gözlemci ile değerlendirme yapmak. Geri bildirim sonucu not. → 10 KP
Kitap - Scrum Handbook - Jeff Sutherland's https://www.researchgate.net/publication/301685699_Jeff_Sutherland's_Scrum_Handbook	1. Okumak → 20 KP 2. Özetle → 10 KP 3. Yorumla, Eleştir → 10 KP 4. Komite, Hoca Değerlendirme Notu → 10 KP
Video - Change - Henric https://www.youtube.com/watch?v=H-TipXmVdaQ	1. İzlemek → 1 KP 2. Change Management ile ilgili makale/paper okumak, araştırmak. → 3 KP 3. Yorumlamak → 5 KP 4. Organizasyonda bu konuda problemi yer tespit etmek ve iyileştirme önerileri hazırlamak. En az 1 gözlemci ile değerlendirme yapmak. Geri bildirim sonucu not. → 10 KP

Şekil 12: IE396 - Ödev Örnekleri

3.4.3. 100 Puan Oyunu

Takım ve bireysel performans değerlendirmelerinde en etkili yöntemlerden biri de ekran değerlendirmesidir. Buna göre takımlar birbirini TABLO 5'te gösterildiği gibi "100 Point Game" [23] tekniği ile değerlendirerek takım ve bireylerin katkı ve performanslarını belirlemiştir. Derslere aktif katılım, takım çalışmasına ve çıktılara katkı sağlama konusunda etkili öğrencilerin bu teknik ile takım tarafından öne çıkarıldığı gözlemlenmiştir.

TABLO 6'da gösterildiği gibi eğitmen puanlama sonuçlarının, takımların birbirlerini değerlendirmesi toplam puan sıralaması ile paralellik gösterdiği anlaşılmaktadır. Eğitmen puanlama formülünde, dönem başında iletilen beklentilere göre, proje hedeflerinin netliği (Neden), Sprint'ler boyunca nasıl bir çözüm ürettikleri (Nasıl) ve sonuçta ortaya çıkarılan fayda/değer (Ne) değerleri ayrı ayrı hesaplanmıştır.

Bazı takımların başarısının sınav sonuçlarına da yansıtıldığı gözlemlenmiştir. Bu varsayımıla takım olarak birlikte hareket ederek, kendi kendini yönetebilen, kendi kararlarını alabilecek yüksek performanslı takımların başarılı olmalarının daha yüksek oranda gerçekleşeceği sonucuna varılabilir.

Tablo 5: Proje Takımları ve 100 Puan Dağılım Tablosu

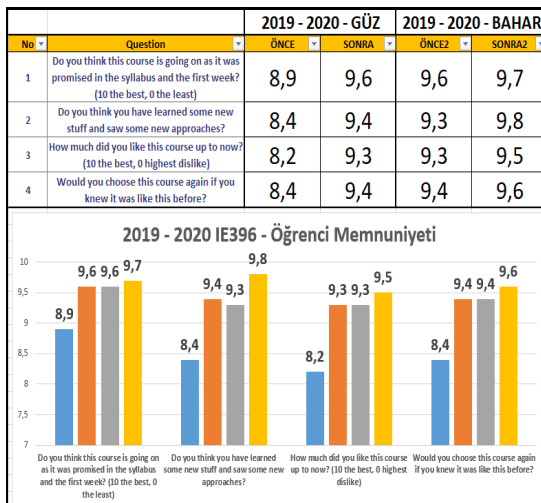
Takım	Alınan Puanlar					Top. Oy	Top. Puan
	10	20	30	40	50		
Devr-i Alem	8	11	2	1	8	38	800
G8tUp	8	11	1	5	2	36	630
MürekKEP	13	6	7	1	2	38	600
GOGO	12	6	4	2	3	36	590
8'li Priz	15	4	0	1	2	32	370
Ocean	8	5	2	0	2	34	340
Transformers	11	3	1	0	3	33	350
Darlan Seyahat	13	3	1	0	0	32	220

Tablo 6: Eğitmen Proje Puanlama Dağılım Tablosu

Takım	Eğitmen Puanlama W-H-W *20	Neden? [1-5] %30	Nasıl? [1-5] %40	Ne? [1-5] %30
Devr-i Alem	94	5	5	4
G8tUp	94	5	5	4
MürekKEP	94	5	5	4
GOGO	89	5	4	4,5
8'li Priz	94	5	5	4
Ocean	94	5	5	4
Transformers	82	5	3,9	3,5
Darlan Seyahat	94	5	5	4

3.4.4. Anket Sonuçları

Şekil 13'te 2019-2020 GÜZ ve BAHAR dönemlerine ait, ÖNCE ve SONRA Öğrenci Memnuniyet sonuçları karşılaştırmalı gösterilmiştir. Her 2 dönemde, 1. Sprint sonlarına doğru, 5. haftada ve Final sonrası yapılan memnuniyet anketlerinde 100'e yakın öğrenci, 4 temel soruda 10 üzerinden değerlendirmede bulunmuştur. Öğrenci memnuniyetinin önce ve sonra anket değerlerine bakıldığında, 1. Dönem yaklaşık +%11, 2. dönemde ise 1.dönem sonuçları üzerine +%2 iyileşerek sonuçlandığı ortaya çıkmıştır.



Şekil 13: Vize – Final sonrası Anket Sonuç İlişkisi

Benzer şekilde 2020-2021 GÜZ döneminde, Özyeğin Üniversitesi'nin yaptığı sene sonu ders değerlendirme

anketine, 58 öğrenciden 50 öğrenci katılmış, dersin genel puanı, 5 üzerinden 4.84 olarak gerçekleşmiştir. Anketler açıklama bölümünde yapılan genel ders yorumları Tematik Analiz metodu [24] ile analiz edilmiş, ortaya çıkan kod ve tema değerleri TABLO 7'de gösterilmiştir. Buna göre tecrübe, keyifli iş yapma ve kendini gösterebileceğin bir ortamda öğrenmenin yüksek seviyede olduğu sonucuna varılabilir.

Tablo 7: Tematik Analiz ile Ders Anketi Tablosu

Tema	Kodlar	Sıklık
Öğrenme Yolculuğu	Tecrübe, Keyif, nasıl hissettiğimizizin düşünülmesi, kendini ifade etme	9
Takım çalışması	Ekip üyelerimiz, grup projeleri	3
Karar verme	Fikirler, tartışmayı sevmek, tüm ekip üyeleriyle anlaşma zorluğu, fikirlerini paylaşmak, geri bildirim almak, her şeyi tartışmak, fikir geliştirmek	3

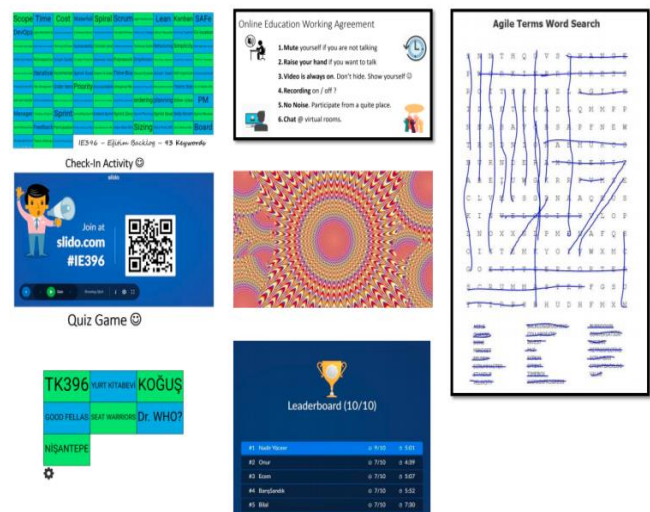
3.5. Grup Aktiviteleri

Öğrenmeyi kalıcı hale getirmek için uygulanan sınıf içi aktivite ve çalıştaylar, öğrencilerin verdiği geri bildirimler de dikkate alındığında derse katılımı artırmıştır. Ayrıca notlandırma (3.4) ve anket sonuçlarına (3.4.4) göre de öğrenmenin ve ders başarı notunun olumlu yönde etkilendiği anlaşılmıştır.

3.5.1. Derse Giriş Etkinlikleri

Dersin etkinliğini arttırmak için ilk girişte uygulanacak yöntemlerin çeşitliliği ve oyunlaştırma yaklaşımı derse olan ilgi ve odaklanmayı artırmış, aktif katılım göstermelerine katkı sağladığı gözlemlenmiştir. Bu etkinliklerden bazıları Şekil 14'te gösterilmiştir.

- Ders Anlaşması
- Ders anahtar konuları rastgele soru-cevaplar
- Çevrimiçi anlık sınav ve lider panosu
- Ders içeriği ile ilgili "Kelime Avı" Bulmaca



Şekil 14: Derse Giriş Etkinlikleri.

3.5.2. Takım Olma Etkinlikleri

3.5.2.1 Marshmallow Meydan Okuması

Öğrencilerin birlikte takım olarak bir ürün geliştirmeyi deneyimlemesi için tasarlanan aktivitelerden biri Marshmallow Meydan Okumasıdır (Challenge) [25]. Şekil 15'te bu etkinlikten bazı çalışmalar gösterilmiştir. Birlikte nasıl daha iyi çalışılabildiği, hedef doğrultusunda nasıl organize olup yaratıcı ve özgün fikirler üretilebildiği, takımlar arası yardımlaşma ve rekabetin avantajları gözlemlenmiştir.



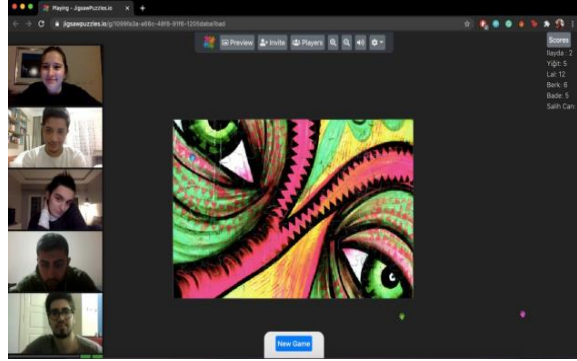
Şekil 15: Takım Etkinliği : Marshmallow Challenge

3.5.2.2 KATA Çalıştayı

Takım olarak başarıya ulaşma ve sürekli iyileştirecek mükemmelle ulaşmayı hedefleyen tekniklerden biri de KATA çalıştayıdır [26]. Çevik proje takımları 35 parçalık yapbozları en kısa sürede bitirmek için çabalayarak, kendileri ile yarışmışlardır. Her bir turda tüm Scrum etkinlikleri deneyimlenerek, takımların ölçerek nasıl hedefe yaklaşabildikleri, takım hızı öğrenme, tahminleme ve ölçme kabiliyetleri ve bunun deneyimle olan ilişkisi oyunlaştırma tekniği ile Şekil 16'da sınıf etkinliğinde gösterildiği gibi, Şekil 17'de de uzaktan çevrimiçi dijital uygulama üzerinden gösterildiği gibi uygulanmıştır.



Şekil 16: Sınıf içi Takım Etkinliği: KATA Çalıştayı



Şekil 17: Çevrimiçi Dijital Takım Etkinliği: KATA Çalıştayı

4. Tartışma

Aşağıdaki alt bölümlerde bu çalışmanın katkıları, sektöre etkisi ve kısıtları tartışılmıştır.

Bu alandaki diğer güncel araştırmalarla karşılaştırıldığında, önerimizin çeşitli avantajları vardır. Bu avantajlar ve sektöre etkileri özetlenmiştir:

- (1) Geleneksel öğrenme ve ders eğitim yaklaşımlarına karşın uzaktan, etkileşimli ve oyunlaştırma teknikleri ile kendi kendine öğrenme konusunda pozitif yaklaşım sergilenmiştir.
- (2) Endüstri beklentileri ile akademik eğitim kapsamı arasındaki açığın kapanması ve teorik ile pratik arasında bir köprü niteliğinde projelere odaklanılmıştır. Bu sayede gerçek dünyanın sorunlarına çevik çerçeve değer ve prensipleri ile döngüsel ürün geliştirme simülasyonu gerçekleştirilmiştir.
- (3) Ders notu belirlemede geleneksel ve diğer benzer çalışmalardan farklı olarak "100 Point Game" yaklaşımı ile performans değerlendirmeye farklı bir bakış açısı getirilmiştir.
- (4) Öğrenci memnuniyeti sonuçlarına bakıldığında ilk GÜZ döneminde yaklaşık 1 puanlık bir artışla 9,4 olarak gerçekleşmiş, ikinci BAHAR dönemi başında da ekran tavsiyeleri ile zaten yüksek memnuniyet notu ile başlanmış ve sene sonuna doğru yine yukarı yönde artışla ortalama yüksek memnuniyet puanı ile sonuçlanmıştır.

5. Geçerliliğe Yönelik Tehditler

Bu çalışmanın geçerliliğinin artırılması için farklı öğrenci kitlelerinin farklı zamanlarda aldığı derslerde zenginleştirilerek denenmesi önerilmektedir. Ayrıca eğitimin çevrimiçi, sınıf ortamında veya hibrit şekillerde gerçekleştirildiği durumlarda modelin geliştirilmesi gereken tarafları ortaya çıkabilir. Bunun için modelin farklı şekillerde gerçekleştirilen eğitim mecralarında validasyonu önemlidir. Proje başarısına takımdaki her bireyin katkı seviyesi, birey performansı ve ekip performansının ilişkisi dikkatle gözlemlenmesi gereken bir alan olarak karşımıza çıkmıştır. Konunun bu yönü, tartışmaya açık bir araştırma alanı olarak gelecekte irdelenebilir.

6. Sonuçlar ve Gelecek Çalışmalar

Bu çalışmada, Türkiye'deki bilgi ve iletişim teknolojileri alanında mühendislik eğitimi ile endüstri beklentileri

arasındaki farkın kapanmasına yönelik yeni bir eğitim yaklaşımı olan “Çevik Eğitim Modeli” tasarımı ve örnek sonuçları yer almaktadır. Özyeğin Üniversitesi’nde “Çevik Yazılım ve Ürün Geliştirme” dersinde, öğrenciler ile 15 haftada, 4 haftalık Sprint’lerle, 3 Sprint koşarak kendi kendine öğrenme, öğrencilerin başarıya olan katkılarının takım çalışması olarak farkı, hedefe koşma ve öğrenmenin oyunlaştırma ile etkinliği test edilmiştir.

Buradaki sonuçlara bakıldığında;

- Ders katılım oranının akşam dersi olmasına rağmen diğer derslere nazaran gözle görülür derecede arttığı, gündüz derslere göre çok daha fazla katılım olduğu gözlemlenmiştir. Dersi alan ~60 öğrencinin ortalama %50’si derslere katılmıştır.
- Öğrencilerin kendi kendilerine organize olarak gruplara ayrılması ve projeleri de yine kendilerinin belirlemesinin sahiplenme ve hedefe ulaşmaya anlamlı bir katkısı olduğu, önemli bir motivasyon unsuru olduğu ve böylece derse aktif katılımı artırdığı görülmüştür. Ayrıca ilgili literatürde [27] belirtilen, teknik yeteneklerin tek başına yetersiz olduğu, bunun yanı sıra örneğin ekip oyunculuğu gibi davranışsal yetkinliklerin de (soft skills) çalışma hayatına hazırlık açısından mutlaka deneyimlenerek geliştirilmesinin ve kişisel öğrenmenin ve gelişimin önemi bu çalışmada teyit edilmiştir.
- Dinamik ders kataloğuna yer vermek, öğrencilerin, geri bildirimlerinin dikkate alındığını görmesine yol açmış ve derse olan ilgilerinin artmasını sağlamıştır. Ayrıca bu yöntemin daha çok sorumluluk alma ve aksiyona geçme konusunda etkili olduğu görülmüş, şeffaflık sorunların ele alınabilmesini kolaylaştırması bakımından da faydalı olduğu görülmüştür.

Bu çalışmanın devamında, eğitimlerde takımlarla birçok farklı proje yapılması yerine önceden belirlenmiş tek bir ürün üzerinden, aynı ürüne katkı sağlayan takımların büyük ölçekteki organizasyon performansı gözlemlenebilir.

Ayrıca kısıtlarda belirtildiği gibi, eğitimin hem sınıf içi hem de çevrimiçi olması, yani hibrit eğitim düzenine göre uygulanması gerektiğinde, modelin geliştirilmesi için çalışmalar yapılabilir. Benzer bir şekilde takım çalışmalarında performans değerlendirme, ders notu ile ilişkilendirme konularında farklı yaklaşımlar geliştirilebilir.

7. Kaynaklar

- [1] Collabnet Versionone, “Introducing the 13th Annual State of Agile Report”, <https://www.stateofagile.com/#ufh-i-521280178-blog-introducing-the-13th-annual-state-of-agile-report/473508>, 2019.
- [2] V. Garousi, G. Giray, E. Tüzün, C. Catal and M. Felderer, "Aligning software engineering education with industrial needs: A meta-analysis", *J. Syst. Softw.*, vol. 156, pp. 65-83, Oct. 2019.
- [3] K. Beck, M. Beedle, A. van Bennekum, A. Cockburn, W. Cunnin-gham, M. Fowler, J. Grenning, J. Highsmith, A. Hunt, R. Jeffries, J. Kern, B. Marick, R. C. Martin, S. Mellor, K. Schwaber, J. Sutherland, and D. Thomas, “Manifesto for Agile Software Development”, 2001.
- [4] E. Sarıkaya, S. Bağrıyanık and M. Gökalg, “Teaching Agile Software Development Using Agile Methods: A Case Study,” 2020 Turkish National Software Engineering Symposium (UYMS), Istanbul, Turkey, 2020, pp. 1-6, doi: 10.1109/UYMS50627.2020.9247027.
- [5] K. Schwaber, J. Sutherland, “Scrum Guides”, <https://www.scrumguides.org/>, (Son erişim: 15 Şubat 2020).
- [6] Beck, K., Fowler, M., “Planning Extreme Programming”, Addison-Wesley, Reading, 2000.
- [7] Takeuchi, H. & Nonaka, I., “The New New Product Development Game”, *Harvard Business Review*, <https://hbr.org/1986/01/the-new-new-product-development-game>, 1986.
- [8] Delhij, A., van Solingen, R., & Wijnands, W., “The eduScrum Guide (No. 1.2)”, (p. 21). Retrieved from https://www.eduscrum.nl/img/The_eduScrum_guide_English_2.pdf, 2015.
- [9] Dr. David F. Rico, Dr. Hasan H. Sayani, “Use of Agile Methods in Software Engineering Education”, <http://w.davidfrico.com/rico09a.pdf>, (Son erişim: 21 Ocak 2020).
- [10] Melnik, G., & Maurer, F., “Introducing Agile Methods in Learning Environments: Lessons Learned”, In *Conference on Extreme Programming and Agile Methods* (pp. 172–184), 2003.
- [11] Alfonso, M. I., & Botia, A., “An Iterative and Agile Process Model for Teaching Software Engineering”, In *IEEE International Conference on Software Engineering Education and Training (CSEE&T)* (pp. 9–16), 2005.
- [12] Meerbaum-Salant, O., & Hazzan, O., “An Agile Constructionist Mentoring Methodology for Software Projects in the High School”, *ACM Transactions on Computing Education (TOCE)*, 9(4), 21, 2010.
- [13] Scharff, C., & Verma, R., “Scrum to Support Mobile Application Development Projects in a Just-in-Time Learning Context”, In *ICSE Workshop on Cooperative and Human Aspects of Software Engineering* (pp. 25–31), 2010.
- [14] Heinonen, K., Hirvikoski, K., Luukkainen, M., & Vihavainen, A., “Learning Agile Software Engineering Practices Using Coding Dojo.”, In *ACM SIGITE Conference on Information Technology Education (SIGITE)* (pp. 97–102), 2013.
- [15] Ahmad, M. O., Liukkunen, K., & Markkula, J., “Student Perceptions and Attitudes Towards the Software Factory as a Learning Environment”, In *IEEE Global Engineering Education Conference (EDUCON)* (pp. 422–428), 2014.
- [16] Paasivaara, M., Heikkilä, V., Lassenius, C., & Toivola, T., “Teaching Students Scrum Using LEGO Blocks”, In *IEEE/ACM International Conference on Software Engineering: Software Engineering Education and Training Track (ICSE-SEET)* (pp. 382–391), 2014.
- [17] Noguera, I., Guerrero-Roldán, A.-E., & Masó, R., “Collaborative Agile Learning in Online Environments: Strategies for Improving Team Regulation and Project Management”, *Computers & Education*, 116, 110–129, 2018.
- [18] Ertaban, C., Gezgin, S., Bağrıyanık, S., Albey, E., & Karahoca, A., “Çevik yöntemlerde cosmic işlev puanı ve hikaye puanının birlikte kullanımı”, In *CEUR Workshop Proceedings. CEUR-WS*, 2017.
- [19] Scrum.org., <https://www.scrum.org/resources/scrum-values-poster> (2016), (Son erişim: 15 Şubat 2020).
- [20] G. Verheyen., <https://guntherverheyen.com/2013/05/03/theres-value-in-the-scrum-values/>, (Son erişim: 15 Şubat 2020).
- [21] Lighthouse., <https://getlighthouse.com/blog/developing-leaders-team-grows-big>, (Son erişim: 22 Aralık 2021).

- [22] Dennis D. Boos & Leonard A. Stefanski, “P-Value Precision and Reproducibility, The American Statistician”, 65:4, 213-221, DOI: 10.1198/tas.2011.10129, 2021.
- [23] Leffingwell, Dean & Widrig, Don, “Managing Software Requirements: A Use Case Approach”, 2003.
- [24] V. Braun and V. Clarke, “Using thematic analysis in psychology”, Qual. Res. Psychol., vol. 3, no. 2, pp. 77–101, 2006.
- [25] T. Wujec., “Marshmallow Challenge”, <https://www.tomwujec.com/marshmallowchallenge>, (Son erişim: 15 Şubat 2020).
- [26] Creative Commons Attribution, www.katagrow.com, (Son erişim: 15 Şubat 2020).
- [27] Bagriyanik, S., & Karahoca, A., “Yazılım Mühendisliği Profesyonellerinin Kişisel Öğrenmesi: Türkiye’den Bir İnceleme”, In National Software Engineering Proceedings. UYMS, 2019.

Özgeçmişler



Erkan Sarıkaya Marmara Üniversitesi Bilgisayar Mühendisliği Bölümü'nden mezun oldu.

15+ yıllık Telekomünikasyon, Yazılım, Ürün Geliştirme ve yöneticilik deneyimine sahiptir.

Turkcell'de 9 yıl Yazılım Yöneticiliği sonrası "Çevik Ofis" kuruluşunda yer alarak, son 5 yılda 100+ çevik takım dönüşümünde aktif rol aldı.

Farklı organizasyonlarda, IT, non-IT ekiplere ve liderlere "Çevik" ve "Liderlik" kapsamında bir çok eğitim verdi.

2020 yılından itibaren "Çevik Ofis" markasıyla "Çevik Koç / Danışman" olarak Agile, Lean, Çevik Dönüşüm konularında organizasyonlara eğitim & danışmanlık hizmeti vermeye devam ediyor.

Halen Özyeğin Üniversitesi'nde Çevik Yazılım ve Ürün Geliştirme dersini vermektedir.

UYMS de "Eğitimde Çeviklik" konulu bir yayını ve XP 2018 konferansında "Çevik Metrikler" ile ilgili poster yayını bulunmaktadır.

Çeviklik ile ilgili PSM-II, SPS, PMI-ACP, PAL-I, PSM-I, PSPO-I sertifikalarını alan Erkan Sarıkaya evli ve 3 çocuk babasıdır.



Selami Bağrıyanık, Lisans ve Yüksek Lisans eğitimini İTÜ Bilgisayar Mühendisliğinde, MBA ve Bilgisayar Mühendisliği doktora eğitimini ise sırasıyla Bilgi Üniversitesinde ve Bahçeşehir Üniversitesinde tamamlayan Selami Bağrıyanık, 24 seneyi aşkın bir süredir Bilgi ve İletişim Teknolojileri sektöründe çalışmaktadır. Bu süre zarfında, çeşitli önde gelen firmalarda bir çok yeni yazılım ürünü ve uygulamasının geliştirme mutfağında uzman ve lider rolleri ile yer almıştır. Mevcutta Singularity Software'de Genel Müdür ve Esfa Grup'ta CTO olarak çalışmaktadır. Endüstrideki çalışmalarının yanı sıra üniversitelerde lisans ve yüksek lisans dersleri vermekte ve Ulusal/Uluslararası mesleki/bilimsel kuruluşlarda görev almaktadır. Yazılım mühendisliği, yazılım geliştirme, bulut bilişim, veri bilimi, yapay zeka ve diğer yazılım tabanlı yeni teknolojiler başlıca ilgi alanlarıdır.



Mesut Gökalp, 2007 yılından itibaren ETİ, Keytorc, Superonline ve uzun yıllar Turkcell başta olmak üzere farklı firmalarda Yazılım, Test Mühendisi ve Data Analyst gibi rollerde görevler aldı. Turkcell'in ICT Çevik Dönüşüm Yolculuğunda aktif rol almasının ardından, Eureka Sigorta'da Çevik Koç'luk görevine devam etmektedir.

Çeviklik, Yalın Felsefe, Liderlik, Hedef Belirleme (OKR), Girişimcilik (Lean Startup) gibi bir çok konuda eğitimler vermektedir.

İstanbul Proje Yönetim Derneği (IPYD) Yönetim Kurulu üyesi olup, aktif olarak Özyeğin Üniversitesi'nde Agile Management dersini vermeye devam etmektedir.

Trakya Üniversitesi Bilgisayar Mühendisliği Lisans ve Bahçeşehir Üniversitesi Bilgi Teknolojileri Yüksek Lisans mezunu, evli ve 2 çocuk babasıdır.

LinkedIn :

<https://www.linkedin.com/in/mesutgokalp/>

Yapay Zeka Tabanlı Klinik Karar Destek Sistemi ile Tüp Bebek Tedavisi Gebelik Sonucu Tahmini

Predicting Pregnancy Outcome in In Vitro Fertilization Using an Artificial Intelligence Based Clinical Decision Support System

Zeynep Pacci¹, Yasemin Atılğan Şengül², Oya Alagöz³, Rukset Attar³, Aslı Uyar⁴



¹ Okan Üniversitesi, Fen Bilimleri Enstitüsü, Bilgisayar Mühendisliği Bölümü, İstanbul, Türkiye
zpacci@gmail.com

² Doğuş Üniversitesi, Mühendislik Fakültesi, Endüstri Mühendisliği Bölümü, İstanbul, Türkiye
yatilgan@dogus.edu.tr

³Yeditepe Üniversitesi, Tıp Fakültesi, Kadın Hastalıkları ve Doğum, İstanbul, Türkiye
oakcin@yeditepe.edu.tr, rattar@yeditepe.edu.tr

⁴Okan Üniversitesi, Mühendislik Fakültesi, Bilgisayar Mühendisliği Bölümü, İstanbul, Türkiye
asli.uyar@okan.edu.tr

Öz

Tüp bebek tedavisi başarı olasılığının henüz tedavi başlamadan belirlenmesi hastalar ve klinisyenler açısından önem taşımaktadır. Yapay zeka tabanlı klinik karar destek sistemleri, geçmiş tedavi verilerini analiz ederek yeni tedavilerde gebelik sonucunun tahmin edilmesine olanak sağlar. Bu çalışmada, tüp bebek tedavisine başlayacak hastaya ait öznitelikler kullanılarak pozitif gebelik olasılığını hesaplayan bir model geliştirilmiştir. Çalışmada kullanılan veri kümesi Yeditepe Üniversitesi Hastanesi Tüp Bebek Kliniği'nde 2013-2019 yılları arasında gerçekleştirilen 1154 adet tedavi siklusuna ait elektronik sağlık kayıtlarından oluşmaktadır. Veri kümesi üzerinde beş farklı sınıflandırma yöntemi (Destek Vektör Makineleri, Çok Katmanlı Algılayıcı, Rastgele Orman, Aşırı Gradyan Artırma ve Hafif Gradyan Artırma) 5-katlı çapraz doğrulama yöntemi kullanılarak karşılaştırmalı olarak incelenmiştir. Gebelik sonucu tahmininde en yüksek sınıflandırma performansı Destek Vektör Makineleri yöntemi ile elde edilmiş (AUC=0.70) ve sınıflandırma olasılık sonuçlarında karar eşik değerinin optimizasyonu ile tahmin doğruluğu daha da iyileştirilerek gebelik sonucunun %71.7 Doğru Pozitif ve %59.4 Doğru Negatif oranı ile tahmin edilmesi sağlanmıştır.

Anahtar Kelimeler: klinik karar destek sistemi, yapay zeka, tüp bebek tedavisi

Abstract

In in vitro fertilization (IVF) treatment, it is important for patients and clinicians to determine the success probability of the treatment before it begins. Artificial intelligence-based clinical decision support systems provide the opportunity of estimating pregnancy outcome in new treatments by analyzing past treatment data. This study presents a model developed to predict the positive pregnancy outcome using patients' features. Data set used in the study consists of 1154 treatment cycles performed in Yeditepe University Hospital IVF Clinic between 2013-2019. Five different classification methods (Support Vector Machine, Multi-Layer Perceptron, Random

Forest, Extreme Gradient Boosting, Light Gradient Boosting) are applied using five-fold cross validation method and analyzed comparatively. Among five classification methods the one with highest performance was Support Vector Machine, of which AUC could reach 0.70 and by optimizing the decision threshold value of the classification probabilities, accuracy of pregnancy prediction is improved as 71.7% True Positive and 59.4% True Negative.

Keywords: clinical decision support system, artificial intelligence, IVF treatment

1. Giriş

Yapay zeka tabanlı karar destek sistemleri, daha önce uygulanmış çeşitli tedavi yöntemlerinin sonuçlarının ve "büyük veri" boyutlarındaki klinik veri tabanlarının analizi ile hastaya en uygun tedavinin belirlenmesini hedeflemektedir [1]. Bu sistemlerin uygulanması ile kanser, bağışıklık sistemi hastalıkları, nörolojik hastalıklar gibi tanısı çok sayıda parametreye dayanan ve tedavisi "hassas tıp (precision medicine)" yaklaşımı gerektiren hastalıklar için hızlı bir şekilde tanı konulması ve tedavi başarı oranlarının artırılması sağlanmaktadır [2][3][4].

