

FEN BİLİMLERİ ENSTİTÜSÜ  
BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI



BİLGİ İŞLEMSEL DÜŞÜNME BECERİLERİNİN MAKİNA ÖĞRENMESİ  
KULLANARAK PERFORMANSA DAYALI DEĞERLENDİRİLMESİ

YÜKSEK LİSANS TEZİ

ELİF KARAKAŞ

tarafından

YÜKSEK LİSANS

derecesi şartını sağlamak için hazırlanmıştır.

Ağustos 2019

Program: Bilgisayar Mühendisliği

BİLGİ İŞLEMSEL DÜŞÜNME BECERİLERİNİN MAKİNA ÖĞRENMESİ  
KULLANARAK PERFORMANSA DAYALI DEĞERLENDİRİLMESİ

YÜKSEK LİSANS TEZİ

ELİF KARAKAŞ

tarafından

İSTANBUL OKAN ÜNİVERSİTESİ

Bilgisayar Mühendisliği Anabilim Dalına

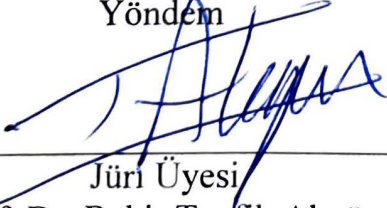
Yüksek Lisans

derecesi şartını sağlamak için sunulmuştur.

Onaylayan:



Danışman  
Dr. Öğr. Üyesi Meltem Turhan  
Yönetim



Jüri Üyesi  
Prof. Dr. Bekir Tefik Akgun



Jüri Üyesi  
Doç.Dr. Sanem Sariel  
(İstanbul Teknik Üniversitesi)

Ağustos 2019

Program: Bilgisayar Mühendisliği

## ÖZET

Bu tez çalışması Makine Öğrenmesi tekniklerini kullanarak, kullanıcının Bilgi İşlemsel Düşünme (BİD) becerilerini Blok Tabanlı Programlama (BTP) araçlarındaki performansına dayalı olarak ölçen bir değerlendirme aracıdır. Değerlendirme aracı, Makine Öğrenmesi modeli ile Bilgi İşlemsel Düşünme becerilerinin 7 hepsi için tek tek değerlendirme yapmaktadır. Geliştirilen araç ile Blok Tabanlı Programlama dillerinde sınırlı olan değerlendirme araçlarına hem alternatif üretmek hem de tarafsız ve şeffaf değerlendirme yapmak amaçlanmıştır.

Literatür taraması, Makine Öğrenmesi ve Blok Tabanlı Programlama dilleri olmak üzere iki yönlü yapılmıştır. İlk aşamada Makine Öğrenmesinde eğitim alanında model eğitilirken kullanılan veri setleri ve yöntemler, ikinci aşamada Blok Tabanlı Programlama dillerinde performansı ölçerken kullanılan araçlar incelenmiştir.

Literatürde Bilgi İşlemsel Düşünme becerilerinin öğretilmesinde çeşitli kaynakların olmasının yanı sıra BİD becerilerini ölçen araçların yetersizliği ile karşılaşmıştır. BİD becerilerini değerlendirmek için bir Makine Öğrenmesi modeli eğitilmiştir. Eğitilen model hem öğretmenlerin hem de öğrencilerin rahatlıkla BİD becerilerini değerlendirebilecekleri çevrimiçi değerlendirme aracı haline getirilmiştir. Son aşamada geliştirilen çevrimiçi değerlendirme aracı ve kullanılması incelenmiştir.

**Keywords:** Makine Öğrenmesi, Blok Tabanlı Programlama dilleri, Bilgi İşlemsel Düşünme becerileri

## ABSTRACT

This thesis study is an evaluation tool that measures the user's Computational Thinking (CT) according to its performance on Block-Based Programming Languages (BBPL) by using Machine Learning Techniques. The evaluation tool makes an evaluation for Machine Learning Model and each of the Computational Thinking seven fields. Via this developed tool, the objective is to product an alternative to the limited evaluation tools on Block-Based Programming Languages.

Literature research was brought out in two sides as Machine Learning and Block-Based Programming Languages. In the first stage, on Machine Learning, the data set and the methods used in educational field were researched. In the second stage, the devices which are used to measure the performance on Block- Based Programming Languages were analyzed.

On literature, there are various sources for teaching Computational Thinking ,but it occurs that measuring instruments that evaluates Computational Thinking are inadequate. To evaluate Computational Thinking a machine learning model was trained. Trained model was turned into online evaluation tool by which both the tutors and the students evaluate Computational Thinking easily. In the final stage, the developed online evaluation tool and the outcome was analyzed.

**Keywords:** Machine Learning, Block-Based Programming Languages, Computational Thinking(CT)



“Annem Figen KARAKAŞ, Babam Hasan KARAKAŞ ve kardeşim Mustafa Karakaş başta  
olmak üzere, tüm sevdiklerime...”



## TEŞEKKÜR

Kendisiyle çalışmanın başlı başına bambaşka bir deneyim olduğu değerli hocam Dr. Meltem Turhan Yöndem'e tez çalışmamı hazırladığım süre boyunca göstermiş anlayış, hoş görü ve bilgi birikimini paylaşmasından ötürü çok teşekkür ederim.

Yüksek Lisans eğitimim boyunca her türlü akademik desteklerini esirgemeyen ve çok değerli katkıları olan kıymetli hocam Prof. Dr. Bekir Tevfik Akgün'e verdiği eğitimler ve katkıları için teşekkür ederim.

Tüm süreç boyunca beni her zaman her durumda destekleyen, zor zamanlarımda dahi beni cesaretlendiren başta Annem Figen KARAKAŞ, Babam Hasan KARAKAŞ ve kardeşim Mustafa KARAKAŞ'a teşekkürü borç bilirim.

Elif KARAKAŞ

Mayıs, 2019

# İÇİNDEKİLER

ÖZET .....	i
ABSTRACT .....	ii
TEŞEKKÜR .....	iv
İÇİNDEKİLER.....	v
TABLO LİSTESİ .....	viii
ŞEKİL LİSTESİ.....	x
KISALTMALAR .....	xii
I. GİRİŞ .....	1
II. MAKİNE ÖĞRENMESİ .....	3
1.1. GÖZETİMLİ ÖĞRENME (SUPERVISED LEARNING).....	4
1.1.1. Regresyon (Regressions).....	5
1.1.1.1. Basit Doğrusal Regresyon (Simple Linear Regression).....	5
1.1.2. Sınıflandırma (Classification) .....	5
1.1.2.1. Derin Sinir Ağları (Deep Neural Networks) .....	5
1.2. GÖZETİMSİZ ÖĞRENME (UNSUPERVISED LEARNING) .....	6
1.2.1. Kümeleme.....	6
1.2.2. Boyutsal Azaltma (Dimensionality Reduction) .....	7
1.3. PEKİŞTİRMELİ ÖĞRENME (REINFORCEMENT LEARNING) .....	7
1.4. ÇEVİRİMİÇİ MAKİNE ÖĞRENMESİ (ON-LINE LEARNING).....	7
III. EĞİTİM ALANINDA MAKİNE ÖĞRENMESİ KULLANIMI.....	8
IV. BİLGİ İŞLEMSEL DÜŞÜNME BECERİLERİ, ÖLÇME VE DEĞERLENDİRME YÖNTEMLERİ.....	10
1.5. BİLGİ İŞLEMSEL DÜŞÜNME BECERİLERİ (COMPUTATIONAL THINKING).....	10
1.6. BLOK TABANLI PROGRAMLAMA (BLOCK BASED PROGRAMMING- BTP) ...	11

1.6.1. Eğitimde Ölçme ve Değerlendirme.....	13
1.6.1.1. Ölçme ve Değerlendirme nedir?.....	13
1.6.1.2. Ölçme İşleminde yapılan hata kaynakları .....	14
1.6.1.3. Ölçmede yapılan hata türleri .....	14
Ölçme de yapılan hata türü üçtür [19]. .....	14
1.6.2. Blok Tabanlı Programlama (BTP) ile Bilgi İşlemsel Düşüneme (BİD) Becerilerinin Değerlendirilmesi .....	14
1.6.2.1. BTP İle BİD Becerilerini Değerlendirme Kriterleri .....	17
V. UYGULAMA .....	21
5.1. KULLANILAN VERİ TABANI .....	22
5.2. KULLANILAN ARAÇLARIN AÇIKLANMASI .....	28
5.3. VERİ ANALİZİ, ÖZELLİK SEÇİMİ VE ÖZELLİK MÜHENDİSLİĞİ.....	30
5.3.1. VERİ ANALİZİ .....	30
5.3.2. ÖZELLİK SEÇİMİ .....	34
5.3.2.1. Blok Sırası (Block-Rank).....	35
5.3.2.2. Kukla Sırası (Script -Rank) .....	36
5.3.2.3. Kukla Tipi (Scrip-Type) .....	37
5.3.2.4. Blok Tipi (Block-Type) .....	38
5.3.3. ÖZELLİK MÜHENDİSLİĞİ.....	39
5.4. UYGUN MODELİN BULUNMASI, EĞİTİLMESİ VE DEĞERLENİLMESİ.....	40
5.4.1. DERİN SINIR AĞLARI (DEEP NEURAL NETWORKS) MODELİ .....	41
5.4.2. BASİT DOĞRUSAL REGRESYON (SIMPLE LINEAR REGRESSION) MODELİ.....	43
5.4.1.1. Abstraction (Soyutlama Ve Problem Ayrıştırma) alanı model eğitilmesi..	44
5.4.1.2. Paralellik (Parallelism) alanı model eğitilmesi .....	49
5.4.1.3. Mantıksal Düşünme (Logic) alanı model eğitilmesi .....	53
5.4.1.4. Eş Zamanlılık (Synchronization) alanı model eğitilmesi.....	57
5.4.1.5. Akış Kontrolü (FlowControl) alanı model eğitilmesi .....	61
5.4.1.6. Kullanıcı Etkileşimi (UserInteractivity) alanı model eğitilmesi.....	65
5.4.1.7. Veri Gösterimi (DataRepresentation) alanı model eğitilmesi.....	69
5.4.2. Uygun Modelin Değerlendirilmesi.....	74
5.5. EĞİTİLEN MODELİ ÇEVİRİMİÇİ DEĞERLENDİRME ARACINA DÖNÜŞTÜRME .....	75

5.6. UYGULMANIN TANITIMI.....	77
5.7. ÖRNEK OLAY İLE UYGULAMANIN KULLANILMASI .....	84
5.7.1. GELİŞTİRİLEN ARAÇ İLE BİD BECERİLERİNİN 7 BOYUTUNUN HESAPLAMASI.....	84
5.7.1.1. BİD BECERİLERİNİN 1 PUAN OLMA DURUMU.....	85
5.7.1.2. BİD BECERİLERİNİN 2 PUAN OLMA DURUMU.....	89
5.7.1.3. BİD BECERİLERİNİN 3 PUAN OLMA DURUMU.....	94
5.7.2. ÖRNEK OLAY İLE DEĞERLENDİRME ARACININ ÖĞRENCİ ÜZERİNDEKİ ETKİSİNİN GÖSTERİLMESİ .....	99
VI. SONUÇLAR .....	103
VII. KAYNAKÇA .....	104
VIII. EKLER .....	108
IX. ÖZGEÇMİŞ .....	127

## TABLO LİSTESİ

Tablo 1. Makine Öğrenmesi Yöntemleri ve Modellerinin Sınıflandırılması

Tablo 2. Proje Veri tabanı Alanları

Tablo 3. Meta Veri tabanının şeması

Tablo 4. Veri tabanındaki alanların her proje bazında ortalaması

Tablo 5. Veri tabanına alınan Proje verilerinin analizi

Tablo 6. BİD becerileri Seviye metriği

Tablo 7. Abstraction alanının optimizasyon değerleri

Tablo 8. Eğitim Hesaplaması

Tablo 9. Tahminler için yertutucular

Tablo 10. Doğruluk Hesabı

Tablo 11. Abstraction alanının model eğitimi çıktıları

Tablo 12. Paralellik (Parallelism) alanının optimizasyon değerleri

Tablo 13. Paralellik (Parallelism) alanının model eğitimi çıktıları

Tablo 14. Mantıksal Düşünme (Logic) alanının optimizasyon değerleri

Tablo 15. Mantıksal Düşünme (Logic) alanının model eğitimi çıktıları

Tablo 16. Eş Zamanlılık (Synchronization) alanının optimizasyon değerleri

Tablo 17. Eş Zamanlılık (Synchronization) alanının model eğitimi çıktıları

Tablo 18. Akış Kontrolü (FlowControl) alanının optimizasyon değerleri

Tablo 19. Akış Kontrolü (FlowControl) alanının model eğitimi çıktıları

Tablo 20. Kullanıcı Etkileşimi (UserInteractivity) alanının optimizasyon değerleri

Tablo 21. Kullanıcı Etkileşimi (UserInteractivity) alanının model eğitimi çıktıları

Tablo 22. Veri Gösterimi (DataRepresentation) alanının optimizasyon değerleri

Tablo 23. Veri Gösterimi (DataRepresentation) alanının model eğitimi çıktıları

Tablo 24. Basit Doğrusal Regresyon Modeli sonuçları

Tablo 25. Model Çeviri Aşaması

Tablo 26. Eğitilen modelin Javascript ortamında kullanımı

Tablo 27. Derin sinir ağları model sonuçları

Tablo 28. Eğitilen Modeli kaydedilmesi

Tablo 29. Bilgi İşlemsel Düşünme Becerileri Değerlendirme kriterleri

Tablo 30. BİD Becerileri 1 Puan Olan Örnek Uygulamanın hesaplama yapması için oluşturduğu CSV dosyası

Tablo 31. BİD Becerileri 2 Puan Olan Örnek Uygulamanın hesaplama yapması için oluşturduğu CSV dosyası

Tablo 32. BİD Becerileri 3 Puan Olan Örnek Uygulamanın hesaplama yapması için oluşturduğu CSV dosyası

Tablo 33. 20 kullanıcının yaptığı uygulamaların V1 ve V2 puanlarının karşılaştırılması

Tablo 34. Değerlendirme aracı kullanmadan öncesi ve sonrasındaki kullanıcı durumları

## ŞEKİL LİSTESİTESİ

Şekil 1. Basit Doğrusal Regresyon Modeli

Şekil 2. Google Blockly Uygulaması Ekran görüntüsü

Şekil 3. Scratch Uygulaması Ekran Görüntüsü

Şekil 4. CODE.ORG Uygulaması Ekran Görüntüsü

Şekil 5. Scratch Ekranında Örnek Uygulama

Şekil 6. MineCraft Ekranında Örnek olayı

Şekil 8. Scratch programında “Eğer İse Değilse” bloğu

Şekil 7. Mantıksal düşünme becerisinin Temel, Geliştirme ve Yeterli puan türüne göre örneklenmesi

Şekil 9. Scratch senaryo ekran alanlarının

Şekil 10. Kukla Tipi (Scrip-Type) veri gösterimi

Şekil 11. Blok tipi alanındaki veri tiplerinin kullanım sayıları

Şekil 12. Blok sırasının, meta veri tabanındaki 7 puan türüne etkisi

Şekil 13. Kukla Sırası (Script -Rank) meta veri tabanındaki 7 puan alanına etkisi

Şekil 14. Kukla Tipi (Scrip-Type) meta veri tabanındaki 7 puan alanına etkisi

Şekil 15. Blok Tipi (Block-Type) meta veri tabanındaki 7 puan alanına etkisi

Şekil 16. Çevrimiçi değerlendirme aracı ekranı

Şekil 17. BİD becerileri değerlendirme kriterleri sayfası



Şekil 18. Form sayfası ile Projeni oluştur

Şekil 19. Csv Dosyası ile Projelerini oluştur sayfası

Şekil 20. Geliştirilen Aracın Değerlendirme Ekranı

Şekil 21. BİD Becerileri 1 Puan Olan Örnek Uygulama Ekranı

Şekil 22. BİD Becerileri 1 Puan Olan Örnek Uygulamada kullanılan kod bloklarının detaylı gösterimi

Şekil 23. Örnek olay Proje1'in değerlendirme aracı ile puanlama sonucu

Şekil 24. BİD Becerileri 2 Puan Olan Örnek Uygulama Ekranı

Şekil 25. Örnek olay Proje1'in değerlendirme aracı ile puanlama sonucu

Şekil 26. BİD Becerileri 3 Puan Olan Örnek Uygulama Ekranı

Şekil 27. BİD Becerileri 3 Puan Olan Örnek Uygulamada kullanılan kod bloklarının detaylı gösterimi

Şekil 28. Örnek olay Proje2'in değerlendirme aracı ile puanlama sonucu

## KISALTMALAR

**BTP:** Blok Tabanlı Programlama      Block-Based Programming

**BİD:** Bilgi İşlemsel Düşünme      Computational Thinking

**ICS:** Akıllı Ders Sistemleri      Intelligent Course Systems

**EDM:** Eğitsel Veri Madenciliği      Educational Data Mining

**NPM:** Node Paket Yöneticisi      Node Packet Management

**DNN:** Derin Sinir Ağları      Deep Neural Networks

**CSV:** Virgülle Ayrılmış Değerler  
dosyası      Comma Separated Values

# I. GİRİŞ

Bilgisayarların ve yazılımların hayatımıza girdiği günden bu yana teknoloji gelişerek hayatımızın vazgeçilmez bir parçası olmuştur. İnsanlığın işini kolaylaştıran bilgisayar ve yazılım teknolojileri, günümüzde kendi kendine öğrenme işlemi yapmakta ve bunu yeni durumlara aktarabilmektedir.

Bilgisayarlara insan gibi düşünme ve karar verme yeteneği kazandırmak ve kendi kendilerine öğrenme işlemi yaptırmak için Makine Öğrenmesi yöntemleri ortaya çıkmıştır. Makine Öğrenmesi, son yıllarda eğlenceden sağlığa, ulaşımdan eğitime kadar birçok alanda yapılan geliştirmelerde karşımıza çıkan bilgisayar bilimi alanlarından biridir. Makine Öğrenmesi, arama motorlarında çıkan sonuçlarda ve arama motoru reklamlarında, elektronik maillerde spama düşme özelliğinde, sosyal medyada yayın akışı belirlemede ve sosyal medyada paylaşılan resimlerde insan yüzünü otomatik olarak tanıma gibi farkında olmadan günlük hayatta yaygın şekilde kullanılmaktadır.

Bu tez çalışmasının da literatür taraması iki yönlü yapılmıştır; birincisi Makine Öğrenmesi yöntemleri, ikincisi ise Blok Tabanlı Programlama dillerinde değerlendirmedir. Çalışmanın ilk yönü Makine Öğrenmesinin eğitim bilimleri alanında; model eğitirken kullanılan veri setlerine, veri alanlarına ve Makine Öğrenme yöntemlerine göre yapılan çalışmalar incelenmiştir. İkinci yönü Blok Tabanlı Programlama dillerinde öğrenci değerlendirmesi yapılırken kullanılan yöntem ve araçlar hakkında incelemeler yapılmıştır.

İncelemeler doğrultusunda Blok Tabanlı Programlama dilleri ile yapılan uygulamaların öğrenci performansını Bilgi İşlemsel Düşünme becerilerine göre

değerlendirmek için Makine Öğrenmesi yöntemlerinin kullanılması savunulmuştur. Önceden yapılmış çalışmalardan farklı olarak, doğrudan öğrenci performansı üzerinden değerlendirilme yapmasıyla hem Makine Öğrenmesi yöntemlerine hem de eğitim bilimlerine katkı sağlanması hedeflenmiştir.

Model eğitmek için, Blok Tabanlı Programlama dillerinden Scratch programındaki öğrenci projelerinin değerlendirilmesiyle oluşturulan veri setinden faydalanılmıştır. Model bu veri seti üzerinden eğitilmiştir. Eğitilen model öğrenciler tarafından yapılan projeleri Bilgi İşlemsel Düşünme becerilerine göre değerlendirmektedir. Modelin öğrenci ve öğretmenler tarafından rahatlıkla kullanılabilmesi için, model çevrimiçi değerlendirme aracına dönüştürülmüştür. Beşinci bölümde model ve çevrimiçi değerlendirme aracı detaylı olarak açıklanmıştır. Son bölümde geliştirilen çevrimiçi değerlendirme aracının örnek olay üzerinden anlatılması ve öğrenciler üzerindeki etkisi incelenmiştir.

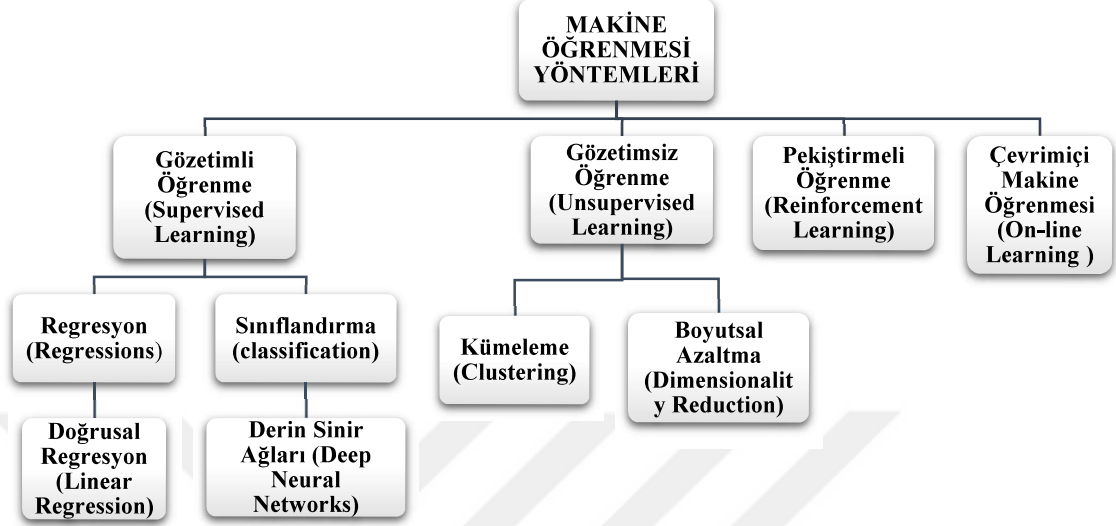
## II. MAKİNE ÖĞRENMESİ

İnsanlarda olduğu gibi yazılım algoritmaları da her geçen gün kendini yeni yetenekler ile geliştirmektedir. Hayatımıza giren internet reklamları, film kitap müzik vb. öneri sistemleri, konuşma tanıma sistemleri, otonom araç sistemleri ve birçok alan da gelişen algoritmalarda Makine Öğrenmesi karşımıza çıkmaktadır.

Makine Öğrenmesi, bilgisayarlara programlama kurallarını ve aşamalarını sunmadan, mevcut veriler üzerinden öğrenmesini ve çıkarımlar yapmasını sağlayan bilim dalıdır. [1] Makine Öğrenmesi, algoritmaların kullanılmasıyla ham veriden bilgiyi almakta ve buna göre modeli eğitmektedir. Eğitilen model yeni durumlarla karşılaştığında veri üzerinden karar verme ve tahminde bulunabilmektedir.

Makine Öğrenmesi, çalışmalarında problemin durumu ve kullanılan veri setine göre tercih edilebilecek çeşitli yöntemler mevcuttur. “Tablo 1. ” da belirtildiği gibi bu yöntemlerin en genel sınıflandırılması; Gözetimli Öğrenme (Supervised Learning), Gözetimsiz Öğrenme (Unsupervised Learning), Pekiştirmeli Öğrenme (Reinforcement Learning) Çevrimiçi Öğrenme (On-line Learning ) olmak üzere dört ana başlıkta toplanmaktadır. [2]

Tablo 1. Makine Öğrenmesi Yöntemleri ve Modellerinin Sınıflandırılması [2]



### 1.1. Gözetimli Öğrenme (Supervised Learning)

Gözetimli öğrenme model oluştururken girdi ve çıktı değişkenlerinin bir algoritma kullanarak girdiden çıktıya doğru bir eşleme sağlamasıyla model oluşturmaktadır. Gözetimli öğrenme, oluşturduğu modele göre yeni veri girildiğinde çıktının tahmin edilmesi veya sınıflandırılmasına odaklanmaktadır. Makine Öğrenmesi'nde algoritma girdilerinin göreceli özelliklerine dayanarak model eğitmeye çalışmaktadır. Modelin doğruluğu, tahmin edilen çıktılar ile gerçek çıktılarının karşılaştırılması ile sağlanmaktadır. [3]

Gözetimli öğrenme problemleri regresyon ve sınıflama olarak iki farklı türe ayrılmaktadır.

### 1.1.1. Regresyon (Regressions)

Modelin çıktısı sayısal değer ile ifade edildiğinde ve veriler gerçek değerleri tahmin ettiğinde kullanılmaktadır. Geliştirilen araçta kullanılan model Doğrusal Regresyondur.

#### 1.1.1.1. Basit Doğrusal Regresyon (Simple Linear Regression)

Basit Doğrusal Regresyon bağımlı ve bağımsız değişkenler arasındaki sebep sonuç ilişkisini doğrusal bir fonksiyonel olarak bir modelle tanımlamaktadır.

$$Y = a+bx$$

Şekil 1. Basit Doğrusal Regresyon Modeli [4]

“Şekil 1.” de Basit Doğrusal Regresyon modeli, y bağımlı değişkeni x bağımsız değişkeninin a sabit regresyon değerinin b regresyon eğiminin yansıması tahmini gösterilmiştir [4]. Y bağımlı değişken, x bağımsız değişkeni ile fonksiyonel bir ilişki içerisindedir. Geliştirilen uygulamana Basit Doğrusal Regresyon kullanılmaktadır.

### 1.1.2. Sınıflandırma (Classification)

Sınıflandırma modeli, çıktılarının kategorik değerler şeklinde olduğu durumlarda kullanılmaktadır. Eğitim veri setinin bir sınıf veya etikete sahip olduğu, oluşturulan modelin yeni girdiye değişken ve etiket verebilme durumudur. Yöntemin temel amacı, verileri basit ve analizi kolay hale getirmektir. Bu çalışmada model eğitirken test edilen modellerden biri de Derin Sinir Ağları sınıflandırmasıdır.

#### 1.1.2.1. Derin Sinir Ağları (Deep Neural Networks)

Derin sinir ağıları, sınıflandırma ve fonksiyon yaklaşımlarında kullanılan güçlü desen sınıflandırıcılarıdır. Sınıflandırma problemlerinde girdi ve çıktı ilişkilerinin öğrenmesini, gradyan kurallarına göre ağ ağırlıklarıyla yayılma algoritması ile eğitilmektedir [5]. Katman sayısı düşük olan yayılma algoritmalarında performansı iyidir, katman sayısı artıkça performans düşer.

## **1.2. Gözetimsiz Öğrenme (Unsupervised Learning)**

Gözetimsiz Öğrenme, gözetimli öğrenmeden farklı olarak cevapları yoktur, modeli eğitirken sadece verinin girdi özelliğinin olduğu, girilen verilerin çıktı özelliklerinin olmadığı yöntemdir. Model eğitilirken verideki temel yapı ve dağılım keşfederek modellenmektedir. Öğrenme esnasında denetim sağlayan bir veri parametresi bulunmamaktadır ve verilerin içerisindeki ilişkiyi sistem kendisi öğrenmektedir.[6] Bilim insanları tarafından keşfedilmeyen veri ilişkilerini keşfetmek amacıyla kullanılmaktadır.[3]

Gözetimsiz öğrenmeler problemleri kümeleme ve ilişkilendirme de iki farklı türe ayrılmaktadır.