Başarı oranlarının artırılması için klinik karar destek sistemlerinin kullanıldığı bir diğer tıp alanı da infertilite tanısı sonucu uygulanan tüp bebek tedavi sürecidir. İnfertilite, çiftlerin minimum bir yıl süreyle, herhangi bir korunma yöntemi kullanmamalarına ve düzenli cinsel ilişkide bulunmalarına rağmen, çocuk sahibi olamamaları durumudur [5].

Son zamanlarda, infertilitenin dünya genelinde çiftlerin %8 ile %12'sini etkilediği tahmin edilmekte ve bu oran giderek artmaktadır [6]. Tüp bebek tedavisi, infertilite ile ilgili üremeye yardımcı tedaviler içinde en yaygın kullanılan tedavi türüdür. Tüp bebek tedavi sürecinin ilk aşaması yumurtalıklardan ideal sayı ve kalitede yumurta toplamak amacıyla yapılan hormon

tedavisidir. Bu aşamadan sonra elde edilen yumurtalar toplanır ve paralel olarak hazırlanan sperm ile yumurtalar laboratuvar ortamında döllenir. Döllenen yumurtalar laboratuvar ortamında 2 ile 6 gün arası geliştirilir ve gelişimleri gözlemlenir. Son olarak da seçilen bir ya da birden fazla embriyonun kadın rahmine transfer edilmesiyle tedavi süreci sonlanır. Transferi takip eden sekizinci hafta sonunda bebeğin kalp atışlarının oluşup oluşmamasına bağlı olarak tedavinin pozitif veya negatif gebelik ile sonuçlandığı belirlenir.

İlk tüp bebek çalışması 1975 yılında yayınlanmış olup [7], tüp bebek tedavisi sonucu ilk bebek 1978 yılında dünyaya gelmiştir [8]. Aradan 43 yıl geçmesine ve bu süreçte büyük bir ivme ile gelişen teknolojiye rağmen tüp bebek tedavisindeki başarı oranları tüm dünyadaki farklı kliniklerden %10 ila %40 arasında raporlanmaktadır [9]. Oranların bu denli düşük olmasının sebeplerinden birisi de sürecin karmaşıklığı ve farklı aşamalarda kritik kararların tedavi sonucunu doğrudan etkilemesidir. Tüp bebek tedavi sürecinin, hastaların genlerinin,

yaşam tarzı ve çevresel etkenlerin dikkate alındığı bir yaklaşım olan hassas tıp yaklaşımı ile optimize edilebileceği ve bu yöntemle başarı oranlarının artırılacağı de önerilmektedir [10].

Bu karmaşık ve hassas tedavi süreci, yapay zeka tabanlı karar destek sistemlerinin katkı sağlayabileceği bir alandır. Bugüne kadar tüp bebek konusunda yapılmış birçok yapay zeka çalışması bulunmaktadır [11][12][13]. Bu alanda gerçekleşen ilk yapay zeka odaklı çalışma, 1997 yılında Kaufmann tarafından yapılmıştır. Çalışmada, canlı doğum oranını tahminleme üzerine bir makine öğrenmesi modeli oluşturulmuştur [14]. Yapılan bu ilk çalışmadan itibaren canlı doğum oranı tahminleme [15] [16], semen analizi sınıflandırması [17], embriyo seçimi [18], uterus analizi [19], embriyo tabanlı tedavi sonucu tahminlemesi [20] gibi infertilite tedavisine katkı sağlamak amaçlı birçok yapay zeka tabanlı çalışma mevcuttur. Bu çalışmalarda kullanılan öznitelikler, sınıflandırma yöntemleri ve sonuçlar Tablo 1'de özetlenmiştir.

Tablo 1: İnfertilite tedavisi alanında yapılmış Yapay Zeka tabanlı çalışmalar hakkında özet bilgi

Tedavi Başarı Kriteri	Çalışma Popülasyonu	Öznitelikler	Sınıflandırma Yöntemi	Performans Ölçütü	Sonuç	Referanslar
implantasyon	637 (hasta)	embriyo morfolojik parametreleri	Yapay Sinir Ağı	AUC*	[0.64, 0.77]	Bori vd., 2020 ^[12]
stimulasyon	2,603 (siklus)	stimulasyon hormon değerleri	Yapay Sinir Ağı, Destek Vektör Makineleri, Karar Ağaçları, Rasgele Orman, Lojistik Regresyon	Doğruluk	[%82, %96]	Letterie, Mc Donald, 2020 ^[13]
gebelik	455 (hasta)	hasta klinik verileri ve siklus parametreleri	Yapay Sinir Ağı	Doğruluk	%59	Kaufmann vd., 1997 ^[14]
canlı doğum	426 (siklus)	hasta klinik verileri ve embriyo parametreleri	Yapay Sinir Ağı	Duyarlılık ve özgüllük	Duyarlılık=%76.7; özgüllük=%73.4	Vogiatzi vd., 2019 ^[15]
canlı doğum	381 (hasta)	Anti Müllerian Hormon (AMH) değeri ve yaş	Lojistik regresyon	Doğruluk, duyarlılık ve özgüllük	Doğruluk=%53.5; duyarlılık=%79.2; özgüllük %44.2	La Marca vd., 2011 ^[16]
implantasyon	63 (siklus)	hasta klinik verileri ve embriyo morfolojik parametreleri	Bayes sınıflandırıcı	Doğruluk	%71.43	Morales vd., 2008 ^[18]
implantasyon	2453 (embriyo)	hasta klinik verileri ve siklus/embriyo parametreleri	naïve Bayes sınıflandırıcı	AUC	0.75	Uyar vd., 2015 ^[20]
gebelik ve canlı doğum	136 (hasta)	hasta klinik verileri ve siklus parametreleri	Yapay Sinir Ağı ve Destek Vektör Makineleri	Doğruluk	[%63, %90]	Barnett-Itzhaki vd., 2020 ^[22]
canlı doğum	7188 (hasta)	hasta klinik verileri	Lojistik regresyon, Rasgele Orman, Destek Vektör Makineleri, XGBoost	AUC	XGBoost - 0.73	Qui vd., 2019 ^[23]

*ROC eğrisi altında kalan alan

Tüp bebek tedavisine başlamadan önce başarı olasılığını öğrenmek tedavi planlamada, psikolojik danışmanlık

hizmetlerinde ve maliyet hesaplamada karar desteği sağlar. Bu çalışmada tüp bebek tedavisine başlamadan önce bilinen öznitelikler olan hastanın yaş, vücut kitle indeksi (VKİ), primer/sekonder infertilite bilgisi ve infertilite teşhisi bilgileri

ile gebe kalabilme ihtimalini hesaplayan bir Klinik Karar Destek Sisteminin (KKDS) tedavi başarı oranının ne kadar doğrulukla tahmin edilebileceği analiz edilmiştir.

Literatürde şu ana kadar yapılan benzer çalışmalarda genellikle tedavinin ilerleyen aşamalarında embriyolar oluşturulduktan sonra tedavi sonucu tahminlemesi yapılmış ve tahmin performans ölçütü olarak yanlışlık oluşturabilecek doğruluk değeri kullanılmıştır. Bu çalışmada ise hasta kliniğe ilk geldiğinde henüz tedavi başlamadan başarı olasılığının belirlenmesi amaçlanmış ve iki sınıf sınıflandırma problemlerinde daha objektif performans değerlendirme kriteri olarak kabul edilen ROC eğrisi altında kalan alan (AUC) kriteri kullanılmıştır. Ayrıca, tahmin modeli oluşturma ve değerlendirmede kullanılan veri kümesi diğer pek çok çalışmadaki veri kümesinden daha fazla sayıda hasta kaydı içermektedir.

Çalışmanın bundan sonraki kısımları şu şekilde organize edilmiştir. Öncelikle araştırmada kullanılan veri kümesi ve veri üzerinde yapılan ön işleme sonucu çalışmada kullanılabilir hale getirilmiş ve veri kümesi açıklanmış, daha sonra hedeflenen klinik karar destek sistemi için uygulanan yöntemler anlatılmıştır. Son olarak çalışmanın sonucu ve bu alana sağladığı etkiler tartışılarak çalışma sonlandırılmıştır.

2. Yöntemler

2.1. Çalışmada Kullanılan Veri Kümesi

Yeditepe Üniversitesi Hastanesi Tüp Bebek Kliniği'nin elektronik sağlık kayıtları retrospektif olarak incelenmiştir. Çalışma Doğuş Üniversitesi Etik Kurulu tarafından 24 Aralık 2019 tarihinde gerçekleştirilen toplantıda onaylanmıştır (Sayı: 2019/25). Hastane tarafından sağlanan işlenmemiş veriler, Ocak 2013 ile Aralık 2019 arasında gerçekleştirilen 1154 adet embriyo transfer siklusunu içermektedir. Bu transferlerin 691 adedi negatif, 463 adedi pozitif yani gebe olarak sonuçlanmıştır.

Veri kümesi dört adet öznitelik içermektedir: yaş, vücut kitle indeksi, infertilite tanısı ve primer/sekonder infertilite bilgisi (Tablo 2). Her bir siklus kaydı, embriyo transferini takip eden sekizinci haftadaki ultrason sonucuna göre Pozitif (gebe) ya da Negatif sınıf etiketi ile belirtilmiştir. Sayısal özniteliklerin değer aralıkları farklılıklarını ortadan kaldırmak için değişkenlerin ortalaması 0, standart sapması 1 olacak şekilde Z-skora normalizasyonu uygulanmıştır.

Tablo 2. Öznitelikler

Öznitelik adı	Veri tipi
Primer/Sekonder Gebelik	Kategorik
İnfertilite Teşhisi	Kategorik
Yaş	Sayısal
VKİ*	Sayısal

*Vücut Kitle İndeksi

Tahmin modelleri oluşturulurken kullanılan veriyi tanımak uygun analiz yöntemlerini seçmek açısından önemlidir. Bu amaçla, özniteliklerin sınıflar üzerindeki dağılımları incelenmiştir.

ve bu dağılımlar sayısal değerler için kutu grafik, kategorik değerler için çubuk grafik kullanılarak sunulmuştur. Özniteliklerin birbirleri ile korelasyonu ise ısı haritası ile incelenmiştir. Örneklerin sınıflar arasındaki dağılımlarını ve kümelenmelerini incelemek için Temel Bileşen Analizi (TBA) yöntemi kullanılmıştır.

2.2. Sınıflandırma Algoritmaları

Sınıflandırma, denetimli öğrenme yöntemlerinden biridir. Daha önceden sınıf etiketi bilinen girdiler üzerinde eğitilen model ile sınıfı bilinmeyen girdilerin tahmin edilen sınıfa atanması yöntemine denir.

Çalışmada, en yüksek doğruluğu elde edebilmek amacıyla beş adet sınıflandırma yönteminin sonuçları karşılaştırılmıştır. Kullanılan sınıflandırma yöntemlerinden ilki Destek Vektör Makinelidir (SVM-Support Vector Machine). Bu algoritmanın amacı, N-boyutlu bir alanda veri noktalarını ayrı ayrı sınıflandıran bir hiper düzlem bulmaktır. Diğer bir yöntem olan çok katmanlı algılayıcı (MLP-Multi-Layer Perceptron), Yapay Sinir Ağı tabanlı bir yöntemdir. Birden fazla algılayıcıdan oluşmaktadır. Sinyali almak için bir giriş katmanı, giriş hakkında bir karar/tahmin yapan bir çıkış katmanı ve bu ikisi arasında çok katmanlı algılayıcının gerçek hesaplama motoru olan rastgele sayıda gizli katmandan oluşmaktadır. Çalışmada kullanılan diğer üç sınıflandırma yöntemi karar ağaçları tabanlı algoritmalar. Gradyan artırma sınıflandırma yöntemi olan Aşırı Gradyan Artırmada (XGBoost-Extreme Gradient Boosting) tahmin edilen ve hedef çıktılar arasındaki farka dayanarak oluşturulan kayıp fonksiyonunu ve model karmaşıklığı için bir ceza terimini birleştiren düzenli objektif bir fonksiyon kullanılır. Eğitim tekrarlanarak önceki ağaçların kalıntılarını veya hatalarını tahmin eden yeni ağaçlar eklenir ve daha sonra son tahmini yapmak için önceki ağaçlarla birleştirilir. Hafif Gradyan Artırma (LightGBM-Light Gradient Boosting) yönteminde ise ağaç, yaprak aşamalı olarak dikeyde büyür. Sınıflandırmada çok iyi performans sergilediği için sıklıkla tercih edilen rastgele orman (Random Forest) algoritmasında ise torbalama (bagging) ve rastgele öznitelik seçimi birlikte ele alınır. Rastgele öznitelik seçimi için öncelikle gerçek veri kümesinde yer değiştirmeli olarak yeni bir eğitim veri kümesi oluşturulur ve bu eğitim kümesi ile bir ağaç geliştirilir [21]. Geliştirilen ağaçlar budanmaz, bu nedenle diğer ağaç algoritmalarından daha avantajlıdır.

Sınıflandırma algoritmalarında hiper-parametre olarak adlandırılan model parametrelerinin optimum değerlerinin belirlenmesi, sınıflandırma performansının iyileştirilmesi açısından kritik önem taşımaktadır. Bu çalışmada kullanılan tüm sınıflandırma modelleri için hiper-parametre optimizasyonu uygulanmıştır.

2.3. ROC Analizi

Alıcı işlem karakteristikleri (Receiver Operating Characteristics - ROC) eğrileri iki-sınıf sınıflandırma problemlerinde doğru pozitif ve yanlış pozitif oranlarını kullanarak objektif ve tutarlı performans analizi sağlar. Bu

çalışmada da, gebelik tahmininde SVM sınıflandırıcının performans analizi için ROC eğrisi kullanılmıştır. İki sınıflı tahmin modellerinde gerçek sınıf değeri ve tahmin edilen değere bağlı olarak hesaplanan performans metrikleri Tablo 3'te gösterilmiştir.

Tablo 3: Hata Matrisi

Gerçek sınıf etiketi	Tahmin edilen sınıf	
	Pozitif	Negatif
Pozitif	DP	YN
Negatif	YP	DN

Tıbbi uygulamalarda kullanılan doğru pozitif oranı (DPO) ve yanlış pozitif oranı (YPO) gibi performans metrikleri yukarıdaki tablodaki değerler kullanılarak hesaplanabilir.

- **Doğru Pozitif Oranı (DPO)** pozitif örneklerin doğru sınıflandırma oranını belirten doğruluk değeridir. Gerçek sınıf etiketi ve tahmin edilen sınıf etiketi pozitif olan örnek sayısının veri kümesindeki toplam pozitif örnek sayısına bölünmesi ile hesaplanır.

$$DPO = (DP)/(DP + YN) \quad (1)$$

- **Yanlış Pozitif Oranı (YPO)** gerçek sınıf etiketi negatif olan örneklerin pozitif olarak tahmin edilme oranını belirten doğruluk değeridir ve aşağıdaki şekilde hesaplanır:

$$YPO = (YP)/(DN + YP) \quad (2)$$

SVM algoritmasında DPO ve YPO hesaplanırken sınıflandırma olasılık eşik değeri, t , 0.5 olarak kullanılır ve bu değer ROC eğrisinde tek bir noktaya karşılık gelir. Karar eşik değeri $[0, 1]$ aralığında değiştirilerek farklı DPO ve YPO değerleri elde edilebilir.

$$\begin{cases} P, & \text{if } p(C_p|x) \geq t \\ N, & \text{if } p(C_p|x) < t \end{cases} \quad (3)$$

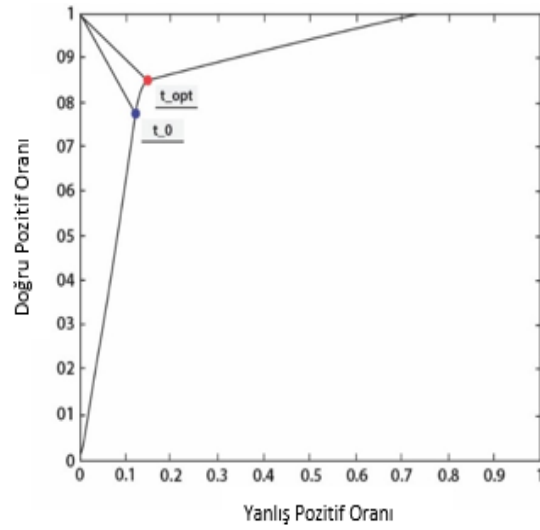
Eğer, x örneğinin pozitif sınıfa (P) ait olma olasılığı, $p(C_p|x)$, olasılık karar eşik değeri t 'den büyük veya eşit ise x örneği P sınıfına, değilse negatif (N) sınıfına atanır. Oluşan (DPO, YPO) değerlerinin iki boyutlu uzayda oluşturduğu ROC eğrisi değişen karar eşik değerleri için olası bütün sınıflandırma sonuçlarını gösterir.

Örnek bir ROC eğrisi Şekil 1'de gösterilmiştir. Şekildeki sol alt nokta (0,0) bütün örneklerin negatif sınıfa atanması sonucu DPO ve YPO oranlarının 0 olduğu durumu gösterir. Sağ üst nokta (1,1) ise bütün örneklerin pozitif olarak sınıflandırılmasına karşılık gelir. DPO değerinin 1, YPO değerinin 0 olduğu sol üst nokta (0,1) ise mükemmel sınıflandırma sonucu oluşur. ROC eğrisi üzerinde sol üst köşeye en yakın noktadaki DPO ve YPO değerleri ile mümkün

olan en iyi sınıflandırma sonucu elde edilmiş olur. Bu noktadaki karar eşik değeri optimum değer olarak kabul edilir ve (0,1) noktasına olan minimum Öklid uzaklığının hesaplanması ile belirlenir:

$$uzaklık = \sqrt{(1 - DPO)^2 + (0 - YPO)^2} \quad (4)$$

İkili sınıflandırıcıların sınıflandırma performansı değerlendirilmesi ve farklı sınıflandırıcıların karşılaştırılması için ROC eğrisi altında kalan alanı ifade eden AUC (Area Under the ROC Curve) metriği yaygın olarak kullanılır.



Şekil 1. ROC eğrisi örneği, ve varsayılan (t_0) ve optimum (t_{opt}) karar eşik değerleri

3. BULGULAR

3.1 Öznelik Dağılımları ve Korelasyonu

Veri kümesinde yer alan özneliklerin sınıflara göre dağılımları Şekil 2'de gösterilmiştir. Yaş özneliği gebe grubunda negatif gruba oranla istatistiksel anlamlı olarak daha düşüktür ($p < 0.05$). İnfertilite nedeni dağılımına göre negatif gebeliklerde ileri anne yaşı (AY) ve düşük over rezervi (DOR) daha yüksek oranlarda yer alırken, pozitif gebeliklerde açıklanamayan infertilite ve erkek faktör (Male) öne çıkmaktadır. VKİ ve primer/sekonder infertilite bilgisi gruplar arasında farklılık göstermemektedir.

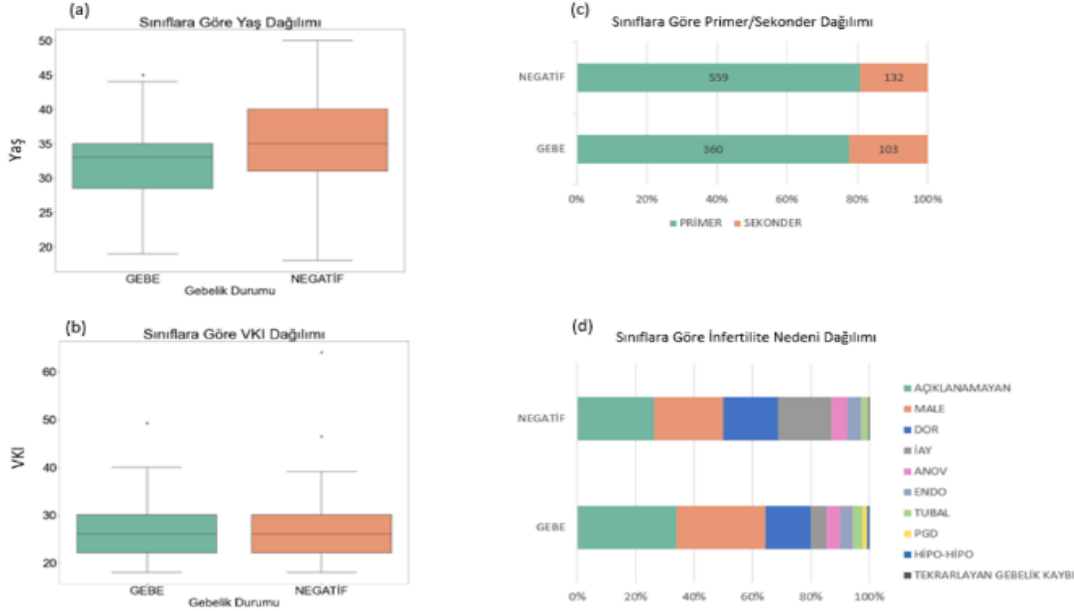
Öznelikler arasındaki korelasyon Sekil 3'te ısı haritası şeklinde sunulmuştur. Buna göre, yaş ile infertilite nedeni arasında yüksek korelasyon olduğu gözlemlenmiştir.

3.2 Temel bileşen analizi

Sekil 4'teki PCA grafiğinde dört öznelikten oluşan veri kümesi iki temel bileşen düzleminde gösterilmiştir. İki temel bileşen veri kümesindeki varyasyonun %64.4'ünü ifade

etmektedir. Örneklerin sınıflar üzerindeki dağılımı ve sınıfların ayrışması incelendiğinde, gebe ve negatif sınıflar arasında net

bir ayrışma olmadığı, gebelik tahmini probleminin göreceli olarak karmaşık bir problem olduğu görülmektedir.



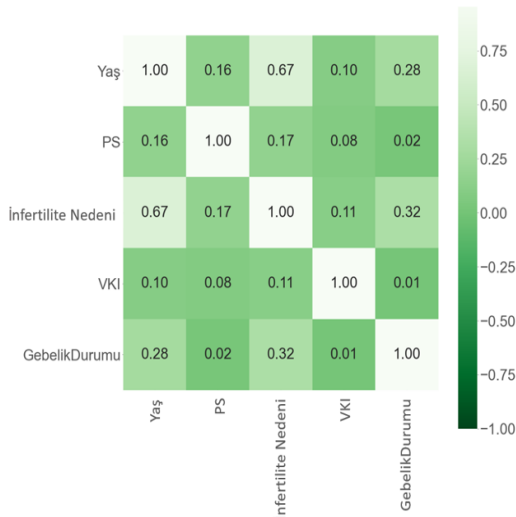
Şekil 2. Özniteliklerin sınıflara göre dağılımları

*Male (Erkek faktör), DOR (Düşük Over Rezervi), İAY (İleri Anne Yaşı), ANOV (Anovulasyon), ENDO (Endometriozis), PGD (Preimplantasyon Genetik Yapı), HİPO-HİPO (hipogonadotropik hipogonadizm)

3.3 Sınıflandırma Algoritmalarının Karşılaştırılması

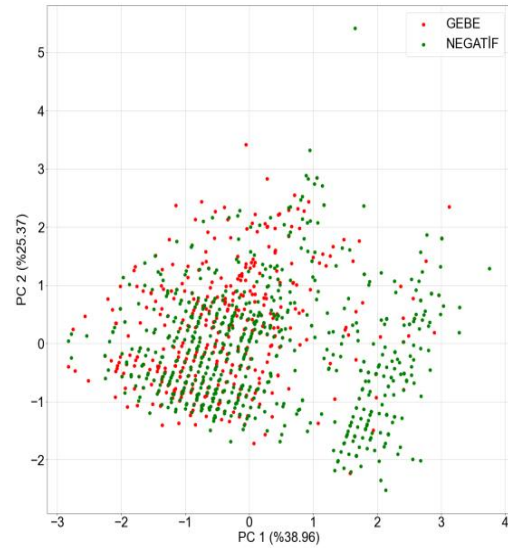
Gebelik sonucu tahmininde kullanılan sınıflandırma algoritmalarının 5-katlı çapraz doğrulama sonuçlarına göre SVM yöntemi ortalama 0.70 AUC değeri ile en yüksek sınıflandırma performansını sergilemiştir (Tablo4).

Hiper-parametre optimizasyonu sonrası tüm sınıflandırma modellerinde doğruluk oranlarının yükseldiği gözlemlenmiştir. Sınıflandırma modelleri için kullanılan varsayılan (default) değerler ve optimum değerler Tablo 5'te gösterilmiştir.



Şekil 3. Öznitelik korelasyon ısı haritası

*PS (Primer/Sekonder Infertility), VKİ (Vücut Kitle İndeksi)



Şekil 4. Veri kümesi temel bileşen analizi

*PC(Principal Component-Temel Bileşen)

Tablo 4. Sınıflandırma AUC sonuçları

Sınıflandırma algoritması	Varsayılan parametreler ile AUC	Optimizasyon sonrası AUC
Destek Vektör Makineleri	0.68 (±2.59)	0.70 (±2.09)
Çok katmanlı algılayıcı	0.69 (±2.53)	0.69 (±4.06)

Aşırı Gradyan Artırma	0.58 (± 2.56)	0.56 (± 2.74)
Hafif Gradyan Artırma	0.60 (± 3.24)	0.67 (± 2.16)
Rasgele Orman	0.57 (± 3.37)	0.67 (± 2.27)

Tablo 5. Sınıflandırma algoritmalarının varsayılan ve optimum hiper-parametreleri

Sınıflandırma modeli	Varsayılan değerler	Optimum değerler
SVM	C=1, kernel='rbf', probability=True	C=1, kernel='linear', probability=True
MLP	activation='relu', alpha=0.0001, hidden_layer_sizes=(100), solver='adam'	activation='tanh', alpha=0.05, hidden_layer_sizes=(20), solver='sgd'
XGBoost	colsample_bytree=1, max_depth=3, n_estimators=100, reg_alpha=0, reg_lambda=1, subsample=1	colsample_bytree=0.7, max_depth=15, n_estimators=400, reg_alpha=1.2, reg_lambda=1.2, subsample=0.8
LightGBM	colsample_bytree=1, max_depth=-1, min_split_gain=0, n_estimators=100, num_leaves=31, reg_alpha=0, reg_lambda=0, subsample=0.8, subsample_freq=0	colsample_bytree=0.8, max_depth=15, min_split_gain=0.3, n_estimators=400, num_leaves=50, reg_alpha=1.1, reg_lambda=1.3, subsample=0.8, subsample_freq=20
RandomForest	criterion='gini', max_depth=None, max_features='auto', n_estimators=100, verbose=0	max_depth=10, max_features='log2', min_samples_leaf=5, min_samples_split=3, n_estimators=75, verbose=False

3.4 Karar Eşik Değeri Optimizasyonu

ROC eğrisi üzerinde mükemmel sınıflandırma sonucu olan sol üst köşeye en yakın DPO ve YPO değerleri 0.63 karar eşik değeri ile elde edilmiştir. Optimum karar eşik değeri kullanılarak gerçekleştirilen sınıflandırma sonucu oluşan hata matrisine göre en yüksek DPO %71.7 ve en yüksek YPO %59.4 olarak hesaplanmıştır (Tablo 6).

Tablo 6. SVM Sınıflandırma algoritması hata matrisi sonuçları

		Tahminlenen Değerler	
		Gebe	Negatif
Gerçek Değerler	Gebe	66	26
	Negatif	56	82

4. SONUÇ

Bu çalışmada, tüp bebek tedavi sürecinde hastaların tedaviye başlamadan önceki verilerini kullanarak pozitif gebelik olasılığını hesaplayan bir model geliştirilmiştir. Model

oluşturulurken beş farklı sınıflandırma algoritması, 5-katlı çapraz doğrulama yöntemi ile test edilmiştir. En yüksek sınıflandırma performansı SVM yöntemi ile edilmiş (AUC=0.70) ve sınıflandırma olasılıklarında karar eşik değerinin optimizasyonu ile en yüksek DPO %71.7 olarak hesaplanmıştır.

Literatürde benzer çalışmalarda yaş, VKİ, önceki gebelik sayısı, infertilite süresi ve önceki tedavi girişimi sayısı gibi tedavi öncesi belirlenebilen öznitelikler kullanılarak gebelik sonucu tahmininde 0.63-0.73 arasında AUC değerleri elde edildiği raporlanmıştır [22, 23]. Fakat bu çalışmaların bazılarında örnekleme yanlılığının yüksek olduğu çok küçük veri kümeleri kullanılmış [22], ya da çapraz doğrulama yöntemi kullanılmadan veri kümesi sadece bir defa eğitim ve test kümesi olarak ayrılmıştır [23]. Ayrıca, çalışmaların çoğunda performans kriteri olarak sadece doğruluk sonucu sunulmuş ve doğruluk değeri olasılık karar eşik değeri 0.5 varsayılan değer ile hesaplanmıştır. Tüp bebek tedavisi gibi negatif sonuçların pozitif sonuçlardan daha fazla olduğu dengesiz sınıf dağılımı problemlerinde doğruluk yeterli bir performans kriteri değildir. Bizim çalışmamızda ise tahmin modeli eğitmek için yeterli büyüklükte bir veri kümesi kullanılmış, çapraz geçerlilik yöntemi uygulanmış ve karar eşik değeri optimizasyonu ile DPO ve YPO arasında ideal denge sağlanmıştır.

Tedavi öncesi gebelik tahmini konusunda yapılan çalışmalarda elde edilen performans sonuçları kullanılan veri kümesine, veri kümesine uygulanan yöntemlere, model optimizasyonu yöntemlerine bağlı olarak farklılık gösterse de belli bir oranın üzerine çıkmadığı da gözlemlenmektedir. Diğer taraftan gebelik tahmini konusunda yapılan ilk yapay zeka tabanlı çalışmadan [14] bu yana sınıflandırma performansları büyük bir aşama kaydetmiştir. Elde edilen bu olumlu sonuçlara rağmen yapay zeka yöntemleri kullanılarak geliştirilen klinik karar destek sistemleri kullanılırken tüp bebek tedavi sürecinin karmaşıklığı sebebiyle tedavi başarısını etkileyen farklı birçok faktör bulunduğu da göz ardı edilmemelidir.