### **1.2.1. Kümeleme**

Veri içerisindeki özellik veya davranış durumlarındaki benzerliklerine göre birlikte gruplandırma sürecidir. Kümelerin içindeki grupların netlikleri belirlenmekte ve farklı gruplar arasındaki kalıplar keşfedilmektedir. Bu kümelemeler yeni girdilere referans olabilmektedir.



### **1.2.2. Boyutsal Azaltma (Dimensionality Reduction)**

Veri setinde sınıflama yapılırken çok fazla birbiriyle ilişkili ve gereksiz özniteliklerin olduğu durumlarda, boyutsal azaltma algoritmaları kullanılmaktadır. Boyutsal azaltma işlemi ana değişkenleri elde etmek amacıyla, özniteliklerin boyutunu azarlatılmasıdır. Çok boyutlu veri setlerinde verileri görselleştirmeyi kolaylaştırmaktadır.

### **1.3.Pekiştirmeli Öğrenme (Reinforcement Learning)**

Ortamda bulunan bilgilere dayanarak deneme yanılma yoluyla sonuca ulaşmak için bir çerçevedir.[3] Modeli eğitirken gözetimli ve gözetimsiz öğrenmeden faydalanmaktadır. Geribildirimlere bağlı kalarak dinamik şekilde doğru ve yanlış performansa göre ödül ya da ceza alarak insan müdahalesi olmadan öğrenmektedir. Özetle, gerçekleşen eylemlerin sonuçlarına göre nasıl davranması gerektiğini öğrenen bir Makine Öğrenmesi yöntemidir.

### **1.4. Çevrimiçi Makine Öğrenmesi (On-line Learning )**

Büyük ve sürekli veri üreten sistemlerde modelin çevrimiçi olarak oluşturulduğu sistemlerdir.[2] Gelişen ve büyüyen sistemler olması dolayısıyla çevrimiçi olma ihtiyacı duymaktadır. Bu yöntem modeli belirli aralıklar ile güncellemekte ve veriler yenilendikçe modelleri gerçek zamanlı olarak güncelleyebilmektedir.

### III. EĞİTİM ALANINDA MAKİNE ÖĞRENMESİ KULLANIMI

Eğitimde kullanılan Makine Öğrenmesi, eğitim ortamlarından verileri keşfederek anlamlı kalıplar oluşturmayı amaçlamaktadır.[7] Makine Öğrenmesi ile oluşturulan anlamlı kalıplar, öğretim ortamlarında öğretmen ve öğrenciye yardımcı olmaktadır. Öğretim ortamlarında oluşturulurken kullanılan veri setleri; çevrimiçi eğitimler ve öğrenci ilerlemesi, öğrencilerin akademik ve sosyoekonomik durumu, okul kayıt kabul bilgi vb. özelliklerin depolanmasıyla oluşturulmaktadır. Eğitim alanında Makine Öğrenmesi temelde öğrenci performansını tahmin etme ve öğrenme tarzını belirleme üzerinden eğitime katkı sağlamaktadır.

Öğrenci Performansının İzlenmesi ve Tahmini İçin Makine Öğrenme Yaklaşımlarında özellik seçimi olarak öğrencilerin geçmiş akademik başarılarına göre yordama yapılmaktadır.

Jie Xu, K.H. Moon ve M.D. Schaar tarafından geliştirilen model, Öğrencinin lise not ortalaması, dönem içerisinde aldıkları dersleri, ders kredilerini ve elde edilen notları üzerinden tahminler yapmaktadır. Lisedeki dönem için ders notları ve lise genel not ortalamaları ile üniversite final not ortalaması arasındaki ilişki üzerinden tahminler yapılmaktadır [8]. Genel eğilim, dönem için ders notları daha yüksek olan öğrencilerin, lisans programında daha yüksek final not ortalaması aldıkları tahmin edilmektedir.

H. Agrawal ve H. Mavani tarafından geliştirilen model, Öğrencilerin akademik performansını tahmin etmek için geçmiş performanslarına bağlı kalarak modelleme yapılmaktadır. Öğrencilerin derse katılım ve ev ödevi not verilerinden faydalanarak öğrencinin performansı tahmin edilmektedir [9].

E. Er'in [10] riskli Öğrencilerin Makine Öğrenim Tekniklerini Kullanarak Belirlenmesi üzerine yaptığı çalışmada, yaş- cinsiyet vb zamanla değişmeyen veri özelliklerinin model eğitimi üzerine etkisi ölçülmüştür. Zamanla değişmeyen verilerin model dışında tutulması ve zamanla değişen verilerin; öğrenci ders devam bilgisi, ödev notu, derse devam notuna kıyasla önemli bir etkisinin olmadığını kanıtlanmıştır [10].

M.Ciolacu, A.F. Tehrani, R. Beer ve H. Popp Makine Öğrenmesi ile öğrenci performansını desteklemek için yaptıkları çalışmada; tam çevrimiçi eğitim ve harmanlanmış eğitimi kıyaslayarak Zaman içerisinde tıklama sayısı ve derse giriş sayısı Tam zamanlı sanal kursta eğitime ne zaman başlandığının önemi olmadığını, harmanlanmış eğitimde derse başlama ve tıklanma sürelerinin etkilediği belirtilmiştir [11].

D. Kelly ve B. Tangney tarafından ilk yardım için geliştirilen Makine Öğrenmesi ile öğrenme tarzlarının tanıma üzerine araştırma yapılmıştır. Araştırma online ilk yardım eğitimi sırasında öğrencinin hesabı üzerinden yaptığı davranışlara göre öğrenme stilini belirlemekte ve buna göre yönlendirme yapmaktadır. Özellik seçimi olarak, Kişinin eğitim sürecinde öğrenme aktiviteleri (genel bakış, hedefler, özet, test, bitiş), kaynak türü (metin, ses, resim, metin ses, metin resmi, metin animasyonu)sistemdeki bu tercihlerinden faydalanarak, öğrenme tarzı belirlenmektedir [12].

[9], [8] ve [10] da bulunan çalışmalar detaylı incelendiğinde, Makine Öğrenmesi yöntemleriyle öğrenme performansı tahmin etmek için birçok model önerilmiştir. Bu modellerde kullanılan veri setleri, öğrenenin bir önceki seviye ya da derste aldığı not ortalaması, ev ödevi notu vb. daha önceden değerlendirilmiş becerilerine göre öğrencinin performansını tahmin ve çıkarım yapmaktadır.

## IV. BİLGİ İŞLEMSEL DÜŞÜNME BECERİLERİ, ÖLÇME VE DEĞERLENDİRME YÖNTEMLERİ

### 1.5.BİLGİ İŞLEMSEL DÜŞÜNME BECERİLERİ (Computational Thinking)

Teknoloji hayatın her alanında gelişerek insanlığa güçlü çözümler getirmektedir. Bu gelişmelerle beraber dünya genelinde teknoloji eğitimleri vazgeçilmez bir noktaya gelmiştir. Günümüzde bilgisayar bilimi eğitiminde Bilgi İşlemsel Düşünme becerileri çok popüler hale gelmiştir. Tüm dünyada ve Türkiye de Bilgi İşlemsel Düşünme becerileri erken yaşlardan itibaren bilgisayar programlaması yoluyla geliştirmek için kurslar ve okullarda bu eğitim yaygın olarak verilmektedir. J. Wing[13] Bilgi İşlemsel Düşünme (BİD) becerisini mühendislik ve matematiksel düşünmenin tamamlayıcısı, problem çözme sistemlerinin geliştiricisi, bilgisayar yazılımının temel ilkeleri ve kavramlarını oluşturan; problemi soyutlama ve çözüme, tekrar eden kalıpları belirleme ve genelleme gibi zihinsel aktiviteler olduğunu savunmaktadır. BİD becerilerinin bir diğer tanımı ise bilgisayar bilimlerinden faydalanarak, soyut düşünmeyi temele alan, problem çözme becerileri ve algoritma yeteneğini geliştiren bilişsel öğrenme alanıdır [14].

BİD becerilerinin ana özellikleri şunları içermektedir [15]:

- Veri analizi ve mantıksal düzenleme
- Veri modelleme, veri soyutlama ve simülasyonu
- Problemlerin formüle edilmesi
- Çözümü tanımlama, test etme ve uygulama

- Algoritmik düşünme yoluyla Otomatik çözüm üretme
- Çözümü genelleştirmek ve benzer problemlere uygulama

BİD becerileri, öğrencinin başarısını geliştirmekte ve okulda öğrenilen bilgileri gerçek hayata farklı birçok disipline aktarabildikleri düşünülmektedir [14]. Öğrenci gerçek hayattaki sorunları BİD becerilerini kullanarak çözümlerini bulacak ve çözümleri otomatik hale getirebilecektir. Bu disiplin sayesinde öğrencinin problem çözme ve eleştirel düşünme becerileri gelişecek ve zihinsel aktiviteleri daha yetkin duruma gelecektir.

BİD becerilerinin geliştirilmesinde en yaygın olarak kullanılan araçlar, blok tabanlı programlama dilleridir.

Bu yetkinlik öğrenilirken öğretmen ve öğrenciye yardımcı olacak çok fazla seçenek varken, değerlendirme aşamasındaki araçların kısıtlılığı ve eksikliği bulunmaktadır.[16]

### **1.6.Blok Tabanlı Programlama (Block Based Programming- BTP)**

BTP programlama becerilerinde grafiksel yaklaşımın geleneksel metin-bazlı programlama özellikleriyle karışımını harmanlamaktadır. BTP, kullanıcıya komutları bir bulmaca parçası şeklinde sunmakta ve sürükle bırak talimatlarıyla çalışmaktadır. Ekranında görevin hakkında ipuçları ve açıklamalar bulunur. Uyumsuz talimatlar hakkında uyarı mesajları vermektedir. Böylece sözdizimi hatlarını önler. BTP popülaritesinin öncülüğünde, çok çeşitli BTP araçlarının giderek artan sayıda araç ve kütüphane bulunmaktadır. “Şekil 2.”, “Şekil 3.” ve “Şekil 4.” de ekran görüntüsü verilen; Scratch, Google Blockly ve Code.org en popüler BTP araçlardandır. BTP

araçlarından en çok kullanılan Scratch aracıdır. Scratch [15], 8 ile 16 yaş arası çocukların bilgisayar programlamasına giriş yapmaları için geliştirilmiş blok tabanlı bir programlama dilidir.



Şekil 2. Google Blockly Uygulaması Ekran görüntüsü



Şekil 3. Scratch Uygulaması Ekran Görüntüsü



#### Şekil 4. CODE.ORG Uygulaması Ekran Görüntüsü

Scratch istatistikleri web sitesine göre 39 milyondan fazla kayıtlı kullanıcı olduğu ve 196 milyondan fazla projenin Scratch web kayıtlarında açık kaynaklı olarak depolandığı bildirilmektedir [18].

Sonraki bölümde Blok Tabanlı Programlama dillerinde ölçme ve değerlendirmeye giriş yapılmıştır, öncesinde eğitim bilimlerinde ölçme ve değerlendirme incelenmiştir.

#### **1.6.1. Eğitimde Ölçme ve Değerlendirme**

Eğitim alanında yapılan çalışmalarda, istatistiksel ölçme araçları ve uygulamaları geliştirmeden evvel eğitim bilimleri disiplinine göre ölçme değerlendirme kriterlerine öncelik verilmelidir.

Eğitim ve öğretimde alanında, ölçme ve değerlendirme de eğitimin bir parçasıdır. Eğitimin tam olabilmesi için öğrenene ölçme, değerlendirme ve geribildirim verilmelidir. Öğrenenin değerlendirilmesi ve geribildirim esnasında ölçme ve değerlendirme; doğru, güvenilir ve hatasız bir şekilde yapılması gerekmektedir.

#### **1.6.1.1. Ölçme ve Değerlendirme nedir?**

Ölçme; gözlemlenen değişkenlerin, gözlem sonucu sembol veya sayılar ile ifade edilmesidir. Değerlendirme; ölçme sonuçlarının daha önceden belirlenen ölçüt(kriterler) ile karşılaştırılarak ölçülen hakkında karar ve sonuç verme sürecidir [27].

### **1.6.1.2.Ölçme İşleminde yapılan hata kaynakları**

Ölçenden kaynaklanan hatalar: Ölçümü yapan kişiden kaynaklanan bu hatalar, ölçenin yetersiz fiziksel, pisikolojik ve eğitsel özellikleridir. Ölçüm esnasında örneğin fiziksel durumu yorgunluk, uykusuzluk gibi sebepler hata sebebidir.

Ölçülenden kaynaklanan hatalar: Ölçülen kaynağın, uygun özellik ve araçla ölçülmesi gerekmektedir. Örnek olarak; elektrik telleri mevsime göre gevşeyip gerginleşmektedir. Eğer yazın ölçüm yapıyorsak tellerin gevşediği özelliğini dikkate alınarak ölçüm yapılması gerekmektedir.

### **1.6.1.3.Ölçmede yapılan hata türleri**

Ölçme de yapılan hata türü üçtür [19].

1.Sabit Hata: Bir Ölçümden diğer ölçüme değışmeyen hata türüdür. Ölçen araçtan kaynaklanır.

2. Sistematik Hata: Bir ölçmeden diğer ölçmeye tekrar eden hatalardır. Örnek olarak; ölçüm yapan metrenin 100 cm olması gerekirken 99 cm olmasıdır. Her ölçümde sistematik olarak değerler değışkenlik gösterir.

3. Tesadüfi Hata: Ne zaman ve nasıl ortaya çıkacağı belli olmayan, önlenemeyen hata türüdür.

## **1.6.2. Blok Tabanlı Programlama (BTP) ile Bilgi İşlemsel Düşüneme (BİD)**

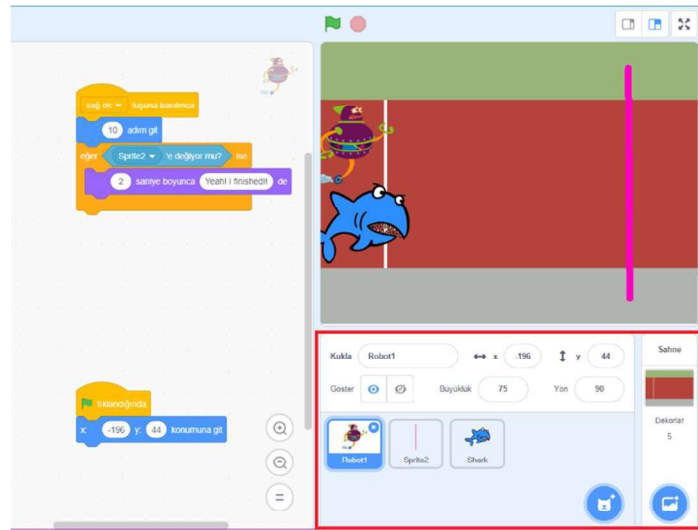
### **Becerilerinin Değerlendirilmesi**

BİD becerilerinin öğretilmesi ve değerlendirilmesinde en yaygın olarak kullanılan araçlar, blok tabanlı programlama dilleridir.



Blok Tabanlı Programlama (BTP) dillerinde, Bilgi İşlemsel Düşünme (BİD) becerilerin geliştirmek için birçok araç mevcutken, bu becerilerin değerlendirilmesi aşamasına gelindiğinde sınırlı ve az sayıda seçenek dolayısıyla öğrenci ve öğretmenler zorlanmaktadır. Mevcut BTP programlarındaki değerlendirme yöntemleri incelendiğinde, ölçme hataları dolayısıyla sonuçların etkilendiği görülmektedir.

Hatanın ilk sebebi ölçenden kaynaklanma durumudur. BTP grafik tabanlı oluşu öğretmenin görsele göre taraflı değerlendirme yapmasına sebep olmaktadır. Buna ek olarak öğretmenin öğrenciye olan tutumunun olumlu veya olumsuz olmasından etkilenecek değerlendirilmeye hata karıştırılacaktır. “Şekil 5.” de gösterilen ekranda 3 kukla, 1 dekor bulunmakta. Kod kısmı değerlendirildiğinde 7 adet blok mevcuttur. Değerlendirmeyi yapan öğretmen ekranda en az 2 kukla olmasından etkilenecek kullanılan blokları değerlendirmeyi ikinci plana atabilir. Diğer taraftan değerlendirme yapan öğretmenin, öğrenciye karşı olan tutumu da bu değerlendirme olumsuz yönde etkileyebilmektedir.



Şekil 5. Scratch Ekranında Örnek Uygulama

Hatanın ikinci sebebi ölçülenden kaynaklanan hatadır, grafik tabanlı olan BTP uygulamaları kullanıcıyı hedefe yönlendirmeler ile ulaştırabilir. Kişinin BİD becerilerin dikkate almadan tek boyutta hedefe ulaşmasının başarı olarak kabul edilmesi. Bu durum ölçme aracından kaynaklanan hatalara girmektedir. Çözümü ürünün değil sürecin değerlendirilmesi ile çözülebilir. “Şekil 6” da gösterilen alıştırmada kullanıcı ekranın sol tarafında bulunan uygulama ekranına göre hedefi gerçekleştirilmiştir. Sağ tarafta bulunan kod ekranı kontrol edildiğinde ilk bakışta kullanıcı birçok blok kullanabildiği düşünülebilir. Fakat BİD becerileri dikkate alındığında çok fazla kod kullanılmış olması, ürünün ve sürecin yeterli olduğu anlamına gelmez. “Şekil 4.” incelenirse, ekranın sol tarafında görsel olarak hedefin tamamlandığı görülmekte. Sağ taraftaki kod blokları incelendiğinde ise kullanıcının farklı blok türleri ve nicel olarak “Şekil 6.” ya göre daha az blok kullanıldığı görülmektedir. BİD becerilerine göre “Şekil 4.”, “Şekil 5.” ‘e kıyaslandığında, her iki yöntem ile hedefe ulaşıldığı görülmektedir.

Popüler BİD becerilerini değerlendirme uygulamalarında kullanıcının süreç esnasında kullandığı farklı tür bloklar ve çözüme en kısıyol ile ulaşmış olması önemli değildir, önemli olan hedefe ulaşmaktır. Bu yöntem de BİD becerilerini tam olarak ölçmemekte ve ölçmeye ölçülenden kaynaklanan hatayı karıştırmaktadır.



Şekil 6. MineCraft Örnek

Son olarak, Değerlendirmeyi öğretmen veya bir canlının yapması tesadüfi hataların ortaya çıkma olasılığını artırmaktadır.

#### 1.6.2.1.BTP İle BİD Becerilerini Değerlendirme Kriterleri

BİD becerilerini öğrenen ve öğretmeye yardımcı birçok kaynak olmasına rağmen değerlendirmeye yönelik kaynaklar çok kısıtlı kalmaktadır. En çok tercih edilen BTP dili olan Scratch geliştiricileri tarafından bu eksikliği gidermek adına, Dr.Scratch yazılım kalite değerlendirme aracı oluşturulmuştur.

Dr. Scratch, Scratch projelerini BİD becerisini analiz etmek, geliştirmek ve öğrenciyi teşvik etmek amacıyla, Hairball eklentilerini kullanarak açık kaynaklı yazılım kalite değerlendirme aracıdır.[20]. Dr.Scratch aracında, Scratch uygulamasında gerçekleştirilen projeler sisteme yüklenir ve BİD becerilerinin 7 boyutu hakkında değerlendirme sonuçları alınmaktadır. Bu araç yazılım karmaşıklığını ölçmek için BİD yetkinliğini; Soyutlama Ve Problem Ayırıştırma, Mantıksal Düşünme, Eş Zamanlılık,

Paralellik, Akış Kontrolü, Kullanıcı Etkileşimi ve Veri Gösterimi; olmak üzere 7 farklı boyut da değerlendirmektedir. Değerlendirme gerçekleştirilmek için kullanılan metrikler “Tablo 6. ” de belirtilmiştir. Her boyut için 0 ila 3 puan arasında toplamda 21 puan üzerinden değerlendirmeye dayanmaktadır [21].

Tablo 6. BİD becerileri Seviye metriği [20]

BİD BOYUTLARI	YETERLİLİK SEVİYESİ			
	BOŞ (0)	TEMEL (1 PUAN)	GELİŞTİRME (2 PUAN)	YETERLİ (3 PUAN)
Soyutlama ve Problem Ayırıştırma	-	Birden fazla senaryo ve sprite (kukla/sahne)	Özel bloklar / Blok tanımlama	Klonlama blokları
Paralellik	-	İki adet Yeşil Bayrak (Tıklanınca komutu)	İki adet haberi aldığı zaman komutu, iki adet senaryo tıklanınca komutu (aynı senaryo için)	İki adet senaryo mesaj alındığında, kopya oluştur, iki adet senaryo %değer, %diğer değerden büyük olduğunda, iki senaryo dekor değişince
Eş Zamanlılık	-	Bekle komutu	Haberi sal, Mesajı alındığında yayımla, hepsini durdur, programı durdur, program kuklalarını durdur.	Kadar bekle, Dekor değiştiği zaman, haberi sal ve bekle
Mantıksal Düşünme	-	İf (kullanımı)	İf else (kullanımı)	Matematiksel operatörler (kullanımı)
Akış Kontrolü	-	Dizi blokları	Kadar tekrarla, sürekli tekrarla	Olan kadar tekrarla
Kullanıcı Etkileşimi	-	Yeşil Bayrak (Tıklanınca komutu)	Tuşa basılınca, sprite (kukla/sahne) basılınca, sor ve bekle, fare blokları	%değer, %diğer değerden büyük olduğunda, video, ses
Veri Gösterimi	-	Sprite (kukla/sahne) Bloklarının özellikleri	Değişkenlerin üzerindeki operatörler	Liste operatörleri

“Şekil 7. ” de BİD becerilerinden Mantıksal düşünme boyutundan 3 puan türüne örnek blok değerlendirilmesi verilmiştir. “Tablo 6. ” de belirtildiği üzere mantıksal düşünme seviye metriğinde if kullanımı temel seviye, if else kullanımı geliştirme seviyesi, matematiksel operatörler kullanımı yeterli seviye metrikleridir. “Şekil 7.” de Birinci örnekte sadece if komutu kullanılmıştır, eğer fareye değdiği ise hepsini durdur komutu temel seviyesi, İkinci örnekte fareye değdi ise hepsini durdur değilse 10 adım git geliştirme seviyesidir. Üçüncü örnekte fareye değdi ise komutunun değil komutuyla kullanılması yeterli seviye olduğunu göstermektedir.



Şekil 7. Mantıksal düşünme becerisinin Temel, Geliştirme ve Yeterli puan türüne göre örneklenmesi

Değerlendirme sonuçlarına göre 7 puana kadar olan projeler Temel Seviye, 8 ila 14 puan arasındaki projeler Geliştirme Seviyesinde ve 15'ten fazla puana sahip projelerin ise Yeterli Seviye olarak kabul edilir [21] .

Değerlendirme yapmak için belirli bir blok ya da blok grubunun kullanılmasının yanı sıra; Sprite adlandırma veya tekrarlanan kod gibi değerlendirmelerde, blokların doğru kullanılıp kullanılmadığını tespit etmeyi amaçlayan Dr. Scratch aracına dâhil edilmiştir [22]. Bunun yanı sıra bu araçta çalışmanın en büyük kısıtlılığı, hata ayıklama veya remiksleme gibi bazı temel BID becerilerinin, projenin kodunu analiz etmede doğrudan ilişkili olmadığı gözlemlenmiştir..

Literatürde yapılan diğer arařtırmalar incelendiğinde Yazılım Karmařıklığı alanında dünya çapında olarak kabul edilen iki metrik McCabe'nin Döngüsel Karmařıklığı (Cyclomatic Complexity -CC) ve Halstead'in metrikleri mevcuttur. Bu metriklere göre Dr. Scratch BID becerileri puanı ile analiz edilen Scratch projelerinin pozitif ve anlamlı bir korelasyona sahip olduđu görölmektedir. Dr. Scratch korelasyona düşünme puanı, hem Döngüsel Karmařıklığı hem de Kelime Bilgisi ile güçlü bir korelasyona sahip, Uzunluk ile pozitif orta derecede korelasyona sahiptir[21].

## V. UYGULAMA

Bu bölümde Makine Öğrenmesini kullanarak gerçekleştirilen çevrimiçi değerlendirme aracının hazırlık süreci, oluşturulması ve tanıtılması detaylı olarak anlatılmıştır.

İlk olarak kullanılan veri tabanı detaylı olarak incelenmiştir. Uygulama sürecinde kullanılacak araçlar detaylı şekilde incelenmiş ve seçilmiştir. Uygun araçlar ile verilerin analizi, özellik seçimi ve özellik mühendisliği işlemleri yapılarak modelin eğitilmesine hazırlık sağlanmıştır. Belirlenen özellik seçimlerine göre model denemeleri yapılmış, uygun model bulunmuş, model veri seti ve ihtiyaç doğrultusunda eğitilmiştir. Modelin sonuçları değerlendirilmiştir.

Model öğretmen ve öğrencilerin performans değerlendirmesinde rahatlıkla kullanabilmesi için bir çevrimiçi değerlendirme aracına dönüştürülmüştür. Gerçekleştirilen bu çevrimiçi değerlendirme aracı örnek öğrenci projesi değerlendirmesi yapılarak tanıtılmıştır.

## 5.1. KULLANILAN VERİ TABANI

E. Aivaloglou, F. Hermans, J. Moreno-Leon, ve G. Robles [23] tarafından Scratch blok tabanlı programlama dilinin veritabanından çıkarılmış 100.000 den fazla farklı kullanıcının 250.000 Scratch proje veriseti hazırlanmıştır. Bu veri setini kullanmak için alınan gerekli izin Ek-A da belirtilmiştir.

Scratch projelerinin içinde bulunan tüm bu veriler detaylı olarak Scratch web arayüzünden JSON dosya kazıma aracını kullanarak Scratch wiki [24] sayfasındaki blok şekil ve kategori bilgisi rehberliğinde görsel bloklar metine çevrilmiştir. “Şekil 8.” Dr. Scratch arayüzünden kazınan kontrol bloklarından “Eğer İse Değilse” örnek blok görüntüsü bulunmaktadır. Kazınan bu kod diğer programlama dillerindeki “If () Then” koşul komutu anlamına gelmektedir ve Scratch verileri işlenirken “doIf” biçiminde veri blok tipi alanına işlenmektedir...



Şekil 8. Scratch programında “Eğer İse Değilse” bloğu

İşlenen tüm bu bilgiler proje verileri ve meta veriler adında ilişkisel iki veri tabanına aktarılmıştır. Proje veri tabanında bir projenin içerisindeki her senaryo, her blok tipi, her kukla/ sahne isim ve değişkenlerin isimleri detaylı olarak işlenmiştir. “Tablo 2.” de proje veri tabanının bilgileri bulunmaktadır.