Kaynaklar

- [1] R. Bhardwaj, A. R. Nambiar, D. Dutta, "A Study of Machine Learning in Healthcare", IEEE 41st Annual Computer Software and Applications Conference (COMPSAC), Turin, 236-241, July 2017.
- [2] W. Liu, P. Shen, Y. Qu, D. Xia, "Fast algorithm of support vector machines in lung cancer diagnosis", International Workshop on Medical Imaging and Augmented Reality, Hong Kong, 188-192, 2001.
- [3] S.M., McKinney, M., Sieniek, V. Godbole, et al. "International evaluation of an AI system for breast cancer screening", Nature, 577, 89-94, 2020.
- [4] F. Jiang et al., "Artificial intelligence in healthcare: Past, present and future", Stroke Vasc. Neurol., 2(1), 230-243, 2017.
- [5] P.C. Steptoe, R.G. Edwards, "Birth after the reimplantation of a human embryo", Lancet, 2, 366, 1978.
- [6] M. Vander Borcht, C. Wyns, "Fertility and infertility: Definition and epidemiology," Clinical Biochemistry, 62, 2018.

- [7] R. G. Edwards and P. C. Steptoe, "Induction of follicular growth, ovulation and luteinization in the human ovary.," *J. Reprod. Fertil. Suppl.*, 1975.
- [8] M. E. Cohen, "The 'brave new baby' and the law: fashioning remedies for the victims of in vitro fertilization.," *Am. J. Law Med.*, 1978.
- [9] M. R. Sadeghi, "The 40th anniversary of IVF: Has ART's success reached its peak?," *Journal of Reproduction and Infertility*, 2018.
- [10] B. Mesko, "The role of artificial intelligence in precision medicine," *Expert Review of Precision Medicine and Drug Development*, 2(5), 239-241, 2017.
- [11] A. Uyar, Y. Sengul, A. Bener, "Emerging technologies for improving embryo selection: a systematic review," *Adv. Health Care Technol.*, 1, 55-64, 2015.
- [12] L. Bori, E. Paya, L. Alegre, et al., "Novel and conventional embryo parameters as input data for artificial neural networks: an artificial intelligence model applied for prediction of the implantation potential", *Fertility and Sterility*, 114(6), 1232 -1241, 2020.
- [13] G. Letterie, A. Mac Donald, "Artificial intelligence in in vitro fertilization: a computer decision support system for day-to-day management of ovarian stimulation during in vitro fertilization", *Fertility and Sterility*, 114(5), 1026-1031, 2020.
- [14] S. J. Kaufmann, J. L. Eastaugh, S. Snowden, S. W. Smye, and V. Sharma, "The application of neural networks in predicting the outcome of in-vitro fertilization," *Hum. Reprod.*, 12(7), 1454-1457, 1997.
- [15] P. Vogiatzi, A. Pouliakis, C. Siristatidis, "An artificial neural network for the prediction of assisted reproduction outcome", *J Assist Reprod Genet*, 36(7), 1441-1448, 2019.
- [16] A. La Marca *et al.*, "Anti-Müllerian hormone-based prediction model for a live birth in assisted reproduction," *Reprod. Biomed. Online*, 22(4), 341-349, 2011.
- [17] W. J. Yi, K. S. Park, J. S. Paick, "Morphological classification of sperm heads using artificial neural networks", *Studies in Health Technology and Informatics*, 52, 1071-1074, 1998.
- [18] D. A. Morales *et al.*, "Bayesian classification for the selection of in vitro human embryos using morphological and clinical data", *Comput. Methods Programs Biomed.*, 90(2), 104-116, 2008.
- [19] P. Burai, A. Hajdu, F. R. E. Manuel, B. Harangi, "Segmentation of the uterine wall by an ensemble of fully convolutional neural networks", *40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, Honolulu, HI, 49-52, 2018.
- [20] A. Uyar, A. Bener, H.N. Ciray, "Predictive Modeling of Implantation Outcome in an In Vitro Fertilization Setting: An Application of Machine Learning Methods", *Med Decis Making*, 35(6), 714-725, 2015.
- [21] L. Breiman, "Random forests," *Mach. Learn.*, 45, 5-32, 2001.
- [22] Z. Barnett-Itzhaki *et al.*, "Machine learning vs. classic statistics for the prediction of IVF outcomes", *J Assist Reprod Genet*, 37(10), 2405-2412, 2020.
- [23] J. Qui *et al.* "Personalized prediction of live birth prior to the first in vitro fertilization treatment: a machine learning method", *J Transl Med*, 17, 317-324, 2019.

Özgeçmişler



Prof. Dr. Rukset Attar, MD, PhD

04.01.1968 yılında Tarsus'ta doğdu. 1979 yılında Moda İlkokulu'ndan mezun oldu. 1986 yılında Kadıköy Anadolu Lisesi'nden mezun oldu. 1986-1992 yılları arasında İstanbul Üniversitesi Tıp Fakültesine okudu. 1992 yılında mezun olduktan sonra Sağlık Bakanlığı Zeynep Kamil Kadın ve Çocuk Hastalıkları Eğitim ve Araştırma Hastanesi'nde kadın hastalıkları ve doğum ihtisası yaptı. 1994-1995 yılları arasında Amerika Birleşik Devletleri'nde Yale Üniversitesi Tıp Fakültesi Kadın Hastalıkları ve Doğum Anabilim Dalı Üreme Endokrinolojisi Bölümünde Araştırma görevlisi olarak çalıştı. 1997 yılında TEKEL Sağlık Merkezi ve Reviri Başhekimliği'nde uzman doktor olarak göreve başladı. 1997-2007 yılları arasında TEKEL Sağlık Merkezi ve Reviri Başhekimliği'nde başhekim olarak görev yaptı. 1998-2004 yılları arasında İstanbul Üniversitesi DETAE Moleküler Tıp Anabilim Dalı'nda doktora yaptı. 2007 yılında Yeditepe Üniversitesi Kadın Hastalıkları ve Doğum Anabilim Dalı'nda yardımcı doçent doktor olarak görev yapmaya başladı. 2012 yılında Kadın Hastalıkları ve Doğum doçenti ünvanını ve 2017 yılında da profesör ünvanını aldı. 2014-2016 yılları arasında İstanbul AREL Üniversitesi AREL UZEM Sağlık Kurumları İşletmeciliği yüksek lisans programını bitirdi. 2008 yılında Yeditepe Üniversitesi Tüp Bebek Ünitesi'nden Yardımcı Üreme Teknikleri (Tüp Bebek) sertifikası aldı. 2018 yılından itibaren Yeditepe Üniversitesi Tüp Bebek Ünitesi Sorumlu Hekim olarak görev yapıyor. Halen Yeditepe Üniversitesi Tıp Fakültesi Kadın Hastalıkları ve Doğum Anabilim Dalı'nda, Yeditepe Üniversitesi Tıp Fakültesi Moleküler Tıp Anabilim Dalı'nda ve Yeditepe Üniversitesi Tıp Fakültesi Histoloji ve Embriyoloji Anabilim Dalı'nda görev yapıyor. Yeditepe Üniversitesi'nde akademik performans dördüncülük ödülü ve daha sonra ikincilik ödülü aldı. Tıp Fakültesinde , Moleküler Tıp Anabilim Dalı'nda ve Histoloji ve Embriyoloji Anabilim Dalı'nda yüksek lisans ve doktora dersleri vermekte.



Yasemin ATILGAN ŞENGÜL, lisans ve yüksek lisans derecelerini sırasıyla 1999 ve 2002 yıllarında, doktora derecesini 2010 yılında Marmara Üniversitesi Bilişim Alanında bitirdi. 2000 yılında Doğuş Üniversitesi Bilgisayar Mühendisliği Bölümünde Araştırma Görevlisi olarak göreve başladı. 2014 yılında aynı üniversitede Dr. Öğr. Üyesi olarak atandı. 2018 yılında Meslek Yüksekokulu Müdür Yardımcılığı görevine başladı. Doğuş Üniversitesi Meslek Yüksekokulunda Müdür olarak görevine devam etmektedir.



Oya Alagöz 1973 yılında Bursa'da doğdu. 1990 yılında Çamlıca Kız Lisesinden mezun oldu. 1996 yılında İstanbul Üniversitesi Tıp Fakültesini bitirdi. İ. Ü. İstanbul Tıp Fakültesi Histoloji ve Embriyoloji Anabilim Dalı'nda 2001 yılında Uzmanlık Eğitimini tamamladı. 2002-2003 yılları arasında Hattat Hastanesi ÜYTM Laboratuvar Direktörü olarak görev yaptı. 2004-2005 yılları arasında İstanbul Memorial Hastanesi Tüp Bebek ve Genetik Merkezi Ar-Ge Laboratuvarı (Proje Sorumlusu) ve Üremeye Yardımcı Tedavi Yöntemleri Kursu Eğitim Sorumlusu olarak görev yaptı. 2005 yılından beri Yeditepe Üniversitesi Hastanesi (Histoloji ve Embriyoloji Anabilim Dalı Öğretim Görevlisi) ÜYTM Lab. Direktörü olarak görev yapmaktadır.



Zeynep Pacci, 2005 yılında Okan Üniversitesi Bilgisayar Mühendisliği bölümünden mezun olmuştur. Yüksek lisans derecesini Okan Üniversitesi Bilgisayar Mühendisliği bölümünden 2019 yılında almıştır. 2012 yılından beri Türkiye İş Bankası Bilgi Teknolojileri Bölümünde çalışmaktadır. Akademik araştırma konuları arasında yapay zeka destekli karar destek sistemleri yer almaktadır.



Aslı UYAR, lisans eğitimini İstanbul Teknik Üniversitesi'nde 2003 yılında, yüksek lisans ve doktora eğitimini Bogazici Üniversitesi Bilgisayar Mühendisliği Bölümü'nde 2006 ve 2011 yıllarında tamamlamıştır. Lisansüstü çalışmaları Makine Öğrenmesi tabanlı klinik karar destek sistemleri konusunda yoğunlaşmıştır. 2011-2014 yılları arasında doktora sonrası araştırmacı olarak Yale Üniversitesi'nde biyoinformatik alanında çalışmalar yapmıştır. Şu an, Okan Üniversitesi Bilgisayar Mühendisliği Bölümü'nde öğretim üyesi ve Jackson Laboratuvarı'nda araştırmacı bilim insanı olarak akademik ve bilimsel çalışmalarına devam etmektedir. Başlıca araştırma konuları, ileri biyoinformatik ve makine öğrenmesi yöntemlerinin genomik tıp alanında uygulanmasını içerir. Dr. Uyar'ın uluslararası bilimsel dergilerde yayınlanan 30'dan fazla makalesi bulunmaktadır. Kendisi aynı zamanda uluslararası yüksek etki faktörlü dergilerde editör ve hakem olarak görev almaktadır.

Güneş Enerjisinin Kısa-Dönem Tahmininde Farklı Makine Öğrenme Yöntemlerinin Karşılaştırılması

Comparison of different machine learning methods in short-term forecasting of solar energy

M. Fatih Bekçioğulları^{*1}, Bünyamin Dikici¹, Hakan Açıkgöz¹, Ö. Fatih Keçecioglu²

¹Gaziantep İslam Bilim ve Teknoloji Üniversitesi, Mühendislik ve Doğa Bilimleri Fakültesi, Elektrik-Elektronik Mühendisliği Bölümü, Gaziantep

mehmetfatih.bekciogullari@gibtu.edu.tr, bunyamin.dikici@gibtu.edu.tr, hakan.acikgoz@gibtu.edu.tr

²Kahramanmaraş Sütçü İmam Üniversitesi, Mühendislik ve Mimarlık Fakültesi, Elektrik-Elektronik Mühendisliği Bölümü, Kahramanmaraş

fkececioglu@ksu.edu.tr

Öz

Günümüzde, güneş enerji santrallerine yapılan yatırımlar her geçen gün kayda değer derecede artış göstermektedir. Güneş enerjisinin meteorolojik parametrelere bağlı olarak değişken olması enerji üretiminin planlanmasını nispeten zorlaştırmaktadır. Bu durum şebeke işletme problemlerine de yol açabileceğinden güneş enerjisi tahmini önemli görülmektedir. Bu çalışmada, makine öğrenmesi metodlarının kısa-dönemli güneş enerjisi tahmini yapabilirliği karşılaştırılmıştır. İlk olarak, Avustralya, Alice Springs’de bulunan bir güneş enerji merkezindeki 1B: Trina isimli santralin çıkış gücü verisi ve bölgedeki ölçüm istasyonundan alınan meteorolojik parametrelerden oluşan bir veri seti elde edilmiştir. Daha sonra güç verisi Ampirik mod ayrıştırma yöntemi ile alt bileşenlerine ayrılmıştır. Bu çalışmada, güneş enerjisinin tahmini için Doğrusal Regresyon (DR), Destek Vektör Makinesi (DVM), Karar Ağacı Regresyonu (KAR), Gauss Süreç Regresyonu (GSR) ve Topluluk Regresyonu (TR) gibi en çok kullanılan yöntemler tercih edilmiştir. Bu yöntemlerin farklı tahmin ufuklarındaki tahmin performanslarını değerlendirmek için karşılaştırma çalışmaları gerçekleştirilmiştir. Bu çalışmalarda, tüm makine öğrenme yöntemleri için R, RMSE ve MAE gibi performans metrik sonuçları hesaplanmıştır. Elde edilen metrik sonuçlarına göre DVM’nin en iyi tahmin sonucunu sağladığı gözlemlenmiştir.

Anahtar Kelimeler: Ampirik mod ayrıştırıcı, Güneş Enerjisi, Makine Öğrenmesi, Tahmin

Abstract

Nowadays, the investment on Photovoltaic (PV) power generation is increasing significantly day by day. The fact that solar energy is variable depending on meteorological parameters which makes it relatively difficult to plan energy generation. Since this problem may also lead to grid operations problems, solar energy forecasting is

considered important. In this study, the short-term solar energy forecasting capability of machine learning methods are compared. Firstly, a data set consisting of the output power data of the power plant named 1B: Trina in a solar power center in Alice Springs, Australia, and meteorological parameters taken from the measurement station in the region are obtained. Then, the power data is divided into sub-components with the empirical mode decomposition method. In this study, the most used methods such as Linear Regression (LR), Support Vector Machine (SVM), Decision Tree Regression (DTR), Gaussian Process Regression (GPR) and Ensemble Regression (ER) are preferred for solar energy forecasting. The comparison studies are carried out to evaluate the forecasting performances of these methods in different forecasting horizons. In these comparison studies, performance metric results such as R, RMSE and MAE are calculated for all machine learning methods. According to the metric results obtained, it has been observed that SVM provides the best forecasting result.

Keywords: Empirical mode decomposition, Solar energy, Machine learning, Forecast

1. Giriş

Günümüzün en büyük problemlerden biri olarak görülen küresel ısınmanın çözümü için bilim insanları arayış içindedir. Artan insan nüfusu ve sanayileşme ile birlikte fosil yakıtlar küresel ısınmanın en önemli nedenlerinden bir tanesidir. İnsan neslinin daha temiz ve yaşanabilir bir dünyada yaşayabilmesi için tüm dünyada enerji ihtiyacı için yenilenebilir enerji temelli santraller kurulmaya hızla devam edilmektedir. Bu enerji kaynaklarından biri olan Güneş enerjisi ise son yıllarda gittikçe artan bir ilgiye sahiptir. Tüm dünyada olduğu gibi ülkemizde de devletimizin verdiği teşviklerin de etkisiyle güneş enerjisinden elektrik üretiminde son yıllarda büyük sıçrama yaşanmıştır [1-3].

Güneş enerjisinin meteorolojik parametrelere bağlı olarak değişim gösterdiği bilinen bir gerçektir. Güneş radyasyonu, sıcaklık, nem, rüzgâr hızı ve bulutluluk gibi meteorolojik parametreler güneş enerjisinden elektrik üretimini doğrudan veya dolaylı olarak etkilemektedir. Meteorolojik değişimler güneş enerjisinde ani dalgalanmalara sebep olduğundan bu enerji ile devamlık gösteren enerji üretimi mümkün olmamaktadır. Ayrıca güneş enerjisinin belirsiz olması ve gün içinde meteorolojik parametrelere bağlı olarak dalgalanma göstermesi, enerjinin verildiği şebekede ani gerilim düşümlerine sebep olabileceği için kısa dönemli enerji tahmini önemli bir konu haline gelmiştir. Kısa dönemli yapılacak olan enerji tahmini ile güneş enerjisi dalgalanmasını kompanze ederek şebekede olabilecek işletme problemleri de böylelikle çözülebilmektedir. Güvenilir enerji tahmini ile üretilecek enerjinin öngörülebilir olması aynı zamanda enerji piyasasında enerjinin makul fiyatlarda pazarlanmasının önünü açmaktadır [4-6].

Son yıllarda Güneş enerji tahmini üzerine literatürde birçok çalışma yapılmıştır. Demolli vd. [7], Destek Vektör Makinesi (DVM), Lasso Regresyon ve kNN algoritmalarını kullanarak Niğde İli'nin güneş enerjisini tahmin etmişlerdir. Bu yöntemler birkaç performans metriği açısından karşılaştırılmış ve genel bir değerlendirme yapılmıştır. Elde edilen sonuçlardan DVM'nin daha iyi tahmin sonuçlarına sahip olduğu belirtilmiştir. Gök vd. [8] çalışmalarında Yapay Sinir Ağlarını (YSA) kullanmışlardır. Veri seti olarak güneş enerji santralinin 1 yıllık geçmiş güneş enerji verisi ile küresel tahmin sisteminden alınan bulutluluk verisi kullanılmıştır. 1 gün sonrası için güneş enerjisi tahmini yapılmıştır. Bulutluluk verisi kullanılan ağ ile kullanılmayan ağın karşılaştırıldığı bu çalışmada bulutluluk verisi kullanılan ağın tüm performans metrik sonuçlarında daha iyi performans sergilediği görülmüştür. Wolff vd. [9], güneş enerjisi tahmini için DVM tabanlı melez bir metot önermişlerdir. Veri setinde 15 dakika aralıklarla ölçülmüş güneş enerjisi çıkış gücü ve aynı aralıkla ölçülmüş sıcaklık verisi ile bulutluluk verisi kullanılmıştır. Optimize edilmiş DVM metodu ile alınan sonuçlar fiziksel yaklaşım tabanlı tahmin yöntemleri ile karşılaştırılmış ve elde edilen tüm metrik sonuçları kapsamlı bir şekilde değerlendirilmiştir. Mahmud vd. [10], coğrafi olarak güneş enerjisi bakımından en zengin yerlerden birisi olduğu belirtilen Avustralya'nın Alice Springs bölgesinde bulunan santralden elde edilen güneş enerjisinin tahmini için farklı makine öğrenme algoritmalarını kullanmışlardır. Veri seti olarak güneş enerjisi gücü ile birlikte bağıl nem, sıcaklık, radyasyon ve günlük yağış miktarı gibi meteorolojik parametreler kullanılmıştır. Tüm performans metrik sonuçlarından Rastgele Orman Algoritması (ROA) yönteminin en iyi sonuçları sağladığı görülmüştür. Massaoudi vd. [11], güneş enerji tahmini için doğrusal olmayan dışsal girdili otoregresif ağ ve Uzun-Kısa Süreli Bellek (UKSB) modeli tabanlı melez bir yöntem önermişlerdir. Tahmin ufku 1 ile 24 saat ilerisi olarak seçilmiş ve karşılaştırma çalışmaları yapılmıştır. Elde edilen sonuçlar ise birçok metrik bakımından incelenmiştir. Korkmaz vd. [12], güneş enerjisi tahmini için derin öğrenme tabanlı bir yöntem geliştirmişlerdir. İlk olarak santralin çıkışından elde edilen güç verileri Ampirik mod ayrıştırma yöntemi ile bileşenlerine ayrılmıştır. Daha sonra ise veri setindeki tüm girişler resme dönüştürülmüştür. Önerilen derin öğrenme yöntemin tahmin performansı regresyon metotları ile birçok yönden karşılaştırılmıştır. Yıldız vd. [13], kernel aşırı öğrenme makinesi temelli bir metodu Kilis ilinde bulunan

bir güneş enerji santralinin çıkış gücünü tahmin etmek için önermişlerdir. Önerilen tahmin metodunun performansını incelemek için detaylı karşılaştırma çalışmaları gerçekleştirilmiştir. Uğuz vd. [14] çalışmalarında 125 farklı bölgede kurulması planlanan güneş enerji santrallerinin çıkış gücünün farklı makine öğrenmesi metotları ile tahmin edilebilirliğini araştırmışlardır. Bu bölgelerden alınan rakım sıcaklık ve güneşlenme süresi gibi yedi farklı parametre kullanarak veri seti oluşturulmuştur. Bu veri seti ile farklı makine öğrenmesi metotlarından sonuçlar alınmıştır. Performans değerlendirme sonuçları YSA modelinin güneş enerjisinin tahmininde daha güvenilir sonuçlar sağladığını göstermiştir. Özdemir vd. [15] çalışmalarında YSA ve çift yönlü UKSB metotlarıyla güneş enerjisi güç tahmini yapmışlardır.

Bu çalışmada, güneş enerji tahmininde farklı regresyon yöntemlerinin performansları detaylı bir şekilde incelenmiştir. İlk olarak 2018-2019 yıllarını kapsayan bir veri seti Avusturalya'da bulunan Alice Spring bölgesinden toplanmıştır. Güneş enerji santralinin çıkış gücü ilk olarak Ampirik mod ayrıştırma yöntemi ile bileşenlerine ayrılmıştır. Bu bileşenler ve diğer meteorolojik verilerden 1-3 saat ilerisi için tahmin çalışmaları gerçekleştirilmiştir. Bu çalışmanın diğer kısımları ise şu şekilde kısaca özetlenebilir. 2. kısımda makine öğrenme yöntemlerine ait genel bilgiler verilmiştir. 3. kısımda Ampirik mod ayrıştırma yöntemi sunulmuştur. Performans değerlendirme kriterleri ise 4. kısımda verilmiştir. 5. kısımda tüm makine öğrenme yöntemleri için deneysel çalışmalar gerçekleştirilmiştir. Son kısımda ise elde edilen sonuçlar değerlendirilmiştir.

2. Makine Öğrenme Yöntemleri

2.1. Doğrusal regresyon

Doğrusal Regresyon (DR) yöntemi bağımsız değişken (X) ile bağımlı değişken (Y) arasındaki doğrusal ilişkiyi ortaya koymaktadır. Ayrıca bu yöntem bağımsız ve bağımlı değişkenlerin ilişkisinin yönünü de pozitif veya negatif olarak belirtmektedir. Bu model çıkış olarak değişkenler arasındaki ilişkiyi kullanarak belirlenen zaman ufku tahmin sonucu vermektedir.[16]. Bağımsız değişkenin değeri bilindiği takdirde bağımlı değişkenin değeri aşağıdaki denklemde gösterildiği gibi belirlenebilir:

$$f(x) = ax + b \quad (1)$$

$$a = \frac{\sum_{i=1}^n (x_i - \bar{x})(f(x_i) - f(\bar{x}))}{\sum_{i=1}^n (x_i - \bar{x})^2} \quad (2)$$

$$b = f(\bar{x}) - a\bar{x} \quad (3)$$

Burada, a regresyon katsayısını ifade eder. b ise sabit bir sayıdır.

2.2. Destek vektör makinesi

DVM, yapısal risk minimizasyonu ilkesi ve istatistiksel öğrenme teorisine dayanarak geliştirilmiş olan, regresyon problemlerinin çözülmesinde kullanılan makine öğrenmesi yöntemidir. DVM eğitim hatasını minimize etmek ve yapay zekada karşılaşılan yerel optimumun içine düşmekten kaçınmak için deneysel risk minimizasyonu prensibi yerine yapısal risk minimizasyonu prensibini kullanmaktadır[17]. Ayrıca bu makine öğrenmesi yöntemi eğitim verisini doğrusal olmayan bir şekilde düşük boyutlu düzlemde

yüksek boyutlu uzaya haritalayabilme olanağı sunmaktadır. DVM'nin yukarıda bahsedilen karakteristiklerine dayanarak geleneksel tahmin modellerinin eksiklik gösterdiği noktaların üstesinden etkili bir şekilde gelebildiğini söylemek mümkündür. Doğrusal olmayan regresyon modelleri arasında gösterilen DVM yöntemi durağan varsayımına gerek olmadan tahmin gerektiren problemlerin çözümünde esneklik sağlayabilmektedir[18]. DVM hata değerinin olması gereken maksimum değerini belirlemeye izin vermekte ve veri setine uygun olan en iyi modelin bulunmasına olanak sağlamaktadır. Aşağıda, DVM için genel bir ifade verilmiştir.

$$y_t = a + w_t x_t + \varepsilon_t = 1, \dots, T \quad (4)$$

Burada, y_i çıkışı (tahmin edilen değer), a sınırları çizilen kısmı, w_i ağırlığı, x_i girişi (tahmin edici veriler) ve ε_i hata terimini ifade eder. Karesel hataların toplamını DVM gibi doğrusal modeller yardımıyla en aza indirilebilmektedir.

$$\min \sum_{t=1}^T (y_t - a - w_t x_t)^2 \quad (5)$$

DVM hata büyüklüğünü belirleme konusunda esneklik sağlayabilmektedir. Amaç fonksiyonu aşağıdaki gibi tanımlanabilir:

$$\min \frac{1}{2} \|w\|^2 \quad (6)$$

$$|y_t - a + w_t x_t| < \varepsilon_t \quad (7)$$

Amaç fonksiyonunu mümkün olduğunca minimize etmek için sapmaları da ilgili denkleme ekleyerek aşağıdaki formül elde edilebilir.

$$\min \frac{1}{2} \|w\|^2 + C \sum_{t=1}^T |\xi_t| \quad (8)$$

$$|y_t - a + w_t x_t| < \varepsilon + |\xi_t| \quad (9)$$

Doğrusal olmayan DVM için kullanılan fonksiyon olan radyal kernel fonksiyonu yaygın olarak şu şekilde ifade edilir:

$$\exp\left(-\frac{\gamma}{2} \|x_t - x_t\|^2\right) \quad (10)$$

Burada, γ verideki değişikliği kontrol eden hassas bir parametredir.

2.3. Karar ağacı regresyonu

Karar ağacı regresyonu (KAR) makine öğrenmesinde ve görüntü işlemede yaygın olarak kullanılan metodların başında gelmektedir. KAR'ın anlaşılabilir konfigürasyon sağlamasından dolayı kullanıcıya kolaylık sağladığı bilinmektedir. KAR değişkenler arasındaki bağlantıyı göstermek için basit grafik yapılarını bünyesinde bulundurur. Bu özellik ise diğer makine öğrenme metodlarından birisi olan yapay sinir ağlarının belirsiz içyapısı ve sonuçlarının grafiğe dökülebilmesi yönüyle karşılaştırıldığında KAR'ın daha avantajlı olduğunu göstermektedir. Kavramsal kurallar YSA'daki gibi sayısal ağırlıklar olmadığından daha kolay bir şekilde inşa edilir. KAR yapısı düğümler ve dallardan oluşan bir ağaç modeline benzetilebilir. Her bir düğüm sınıflandırılacak kategorinin içindeki bir özelliği temsil ederken her bir alt bileşen de düğümün alabileceği değeri tanımlar [19-20].

2.4. Topluluk regresyonu

Topluluk regresyonu (TR), çoklu karar ağaçlarının toplanmasıyla tahmin performansı bakımından güçlü hale getirilmiş olan makine öğrenmesi tabanlı regresyon modelidir. Bu model zayıf öğrencilerin tahmin çıkışlarını toplayarak tahmin yapmaktadır. Algoritmanın çalışma hızı ve güvenilir sonuç vermesi göz önünde bulundurularak zayıf öğrencilerin sayısı deneysel olarak belirlenir[21].

2.5. Gauss süreç regresyonu

Gauss süreci regresyonu (GSR), esnek regresyon modelleri ve sınıflandırma için başlangıç dağılımını tanımlamak amacıyla kullanılan etkin bir yöntemdir [22]. GSR'nin en önemli özelliklerinden biriyse araştırmacılara farklı derecelerde fonksiyon oluşturabilme özelliği sağlayarak en doğru seçeneği belirlemesine imkân veren kovaryans fonksiyonlarının çeşitliliğine sahip olmasıdır. Birçok mühendislik probleminin çözülmesinde kullanılan regresyon süreci, olası parametrik olmayan modelleme metotları sağlamaktadır. Gauss süreci, gauss dağılımının genelleştirilmiş halidir. Gauss dağılımı rastgele değişimler arasındaki yayılımı iken Gauss süreci fonksiyonlar arasındaki dağılımı olarak açıklanır. Gauss süreç fonksiyonu olan $f(x)$ kovaryans ve ortalama $m(x)$ fonksiyonu ile tanımlanır. Aralarındaki matematiksel ilişki aşağıdaki denklem ile ifade edilebilir.

$$m(x) = E(f(x)) \quad (11)$$

$$k(x, x') = E((f(x) - m(x))(f(x') - m(x')))) \quad (12)$$

Burada, $k(x, x')$ kernel veya kovaryans fonksiyonunu ifade eder. $f(x)$ fonksiyonu ise şu şekilde tanımlanabilir:

$$f(x) = GP(m(x), k(x, x')) \quad (13)$$

Genellikle basitleştirme amacıyla ortalama fonksiyon değeri sıfır olarak kabul edilir. Gauss sürecinde giriş vektörü ile amaç vektörü arasındaki ilişki şu şekildedir:

$$y_i = f(x_i) + \varepsilon \quad (14)$$

Burada, $f(x)$ ve ε sırasıyla regresyon fonksiyonunu ve gauss dağılımının gürültü değerini ifade etmektedir. Ayrıca, kovaryans matrisi aşağıdaki gibi verilebilir.

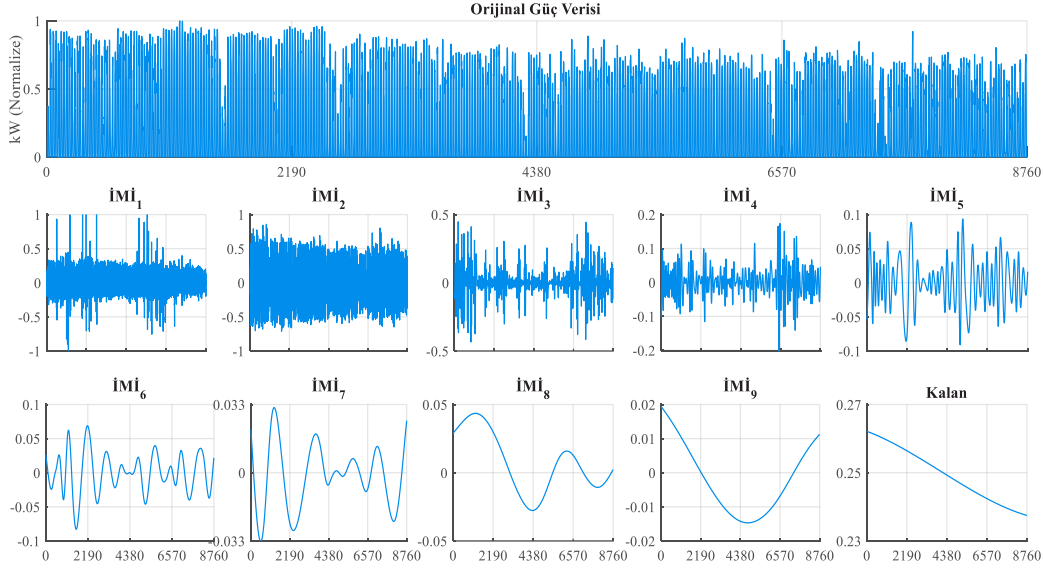
$$k_{i,j} = k(x_i, x_j) \quad (15)$$

$$k = \begin{bmatrix} k(x_1, x_1) & \dots & k(x_1, x_n) \\ \vdots & \ddots & \vdots \\ k(x_n, x_1) & \dots & k(x_n, x_n) \end{bmatrix} \quad (16)$$

Burada, k_i gizli fonksiyonlar olan $f(x_i)$ ve $f(x_j)$ değerleri arasında belirlenen kovaryanstır.

3. Ampirik Mod Ayırıştırıcı

Ampirik mod ayırıştırıcının temel kullanım amacı, yerel karakteristik zaman serilerine dayalı içsel salınım fonksiyonlarını belirlemektir [23]. Aşağıdaki formülde görüldüğü üzere bu yöntem, orijinal zaman serisini bir dizi içsel mod işlevine (İMİ) ve kalana dönüştürmektedir. Matematiksel olarak şu şekilde ifade edilebilir:



Şekil 1: Ampirik mod ayrıştırımından elde edilen içsel mod işlevleri

$$x(t) = \sum_{j=1}^a \text{İMİ}_j(t) + r_a(t) \quad (17)$$

Burada, “ $x(t)$ ” orijinal zaman serisini, “ j ” İMİ sayısını ve “ $r(t)$ ” ise kalan kısmı ifade etmektedir. Ampirik mod ayrıştırımı algoritması şu şekilde ifade edilebilir: İlk olarak verilen zaman serisinin maksimum ve minimum noktaları belirlenir. Daha sonra üst $u(t)$ ve alt bürüm $l(t)$ ’yi elde etmek için kübik bağlayıcı algoritması kullanılır. Daha sonra ise $u(t)$ ve $l(t)$ ’nin ortalaması alınarak $m(t)$ elde edilir. Bu adımdan sonra ilk bileşen hesaplanır, $h(t) = x(t) - m(t)$. Eğer $m(t)$ ’nin sifra yakın olması ve sifirgeçiş nokta sayısı ile yerel ekstrem nokta sayısı arasındaki fark en fazla bir olması şartları sağlanırsa $h(t)$ bir İMİ gibi düşünülebilir. Aksi halde yapılan işlemler ilk adımdan başlanarak $h(t)$ için tekrar edilir. Bu adımların ardından, ilk İMİ $h(t)$ ’ye eşittir ve sonra ilk artık bileşen hesaplanır. $r_1(t) = x(t) - h(t)$. Diğer $\text{İMİ}_{j+1}(t)$ ve $r_{j+1}(t)$ değerlerini elde etmek için ise yine ilk adımdan itibaren uygulanan işlemler $r_j(t)$ ’ye uygulanır. Eğer $r_{j+1}(t)$ ’deki ekstrem nokta sayısı ikiden küçük veya eşit olursa algoritma sona erdirilir. Ampirik mod ayrıştırma yöntemi sonrası güneş santralinden elde edilen güç örneği Şekil 1’deki gibi elde edilir.

4. Performans Değerlendirme Kriterleri

Bu bölümde, önerilen modelin diğer makine öğrenmesi metodları ile nicel performansını karşılaştırmak için kullanılan değerlendirme kriterleri anlatılmaktadır. Çalışmada, Korelasyon Katsayısı (R), Ortalama Karesel Hata (RMSE) ve Ortalama Mutlak Hata (MAE) olmak üzere üç tane performans değerlendirme kriteri belirlenmiştir. R, giriş ve çıkış parametrelerinin ne kadar bağlantılı olduğunu belirtir. RMSE ise tahmin hatasının standart sapması olarak tanımlanabilir. MAE ise tahmindeki hataların ortalama büyüklüğünü verir [24]. Bu bölümde bahsedilen değerlendirme metriklerinin her biri farklı bir değerlendirme olanağı sağladığı için makine öğrenme yöntemlerinden farklı zaman ufuklarında elde edilen tahmin performans sonuçları detaylı olarak incelenebilmektedir. Yukarıda bahsedilen performans değerlendirme kriterleri matematiksel olarak aşağıdaki gibi ifade edilebilir:

$$R = \frac{\sum_{i=1}^N [(P_i^f - \bar{P}^f) - (P_i - \bar{P})]}{\sqrt{\sum_{i=1}^N (P_i^f - \bar{P}^f)^2 \sum_{i=1}^N (P_i - \bar{P})^2}} \quad (18)$$

$$RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^N (P_i^f - P_i)^2} \quad (19)$$

$$MAE = \frac{1}{N} \sum_{i=1}^N |P_i^f - P_i| \quad (20)$$

Burada, P_i , santralin gerçek (ölçülen) çıkış gücünü verirken, P_i^f tahmin metodlarından elde edilen çıkış gücünü ifade etmektedir. \bar{P}^f ve \bar{P} sırasıyla gerçek ve tahmin edilen gücün ortalamasıdır.

5. Deneysel Çalışmalar

Bu kısımda, Avustralya, Alice Springs’de bulunan bir güneş enerji merkezindeki 1B: Trina isimli santralden [25] elde edilen elektrik enerjisinin kısa dönem tahmini için farklı makine öğrenmesi yöntemlerinin performansları incelenmiştir. Şekil 2’de Alice Springs’e ait genel yapı verilmiştir. Şekilden de görüleceği üzere bölgede birçok santral mevcuttur.

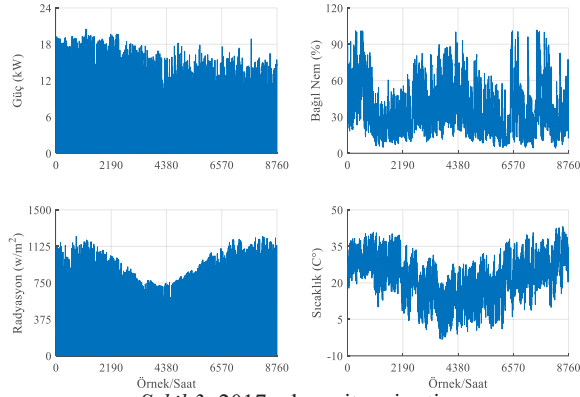


Şekil 2: Alice Springs’in genel gösterimi

Çalışmada kullanılan yöntemlere ait parametreler ise Tablo 1’de verilmiştir. Bu santralden toplanan veri seti 2017 yılına aittir ve saatlik aktif güç, bağıl nem, radyasyon ve sıcaklık gibi meteorolojik ölçümlerden oluşmaktadır. Elde edilen veri seti Şekil 3’te gösterilmiştir.

Tablo 1: Çalışma kullanılan yöntemlere ait parametreler

Parametre	DR	DVM	KAR	TR	GSR
Yaprak sayısı	-	-	4	4	-
Öğrenme sayısı	-	-	-	25	-
Dayanıklılık seçeneği	off	-	-	-	-
Fonksiyon ve aralığı	-	Kernel 0.60	-	-	Rasyonel Kuadratik otomatik



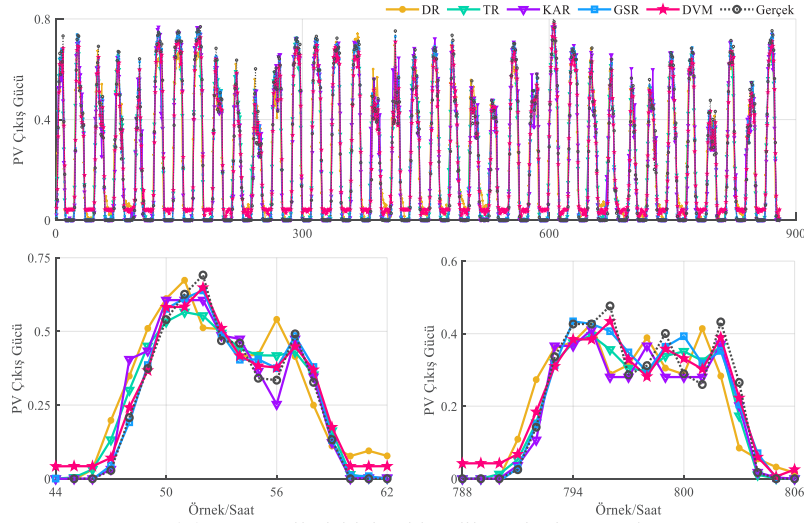
Şekil 3: 2017 yılına ait veri seti

Ayrıca, veri seti bir günün tüm saatlerini içerdiği için her bir meteorolojik ölçümler ve aktif güç için 8760 örnek kaydedilmektedir. Avustralya'da Kış; Haziran-Temmuz-Ağustos, İlkbahar; Eylül-Ekim-Kasım Sonbahar; Mart-Nisan-Mayıs ve Yaz ise Aralık-Ocak-Şubat aylarından oluşmaktadır. Güç örnekleri Ampirik mod ile bileşenlerine ayrıştırıldıktan sonra tüm veri seti 113,880 örnekten oluşmaktadır.

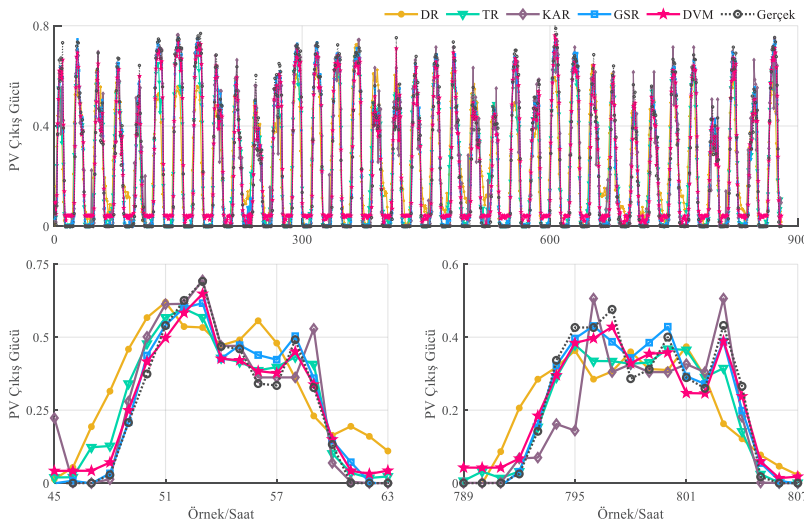
Veri setinde bulunan ölçüm değerlerinin minimum (Min), maksimum (Mak), medyan (Med), ortalama (Ort), standart sapma (Std) gibi istatistiksel özellikleri Tablo 2'de sunulmuştur.

Tablo 2: Veri setine ait istatistiksel özellikleri

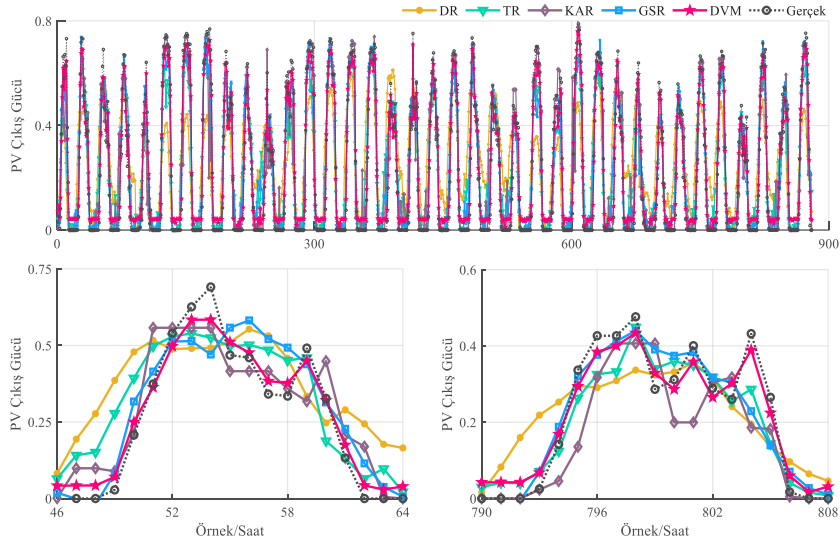
Değişken	Güç (kW)	Sıcaklık (°C)	Nem (%)	Radyasyon (Wh/m ²)
Min	0	-3.24	4.32	1.31
Mak	20.49	43.14	101.70	1226.90
Med	0.274	22.53	29.52	11.92
Ort	5.26	21.95	34.74	267.34
Std	6.46	9.40	20.82	356.81



Şekil 4: 1-saat ilerisi için elde edilen tahmin sonuçları



Şekil 5: 2-saat ilerisi için elde edilen tahmin sonuçları



Şekil 6: 3-saat ilerisi için elde edilen tahmin sonuçları

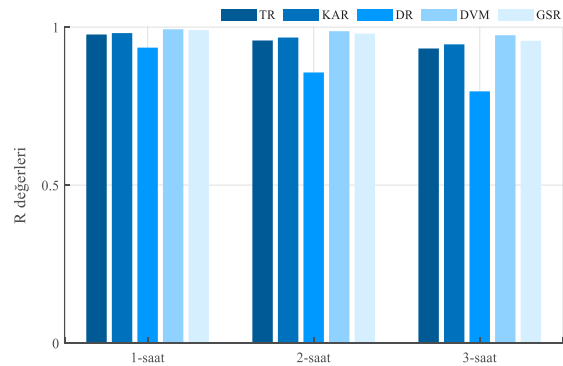
Daha sonra tahmin yöntemleri için eğitim, doğrulama ve test verileri hazırlanmıştır. Veri setinin %80'i eğitim için kullanılırken %10'u doğrulama ve %10'u da test için seçilmiştir. Eğitim boyunca seçilen veriler rastgele olacak şekilde tahmin yöntemlerine giriş olarak uygulanmıştır. En sık kullanılan makine öğrenme yöntemlerinden olan DR, DVM, KAR, TR ve GSR'nin güneş enerjisinin tahminindeki performansları farklı tahmin ufuklarında incelenmiştir. 1, 2 ve 3-saat ilerisi için tüm makine öğrenmesi yöntemlerinden elde edilen metrik sonuçları Tablo 3'te verilmiştir. Ayrıca, tahmin ufuklarından elde edilen sonuçlar Şekil 4, 5 ve 6'da detaylıca verilmiştir.

Tablo 3'teki metrik değerleri analiz edildiğinde tüm tahmin ufuklarında DVM'nin daha iyi sonuçlara sahip olduğu gözlemlenebilir. Ayrıca, tüm makine öğrenme yöntemlerinden elde edilen R değerlerinin daha iyi incelenebilmesi için Şekil 7 verilmiştir. R değerleri incelendiğinde DVM'nin değerleri 1-saat ilerisi için 0.9932, 2-saat ilerisi için 0.9871 ve 3-saat ilerisi için 0.9744 olarak hesaplanmıştır. En iyi ikinci R değerlerini ise 1-saat ilerisi için 0.9908, 2-saat ilerisi için 0.9795 ve 3-saat ilerisi için 0.9568 ile GSR sağlamıştır. En kötü R değerlerini ise 0.9351-0.8565-0.7967 değerleri ile DR göstermiştir. Elde edilen tüm R sonuçlarından, DVM'nin daha iyi ve güvenilir tahmin sonucu sağlayacağı açık bir şekilde ifade edilebilir.

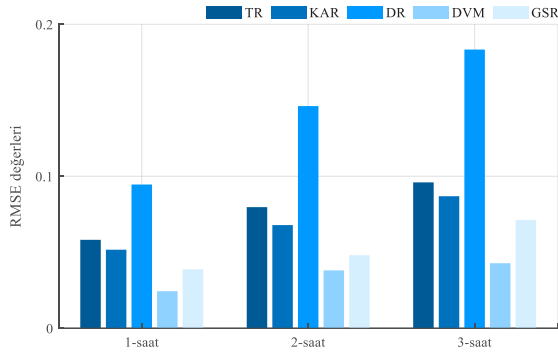
Diğer metrik RMSE için elde edilen değerler tüm tahmin ufukları için değerlendirildiğinde yine DVM'nin daha tatmin edici tahmin sonuçlarını sağladığı görülmektedir. RMSE değerlerinin daha iyi gözlemlenebilmesi için Şekil 8'deki grafik verilmiştir. 1, 2 ve 3-saat ilerisi için DVM'nin RMSE değerleri sırasıyla 0.0244, 0.0381 ve 0.0428 olarak hesaplanmıştır. GSR için RMSE değerleri 0.0388-0.0481-0.0712 olarak hesaplanmıştır. En kötü sonuçları ise yine DR, 1-saat ilerisi için 0.0946, 2-saat ilerisi için 0.1462 ve 3-saat ilerisi için 0.1834 değerleri ile vermiştir. Bu performans metriği sonuçları değerlendirildiğinde tahmin ufukunun değişimine karşı en kararlı sonuçları DVM sağlamıştır. Bu değerlerden DVM'nin uzun tahmin ufuklarında bile güvenilir tahmin sonuçlarını garanti edeceği anlaşılabilir.

Tablo 3: Tahmin ufuklarından elde edilen metrik sonuçları

Tahmin Ufku	Yöntem	Performans Metriği		
		R	RMSE	MAE
1-saat	TR	0.9767	0.0582	0.0344
	KAR	0.9811	0.0517	0.0272
	DR	0.9351	0.0946	0.0694
	DVM	0.9932	0.0244	0.0154
	GSR	0.9908	0.0388	0.0257
2-saat	TR	0.9577	0.0797	0.0527
	KAR	0.9671	0.0679	0.0386
	DR	0.8565	0.1462	0.1138
	DVM	0.9871	0.0381	0.0255
	GSR	0.9795	0.0481	0.0392
3-saat	TR	0.9324	0.0960	0.0707
	KAR	0.9455	0.0869	0.0488
	DR	0.7967	0.1834	0.1490
	DVM	0.9744	0.0428	0.0371
	GSR	0.9568	0.0712	0.0463

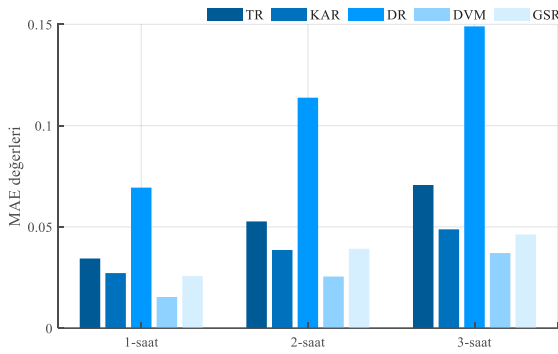


Şekil 7: Tüm tahmin ufuklarından elde edilen R değerleri



Şekil 8: Tüm tahmin ufuklarından elde edilen RMSE değerleri

Tüm makine öğrenme yöntemleri MAE değerleri açısından incelendiğinde yine en iyi değerlere DVM sahiptir. En küçük MAE değerleri ile daha az hata olduğu anlaşılabilmektedir. MAE değerlerinin daha iyi analizi için Şekil 9'daki grafik eklenmiştir. DVM'nin MAE değerleri 1-saat ilerisi için 0.0154, 2-saat ilerisi için 0.0255 ve 3-saat ilerisi için 0.0371 olarak hesaplanmıştır. DVM'yi yine en iyi ikinci performansa sahip olan GSR takip etmiştir. En kötü MAE değerlerine ise 1-saat ilerisi için 0.0694, 2-saat ilerisi için 0.1138 ve 3-saat ilerisi için 0.1490 ile DR vermiştir. Elde edilen MAE değerlerin incelendiğinde DVM'nin güvenilir tahmin sonuçlarını artan zaman ufkuna rağmen tatmin edici bir şekilde sağladığı görülmektedir.



Şekil 9: Tüm tahmin ufuklarından elde edilen MAE değerleri

6. Sonuçlar

Günümüz tüketim toplumunda enerji ihtiyacı her geçen gün büyüyerek artmaktadır. Enerjiye ihtiyaç artarken, enerji üretmek için kullanılan kaynaklar da günden güne azalmaktadır. Yenilenebilir enerji kaynaklarının giderek azalması ile yenilenebilir enerji kaynaklarına yönelimde bir hayli artmış durumdadır. Bu sebeple makine öğrenmesi metodlarının kullanılmasıyla Güneş'ten elde edilen elektrik enerjisinin tahmini ve bu doğrultuda enerji planı oluşturmak büyük bir önem taşır hale gelmiştir. Bu çalışmada, Avustralya, Alice Springs'de bulunan bir güneş enerji merkezindeki 1B:Trina isimli santralden elde edilen veri seti kullanılarak farklı regresyon yöntemlerinin tahmin performansları detaylı bir şekilde analiz edilmiştir. Bu amaç için sıklıkla kullanılan DR, DVM, KAR, TR ve GSR gibi makine öğrenmesi metodları tercih edilmiştir. Bu metodlardan elde edilen tahmin cevaplarını değerlendirmek için R, RMSE ve MAE gibi performans metrikleri kullanılmıştır. Ayrıca, tahmin ufkunun değişimine karşı makine öğrenmesi metodlarının dayanıklılığı ve

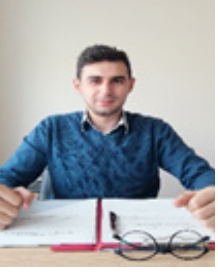
güvenilirliği analiz edilmiştir. Bu amaç için tahmin ufku 1 ila 3 saat sonrası olarak seçilmiştir. Tüm tahmin ufuklarından elde edilen metrik sonuçları için en iyi tahmin performansını DVM sağlamıştır. DVM'yi sırasıyla GSR, KAR, TR ve DR takip etmiştir. İleriki çalışmalarda ise farklı ayırıştırma yöntemlerinin tahmin performansı üzerine olan etkileri araştırılacaktır. Ayrıca, daha fazla meteorolojik verilerin elde edilmesi ile daha doğru tahmin sonuçlarının sağlanabilmesi hedeflenmektedir.

7. Kaynakça

- [1] Ahmed, R., Sreeram, V., Mishra, Y., Arif, M.D. "A review and evaluation of the state-of-the-art in PV solar power forecasting: Techniques and optimization", *Renewable and Sustainable Energy Reviews*, 124, pp. 1-26, 2020.
- [2] Akhter, M.N., Saad, M., Hazlie, M., Noraisyah, M. S. "Review on forecasting of photovoltaic power generation based on machine learning and metaheuristic techniques", *IET Renewable Power Generation*, 13(7), pp. 1009-1023, 2019.
- [3] Kim, S., Jung, J., Sim, M. "A two-step approach to solar power generation prediction based on weather data using machine learning", *Sustainability*, 2019.
- [4] Sobri, S., Koohi-Kamali, S., Rahim, N. "Solar photovoltaic generation forecasting methods: a review", *Energy Conversion and Management*, 156 pp. 459-97, 2017.
- [5] Raza, M., Q., M. Nadarajah, C. Ekanayake. "On recent advances in PV output power forecast", *Solar Energy*, 136, pp.125-144, 2016.
- [6] Şahan, M., Okur, Y. "Akdeniz bölgesine ait meteorolojik veriler kullanılarak yapay sinir ağları yardımıyla güneş enerjisinin tahmini", *Süleyman Demirel Üniversitesi Fen Edebiyat Fakültesi Fen Dergisi*, 11(1), ss. 61-71, 2016.
- [7] Demolli, H., Dokuz, A. Ş., Gokcek, M., Ecemiş, A. "Makine Öğrenmesi Algoritmalarıyla Güneş Enerjisi Tahmini: Niğde İli Örneği", *International Turkic World Congress on Science and Engineering*, ss. 775-783, 2019.
- [8] Gök, A.O., Yıldız, C., Şekkeli M. "Yapay sinir ağları kullanarak kısa dönem güneş enerjisi santrali üretim tahmini: Kahramanmaraş örnek çalışması", *Uluslararası Doğu Anadolu Fen Mühendislik ve Tasarım Dergisi*, 1(2), ss.186-195, 2019.
- [9] Wolff, B., Kühnert, J., Lorenz, E., Kramer, O., Heinemann, D. "Comparing support vector regression for PV power forecasting to a physical modeling approach using measurement, numerical weather prediction, and cloud motion data", *Solar Energy*, 135, ss. 197-208, 2016.
- [10] Mahmud, K., Azam, S., Karim, A., Zobaed, S. "Machine learning based pv power generation forecasting in alice springs", *IEEE Access*, pp. 1-13, 2021.

- [11] Massaoudi, M., Chihi I., Sidhom L., Trabelsi M., Refaat, S., Abu-Rub, H. "An effective hybrid narx- lstm model for point and interval pv power forecasting", *IEEE Access*, 9, pp. 36571-36588, 2021.
- [12] Korkmaz, D., Açıköz, H., Yıldız C. "A novel short-term photovoltaic power forecasting approach based on deep convolutional neural network", *Int J Green Energy*, 18, pp. 1–15, 2021.
- [13] Yıldız, C., Açıköz, H. "A kernel extreme learning machine-based neural network to forecast very short-term power output of an on-grid photovoltaic power plant", *Energy Sources Part A: Recovery, Utilization and Environmental Effects*, 43(4), pp. 395–412, 2021.
- [14] Uğuz, S., Oral, O., Çağlayan, N. "PV Güç Santrallerinden Elde Edilecek Enerjinin Makine Öğrenmesi Metotları Kullanılarak Tahmin Edilmesi", *Uluslararası Mühendislik Araştırma ve Geliştirme Dergisi*, 11(3), ss.769-779,2019.
- [15] Özdemir, M.H., İnce, M., Aylak, B.L., Oral, O., Taş, M.A. "Installed Solar Power Prediction For Turkey Using Artificial Neural Network And Bidirectional Long Short-Term Memory", *Business & Management Studies: An International Journal*, 8(5), pp. 4047-4068, 2020
- [16] Yulita, I. N., Abdullah, A. S., Helen, A., Hadi, S., Sholahuddin, A., Rejito J. "Comparison multi-layer perceptron and linear regression for time series prediction of novel coronavirus covid-19 data in West Java", *Journal of Physics: Conference Series*, 1722, pp. 1-8, 2021.
- [17] Abdullah, A. S., Ruchjana, B. N., Jaya, M., Soemartini. "Comparison of sarima and svm model for rainfall forecasting in bogor city indonesia" *Journal of Physics: Conference Series*, 1722, pp. 1-8, 2020.
- [18] Guo, W., Che, L., Shahidehpour, M., Wan X. "Machine-Learning based methods in short-term load forecasting", *The Electricity Journal*, 34, 2021.
- [19] Jijo, B. T., Abdulazeez, A. M. "Classification based on decision tree algorithm for machine learning", *Journal Of Applied Science And Technology Trends*, 2(1), pp. 20-28, 2021.
- [20] Murlidhar, B. R., Bejarbaneh, B. Y., Armaghani D. J., Mohammed A. S., Mohamad E. T. "Application of tree-based predictive models to forecast air overpressure induced by mine blasting", *Natural Resources Research*, 30(2), 2021.
- [21] Saha, M., Santara, A., Mitra, P., Chakraborty, A. Nanjundiah R S. "Prediction of the indian summer monsoon using a stacked autoencoder and ensemble regression model", *International Journal of Forecasting*, 37, pp. 58-71, 2021.
- [22] Ghasemi, P., Karbasi, M., Nouri, A. Z., Tabrizi, M. S., Azamathulla, H. M. "Application of gaussian process regression to forecast multi-step ahead SPEI drought index" *Alexandria Engineering Journal*, 60(6), pp. 5375–5392, 2021.
- [23] Qiu, X., Ren, Y., Suganthan, P. N., Amaratunga, G. "Empirical Mode decomposition based ensemble deep learning for load demand time series forecasting", *Applied Soft Computing Journal*, 54, pp. 246–55, 2017.
- [24] Acikgoz, H., Budak, U., Korkmaz, D., Yildiz, C. "WSFNet: An efficient wind speed forecasting model using channel attention-based densely connected convolutional neural network", *Energy*, 233, pp. 1-16, 2021.
- [25] DKASC. Alice Springs, 1B: Trina 2020. <http://dkasolarcentre.com.au/source/alice-springs/dka-m9-a-c-phases>

Özgeçmişler



M. Fatih BEKÇİOĞULLARI, Lisans eğitimini 2019 yılında Kahramanmaraş Sütçü İmam Üniversitesi Elektrik-Elektronik Mühendisliği bölümünde tamamlamıştır. 2020 yılında aynı üniversitenin Fen Bilimleri Enstitüsü Elektrik-Elektronik Mühendisliği Anabilimdalı'nda yüksek lisans eğitimine başlamıştır. 2021 yılında Gaziantep İslam Bilim ve Teknoloji Üniversitesi'nde Araştırma Görevlisi olarak çalışmaya başlamıştır. Araştırma alanları derin öğrenme, makine öğrenmesi ve enerji tahminidir.