Tablo 2. Proje Veri tabanı Alanları

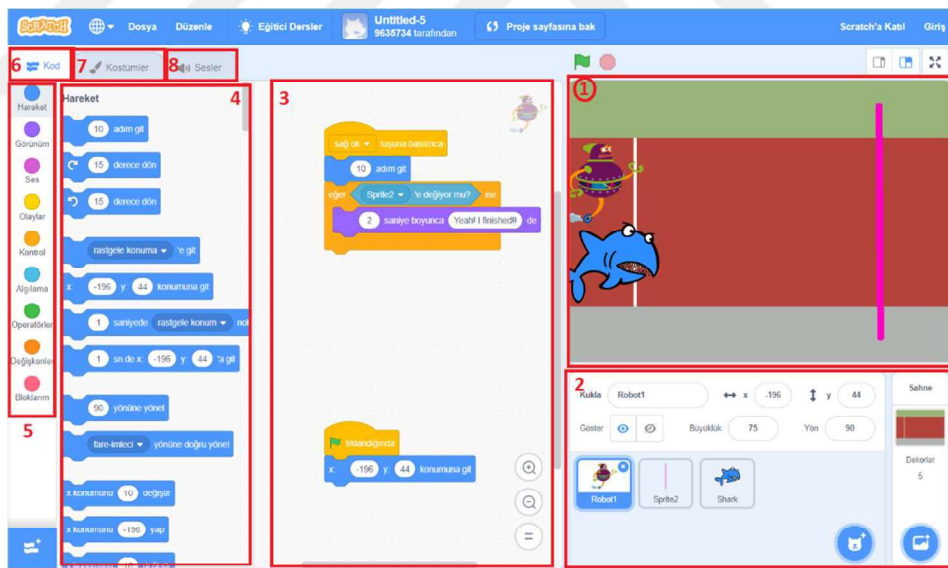
p_ID	Proje kimlik numarası
script -rank(senaryo sırası)	Proje içerisindeki kod grubu işlem sırası
sprite – type( kukla tipi)	Hareket ettirilen işlem yapılan alan türü; Sahne, Kukla, özel taş
sprite-name (kukla ismi)	Kukla tipinde bulunan kukla türünün kullanıcı tarafından hangi ad ile kullanıldığı
coordinates (koordinatlar)	Kuklanın ya da dekorun sahne üzerinde bulunduğu koordinatlar
total-blocks (her bir sprite kukla/sahne içerisinde kaç blok var)	Toplam kaç satır
block-rank (blok sırası)	Proje içerisinde blok kaçınıcı blok sırasında
block-type (blok tipi)	Scratch wiki de belirtilen blok türleri rehberliğinde, blok tipleri belirlenmiş
param1 .....10 (değişkenler)	Toplam 10 adet değişkenlerin isimleri alanı mevcuttur

“Tablo 2.” de bulunan veri tabanı alanları, “Şekil 9.” da belirtilen Scratch görsel programlama ekranından veri tabanına eklenmektedir. Şekilde bulunan 1 numaralı alan sahnededir, burada istenilen sayıda kukla kullanılabilir. 2 numaralı alan Kukla kod Sırası (script -rank), kukla tipi (sprite – type), kukla ismin (sprite-name) ve kuklanın bulunduğu koordinat (coordinates) alanlarının olduğu yerdir. Kullanıcının 2 numaralı

alandan görsel eklediği kuklanın veri tabanına kaydedilme esnasında kuklanın sırası, tipi, ismi ve kukla mı dekor mu tipi veri tabanında detaylı olarak yazılmaktadır.

3 numaralı alan blok tipi (block-type), blok sırası (block-rank) ve toplam blok sayısı (total-blocks) bilgilerinin bulunduğu yerdir. Proje veri tabanının ana bölümünü oluşturur. Proje veri tabanı dosyasındaki kullanılan kukla ya da dekorun senaryosunun bulunduğu kullanıcının kodlarını yazdığı ve veri tabanına kullanılan kodlar olarak işlenen alandır.

4 numaralı alanda blok türleri e 5 numaralı alanda blok kategorileri bulunmaktadır. 6 numaralı alan kod bölümünü açmayı, 7 numaralı alan kuklanın görsel hareketlerini ya da varsa dekorun görsel hareketlerinin ayarlandığı ekranı açan kısımdır. 8 numaralı alan kukla ya da dekora ses eklemek için kullanılan alandır.



Şekil 9. Scratch senaryo ekran alanları

İkinci veri tabanı olan meta veri tabanının bilgileri “Tablo 3.” de detaylı gösterilmiştir. Proje veri tabanında bulunan blok tipi, blok sırası, kukla tipi ve Kukla Kod Sırası özelliklerine göre Dr.Scratch tarafından BID becerilerinin her 7 boyutu için; Soyutlama Ve Problem Ayırıştırma, Mantıksal Düşünme, Eş Zamanlılık, Paralellik,

Akış Kontrolü, Kullanıcı Etkileşimi ve Veri Gösterimi 0 ila 3 arasında toplamda 21 puan olmak üzere master alanında bulunmaktadır. Dr. Scratch, bu puanlamayı yaparken “Tablo 1.” de belirtilen BİD kriterlerine göre her projeyi değerlendirmektedir.

Tablo 3. Meta Veri tabanının şeması

p_id (Proje İd)	Proje kimlik numarası
project-name (proje adı )	Proje adı
username (kullanıcı adı )	Proje sahibi hesabın kullanıcı adı
total-views (toplam- görüntüleme)	Proje görüntüleme sayısı
total-remixes (remix - sayısı)	Projenin kaç kez remiks edildiği, farklı bir projede kullanıldığı
total-favorites (toplam-favori-sayisi)	Projenin kullanıcılar tarafından favorilere eklenme sayısı
total-loves (toplam-begeni-sayisi)	Projenin kullanıcılar tarafından beğenilme sayısı
is-remix (remiks-mi)	Proje başka bir projenin remiksini
Remixed-from (dan-remiks)	Proje hangi projeden remiks
Abstraction (Soyutlama Ve Problem Ayrıştırma)	BİD becerilerine göre Soyutlama Ve Problem Ayrıştırma puanı
Parallelism (Paralellik)	BİD becerilerine göre Paralellik puanı
Logic (Mantıksal Düşünme)	BİD becerilerine göre Mantıksal Düşünme puanı
Synchronization (Eş Zamanlılık)	BİD becerilerine göre Eş Zamanlılık puanı
FlowControl	BİD becerilerine göre Akış Kontrolü puanı

(Akış Kontrolü)	
UserInteractivity (Kullanıcı Etkileşimi)	BİD becerilerine göre, Kullanıcı Etkileşimi puanı
DataRepresentation (Veri Gösterimi)	BİD becerilerine göre Veri Gösterimi puanı
Mastery	BİD becerilerine göre 7 boyutun toplam puanı
Clones (klon-mu)	Proje kopya mı?
CustomBlocks (özel-tas-blok)	Proje içerisinde özel taşlar kullanılmış mı?
InstancesSprites	Projede klon oluşturacak özellikler kullanılmış mı?
URL	Projenin orjinal belgesinin scratch web alanındaki adresi/urlsi

Sunulan veri tabanının her proje bazına ortalama değerleri “Tablo 4.” de sunulmaktadır. Projelerin çoğu bir kez dahi beğenilmemiş ve remikslenmemiştir. 250.000 projeden sadece 10.000 proje remiks yapılmıştır [23]. Sunulan istatistiksel verilere göre özel blok tanımlaması, toplam beğeni sayısı, toplam remiks sayısı, favorilere ekleme sayısı ayırt edici veri özelliği olarak kullanılamamaktadır. Tez çalışmasında kullanılan veri setindeki ayırt edici olarak; kukla tipi, kukla sırası, bloklar ve Dr.Scratch puanları üzerine odaklanılmaktadır.

Tablo 4. Veri tabanındaki alanların her proje bazında ortalaması [23]

Alan adı	Proje Başına Ortalama Kullanım Oranları
Sprite (kukla)	5.3
Script (senaryo)	16.2
Blocks (blok)	144.3
Custom block (Özel taş)	0.6
Total view (toplam- görüntüleme)	5.8
Total-remiks (remix - sayısı)	0.1
Totoal-favorite (toplam-favori-sayisi)	0.5
Dr.Scratch Mastery (toplam puan)	8.9

## 5.2.KULLANILAN ARAÇLARIN AÇIKLANMASI

Uygulama geliştirme iki aşamadan oluşmaktadır; ilk aşamada modelin eğitilmekte, ikinci aşamada eğitilen modelin web ortamında kullanılmak üzere dönüştürülmesidir.

Modelin eğitilmesi aşamasında, Kullanılan veri tabanı 5 gb boyutunda alan barındırmaktadır. Veri tabanındaki verileri hızlı ve esnek bir şekilde işlemek, temizlemek ve modelin eğitmek için Colaboratory geliştirme ortamı tercih edilmiştir. Colaboratory [24] , Google tarafından derin öğrenme ve Makine Öğrenmesi projelerinde kullanılmak üzere bulut depolama ortamında, Grafik İşleme Birimi (GPU) desteği ile Jupyter not defterleri üzerinde çalışmaktadır. Jupyter not defterleri Python dilinin görselleştirme kütüphanelerinden Matplotlib Kütüphanesi, veri analizi için Pandas kütüphanesi ve Makine Öğrenmesi için Tensorflow kütüphanelerini GPU desteği ile rahatlıkla çalıştırmaktadır. Veri analizi ve özellik seçimi yapılırken Jupyter defterlerinde rahatlıkla çalışan bu kütüphaneler tercih edilmiştir.

Model oluşturulurken not defterleri üzerinde Python dili kullanılarak eğitilen modeller, Tensorflow kütüphanesi yardımı ile kaydedilmiştir. Kaydedilen bu model web ortamında kullanılmak üzere Tensorflow.js kütüphanesi yardımı ile Javascript dilinde kullanılabilir hale getirilmiştir.

Tensorflow Makine Öğrenmesi ve derin öğrenme modellerinin oluşturulması ve geliştirilmesi için oluşturulmuş açık kaynaklı yazılım kütüphanesidir [25]. Tensorflow masaüstü, bulut depolama, mobil ve İot cihazlar, CPU'lar ve Javascript gibi her ortamda Makine Öğrenmesinin çalışmasını sağlamaktadır. Tensorflow, büyük ölçekli Makine Öğrenmesi ve derin öğrenme hesaplamalarını performansı düşürmeden çözümlenmektedir. Tensorflow çok sayıda soyutlama sunması sayesinde kolay model

oluřturma imkânı vermektedir. Karmařık modeller oluřturmak için Keras Apisini kullanmaktadır [25].

Tensorflow.js, javascript ve üst düzey apileri kullanarak tarayıcıda ve Node.js de Makine Öğrenmesi modellerini tanımlama, eğitme ve çalıştırma amacıyla oluřturulmuř, Javascript kütüphanesidir [26]. Bu kütüphane, Makine Öğrenmesi modelini üç farklı akıřla oluřturmaktadır; önceden eğitilmiş modeli içeri alınıp kullanılması, içe alınan modelin tekrar eğitilmesi ve doğrudan tarayıcı üzerinde modelin oluřturulması. Bu üç farklı akıř durumunda da tarayıcı üzerinden Makine Öğrenmesi kullanımı alanı çek geliřmiştir.

İkinci aşamada, Colaboratory ortamında oluřturulan modelin, web ortamında çalışması amacıyla bazı araçlar kullanılmıştır. Öncelikle Node.j, Javascript çalışma zamanı ortamı kullanılmıştır. Node.js, sunucu ve ağ bağlantılı uygulamaları geliřtirmek için oluřturulmuř çalışma ortamlarıdır. Node.js sayesinde Javascript, betik dillerin özelliklerine sahip olmaktadır.

Javascript kullanımında Node.js için kütüphaneler gerekmektedir, bu kütüphaneleri sahip olmak ve yönetmek için Node Paket Yöneticisi (Node Packet Management -Npm) kullanılmıştır. Bu paket yöneticisi sayesinde Node.js Javascript çalışma ortamında istenilen tüm Javascript kütüphanelerine erişilmiştir. NPM paket yöneticisine ek olarak, tüm Html ve Javascript sayfalarını yönetmek için Yarn paket yöneticisi kullanılmıştır. Colaboratory de Tensorflow.js kütüphanesi ile tarayıcı ortamında kullanılmak için kaydedilen model sunucu içerisine alınmıştır. Sunucuya alınan model tarayıcıda kullanılmak üzere web sayfası içerisindeki. js uzantısı kaydedilerek, modelin tarayıcıda çalışabilecek hale gelmesi sağlanmıştır.

### 5.3.VERİ ANALİZİ, ÖZELLİK SEÇİMİ ve ÖZELLİK MÜHENDİSLİĞİ

#### 5.3.1. VERİ ANALİZİ

Öncelikle, veri tabanında bulunan alanları analiz etmek ve özellik seçi yapmak için Colaboratory programlama ortamında proje ve meta veri tabanı yüklenmiştir. Özellik seçimi ve veri alanlarının analizinin hızlı ve kolay olması için Jupyter defterinde proje veri tabanının ilk 10 milyon satırı ve ilişkili meta verileri kullanılmıştır. Bu veri alanlarının, toplam boş değer sayısı, o alandaki toplam kaç farklı veri tipi olduğu ve alan başına düşen veri tipi bulunmuştur. Bu sayılar “Tablo 5.” de detaylı olarak gösterilmiştir.

Tablo 5. Veri tabanına alınan Proje verilerinin analizi

Alan Adı	Analiz için seçilen ilk 10.000.000 veri üzerinde			
	Veri Türü	Boş değer sayısı	Veri tipi sayısı	Alan başına düşen veri tipi sayısı (alan sayısı/ veri tipi sayısı)
Kukla Sırası (Script -Rank)	Sayısal	0	1429	6997
Kukla Tipi (Sprite – Type)	Metin	0	3	3333333
Kukla İsmi (Sprite-Name)	Metin	997	7061	1416
Koordinatlar (Coordinates)	Metin	0	42970	232
Toplam Blok Sayısı (Total-Blocks )	Sayısal	0	255	39215
Blok Sırası (Block-Rank)	Sayısal	0	987	10131
Blok Tipi (Block-Type)	Metin	0	139	71942



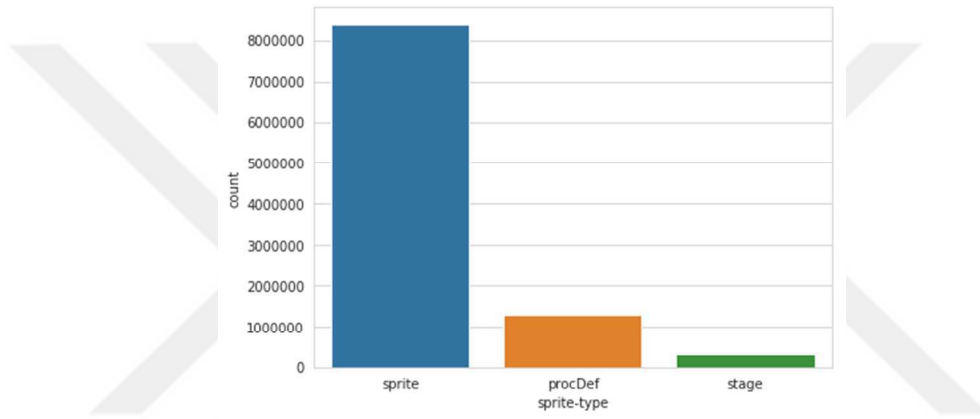
Değişken1 (param1)	Metin	3475320	29365	340
Değişken2 (param2)	Metin	8578440	2872	3481
Değişken3 (param3)	Metin	9908600	1154	8665
Değişken4 (param4)	Metin	9992856	98	102040
Değişken5 (param5)	Metin	9996676	37	270270
Değişken6 (param6)	Sayısal	9997896	19	526315
Değişken7 (param7)	Sayısal	9999169	8	1249999
Değişken8 (param8)	Sayısal	9999897	1	9999999
Değişken9 (param9)	Sayısal	9999897	1	9999999
Değişken10 (Param10)	Sayısal	9999897	2	9999999

Analiz sonucunda “Tablo 5.” de belirtilen Değişken1 alanının boş değer sayısının yüksek, kullanılan veri tipi sayısının çok fazla ve alan başına düşen veri tipi sayısının az olması dolayısıyla değerlendirmeye alınmamıştır.

Değişken2’den 10’ kadar olan alanlarda, 10 milyon satırda kullanım sayısı çok düşük olması, alan başına düşen veri tipi sayısının yüksek oluşu ve blok tipi sayısının bu kadar yüksek olması dolayısıyla değerlendirmeye alınmamıştır.

Kukla İsmi ve Koordinatlar veri alanı çok fazla veri tipi içermesi ve alan başına düşen veri tipi oranının az olması bu alanın ilişkisel olmayan çok sayıda veriyi içerdiğini kanıtlamakta, dolayısıyla değerlendirmeye alınmamaktadır.

Kukla Tipi (Scrip-Type) alanın boş satır sayısı 0, veri tipi sayısı 3 ve alan başına düşen veri tipi sayısı 3333333 olduğu belirlenmiştir. Bu veri 3 tip; Kukla, Özel Taş ve Sahne den oluşmaktadır. “Şekil 10.” da gösterildiği gibi bu alanda en çok kullanılan veri tipi kukla (sprite), sonrasında Özel taşlar(procDef) ve son olarak da sahne(stage) sıralaması gelmektedir. Özellik seçiminde değerlendirilecek alanlar arasında bulunmaktadır.



**Şekil 10.** Kukla Tipi (Scrip-Type) veri gösterimi

“Tablo 5.” Blok tipi alanı incelendiğinde boş değer sayısı 0, veri tipi sayısı 139 ve alan başına düşen veri tipi sayısı 71942 olarak belirtilmiştir. “Şekil 11.” de 139 veri tipinin her birinin veri tabanında kullanım sayısı gösterilmiştir. Özellik seçiminde değerlendirilecek alanlar arasında bulunmaktadır.

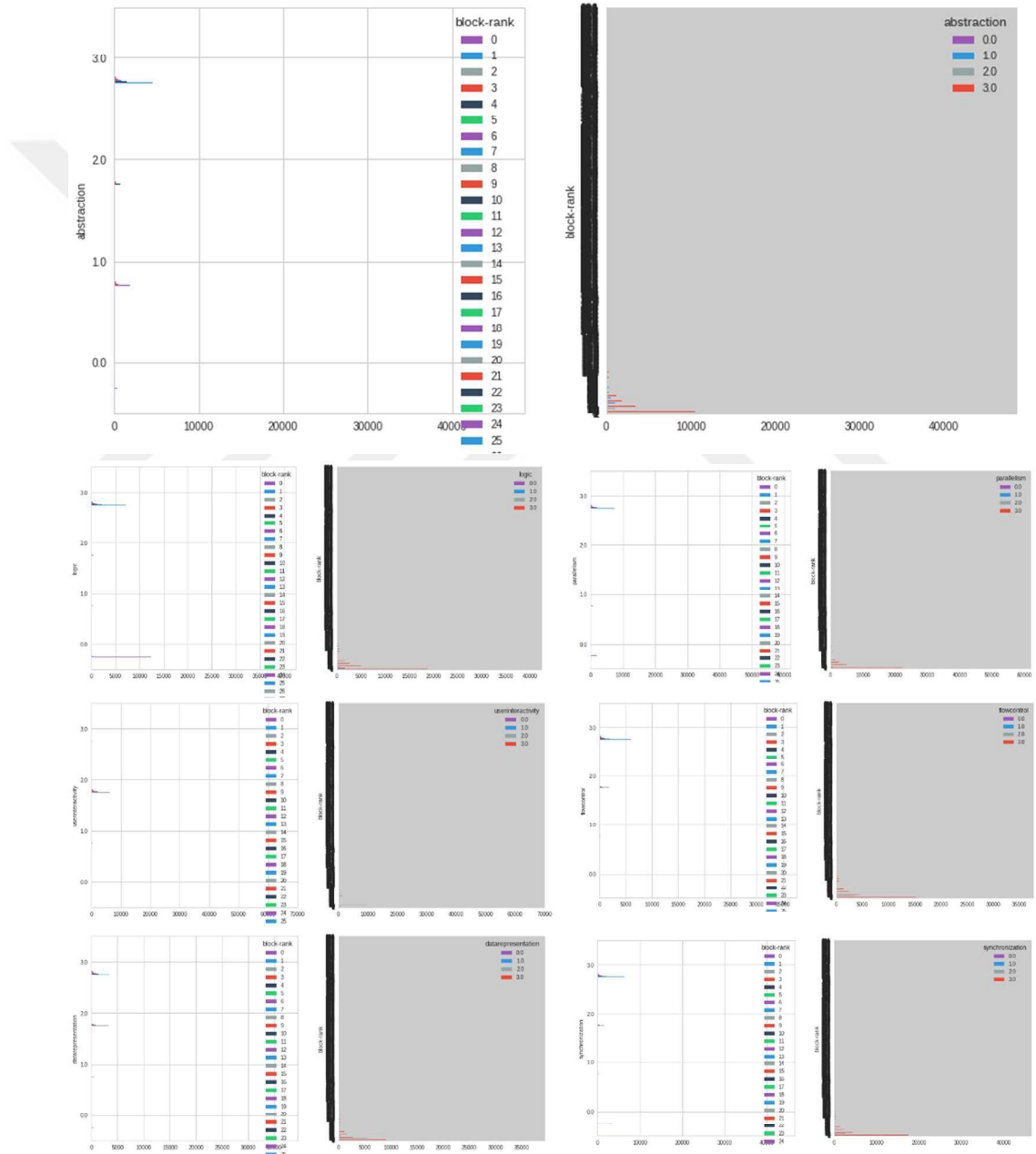


### 5.3.2. ÖZELLİK SEÇİMİ

Bu bölümde proje ve meta adındaki ilişkisel veri tabanındaki veriler analiz edilmektedir. Proje veri tabanındaki Kukla Sırası (Script -Rank), Blok Sırası (Block-Rank), Blok Tipi (Block-Type) ve Kukla Tipi (Scrip-Type) veri alanlarının, Meta veri tabanında bulunan 7 puan türüne; Soyutlama ve Problem Ayırıştırma (Abstraction), Paralellik (Parallelism), Mantıksal Düşünme (Logic), Eş Zamanlılık (Synchronization), Akış Kontrolü (FlowControl), Kullanıcı Etkileşimi (UserInteractivity) ve Veri Gösterimine (DataRepresentation) olan etkisi tek tek incelenmiştir.

### 5.3.2.1. Blok Sırası (Block-Rank)

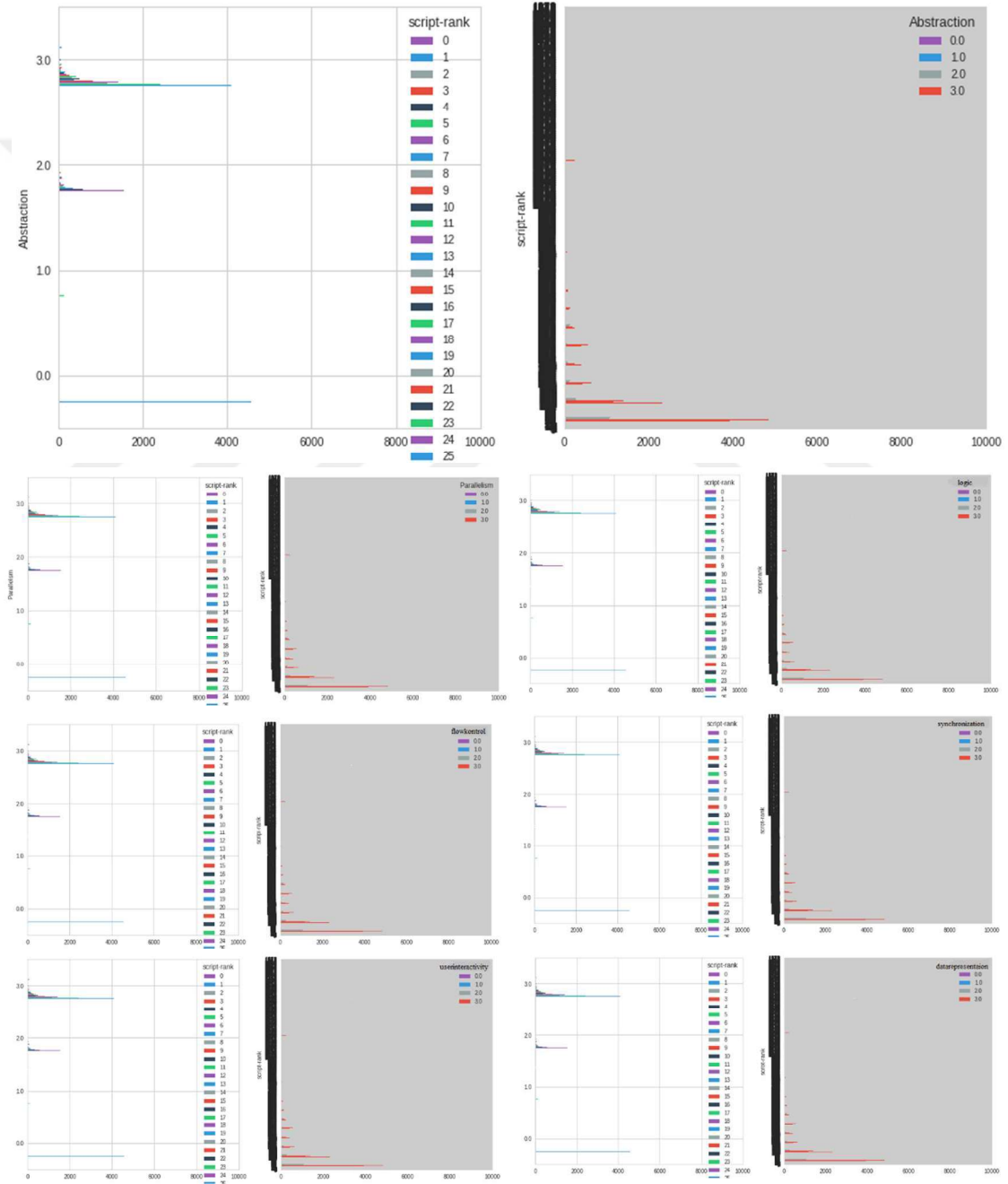
"Şekil 12." de proje veritabanında bulunan blok sırasının, meta veritabanındaki 7 puan türü üzerindeki etkisi incelenmiştir. Blok sırası veri alanının bu paunlar ile ilişkisinin çok düşük olduğu ve puanları tahmin etmede yetersiz kalacağı gözlemlenmiştir.