Bünyamin DİKİCİ, 2021' de Gaziantep Üniversitesi Elektrik Elektronik Mühendisliği Bölümün' de yüksek lisans eğitimini tamamlamıştır. Aynı yıl içerisinde yine aynı okul ve bölümde doktora eğitimine başlamıştır. 2019 yılından beri Araştırma görevlisi olarak Gaziantep İslam Bilim ve Teknoloji Üniversitesin' de Elektrik-Elektronik Mühendisliği Bölümünde görev yapmaktadır. Araştırma alanları yapay zeka, derin öğrenme, sınıflandırma, tahmin ve haberleşmedir.



Hakan AÇIKGÖZ, Kahramanmaraş doğumludur. Lisans ve Yüksek Lisans eğitimini sırasıyla 2008 ve 2010 yıllarında Fırat Üniversitesinde tamamlamıştır. 2013-2018 yılları arasında, Kahramanmaraş Sütçü İmam Üniversitesi Elektrik-Elektronik Mühendisliği alanında doktora çalışmasını tamamlamıştır. 2010-2012 yılları arasında Batman Üniversitesi Mekatronik Mühendisliği bölümünde Araştırma Görevlisi olarak çalışmıştır. 2013-2019 yılları arasında ise Kilis 7 Aralık Üniversitesi Elektrik-Enerji bölümünde Öğretim Görevlisi olarak çalışmıştır. 2020 yılından itibaren ise Gaziantep İslam Bilim Ve Teknoloji Üniversitesi Mühendislik ve Doğa Bilimleri Fakültesi Elektrik-Elektronik Mühendisliği Bölümünde Öğretim Üyesi olarak çalışmaya devam etmektedir. Araştırma alanları güç dönüştürücülerinin denetimi, optimizasyon, yapay zekâ ve makine öğrenmesidir.



Ö. Fatih KEÇECİÖĞLÜ, Lisans, Yüksek Lisans ve Doktora Eğitimlerini sırasıyla 2010, 2012 ve 2017 yıllarında Kahramanmaraş Sütçü İmam Üniversitesi Elektrik-Elektronik Mühendisliği bölümünde tamamlamıştır. 2011-2017 yılları arasında Kahramanmaraş Sütçü İmam Üniversitesi Elektrik-Elektronik Mühendisliği bölümünde Araştırma Görevlisi olarak çalışmıştır. 2017-2021 yılları arasında aynı bölümde Dr. Öğr. Üyesi olarak çalışmıştır. 2021 yılından itibaren ise Elektrik-Elektronik Mühendisliği bölümünde Doçent olarak görev yapmaktadır. Araştırma alanları güç kalitesi, harmonik filtreler, güç elektroniği, reaktif güç kompanzasyonu ve yenilenebilir enerji sistemleridir.

Türkiye'deki Yazılım Organizasyonlarının Mikroservis Tabanlı Mimaride Uyguladığı Analiz ve Tasarım Yöntemleri Üzerine Bir Araştırma

A Survey on Analysis and Design Practices of Turkish Software Organizations for Microservice Based Architectures

Hüseyin Ünlü, Burak Bilgin, Onur Demirörs



Bilgisayar Mühendisliği Bölümü

İzmir Yüksek Teknoloji Enstitüsü

{huseyinunlu, burakbilgin, onurdemirors}@iyte.edu.tr

Özet

Günümüzde esnek, güvenilir ve duyarlı (*responsive*) yazılımlara olan talebin artması ve bu talebe karşılık verebilen internet altyapısının olması yeni nesil bulut uygulamalarının gelişmesine öncülük etmiştir. Bunun sonucunda, bulut tabanlı dağıtık web uygulamalarının birlikte çalışarak oluşturduğu mikroservis tabanlı mimari popülerlik kazanmıştır. Mikroservis tabanlı mimari oldukça yeni olup bu mimariyi kullanarak yazılım geliştiren firmaların kültürlerini değiştirme gerekliliği doğmuştur. Ancak, literatürde mikroservis tabanlı mimaride analiz ve tasarım konusunda uygulanan yöntemleri ele alan çalışma sayısı çok azdır. Bu çalışmada, Türkiye'deki yazılım organizasyonlarının mikroservis tabanlı proje geliştirirken başvurdukları analiz ve tasarım yöntemlerini ele alan bir anket düzenlenmiştir. Anket sonuçları, mikroservis tabanlı proje konusunda tecrübesi olan yazılım organizasyonlarının analiz ve tasarım konusundaki bakış açılarını ortaya çıkarmaktadır. Elde edilen sonuçlar mikroservis kullanan organizasyonlar ile ilgili genel tabloyu göstermekte ve araştırmacılar için çalışma yapılabilecek konuları önermektedir.

Anahtar kelimeler: mikroservis, yazılım analizi, yazılım tasarımı, anket, bulut platformu

Abstract

The demand for more flexible, responsive and reliable software applications and the availability of internet infrastructure to respond to this demand, led to a new generation of cloud-based web applications. As a result, cloud-based distributed web applications working together in a microservice-based architecture has gained popularity. The concept of microservice-based architecture is quite new and software organizations need to transform their culture to develop applications in this fashion. However, there is a lack of research studies in the literature that explores the common practices for the analysis and design of microservices. Thus, we performed a survey to explore the organizational choices on software analysis and design when working with microservices. In this paper, we present the results from Turkey. The results provide a snapshot of the software industry that utilizes microservices and suggests a set of challenges researchers can focus on in the area.

Keywords: microservices, software analysis, software design, practitioner survey, cloud applications

1. Giriş

Günümüz yazılımları, ölçeklenebilir ve neredeyse sınırları olmayan kaynaklardan oluşan uygulamalar yönünde değişmektedir. Bulut platformu ise günümüz yazılımlarının ölçeklenebilir ve neredeyse sınırları olmayan kaynaklardan oluşabilme özelliklerini sağlamaya yöneliktir. Bulut platformu sayesinde sağlanan bu özellikler kullanılarak iş mantığı farklı platformlara dağıtılabilir [1]. Mikroservis tabanlı mimari, iş mantığını farklı platformlara dağıtabilmek için bir çözüm olarak yer alır. Mikroservis mimarisinde gevşek bağlı (*loosely-coupled*) servisler yüksek kohezyon içerisinde (*cohesion*) çalışır. Bu servisler birbirinden bağımsız olarak dağıtılabilir, ölçeklendirilebilir ve test edilebilir; aralarındaki iletişim ise REST benzeri arayüzler kullanarak ağ aracılığı ile gerçekleştirilir [1], [2].

Mikroservis mimarisinin yazılım endüstrisindeki popülerliği geçtiğimiz yıllarda daha da artmaya başlamıştır. Mikroservis tabanlı mimaride, servislerin birbirinden bağımsız olarak geliştirilebilmesi ve dağıtılabilmesi, bu servislerin sürdürülebilir ve ölçeklendirilebilir olmasını sağlar. Mikroservislerin aslında birer servis olmasına rağmen Servis Odaklı Mimariden temel karakteristik farklılıkları vardır. Bu farklıklar boyut, sınırlı bağlam (*bounded context*) ve bağımsızlık olarak sıralanabilir [3]. İlk olarak, mikroservis boyutu bilinen servislere göre daha küçüktür; her bir servis genel olarak sadece bir iş mantığını sağlar. İkinci olarak, bir mikroservis ilgili bütün fonksiyonallikleri tek bir iş mantığı çerçevesinde toplar. Son olarak ise, her bir mikroservis bağımsız birer servistir. Her bir mikroservisin tek sorumluluğu olduğu için değiştirilmesi için sadece bir neden olmalıdır ve diğer mikroservisler bu değişimden etkilenmeyecek şekilde dizayn edilmelidir.

Servis odaklı mimari ile yazılım geliştiren organizasyonlar mikroservisler ile benzer konseptleri kullandıklarını, DevOps uygulayan organizasyonlar ise otomatikleştirilmiş dağıtım başladıklarını gözlemleyebilirler. Benzer olarak, çevik yazılım geliştiren organizasyonlar aslında kültürlerini mikroservis dünyasına uydurmaya başladıklarını gösterir [3]. Ancak, bu değişim görüldüğü kadar kolay olmamakla birlikte görülmemiş zorlukları da beraberinde getirir. Mikroservis tabanlı mimari ile yazılım geliştirmeye başlayan organizasyonlar bir yandan yazılımda sorumluluğu dağıtma ve özerklik ile diğer yandan da yazılım takımlarının çıktılarını etkili bir şekilde yönetme ve entegre

etmek ile uğraşmaktadırlar. Bunun yanı sıra, geleneksel analiz ve tasarım tekniklerinin mikroservisler ile çalışırken etkili olmadığını görmekteyiz. Bunun en önemli nedeni ise, en temel soyutlama kavramı olan “nesne”nin bile değişmek zorunda kalabilmesidir.

Literatürde, geleneksel tekniklerin yanı sıra mikroservis tabanlı mimari için birçok farklı modelleme notasyonu bulunmaktadır. Ancak, bu modeller incelendiğinde sistematik bir metodoloji sağlamadıkları ve dolayısıyla bu modelleme tekniklerinin anlaşılması güç olduğu görülmektedir. Ancak, kullanılan modelleme tekniklerinin anlaşılabilir olması organizasyonlar için büyük önem taşımaktadır [4]. 14. Ulusal Yazılım Mühendisliği Sempozyumu’nda (UYMS 2020) sunduğumuz bildirimizde Türkiye’deki organizasyonların bu zorluklar karşısındaki uyguladıkları yöntemleri ortaya çıkarmak amacıyla düzenlediğimiz anketin sonuçları gösterilmiştir [5].

Çalışmanın başlarında, mikroservis tabanlı mimaride analiz ve tasarım için başvurulan yöntemler ile ilgili sistematik literatür taraması yapılmış amaçlanmıştır. Ancak, bu konuda literatürde çalışmalarda eksiklik olduğu görülmüştür ve metodoloji organizasyonların mikroservis tabanlı yazılım geliştirirken izlediği analiz ve tasarım tekniklerini ele alan bir anket çalışması olarak değiştirilmiştir. Çalışmada, Türkiye’deki yazılım organizasyonların mikroservis tabanlı projelerde uyguladığı analiz ve tasarım teknikleri ile ilgili sonuçları ele almak amaçlanmıştır.

Daha önce gerçekleştirdiğimiz çalışmada, mikroservis tabanlı projeler geliştiren organizasyonların geleneksel nesne tabanlı analiz ve tasarım yöntemlerini sıklıkla kullanmaya devam ettikleri görülmüştür [5]. Ancak, mikroservis tabanlı mimarinin sahip olduğu önemli karakteristik özellikler bulunmaktadır ve bu özellikler nesne tabanlı mimariye göre farklılık göstermektedir. Bu nedenle, mikroservis tabanlı projelerde başvurulan analiz ve tasarım yöntemlerinin bu karakteristik özellikleri sağlamaya yönelik yol gösterici olması gerekmektedir. Bu çalışmanın amacı yazılım organizasyonlarının mikroservis tabanlı projelerde başvurduğu analiz ve tasarım yöntemlerini ortaya çıkarmanın yanında mikroservis tabanlı mimarinin karakteristik özellikleri ile ilgili de okuyucuyu bilgilendirmektir.

Bu çalışmanın geri kalan kısımları şu şekilde organize edilmiştir. 2. Bölümde mikroservis tabanlı mimarinin karakteristik özellikleri tartışılmaktadır. Bölüm 3 ilgili çalışmalarını, Bölüm 4 araştırma yöntemini vermektedir. Anket sonuçları 5. Bölümde verilmekte ve 6. Bölümde bulgular tartışılmaktadır. Son olarak, 7. Bölümde sonuçlar ve gelecekte planlanan çalışmalar ele alınmaktadır.

2. Mikroservis Mimarisinin Özellikleri

Mikroservis tabanlı mimarinin sahip olduğu önemli karakteristik özellikler vardır. Bu özellikler; gevşek bağlılık, yüksek kohezyon, izolasyon, özerklik, birleştirilebilirlik, sınırlı bağlam, asenkron iletişim, tek sorumluluk, ölçeklenebilirlik ve hata toleransı olarak sayılabilir [2], [6]–[8]. Ancak, bu özellikler nesne tabanlı geleneksel mimariye göre farklılık göstermektedir. Çalışmamızın bu bölümünde yukarıda sayılan karakteristik özellikler ile ilgili bilgi vermek amaçlanmıştır.

- *Gevşek bağlılık*: Mikroservis tabanlı mimaride, modüllerin birbirine bağlılığının olabildiğince düşük olması beklenmektedir. Buna bağlı olarak, mimari

tasarlanırken her bir iş yeteneğinin farklı bir servis tarafından ele alınacak şekilde ayrıştırılması gerekmektedir. Geleneksel mimaride (nesne tabanlı mimari) gevşek bağlılığın sınıflar arasında sağlanması hedeflenir. Ancak, mikroservis mimarisinde gevşek bağlılığın iş yetenekleri veya alt alanlara göre sağlanması gerekmektedir. Bunu sağlamanın bir diğer yolu ise farklı alt süreçlerin birbirinden ayrıştırılmasıdır. Mikroservislerin diğer önemli karakteristik özelliklerin sağlanabilmesinde mimarinin birbirine gevşek bağlı olan süreç ve iş yeteneklerine göre tasarlanmasının önemi büyüktür.

- *Yüksek kohezyon*: Mikroservis tabanlı mimaride belirli bir süreç ile ilgili olan tüm fonksiyonlilerin bir arada olması beklenmektedir. Bu sayede, aynı fonksiyonliler ile ilgili elementlerin akışı bir arada sağlanabilir. Geleneksel mimaride ise yüksek kohezyonun sınıflar bazında olması hedeflenir. Bir sınıfa ait metodların fonksiyonel olarak birbirine bağlı olmasına rağmen ilgili süreçte ait tüm fonksiyonlilerin bir arada tutmayı sağlaması her zaman mümkün olmayabilir. Bu bağlamda, mimari tasarımında ayrıştırmanın iş yetenekleri ve sınırlı bağlama göre yapılması, aynı süreçte ait fonksiyonliler bir arada tutmak için önemi büyüktür.
- *İzolasyon*: Mikroservis tabanlı mimari, her bir mikroservisin birbirini etkilemeden ve bağımsız olarak test edilebileceği, sürdürülebileceği ve dağıtılabileceği şekilde izole bir tasarım gerektirmektedir. Bu şekilde izole olarak tasarlanan mikroservisler, özerklik ve hata toleransı sağlamak için bir önkoşuldur. İzole mikroservislerin sağlanabilmesi için ortak veriye bağımlılığın olabildiğince engellenmesi gerekmektedir; dağıtık veri tabanı sistemleri sağlanmalıdır. Her bir mikroservisin kendisine ait veri koleksiyonu olmalıdır. Geleneksel mimaride ise genel olarak paylaşımlı veri yapıları kullanılmaktadır. Dağıtık veri tabanları olay günlüğü (event log) gibi yapılar kullanılarak sağlanabilir. Olay günlüğü yapısında, her mikroservis kendisine ait veri koleksiyonuna sahip olur. Bu sayede, ortak veriye bağımlılık engellenebilir.
- *Özerklik*: Her bir mikroservisin, izole veritabanına ek olarak, tek bir iş kapasitesine ait tam sorumluluğu alıp özerk bir şekilde çalışması beklenmektedir. Tam sorumluluk ise veri depolama ile birlikte sunum, API istekleri ve ilgili iş mantığını kapsamaktadır. Geleneksel mimaride sistemin bir bütün olarak ayakta durması gerekmektedir. Ancak, mikroservis mimarisinde her bir mikroservis birbirinden bağımsız tasarlanarak farklı kullanıcı veya mikroservisler tarafından bağımsız olarak çalıştırılabilir şekilde olmalıdır.
- *Birleştirilebilirlik*: Var olan mikroservisler, daha karışık veya yeni iş süreçlerini gerçekleştirebilecek şekilde yeniden birleştirilebilir olmalıdır. Mikroservisler, birbirleriyle işbirliği yaparak ve koordine olarak belirli bir süreci bir bütün olarak yönetmenin yanı sıra, harici mikroservisler ile de entegre olarak farklı kullanım senaryolarını da gerçekleştirebilir. Bu kompozisyonlar, var olan mikroservisler üzerinde ek değişikliklere neden olmamalıdır. Bu nedenle, SOA ilkeleri ve REST mimarisi ile uyum içinde arayüzlere sahip olmalıdırlar. Bu bağlamda, durumsuz (stateless) servislere ihtiyaç

duyulur. Geleneksel mimaride ise buna zıt olarak durumsal bir yapı bulunmaktadır.

- **Sınırlı bağlam:** Mikroservis boyutu bilinen servislere göre daha küçüktür; her bir servis genel olarak sadece bir iş mantığını sağlar. Geleneksel mimaride, sınırlar sınıf yapısına göre belirlenir; sınıf diyagramı sınırlı bağlamı belirlemek için kullanılan bir yoldur. Mikroservis tabanlı mimaride ise, sınırların belirlenmesinde iş mantığı ve ilgili veri önemli rol oynar. Bu nedenle, sınırların belirlenmesinde süreçler göz önünde bulundurulmalıdır.
- **Asenkron iletişim:** Mikroservis tabanlı mimaride, kaynakları daha verimli kullanmak ve daha iyi bir performans için asenkron iletişime ihtiyaç duyulmaktadır. Bu yapıyı sağlamak için en çok kullanılan yapılardan biri ise mesaj kuyruğudur. Bunu sağlayabilmek için olay tabanlı bir tasarıma ihtiyaç duyulmaktadır. Bu sayede, bir mikroservis bir olay tarafından tetiklenebilir ve görevini tamamladıktan sonra yeni bir olay oluşturabilir. Bu olaylar, mesaj kuyruklarında yığılanarak işleme alınabilir. Geleneksel tasarım yöntemleri olay tabanlı olmadığı için bu yapıyı oluşturmada bir strateji sağlayamamaktadır.
- **Tek sorumluluk:** Mikroservis tabanlı mimaride her bir mikroservis tek bir iş sürecini gerçekleştirecek şekilde ayrıştırılmalıdır. Mikroservislerin bu özelliği geleneksel mimari ile benzerdir. Geleneksel mimaride ayrıştırma sınıflar bazında gösterilir. Her bir sınıf tek bir iş sürecini gerçekleştirecek şekilde tasarlanır. Ancak, bu kompozisyon olay tabanlı olmak yerine nesne tabanlıdır. Mikroservis tabanlı mimaride ise nesne kavramı anlamını yitirmekte; yerine süreç kavramı önem kazanmaktadır. Bu nedenle, mikroservisler tek bir iş sürecini bir bütün olarak ele alacak şekilde tasarlanmalıdır.
- **Ölçeklenebilirlik:** Mikroservis tabanlı mimarinin en önemli özelliklerinden biri de sistemin ölçeklenebilir olmasıdır. Geleneksel mimariden farklı olarak her bir iş süreci ile ilgili mikroservisin kapasitesi anlık olarak artırılabilir veya düşürülebilir olmalıdır. Bunu sağlamak için her bir iş süreci birbirinden bağımsız olarak dağıtılabilir. Bu nedenle, nesne tabanlı tasarımın ötesinde süreçlere bağlı bir tasarım gerekmektedir.
- **Hata toleransı:** Mikroservis tabanlı mimarinin sağladığı gevşek bağlılık, izolasyon ve özerklik özellikleri monolit ve servis odaklı mimarilere kıyasla daha fazla hata toleransı sağlar. Her mikroservis diğer mikroservislerin yokluğunda da çalışmaya devam edebilmelidir. Ancak, bu senaryoda herhangi bir veri kaybı olmamalıdır. Geri gelen mikroservis, verileri bir yerden alarak çalışmaya devam etmelidir. Geleneksel mimaride, sınıflar farklı alt-süreçler tarafından kullanılabilirdiğinden paylaşım metodları ve veri tabanları bağlı olarak süreçler arası bağımlılık oluşturabilmektedir. Servis odaklı mimaride ise süreçlerin farklı servisler tarafından yönetilmesine rağmen bu servislerin ortak bir veri tabanına bağlı olmasından dolayı süreçler arası bağımlılık devam etmektedir. Ancak, mikroservis tabanlı mimaride her bir süreç bir mikroservis tarafından yönetilecek şekilde tasarlanmalıdır.

Mikroservis tabanlı mimarinin sağladığı özellikler geleneksel nesne tabanlı mimariye göre farklılık

göstermektedir. Mikroservis tabanlı mimaride kullanılan analiz ve tasarım yöntemlerinin bu önemli özellikleri sağlaması önem göstermektedir. Bu nedenle, geleneksel analiz ve tasarım yöntemlerinin mikroservis tabanlı mimariler ile çalışırken etkili olmadığı görülmektedir. Literatürde farklı modelleme tekniklerinin mikroservislerde kullanıldığı görülmektedir. Ancak, bu tekniklerin anlaşılabilir olması önem göstermektedir.

3. İlgili Çalışmalar

Literatürdeki çalışmalar incelendiğinde sektördeki organizasyonların mikroservis tabanlı projelerin analiz ve tasarım aşamalarını nasıl gerçekleştirdiğini ele alan bir araştırma olmadığı görülmektedir. Ancak, mikroservis alanında farklı konuları ele alan sistematik literatür taramaları bulunmaktadır. Pahl ve Jamshidi [9] tarafından yapılan çalışmada mikroservis ile ilgili 33 çalışma metodoloji, mimari ve platform desteği başlıklarında kategorize edilmiştir. Bir diğer çalışmada, Di Francesco ve diğerleri [10] mikroservis mimarisindeki çalışmaları yayın trendi, araştırma odağı ve endüstriyel adaptasyon potansiyeli başlıkları ile ele almıştır. Bu çalışma, mikroservisler ile ilgili yayınların 2014 yılında başladığını ve 2015 yılında yayın sayısının en fazla olduğunu göstermiştir. Alshuqayran ve diğerleri [11] tarafından yapılan çalışmada ise mikroservis ile ilgili çalışmalar mimari zorluklar, kalite nitelikleri ve mimari diyagramlar yönünden ele alınmıştır. Çalışma sonuçları mikroservis mimarisi için en çok kullanılan diyagramların bileşen (component), sıralama (sequence), süreç (process), dağılım (deployment), sınıf (class) ve kullanım durumu (use case) diyagramları olduğunu göstermektedir. Son olarak, Taibi ve diğerleri [12] tarafından yapılan çalışmada ise mikroservis tabanlı mimari prensip ve kalıpları ele alınmış ve bu kalıplar avantaj ve dezavantajları ile birlikte tartışılmıştır.

4. Araştırma Yöntemi

Bu çalışmada, organizasyonların mikroservis tabanlı projelerde uyguladıkları analiz ve tasarım yöntemleri hakkında bilgi edinebilmek amacıyla anket metodolojisine başvurulmuştur. Anket, belli kişilerin ya da grupların bir konu üzerindeki deneyimlerini kısa sürede anlayabilmek için sıklıkla başvurulan bir yöntemdir. Birçok farklı alanda başvurulduğu gibi, yazılım mühendisliği alanında da çeşitli konularda örnekleri bulunmaktadır [13]–[15]. Bu amaçla, anonim çevrimiçi anket yapılmasına karar verilmiştir. Bu seçiminin arkasında 3 ana neden bulunmaktadır. İlk olarak, anketin çevrimiçi olarak uygulanmasının nedeni bu sayede daha fazla katılımcıya ulaşabilmektir. İkinci olarak, anketin anonim olması ile katılımcıların şirket bilgilerini paylaşmadan tecrübelerini paylaşmaları sağlanmıştır. Son olarak ise elde edilen verinin daha kolay kategorize ve analiz edilmesini sağlanmıştır. Anket Google Forms uygulaması aracılığı ile uygulanmıştır.

Anketlerde olan ana problemlerden biri de anketin belirsiz sorular içerebilmesidir [16]. Olası belirsizlikleri kaldırmak amacı ile öncelikle bir pilot anket çalışması uygulanmış ve tamamlama süresi ile birlikte geri dönüşler alınmıştır. Bu uygulamadan sonra sorularda bazı düzeltmeler yapılmış ve anket bu düzeltmelerden sonra İzmir Yüksek Teknoloji Enstitüsü Etik Komitesi'nden onay alınarak uygulamaya başlanmıştır.

Anketin hedef grubu mikroservis tabanlı mimaride tecrübesi olan yazılım alanında çalışan uzmanlar olup bu

katılımcılara ulaşmak için çeşitli e-posta grupları, LinkedIn ve araştırmacıların çevresindeki bu konuda deneyimli uzmanlar ile anket paylaşılmıştır.

4.1. Amaç ve Araştırma Soruları

Bu çalışmada, organizasyonların mikroservis tabanlı projelerde başvurduğu analiz ve tasarım yöntemlerini belirlemek amacı ile araştırma soruları aşağıdaki şekilde belirlenmiştir:

- Organizasyonlar mikroservis tabanlı proje geliştirmede hangi analiz tekniklerini kullanmaktadır?
- Organizasyonlar mikroservis tabanlı proje geliştirmede hangi tasarım tekniklerini kullanmaktadır?

Tablo 1: Analiz ve tasarım soruları

Bölüm	Soru	Soru Tipi	
		Tek Yanıt	Çok Yanıt
5	Yazılım analizi için standart bir süreç kullanılıyor mu?	✓	
5	Şu anki (eğer bittiyse en son) mikroservis tabanlı projenizde fonksiyonel gereksinimlerini nasıl gösterdiniz?		✓
5	Şu anki (eğer bittiyse en son) mikroservis tabanlı projenizde problemi nasıl analiz ettiniz?		✓
5	Şu anki (eğer bittiyse en son) mikroservis tabanlı projenizde analiz için hangi notasyonları kullandınız?		✓
5	Şu anki (eğer bittiyse en son) mikroservis tabanlı projenizde problemi nasıl mikroservislere ayırtırdınız?	✓	
6	Şu anki (eğer bittiyse en son) mikroservis tabanlı projenizde tasarımı göstermek için hangi notasyonları kullandınız?		✓
6	Şu anki (eğer bittiyse en son) mikroservis tabanlı projenizde hangi tasarım kalıplarını (pattern) kullandınız?		✓

4.2. Anket Tasarımı

Anket, mikroservis mimarisi ile ilgili 6 bölümden oluşmaktadır. İlk bölüm anket ile ilgili bilgi vermekte ve katılımcının onayını almaktadır. İkinci bölüm katılımcılar ve çalıştıkları organizasyonlar hakkında bilgi alma amaçlı demografik sorular içermektedir. Bu bölümün son sorusu katılımcının mikroservis tecrübesi olup olmadığını sormaktadır. Katılımcı bu soruda tecrübesi olmadığını belirttiği takdirde anket sona ermektedir. Üçüncü bölümde

katılımcının mikroservis tecrübesi ile ilgili genel sorular sorulmaktadır. Dördüncü bölüm yazılım ölçümü ve efor kestirimi ile ilgili sorular içermektedir. Beşinci bölümde yazılım analizi ve altıncı bölümde is yazılım tasarımı ile ilgili sorular yer almaktadır. Bu çalışmada mikroservis tabanlı projelerde yazılım analizi ve tasarımı bölümleri ile ilgili sonuçlar ele alınacak olup bu iki bölümde yer alan sorular Tablo 1’de gösterilmektedir.

4.3. Doğrulama Kriterleri

Anketin hedef grubu mikroservis tecrübesi olan katılımcılar olduğu için mikroservis tabanlı proje geliştirmemiş katılımcıların cevapları sonuçlara yansımamalıdır. Bu nedenle, ikinci bölümde yer alan yedinci soruda katılımcının mikroservis tecrübesi olup olmadığı sorulmuştur. Bu soruya olumsuz cevap veren katılımcılarda anket sonlandırılmış ve diğer bölümlerdeki soruları yanıtlaması engellenmiştir.

5. Anket Sonuçları

Bu anket çalışmasına yazılım sektöründe çalışan 46 kişi katılmıştır, ancak bu katılımcıların 29’unun mikroservis ile yazılım geliştirme deneyimi bulunmaktadır ve bu başlık altında sadece deneyimi olan 29 katılımcının sonuçları göz önünde bulundurulacaktır.

Katılımcılarımızın lisans eğitimleri; Bilgisayar Mühendisliği, Yazılım Mühendisliği, Bilgisayar Bilimi, Matematik Mühendisliği, Elektrik ve Elektronik Mühendisliği ve Deniz Ulaştırma Mühendisliği, bölümlerinde iken, yüksek lisans eğitimleri; Bilgisayar Mühendisliği, Bilgisayar Bilimi, Bilgi Sistemleri, Elektrik ve Elektronik Mühendisliği ve Endüstri Mühendisliği bölümlerindedir. Katılımcılarımızın organizasyondaki rolleri ise; Yazılım Geliştirici (Developer), Kıdemli Yazılım Geliştirici (Senior Developer), Yazılım Geliştirme Yöneticisi, Yazılım Mimarı, Proje Yöneticisi ve Teknolojiden Sorumlu Yönetici (CTO)’dur. Katılımcılarımızın mevcut rollerindeki deneyimleri 1 yıldan daha az süre ile 15 yıl arasında değişmektedir ve ortalaması 4 yıldır. Yazılım sektöründeki deneyimleri ise 1 yıl ile 30 yıl arasında değişmektedir ve ortalaması 8 yıldır. Katılımcıların mikroservisler ile olan deneyimleri ise 1 yıl ile 10 yıl arasında değişmektedir ve ortalamaları 2,5 yıldır.

Katılımcıların çalıştıkları son mikroservis tabanlı projelerin bağlı olduğu sektörler; Otomasyon, Müşteri İlişkileri Yönetimi, E-ticaret, İnternet, Lojistik, Finans, Milli Yazılım, Mobil Yazılım, Telekom ve Kamu Hizmetleri’dir. Çalıştıkları son mikroservis projelerinin türlerine bakıldığında ise 13 katılımcı için (%44,8) monolitik mimari ile geliştirilmiş bir yazılım mikroservis mimarisinde yeniden yazılması, 12 katılımcı için (%41,4) yeni bir mikroservis tabanlı proje yazılması, 3 katılımcı için (%10,3) bir sistemin bazı parçalarını monolitik mimaride bir proje ile beraber çalışmak amacıyla mikroservis olarak yazılması, 1 katılımcı için (%3,4) ise var olan bir mikroservis tabanlı sistemin düzenlenmesi olduğu görülmüştür. Mikroservisler ile çalışırken başvurulan yazılım geliştirme modellerinde ise genellikle Çevik (Agile) yöntemlere başvurulduğu görülmüştür. 15 katılımcı (%51,7) sadece Scrum, 4 katılımcı (%13,8) sadece Kanban, 3 katılımcı (%10,3) Kanban ile Scrum, 2 katılımcı (%6,9) Kanban ile Ekstrem Programlama (EP), 2 katılımcı (%6,9) Kanban, Scrum ve EP, 2 katılımcı (%6,9) isim olarak bir yazılım geliştirme çerçevesi belirtmeden genel olarak Çevik (Agile) yazılım geliştirme, 1 katılımcı (%3,4) ise Şelale (Waterfall) modelini takip

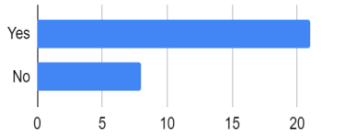
ettiklerini belirtmiştir. Katılımcıların çalıştıkları son mikroservis tabanlı projedeki personel sayısı minimum 2, maksimum 50, ve ortalama 9~ kişidir, organizasyonlarında paralel olarak devam eden mikroservis projesi sayısı ise; minimum 1, maksimum 75, ve ortalama 13'tür. Katılımcılarımız çoğunluğu (58.6%) aynı anda birden fazla mikroservis projesinde çalışmaktadır.

Katılımcıların organizasyonlarının çoğunluğunda (%82,8) DevOps uygulanmaktadır ve Jenkins en çok tercih edilen (%47,9) entegrasyon aracıdır. Mikroservis projelerinin test otomasyonuna bakıldığında ise, katılımcıların çoğunluğu (%37,9) 1-5 arası Likert ölçeğinde 4 olarak oylamıştır. Mikroservis tabanlı mimariler karşılaşılan en kritik problem sorulduğunda ise; katılımcıların çoğunluğu (%41,4) geçiş süreçlerin yönetimini, peşinden ise monolitik tabanlı bir yazılımı parçalamanın zorluğunu (29.3%), asenkron iletişimin zorluğunu (%18,9), veri tabanı ayrımlarının zorluğunu (%6,9), ve sürüm çıkılırken karşılaşılan kompleks adımların zorluğunu (%3,4) belirtmiştir.

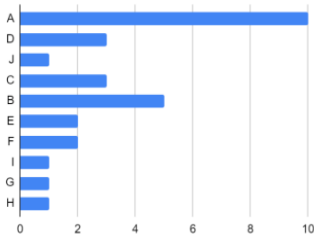
5.1. Yazılım Analizi

Bu başlık altında anketin yazılım analizi ile ilgili olan sonuçları verilmektedir. Verilen cevapları gösteren grafikler o seçeneğe cevap veren katılımcı sayısını göstermektedir.

Bu bölümde katılımcılarımızın cevapladığı ilk soru organizasyonlarında yazılım analizi için standart bir yazılım sürecinin olup olmadığıdır. Sonuçlar organizasyonların çoğunluğunun (%72,4) standart bir yazılım analiz sürecini takip ettiğini göstermektedir (Şekil. 1).



Şekil 1: Yazılım analizi için standart bir süreç kullanılıyor mu?



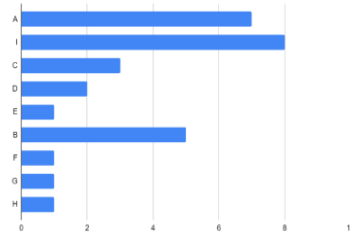
Şekil 2: Şu anki (eğer bittiyse en son) mikroservis tabanlı projenizde fonksiyonel gereksinimlerini nasıl gösterdiniz?

Sonuçlara göre, başka yöntemler ile beraber kullanılsa bile kullanıcı hikayesi yönteminin fonksiyonel gereksinimlerin belirtilmesinde organizasyonlar arasında en tercih edilen yöntem olduğu söylenebilir.

Üçüncü soruda katılımcılara mikroservis tabanlı projelerde problemi nasıl analiz ettikleri sorulmuştur. Sonuçların %24,1'i olay fırtınası (event storming) (A), %17,2'si Ad-hoc (B), %10,3'ü olay tabanlı (event-based) modellemeyi (C), %6,9'u olay fırtınası ve olay tabanlı modellemeyi (D), %3,4'ü Ad-hoc, olay fırtınası, ve olay tabanlı modellemeyi (E), %3,4'ü Ad-hoc ve olay fırtınası (F), %3,4'ü Ad-hoc ve olay tabanlı modellemeyi (G), %3,4'ü olay fırtınası, olay tabanlı model ve İş Süreci Model ve Gösterim (Business Process Model and Notation (BPMN)) (H),

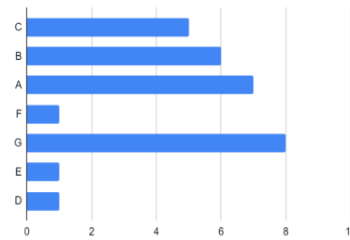
%27,6'sı ise hiçbirini (I) seçmiştir (Şekil. 3). Bu soruda hiçbirini seçeneğini seçen katılımcıların bu cevap ile problem analizi yapmadıkları mı yoksa kendileri için başka takımların mı problem analizi yaptığını bilmemek de en yüksek oranı bu cevabın almış olması dikkat çekicidir.

Sonuçlara göre, başka yöntemler ile beraber kullanılsa bile kullanıcı hikayesi yönteminin fonksiyonel gereksinimlerin belirtilmesinde organizasyonlar arasında en çok tercih edilen yöntem olduğu söylenebilir.

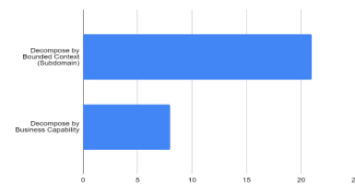


Şekil 3: Şu anki (eğer bittiyse en son) mikroservis tabanlı projenizde problemi nasıl analiz ettiniz?

Dördüncü soruda katılımcıların mikroservis tabanlı projelerin analizinde kullandıkları notasyonlar sorulmuştur. Katılımcıların %24,1'i sadece akış şeması (flow chart) (A), %20,7'si akış şeması ve aktivite diyagramı (B), %17,2'si sadece aktivite diyagramı (C), %3,4'ü akış şeması ve zihin haritası (mindmap) (D), %3,4'ü akış şeması, Aktivite diyagramı, değer akışı (value stream) haritaları ve BPMN (E), %3,4'ü eEPC (F), %27,6'sı ise hiçbir (G) notasyon kullanmadıklarını belirtmiştir (Şekil. 4). Önceki soruda olduğu gibi hiçbirini seçeneğini seçen katılımcıların bir notasyon kullanmadığını mı yoksa kendilerinin kullanmadıklarını belirttiklerini bilmemek de mikroservis tabanlı projelerin analizleri sırasında bir notasyona ihtiyaç duyulduğunda akış şemasının popüler bir tercih olduğu söylenebilir.



Şekil 4: Şu anki (eğer bittiyse en son) mikroservis tabanlı projenizde analiz için hangi notasyonları kullandınız?



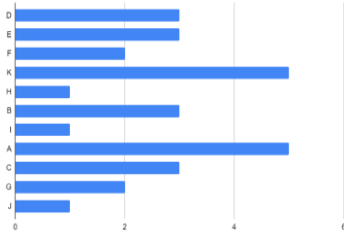
Şekil 5: Şu anki (eğer bittiyse en son) mikroservis tabanlı projenizde problemi nasıl mikroservislere ayrıştırdınız?

Son olarak katılımcılara problemi mikroservislere nasıl ayrıştırdıkları sorulmuştur. Sonuçlar bize organizasyonların çoğunlukla mikroservisleri sınırlı bağlama (bounded context) göre ayrıştırdıklarını (%72,4) ve bazı organizasyonların ise

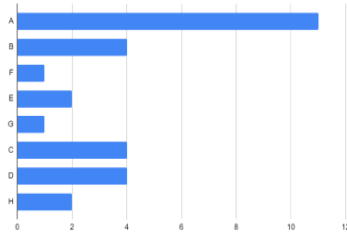
ayrıştırılmayı iş yeteneklerine (business capability) göre (%27,6) gerçekleştirdiklerini göstermiştir (Şekil. 5).

5.2. Yazılım Tasarımı

Yazılım tasarımı ile ilgili olarak katılımcılara ilk soruda mikroservis tabanlı projelerinde tasarımı göstermek için hangi notasyonları kullandıkları sorulmuştur. Sonuçların %17,2'si ardıl işlem (sequence) diyagramı (A), %10,3'ü sınıf diyagramı (B), %10,3'ü ardıl işlem ve sınıf (class) diyagramı (C), %10,3'ü sınıf, ardıl işlem ve aktivite (activity) diyagramı (D), %10,3'ü sınıf, ardıl işlem, aktivite ve varlık ilişki modeli (Entity Relationship Diagram (ERD)) (E), %6,9'u ardıl işlem ve aktivite diyagramı (F), %6,9'u sınıf diyagramı ve ERD (G), %3,4'ü aktivite diyagramı (H), %3,4'ü ERD (I), %3,4'ü sınıf, ardıl işlem diyagramları ve ERD (J), %17,2'si ise hiçbirinin (K) kullanılmadığını göstermiştir (Şekil. 6). Bu cevaplardan organizasyonlar arasında yazılım tasarımı için bir diyagrama ihtiyaç duyulduğunda ardıl işlem ve sınıf diyagramlarının tercih edilen bir yansıtma sistemi olduğu çıkarılabilir.



Şekil 6: Şu anki (eğer bittiyse en son) mikroservis tabanlı projenizde tasarımı göstermek için hangi notasyonları kullandınız?



Şekil 7: Şu anki (eğer bittiyse en son) mikroservis tabanlı projenizde hangi tasarım kalıplarını (pattern) kullandınız?

İkinci soruda ise, katılımcılara son mikroservis tabanlı projelerinde kullandıkları tasarım kalıpları (design pattern) sorulmuştur. Sonuçların %27,6'sı devre kesici (circuit breaker) (A), %24,1'i olay kaynağını belirleme düzeni (event sourcing) (B), %13,8'i komut ve sorgu sorumluluğu ayrımı (Command Query Responsibility Segregation (CQRS)) (C), %13,8'i devre kesici, olay kaynağını belirleme düzeni ve CQRS (D), %6,9'u olay kaynağını belirleme düzeni ve CQRS (E), %3,4'ü devre kesici ve bölme perdesi düzeni (bulkhead) (F), %3,4'ü API ağ geçidi (gateway), servis paylaşımını veri tabanı (shared database per service), log birleştirme (aggregation), performans metrikleri, dağıtılmış izleme (distributed tracing) (G), %6,9'u ise hiçbirini (H) kullanmadıklarını göstermiştir (Şekil. 7). Bu sonuçlar devre kesici modelinin mikroservis tabanlı mimariler sıkça tercih edilen bir model olduğunu, peşinden de olay kaynağını belirleme düzeni modelinin geldiğini göstermiştir.

6. Tartışma

Bu çalışmada anketimizin Türkiye sonuçları verilmiştir. Anket sonuçlarına göre, katılımcıların mikroservis tabanlı projelerde analiz ve tasarım için genel olarak başvurdukları bir yaklaşım olmadığını göstermektedir.

Katılımcıların çoğunun çalıştığı organizasyonlarda yazılım analizinde standart bir süreç kullandıklarını belirtmişlerdir. Ancak, bu süreçte başvurdukları yol değişkenlik göstermektedir. Kullanıcı hikayelerinin bu konuda en çok başvurulan yöntem olduğu ve bunun yanında doğal dil ve kullanım durumu (use case) senaryolarının da sıklıkla kullanıldığı görülmektedir. Bu sonuç, nesne tabanlı analiz yöntemlerinin mikroservis tabanlı projelerde de fonksiyonel gereksinimlerin belirlenmesinde kullanıldığını göstermektedir. Katılımcıların mikroservis tabanlı projelerde başvurdukları problem analiz yöntemine bakıldığında ise olay fırtınası (event storming) yönteminin en çok ve ad-hoc path ile olay tabanlı (event-based) modellemenin de sıklıkla kullanıldığı görülmektedir. Problem analizinde olay tabanlı yöntemlerin sıklıkla kullanılmasına rağmen fonksiyonel gereksinimlerin çıkarılmasında nesne tabanlı yöntemlere başvurulması dikkat çekmektedir. Analiz sürecinde bu organizasyonların nesne tabanlı yöntemler ile başlayıp sonrasında ise olay tabanlı modelleme ile devam etmeleri ve bu geçişi nasıl sağladıkları incelenmesi gereken bir konudur. Analiz için kullanılan notasyonlarda ise yine nesne tabanlı yöntemler (akış şeması ve aktivite diyagramı) kullanıldığı göze çarpmaktadır. Sadece bir katılımcı olay tabanlı modelleme şekli olan eEPC kullandığını belirtmiştir. Analiz konusunda son olarak katılımcıların problemi mikroservisler nasıl ayrıştırdıkları sorulduğunda büyük çoğunluğun sınırlı bağlama (bounded context) göre gerçekleştirdikleri görülmektedir. Bu katılımcıların problemi olay bazlı yöntemlere başvurarak analiz ederken geleneksel olan nesne tabanlı notasyonlar kullanarak göstermesi sonucunda sınırlı bağlamı nasıl sağladıkları ele alınabilecek bir başka konudur.

Mikroservis tabanlı projelerde tasarım nesne tabanlı projelere göre farklılık göstermektedir. Bunun en önemli nedeni, mikroservislerde sınırlı bağlamın nesnelere ile sağlanmamasıdır. Sonuçlara bakıldığında, nesne tabanlı tasarımın en önemli gösterim şekillerinden biri olan sınıf diyagramlarının mikroservis tabanlı projelerde de sıklıkla kullanıldığı görülmektedir. Ancak, mikroservis mimarisinde sınıf kavramı ve bu sınıfların arasındaki ilişki artık önemini yitirmekte ve farklı gösterim şekillerine ihtiyaç duyulmaktadır. Kullanılan tasarım kalıplarına bakıldığında ise yine olay tabanlı bir kalıp olan olay kaynağını belirleme düzeni (event sourcing) ikinci sırada görülmektedir. Bunun yanında kapalı devre (circuit breaker) en çok kullanılan tasarım kalıbıdır. Analiz ile paralel olarak, tasarım sürecinde de yine nesne tabanlı ve olay tabanlı yöntemlerin birlikte kullanıldığı görülmektedir. Ancak, nesne tabanlı gösterimden olay tabanlı tasarım kalıplarına nasıl geçiş yapıldığı yine incelenmesi gereken bir konudur.

Sonuç olarak, anketin Türkiye sonuçları organizasyonların, mikroservis tabanlı projelerde bir süreç izlediği ancak bu sürecin organizasyonlara göre farklılaştığı görülmektedir. Ancak, nesne tabanlı mimaride kullanılan yöntemler gibi kalıplaşmış bir yöntem bulunmamaktadır. Nesne tabanlı ve olay tabanlı yöntemler ile farklı kombinasyonlar kullanıldığı görülmektedir.

Mikroservis tabanlı projelerde başvurulan analiz ve tasarım yöntemlerinin mikroservislerin sahip olduğu

karakteristik özellikleri sağlayacak şekilde yol göstermesi gerekmektedir. Bu özellikler ise nesne tabanlı mimariye göre farklılık göstermektedir. Bu nedenle, nesne tabanlı analiz ve tasarım yöntemlerinin mikroservis tabanlı projelerin özelliklerini ne şekilde sağladığı bir soru işareti olarak kalmaktadır.

Literatür ile sektörü karşılaştırdığımızda; Alshuqayran ve diğerleri [11] tarafından yapılan çalışmada, mikroservis mimarisi için en çok kullanılan diyagramların bileşen (component), ardıl işlem (sequence), süreç (process), dağılım (deployment), sınıf (class) ve kullanım durumu (use case) diyagramları olduğu belirtilmektedir. Anket sonuçları da sektörde bu gösterimlerin sıklıkla kullanıldığını, nesne tabanlı yöntemlere yine sıklıkla başvurulduğunu göstermekte ve literatür ile paralel bir sonuç göstermektedir.

7. Sonuç

Mikroservis tabanlı mimari, yazılım geliştirmede popüler ve etkili bir yol olarak değerlendirilmektedir. Ancak, mikroservis yeni bir konsept olup yazılım organizasyonlarının kültürlerini değiştirmelerini gerektirmektedir. Yazılım analizi ve tasarımı teknikleri mikroservis tabanlı projelerde farklılaşmaktadır. Literatürde, mikroservis tabanlı projelerde analiz ve tasarım süreçlerini konu alan çalışmalarda eksiklik görülmektedir.

Bu çalışma kapsamında, yazılım sektöründeki organizasyonların mikroservis tabanlı proje geliştirirken başvurduğu analiz ve tasarım tekniklerini analiz etmek amaçlı bir anket düzenlenmiş ve bu anketin sonuçları organizasyonların bu konudaki tecrübelerini ortaya çıkarmak amacı ile sunulmuştur. Ankete Türkiye'den 46 kişi katılmıştır. Sonuçlar, ankete katılan organizasyonlarda mikroservis tabanlı projelerin analiz ve tasarımı için genel olarak benimsenen tutarlı bir yaklaşım olmadığını göstermektedir. Ankete katılan organizasyonlarda geleneksel nesne tabanlı analiz ve tasarım yöntemlerinin mikroservis tabanlı projelerin geliştirilmesinde de ağırlıklı olarak kullanıldığı görülmektedir. Ancak, geleneksel yöntemlerin mikroservis tabanlı projelerin analiz ve tasarım sürecinde ne derecede etkili olduğu tartışılması gereken önemli bir konudur.

Bu çalışmada, sektördeki organizasyonların mikroservis tabanlı projeler için analiz ve tasarım konusundaki bakış açılarını ortaya çıkarmak amaçlanmıştır. Anket sonuçlarının istatistiksel analiz yöntemleri kullanılarak değerlendirilmesi ve sonuçlar arasındaki korelasyonun incelenmesi planlanmaktadır.

8. Kaynaklar

- [1] A. R. Sampaio *et al.*, "Supporting Microservice Evolution," in *2017 IEEE International Conference on Software Maintenance and Evolution (ICSME)*, Sep. 2017, pp. 539–543, doi: 10.1109/ICSME.2017.63.
- [2] J. Thönes, "Microservices," *IEEE Softw.*, vol. 32, no. 1, Art. no. 1, Jan. 2015, doi: 10.1109/MS.2015.11.
- [3] N. Dragoni *et al.*, "Microservices: Yesterday, Today, and Tomorrow," in *Present and Ulterior Software Engineering*, M. Mazzara and B. Meyer, Eds. Cham: Springer International Publishing, 2017, pp. 195–216.
- [4] A. Dikici, O. Turetken, and O. Demirors, "Factors influencing the understandability of process models: A systematic literature review," *Inf. Softw. Technol.*, vol.

- 93, pp. 112–129, Jan. 2018, doi: 10.1016/j.infsof.2017.09.001.
- [5] B. Bilgin, H. Ünlü, and O. Demirors, "Analysis and Design of Microservices: Results from Turkey," presented at the 14th Turkish National Symposium on Software Engineering (Ulusal Yazılım Mühendisliği Sempozyumu, UYMS), Turkish National Symposium on Software Engineering (Ulusal Yazılım Mühendisliği Sempozyumu, UYMS), Oct. 2020.
- [6] J. Bonér, *Reactive Microservices Architecture*. O'Reilly Media, Inc., 2016.
- [7] J. Bonér, *Reactive Microsystems*. O'Reilly Media, Inc., 2017.
- [8] I. Nadareishvili, R. Mitra, M. McLarty, and M. Amundsen, *Microservice Architecture: Aligning Principles, Practices, and Culture*. O'Reilly Media, Inc., 2016.
- [9] C. Pahl and P. Jamshidi, "Microservices: A Systematic Mapping Study.," in *CLOSER (1)*, 2016, pp. 137–146.
- [10] P. Di Francesco, P. Lago, and I. Malavolta, "Architecting with microservices: A systematic mapping study," *J. Syst. Softw.*, vol. 150, pp. 77–97, Apr. 2019, doi: 10.1016/j.jss.2019.01.001.
- [11] N. Alshuqayran, N. Ali, and R. Evans, "A Systematic Mapping Study in Microservice Architecture," in *2016 IEEE 9th International Conference on Service-Oriented Computing and Applications (SOCA)*, Nov. 2016, pp. 44–51, doi: 10.1109/SOCA.2016.15.
- [12] D. Taibi, V. Lenarduzzi, and C. Pahl, "Architectural patterns for microservices: a systematic mapping study," SCITEPRESS, 2018.
- [13] V. Garousi, A. Coşkunçay, A. Betin-Can, and O. Demirörs, "A survey of software engineering practices in Turkey," *J. Syst. Softw.*, vol. 108, pp. 148–177, Oct. 2015, doi: 10.1016/j.jss.2015.06.036.
- [14] V. Garousi, A. Coşkunçay, and O. Demirörs, "A survey of software testing practices in Turkey."
- [15] D. Akdur, V. Garousi, and O. Demirörs, "A survey on modeling and model-driven engineering practices in the embedded software industry," *J. Syst. Archit.*, vol. 91, pp. 62–82, Nov. 2018, doi: 10.1016/j.sysarc.2018.09.007.
- [16] F. Shull, J. Singer, and D. I. Sjøberg, *Guide to advanced empirical software engineering*. Springer, 2007.

Özgeçmişler



Hüseyin Ünlü lisans derecesini (2016) Orta Doğu Teknik Üniversitesi (ODTÜ) Kuzey Kıbrıs Kampusu Bilgisayar Mühendisliği programından, yüksek lisans derecesini (2019) ise yine aynı üniversitenin Sürdürülebilir Enerji ve Çevre Sistemleri programından aldı. Yüksek lisans eğitimi süresince yine aynı kurumda Bilgisayar Mühendisliği programında öğretim asistanı olarak görev aldı. Doktora eğitimine İzmir Yüksek Teknoloji Enstitüsü (İYTE) Bilgisayar Mühendisliği bölümünde Prof. Dr. Onur Demirörs danışmanlığında devam etmekte ve aynı bölümde araştırma görevlisi olarak çalışmaktadır. Araştırma alanları arasında yazılım büyüklük ölçümü, mikroservis tabanlı mimariler, yazılım süreç geliştirme, yazılım kalite yönetimi konuları bulunmaktadır.



Burak Bilgin lisans derecesini (2017) İzmir Yüksek Teknoloji Enstitüsü (İYTE) Bilgisayar Mühendisliği programından almıştır. Mezuniyet sonrası Türk Ekonomi Bankası (TEB) Bilgi Teknolojisi departmanında yazılım geliştiricisi olarak çalışmaya başlamıştır ve çeşitli web uygulamaları geliştirmelerinde rol almıştır. 2021 yılından itibaren TEB Arf bünyesinde çalışmaya devam etmektedir. Araştırma alanları arasında çecik yazılım geliştirme ve yazılım ölçümü konuları bulunmaktadır.



Prof. Dr. **Onur Demirörs** lisans diplomasını Orta Doğu Teknik Üniversitesi, Bilgisayar Mühendisliği Bölümü'nden aldı. 1990'da Ege Üniversitesi, 1993'te ABD'de Southern Methodist Üniversitesi Bilgisayar Mühendisliği Bölümleri'nden yüksek lisans dereceleri elde etti. 1995'te ise yine Southern Methodist Üniversitesi'nde doktora çalışmalarını tamamladı. Bu üniversitede ve sonrasında Dokuz Eylül ve Orta Doğu Teknik Üniversiteleri'nde geçen uzun ve verimli bir akademik deneyimin ardından, 2017 yılında İzmir Yüksek Teknoloji Enstitüsü Bilgisayar Mühendisliği bölümüne katıldı. Araştırma alanları arasında yazılım süreç geliştirme ve süreç modelleri: metodlar, araçlar, yöntemler, iş süreç yönetimi, yazılım kalite yönetimi, yazılım ölçme, yazılım proje yönetimi ve kavramsal modelleme konuları bulunmaktadır.

Otomotiv Endüstrisinde Zamanlama Mimarilerinin Değerlendirilmesi

Evaluation of Scheduling Architectures in Automotive Industry

Berkay Saydam^{1,2}, Tolga Ayav²



¹Ar-Ge Merkezi

TTTech Auto Turkey

berkay.saydam@tttech-auto.com

²Bilgisayar Mühendisliği Bölümü
İzmir Yüksek Teknoloji Enstitüsü

tolgaayav@iyte.edu.tr

berkaysaydam@iyte.edu.tr

Özet

Teknolojik gelişmeler araçlara yansırken güvenlikten ödün vermeden araçlara yeni işlevler ekleme zorluğunu da beraberinde getirmektedir. Araçlarda çeşitli işlevleri yerine getiren görevler farklı karakteristiklere sahiptir. Güvenlik ve performans, bu görevlerin karakteristiğini belirlemek için kullanılan iki temel kriterdir. Görevlerin karakteristikleri, Otomotiv Güvenlik Bütünlük Seviyeleri olarak bilinen güvenlik seviyelerine göre sınıflandırılabilir. Donanım ve yazılım tasarımıyla birlikte bu tasarımların doğrulanması ve testi otomotiv endüstrisinde uzun soluklu bir süreçtir. Bir Elektronik Kontrol Birimi sahada kullanılmaya başlandıktan sonra donanım tasarımındaki herhangi bir değişiklik oldukça maliyetlidir. Bu makalenin hipotezine göre, uygun görev çalıştırma sıralarını belirlemek için kullanılan zamanlama algoritmaları Merkezi İşlem Birimi tarafından özenle seçilmelidir. Ayrıca donanım ve yazılım tasarımında bu karakteristik özellikler ve algoritmalar da dikkate alınmalıdır. Aksi takdirde görevler, bir kritik bileşen için zaman kısıtının kaçırılması gibi güvenlik açısından önemli sorunlara neden olabilmektedir. Bu makalede, zamanlama mimarileri değerlendirilerek hangi zamanlama mimarilerinin hangi amaçla kullanılması gerektiği belirtilmektedir. Algoritmaların avantaj ve dezavantajları sunulmaktadır.

Anahtar kelimeler: zamanlama algoritmaları, katı gerçek zamanlı sistemler, OSEK/VDX sertifikalı işletim sistemi, statik zamanlama analizi, araçlarda güvenlik

Abstract

Technological advancements are reflected to the vehicles as well, but it brings the challenge of adding new functionalities to vehicles without compromising safety. Tasks are used to provide functionalities which are used in car. These tasks have different characteristics. Safety and performance are two main criteria to determine the characteristics of tasks. These characteristics can be classified according to their safety levels which are known as Automotive Safety Integrity Levels. Design of hardware and software together with their verification and testing is a long progress in automotive industry. Any changes on either hardware or software may be quite costly when an ECU began to be used in field. According to the hypothesis of

this article, scheduling algorithms run by Central Processing Unit to determine the proper sequence of task executions, should be well known. Besides, the design of hardware and software should be done according to the characteristics and algorithms. Otherwise, tasks may cause serious problems like missing deadlines for safety-critical component. In this article, the scheduling architectures are evaluated distinguishing the purposes of them. The advantages and disadvantages of the algorithms are also explained.

Keywords: scheduling algorithms, hard real-time systems, OSEK/VDX certified operating system, static scheduling analysis, safety in vehicle

1. Giriş

Gelişen teknoloji farklı sektörlerde birçok ihtiyacı beraberinde getirdi. Bu gelişmeler aynı zamanda akıllı araçların ortaya çıkmasında önemli rol oynamaktadırlar. Güvenlikten ödün vermeden araçlara yeni işlevsellik kazandırmak için araç performansının iyileştirilmesi önem arz etmektedir. Araçlarda elektrikli sistemleri kontrol eden gömülü sistemlere Elektronik Kontrol Birimi (EKB) adı verilmektedir. Araçlara görevler aracılığıyla işlevsellik sağlayan mikrodenetleyiciler EKB tasarımının ilk adımında belirlenmektedir. Bu karar proje geliştirme sürecinin sonraki adımlarını önemli derecede etkilemektedir. Donanım, yazılım tasarımı ve test süreci otomotiv endüstrisinde oldukça uzun bir süreçtir. EKB'ler sahada kullanılmaya başlandıktan sonra donanımda yapılacak değişiklikler oldukça maliyetlidir. Bu nedenlerden dolayı mikrodenetleyici seçimi müşteriye sağlanacak işlevsellikleri ve maliyeti etkilemektedir.

Otomotiv sistemlerinde görevler araçlarda kullanılan işlevleri sağlamak için kullanılmaktadır. Bu görevler farklı karakteristiklere sahiptir. Güvenlik ve performans görevlerin karakteristiklerinin belirlenmesinde iki ana kriter olmaktadır. Karakteristikleri, Otomotiv Güvenliği Bütünlük Seviyeleri (ASIL) olarak bilinen güvenlik seviyelerine göre sınıflandırılabilir. ASIL-A seviye görevler performans sağlamak için birden fazla prosese ihtiyaç duyarken ASIL-D seviye görevler katı yanıt süresine ihtiyaç duyarlar. Makalede savunulan hipoteze göre, Merkezi İşlem Birimi (MİB) tarafından görev yürütme sıralarını belirlemek için kullanılan

zamanlama algoritmaları iyi seçilmelidir. Ayrıca donanım ve yazılım tasarımı bu görevlerin karakteristiklerine ve algoritmalara göre yapılmalıdır. Aksi durumda güvenlik-kritik görevlerin zaman sınırını kaçırmaması gibi ciddi sorunlara neden olacaktır. Birden fazla donanım ve yazılım tasarımı süreci geçireceği için ek maliyete de sebep olacaktır.

Bu makalede zamanlama mimarileri değerlendirilmiş ve hangi zamanlama mimarisinin hangi amaçla kullanılması gerektiği belirlenmiştir. Ayrıca avantaj ve dezavantajları açıklanmıştır. Böylelikle EKB tasarımından sorumlu bir yazılım mimarı, amacına yönelik olarak hangi zamanlama mimarisini kullanması gerektiğini bilecek ve doğru mikrodenetleyici seçim kararını sonuç kısmında bulunan kriter ve tasarım fikirlerine göre verebilecektir. Otomotiv endüstrisine katkı olarak bu çalışmada bir EKB'nin hangi durumda birden fazla mikrodenetleyici içereceği, ve bu mikrodenetleyicilerin hangi amaçlara göre seçilmesi gerektiği konularına da ışık tutmaktadır.