Şekil 12. Blok sırasının, meta veri tabanındaki 7 puan türüne etkisi

### 5.3.2.2.Kukla Sırası (Script -Rank)

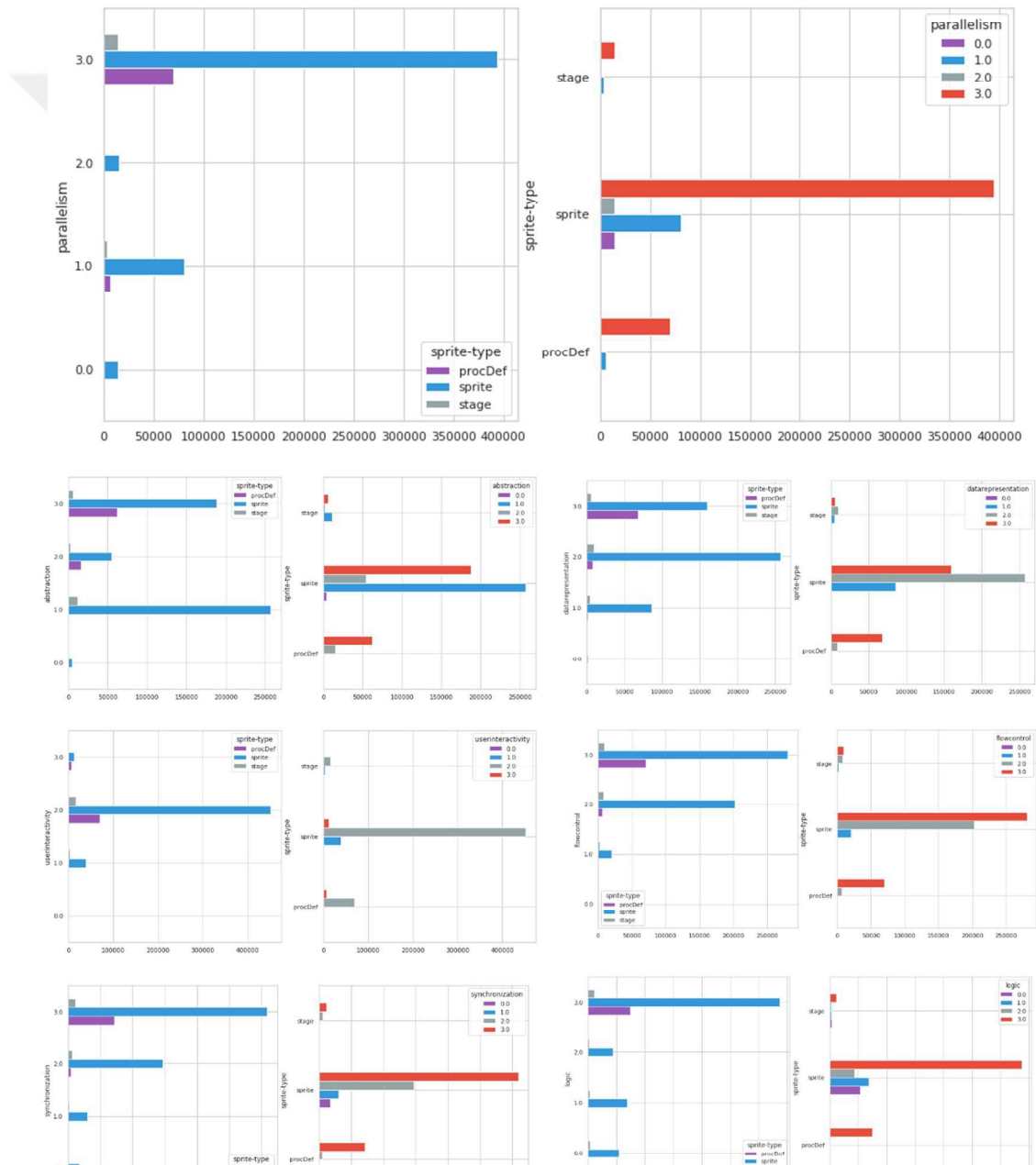
"Şekil 13." de proje veri tabanında bulunan Kukla Sırası, meta veri tabanındaki 7 puan türü üzerindeki etkisi incelenmiştir. Kukla sırası veri alanının bu puanlar ile ilişkisinin çok düşük olduğu ve puanları tahmin etmede yetersiz kalacağı gözlemlenmiştir.



Şekil 13. Kukla Sırası (Script -Rank) meta veri tabanındaki 7 puan alanına etkisi

### 5.3.2.3.Kukla Tipi (Scrip-Type)

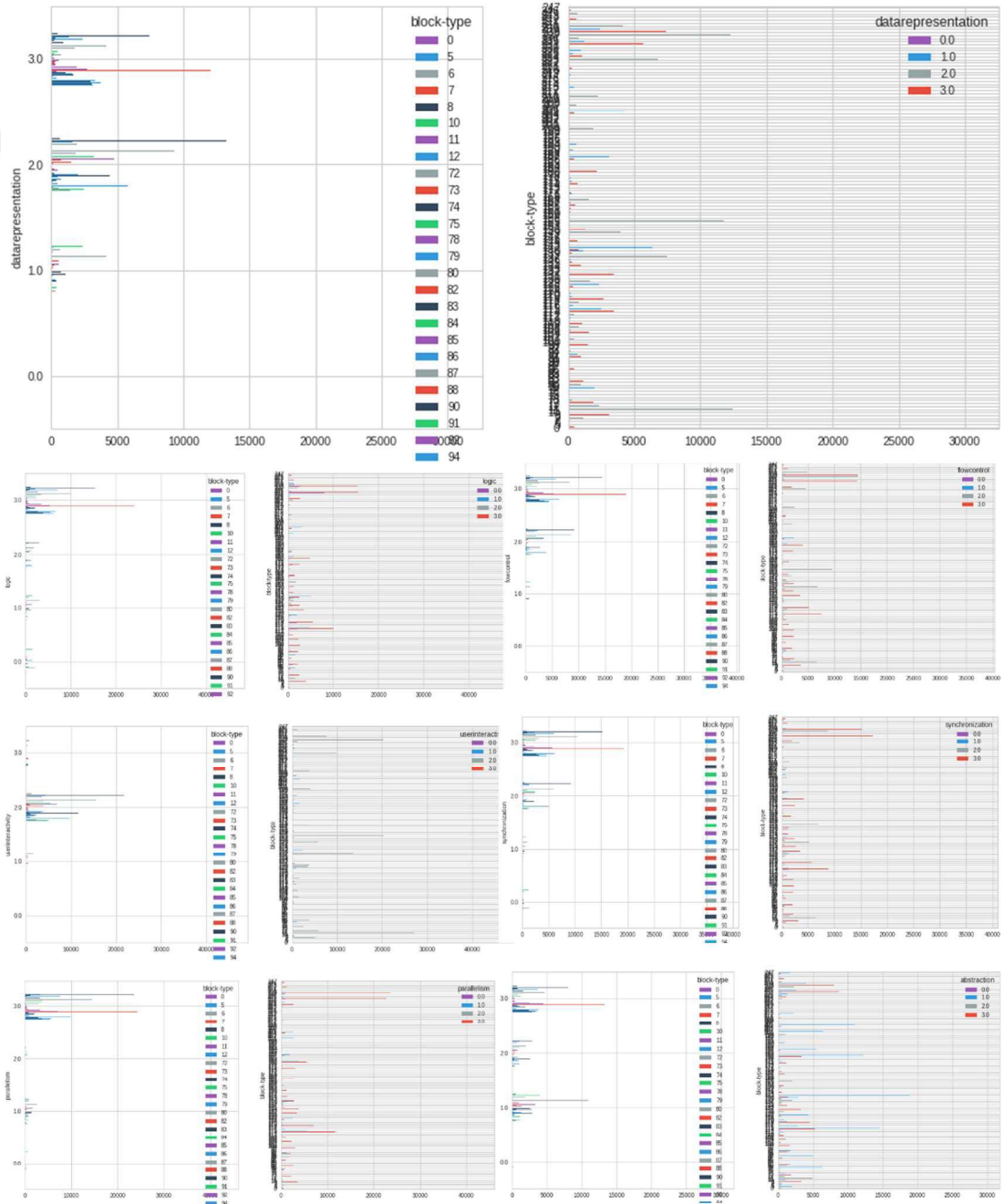
“Şekil 14.” de Proje veritabanında bulunan kukla tipi alanının, meta veritabanındaki 7 puan türü üzerindeki etkisi incelenmiştir. Kukla tipi alanının bu paunlar ile ilişkisinin çok yüksek olduğu gözlemlenmiştir. Fakat puanları tahmin etmede anlamlı olacak kadar kategoriye sahip olmaması özellik seçimi için yetersiz kalmıştır.



Şekil 14. Kukla Tipi (Scrip-Type) meta veri tabanındaki 7 puan alanına etkisi

### 5.3.2.4. Blok Tipi (Block-Type)

“Şekil 15.” Proje veritabanında bulunan blok tipi alanının, meta veritabanındaki 7 puan türü üzerindeki etkisi incelenmiştir. 139 veri tipi olan bu alan ile 7 puan türü arasındaki ilişkinin çok önemli olduğu gözlemlenmiştir. Makine Öğrenmesinin puan tahmini için en önemli alanın blok tipi olduğu belirlenmiştir.



Şekil 15. Blok Tipi (Block-Type) meta veri tabanındaki 7 puan alanına etkisi



### 5.3.3. ÖZELLİK MÜHENDİSLİĞİ

Blok tipi alanındaki değerler analiz edilirken kategorik veriye dönüştürebileceği gözlemlenmiştir. Bu işlemi yapmak için önce 139 benzersiz blok tipinin her biri için yeni bir sütun oluşturularak, numaralandırılmış bir sözlüğe dönüştürülmüştür. Ardından proje veri tabanındaki blok tipi alanını bu dizge değerleriyle sayısal veriye dönüştürülmüştür. Bu sayede algoritmanın kategorik veriler arasındaki ilişikliği görebilmesi ve sayısal değerlerle daha rahat işlem yapması sağlanmıştır. Blok tipi alanında dönüştürülen sözlük Ek-B. de sunulmuştur.

#### **5.4.UYGUN MODELİN BULUNMASI, EĞİTİLMESİ VE DEĞERLENİLMESİ**

Modellerin karşılaştırılması ve en iyi performanslı modeli bulunması için öncelikle özellik seçimi bölümünde belirlenen proje veri setindeki blok tipi alanı ve proje numaraları Pandalar yardımıyla Colaboratory platformunda içeriye alınmıştır.

Meta veri tabanında proje numaraları ile beraber şu alanlardan biri içeriye alınmıştır; Soyutlama ve Problem Ayrıştırma, Paralellik, Mantıksal Düşünme, Eş Zamanlılık, Akış Kontrolü, Kullanıcı Etkileşimi, Veri Gösterimi. Bu alanlar içeriye alınırken hiçbir değer girilmediği “null” olan veriler çıkarılmıştır. Model eğitilmesi yapılırken belirtilen 7 meta veri alanı için tek tek model denemesi yapılmıştır.

İlişkili verilerimizin olduğu proje ve meta veri setleri birincil anahtar olarak proje numaraları üzerinden birbirine bağlanmıştır. Bu işlemler için gerçekleştirilen kodlar komutlar EK-C ye eklenmiştir.

Bu aşamadan sonra modellerin denenmesi için çalışmaya başlanmıştır. Model seçimi yapılırken veri setinde etiketli verilerin bulunması dolayısıyla, Gözetimli öğrenme yöntemleri denenmiştir. Öncelikle Sınıflanma algoritmalarından Derin Sinir Ağları sonrasında Regresyon algoritmalarından Basit kullanılarak model eğitilmiş ve sonuçlara göre en uygun model seçilmiştir.

### 5.4.1. Derin Sinir Ağları (Deep Neural Networks) Modeli

Sınıflama yöntemlerinden Derin Sinir Ağları yöntemiyle model eğitimeye başlamıştır [28]. Modeli eğitmek için özellik seçimi bölümünde proje veri setindeki blok tipi alanı belirlenmiştir. Daha önceden Meta verilerle birleştirilen blok tipi alanı herhangi bir sorunda kaybolmaması için bir turşuya (pickle) atılmıştır. Modeli eğitmek için içeriye veri setinin %50si alınmıştır.

Tensorflow kütüphanesinden faydalanarak Proje veri tabanındaki blok tipi alanından özellik sütunları oluşturulmuştur. Etiket belirlemek için proje veri setindeki BİD becerilerinin 7 boyutundan ilki Abstraction (Soyutlama ve Problem Ayırıştırma) alanı kullanılmıştır. Etiket ve Özellik sayısı 30000000 olarak gözlemlenmiştir. Proje veri tabanındaki blok tipi alanını içeren etiketler sayısal verilere dönüştürülerek model eğitime uygun hale getirilmiştir. İçeri alınan veri setinin %80 eğitim, %20 si test için ayrılmıştır. Sınıf sayısı 0,1,2 ve 3 değerlerini alan Abstraction alanından 4 olarak belirlenmiş, gizli birim katmanı 512 ile 256 arası girilmiştir. Model eğitimi 200000 adımdan oluşmaktadır. Bu aşamaya kadar olan tüm işlemler EK-D de belirtilmiştir.

Modelin eğitimi sonucunda sınıflama doğruluğu %56.12 elde edilmiştir. Elde edilen modelin sonuçları “Tablo 27.” de verilmiştir. Bu sonuçlar sırayla; projenin numarası, veri setindeki puanı ve modelin tahmin ettiği sonuçlar şeklinde belirtilmiştir. Test sonuçlarının tahmin oranının düşük olması bu modelin bu veri seti için uygun olmadığını göstermektedir.

Tablo 27. Derin sinir ağıları model sonuçları

(60002538, 3, 0), (60034304, 3, 2), (60058876, 3, 0), (60063608, 3, 0), (60068444, 3, 0), (60074986, 3, 3), (60082620, 3, 2), (60122018, 3, 3), (60122926, 3, 1), (60145364, 3, 3), (60204614, 3, 0), (60225348, 3, 0), (60229414, 3, 1), (60234624, 3, 1), (60295774, 3, 2), (60338738, 3, 2), (60440188, 3, 2), (60505632, 3, 2), (60530658, 3, 3), (60531878, 3, 1), (60538568, 3, 3), (60601472, 3, 1), (60640752, 3, 3), (60660644, 3, 1), (60664714, 3, 3), (60703742, 3, 3), (60709712, 3, 3), (60771904, 3, 3), (60817948, 3, 0), (60819578, 3, 0), (60864856, 3, 1), (60876876, 3, 1), (60905034, 3, 1), (60916678, 3, 0), (60948720, 3, 1), (61068370, 3, 0), (61099246, 3, 1), (61101312, 3, 1), (61103088, 3, 3), (61179410, 3, 0), (61217620, 3, 0), (61223046, 3, 1), (61231694, 3, 0), (61253490, 3, 2), (61298472, 3, 2), (61341624, 3, 3), (61367188, 3, 3), (61408310, 3, 3), (61412810, 3, 2), (61422754, 3, 0), (61435682, 3, 2), (61438162, 3, 2), (61442058, 3, 3), (61442068, 3, 3), (61484518, 3, 2), (61532642, 3, 1), (61580118, 3, 3), (61588702, 3, 2), (61590618, 3, 3), (61605534, 3, 1), (61633772, 3, 1), (61690732, 3, 1), (61698128, 3, 1), (61731232, 3, 3), (61738156, 3, 1), (61741688, 3, 1), (61779758, 3, 1), (61789996, 3, 1), (61815506, 3, 2), (61830498, 3, 0), (61842086, 3, 3), (61865206, 3, 0), (61903538, 3, 2), (61959788, 3, 2), (61964618, 3, 1), (62031836, 3, 1), (62035068, 3, 1), (62037416, 3, 0), (62048560, 3, 0), (62050404, 3, 0), (62052128, 3, 2), (62065184, 3, 1), (62070302, 3, 2), (62076846, 3, 1), (62079856, 3, 1), (62080490, 3, 1), (62197046, 3, 2), (62239456, 3, 3), (62256746, 3, 1), (62262954, 3, 1), (62303268, 3, 2), (62371622, 3, 1), (62374770, 3, 1), (62488770, 3, 1), (62505930, 3, 1), (62537918, 3, 3), (62557982, 3, 0), (62579570, 3, 2), (62597294, 3, 2), (62599884, 3, 2),
-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

#### 5.4.2. Basit Doğrusal Regresyon (Simple Linear Regression) Modeli

Modeli eğitmek için özellik seçimi bölümünde proje veri setindeki blok tipini alanı belirlenmiştir, tek girişli veri tahmin edilmek istendiğinden Basit Doğrusal Regresyon yöntemiyle model eğitilmiştir [29]. Öncelikle Proje veri tabanının blok tipi alanındaki değerlerden benzersiz olanlar listelenmiş ve 0'dan 139'a kadar numaralandırılarak EK-B de gösterilen blok tipi çeviri sözlüğü oluşturulmuştur. Bu sözlükten faydalanarak özellik mühendisliği bölümünde açıklandığı üzere proje veri tabanındaki blok tipi alanı sayısal verilere dönüştürülerek model eğitime uygun hale getirilmiştir. Daha önceden Meta verilerle birleştirilen blok tipi alanı herhangi bir sorunda kaybolmaması için bir turşuya (pickle) atılmıştır

Tahminleri meta veri tabanındaki BİD becerilerinin 7 boyutu için tek tek model eğitilmiştir. Her bir tahminde, seçilen BİD becerileri alanının blok tipinin 139 kategorik değeri için etiket benzeri tahmin vektörü oluşturulmuştur. Meta verilerin 7 boyutunun her biri 0,1,2 ve 3 olmak üzere 4 etiket türü içermektedir.

Verilerin %80 eğitim ve doğrulama için, %20 si test için bölünmüştür. Eğitim setinin %90'nı eğitim için %10 doğrulama için bölünmüştür. Bu aşamaya kadar yapılan tüm işlemlerin kodları Ek-E de belirtilmiştir.

Meta veri tabanındaki BİD becerilerinin 7 boyutu; Abstraction (Soyutlama ve Problem Ayırıştırma), Paralellik (Parallelism), Mantıksal Düşünme (Logic), Eş Zamanlılık (Synchronization), Akış Kontrolü (FlowControl), Kullanıcı Etkileşimi (UserInteractivity) ve Veri Gösterimi (DataRepresentation) için tek tek model eğitimi yapılmıştır. 5.4.2.1. den başlayarak 5.4.2.7.ye kadar sırayla model eğitimleri açıklanmıştır.

#### 5.4.1.1. Abstraction (Soyutlama Ve Problem Ayrıştırma) alanı model eğitilmesi

Model denemeleri sonucu model optimizasyonu “Tablo 7.” de belirtildiği üzere yapılmıştır. Döngünün çalışacağı adım sayısını 10000, algoritmanın çalışacağı yığın boyutu 16384 ve algoritmanın çalışacağı öğrenme hızı 1e-2 olarak belirlenmiştir.

Tablo 7. Abstraction alanının optimizasyon değerleri

<b>Adım_Sayısı=10000</b>
<b>Yığın_Boyutu=16384</b>
<b>Oğrenme_hızı= 1e-2</b>

Model eğitimi sırasında verilerin tutulabilmesi için yer tutucu ve değişkenlerden yararlanılmıştır. Tensorflow Makine Öğrenmesi Kütüphanesi yardımıyla model eğitimi öncesi eğitim doğrulama ve test veri seti için değişkenler, yertutucular (placeholder) tanımlanmıştır. Değişken kısmında Bias (ağırlık) için özellik ve etiket sayısını, Weight(Önyargı) için etiket sayısı değişkenleri tanımlanmıştır. Sonrasında “Tablo 8.” de belirtilen eğitim hesaplaması, geçerlilik eğitim hesaplaması ve kayıp hesabı tanımlanmıştır. Bir sonraki aşamada “Tablo 7.” ’e göre Öğrenme Hızına göre optimizasyon ayarları belirlenmiştir. Son aşamada “Tablo 9.” da gösterilen eğitim, doğrulama ve test tahminleri yer tutucuları tanımlanmıştır.

Tablo 8. Eğitim Hesaplaması

```

logit = tf.matmul(tf_train_dataset, weights) + biases
valid_logit = tf.matmul(tf_valid_dataset, weights) + biases
loss = tf.reduce_mean(tf.nn.softmax_cross_entropy_with_logits_v2(
labels=tf_train_labels, logits=logits))

```

Tablo 9. Tahminler için yertutucular

```

train_predicton = tf.nn.softmax(logit)
valid_predicton = tf.nn.softmax(valid_logit)
test_predicton = tf.nn.softmax(tf.matmul(tf_test_dataset, weights) + biases)
tf_Abstraction_predicton = tf.nn.softmax(tf.matmul(tf_Abstraction_dataset,
weights) + biases)

```

“Tablo 10.” da Doğruluk puanları için kullanılan hesaplama gösterilmiştir; doğruluk puanı, doğru tahmin edilen değerlerin sayısının tahmin edilen değerlerin sayısına bölünmesiyle hesaplanmaktadır.

Tablo 10. Doğruluk Hesabı

$$\text{Doğruluk Puanı} = \frac{\text{Doğru Tahmin Edilen Değerlerin Sayısı}}{\text{Tahmin Edilen Değerlerin Sayısı}}$$

Model eğitime belirtilen etiket ve özelliğe göre başlatılmıştır. Minimum yığınlar ayrılmış, yığınlar %500 de bir yenilenirken çıktı olarak ekrana aşama başına Yığın Kaybı, minimum Yığın Doğrulama ve Doğrulama (Validation) Doğruluğu yazdırılmıştır. Yığınlar bittiğinde test doğruluğu ve Abstraction tahmin doğruluğu yazılmıştır. Bu işlemler sonlanırken ileride çevrimiçi değerlendirme aracına

dönüştürebilmek için Abstraction klasörüne Önyargı (Bias) ve Ağırlık (Weight) verileri csv dosyası olarak kaydedilmiştir. Yapılan bu işlemlerin kodları EK-F'ye eklenmiştir.

Model eğitimi çıktıları “Tablo 11.” de gösterilmiştir. Sonuçlara göre Proje veri tabanındaki Blok Tipi alanı ile Meta veri tabanındaki Abstraction (Soyutlama Ve Problem Ayırıştırma) alanını tahmin doğruluğu %92.3 olarak gözlemlenmiştir.

Tablo 11. Abstraction alanının model eğitimi çıktıları

Eğitim Başladı
Asamada_yığın_kaybı0: 27.65372085571289
Yığın_dogrulugu:31.8%
Gecerlilik_dogrulugu:27.0%
Asamada_yığın_kaybı500: 3.8299779891967773
Yığın_dogrulugu:63.4%
Gecerlilik_dogrulugu:61.5%
Asamada_yığın_kaybı1000: 1.4377601146697998
Yığın_dogrulugu:72.1%
Gecerlilik_dogrulugu:71.6%
Asamada_yığın_kaybı1500: 1.1244620084762573
Yığın_dogrulugu:81.1%
Gecerlilik_dogrulugu:79.8%
Asamada_yığın_kaybı2000: 1.2582838535308838
Yığın_dogrulugu:76.6%
Gecerlilik_dogrulugu:85.3%
Asamada_yığın_kaybı2500: 0.8558763265609741



Yığın\_dogrulugu:81.1%

Gecerlilik\_dogrulugu:87.6%

Asamada\_yığın\_kaybı3000: 0.8637853860855103

Yığın\_dogrulugu:82.9%

Gecerlilik\_dogrulugu:86.7%

Asamada\_yığın\_kaybı3500: 0.6530756950378418

Yığın\_dogrulugu:84.7%

Gecerlilik\_dogrulugu:89.0%

Asamada\_yığın\_kaybı4000: 0.7955299615859985

Yığın\_dogrulugu:82.4%

Gecerlilik\_dogrulugu:89.5%

Asamada\_yığın\_kaybı4500: 0.3803594410419464

Yığın\_dogrulugu:89.0%

Gecerlilik\_dogrulugu:89.9%

Asamada\_yığın\_kaybı5000: 0.27404001355171204

Yığın\_dogrulugu:92.1%

Gecerlilik\_dogrulugu:89.5%

Asamada\_yığın\_kaybı5500: 0.5251133441925049

Yığın\_dogrulugu:86.6%

Gecerlilik\_dogrulugu:89.9%

Asamada\_yığın\_kaybı6000: 0.25509393215179443

Yığın\_dogrulugu:92.7%

Gecerlilik\_dogrulugu:89.9%

Asamada\_yıgın\_kaybı6500: 0.25923195481300354

Yıgın\_dogrulugu:91.5%

Gecerlilik\_dogrulugu:90.4%

Asamada\_yıgın\_kaybı7000: 0.2862686514854431

Yıgın\_dogrulugu:92.1%

Gecerlilik\_dogrulugu:90.4%

Asamada\_yıgın\_kaybı7500: 0.21568141877651215

Yıgın\_dogrulugu:93.3%

Gecerlilik\_dogrulugu:90.8%

Asamada\_yıgın\_kaybı8000: 0.4104258418083191

Yıgın\_dogrulugu:90.2%

Gecerlilik\_dogrulugu:91.3%

Asamada\_yıgın\_kaybı8500: 0.3585742115974426

Yıgın\_dogrulugu:93.9%

Gecerlilik\_dogrulugu:91.3%

Asamada\_yıgın\_kaybı9000: 0.24528570473194122

Yıgın\_dogrulugu:93.9%

Gecerlilik\_dogrulugu:91.8%

Asamada\_yıgın\_kaybı9500: 0.33058595657348633

Yıgın\_dogrulugu:94.5%

Gecerlilik\_dogrulugu:90.8%

Test-doğruluğu-Test-accuracy:92.3%

Test-doğruluğu-Test-accuracy:100.0%

#### 5.4.1.2.Paralellik (Parallelism) alanı model eğitilmesi

Model denemeleri sonucu model optimizasyonu “Tablo 12.” da belirtildiği üzere yapılmıştır. Döngünün çalışacağı adım sayısını 10000, algoritmanın çalışacağı yığın boyutu 16384 ve algoritmanın çalışacağı öğrenme hızı  $1.5 \cdot 10^{-2}$  olarak belirlenmiştir.

Tablo 12. Paralellik (Parallelism)alanının optimizasyon değerleri

<b>Adım_Sayısı=10000</b>
<b>Yığın_Boyutu=16384</b>
<b>Öğrenme_hızı= <math>1.5 \cdot 10^{-2}</math></b>

Model eğitimi sırasında verilerin tutulabilmesi için yer tutucu ve değişkenlerden yararlanılmıştır. Tensorflow Makine Öğrenmesi Kütüphanesi yardımıyla model eğitimi öncesi eğitim doğrulama ve test veri seti için değişkenler, yertutucular (placeholder) tanımlanmıştır. Değişken kısmında Bias (ağırlık) için özellik ve etiket sayısını, Weight(Önyargı) için etiket sayısı değişkenleri tanımlanmıştır. “Tablo 8’e göre eğitim hesaplaması, geçerlilik eğitim hesaplaması ve kayıp hesaplaması tanımlanmıştır. Bir sonraki aşamada Öğrenme Hızına göre “Tablo 12.” deki optimizasyon ayarları belirtilmiş. Son aşamada “Tablo 9.” da gösterilen eğitim, doğrulama ve test tahminleri yertutucuları tanımlanmıştır.

“Tablo 10.” da belirtilmiş olan Doğruluk puanı hesaplaması gösterilmiştir; doğruluk puanı, doğru tahmin edilen değerlerin sayısının tahmin edilen değerlerin sayısına bölünmesiyle hesaplanmaktadır.

Model eğitilmeye belirtilen etiket ve özelliğe göre başlatılmıştır. Minimum yığınlara ayrılmış, yığınlar %500 de bir yenilenirken çıktı olarak ekrana aşama başına

Yığın Kaybı, minimum Yığın Doğrulama ve Doğrulama (Validation) Doğruluğu yazdırılmıştır. Yığınlar bittiğinde test doğruluğu ve Paralellik (Parallelism) tahmin doğruluğu yazılmıştır. Bu işlemler sonlanırken ileride çevrimiçi değerlendirme aracına dönüştürebilmek için Paralellik klasörüne Önyargı (Bias) ve Ağırlık (Weight) verileri csv dosyası olarak kaydedilmiştir. Yapılan bu işlemlerin kodları EK-G'ye eklenmiştir.

Model eğitimi çıktıları “Tablo 13.” de gösterilmiştir. Sonuçlara göre Proje veri tabanındaki Blok Tipi alanı ile Meta veri tabanındaki Paralellik (Parallelism) alanının tahmin doğruluğu %92.3 olarak gözlemlenmiştir.

Tablo 13. Paralellik (Parallelism) alanının model eğitimi çıktıları

Başladı
Asamada_yığın_kaybı0: 27.65372085571289
Yığın_dogrulugu:31.8%
Gecerlilik_dogrulugu:27.0%
Asamada_yığın_kaybı500: 3.8299779891967773
Yığın_dogrulugu:63.4%
Gecerlilik_dogrulugu:61.5%
Asamada_yığın_kaybı1000: 1.4377601146697998
Yığın_dogrulugu:72.1%
Gecerlilik_dogrulugu:71.6%
Asamada_yığın_kaybı1500: 1.1244620084762573
Yığın_dogrulugu:81.1%
Gecerlilik_dogrulugu:79.8%
Asamada_yığın_kaybı2000: 1.2582838535308838
Yığın_dogrulugu:76.6%

Gecerlilik\_dogrulugu:85.3%

Asamada\_yigin\_kaybi2500: 0.8558763265609741

Yigin\_dogrulugu:81.1%

Gecerlilik\_dogrulugu:87.6%

Asamada\_yigin\_kaybi3000: 0.8637853860855103

Yigin\_dogrulugu:82.9%

Gecerlilik\_dogrulugu:86.7%

Asamada\_yigin\_kaybi3500: 0.6530756950378418

Yigin\_dogrulugu:84.7%

Gecerlilik\_dogrulugu:89.0%

Asamada\_yigin\_kaybi4000: 0.7955299615859985

Yigin\_dogrulugu:82.4%

Gecerlilik\_dogrulugu:89.5%

Asamada\_yigin\_kaybi4500: 0.3803594410419464

Yigin\_dogrulugu:89.0%

Gecerlilik\_dogrulugu:89.9%

Asamada\_yigin\_kaybi5000: 0.27404001355171204

Yigin\_dogrulugu:92.1%

Gecerlilik\_dogrulugu:89.5%

Asamada\_yigin\_kaybi5500: 0.5251133441925049

Yigin\_dogrulugu:86.6%

Gecerlilik\_dogrulugu:89.9%

Asamada\_yigin\_kaybi6000: 0.25509393215179443

Yigin\_dogrulugu:92.7%

Gecerlilik\_dogrulugu:89.9%

Asamada\_yigin\_kaybi6500: 0.25923195481300354

Yigin\_dogrulugu:91.5%

Gecerlilik\_dogrulugu:90.4%

Asamada\_yigin\_kaybi7000: 0.2862686514854431

Yigin\_dogrulugu:92.1%

Gecerlilik\_dogrulugu:90.4%

Asamada\_yigin\_kaybi7500: 0.21568141877651215

Yigin\_dogrulugu:93.3%

Gecerlilik\_dogrulugu:90.8%

Asamada\_yigin\_kaybi8000: 0.4104258418083191

Yigin\_dogrulugu:90.2%

Gecerlilik\_dogrulugu:91.3%

Asamada\_yigin\_kaybi8500: 0.3585742115974426

Yigin\_dogrulugu:93.9%

Gecerlilik\_dogrulugu:91.3%

Asamada\_yigin\_kaybi9000: 0.24528570473194122

Yigin\_dogrulugu:93.9%

Gecerlilik\_dogrulugu:91.8%

Asamada\_yigin\_kaybi9500: 0.33058595657348633

Yigin\_dogrulugu:94.5%

Gecerlilik\_dogrulugu:90.8%

Test-dogruluğu-Test-accuracy:92.3%

Test-dogruluğu-Test-accuracy:100.0%

#### 5.4.1.3.Mantıksal Düşünme (Logic) alanı model eğitilmesi

Model denemeleri sonucu model optimizasyonu “Tablo 14.” da belirtildiği üzere yapılmıştır. Döngünün çalışacağı adım sayısını 10000, algoritmanın çalışacağı yığın boyutu 16384 ve algoritmanın çalışacağı öğrenme hızı 1e-2 olarak belirlenmiştir.

Tablo 14. Mantıksal Düşünme (Logic) alanının optimizasyon değerleri

<b>Adım_Sayısı=10000</b>
<b>Yığın_Boyutu=16384</b>
<b>Oğrenme_hızı= 1e-2</b>

Model eğitimi sırasında verilerin tutulabilmesi için yer tutucu ve değişkenlerden yararlanılmıştır. Tensorflow Makine Öğrenmesi Kütüphanesi yardımıyla model eğitimi öncesi eğitim doğrulama ve test veri seti için değişkenler, yertutucular (placeholder) tanımlanmıştır. Değişken kısmında Bias (ağırlık) için özellik ve etiket sayısını, Weight(Önyargı) için etiket sayısı değişkenleri tanımlanmıştır. Sonrasında “Tablo 8.”e göre eğitim hesaplaması, geçerlilik eğitim hesaplaması ve kayıp hesaplaması tanımlanmıştır. Bir sonraki aşamada Öğrenme Hızına göre “Tablo 14.” deki optimizasyon ayarları belirtilmiş. Bir sonraki aşamada “Tablo 9.” da gösterilen eğitim, doğrulama ve test tahminleri yertutucuları tanımlanmıştır.

“Tablo 10.” da belirtilmiş olan Doğruluk puanı hesaplaması gösterilmiştir; doğruluk puanı, doğru tahmin edilen değerlerin sayısının tahmin edilen değerlerin sayısına bölünmesiyle hesaplanmaktadır.

Model eğitilmeye belirtilen etiket ve özelliğe göre başlatılmıştır. Minimum yığınlara ayrılmış, yığınlar %500 de bir yenilenirken çıktı olarak ekrana aşama başına

Yığın Kaybı, minimum Yığın Doğrulama ve Doğrulama (Validation) Doğruluğu yazdırılmıştır. Yığınlar bittiğinde test doğruluğu ve Mantıksal Düşünme (Logic) tahmin doğruluğu yazılmıştır. Bu işlemler sonlanırken ileride çevrimiçi değerlendirme aracına dönüştürebilmek için Mantıksal Düşünme klasörüne Önyargı (Bias) ve Ağırlık (Weight) verileri csv dosyası olarak kaydedilmiştir. Yapılan bu işlemlerin kodları EK-H'ye eklenmiştir.

Model eğitimi çıktıları “Tablo 15.” de gösterilmiştir. Sonuçlara göre Proje veri tabanındaki Blok Tipi alanı ile Meta veri tabanındaki Mantıksal Düşünme (Logic) alanının tahmin doğruluğu % 94.5 olarak gözlemlenmiştir.

Tablo 15. Mantıksal Düşünme (Logic) alanının model eğitimi çıktıları

Başladı
Asamada_yığın_kaybı 0: 34.26807403564453
Yığın_dogrulugu:17.1%
Gecerlilik_dogrulugu:19.7%
Asamada_yığın_kaybı 500: 5.4691596031188965
Yığın_dogrulugu:59.7%
Gecerlilik_dogrulugu:66.1%
Asamada_yığın_kaybı 1000: 2.435612440109253
Yığın_dogrulugu:69.0%
Gecerlilik_dogrulugu:72.1%
Asamada_yığın_kaybı 1500: 1.4854099750518799
Yığın_dogrulugu:73.1%
Gecerlilik_dogrulugu:79.8%



Asamada\_yığın\_kaybı 2000: 1.6773366928100586

Yığın\_dogrulugu:82.7%

Gecerlilik\_dogrulugu:84.4%

Asamada\_yığın\_kaybı 2500: 0.6536333560943604

Yığın\_dogrulugu:82.9%

Gecerlilik\_dogrulugu:85.8%

Asamada\_yığın\_kaybı 3000: 0.5643453598022461

Yığın\_dogrulugu:87.2%

Gecerlilik\_dogrulugu:85.8%

Asamada\_yığın\_kaybı 3500: 0.5671780109405518

Yığın\_dogrulugu:86.6%

Gecerlilik\_dogrulugu:88.1%

Asamada\_yığın\_kaybı 4000: 0.6057944297790527

Yığın\_dogrulugu:89.0%

Gecerlilik\_dogrulugu:88.5%

Asamada\_yığın\_kaybı 4500: 0.5654155611991882

Yığın\_dogrulugu:88.4%

Gecerlilik\_dogrulugu:90.4%

Asamada\_yığın\_kaybı 5000: 0.47041958570480347

Yığın\_dogrulugu:89.6%

Gecerlilik\_dogrulugu:90.8%

Asamada\_yığın\_kaybı 5500: 0.3453288674354553

Yığın\_dogrulugu:89.6%

Gecerlilik\_dogrulugu:92.2%

Asamada\_yığın\_kaybı 6000: 0.32265105843544006

Yığın\_dogrulugu:90.8%

Gecerlilik\_dogrulugu:93.1%

Asamada\_yığın\_kaybı 6500: 0.43259304761886597

Yığın\_dogrulugu:89.6%

Gecerlilik\_dogrulugu:93.1%

Asamada\_yığın\_kaybı 7000: 0.2794731855392456

Yığın\_dogrulugu:92.7%

Gecerlilik\_dogrulugu:94.0%

Asamada\_yığın\_kaybı 7500: 0.2647860050201416

Yığın\_dogrulugu:93.9%

Gecerlilik\_dogrulugu:93.1%

Asamada\_yığın\_kaybı 8000: 0.22181764245033264

Yığın\_dogrulugu:93.3%

Gecerlilik\_dogrulugu:92.7%

Asamada\_yığın\_kaybı 8500: 0.18523038923740387

Yığın\_dogrulugu:95.1%

Gecerlilik\_dogrulugu:92.7%

Asamada\_yığın\_kaybı 9000: 0.2486323118209839

Yığın\_dogrulugu:92.7%

Gecerlilik\_dogrulugu:93.1%

Asamada\_yığın\_kaybı 9500: 0.33131909370422363

Yığın\_dogrulugu:93.3%

Gecerlilik\_dogrulugu:94.0%

Test-doğruluğu-Test-accuracy:94.5%

Test-doğruluğu-Test-accuracy:100.0%

#### 5.4.1.4.Eş Zamanlılık (Synchronization) alanı model eğitilmesi

Model denemeleri sonucu model optimizasyonu “Tablo 16.” da belirtildiği üzere yapılmıştır. Döngünün çalışacağı adım sayısını 20000, algoritmanın çalışacağı yığın boyutu 16384 ve algoritmanın çalışacağı öğrenme hızı  $3*1e-2$  olarak belirlenmiştir.

Tablo 16. Eş Zamanlılık (Synchronization) alanının optimizasyon değerleri

**Adım\_Sayısı= 20000**

**Yığın\_Boyutu=16384**

**Öğrenme\_hızı=  $3*1e-2$**

Model eğitimi sırasında verilerin tutulabilmesi için yer tutucu ve değişkenlerden yararlanılmıştır. Tensorflow Makine Öğrenmesi Kütüphanesi yardımıyla model eğitimi öncesi eğitim doğrulama ve test veri seti için değişkenler, yertutucular (placeholder) tanımlanmıştır. Değişken kısmında Bias (ağırlık) için özellik ve etiket sayısını, Weight(Önyargı) için etiket sayısı değişkenleri tanımlanmıştır. Sonrasında “Tablo 8.”’e göre eğitim hesaplaması, geçerlilik eğitim hesaplaması ve kayıp hesaplaması tanımlanmıştır. Bir sonraki aşamada Öğrenme Hızına göre “Tablo 16.” daki optimizasyon ayarları belirtilmiştir. Son aşamada bulunan “Tablo 9.” da gösterilen eğitim, doğrulama ve test tahminleri yertutucuları tanımlanmıştır.

“Tablo 10.” da belirtilmiş olan Doğruluk puanı hesaplaması gösterilmiştir; doğruluk puanı, doğru tahmin edilen değerlerin sayısının tahmin edilen değerlerin sayısına bölünmesiyle hesaplanmaktadır.

Model eğitime belirtilen etiket ve özelliğe göre başlatılmıştır. Minimum yığınlara ayrılmış, yığınlar %500 de bir yenilenirken çıktı olarak ekrana aşama başına Yığın Kaybı, minimum Yığın Doğrulama ve Doğrulama (Validation) Doğruluğu yazdırılmıştır. Yığınlar bittiğinde test doğruluğu ve Eş Zamanlılık (Synchronization) tahmin doğru yazılmıştır. Bu işlemler sonlanırken ileride çevrimiçi değerlendirme aracına dönüştürebilmek için Eş Zamanlılık klasörüne Önyargı (Bias) ve Ağırlık (Weight) verileri csv dosyası olarak kaydedilmiştir. Yapılan bu işlemlerin kodları EK-I'ye eklenmiştir.

Model eğitimi çıktıları “Tablo 17.” de gösterilmiştir. Sonuçlara göre Proje veri tabanındaki Blok Tipi alanı ile Meta veri tabanındaki Eş Zamanlılık (Synchronization) alanının tahmin doğruluğu % 94.9 olarak gözlemlenmiştir.

Tablo 17. Eş Zamanlılık (Synchronization) alanının model eğitimi çıktıları

Başladı
Asamada_yığın_kaybı0: 18.583171844482422
Yığın_dogrulugu:26.2%
Gecerlilik_dogrulugu:27.5%
Asamada_yığın_kaybı500: 2.542501449584961
Yığın_dogrulugu:62.2%
Gecerlilik_dogrulugu:64.6%
Asamada_yığın_kaybı1000: 1.761002540588379

Yığın\_dogrulugu:74.4%

Gecerlilik\_dogrulugu:73.3%

Asamada\_yığın\_kaybı1500: 1.000593662261963

Yığın\_dogrulugu:78.0%

Gecerlilik\_dogrulugu:78.8%

Asamada\_yığın\_kaybı2000: 1.1290943622589111

Yığın\_dogrulugu:78.9%

Gecerlilik\_dogrulugu:80.2%

Asamada\_yığın\_kaybı2500: 0.576217532157898

Yığın\_dogrulugu:88.4%

Gecerlilik\_dogrulugu:83.5%

Asamada\_yığın\_kaybı3000: 0.5618574023246765

Yığın\_dogrulugu:88.4%

Gecerlilik\_dogrulugu:85.3%

Asamada\_yığın\_kaybı3500: 0.5538959503173828

Yığın\_dogrulugu:90.8%

Gecerlilik\_dogrulugu:85.3%

Asamada\_yığın\_kaybı4000: 0.41118323802948

Yığın\_dogrulugu:91.5%

Gecerlilik\_dogrulugu:87.6%

Asamada\_yığın\_kaybı4500: 0.3331937789916992

Yığın\_dogrulugu:88.4%

Gecerlilik\_dogrulugu:87.6%

Asamada\_yığın\_kaybı5000: 0.26994121074676514

Yığın\_dogrulugu:92.1%

Gecerlilik\_dogrulugu:89.0%

Asamada\_yığın\_kaybı5500: 0.3884431719779968

Yığın\_dogrulugu:89.0%

Gecerlilik\_dogrulugu:90.4%

Asamada\_yığın\_kaybı6000: 0.3346491754055023

Yığın\_dogrulugu:94.5%

Gecerlilik\_dogrulugu:92.7%

Asamada\_yığın\_kaybı6500: 0.2498801052570343

Yığın\_dogrulugu:94.5%

Gecerlilik\_dogrulugu:91.8%

Asamada\_yığın\_kaybı7000: 0.28970032930374146

Yığın\_dogrulugu:93.3%

Gecerlilik\_dogrulugu:93.6%

Asamada\_yığın\_kaybı7500: 0.6980068683624268

Yığın\_dogrulugu:89.6%

Gecerlilik\_dogrulugu:93.1%

Asamada\_yığın\_kaybı8000: 0.30090299248695374

Yığın\_dogrulugu:92.7%

Gecerlilik\_dogrulugu:93.6%

Asamada\_yığın\_kaybı8500: 0.28120386600494385

Yığın\_dogrulugu:93.3%

Gecerlilik\_dogrulugu:94.0%

Asamada\_yığın\_kaybı9000: 0.181264266371727

Yığın\_dogrulugu:95.7%  
 Gecerlilik\_dogrulugu:94.5%  
 Asamada\_yığın\_kaybı9500: 0.164870023727417  
 Yığın\_dogrulugu:96.9%  
 Gecerlilik\_dogrulugu:93.6%  
 Test-doğruluğu-Test-accuracy:94.9%  
 Test-doğruluğu-Test-accuracy:100.0%

#### 5.4.1.5. Akış Kontrolü (FlowControl) alanı model eğitilmesi

Model denemeleri sonucu model optimizasyonu “Tablo 18.” de belirtildiği üzere yapılmıştır. Döngünün çalışacağı adım sayısını 12000, algoritmanın çalışacağı yığın boyutu 16384 ve algoritmanın çalışacağı öğrenme hızı  $2*1e-2$  olarak belirlenmiştir.

Tablo 18. Akış Kontrolü (FlowControl) alanının optimizasyon değerleri

<b>Adım_Sayısı= 12000</b>
<b>Yığın_Boyutu=16384</b>
<b>Oğrenme_hızı= <math>2*1e-2</math></b>

Model eğitimi sırasında verilerin tutulabilmesi için yer tutucu ve değişkenlerden yararlanılmıştır. Tensorflow Makine Öğrenmesi Kütüphanesi yardımıyla model eğitimi öncesi eğitim doğrulama ve test veri seti için değişkenler, yertutucular (placeholder) tanımlanmıştır. Değişken kısmında Bias (ağırlık) için özellik ve etiket sayısını, Weight(Önyargı) için etiket sayısı değişkenleri tanımlanmıştır. Sonrasında “Tablo 8.” ’e göre eğitim hesaplaması, geçerlilik eğitim hesaplaması ve

kayıp hesaplaması tanımlanmıştır. Bir sonraki aşamada Öğrenme Hızına göre “Tablo 18.” deki optimizasyon ayarları belirtilmiş. Son aşamada bölüm 5.4.2.1. de bulunan “Tablo 9.” da gösterilen eğitim, doğrulama ve test tahminleri yertutucuları tanımlanmıştır.

“Tablo 10.” da belirtilmiş olan Doğruluk puanı hesaplaması gösterilmiştir; doğruluk puanı, doğru tahmin edilen değerlerin sayısının tahmin edilen değerlerin sayısına bölünmesiyle hesaplanmaktadır.

Model eğitime belirtilen etiket ve özelliğe göre başlatılmıştır. Minimum yığınlar ayrılmış, yığınlar %500 de bir yenilenirken çıktı olarak ekrana aşama başına Yığın Kaybı, minimum Yığın Doğrulama ve Doğrulama (Validation) Doğruluğu yazdırılmıştır. Yığınlar bittiğinde test doğruluğu ve Akış Kontrolü (FlowControl) tahmin doğru yazılmıştır. Bu işlemler sonlanırken ileride çevrimiçi değerlendirme aracına dönüştürebilmek için Akış Kontrolü klasörüne Önyargı (Bias) ve Ağırlık (Weight) verileri csv dosyası olarak kaydedilmiştir. Yapılan bu işlemlerin kodları EK-J’ye eklenmiştir.

Model eğitimi çıktıları “Tablo 19.” da gösterilmiştir. Sonuçlara göre Proje veri tabanındaki Blok Tipi alanı ile Meta veri tabanındaki Akış Kontrolü (FlowControl) alanının tahmin doğruluğu % 92.3 olarak gözlemlenmiştir.

Tablo 19. Akış Kontrolü (FlowControl) alanının model eğitimi çıktıları

Başladı
Asamada_yığın_kaybı0: 35.657928466796875
Yığın_dogrulugu:27.9%
Gecerlilik_dogrulugu:32.1%



Asamada\_yığın\_kaybı500: 3.0182137489318848

Yığın\_dogrulugu:66.5%

Gecerlilik\_dogrulugu:63.2%

Asamada\_yığın\_kaybı1000: 1.3050787448883057

Yığın\_dogrulugu:70.7%

Gecerlilik\_dogrulugu:71.9%

Asamada\_yığın\_kaybı1500: 1.438827633857727

Yığın\_dogrulugu:75.0%

Gecerlilik\_dogrulugu:75.3%

Asamada\_yığın\_kaybı2000: 1.3762028217315674

Yığın\_dogrulugu:80.5%

Gecerlilik\_dogrulugu:80.3%

Asamada\_yığın\_kaybı2500: 0.678805947303772

Yığın\_dogrulugu:84.7%

Gecerlilik\_dogrulugu:83.5%

Asamada\_yığın\_kaybı3000: 0.45856887102127075

Yığın\_dogrulugu:88.2%

Gecerlilik\_dogrulugu:86.7%

Asamada\_yığın\_kaybı3500: 0.6045714020729065

Yığın\_dogrulugu:84.1%

Gecerlilik\_dogrulugu:87.2%

Asamada\_yığın\_kaybı4000: 0.4494432806968689

Yığın\_dogrulugu:86.6%

Gecerlilik\_dogrulugu:88.1%

Asamada\_yığın\_kaybı4500: 0.35665979981422424

Yığın\_dogrulugu:89.4%

Gecerlilik\_dogrulugu:88.1%

Asamada\_yığın\_kaybı5000: 0.39183172583580017

Yığın\_dogrulugu:92.1%

Gecerlilik\_dogrulugu:87.6%

Asamada\_yığın\_kaybı5500: 0.26545295119285583

Yığın\_dogrulugu:92.5%

Gecerlilik\_dogrulugu:88.1%

Asamada\_yığın\_kaybı6000: 0.28223326802253723

Yığın\_dogrulugu:92.1%

Gecerlilik\_dogrulugu:89.5%

Asamada\_yığın\_kaybı6500: 0.22155803442001343

Yığın\_dogrulugu:95.7%

Gecerlilik\_dogrulugu:90.4%

Asamada\_yığın\_kaybı7000: 0.2668845057487488

Yığın\_dogrulugu:94.5%

Gecerlilik\_dogrulugu:89.0%

Asamada\_yığın\_kaybı7500: 0.2825036644935608

Yığın\_dogrulugu:90.5%

Gecerlilik\_dogrulugu:90.4%

Asamada\_yığın\_kaybı8000: 0.28922542929649353

Yığın\_dogrulugu:93.9%

Gecerlilik\_dogrulugu:91.3%

Asamada_yığın_kaybı8500: 0.20011310279369354
Yığın_dogrulugu:97.6%
Gecerlilik_dogrulugu:91.3%
Asamada_yığın_kaybı9000: 0.21669641137123108
Yığın_dogrulugu:92.1%
Gecerlilik_dogrulugu:91.3%
Asamada_yığın_kaybı9500: 0.16337016224861145
Yığın_dogrulugu:95.1%
Gecerlilik_dogrulugu:91.8%
Test-doğruluğu-Test-accuracy:92.3%
Test-doğruluğu-Test-accuracy:100.0%

#### 5.4.1.6.Kullanıcı Etkileşimi (UserInteractivity) alanı model eğitilmesi

Model denemeleri sonucu model optimizasyonu “Tablo 20.” da belirtildiği üzere yapılmıştır. Döngünün çalışacağı adım sayısını 10000, algoritmanın çalışacağı yığın boyutu 16384 ve algoritmanın çalışacağı öğrenme hızı  $2*1e-2$  olarak belirlenmiştir.

Tablo 20. Kullanıcı Etkileşimi (UserInteractivity) alanının optimizasyon değerleri

<b>Adım_Sayısı= 10000</b>
<b>Yığın_Boyutu=16384</b>
<b>Oğrenme_hızı= <math>2*1e-2</math></b>

Model eğitimi sırasında verilerin tutulabilmesi için yer tutucu ve değişkenlerden yararlanılmıştır. Tensorflow Makine Öğrenmesi Kütüphanesi yardımıyla model eğitimi öncesi eğitim doğrulama ve test veri seti için değişkenler, yertutucular (placeholder) tanımlanmıştır. Değişken kısmında Bias (ağırlık) için özellik ve etiket sayısını, Weight(Önyargı) için etiket sayısı değişkenleri tanımlanmıştır. Sonrasında “Tablo 8.”’e göre eğitim hesaplaması, geçerlilik eğitim hesaplaması ve kayıp hesaplaması tanımlanmıştır. Bir sonraki aşamada Öğrenme Hızına göre “Tablo 20.”deki optimizasyon ayarları belirtilmiş. Son aşamada “Tablo 9.”da gösterilen eğitim, doğrulama ve test tahminleri yertutucuları tanımlanmıştır.

“Tablo 10.”da belirtilmiş olan Doğruluk puanı hesaplaması gösterilmiştir; doğruluk puanı, doğru tahmin edilen değerlerin sayısının tahmin edilen değerlerin sayısına bölünmesiyle hesaplanmaktadır.

Model eğitime belirtilen etiket ve özelliğe göre başlatılmıştır. Minimum yığınlara ayrılmış, yığınlar %500 de bir yenilenirken çıktı olarak ekrana aşama başına Yığın Kaybı, minimum Yığın Doğrulama ve Doğrulama (Validation) Doğruluğu yazdırılmıştır. Yığınlar bittiğinde test doğruluğu ve Kullanıcı Etkileşimi (UserInteractivity) tahmin doğru yazılmıştır. Bu işlemler sonlanırken ileride çevrimiçi değerlendirme aracına dönüştürebilmek için Kullanıcı Etkileşimi klasörüne Önyargı (Bias) ve Ağırlık (Weight) verileri csv dosyası olarak kaydedilmiştir. Yapılan bu işlemlerin kodları EK-K’ye eklenmiştir.

Model eğitimi çıktıları “Tablo 21.”de gösterilmiştir. Sonuçlara göre Proje veri tabanındaki Blok Tipi alanı ile Meta veri tabanındaki Kullanıcı Etkileşimi (UserInteractivity) alanının tahmin doğruluğu % 92.3 olarak gözlemlenmiştir.

Tablo 21. Kullanıcı Etkileşimi (UserInteractivity) alanının model eğitimi çıktıları

Başladı
Asamada_yığın_kaybı0: 27.65372085571289
Yığın_dogrulugu:31.8%
Gecerlilik_dogrulugu:27.0%
Asamada_yığın_kaybı500: 3.8299779891967773
Yığın_dogrulugu:63.4%
Gecerlilik_dogrulugu:61.5%
Asamada_yığın_kaybı1000: 1.4377601146697998
Yığın_dogrulugu:72.1%
Gecerlilik_dogrulugu:71.6%
Asamada_yığın_kaybı1500: 1.1244620084762573
Yığın_dogrulugu:81.1%
Gecerlilik_dogrulugu:79.8%
Asamada_yığın_kaybı2000: 1.2582838535308838
Yığın_dogrulugu:76.6%
Gecerlilik_dogrulugu:85.3%
Asamada_yığın_kaybı2500: 0.8558763265609741
Yığın_dogrulugu:81.1%
Gecerlilik_dogrulugu:87.6%
Asamada_yığın_kaybı3000: 0.8637853860855103
Yığın_dogrulugu:82.9%
Gecerlilik_dogrulugu:86.7%
Asamada_yığın_kaybı3500: 0.6530756950378418

Yığın\_dogrulugu:84.7%

Gecerlilik\_dogrulugu:89.0%

Asamada\_yığın\_kaybı4000: 0.7955299615859985

Yığın\_dogrulugu:82.4%

Gecerlilik\_dogrulugu:89.5%

Asamada\_yığın\_kaybı4500: 0.3803594410419464

Yığın\_dogrulugu:89.0%

Gecerlilik\_dogrulugu:89.9%

Asamada\_yığın\_kaybı5000: 0.27404001355171204

Yığın\_dogrulugu:92.1%

Gecerlilik\_dogrulugu:89.5%

Asamada\_yığın\_kaybı5500: 0.5251133441925049

Yığın\_dogrulugu:86.6%

Gecerlilik\_dogrulugu:89.9%

Asamada\_yığın\_kaybı6000: 0.25509393215179443

Yığın\_dogrulugu:92.7%

Gecerlilik\_dogrulugu:89.9%

Asamada\_yığın\_kaybı6500: 0.25923195481300354

Yığın\_dogrulugu:91.5%

Gecerlilik\_dogrulugu:90.4%

Asamada\_yığın\_kaybı7000: 0.2862686514854431

Yığın\_dogrulugu:92.1%

Gecerlilik\_dogrulugu:90.4%

Asamada\_yığın\_kaybı7500: 0.21568141877651215

Yığın\_dogrulugu:93.3%  
 Gecerlilik\_dogrulugu:90.8%  
 Asamada\_yığın\_kaybı8000: 0.4104258418083191  
 Yığın\_dogrulugu:90.2%  
 Gecerlilik\_dogrulugu:91.3%  
 Asamada\_yığın\_kaybı8500: 0.3585742115974426  
 Yığın\_dogrulugu:93.9%  
 Gecerlilik\_dogrulugu:91.3%  
 Asamada\_yığın\_kaybı9000: 0.24528570473194122  
 Yığın\_dogrulugu:93.9%  
 Gecerlilik\_dogrulugu:91.8%  
 Asamada\_yığın\_kaybı9500: 0.33058595657348633  
 Yığın\_dogrulugu:94.5%  
 Gecerlilik\_dogrulugu:90.8%  
 Test-doğruluğu-Test-accuracy:92.3%  
 Test-doğruluğu-Test-accuracy:100.0%

#### 5.4.1.7. Veri Gösterimi (DataRepresentation) alanı model eğitilmesi

Model denemeleri sonucu model optimizasyonu “Tablo 22.” da belirtildiği üzere yapılmıştır. Döngünün çalışacağı adım sayısını 10000, algoritmanın çalışacağı yığın boyutu 16384 ve algoritmanın çalışacağı öğrenme hızı  $1e-2$  olarak belirlenmiştir.