Analiz aşamasında zamanlama algoritmalarının avantaj ve dezavantajlarını göstermek için gerekli ortam kurulmuştur. Offene Systeme und deren Schnittstellen für die Elektronik in Kraftfahrzeugen/Vehicle Distributed eXecutive (OSEK/VDX) sertifikalı ve aynı zamanda ücretsiz olan Erika işletim sistemi seçilmiştir. Erika işletim sistemi derlenmiş ve Arduino Uno devre kartına gömülmüştür. Sabit Öncelikli Zamanlama Algoritması olan Oransal Monoton (OM) analiz edilecek ilk zamanlama algoritması olarak seçilmiştir. Bu senaryodan ortaya çıkan davranış görselleştirilmiştir. Zaman sınırını aşan bir görevle rastlanmıştır fakat bu görev düşük önceliğe sahiptir. Bu, OM için bir dezavantaj olsa da süre aşımı düşük öncelikli bir görevde gerçekleşmiştir. Bu görev araç kapısı açık olduğunda araç içerisinde yanan ışığın işlevselliğini sağlayan bir ASIL-A görevi olabilir. Bu görevdeki gecikme sistemde katastrofik sonuçlara sebep olmayacaktır. MİB kullanımını azaltılarak bu süre aşımı elimine edilebilir. Bu analiz sabit öncelikli zamanlama algoritması olan OM'nin performanstan feragat ederek güvenli bir sistem oluşturmak için kullanılabileceğini göstermektedir.

Bir sonraki algoritmayla Dinamik Öncelikli Zamanlama Algoritması olan En Yakın Zaman Sınırı Önce (EYZSÖ) olmuştur. Görevleri zaman sınırlarına göre otomatik yürütmesinden dolayı sabit öncelikli algoritmalara göre daha az efor ihtiyacı vardır. Aynı senaryo EYZSÖ için yürütülmüştür. Herhangi bir zaman sınırı aşımına rastlanmamıştır. Bu bize EYZSÖ'nin sabit öncelikli algoritmalarından daha fazla MİB kullanımını sağladığını göstermektedir. Fakat bu EYZSÖ'nin dezavantaja sahip olmadığını göstermez. EYZSÖ her görev için zaman sınırını yerine getirmeyi garanti etsede verilen herhangi bir görev için minimum yanıt zamanını sağlamak mümkün değildir. OM'de en yüksek öncelikli görev her zaman minimum yanıt zamanına sahiptir, fakat bunu EYZSÖ'de garanti etmek mümkün değildir. Yeni bir görev eklenmesinden dolayı ya da yanlış bir En Kötü Durum Yürütme Süresi (EKDYS) tahmininden dolayı sistem aşırı yüklenirse domino etkisi oluşabilir. Domino etkisi, bir görev bildirilen çalışma süresinden daha fazlasını yürütmekle son tarihini kaçırdıktan sonra diğer tüm görevlerin son tarihlerini kaçırmıştır. Bu da kaza anında airbagleri açan ASIL-D seviyesinde bir görevin zaman sınırını kaçırmamasını mümkün kılabilir. Bu görevdeki gecikme sistemde katastrofik sonuçlara sebep olacaktır. Bu

nedenle EYZSÖ yüksek performans ve düşük güvenlik gerektiren görevlerde kullanılabilir.

Otomotiv endüstrisi sadece performans ve güvenlik parametrelerine sahip değildir. Deterministik ve öngörülebilirlik de otomotiv sektöründeki geliştiriciler için önemlidir. Bu nedenle Zaman Tetiklemeli (ZT) zamanlama mimarisini önemli bir yere sahiptir. OSEK Uygulama Dili (OUD) dosyası işletim sistemlerinin temel parametrelerini belirlemek için sıkça kullanılır. Bu mimari çevrimdışı tanımlanmış görevler ve bu görevlerin tetiklenme zamanlarını içeren zamanlama tablosuna sahiptir. Bu zamanlama tablosu OUD dosyasında hazırlandı ve her görev verilen zamanlarda tetiklendi. Analizler sonucunda olay tetiklemeli ve zaman tetiklemeli mimarileri karşılaştırmak gerekirse olay tetiklemeli zamanlama mimarisini esnek ve yüksek kaynak kullanıma izin vermektedir. Fakat katı gerçek-zamanlı sistemlerde önemlilik arz eden deterministik ve öngörülebilirlik açısından güvenilir değildir. Ayrıca, olay tetiklemeli mimari daha yüksek çalışır-zaman yüküne sahiptir. Zaman tetiklemeli mimari periyodik bir dünyaya sahiptir. Düzensiz görevler için esnek değildir. Deterministik ve öngörülebilirliği düşük kaynak kullanımı sağlar.

Ulusal Yazılım Mühendisliği Sempozyum'unda yayınlanan bildirinin [1] aksine özet ve giriş kısmı makalenin değindiği problemi, amacı ve yazılma motivasyonunu daha anlaşılır bir şekilde anlatmak amacıyla tekrar yazılmıştır. Otomotiv endüstrisinde yapılan güvenlik çalışmaları standartlar dahilinde detaylandırılmıştır. Daha anlaşılır hale getirmek amacıyla gerçek-zamanlı sistemlerin ve algoritmaların kıyaslanması tablolar halinde gösterilip anlatımları detaylandırılmıştır. Zamanlama algoritmalarının kullanım kriterleri verilmiştir. Oransal monoton algoritmasının geçerlilik hesabı yapıp adım adım gösterilmiştir. Önceki yayınlanan bildirinin senaryosu dışında farklı MİB kullanım oranları ve farklı task sayıları ile analizler yapılmış tablo halinde çıktılar verilmiş ve sonuçları yorumlanmıştır.

Güvenlik, otomotiv endüstrisinin çok önemli bir yönüdür. Güvenli bir sistem üretimi ve geliştirmesi için otomotiv şirketleri tarafından bazı standartlar oluşturulmuştur. Bu standartlardan Bölüm 2'de bahsedilecektir. Araçlarda güvenli yolculuk yapılması için zaman önemli bir parametredir. Gerçek zamanlı sistemler, tanımlanmış kısıtlı zamanda yanıt vermeyi garanti etmelerinden dolayı, zaman parametresinin önemli olduğu alanlarda yaygın olarak kullanılır. Bu sistemler hakkında detaylı bilgi Bölüm 3'te bulunacaktır. Sistemdeki görevleri yönetmek için her gerçek zamanlı sistem Merkezi İşlem Birimi içerir. MİB, görevleri belli bir sıraya göre çalıştırmak için kararlar almaktadır. Bu kararları almak adına bazı algoritmalara ihtiyaç duyar. Bu algoritmalara zamanlama algoritmaları denir. Bölüm 4'te konu hakkında bilgi verilecektir. Gerçek zamanlı sistemler, sistem yanıtı geciktiğinde ortaya çıkan tehlikelere göre sınıflandırılabilir. Bunlar esnek gerçek zamanlı, sıkı gerçek zamanlı ve katı gerçek zamanlı sistemlerdir. Gerçek zamanlı sistemler ihtiyaca göre özel işletim sistemleri kullanır. Katı gerçek zamanlı sistemler için kullanılan mevcut işletim sistemleri Bölüm 5'te ele alınmıştır. Bölüm 4'te açıklanan zamanlama algoritmaları, bölüm 5'te açıklanan katı gerçek zamanlı işletim sistemi üzerinde uygulanmıştır. Uygulama adımları, bulgular ve değerlendirmeler Bölüm 6'ya eklenmiştir. Bölüm 6'daki

sonuçlar üzerinde çıkarımlar yapıp sonuç bölümünde çıktılar ayrıca değerlendirilmiştir.

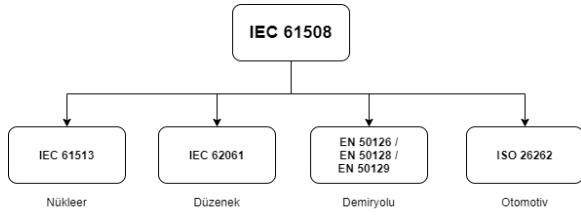
2. Otomotiv Endüstrisinde Güvenlik

Bu bölümde, otomotiv endüstrisinde emniyetin önemi vurgulanmakta ve emniyetle ilgili mevcut standartlar sunulmakta ve gözden geçirilmektedir. İlk olarak ISO 26262 standardının IEC 61508'den türetilmesi ve sonrasında da ISO26262 standardında tanımlı olan V-model ve ASIL seviyeleri hakkında detaylı bilgi verilecektir. Son olarak güvenli yazılım geliştirmek için yararlanılan MISRA'ya değinilecektir.

Güvenliğin sözlük anlamı tehlike, risk ya da yaralanma durumlarından korunma durumudur. İnsanlara zarar vermeyen bir sistem güvenli bir sistem olarak nitelendirilebilmektedir. Tamamen güvenli bir sistem yoktur, bu nedenle güvenli sistemler potansiyel riski kabul edilebilir bir düzeye indirmeye çalışır.

2.1. Güvenlik Standartları

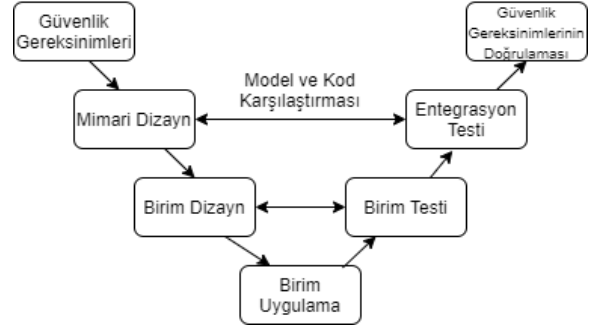
Otomotiv endüstrisinde güvenlik standartlarını belirleyen büyük topluluklar bulunmaktadır. Bunlardan biri Uluslararası Standardizasyon Örgütü'dür (ISO) 26262 [2]. Aslında Uluslararası Elektroteknik Komisyonundan (IEC) 61508 [3] türetilmiştir. IEC 61508, her türlü endüstri için geçerli olan temel bir fonksiyonel güvenlik standardıdır. Ayrıca, ISO 26262, Otomotiv Elektrik/Elektronik (E/E) Sistemleri için IEC 61508'in bir uyarlamasıdır. ISO 26262, güvenlik kapsamında otomotiv E/E ürünlerinin tüm ürün geliştirme yaşam döngüsünde süreçleri kısıtlar.



Şekil 1: ISO26262 türetilmesi.

2.2. V-model

ISO 26262, yazılım geliştirmede V modeli üzerinden tanımlanmaktadır [2]. Model tabanlı yazılım tasarımıdır, çünkü model tabanlı tasarım ve ISO 26262 birbirini tamamlar. Bu model V modeli olarak adlandırılır, çünkü fazlar birbirlerini V harfi şeklinde doğrularlar. Bu model Şekil 2'de gösterilmektedir.

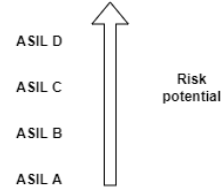


Şekil 2: V-model.

Bu çerçevenin bazı aşamaları vardır. Birinci aşama, güvenli olmayan veya etkisiz olarak donanım veya yazılım hatalarını gidermek için belirlenen birçok güvenlik gereksinimine sahiptir. Bir sonraki aşamada, bir önceki aşamada belirlenen güvenlik gereksinimlerini sağlayan üst mimari, her bileşen için tasarlanmıştır. Diğer aşamada yazılım birimi alt sistemleri de benzer şekilde tasarlanmaktadır. Tasarım aşamaları paralelde test de edilmektedirler. Birim testinden sonra, tüm sistem bütünleşmiş birimlerin davranışından emin olmak için entegrasyon test adımıyla geçer. Son aşamada, güvenlik gereksinimlerini doğrulamak için sistem gerçek ortamda test edilir.

2.3. Otomotiv Güvenlik Bütünlük Seviyeleri

ISO 26262 gerekliliklerinin yapılandırılmasında, gelişimin derecesini belirlemek için Otomotiv Güvenlik Bütünlük Seviyeleri (ASIL) kullanılır [3]. ASIL zararın olasılığı ve kabul edilebilirliğine dayanmaktadır. Standart olarak A, B, C ve D olmak üzere dört seviye vardır. ASIL-A en düşük otomotiv tehlikesini temsil ederken ASIL-D en yüksek dereceyi temsil eder [4]. Örneğin, arka lambalar gibi bileşenler ASIL-A sınıfı gerektirirken, hava yastıkları, kilitlenme önleyici fren sistemleri ASIL-D sınıfı gerektirir.



Şekil 3: ASIL.

Hava yastıklarının arızası, arka lambaların arızasına göre daha yüksek bir hasar olasılığına sahiptir. Her elektronik bileşen için ASIL derecesi ciddiyet, maruziyet ve kontrol edilebilirlik değişkenlerine göre belirlenir. ASIL'in sınıflandırılması, araçlarda en yüksek güvenliği sağlamaya yardımcı olur.

2.4. MISRA C

Kıdemli programcılar kolayca hata yapmaktan kurtulabilirler. Ancak bu durum genç programcılar için geçerli değildir. Öte yandan, donanıma kolay erişim, düşük bellek gereksinimleri ve verimli çalışma zamanı performansı, gömülü sistemlerde C programlama dilinin popüler kullanımının nedenleridir. Bununla birlikte C, teknik olarak yasal olan basit hatalara

eğilimli bir sözdizimi gibi oldukça sınırlı çalışma zamanı denetimi gibi bazı sorunlara sahiptir.

Bu nedenlerden ötürü, Motor Endüstrisi Yazılım Güvenilirlik Derneği (MISRA), güvenlik açısından kritik sistemlerde C programlama dilinin kullanımı için MISRA C adı verilen bir dizi yazılım geliştirme kılavuzu oluşturmuştur [5]. Otomotiv endüstrisinde kullanılan gömülü sistemlerde kod güvenliği, taşınabilirliği ve emniyetini sağlamayı amaçlamaktadır.

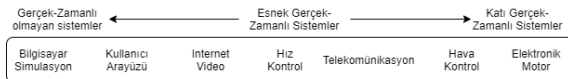
ISO 26262 uyumlu EKB zamanlamaları oluşturmak için görev zamanlama güvenliği doğrulanmalıdır. Zamanlama, araç sistemlerinde güvenilirlik ve güvenlik için kritik bir performans faktörüdür. Yeni teknoloji ile işlevsellik arttıkça zamanlama analizi gittikçe zorlaşmaktadır. Güvenlik gereksinimlerinden biri, kararlı ve öngörülebilir zamanlama davranışına ve gereken bilgi işlem gücü miktarına sahip görevlerdir.

3. Gerçek Zamanlı Sistemler

Bu bölüm gerçek zamanlı sistemlerden zamanlama algoritmalarına kadar detaylı bilgilendirmenin yapıldığı bölümdür. Gerçek zamanlı sistemlerin farklarından bahsedilecek ve gerçek zamanlı sistemlerde sınıflandırma yapılacaktır. Sonrasında zamanlama ile ilgili genel konsept anlatılıp zamanlama algoritmalarının sınıflandırılması ile tamamlanacaktır.

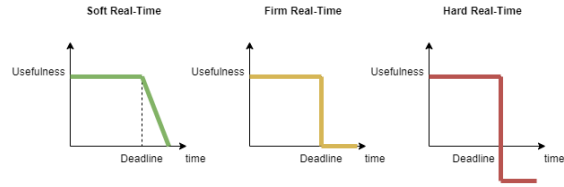
Bir sistem, bir veya daha fazla giriş kümesini ve girişlerle ilgili bir veya daha fazla çıkış kümesini içeren bir kara kutudur. Determinizmin sözlükteki anlamı, tüm olayların önceden var olan nedenlerle tamamen belirlenmesidir [6]. Deterministik bir sistem aynı zamanda belirli bir girdiden her zaman aynı çıktıyı üreten bir sistemdir. Öngörülebilirlik otomotiv endüstrisi için önemli bir özelliktir. Bu nedenle, deterministik sistemler bu endüstride yaygın olarak kullanılmaktadır.

Belirli bir zamanda yanıt veren ve görevleri zamanlayan sistemler gerçek zamanlı sistemlerdir [7][8]. Sürücüsüz araçları düşündüğümüzde kameralar ve sensörler aracılığıyla yayaaları saptarlar. Bu araçlar hızlarını tespit edilen yayalara göre ayarlarlar. Aracın yavaşlaması için acil bir istek gelebilir. Bu istek hızlı bir şekilde MİB ile haberleştirmeyi gerektirir. Gerçek zamanlı sistemler istenilen zamanda bu isteğe yanıt verebilmeye yeteneğine sahiptir. Şekil 4'te sistemlerin kullanım alanları gerçek-zamanlı olmayandan katı gerçek-zamanlıya spektrum şeklinde gösterilmektedir.



Şekil 4: Kullanım alanlarının spektrumu.

Zamanın güvenli araç sistemi sağlamak için çok önemli bir unsur olduğu önceki kısımlarda belirtildi. Zaman, sistemleri gerçek zamanlı ve diğer tip sistemler olarak ayıran ana unsurdur. Gerçek zamanlı sistemler, olaylara belirli bir zaman kısıtlaması içinde tepki vermesi gereken özel sistemlerdir [9]. Bir reaksiyon çok geç ortaya çıkarsa, çok tehlikeli sonuçlara yol açabilir. Sistemler, sebep oldukları tehlikelere göre üç farklı kategoride sınıflandırılmaktadır. Bunlar esnek gerçek zamanlı sistemler, sıkı gerçek zamanlı sistemler ve katı gerçek zamanlı sistemlerdir.



Şekil 5: Gerçek zamanlı sistemlerin davranışı.

Esnek gerçek zamanlı sistem, sistem zaman sınırından sonra yanıt verdiğinde yalnızca performans düşüşüne neden olan bir sistemdir. Sıkı gerçek zamanlı sistemi, esnek ve katı gerçek zamanlı sisteme göre bir ara sistemdir. Bu sistemdeki seyrek zaman sınırı kaçırımları tolere edilebilir. Toleransı en düşük sistem katı gerçek zamanlı sistemdir. Zaman sınırından sonra verilen bir yanıt sistemde yıkıcı sonuçlara neden olabilir. Katı ve esnek gerçek zamanlı sistemler arasındaki genel karşılaştırma Tablo 1'de verilmiştir.

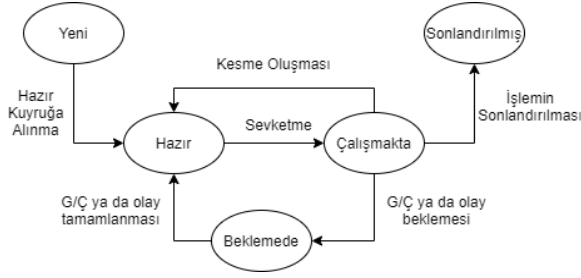
Tablo 1: Katı ve esnek gerçek-zamanlı sistem karşılaştırması

Parametre	Esnek Gerçek-Zamanlı Sistem	Katı Gerçek-Zamanlı Sistem
Yanıt Zamanı	Esnek-Tolere edilebilir	Sıkı-Gerekli
Güvenlik	Kritik değildir	Kritik
Zaman kısıt kaçırıldığında ki sonuçlar	Sistem kalitesinde tolere edilebilir düşme	Sisteme ciddi hasar oluşturma

Katı gerçek zamanlı sistemler [10], ASIL-D sınıfı gereklilikleri sağlamak için otomotiv endüstrisinde kullanılmaktadır. Bu sistemlerin ASIL-A'dan ASIL-D'ye kadar derecesi olan birçok görevi vardır. Her bilgisayarda, çalışan yazılımın içindeki komutları işleyen bir MİB bulunur. Bu MİB, güvenlik sınıfına göre bu görevleri yönetmelidir. Bu amaçla MİB, görev zamanlaması sağlamak için zamanlama algoritmasına sahip zamanlayıcı kullanır.

MİB, işlemler arasında geçiş yaparak bilgisayarı daha verimli hale getirir. Her zaman aralığında bir işlem yürütülmesi hedeflenmektedir. MİB, işlemlerde oluşacak bekleme durumlarında diğer işlemleri çalıştırır. Bellekte birçok süreç var. Bir işlem herhangi bir şekilde bekleme moduna geçtiğinde, MİB başka bir işleme geçer. İşlem yürütme, MİB yürütme ve Giriş / Çıkış (G / Ç) bekleme döngüsünü içerir. İşlemler bu iki durum arasında geçiş yapar. İşlemler MİB patlaması ile çalışmaya başlar ve G / Ç patlaması ile devam eder. Hesaplamalar için bir MİB patlaması gerçekleştirilir. Sistemler arasında veri aktarımını beklemek için bir G / Ç patlaması yapılır.

Her MİB Zamanlayıcısı'nın görev yürütme durumları için bir durum makinesi vardır. Örnek durum makinesi Şekil 6'da gösterilmiştir.



Şekil 6: Görev durum makinası.

MİB bekleme durumuna girdiğinde işletim sistemi hazır kuyruğundan çalıştırmak için bir işlem seçmek zorundadır. Bu seçim MİB zamanlayıcısı tarafından yapılmaktadır. Durumlar arasında geçiş için karar veren iki farklı zamanlama tipi vardır. Bunlar öncelikli ve önceliksiz zamanlamadır. Öncelikli zamanlamada MİB bir işlem için ayrıldıysa çalışır durumdan hazır duruma veya bekleme durumundan hazır duruma geçme kararı verebilir. Önceliksiz zamanlamada MİB bir işlem için ayrıldıysa işlem bitene kadar beklenir.

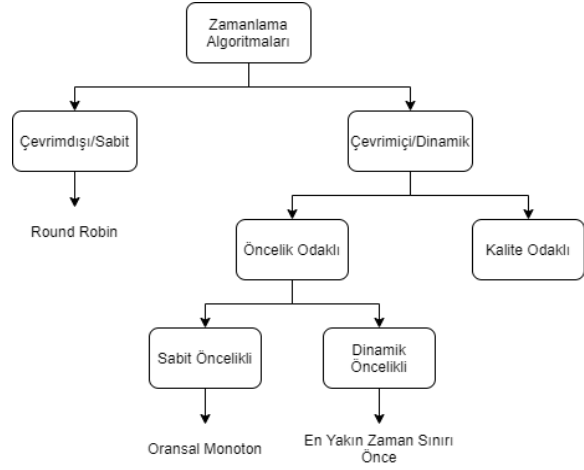
Görevlendirici MİB'ye atanacak işlemi seçen kısımdır. Görevlendiricinin çok hızlı geçiş yapması zorunludur. İşlemler arası geçiş süresine gönderim gecikmesi denir. MİB zamanlama algoritmaları, hazır kuyruğunda bekleyen işlemlerden hangisinin MİB'ye atanacağını belirler. MİB zamanlama algoritmaları birçok farklı kritere göre seçilir. Bunlar:

- MİB kullanımı
- Veri hacmi
- Bekleme süresi
- Yanıtlama süresi
- Dönüş süresi

MİB kullanımı ve veri hacmini maksimuma, bekleme ve yanıtlama süresini minimuma çekmek hedeflenir.

Temel olarak, programlama algoritmaları iki ana başlık altında kategorize edilir. Bunlar çevrimdışı / statik ve çevrimiçi / dinamik zamanlamadır [11]. Çevrimdışı zamanlamada, görevlerin listesini ve etkinleştirme zamanlarını içeren bir zamanlama tablosu vardır. Çalışma zamanında, basit bir dağıtıcı tabloda gösterilen kararları yürütür. Round Robin, çevrimdışı zamanlama için kullanılan bir zamanlama algoritmasıdır. Çevrimiçi zamanlamada, önceden tanımlanmış bir dizi kural vardır. Çalışma zamanında, görev dağıtıcı belirli bir görev kümesine uygulamak için bu önceden tanımlanmış kurallara dayalı bir karar alır.

Çevrimiçi planlama iki bölüme ayrılabilir; bunlar öncelik odaklı ve kalite odaklıdır [12]. Öncelik odağı, katı gerçek zamanlı sistemlerde yaygın bir rol oynamaktadır. Sabit öncelikli bir algoritma, her görevdeki tüm işlere aynı önceliği atar. Öte yandan, dinamik öncelik algoritması her görevdeki bağımsız işlere farklı öncelikler atar. OM ve EYZSÖ, sırasıyla sabit öncelik ve dinamik öncelik zamanlaması için popüler zamanlama algoritmalarıdır.



Şekil 7: Zamanlama algoritmalarının sınıflandırılması.

4. Zamanlama Algoritmaları

Zamanlama algoritmalarının sınıflandırılması önceki bölümde yapılmıştır. Bu zamanlama algoritmaları ve ayrıntıları bu bölümde verilecektir. Şekil 7'ye baktığımızda iki ana kısım bulunmaktadır. Bunlardan biri dinamik kısımdır. Kurallar olayların özelliklerine göre çalışır zamanda işlenmesinden dolayı bu kısma olay tetiklemeli zamanlama mimarisi adını verebiliriz. Diğer kısım statiktir. Görevlerin EKDYSL'eri derleme zamanında belirlendiği ve çalışır zamanda değişmediğinden bu bölüme de zaman tetiklemeli zamanlama mimarisi diyebiliriz.

4.1. Olay Tetiklemeli Zamanlama

Olay tetiklemeli yaklaşımda bir olay diğer bir olay tarafından tetiklenerek başlar. Olay tetiklemeli ve zaman tetiklemeli sistemlerde temel fark olay tetiklemeli sistemlerin hangi görevi işleyeceğine çalışır zamanda karar vermesidir. Kötümser yaklaşımlar her zaman sistemin işlevselliğini doğru bir şekilde ele almayı sağlar. Fakat bu yaklaşım olay-tetiklemeli sistemlerin davranışına kıyasla kaynak gereksinimlerinin gereğinden fazla tahmin edilmesine sebep olur. Bunun yanı sıra en kötü durum senaryosunda doğru sistem davranışını garantilemek için zamanlama analiz metodları da olay tetiklemeli sistemlerde uygulanmaktadır. Zamanlama analiz yöntemlerinin temelinde kötü durum varsayımı ve periyodik işleme vardır. Böylelikle olay tetiklemeli sistemler kaynakları zaman tetiklemeli sistemlere nazaran daha etkili kullanmaktadır. Ancak tahmini kaynak gereksinimleri, katı gerçek zamanlı sistem gereksinimlerini karşılamak konusunda genellikle kötümserdir. Sonuç olarak güvenlik-kritik sistemler için olay tetiklemeli sistemler daha yüksek kaynak kullanımı konusunda tercih edilmemektedir. Buna rağmen olay tetiklemeli sistemler zaman tetiklemeli sistemlerden işlevsellik sağlayan olaylar için daha hızlı reaksiyon göstermektedir.

4.1.1. Oransal Monoton

Oransal Monoton, basit bir kurala sahip olan sabit öncelikli bir programlama algoritmasıdır [13]. Kural, görevlerin önceliklerinin derleme zamanında belirlenmesidir. Öncelikler zamanla değişmez. Öncelikleri, yürütme sıklıkları ile doğru orantılıdır. En kısa süreye sahip olan görev en yüksek önceliğe

sahiptir. Aşırı yüklenmelerde deterministik davranış vardır. Böylece görevler öncelik seviyesinden etkilenir. Bu nedenle, OM statik programlamaya EYZSÖ'den daha yakındır. Statik tarafa yaklaştıkça performans azalmaktadır. OM için üst sınır kullanımının hesaplamasında, Liu ve Layland kanıtlarından yararlanabiliriz [14]. Bir task çalıştırılma süresini gösteren C_i , çalıştırılma sıklığını gösteren T_i , ve task sayısını gösteren n ile tanımlanır. Liu ve Layland, Şekil 8'de gösterilen hesaplamayı formülize etmişlerdir:

$$\left(U = \sum_{i=1}^n \frac{C_i}{T_i} \right) < n(2^{\frac{1}{n}} - 1)$$

Şekil 8: MİB kullanımı üst sınır değeri.

Bu kanıtı göre MİB kullanım oranı U , n büyüdükçe $\ln(2)$ 'ye yakınsar. Yani OM, ancak yaklaşık toplam yük yüzde 69'dan fazla değilse uygulanabilir bir planlamayı garanti edebilir [14]. Hesaplama adımları Şekil 9'da verilmiştir.

$$\begin{aligned} \lim_{n \rightarrow \infty} n \left(2^{\frac{1}{n}} - 1 \right) &= \lim_{n \rightarrow \infty} \left(\frac{-1 + 2^{\frac{1}{n}}}{\frac{1}{n}} \right) \\ &= \lim_{n \rightarrow \infty} \left(\frac{-\ln 2 \cdot 2^{\frac{1}{n}}}{-\frac{1}{n^2}} \right) \\ &= \ln 2 \cdot \lim_{n \rightarrow \infty} \left(2^{\frac{1}{n}} \right) \\ &= \lim_{n \rightarrow \infty} (\ln 2 \cdot 2^{\frac{1}{n}}) \\ &= \ln 2 \cdot \lim_{n \rightarrow \infty} \left(e^{\frac{1}{n} \ln 2} \right) \\ &= \ln 2 = 0.69314 \end{aligned}$$

Şekil 9: OM geçerlilik hesabının adımları.

4.1.2. En Yakın Zaman Sınırı Önce

EYZSÖ bir dinamik öncelik zamanlama algoritmasıdır [15]. Bu algoritmanın kuralı, mutlak süre kısıtlarına göre yürütme görevini seçmesidir. Öncelik değişikliği olduğunda hazır kuyruk azalan önceliklere göre sıralanır. Bu nedenle, bu algoritmanın yükü daha yüksektir. Çekirdek desteği gerektirdiği karmaşık bir uygulamaya sahiptir. Öte yandan iyi ölçeklendirilebilir ve aynı zamanda seyrek gerçekleşen görevlerde iyi performans gösterir. Liu ve Layland'ın kanıtlarına göre [14], EYZSÖ tam işlemci kapasitesinden yararlanabilir.

4.1.3. OM ve EYZSÖ karşılaştırılması

Dinamik zamanlama altındaki öncelik odaklı zamanlama algoritmalarına baktığımızda statik ve dinamik olarak ikiye ayrılmaktadır. OM ve EYZSÖ sırasıyla statik ve dinamik için popüler zamanlama algoritmalarıdır. Bu iki algoritma arasındaki karşılaştırma [16] Tablo 2'de verilmiştir.