Tablo 22. Veri Gösterimi (DataRepresentation) alanının optimizasyon değerleri

<b>Adım_Sayısı= 10000</b>
<b>Yığın_Boyutu=16384</b>
<b>Öğrenme_hızı= 1e-2</b>

Model eğitimi sırasında verilerin tutulabilmesi için yer tutucu ve değişkenlerden yararlanılmıştır. Tensorflow Makine Öğrenmesi Kütüphanesi yardımıyla model eğitimi öncesi eğitim doğrulama ve test veri seti için değişkenler, yertutucular (placeholder) tanımlanmıştır. Değişken kısmında Bias (ağırlık) için özellik ve etiket sayısını, Weight(Önyargı) için etiket sayısı değişkenleri tanımlanmıştır. Sonrasında “Tablo 8.” ’e göre eğitim hesaplaması, geçerlilik eğitim hesaplaması ve kayıp hesaplaması tanımlanmıştır. Bir sonraki aşamada Öğrenme Hızına göre “Tablo 22.” deki optimizasyon ayarları belirtilmiştir. Son aşamada bulunan “Tablo 9.” da gösterilen eğitim, doğrulama ve test tahminleri yertutucuları tanımlanmıştır.

“Tablo 10.” da belirtilmiş olan Doğruluk puanı hesaplaması gösterilmiştir; doğruluk puanı, doğru tahmin edilen değerlerin sayısının tahmin edilen değerlerin sayısına bölünmesiyle hesaplanmaktadır.

Model eğitime belirtilen etiket ve özelliğe göre başlatılmıştır. Minimum yığınlar ayrılmış, yığınlar %500 de bir yenilenirken çıktı olarak ekrana aşama başına Yığın Kaybı, minimum Yığın Doğrulama ve Doğrulama (Validation) Doğruluğu yazdırılmıştır. Yığınlar bittiğinde test doğruluğu ve Veri Gösterimi (DataRepresentation) tahmin doğruluğu yazılmıştır. Bu işlemler sonlanırken ileride çevrimiçi değerlendirme aracına dönüşürebilmek için Veri Gösterimi klasörüne



Önyargı (Bias) ve Ağırlık (Weight) verileri csv dosyası olarak kaydedilmiştir. Yapılan bu işlemlerin kodları EK-L`ye eklenmiştir.

Model eğitimi çıktıları “Tablo 23.” de gösterilmiştir. Sonuçlara göre Proje veri tabanındaki Blok Tipi alanı ile Meta veri tabanındaki Kullanıcı Etkileşimi (UserInteractivity) alanının tahmin doğruluğu % 93.8 olarak gözlemlenmiştir.

Tablo 23. Veri Gösterimi (DataRepresentation) alanının model eğitimi çıktıları

Başladı
Asamada_yığın_kaybı0: 105.73514556884766
Yığın_dogrulugu:14.6%
Gecerlilik_dogrulugu:18.3%
Asamada_yığın_kaybı500: 3.043555498123169
Yığın_dogrulugu:56.7%
Gecerlilik_dogrulugu:56.8%
Asamada_yığın_kaybı1000: 1.9434340000152588
Yığın_dogrulugu:72.0%
Gecerlilik_dogrulugu:69.3%
Asamada_yığın_kaybı1500: 1.3574724197387695
Yığın_dogrulugu:73.7%
Gecerlilik_dogrulugu:76.2%
Asamada_yığın_kaybı2000: 0.8799259662628174
Yığın_dogrulugu:79.2%
Gecerlilik_dogrulugu:77.6%
Asamada_yığın_kaybı2500: 0.5822731852531433

Yığın\_dogrulugu:82.3%

Gecerlilik\_dogrulugu:78.3%

Asamada\_yığın\_kaybı3000: 0.9774267077445984

Yığın\_dogrulugu:81.7%

Gecerlilik\_dogrulugu:81.7%

Asamada\_yığın\_kaybı3500: 0.4641367495059967

Yığın\_dogrulugu:84.8%

Gecerlilik\_dogrulugu:84.3%

Asamada\_yığın\_kaybı4000: 0.6274596452713013

Yığın\_dogrulugu:83.7%

Gecerlilik\_dogrulugu:86.1%

Asamada\_yığın\_kaybı4500: 0.5021786093711853

Yığın\_dogrulugu:82.9%

Gecerlilik\_dogrulugu:87.6%

Asamada\_yığın\_kaybı5000: 0.4350186586380005

Yığın\_dogrulugu:89.6%

Gecerlilik\_dogrulugu:87.6%

Asamada\_yığın\_kaybı5500: 0.42268243432044983

Yığın\_dogrulugu:90.8%

Gecerlilik\_dogrulugu:88.5%

Asamada\_yığın\_kaybı6000: 0.41186219453811646

Yığın\_dogrulugu:90.2%

Gecerlilik\_dogrulugu:89.5%

Asamada\_yığın\_kaybı6500: 0.3134828209877014

Yığın\_dogrulugu:92.1%

Gecerlilik\_dogrulugu:89.9%

Asamada\_yigin\_kaybi7000: 0.3355640172958374

Yığın\_dogrulugu:87.9%

Gecerlilik\_dogrulugu:90.4%

Asamada\_yigin\_kaybi7500: 0.24365679919719696

Yığın\_dogrulugu:93.8%

Gecerlilik\_dogrulugu:92.7%

Asamada\_yigin\_kaybi8000: 0.29001110792160034

Yığın\_dogrulugu:93.9%

Gecerlilik\_dogrulugu:92.2%

Asamada\_yigin\_kaybi8500: 0.20210762321949005

Yığın\_dogrulugu:93.9%

Gecerlilik\_dogrulugu:92.2%

Asamada\_yigin\_kaybi9000: 0.25243234634399414

Yığın\_dogrulugu:92.1%

Gecerlilik\_dogrulugu:93.1%

Asamada\_yigin\_kaybi9500: 0.3671313524246216

Yığın\_dogrulugu:93.3%

Gecerlilik\_dogrulugu:93.1%

Test-doğruluğu-Test-accuracy:93.8%

Test-doğruluğu-Test-accuracy:100.0%

### 5.4.2. Uygun Modelin Değerlendirilmesi

Uygun model bulunurken öncelikle Derin Sınır Ağları Modeli eğitilmiştir. Eğitim sonucunda Sınıflama doğruluğu %56.12 olarak gözlemlenmiştir. Bu sonuçlar veri setinin model eğitimi için sınıflama algoritmalarının uygun olmadığını göstermektedir.

Uygun model bulunurken ikinci aşamada Basit Doğrusal Regresyon modeli denenmiştir. Model eğitilirken BİD becerilerinin 7 boyutunu ifade eden meta veri tabanındaki alanlar için tek tek model denemesi yapılmıştır. “Tablo 24.” de 7 boyutun model değerlendirme sonuçları listelenmiştir.

Tablo 24. Basit Doğrusal Regresyon Modeli sonuçları

<b>Basit Doğrusal regresyon modelinin BİD becerilerinin 7 boyutu için tahmin değerleri</b>						
Soyutlama ve Problem Ayrıştırma (Abstraction)	Paralellik (Parallelism)	Mantıksal Düşünme (Logic)	Eş Zamanlılık (Synchronization)	Akış Kontrolü (FlowControl)	Kullanıcı Etkileşimi (UserInteractivity)	Veri Gösterimi (DataRepresentation)
92.3%	92.3%	94.5%	94.9%	92.3%	92.3%	93.8%

Model sonuçlar incelendiğinde tüm sonuçların büyük ölçüde Blok tipi alanına bağlı olduğu görülmektedir. Doğruluk oranları 92% ile 95% arasında izlenmektedir. Doğruluk oranlarının yüksek olması makinenin kuralları öğrendiğini göstermektedir. İlk bakışta bu veri seti için kural tabanlı klasik koşullar kullanılabileceği düşünülse de, klasik programlama algoritmasıyla 139 kategorik Blok tipi verisinin sınırlı birkaç koşula bağlı kalarak değerlendirilmesi yerine, Makine Öğrenmesiyle çeşitli

kombinasyonları öğrenerek yeni koşullarda beceriyi detaylı analiz edilmesi sağlanmıştır.

## 5.5.EĞİTİLEN MODELİ ÇEVİRİMİÇİ DEĞERLENDİRME ARACINA

### DÖNÜŞTÜRME

Eğitilen model öğretmen ve öğrenciler tarafından değerlendirme işlemi için rahatlıkla kullanabilecekleri bir web ortamına dönüştürülmüştür. Bu işlem için Tensorflow'un web tabanlı model çalışmaları için sunduğu Tensorflowjs kütüphanesinden faydalanarak, Javascript ve Html den oluşan temel çevrimiçi değerlendirme aracı gerçekleştirilmiştir.

Model eğitirken kullandığımız blok tipinin 139 kategorik değeri için oluşturulan EK-B deki Blok tipi çeviri sözlüğü web ortamında model tahmininde kullanılabilmesi için çeviri sözlüğü csv dosyası olarak kaydedilmiştir.

Meta veri setindeki BİD becerilerinin 7 boyutu; Soyutlama ve Problem Ayırıştırma (Abstraction), Paralellik (Parallelism), Mantıksal Düşünme (Logic), Eş Zamanlılık (Synchronization), Akış Kontrolü (FlowControl), Kullanıcı Etkileşimi (UserInteractivity) ve Veri Gösterimi (DataRepresentation) her bir alanı için model eğitilmiştir. Bu modeller eğitilirken çevrimiçi değerlendirme aracına hazırlık olması için Önyargı (Bias) , Ağırlık (Weight) ve model tensorflow kütüphanesiyle 7 boyutun her biri kendi adında klasörlere kaydedilmiştir. “Tablo 28.” modelin kaydedilmesi gösterilmiştir.

Tablo 28. Eğitilen Modeli kaydedilmesi

```
tf.saved_model.simple_save(session, "./abstraction", inputs = {'batch_data':
tf_abstraction_dataset, 'tf_abstraction_weights': weights, 'tf_abstraction_biases':
biases}, outputs = {'tf_abstraction_prediction': tf_abstraction_prediction}))
```

Kaydedilen modeli web ortamına yerleştirebilmek için model Tensorflowjs formatına çevrilmiştir. “Tablo 25.” de modelin Tensorflowjs formatına çevirme aşaması belirtilmiştir. Bu işlem sonrasında model tarayıcıda kullanılacak formata getirilmiştir.

Tablo 25. Model Çeviri Aşaması

```
!rm -rf "./models/Abstraction"
!tensorflowjs-converter
--input-format=tf-saved-model--output-node-names='tf-Abstraction-prediction,tf-
Abstraction-weights,tf-Abstraction-biases,tf-Abstraction-dataset' --saved-model-
tags=serve./Abstraction./models/Abstraction
```

Çevrimiçi değerlendirme aracı iki ana dizinden oluşmaktadır. Birincisi değerlendirilecek veri seti, çevir sözlüğü ve model eğitilirken kaydedilen modelleri olduğu veri dizininden oluşmaktadır. İkincisi temel html, javascript ve tensorflowjs bilgilerinin olduğu tfjs dizininden oluşmaktadır. Modelin çevrimiçi ortama gömülmesi “Tablo 26.” da gösterilmiştir.

Tablo 26. Eğitilen modelin Javascript ortamında kullanımı

```
const allWeights = await this.loadWeights(score);
const weights = allWeights["tf-"+score+"-weights"];
const biases = allWeights["tf-"+score+"-biases"];;
```

```

var prediction = tf.tensor2d(Object.values(clonedRow), [1,
139]).matMul(weights).add(biases);

result.push({"p-ID": p-ID, answer: tf.argmax(prediction, 1).dataSync()});

```

## 5.6. UYGULMANIN TANITIMI

Bu değerlendirme aracı blok tabanlı programlama dillerinde oluşturulan programların BİD Becerilerine göre değerlendirilmesi amacıyla geliştirilmiştir. Araç Scratch blok tabanlı programlama dilinde geliştirilmiş programları Makine Öğrenmesi modellerinden faydalanarak kullanıcıya BİD becerileri hakkında geribildirim vermektedir.

Değerlendirme aracı dört web sayfasından oluşmaktadır. Anasayfa “Şekil 16.” da gösterilmiştir. Bu sayfada aracın hangi amaçla geliştirildiği ve sırayla takip edilmesi gereken sayfaları göstermektedir.



Şekil 16. Çevrimiçi değerlendirme aracı ekranı

Ziyaret edilecek İlk sayfa ‘BİD Becerileri nedir?’ sayfasıdır. Sayfada 7 beceri türünün kısa tanımı ve her bir becerinin 0 ile 3 puan olması için gerekli kriterler hakkında detaylı bilgiler verilmektedir, bu sayfanın ekran görüntüsü “Şekil 17.” mevcuttur.

BİLGİ İŞLEMSEL DÜŞÜNME BECERİLERİNİ DEĞERLENDİRME ARACI	
Bilgi işlemsel düşünme becerilerinin 7 boyutunun her bir alanı 3 puan üzerinden değerlendirilmektedir. 0 puan o becerinin hiç kullanılmadığı ya da ölçülecek dercede detay içermediğini ifade eder. 1 puan o becerinin giriş seviyesinde olduğunu, 2 puan o becerinin gelişme seviyesinde olduğunu, 3 puan o becerinin yeteri seviyesinde olduğunu belirtmektedir.	
<a href="#">Ana Sayfaya Dön</a>	
BİLİŞSEL DÜŞÜNME BECERİLERİ VE SEVİYESİ	
Beceri	Açıklama
Soyutlama ve Problem Ayırıştırma	Soyutlama ve problem ayırıştırma bir problemin daha kolay anlaşılması, programlanması ve hata ayıklaması için daha küçük parçalara bölünmesidir. Algoritmik olarak Fonksiyon gibi kodu parçalara bölme ve tekrar kullanabilme becerisidir. Sonuçları 1 puan olması, karakterleri sırtanışmasını parçalar halinde özetek hareket eden kodun yazıldığını ifade etmektedir. Sonuçları 2 puan olması, kod bloklarını benzerlikler olarak tanımlayanlar her değişikliği farklı değişkenler kod parçasını kullanabildiğini ifade etmektedir. Sonuçları 3 puan olması, aynı işlemi sırtanış birde, farklı karakterler özetlerdir, sırtanışlık ile kod yazarak yerine kopuzunu oluşturabildiğini ifade etmektedir.
Maniksal Düşünme	Maniksal düşünme programın dinamik olmasını, duruma göre çalışacak kodlarda farklılık olmasını sağlar becerisidir. Kontrol bloklarından faydalanılmaktadır. Sonuçları 1 puan olması, aynı işlemi sırtanış birde, farklı karakterler özetlerdir, sırtanışlık ile kod yazarak yerine kopuzunu oluşturabildiğini ifade etmektedir. Sonuçları 2 puan olması, aynı işlemi sırtanış birde, farklı karakterler özetlerdir, sırtanışlık ile kod yazarak yerine kopuzunu oluşturabildiğini ifade etmektedir. Sonuçları 3 puan olması, aynı işlemi sırtanış birde, farklı karakterler özetlerdir, sırtanışlık ile kod yazarak yerine kopuzunu oluşturabildiğini ifade etmektedir.
Paralellik	Paralellik becerisi birçok işlemi aynı anda gerçekleştirme olasıdır. Örneğin ekranda bulunan karakterler olduğunda kontrol blokları da başlatılabilmektedir. Olaylar bloklarında bulunan komutlar doğrudan bu alana ebilmektedir. Sonuçları 1 puan olması, kullanıcıların herhangi bir hata basması veya neşneye çıkması ile gerçekleştirilen kod bloklarıdır. Olay bloklarından hata basılınca veya kukya beklenince işlemlerin etkilerini olarak başlatılabilmektedir. Sonuçları 2 puan olması, kullanıcıların herhangi bir hata basması veya neşneye çıkması ile gerçekleştirilen kod bloklarıdır. Olay bloklarından hata basılınca veya kukya beklenince işlemlerin etkilerini olarak başlatılabilmektedir. Sonuçları 3 puan olması, kullanıcıların herhangi bir hata basması veya neşneye çıkması ile gerçekleştirilen kod bloklarıdır. Olay bloklarından hata basılınca veya kukya beklenince işlemlerin etkilerini olarak başlatılabilmektedir.
Akrig Kontrolü	Makro kontrolü, algoritmada birkaç kez tekrarlanan ifadelerin tekrarı önlemek için kullanılan kod bloklarıdır. Kontrol bloklarından faydalanılır. Sonuçları 1 puan olması, be dizi tekrar eden işlemi aynı arkaya sürmesini becerisinin gerçekleştirildiğini göstermektedir. Sonuçları 2 puan olması, aynı kodun birde farklı tekrar olmasını istediğinde kontrol bloklarından sürekli tekrarla ya da kadar tekrar et gibi kod satırındaki tekrarı özetek ifadeleri kullanıldığını ifade etmektedir. Sonuçları 3 puan olması, tekrar eden işlemi ve duruma göre tekrar eden işlemi, Örneğin aynı kadar tekrar et kod bloklarını kullanıldığını durumunu ifade eder.
Kullanıcı Etkileşimi	Kullanıcı etkileşimi, programın daha etkiliğini olmasını sağlar. Bunun için klavye ve farenin kullanılmasıyla klavye hareketlendirmeyi sağlar. Sonuçları 1 puan olması, klavye komutu gibi programın başlatılmasını sağlayan komutların kullanıldığını ifade etmektedir. Sonuçları 2 puan olması, klavye komutu gibi programın başlatılmasını sağlayan komutların kullanıldığını ifade etmektedir. Sonuçları 3 puan olması, klavye komutu gibi programın başlatılmasını sağlayan komutların kullanıldığını ifade etmektedir.
Veri Gösterimi	Veri gösterimi, her karakterin konumu, işaret ettiği yönü, büyüklüğü gibi birçok bilginin belirtildiği yerdir. Hareket, görünüm ve veri bloklarının kullanıldığı, algoritmik olarak değişken ve liste kullanmayı ifade eden beceri türüdür. Sonuçları 1 puan olması, kullanıcıların programın başlatılmasını sağlayan komutların kullanıldığını ifade etmektedir. Sonuçları 2 puan olması, kullanıcıların programın başlatılmasını sağlayan komutların kullanıldığını ifade etmektedir. Sonuçları 3 puan olması, kullanıcıların programın başlatılmasını sağlayan komutların kullanıldığını ifade etmektedir.
Eş Zamanlılık	Senkronizasyon (Eş zamanlılık) karakterlerin işlenildiği sırada yazmasını sağlayan beceridir. Kontrol yapılarındaki bazı özellikler ile doğrudan ilişkilidir. Sonuçları 1 puan olması, kullanıcıların programın başlatılmasını sağlayan komutların kullanıldığını ifade etmektedir. Sonuçları 2 puan olması, kullanıcıların programın başlatılmasını sağlayan komutların kullanıldığını ifade etmektedir. Sonuçları 3 puan olması, kullanıcıların programın başlatılmasını sağlayan komutların kullanıldığını ifade etmektedir.

Şekil 17. BİD becerileri değerlendirme kriterleri sayfası

Kullanıcılar için tanımlanan detaylı BİD Becerileri tablosu, “Tablo 29.” da detaylı olarak verilmiştir.



Tablo 29. Bilgi İşlemsel Düşünme Becerileri Değerlendirme kriterleri

<b>BİLİŞSEL DÜŞÜNME BECERİLERİ VE SEVİYESİ</b>	
Soyutlama ve Problem Ayırıştırma	Soyutlama ve problem ayırıştırma, bir problemin daha kolay anlaşılması, programlanması ve hata ayıklaması için daha küçük parçalara bölünmesidir. Algoritmik olarak Fonksiyon gibi kodu parçalara bölme ve tekrar kullanabilme becerisidir.
	Sonucunun 1 puan olması, karakterin davranışlarını parçalar halinde bölerek hareket ettiren kodun yazıldığını ifade etmektedir.
	Sonucunun 2 puan olması, kod bloklarını fonksiyon olarak tanımlayarak her çağırışta farklı değişkenle kod parçasını kullanabildiğini ifade etmektedir.
	Sonucunun 3 puan olması, aynı işlemi yapan birden fazla olduğunda, işlemleri tek tek yapmak yerine kopyasının oluşturulabildiğini ifade etmektedir.
Mantıksal Düşünme	Mantıksal düşünme programın dinamik olmasını, duruma göre çalışacak kodlarda farklılık olmasını sağlama becerisidir. Kontrol bloklarından faydalanılmaktadır
	Sonucunun 1 puan olması, eğer şartı sağlıyorsa şunu yap komutu olan if kullanımı gibi temel şart ifadelerini içeren blokların kullanıldığını ifade eder.
	Sonucunun 2 puan olması, eğer şartının yanısıra değilse ifadesini de içeren koşulun sağlanması ya da sağlanmaması durumunda iki ayrı ifade bulunan kod bloklarıdır.
	Sonucunun 3 puan olması, aynı anda birden fazla koşul kullanımı ve Mantıksal Operatörlerden AND, OR ve Not içermesidir.
Paralellik	Paralellik becerisi birçok şeyin aynı anda gerçekleşme olasılığıdır. Örneğin ekranda birden fazla karakter olduğunda ikisinin de başka şeyler yapabilmesidir. Olaylar bloklarında bulunun komutlar doğrudan bu alanı etkilemektedir.
	Sonucunun 1 puan olması, karakterin aynı anda birden fazla kodu aynı anda başlatması ya da başka kuklaları etkilemesi gibi özelliklerdir. Tıklanınca komutuyla ilişkilidir
	Sonucunun 2 puan olması, kullanıcının herhangi bir tuşa basması veya nesneye tıklaması ile gerçekleşen kod bloklarıdır. Olay bloklarından tuşa basılınca veya kuklaya tıklanınca işlemlerin etkileşimli olarak başlatılabilmesidir.
	Sonucunun 3 puan olması, birden fazla olayın aynı anda başlayıp, paralel şekilde işlemler yapması ve bağlantılı sonuçlar çıkarabilmesini ifade etmektedir.

Akış Kontrolü	<p>Akış kontrolü, algoritmada birkaç kez tekrarlanan ifadelerin tekrarını önlemek için kullanılan kod bloklarıdır. Kontrol bloklarından faydalanılır.</p> <p>Sonucun 1 puan olması, bir dizi tekrar eden işlemin arka arkaya sırlanması becerisinin gerçekleştirildiğini göstermektedir..</p> <p>Sonucun 2 puan olması, aynı kodun birden fazla tekrar etmesini istediğimizde kontrol bloklarından sürekli tekrarlar ya da kadar tekrar et gibi kod satırındaki tekrarı önleyen ifadelerin kullanıldığını ifade etmektedir.</p> <p>Sonucun 3 puan olması, tekrar etme sayısı ve durumu bilinmediğinde koşula göre tekrar etme durumudur. Örneğin olana kadar tekrar et kod bloğunun kullanılabilirdiği durumu ifade eder.</p>
Kullanıcı Etkileşimi	<p>Kullanıcı etkileşimi, programın daha etkileşimli olmasını sağlamaktadır. Bunun için klavye ve farenin kullanılmasıyla kuklayı hareketlendirmeyi sağlamaktadır.</p> <p>Sonucunun 1 puan olması, tıklanınca komutu gibi programın başlatılmasını sağlayan komutların kullanıldığını ifade etmektedir.</p> <p>Sonucunun 2 puan olması, klavyeden tuşlara basıldığında kodları başlatmak, fare okunun hareketine göre kodu ayarlamak gibi interaktif kodların kullanıldığını ifade etmektedir.</p> <p>Sonucunun 3 puan olması, kullanıcının hareketi doğrultusunda video ses gibi araçların çalıştırılması ile kullanıcı ile program arasında etkileşimin sağlandığı becerisini göstermektedir.</p>
Veri Gösterimi	<p>Veri gösterimi, her karakterin konumu, işaret ettiği yönü, büyüklüğü gibi birçok bilginin belirtildiği yerdir. Hareket, görünüm ve veri bloklarının kullanıldığı, algoritmik olarak değişken ve liste kullanmayı ifade eden beceri türüdür.</p> <p>Sonucunun 1 puan olması, kuklanın pozisyonu, boyutu, hareketi ve görünüşü gibi bilgilerin verildiğini ifade etmektedir.</p> <p>Sonucunun 2 puan olması, verileri depolayabilmek için değişken oluşturma ve kullanabilmeyi ifade etmektedir.</p> <p>Sonucunun 3 puan olması, aynı anda birden fazla değer tanımlamayı ve saklamayı sağlayan kod bloğudur.</p>
Eş Zamanlılık	<p>Senkronizasyon /Eş zamanlılık karakterin işleri istediğimiz sırayla yapmasını sağlayan beceridir. Kontrol yapılarındaki bazı özellikler ile doğrudan ilişkilidir.</p> <p>Sonucunun 1 puan olması, iki karakterlerin çalışma sırasını en basit olarak saniye bekle komutu kullanarak biri konuşurken diğeri dinler, diğeri dinlerken biri konuşur gibi özelliklerdir.</p> <p>Sonucunun 2 puan olması, Saniye bekle komutlarının yetersiz olduğu daha kapsamlı çalışmalarda kontrol bloklarının yanısıra algılama ve olaylar bloklarından haberi çağır, kod bloğunu çağır gibi komutlarının kullanılmasını ifade eder.</p> <p>Sonucunun 3 puan olması, olana kadar bekle, herşeyi durdur, zemin değiştiğinde gibi kontrol bloklarının kullanılarak koşul sağlanan</p>

Ziyaret edilecek ikinci sayfa, “Şekil 18.” de gösterilen ‘Form Sayfası ile Projeni Oluştur’ sayfasıdır. Bu sayfa, Scratch programında geliştirilen projelerin hesaplanabilmesi için hazırlık aşamasıdır. Her kod bloğu için proje numarası ve kullanılan blok seçilmektedir. ‘Kullanılan Bloklar’ listesi Makine Öğrenmesi Modeli eğitilirken kullanılan 139 blok tipi verisinin Scratch uygulamasındaki Türkçe karşılığıdır. EK-M de 139 blok tipinin Scratch uygulamasındaki karşığı verilmiştir.