Tablo 2: OM ve EYZSÖ karşılaştırması.

Parametre	OM	EYZSÖ
Öncelik	Statik	Dinamik
Uygulama Eforu	Düşük	Yüksek
Kural Kriteri	Periyot	Zaman Kısıtı
MİB Kullanımı	Düşük (~0.69)	Tam

4.2. Zaman Tetiklemeli Zamanlama

Zaman tetiklemeli zamanlamada [17], görevler önceden tanımlanan noktalarda başlatılır. Çalışma zamanı dağıtımı bir dizi kurala göre gerçekleştirilir. Bu kurallar derleme zamanında hazırlanan çizelge tablosunda tanımlanır. Dağıtıcı çizelgeleme kararlarını bu tabloya göre alır. Bu nedenle, çalışma zamanında daha düşük sistem yükü vardır. Bu aynı zamanda esnekliğin olmaması gibi bir dezavantajı da beraberinde getirir.

Bu zamanlama için, çalışma zamanından önce her şey bilinmelidir [18][19]. Bu da bir maliyet oluşturur. Ancak, geliştiriciler için önemli bir nitelik olan determinizmi sağlar.

5. Katı Gerçek-Zamanlı Sistemler

İşletim sistemi standartları, uygulamanın bir platformdan diğerine taşınabilirliğini sağlayabilir. Ayrıca, tek bir uygulama için birkaç çekirdek sağlayıcısına sahip olma olanağına izin verebilirler. Bu nedenle, bu standartlar çok önemlidir. Kullanıldıkları yere göre sınıflandırılabilirler. Gerçek Zamanlı Taşınabilir İşletim Sistemi Arabirimi, genel amaçlı işletim sistemi standardının gerçek zamanlı uzantısı olan popüler bir arayüzdür. APEX ise aviyonik sistemlerde yaygındır.

Offene Systeme und deren Schnittstellen für die Elektronik in Kraftfahrzeugen (OSEK), Alman otomotiv şirketi gömülü sistemler konsorsiyumu tarafından kurulmuş bir standarttır [20]. Bu standartta bir iletişim yığını, bir ağ yönetim protokolü ve diğer ilgili konular hakkında spesifikasyonlar üretilmiştir. EKB'ler için standart yazılım mimarisi olarak tasarlanmıştır. Fransız otomobil imalatçıları, Vehicle Distributed eXecutive (VDX) adında benzer bir projeye sahipti. VDX, konsorsiyuma katıldıktan sonra resmi isim OSEK/VDX olmuştur.

Piyasada yer alan çok sayıda işletim sisteminin bir kısmı otomotiv açısından bazı dezavantajlara sahiptir. Genellikle gerçek zamanlı performansları yeterli değildir. Uygulama Programlama Arabirimleri (UPA), eşzamanlı ve eşzamansız görev zamanlamayı birlikte kullanmak için uygun değildir. OSEK UPA, otomotiv endüstrisi için özel olarak tasarlanmıştır. Bu nedenlerden ötürü, OSEK İşletim Sistemi otomotiv gömülü sistemlerinde kullanılmaktadır.

Bunlara ek olarak OSEK / VDX'in birçok avantajı vardır. Uygulamadan bağımsız bir mimariye sahip olduğu için uygulama yazılımının taşınabilirliğini ve tekrar kullanılabilirliğini destekler. Arayüzlerin özellikleri donanım ve ağdan bağımsızdır. Bu avantaj, bir uygulamanın farklı EKB'lerde kullanılmasını sağlar. Mimarisi verimli bir şekilde tasarlandığından, mevcut işlevler yapılandırılabilir ve ölçeklenebilir. Ayrıca sisteme kolayca yeni işlevler eklenebilir.

Standartlara uygunluk, standardın şartlarını desteklediği anlamına gelir. Erika Enterprise açık kaynaklı bir OSEK/VDX

katı gerçek zamanlı işletim sistemidir [21]. AUTomotive Açık Sistem Mimarisi (AUTOSAR) UPA'sinin bir alt kümesinden esinlenen bir UPA uygular. Erika 2012 yılında sertifikalı ve daha sonra Erika'ya AUTOSAR İşletim Sistemi spesifikasyonları uygulanmıştır. OSEK/VDX uyumlu sertifikalı ilk açık kaynaklı ve ücretsiz gerçek zamanlı işletim sistemidir. Olay tetikleyici mimariyi ve zaman tetikleyici mimariyi destekler. Bu işletim sistemi 1-4 Kb Flash ayak izi ve 8-32 bit mikrodenetleyiciler için uygundur. Ayrıca, ayak izini sınırlamak için kullanılan uygunluk sınıflarına sahiptir. En önemli avantajlarından biri, bir OSEK Uygulama Dili (OD) dosyası [22] veya AUTOSAR Genişletilebilir İşaretleme Dili ile statik olarak yapılandırılabilir.

6. Uygulama ve Değerlendirme

Tek işlemcili sistemler, programlama algoritmalarını değerlendirmek için iyi bir seçimdir. Görevler arasındaki bağlam geçişleri, kaçırılmış zaman sınırları ve her görev için MİB kullanımları bu sistemlerde açıkça görülebilir. Böylece, Arduino Uno uygulamak için seçilebilir. Erika İşletim Sistemi'nin bu kart için desteğinin bulunması da iyi bir avantajdır.

Arduino Uno, Atmega328 işlemci kullanan ve kullanımı oldukça kolay bir karttır [23]. 14 sayısal giriş çıkış pini vardır, bunların 6'sı darbe genişliği modülasyonu çıkışı (PWM) olarak kullanılabilir. 16 MHz kristal osilatör, evrensel seri veri yolu bağlantısı, 2.1 mm güç girişi, devre içi seri programlama başlığı ve sıfırlama düğmesi vardır. Çalıştırmak için DC 7 ~ 12V güç kaynağına bağlanması yeterlidir.

Erika İşletim Sistemi resmi web sitesinden indirilmiştir. Bu işletim sistemi, Erika'nın geliştiricileri tarafından Erika resmi web sitesinde yayınlanan talimat kılavuzuna göre Arduino Uno kartına inşa edilerek gömülmüştür. Daha sonra ışık yayan diyotun yanıp sönmeye gibi bazı örneklerle test edilmiştir. Programlama algoritmalarının artılarını ve eksilerini göstermek için bir senaryo belirlenmiştir. Bu senaryoda sabit öncelikleri olan 3 görev vardır. Spesifikasyonları Tablo 3'te gösterilmiştir.

Tablo 3: Senaryoda kullanılan parametreler

Görev	Çalışma Süresi (s)	Periyot (s)
Task1	1	4
Task2	2	6
Task3	3	8

Bu görevlerin MİB Kullanımı Şekil 8'de belirtilen formüle göre hesaplanır. Hesaplama Şekil 10'de gösterilmiştir.

$$U = \frac{1}{4} + \frac{2}{6} + \frac{3}{8} = 0.96$$

Şekil 10: Senaryoya göre MİB kullanım hesaplaması.

İşletim sistemini yapılandırmak için Bölüm 5'te belirtilen bir OD dosyasına sahiptir. Bu dosya istenen ve gerekli konfigürasyona göre yazılmıştır. Her konfigürasyon algoritması için bazı konfigürasyonlar kolayca değiştirilebilir. OM ilk analiz zamanlama algoritmasıydı. Sabit öncelikli bir algoritmadır. OD dosyası Şekil 11'deki gibi yapılandırılmıştır.

```
KERNEL_TYPE = FP;
};
TASK Task1 {
    PRIORITY = 3;
    STACK = SHARED;
    SCHEDULE = FULL;
    AUTOSTART = TRUE;
};
TASK Task2 {
    PRIORITY = 2;
    STACK = SHARED;
    SCHEDULE = FULL;
    AUTOSTART = TRUE;
};
TASK Task3 {
    PRIORITY = 1;
    STACK = SHARED;
    SCHEDULE = FULL;
    AUTOSTART = TRUE;
};
```

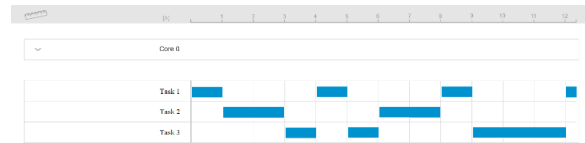
Şekil 11: OM için OD dosyasının ayarlanması.

Bu görevlerin içeriği, araçlardaki EKB uygulamaları gibi uygulama sürelerine göre ayarlanmıştır. Görevlerin yürütme sürelerini göstermek için görevlere zaman damgalı Evrensel Asenkron Alıcı Verici (UART) günlükleri eklenmiştir. OM için UART günlükleri Şekil 12'de görülmektedir.

```
14:32:13.313 -> Cpu Utilization: 0.96
14:32:13.313 -> HyperPeriod: 24 For 3 Task
14:32:14.319 -> TASK1_started
14:32:15.311 -> Task1_finished
14:32:15.311 -> Task2_started
14:32:16.321 -> Task2_continue
14:32:17.318 -> Task2_finished
14:32:17.318 -> Task3_started
14:32:18.339 -> Task3_continue
14:32:18.542 -> TASK1_started
14:32:19.524 -> Task1_finished
14:32:20.310 -> Task3_continue
14:32:20.644 -> Task2_started
14:32:21.632 -> Task2_continue
14:32:22.644 -> Task2_finished
14:32:22.745 -> TASK1_started
14:32:23.763 -> Task1_finished
14:32:24.293 -> Task3_finished
14:32:24.293 -> Task3_started
14:32:25.291 -> Task3_continue
14:32:26.301 -> Task3_continue
14:32:26.963 -> TASK1_started
14:32:27.971 -> Task1_finished
14:32:27.971 -> Task2_started
14:32:28.954 -> Task2_continue
14:32:29.983 -> Task2_finished
14:32:30.256 -> Task3_finished
14:32:31.167 -> TASK1_started
14:32:32.191 -> Task1_finished
14:32:32.191 -> Task3_started
```

Şekil 12: OM UART çıktısı.

Bu senaryonun sonuçlanan davranışı, OM zamanlama algoritmasının artılarını ve eksilerini göstermek için görselleştirilmiştir. İlgili görsel Şekil 13'te verilmiştir.



Şekil 13: OM çıktısının görselleştirilmesi.

Şekil 13'e göre, Görev 3 en düşük önceliğe sahiptir ve yürütme süresi 3 saniyedir ve 8. saniyede tamamlanması gerekirken zaman kısıtını kaçırmaktadır. Bu, OM için bir dezavantajdır, ancak en düşük öncelikli görev için zaman kısıtı kaçırılmıştır. Bu görev ASIL-A, araçta ışığın kapı durumunu göstermesi görevi gibi düşünülebilir. Bu görevin gecikmesi, bu sistemde yıkıcı sonuçlara neden olmaz. Bu zaman kısıtının kaçırılması, MİB kullanımını azaltarak ortadan kaldırılabilir. Bu analiz, bir sabit öncelikli zamanlama algoritması olan OM'nin performanstan fedakarlık ederek güvenli bir sistem oluşturmak için kullanılabilirliğini gösteriyor.

Bir sonraki zamanlama algoritması, Dinamik-Öncelikli Zamanlama Algoritması olan EYZSÖ'dür. Sabit öncelikliden daha az çaba gerektirir, çünkü görevleri zaman kısıtlarına göre otomatik olarak zamanlamaktadır. OUD dosyası EYZSÖ için yeniden düzenlenmiştir. KERNEL_TYPE değişkeni EYZSÖ olarak ayarlanmış olup Şekil 14'te gösterilmiştir.

```

KERNEL_TYPE = EDF;
};
TASK Task1 {
    PRIORITY = 3;
    STACK = SHARED;
    SCHEDULE = FULL;
    AUTOSTART = TRUE;
};
TASK Task2 {
    PRIORITY = 2;
    STACK = SHARED;
    SCHEDULE = FULL;
    AUTOSTART = TRUE;
};
TASK Task3 {
    PRIORITY = 1;
    STACK = SHARED;
    SCHEDULE = FULL;
    AUTOSTART = TRUE;
};

```

Şekil 14: EYZSÖ için OUD dosyasının ayarlanması.

Bu kez, EYZSÖ için aynı senaryo uygulanmaktadır. EYZSÖ için UART günlükleri Şekil 15'te verilmektedir.

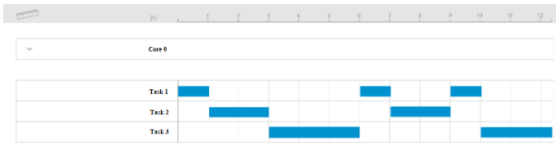
```

14:35:19.063 -> Cpu Utilization: 0.96
14:35:19.063 -> HyperPeriod: 24 For 3 Task
14:35:19.063 -> TASK1_started
14:35:20.060 -> TASK1_finished
14:35:20.060 -> TASK2_started
14:35:21.056 -> TASK2_continue
14:35:22.086 -> TASK2_finished
14:35:22.086 -> TASK3_started
14:35:23.082 -> TASK3_continue
14:35:24.078 -> TASK3_continue
14:35:25.073 -> TASK3_finished
14:35:25.073 -> TASK1_started
14:35:26.103 -> TASK1_finished
14:35:26.103 -> TASK2_started
14:35:27.098 -> TASK2_continue
14:35:28.095 -> TASK2_finished
14:35:28.095 -> TASK1_started
14:35:29.091 -> TASK1_finished
14:35:29.091 -> TASK3_started
14:35:30.121 -> TASK3_continue
14:35:31.117 -> TASK3_continue
14:35:32.114 -> TASK3_finished
14:35:32.114 -> TASK1_started
14:35:33.122 -> TASK1_finished
14:35:33.122 -> TASK2_started
14:35:34.151 -> TASK2_continue
14:35:35.129 -> TASK2_finished
14:35:35.129 -> TASK1_started
14:35:36.126 -> TASK1_finished
14:35:36.126 -> TASK3_started
14:35:37.155 -> TASK3_continue
14:35:38.150 -> TASK3_continue
14:35:39.147 -> TASK3_finished

```

Şekil 15: EYZSÖ UART çıktısı.

Bu senaryonun sonuçlanan davranışı EYZSÖ zamanlama algoritmasının artılarını ve eksilerini göstermek için görselleştirildi. Şekil 16'da verilmiştir.



Şekil 16: EYZSÖ çıktısının görselleştirilmesi.

Şekil 16'ya bakıldığında, kaçırılmış bir zaman kısıtı yoktur. Bu bize EYZSÖ'nin sabit öncelikliden daha yüksek MİB kullanımı sağlayabildiğini göstermektedir. Ancak bu EYZSÖ'nin dezavantajları olmadığını göstermez. Zaman kısıtı zamanlayıcı, her görev için zaman kısıtı içerisinde yerine getirilmesini garanti eder, ancak herhangi bir görev için minimum yanıt

süresi sağlamak mümkün değildir. En yüksek öncelikli görev her zaman OM cinsinden minimum yanıt süresine sahiptir, ancak EYZSÖ için garanti vermek mümkün değildir. Yeni bir görev veya yanlış EKDYS tahmini nedeniyle sistem aşırı yüklenirse, domino etkisi oluşabilir. Domino etkisi, bir görev beyan edilen çalışma süresinden daha fazlasını yürütmek için zaman kısıtını kaçırdıktan sonra diğer tüm görevlerin zaman kısıtlarını kaçırmalarıdır. Bu, çarpışma anında hava yastıklarının açan bir görev gibi ASIL-D görevinin zaman kısıtını da kaçırmaları olasılığıdır. Bu durum, bu sistemde yıkıcı sonuçlara neden olur. Bu nedenle EYZSÖ, yüksek performans ve düşük güvenlik gerektiren görevler için kullanılabilir.

Bunlar çevrimiçi zamanlama algoritmalarının artıları ve eksileriydi. Otomotiv endüstrisinin performans ve güvenlik olarak sadece iki parametresi yoktur. Determinizm ve öngörülebilirlik, otomotiv endüstrisinde çalışan geliştiriciler için diğer önemli parametrelerdir. Bu nedenle, zaman tetikleyici planlama mimarisi bu sektörde önemlidir [19]. Bu mimaride çevrimdışı tanımlanmış görevleri ve bunların tetiklenme zamanlarını içeren zamanlama tablosu bulunur. Bu zamanlama tablosu OUD dosyasında hazırlanmıştır ve her görev belirli bir zamanda çağrılmaktadır. ZT için UART günlükleri Şekil 17'de görülmektedir.

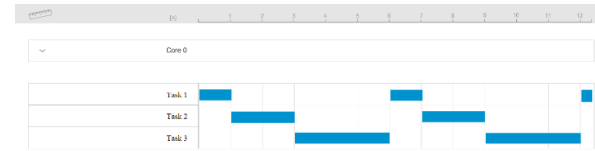
```

15:09:22.443 -> Cpu Utilization: 0.96
15:09:22.443 -> HyperPeriod: 24 For 3 Task
15:09:22.941 -> TASK1_started
15:09:23.969 -> TASK1_finished
15:09:23.969 -> TASK2_started
15:09:24.966 -> TASK2_continue
15:09:25.962 -> TASK2_finished
15:09:25.962 -> TASK3_started
15:09:26.958 -> TASK3_continue
15:09:27.988 -> TASK3_continue
15:09:28.985 -> TASK3_finished
15:09:28.985 -> TASK1_started
15:09:29.982 -> TASK1_finished
15:09:29.982 -> TASK2_started
15:09:30.977 -> TASK2_continue
15:09:32.007 -> TASK2_finished
15:09:32.007 -> TASK3_started
15:09:33.003 -> TASK3_continue
15:09:33.999 -> TASK3_continue
15:09:34.995 -> TASK3_finished
15:09:34.995 -> TASK1_started
15:09:36.024 -> TASK1_finished
15:09:36.024 -> TASK2_started
15:09:37.020 -> TASK2_continue
15:09:38.016 -> TASK2_finished
15:09:38.016 -> TASK3_started
15:09:39.011 -> TASK3_continue
15:09:40.040 -> TASK3_continue
15:09:41.023 -> TASK3_finished

```

Şekil 17: ZT UART çıktısı.

Bu senaryonun sonuçlanan davranışı ZT zamanlama algoritmasının artılarını ve eksilerini göstermek için görselleştirilmiş olup Şekil 18'de verilmektedir.



Şekil 18: ZT çıktısının görselleştirilmesi.

Olay tetiklemeli zamanlama mimarisi ile zaman tetiklemeli zamanlama mimarisi arasında bir karşılaştırma yapmak gerekirse; Olay tetiklemeli mimari daha esnek ve yüksek kaynak kullanımı doğurur. Öte yandan bu, katı gerçek zamanlı sistemlerde önemli olan determinizm ve öngörülebilirlik açısından güvenilir değildir. Ayrıca, tetiklenen olay tetiklenen süreden daha yüksek çalışma zamanı yüküne sahiptir.

Tetiklenen zamanın periyodik bir dünyası vardır. Sporadik görevler için esnek değildir. Determinizm ve öngörülebilirlik sağlar, ancak kaynak kullanımı oranı daha düşüktür.

Benzer işlem farklı MİB kullanımları ile daha fazla görev için denendi. Sonuçlar Tablo 4'te listelenmiştir.

Tablo 4: Farklı parametrelere göre çıktılar.

	MİB Kullanımı (%)	Görev Sayısı	Başarısız Görev Sayısı	Başarılı Görev Sayısı	Başarı Oranı (%)
OM	<%85	5	0	5	%100
EYZSÖ			0	5	%100
ZT			0	5	%100
OM		10	0	10	%100
EYZSÖ			0	10	%100
ZT			0	10	%100
OM		15	0	15	%100
EYZSÖ			0	15	%100
ZT			0	15	%100
OM		20	0	20	%100
EYZSÖ			0	20	%100
ZT			0	20	%100
OM		25	0	25	%100
EYZSÖ			0	25	%100
ZT			0	25	%100
OM		30	0	30	%100
EYZSÖ			0	30	%100
ZT			0	30	%100
OM	%85- %98	5	1	4	%80
EYZSÖ			0	5	%100
ZT			0	5	%100
OM		10	2	8	%80
EYZSÖ			0	10	%100
ZT			0	10	%100
OM		15	3	12	%80
EYZSÖ			0	15	%100
ZT			0	15	%100
OM		20	3	17	%85
EYZSÖ			0	20	%100
ZT			0	20	%100
OM		25	4	21	%84
EYZSÖ			0	25	%100
ZT			0	25	%100
OM		30	4	26	%86.7
EYZSÖ			0	30	%100
ZT			0	30	%100
OM	%98- %100	5	1	4	%80
EYZSÖ			0	5	%100
ZT			0	5	%100
OM		10	3	7	%70
EYZSÖ			0	10	%100
ZT			0	10	%100
OM		15	4	11	%73.3
EYZSÖ			0	15	%100
ZT			0	15	%100
OM		20	4	16	%80
EYZSÖ			0	20	%100

ZT	25	0	20	%100
OM		5	20	%80
EYZSÖ		0	25	%100
ZT	30	0	25	%100
OM		5	25	%83.3
EYZSÖ		0	30	%100
ZT		0	30	%100

Önceki deneyde olduğu gibi OM MİB kullanımı arttıkça başarı oranı düşmeye başlayan ilk zamanlama algoritması olmuştur. Sonuçlara göre görev sayısı arttıkça zaman kısıtını aşma eşiği düşmektedir. Bu aynı zamanda Şekil 8'de verilen formülü kanıtlamaktadır. EYZSÖ olay tetiklemeli zamanlama algoritması olarak daha yüksek MİB kullanımında OM'den daha başarılı bir algoritmadır. Daha önce de bahsettiğimiz gibi ZT bazı dezavantajlara sahip olsa da periyodik dünyada en başarılı algoritmadır.

Olay tetiklemeli ve zaman tetiklemeli zamanlama mimarileri karşılaştırıldığında olay tetiklemeli mimarinin daha esnek olduğu ve yüksek kaynak kullanımı doğurması göze çarpmaktadır. Fakat katı gerçek zamanlı sistemler açısından önemli olan determinizm ve öngörülebilirlik açısından güvenilir değildir. Ayrıca olay tetiklemeli mimari zaman tetiklemeli mimariden daha fazla çalışır zaman yüküne sahiptir. Zaman tetiklemeli periyodik bir dünyaya sahiptir. Sporadik görevler için esnek değildir. Düşük kaynak kullanım oranlarında determinizm ve öngörülebilirlik sağlar.

7. Sonuç

Bu makalede, programlama algoritmaları üç kategoride sınıflandırılmıştır. Bunlar sabit öncelikli zamanlama, dinamik öncelikli zamanlama ve zaman tetikleyici zamanlamadır. Bu programlama algoritmaları otomotiv endüstrisinde önemli olan parametrelere göre analiz edilmiştir. Analiz sonucu, her zamanlama algoritmasının avantajları ve dezavantajları olduğu ve gereksinimlere göre seçilmesi gerektiğidir.

Güvenlik açısından kritik sistemler gibi sıkı güvenlik gereksinimlerine sahip sistemler için Oransal Monoton algoritması iyi bir seçim olacaktır. Daha fazla MİB kullanımı sağlamasından dolayı işlevselliği sağlamayı amaçlayan sistemlerin performans ihtiyacını karşılaması için en yakın zaman sınırı algoritmasını kullanması doğru seçim olacaktır. Deterministik sistemler öngörülebilirlik sağlamak adına zaman tetiklemeli algoritmayı tercih edebilirler.

Çıkarımlardan bir diğeri, iki farklı zamanlama algoritması için bir kartta iki MİB kullanılabilir. Biri performans için, diğeri güvenlik için seçilebilir. Görevler ASIL düzeylerine göre ilgili MİB'lere dağıtılabilir. Başka bir çıkarım ise birden fazla zaman tablosu tanımlanması ve bu zaman tabloları arasında karar veren bir yapay zekanın geliştirilmesidir. Bu yapay zeka çalışma esnasında sporadik görevlerin üstesinden gelmek için gerekli kararları verebilir.

Gelecekteki çalışmalar için bu analiz çok çekirdekli sistemler için yapılabilir. Her bir çekirdek farklı ASIL seviyelerini iletlemek için ayrılabilir.

8. Kaynaklar

- [1] B. Saydam and T. Ayav, "Evaluation of Scheduling Architectures for OSEK/VDX Compliant Hard Real-Time Operating Systems," 2020 Turkish National Software Engineering Symposium (UYMS), 2020, pp. 1-6, doi: 10.1109/UYMS50627.2020.9247064.
- [2] ISO: 26262 – "Road vehicles-Function safety Part 6: Product development at the software level", 2018.
- [3] Redmill, Felix. "IEC 61508-principles and use in the management of safety." *Computing & Control Engineering Journal*, 9.5, pp. 205-213, 1998.
- [4] Wang, Dafang & Song, Peng & Xu, Zexu & Dong, Guanglin & Wei, Hui, "Conceptual Design of Functional Safety of Motor Control System Based on ISO26262", *MATEC Web of Conferences*, 173. 02045. 10.1051/mateconf/201817302045, 2018.
- [5] Introduction to MISRA C, URL: <https://www.embedded.com/introduction-to-misra-c/> (Last accessed: Jan, 5, 2022).
- [6] Phillip Laplante, *Real-Time Systems Design and Analysis - An Engineer's Handbook*, IEEE Press, 1993.
- [7] John A. Stankovic, Krithi Ramamritham, and Marco Spuri, *Deadline Scheduling for Real-Time Systems: Edf and Related Algorithms*, Kluwer Academic Publishers, USA, 1998.
- [8] M. Caccamo, T. Baker, A. Burns, and G. Buttazzo, "Real-time scheduling for embedded systems," in *Handbook of Networked and Embedded Systems*, pp. 173-195, Birkhäuser Boston, 2005.
- [9] Buttazzo, Giorgio C. *Hard real-time computing systems: predictable scheduling algorithms and applications*. Vol. 24. Springer Science & Business Media, 2011.
- [10] A. Burns, "Scheduling hard real-time systems: a review", *Software Engineering Journal*, 6(3), 116-128, 1991.
- [11] Weirong Wang, Aloysius K. Mok, Gerhard Fohler, "Pre-Scheduling: Integrating Offline and Online Scheduling Techniques", *EMSOFT 2003*, Volume 2855, pp. 356-372, 2003.
- [12] D. Rajesh, "Real-time scheduler design for safety-critical systems : A supervisory control approach", PhD dissertation, 2018.
- [13] Bini, E., Buttazzo, G. C., & Buttazzo, G. M. "Rate monotonic scheduling: The hyperbolic bound", *IEEE Transactions on Computers*, Volume 52, Issue 7, 2003, pp. 933-942, <https://doi.org/10.1109/TC.2003.1214341>.
- [14] C. Liu and J. Layland, "Scheduling Algorithms for Multiprogramming in a Hard Real-time Environment", *JACM*, vol. 20, no. 1, 1973.
- [15] Zhang F, "Analysis for EDF scheduled real-time systems", PhD thesis, Dept of Computer Science, University of York, UK, 2009.
- [16] Buttazzo, G. "Rate monotonic vs. EDF: Judgment day", *EMSOFT 2003*, Volume 2855, pp. 67-83, 2003.
- [17] Kopetz, Hermann, and Günther Bauer. "The time-triggered architecture." *Proceedings of the IEEE 91.1* : 112-126, January 2003.
- [18] Kopetz, Hermann. *Real-time systems: design principles for distributed embedded applications*. Springer Science & Business Media, 2011.
- [19] Kopetz, Hermann. "Event-triggered versus time-triggered real-time systems." *Operating Systems of the 90s and Beyond*. Springer, Berlin, Heidelberg, pp. 86-101, 1991.
- [20] Lemieux, Joseph. "The OSEK/VDX Standard: Operating System and Communication." *Embedded Systems Programming* 13.3: pp. 90-109, 2000.
- [21] Erika-enterprise.com. URL:<http://www.erika-enterprise.com/index.php/community.html/> (Last accessed: Apr, 11, 2020).
- [22] Feiler, Peter H. (2018): *Real-Time Application Development with OSEK: A Review of the OSEK Standards*. Carnegie Mellon University. Report. <https://doi.org/10.1184/R1/6582899.v1>.
- [23] Arduino.cc, URL: <https://www.arduino.cc/> (Last accessed: May, 19, 2020).

Özgeçmişler



Berkay SAYDAM, İzmir doğumlu olan Berkay SAYDAM, Dokuz Eylül Üniversitesi Elektrik-Elektronik Mühendisliği bölümü lisans eğitimini 2017 yılında onur öğrencisi olarak tamamlamıştır. 2020 yılında İzmir Yüksek Teknoloji Enstitüsü Bilgisayar Mühendisliği bölümü yüksek lisans eğitimini yüksek onur öğrencisi olarak tamamlamıştır ve aynı bölümde doktora eğitimini sürdürmektedir. Lisans öğrenimi mezuniyetinden itibaren özel sektör firmalarının araştırma geliştirme departmanlarında çeşitli pozisyonlarda görev almış ve sektörde çalışmalarına devam etmektedir. Multidisipliner yaklaşım benimsemesinden kaynaklı gömülü sistemler ve nesnelerin interneti alanlarında çalışmasının yanı sıra araştırma alanları arasında veri madenciliği ve makine öğrenmesi de bulunmaktadır.



Tolga AYAV 1995 yılında Dokuz Eylül Üniversitesi Elektrik-Elektronik Mühendisliği Bölümü'nden mezun olmuştur. Endüstriyel otomasyon alanında özel sektörde çalıştıktan sonra 1999'da İzmir Yüksek Teknoloji Enstitüsü Bilgisayar Mühendisliği'nden yüksek lisans ve 2004'te Ege Üniversitesi Bilgisayar Mühendisliği'nden doktora derecelerini almıştır. 2004-2006 süresince INRIA Rhone-Alpes'te doktora sonrası araştırmacı olarak çalışmıştır. Dr. Ayav 2006 yılından itibaren İYTE Bilgisayar Mühendisliği Bölümü'nde öğretim üyesi olarak akademik çalışmalarını sürdürmektedir. Akademik araştırmaları ağırlıklı olarak gerçek zamanlı ve gömülü sistemler, hataya dayanıklılık, yazılım test ve sinama üzerinedir.