“Şekil 18.” De gösterilen ekranda her bir blok seçildiğinde Ekle butonuna basılır. Yeni projeler ile değerlendirme yapmak istenirse sıfırla butonuna basılır ve tüm eski projeler silinir. Tüm bloklar eklendikten sonra bitir butonuna basılır ve değerlendirme sayfasına yönlendirilir.

**Akışta Kullandığın Kod Parçaları**

**Proje Numarası**

Proje Numarası

**Kullanılan bloklar**

▼

EKLE   SIFIRLA   BİTİR

Ana Sayfaya Dön

Şekil 18. Form Sayfası ile Projeni Oluştur Ekranı

Ziyaret edilecek ikinci sayfa ise “Şekil 19” daki ‘Csv Dosyası İle Projelerini Oluştur’ sayfasıdır. Bu sayfa, daha önceden ‘Form Sayfası ile Projeyi oluştur’ ekranından hazırlanmış projeleri dosyadan seç komutuyla yüklemeyi sağlamaktadır. Diğere bir özelliği ise Ek-M de belirtilen klavuz sayesinde proje CSV dosyalarını sıfırdan daha hızlı bir şekilde hazırlamayı sağlamaktadır. Özellikle çok büyük uygulamalarda tüm blokları tek tek seçme zorluğunu ortadan kaldırmaktadır. Bunun için sayfada bulunan örnek dosyayı indirip, kullanılan kod blokları Excel sayfası üzerinden kolaylıkla eklenebilir.

Scratch dışında benzeri blok tabanlı uygulamalarda da kullanılabilmesi için çeviri sözlüğü fonksiyonellik sağlamaktadır.

**BİLGİ İŞLEMSEL DÜŞÜNME BECERİLERİNİ DEĞERLENDİR**

1. Örnek dosyayı indir
2. Proje id kısmında her bir projeye rakamla değer ver
3. Her projede kullandığın blokları çeviri sözlüğüne bakarak yazılımsal terimine çevir
4. 'input.csv' olarak kaydet
5. Oluşturduğun dosyayı gözet butonunda bul ve yükle butonuna bas.
6. Değerlendirme Ekranına dön butonuna basarak değerlendirmeye başla

Gözet... Yükle

Değerlendirme Ekranı Yenile Örnek Dosyayı İndir Scratch ve Diğer Uygulamalar İçin Çeviri Sözlüğü Yazılımlar İfadeleri Ana Sayfaya Dön

**ÇEVİRİ SÖZLÜĞÜ**

Sıranın: 'Scratch Uygulamasındaki İfadesi' - 'Yazılımlar İfadesi'

1. 'Adım Git (ileri git)' - 'forward:'	2. 'Derece dön (Sola dön)' - 'turnLeft:'	3. 'Derece dön (Sağa dön)' - 'turnRight:'	4. 'Yönüne Dön' - 'heading:'
5. 'Yönü' - 'heading'	6. 'Fare okuna doğru dön (Yönüne Doğru dön)' - 'pointTowards:'	7. 'Saniyede X ve Y noktasına git' - 'gotoXy:'	8. 'Fare okuna git' - 'gotoSpriteOrMouse:'
9. 'Saniyede X ve Y ye Süzül' - 'glideSecs:toXy:elapsed:from:'	10. 'X koordinatında konumunu artır' - 'changeXposBy:'	11. 'Y koordinatında konumunu artır' - 'changeYposBy:'	12. 'X konumunu değiştir' - 'xpos:'
13. 'Y konumunu değiştir' - 'ypos:'	14. 'X konumunu' - 'xpos'	15. 'Y konumunu' - 'ypos'	16. 'Ekrandan Çıktıysa Sek' - 'bounceOffEdge'

Şekil 19. Csv Dosyası İle Projelerini Oluştur Sayfası

Ziyaret edilecek üçüncü sayfa, oluşturulan projelerin Değerlendirileceği yerdir. “Şekil 20.” değerlendirme ekranının görüntüsü mevcuttur. Bu sayfada kullanıcıya her düşünme becerisinde puanlamanın 0 ila 3 puan olduğu; 1 puanın Temel Seviye, 2 puanın

Geliştirme Seviyesi ve 3 puanın Yeterli Seviye olduğu belirtilmiştir. Toplam puanlar Kullanıcı bu sayfada değerlendirme puanını görür ve seviyesini değerlendirmiş olur.

## BİLİŞSEL DÜŞÜNME BECERİLERİNİ DEĞERLENDİR

SONUÇLAR 7 DÜŞÜNME BECERİSİ İÇİN 0 İLA 3 ARASINDA PUAN VERMEKTEDİR.  
 1 PUAN BECERİNİN TEMEL SEVİYE OLDUĞUNU, 2 PUAN BECERİNİN GELİŞTİRME SEVİYESİNDE OLDUĞUNU VE  
 3 PUAN YETERLİ SEVİYE OLDUĞUNU İFADE EDER.  
 MASTER PUANLARI EN FAZLA 21 OLABİLİR. MASTER PUANININ 0 - 7 PUAN ARASI TEMEL SEVİYE OLDUĞUNU ,  
 7-14 ARASI GELİŞTİRME SEVİYESİNDE VE 14 DEN BÜYÜK İSE YETERLİ SEVİYE OLDUĞUNU BELİRTMEKTEDİR.

1.CSV DOSYASI YADA FORM SAYFASINDAN PROJELERİNİ OLUŞTUR  
 2.RESME TIKLAYARAK SONUÇLARI HESAPLA  
 3.SONUÇLARI GÖSTER BUTONUNA BAS

Hesaplanıyor Hesaplandı...

[Sonuçları Göster](#)   [Yenile](#)   [Ana Sayfaya Dön](#)   [BİD becerileri detay](#)

Proje ve Numarası	Soyutlama ve Problem Ayrıştırma	Paralellik	Ezamanlılık	Akış Kontrolü	Kullanıcı Etkileşimi	Veri Gösterimi	Mantıksal Düşünme	MASTER PUAN	Değerlendirme
1	2 Puan	2 Puan	2 Puan	2 Puan	2 Puan	2 Puan	2 Puan	14 / 21	Geliştirme seviyesi
2	0 Puan	0 Puan	0 Puan	2 Puan	2 Puan	1 Puan	2 Puan	7 / 21	Temel seviye
3	0 Puan	3 Puan	0 Puan	3 Puan	3 Puan	0 Puan	3 Puan	12 / 21	Geliştirme seviyesi
4	0 Puan	0 Puan	0 Puan	1 Puan	0 Puan	1 Puan	2 Puan	4 / 21	Temel seviye
5	3 Puan	2 Puan	2 Puan	2 Puan	2 Puan	3 Puan	2 Puan	16 / 21	Yeterli seviye
6	0 Puan	0 Puan	0 Puan	0 Puan	0 Puan	0 Puan	1 Puan	1 / 21	Temel seviye
7	0 Puan	3 Puan	0 Puan	3 Puan	3 Puan	0 Puan	3 Puan	12 / 21	Geliştirme seviyesi
8	0 Puan	0 Puan	0 Puan	2 Puan	2 Puan	1 Puan	2 Puan	7 / 21	Temel seviye
9	2 Puan	2 Puan	2 Puan	2 Puan	2 Puan	2 Puan	2 Puan	14 / 21	Geliştirme seviyesi
10	2 Puan	2 Puan	2 Puan	2 Puan	2 Puan	2 Puan	2 Puan	14 / 21	Geliştirme seviyesi
11	0 Puan	3 Puan	0 Puan	3 Puan	3 Puan	0 Puan	3 Puan	12 / 21	Geliştirme seviyesi
12	0 Puan	0 Puan	0 Puan	0 Puan	0 Puan	0 Puan	1 Puan	1 / 21	Temel seviye
13	2 Puan	2 Puan	2 Puan	2 Puan	2 Puan	2 Puan	2 Puan	14 / 21	Geliştirme seviyesi
14	0 Puan	0 Puan	0 Puan	1 Puan	0 Puan	1 Puan	2 Puan	4 / 21	Temel seviye
15	1 Puan	1 Puan	1 Puan	1 Puan	1 Puan	1 Puan	1 Puan	7 / 21	Temel seviye
16	2 Puan	2 Puan	2 Puan	2 Puan	2 Puan	2 Puan	2 Puan	14 / 21	Geliştirme seviyesi
17	0 Puan	2 Puan	2 Puan	2 Puan	2 Puan	1 Puan	2 Puan	11 / 21	Geliştirme seviyesi
18	0 Puan	0 Puan	0 Puan	0 Puan	0 Puan	0 Puan	1 Puan	1 / 21	Temel seviye
19	0 Puan	0 Puan	0 Puan	1 Puan	0 Puan	1 Puan	2 Puan	4 / 21	Temel seviye
20	0 Puan	3 Puan	0 Puan	3 Puan	3 Puan	0 Puan	3 Puan	12 / 21	Geliştirme seviyesi
21	3 Puan	2 Puan	2 Puan	2 Puan	2 Puan	3 Puan	2 Puan	16 / 21	Yeterli seviye
22	2 Puan	3 Puan	3 Puan	3 Puan	3 Puan	2 Puan	2 Puan	18 / 21	Yeterli seviye
23	2 Puan	2 Puan	2 Puan	2 Puan	2 Puan	2 Puan	2 Puan	14 / 21	Geliştirme seviyesi
24	1 Puan	1 Puan	1 Puan	1 Puan	1 Puan	1 Puan	2 Puan	8 / 21	Geliştirme seviyesi
25	3 Puan	3 Puan	3 Puan	3 Puan	3 Puan	3 Puan	3 Puan	21 / 21	Yeterli seviye
26	1 Puan	1 Puan	1 Puan	1 Puan	1 Puan	1 Puan	1 Puan	7 / 21	Temel seviye
27	2 Puan	2 Puan	2 Puan	2 Puan	2 Puan	2 Puan	2 Puan	14 / 21	Geliştirme seviyesi
28	2 Puan	3 Puan	3 Puan	3 Puan	3 Puan	2 Puan	2 Puan	18 / 21	Yeterli seviye
29	3 Puan	2 Puan	2 Puan	2 Puan	2 Puan	3 Puan	2 Puan	16 / 21	Yeterli seviye
210	2 Puan	3 Puan	3 Puan	3 Puan	3 Puan	2 Puan	2 Puan	18 / 21	Yeterli seviye
211	2 Puan	2 Puan	2 Puan	2 Puan	2 Puan	2 Puan	2 Puan	14 / 21	Geliştirme seviyesi
212	1 Puan	1 Puan	1 Puan	1 Puan	1 Puan	1 Puan	1 Puan	7 / 21	Temel seviye
213	3 Puan	2 Puan	2 Puan	2 Puan	2 Puan	3 Puan	2 Puan	16 / 21	Yeterli seviye
214	1 Puan	1 Puan	1 Puan	1 Puan	1 Puan	1 Puan	2 Puan	8 / 21	Geliştirme seviyesi
215	1 Puan	1 Puan	1 Puan	1 Puan	1 Puan	1 Puan	1 Puan	7 / 21	Temel seviye
216	2 Puan	3 Puan	3 Puan	3 Puan	3 Puan	2 Puan	2 Puan	18 / 21	Yeterli seviye
217	2 Puan	3 Puan	3 Puan	3 Puan	3 Puan	2 Puan	2 Puan	18 / 21	Yeterli seviye
218	1 Puan	1 Puan	1 Puan	1 Puan	1 Puan	1 Puan	1 Puan	7 / 21	Temel seviye
219	1 Puan	1 Puan	1 Puan	1 Puan	1 Puan	1 Puan	2 Puan	8 / 21	Geliştirme seviyesi
220	2 Puan	2 Puan	2 Puan	2 Puan	2 Puan	2 Puan	2 Puan	14 / 21	Geliştirme seviyesi

Şekil 20. Geliştirilen Aracın Değerlendirme Ekranı

## **5.7.ÖRNEK OLAY İLE UYGULAMANIN KULLANILMASI**

Öncelikle Uzman tarafından Scratch blog tabanlı programlama platformunda yapılacak bir örnek olay belirlenmiştir. Örnek olay; kullanıcının girdiği iki sayıyı toplama, çıkarma, çarpma ve bölme işlemi yapabilen bir Scratch uygulamasıdır. Bu bölüm iki alt başlıktan oluşmaktadır. Birinci bölüm de, Scratch uygulaması üzerinden gerçekleştirilen örnek olay ile BİD becerilerinin 7 boyutunun 1 puandan 3 puana kadar nasıl değerlendirildiği gösterilmektedir. İkinci bölüm de Srtach uygulaması üzerinde belirlenen örnek olayın öğrenciler üzerinde uygulanarak, değerlendirme aracının öğrenci üzerindeki etkisi gösterilmektedir.

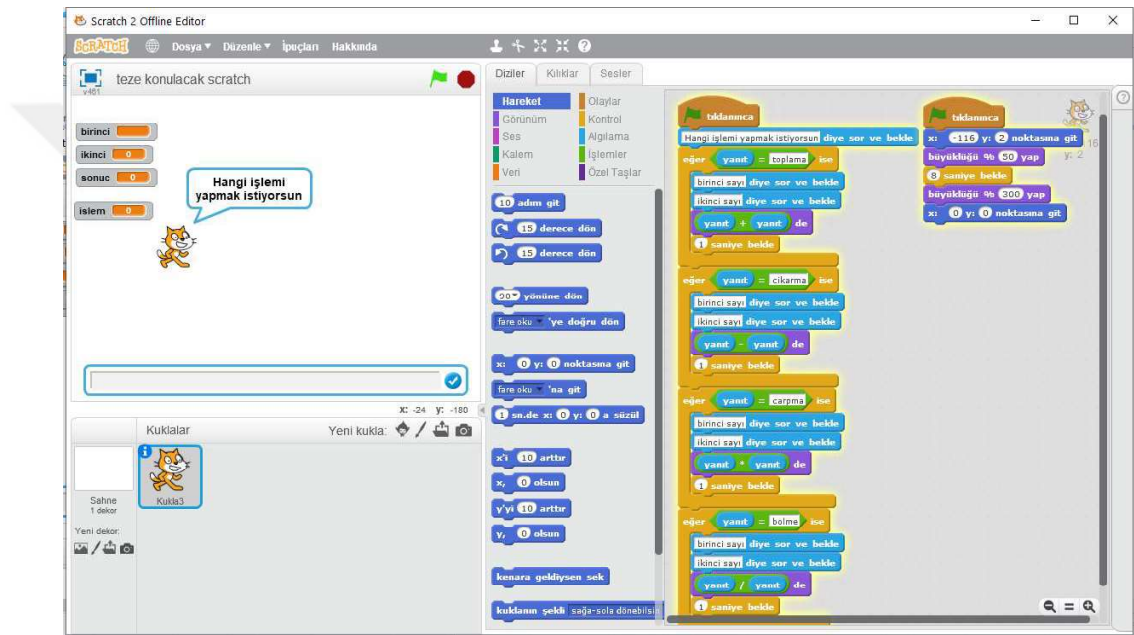
### **5.7.1. Geliştirilen Araç ile BİD Becerilerinin 7 Boyutunun Hesaplaması**

Bu bölümde uzman tarafından belirlenen örnek olayın BİD becerilerinin her biri için 1,2 ve 3 puan olma durumları incelenmektedir.

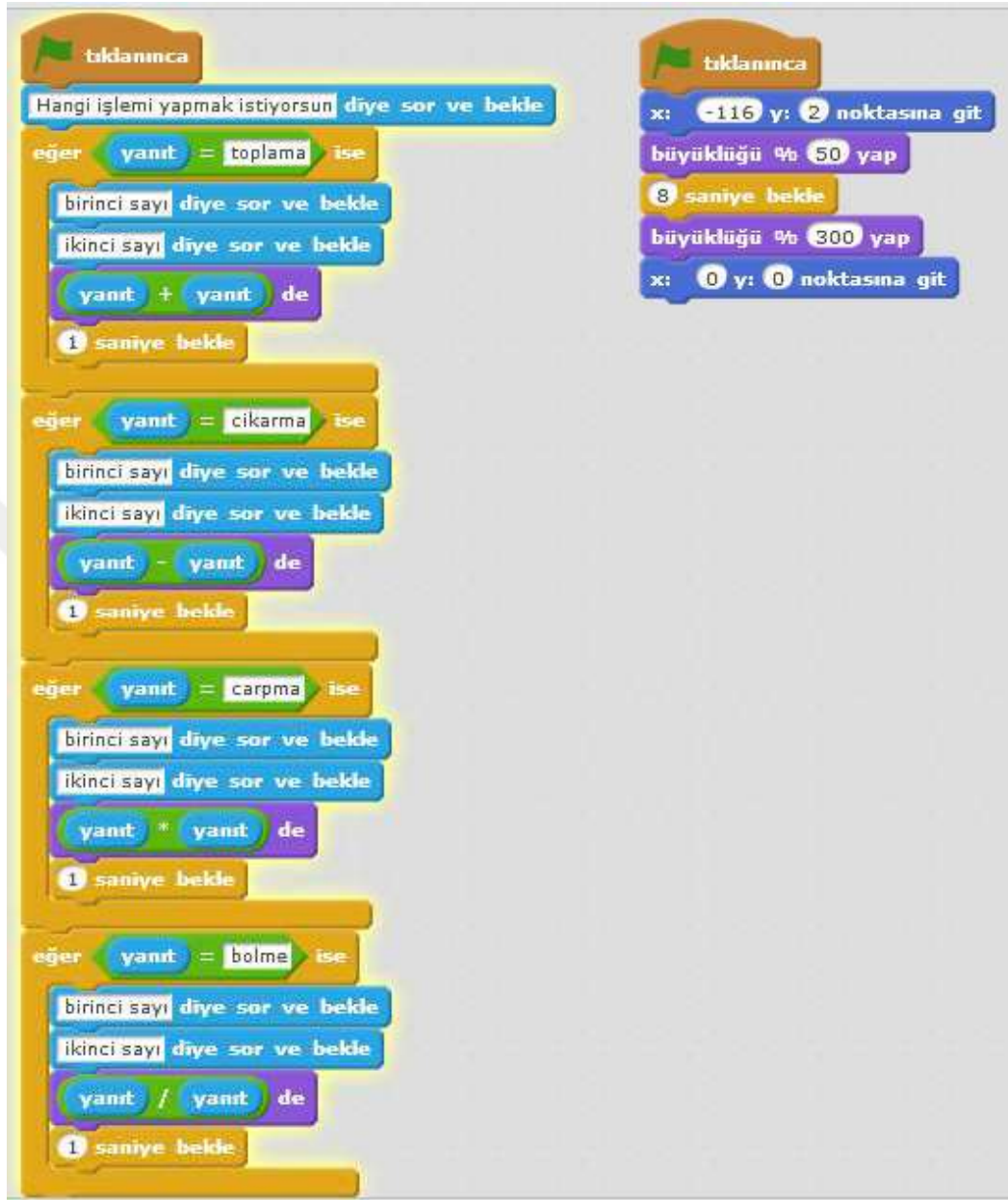


### 5.7.1.1.BİD becerilerinin 1 Puan olma durumu

Becerilerin 1 puan olduğu örnek uygulamada Scratch ekranındaki kedi girilen iki sayının toplama, çıkarma, çarpma ve bölme işlemini yaptıktan sonra ekranda boyutu ve şekli değişmektedir. Uygulamanın tam ekran görüntüsü “Şekil 21.” de, kullanılan kod bloklarının detaylı halı “Şekil 22.” de gösterilmiştir.



Şekil 21. BİD Becerileri 1 Puan Olan Örnek Uygulama Ekranı



Şekil 22. BİD Becerileri 1 Puan Olan Örnek Uygulamada kullanılan kod bloklarının detaylı gösterimi

Bu projenin değerlendirilebilmesi için “Form Sayfası ile Projeyi Oluştur” Ekranından kullanılan kod blokları seçilmiş ve değerlendirme için CSV dosyası oluşturulmuştur. “Tablo 30.” de oluşturulan CSV dosyasının içeriği gösterilmiştir.



Tablo 30. BİD Becerileri 1 Puan Olan Örnek Uygulamanın hesaplama yapması için oluşturduğu CSV dosyası

Proje Numarası, Blok Tipi			
Proje1,whenClicked,	Proje1,wait: elapsed:from:;	Proje1,doAsk,	Proje1,answer,
Proje1,doAsk,	Proje1,dolf,	Proje1,doAsk,	Proje1,\/,
Proje1,dolf,	Proje1,answer,	Proje1,answer,	Proje1,answer,
Proje1,answer,	Proje1,=,	Proje1,*,	Proje1,say:;
Proje1,=,	Proje1,doAsk,	Proje1,answer,	Proje1,whenClicked,
Proje1,doAsk,	Proje1,doAsk,	Proje1,say:;	Proje1,gotoX:y:;
Proje1,doAsk,	Proje1,answer,	Proje1,wait:elapsed: from:;	Proje1,setSizeTo:;
Proje1,answer,	Proje1,-,	Proje1,dolf,	Proje1,wait:elapsed:from:;
Proje1,+,	Proje1,answer,	Proje1,=,	Proje1,setSizeTo:;
Proje1,answer,	Proje1,dolf,	Proje1,doAsk,	Proje1,gotoX:y:;
Proje1,say:;	Proje1,=,	Proje1,doAsk,	

Oluşturulan örnek olay BİD becerilerinin 7 boyutuna göre incelendiğinde;

*Soyutlama ve Problem Ayırıştırma*; projenin 2 parçaya bölünerek, kodların 2 farklı tıklanıcı (whenClicked) kodunun içerisinde olması karakterin davranışlarını bölerek hareket ettirebildiğini ve Proje1'in bu becerisinin 1 puan olduğunu göstermektedir.

*Mantıksal Düşünme*; programda toplama, çıkarma, çarma ve bölme işlemlerinin her biri için eğer (doIf) ifadesi kullanılmıştır. Bu temel şart ifadelerinin kullanılabilmesi Proje1'in bu becerisinin 1 puan olduğunu göstermektedir.

*Paralellik;* kodların 2 farklı tıklanıza (whenClicked) içerisinde karakterin aynı anda ‘Hangi işlemi yapmak istersin? diye sor ve bekle’ (doAsk) komutuyla, ‘x -116 ve y 2 noktasına git ‘(gotoX:y:) komutunu ve benzeri kod bloklarını aynı anda başlatabilmesi Proje1’in bu becerisinin 1 puan olduğunu göstermektedir.

*Akış Kontrolü;* eğer yanıt toplamaysa, birinci ve ikinci sayıyı sor ve bekle, toplama işlemi yap, sonucu ekrana yazdır ve bir saniye bekle komutlarının çarpma, çıkarma ve bölme için birden fazla tekrar eden işlemin arka arkaya sıralanmıştır. Bunun gibi bir dizi tekrar eden işlemleri arka arkaya sıralam becerisidir. Bu sebeplerle Proje1’in bu becerisi 1 puan olarak değerlendirilmektedir.

*Kullanıcı Etkileşimi;* programın başlayabilmesi için yeşil bayrağa tıklanınca (whenClicked) komutu kullanılmıştır. Kullanıcının programı çalıştırabilmesi için her şeyden evvel sol ekrandaki yeşil bayrağa basması gerekmektedir. Bu da kullanıcıyla program arasında kullanıcı etkileşimi sağlamaktadır. Bu sebeple Proje1’in bu becerisi 1 puan olarak değerlendirilmektedir.

*Veri Gösterimi;* Kuklanın büyüklüğü ‘büyüklüğü %50 yap’(setSizeTo:), hareketi ‘x -116 ve y 2 noktasına git ‘(gotoX:y:) gibi kod bloklarının kullanılması dolayısıyla Proje1’in bu becerisi 1 puan olarak değerlendirilmektedir.

*Eş Zamanlılık;* ‘1 saniye ve 8 saniye bekle’(wait:elapsed:from:) komutları bazı kodlar çalışırken bazılarının beklemesini sağlayan bloklardır. Bu bloklar sayesinde senkronizasyonun birbirini bekleme şekilde sağlanmaktadır. Sonuç olarak, Proje1’in bu becerisi 1 puan olarak değerlendirilmektedir.

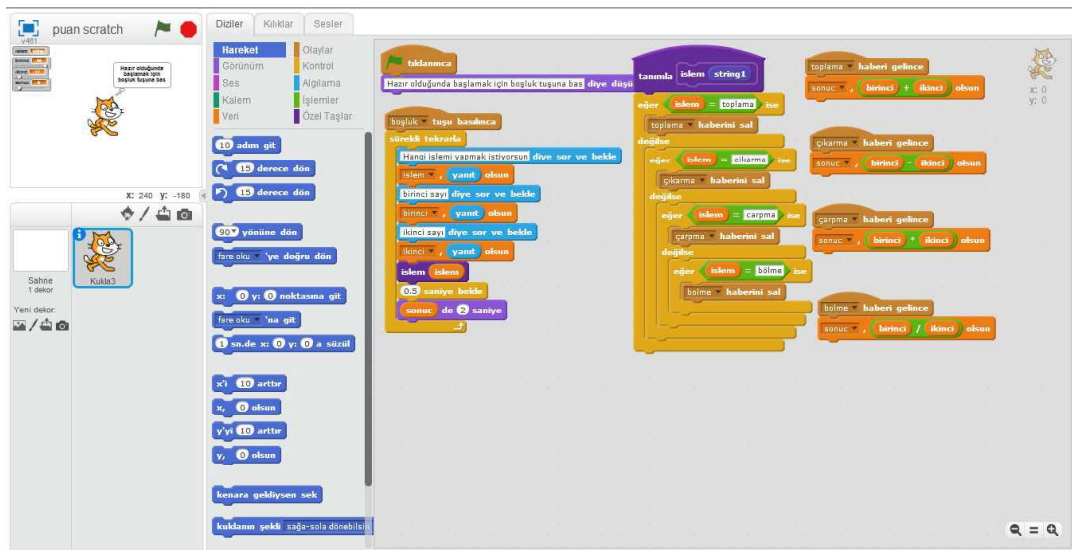
Geliştirilen değerlendirme aracında Proje1 değerlendirildiğinde her bir BİD becerisinin puanı 1, temel seviye olduğu tespit edilmiştir. 7 BİD becerilerinin toplamından oluşan Master puan 7 üzerinden 7 puan gelerek Temel Seviye olarak değerlendirilmiştir. Değerlendirme aracını kullanarak değerlendirilen BİD becerileri sonuçları “Şekil 23.” de gösterilmiştir.

Proje Uyarası	Soyutlama ve Problem Ayrıştırma	Paralellik	Eszamanlilik	Akis Kontrolü	Kullanici Etkilesimi	Veri Gosterimi	Mantiksalsal Dusatma	MASTER PUAN	Değerlendirme
Proje1	1/3	1/3	1/3	1/3	1/3	1/3	1/3	7 / 21	Temel seviye

Şekil 23. Örnek olay Proje1’in değerlendirme aracı ile puanlama sonucu

### 5.7.1.2.BİD becerilerinin 2 Puan olma durumu

Becerilerin 2 puan olduğu örnek uygulamada Scratch ekranındaki kedinin belli tuşlara basana kadar beklemesi ve girilen iki sayının toplama, çıkarma, çarpma ve bölme işlemini yapmasını ifade etmektedir. Uygulamanın tam ekran görüntüsü “Şekil 24.” de, kullanılan kod bloklarının detaylı hali ise “Şekil 25.” de gösterilmiştir.



Şekil 24. BİD Becerileri 2 Puan Olan Örnek Uygulama Ekranı

```

    tıklanınca
    Hazır olduğunda başlamak için boşluk tuşuna bas diye düşün
    boşluk tuşu basılınca
    sürekli tekrarla
    Hangi işlemi yapmak istiyorsun diye sor ve bekle
    işlem , yanıt olsun
    birinci sayı diye sor ve bekle
    birinci , yanıt olsun
    ikinci sayı diye sor ve bekle
    ikinci , yanıt olsun
    işlem işlem
    0.5 saniye bekle
    sonuc de 2 saniye
  
```

```

    tanımla işlem string1
    eğer işlem = toplama ise
    toplama haberi gelince
    sonuc , birinci + ikinci olsun
    değilse
    eğer işlem = çıkarma ise
    çıkarma haberi gelince
    sonuc , birinci - ikinci olsun
    değilse
    eğer işlem = çarpma ise
    çarpma haberi gelince
    sonuc , birinci * ikinci olsun
    değilse
    eğer işlem = bölme ise
    bölme haberi gelince
    sonuc , birinci / ikinci olsun
  
```

Şekil 24. BİD Becerileri 2 Puan Olan Örnek Uygulamada kullanılan kod bloklarının detaylı gösterimi

Bu projenin değerlendirilebilmesi için “Form Sayfası ile Projeni Oluştur” Ekranından kullanılan kod blokları seçilmiş ve değerlendirme için CSV dosyası oluşturulmuştur. “Tablo 31.” de oluşturulan. CSV dosyasının içeriği gösterilmiştir.

Tablo 31. BİD Becerileri 2 Puan Olan Örnek Uygulamanın hesaplama yapması için oluşturduğu CSV dosyası

PROJE NUMARASI, BLOK TİPİ			
Proje2,whenGreenFlag,	Proje2,readVariable,	Proje2,readVariable,	Proje2,readVariable,
Proje2,think:,	Proje2,wait:elapsed:from:,	Proje2,=,	Proje2,-,
Proje2,whenKeyPressed,	Proje2,say:duration:elapsed:from:,	Proje2,broadcast:,	Proje2,readVariable,
Proje2,doForever,	Proje2,readVariable,	Proje2,dof,	Proje2,whenIReceive,
Proje2,doAsk,	Proje2,getParam,	Proje2,readVariable,	Proje2,setVar:to:,
Proje2,setVar:to:,	Proje2,dofElse,	Proje2,=,	Proje2,readVariable,
Proje2,answer,	Proje2,readVariable,	Proje2,broadcast:,	Proje2,*,
Proje2,doAsk,	Proje2,=,	Proje2,whenIReceive,	Proje2,readVariable,
Proje2,setVar:to:,	Proje2,broadcast:,	Proje2,setVar:to:,	Proje2,whenIReceive,
Proje2,answer,	Proje2,dofElse,	Proje2,readVariable,	Proje2,setVar:to:;
Proje2,doAsk,	Proje2,readVariable,	Proje2,+,	Proje2,readVariable,
Proje2,setVar:to:,	Proje2,=,	Proje2,readVariable,	Proje2,√,
Proje2,answer,	Proje2,broadcast:,	Proje2,whenIReceive,	Proje2,readVariable,
Proje2,call,	Proje2,dofElse,	Proje2,setVar:to:	

Oluşturulan örnek olay BİD becerilerinin 7 boyutuna göre incelendiğinde;

*Soyutlama ve Problem Ayırıştırma;* kod bloklarının fonksiyon olarak tanımlayarak gerektiğinde farklı değişkenler ile kod parçalarını çağırabilme yeteneği kullanılmıştır. Scratch uygulamasında kullanılan fonksiyon, ‘Tanımla İşlem’ (getParam) kodunun olması Proje1’in bu becerisinin 2 puan olduğunu göstermektedir.

*Mantıksal Düşünme;* ‘Eğer’(doIf) ifadesinin yanı sıra, ‘Eğerse değilse’ (doIfElse) ifadesinin kullanılması durumudur. İşlem toplama ise farklı toplama değilse farklı işlem yapması, iki durumda da iki ayrı ifadenin bulunması Proje1’in bu becerisinin 2 puan olduğunu göstermektedir.

*Paralellik;* ‘boşluk tuşuna basılınca’ (whenKeyPressed) ya da ‘haberi gelince’ (whenIReceive) gibi olay bloklarının programa başlayınca etkileşimli olarak işlemleri yapabilmesidir. Bu Proje1’in Paralellik becerisinin 2 puan olduğunu göstermektedir.

*Akış Kontrolü;* Kodu yazarken birden fazla tekrar etmesinin engellemek için kullanılan kod bloklarıdır. ‘Sürekli tekrarla’ (doForever) gibi blokların kullanılması bu becerinin 2 puan olarak değerlendirilmesini sağlamaktadır.

*Kullanıcı Etkileşimi;* klavyeden tuşa basıldığında, fare okuna değmesi durumunda, kuklaya tıklanınca gibi bloklar sayesinde kullanıcı ile interaktif olunmasıdır. ‘Boşluk tuşuna basılınca’ (whenKeyPressed) kod bloğu ile kullanıcı etkileşimi sağlanmıştır. Bu sebeple bu beceri 2 puan olarak değerlendirilmektedir.

*Veri Gösterimi;* kullanıcının girdiği cevapları değişkende tutan ve gerektiğinde çağırabilen blok türüdür. ‘İşlem yanıt olsun’(setVar:to:) ve ‘Değişkeni

Oku’(readVariable) bloklarının kullanılması bu becerinin 2 puan olduğunu göstermektedir.

*Eş Zamanlılık*; ‘Haberı Çağır’, kod bloğunu çağır gibi komutları senkronizasyonu sağlaması durumudur. ‘Haberı Sal’(broadcast:) ve ‘heberı Gelince’(whenIReceive) bloklarının kullanılması bu becerinin 2 puan olduğunu göstermektedir.

Geliştirilen değerlendirme aracında proje değerlendirildiğinde her bir BİD becerisinin puanı 2, geliştirme seviyesi olduğu tespit edilmiştir. Master puanı 21 üzerinden 14 puan hesaplanarak Geliştirme Seviye olarak değerlendirilmiştir. Değerlendirme aracının hesaplama sonuçları “Şekil 25.” de gösterilmiştir.

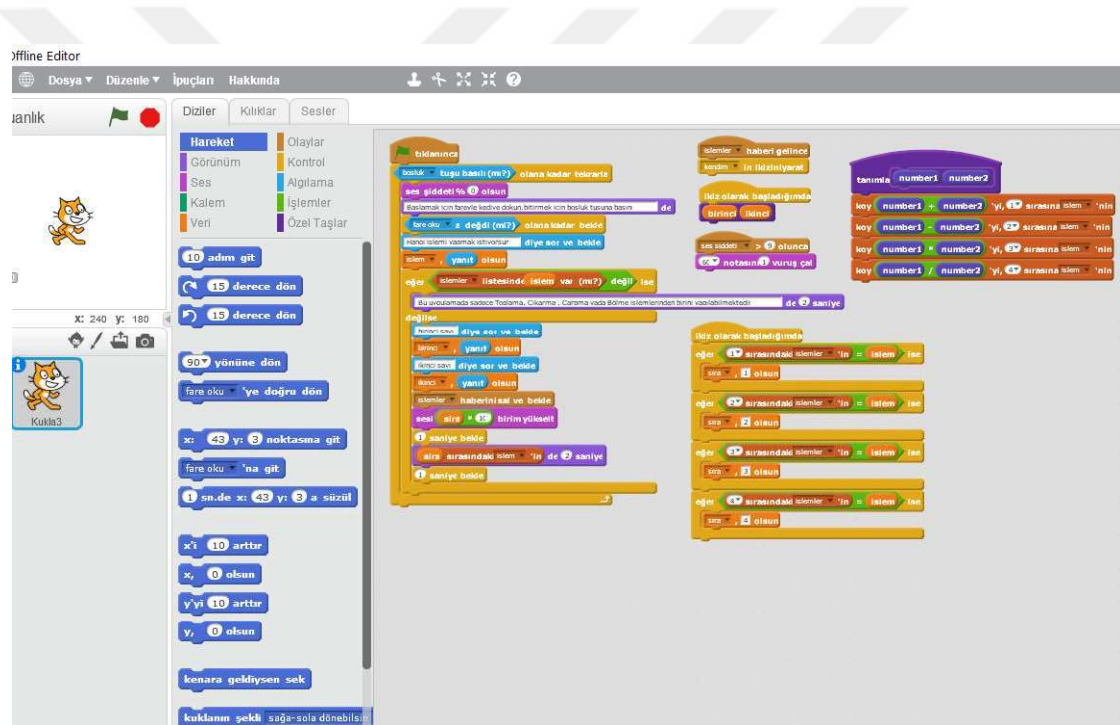
Proje Numarası	Soyutlama ve Problem Ayrıştırma	Paralellik	Eszamanlilik	Akis Kontrolü	Kullanıcı Etkilesimi	Veri Gösterimi	Mantıksal Düşünme	MASTER PUAN	Değerlendirme
Proje2	2/3	2/3	2/3	2/3	2/3	2/3	2/3	14 / 21	Gelistirme seviyesi

Projeler 7 Bilissel Düşünme Boyutunda 0-3 Puan arası değerlendirilmiştir

Şekil 25. Örnek olay Proje2’in değerlendirme aracı ile puanlama sonucu

### 5.7.1.3.BİD becerilerinin 3 Puan olma durumu

Becerilerin 3 puan olduğu örnek uygulamada Scratch ekranındaki kedinin belli tuşlara basana kadar çalıştığı, fare oku değene kadar programa başlamadığı, klavyeden girilen verilere göre işlem yapılırken Scratch kedisinin ses çıkardığı ve girilen iki sayının toplama, çıkarma, çarpma ve bölme işlemini yapabilmektedir. Uygulamanın tam ekran görüntüsü “Şekil 26.” de, kullanılan kod bloklarının detayları “Şekil 27.” de gösterilmiştir.



Şekil 26. BİD Becerileri 3 Puan Olan Örnek Uygulama Ekranı



The image shows a Scratch script for a calculator application. The script is organized into several sections:

- Initialization:** Starts with a 'when green flag clicked' event, followed by a 'while loop' that checks if the spacebar is pressed. Inside the loop, it sets volume to 0, asks for an operation, and checks if it's in a list.
- Operation List:** A list of operations: '+', '-', '\*', and '/'. Each operation is associated with a specific sound effect.
- Number Input:** Asks for two numbers, 'number1' and 'number2', and performs the operation based on the selected operation.
- Volume Control:** A 'when clicked' event that sets the volume to 60 and plays a sound.

The code blocks are as follows:

```

when green flag clicked
  while loop (until spacebar is pressed)
    set volume to 0
    ask "Hangi işlemi yapmak istiyorsun" and wait
    if "işlemler" list contains "işlem" (no)
      then
        ask "Birinci sayı" and wait
        set "birinci" to answer
        ask "İkinci sayı" and wait
        set "ikinci" to answer
        say "işlemler" message
        set volume to "sıra" * 10
        wait 1 seconds
        set "sıra" to "sırasındaki işlem" in de 2 saniye
        wait 1 seconds
      else
        say "Bu uygulamada sadece Toplama, Çıkarma, Çarpma yada Bölme işlemlerinden birini yapılabilir" de 2 saniye
    end if
  end while

when clicked
  set volume to 60
  play sound "ses şiddeti > 9 olunca"

```

Şekil 27. BİD Becerileri 3 Puan Olan Örnek Uygulamada kullanılan kod bloklarının detaylı gösterimi

Bu projenin değerlendirilebilmesi için “Form Sayfası ile Projeni Oluştur” Ekranından kullanılan kod blokları seçilmiş ve değerlendirme için CSV dosyası oluşturulmuştur. “Tablo 32.” de oluşturulan. CSV dosyasının içeriği gösterilmiştir.

Tablo 32. BİD Becerileri 3 Puan Olan Örnek Uygulamanın hesaplama yapması için oluşturduğu CSV dosyası

Proje Numarası, Blok Tipi			
Proje3,whenGreenFlag,	Proje3,setVar:to:,	Proje3,readVariable,	Proje3,getLine:ofList:,
Proje3,keyPressed:,	Proje3,answer,	Proje3,+,	Proje3,=,
Proje3,doUntil,	Proje3,doBroadcastAndWait,	Proje3,readVariable,	Proje3,readVariable,
Proje3,setVolumeTo:,	Proje3,changeVolumeBy:,	Proje3,getLine:ofList:,	Proje3,setVar:to:,
Proje3,say:,	Proje3,readVariable,	Proje3,readVariable,	Proje3,dof,
Proje3,doWaitUntil,	Proje3,*,	Proje3,-,	Proje3,getLine:ofList:,
Proje3,mousePressed,	Proje3,wait:elapsed:from:,	Proje3,readVariable,	Proje3,=,
Proje3,doAsk,	Proje3,say:duration:elapsed:from:,	Proje3,getLine:ofList:,	Proje3,readVariable,
Proje3,setVar:to:,	Proje3,getLine:ofList:,	Proje3,readVariable,	Proje3,setVar:to:,
Proje3,answer,	Proje3,readVariable,	Proje3,*,	Proje3,dof,
Proje3,dofElse,	Proje3,wait:elapsed:from:,	Proje3,readVariable,	Proje3,getLine:ofList:,
Proje3,not,	Proje3,whenIReceive,	Proje3,getLine:ofList:,	Proje3,=,
Proje3,list:contains:,	Proje3,createCloneOf,	Proje3,readVariable,	Proje3,readVariable,
Proje3,readVariable,	Proje3,whenCloned,	Proje3,V,	Proje3,setVar:to
Proje3,say:duration:elapsed:from:,	Proje3,getParam,	Proje3,readVariable,	Proje3,dof,
Proje3,doAsk,	Proje3,readVariable,	Proje3,whenSensorGreaterThan,	Proje3, getLine:ofList:,
Proje3,setVar:to:,	Proje3,readVariable,	Proje3,noteOn:duration:elapsed:from:,	Proje3,=,
Proje3,answer,	Proje3,call,	Proje3,whenCloned,	Proje3, readVariable,
Proje3,doAsk,	Proje3,getLine:ofList:,	Proje3,dof,	Proje3,setVar:to

Oluşturulan örnek olay BİD becerilerinin 7 boyutuna göre incelendiğinde;

*Soyutlama ve Problem Ayırıştırma*; aynı işlem birden fazla olduğunda, işlemleri tek tek yapmak yerine kopyasının oluşturulduğu kod bloğudur. ‘İkiz olarak başlatıldığında’ (whenCloned) ve ‘İkizini Yarat’(createCloneOf) kodunun olması Proje3’ün bu becerisinin 3 puan olduğunu göstermektedir.

*Mantıksal Düşünme*; Birden fazla ‘Eğer’(doIf) ve ‘Eğerse değilse’ (doIfElse) ifadesinin yanı sıra ‘değil’(NOT) ifadesini içermektedir. Olumsuz koşulun sağlanması durumunu kontrol eden kod bloğudur ve Proje3’ün bu becerinin 3 puan olduğunu göstermektedir.

*Paralellik*; ‘Haber Gelince’(whenIReceive) komutuyla beraber ‘ikizini Yarat’(createCloneOf) işlemi gerçekleşirken bir tarafta ‘fonksiyon’(getParam) özelliğinin eş zamanı olarak çalışmaya başlayıp aynı anda işlem yapmasını ve bağlantılı sonuçlar vermesini sağlamaktadır. Bu becerinin 3 puan olduğunu göstermektedir.

*Akış Kontrolü*; Kodun ne kadar tekrar edeceği bilinmediği zamanlarda, ‘Olana kadar tekrarla’(doUntil) gibi interaktif olarak işlemin kaç kez yapılacağını dinamik halde getiren kod bloklarıdır. Proje3’ün bu becerisi 3 puan olarak değerlendirilmiştir.

*Kullanıcı Etkileşimi*; Kullanıcının hareketi sonucu, ‘60 notasını 1 vuruş çal’(playDrum) olduğu, ‘Ses şiddetinin %0 olsun’ (soundLevel) ve ‘Ses şiddeti >10 olunca’ (whenSensorGreaterThan) kodları sayesinde program kullanıcının tepkilerine interaktif olarak cevap vermekte ve kullanıcı etkileşimini en üst seviyeye çıkarmaktadır. Proje3’ün bu becerisi 3 puan olarak değerlendirilmektedir.

*Veri Gösterimi*; aynı anda birden fazla değişken saklamayı ve kullanmayı sağlayan liste bloklarıdır. Proje3 de fonksiyonun içerisinde ve ana kod bloğunda ‘Listeye koy’(append:toList:), ‘listenin belirli Sırasındaki elemanın’(insert:at:ofList) kullanılması ve listeye ilgili işlemlerin olması bu becerinin 3 puan olduğunu göstermektedir.

*Eş Zamanlılık*; ‘Olana Kadar bekle’(doWaitUntil) kodu ile fare oku değene kadar program başlamamakta ve koşul gerçekleştiğinde işlemler eş zamanlı olarak çalışmaktadır. Proje3’ün bu becerisi 3 puan olduğu görülmektedir.

Geliştirilen değerlendirme aracında Proje3 değerlendirildiğinde her bir BİD becerisinin puanı 3, yeterli seviyesi olduğu tespit edilmiştir. Master puanı 21 üzerinden 21 puan hesaplanarak Yeterli Seviye olarak değerlendirilmiştir. Değerlendirme aracının sonuçları “Şekil 28.” de gösterilmiştir.

Proje Numarası	Soyutlama ve Problem Ayrıştırma	Paralellik	Eszamanlılık	Akis Kontrolü	Kullanıcı Etkilesimi	Veri Gösterimi	Mantıksal Düşünme	MASTER PUAN	Değerlendirme
Proje3	3/3	3/3	3/3	3/3	3/3	3/3	3/3	21 / 21	Yeterli seviye

Şekil 28. Örnek olay Proje2’in değerlendirme aracı ile puanlama sonucu

### **5.7.2. Örnek Olay ile Değerlendirme Aracının Öğrenci Üzerindeki Etkisinin Gösterilmesi**

Değerlendirme aracının öğrenme üzerinde etkisini ölçebilmek için, öncelikle Uzman tarafından scratch programıyla yapılacak bir proje belirlendi. Bu proje iki sayının toplama, çıkarma, çarpma ve bölme işlemlerini gerekli olan tüm detaylarıyla yapabildiği bir scratch uygulamasıdır.

20 Ortaokul seviyesindeki öğrenciye özellikleri belirtilen uygulamayı ön hazırlık olmadan kendi başlarına yapmaları istendi. Sonrasında ‘Form Sayfası ile Proje Oluştur’ kısmında yapılan uygulamalar değerlendirme aracında yapılması gereken sırayla tanımlandı ve değerlendirme işlemi yapıldı.

Uzman tarafından bireysel olarak yapılmış örnek olayda, uzmanın puanı 21 üzerinden 21 olarak değerlendirilmiştir. Uzman tarafından yapılan Scratch uygulaması “Şekil 26.” da detaylı olarak gösterilmiş ve puanlama sonucu “Şekil 28.” de gösterilmiştir.

Değerlendirme aşamaları bittikten sonra kullanıcılar BİD becerileri kriterleri ekranından becerilerinin durumunu öğrenmiş oldu. Uzman beceri puanı 1 puan ve atılan öğrencilerle her beceri türü için farklı örnekler yapılarak beceriler iyileştirildi, sonrasında aynı kullanıcılar yine aynı uygulamayı tekrar gerçekleştirildi. “Tablo 33” da 20 kullanıcının birinci ve ikinci aşamada aynı düşünme becerilerinden kaç aldıkları aldıkları puanlar birinci versiyon için V1 ve ikinci versiyon için V2 ile ifade edilmiştir..

Tablo 33. 20 kullanıcının yaptığı uygulamaların V1 ve V2 puanlarının karşılaştırılması

Proje Numarası	Soyutlama ve Benzerlik	Paralellik	Eszamanlilik	Akış Kontrolü	Kullanıcı Etkilesimi	Veri Gösterimi	Mantıksal Düşünme	MASTER PUAN	Değerlendirme
1 V1	2/3	2/3	2/3	2/3	2/3	2/3	2/3	14 / 21	Gelistirme seviyesi
1 V2	3/3	2/3	2/3	2/3	2/3	3/3	2/3	16 / 21	Yeterli seviye
2 V1	0/3	0/3	0/3	2/3	2/3	1/3	2/3	7 / 21	Temel seviye
2 V2	2/3	3/3	3/3	3/3	3/3	2/3	2/3	18 / 21	Yeterli seviye
3 V1	0/3	3/3	0/3	3/3	3/3	0/3	3/3	12 / 21	Gelistirme seviyesi
3 V2	2/3	2/3	2/3	2/3	2/3	2/3	2/3	14 / 21	Gelistirme seviyesi
4 V1	0/3	0/3	0/3	1/3	0/3	1/3	2/3	4 / 21	Temel seviye
4 V2	1/3	1/3	1/3	1/3	1/3	1/3	2/3	8 / 21	Gelistirme seviyesi
5 V1	3/3	2/3	2/3	2/3	2/3	3/3	2/3	16 / 21	Yeterli seviye
5 V2	3/3	3/3	3/3	3/3	3/3	3/3	3/3	21 / 21	Yeterli seviye
6 V1	0/3	0/3	0/3	0/3	0/3	0/3	1/3	1 / 21	Temel seviye
6 V2	1/3	1/3	1/3	1/3	1/3	1/3	1/3	7 / 21	Temel seviye
7 V1	0/3	3/3	0/3	3/3	3/3	0/3	3/3	12 / 21	Gelistirme seviyesi
7 V2	2/3	2/3	2/3	2/3	2/3	2/3	2/3	14 / 21	Gelistirme seviyesi
8 V1	0/3	0/3	0/3	2/3	2/3	1/3	2/3	7 / 21	Temel seviye
8 V2	2/3	3/3	3/3	3/3	3/3	2/3	2/3	18 / 21	Yeterli seviye
9 V1	2/3	2/3	2/3	2/3	2/3	2/3	2/3	14 / 21	Gelistirme seviyesi
9 V2	3/3	2/3	2/3	2/3	2/3	3/3	2/3	16 / 21	Yeterli seviye
10 V1	2/3	2/3	2/3	2/3	2/3	2/3	2/3	14 / 21	Gelistirme seviyesi
10 V2	2/3	3/3	3/3	3/3	3/3	2/3	2/3	18 / 21	Yeterli seviye
11 V1	0/3	3/3	0/3	3/3	3/3	0/3	3/3	12 / 21	Gelistirme seviyesi
11 V2	2/3	2/3	2/3	2/3	2/3	2/3	2/3	14 / 21	Gelistirme seviyesi
12 V1	0/3	0/3	0/3	0/3	0/3	0/3	1/3	1 / 21	Temel seviye
12 V2	1/3	1/3	1/3	1/3	1/3	1/3	1/3	7 / 21	Temel seviye

<b>13 V1</b>	2/3	2/3	2/3	2/3	2/3	2/3	2/3	14 / 21	Gelistirme seviyesi
<b>13 V2</b>	3/3	2/3	2/3	2/3	2/3	3/3	2/3	16 / 21	Yeterli seviye
<b>14 V1</b>	0/3	0/3	0/3	1/3	0/3	1/3	2/3	4 / 21	Temel seviye
<b>14 V2</b>	1/3	1/3	1/3	1/3	1/3	1/3	2/3	8 / 21	Gelistirme seviyesi
<b>15 V1</b>	1/3	1/3	1/3	1/3	1/3	1/3	1/3	7 / 21	Temel seviye
<b>15 V2</b>	1/3	1/3	1/3	1/3	1/3	1/3	1/3	7 / 21	Temel seviye
<b>16 V1</b>	2/3	2/3	2/3	2/3	2/3	2/3	2/3	14 / 21	Gelistirme seviyesi
<b>16 V2</b>	2/3	3/3	3/3	3/3	3/3	2/3	2/3	18 / 21	Yeterli seviye
<b>17 V1</b>	0/3	2/3	2/3	2/3	2/3	1/3	2/3	11 / 21	Gelistirme seviyesi
<b>17 V2</b>	2/3	3/3	3/3	3/3	3/3	2/3	2/3	18 / 21	Yeterli seviye
<b>18 V1</b>	0/3	0/3	0/3	0/3	0/3	0/3	1/3	1 / 21	Temel seviye
<b>18 V2</b>	1/3	1/3	1/3	1/3	1/3	1/3	1/3	7 / 21	Temel seviye
<b>19 V1</b>	0/3	0/3	0/3	1/3	0/3	1/3	2/3	4 / 21	Temel seviye
<b>19 V2</b>	1/3	1/3	1/3	1/3	1/3	1/3	2/3	8 / 21	Gelistirme seviyesi
<b>20 V1</b>	0/3	3/3	0/3	3/3	3/3	0/3	3/3	12 / 21	Gelistirme seviyesi
<b>20 V2</b>	2/3	2/3	2/3	2/3	2/3	2/3	2/3	14 / 21	Gelistirme seviyesi

Gerçeklenen örnek olay da 7 beceri türü için tek tek 0 ile 3 arası puanlama yapılmıştır. Master puanlar bunların toplamı olarak maksimum 21 olabilmektedir. 0 ile 7 puan arası temel seviye, 7 ile 14 arası geliştirme seviyesi ve 14 üzeri yeterli seviye olarak değerlendirilir.

Uzmanın yaptığı uygulama 21 puan üzerinden 21 puan almıştır. Diğer kullanıcılara bakıldığında en yüksek puan 18 dir. “Tablo 34.” de Değerlendirme aracı kullanılmadan önce ve sonrasındaki master puanlarına dayalı olarak beceri seviyelerinin durumu belirtilmiştir.

Tablo 34. Değerlendirme aracı kullanmadan öncesi ve sonrasındaki kullanıcı durumları

MASTER PUANLARA GÖRE DEĞERLENDİRME		
	1. UYGULAMA	2. UYGULAMA
TEMEL SEVİYE	9 kişi	4 kişi
GELİŞTİRME SEVİYESİ	10 kişi	8 kişi
YETERLİ SEVİYE	1 kişi	8 kişi

20 kullanıcı üzerinde yapılan örnekleme genel ve tekil durumlar incelendiğinde tüm kullanıcılar kendi seviyelerine göre ilerleme kaydetmiş ve becerilerini geliştirmiştir.



## VI. SONUÇLAR

Bu tez çalışması Makine Öğrenmesi tekniklerini kullanarak, kullanıcının Bilgi İşlemsel Düşünme becerilerini Blok Tabanlı Programlama araçlarında gerçekleştirilen performansa dayalı olarak ölçen bir değerlendirme aracıdır. Geliştirilen araç için öncelikle Öğrencinin Blok Tabanlı programlama ortamında sergilediği performansı, BİD becerilerinin 7 boyutuna göre değerlendiren bir veri seti seçilmiştir. Seçilen veri seti Basit Doğrusal Regresyon modeli ile eğitilmiştir.

Eğitilen model öğretmen ve öğrencilerin performanslarını hızlı ve rahat şekilde ölçebilecekleri çevrimiçi performans değerlendirme aracına dönüştürülmüştür. Araç Blok tabanlı programlama alanındaki kısıtlı ölçme araçlarına katkı sağlamıştır. Geliştirilen çevrimiçi performans değerlendirme aracı hem öğretmen hem de öğrencilere değerlendirme yapılması için yardımcı kaynak oluşturmaktadır.

Sonuçta, Blok Tabanlı Programlama dillerinde BİD becerilerinin 7 boyutunu Makine Öğrenmesi modeli ile değerlendirilmesi öğretmen ve öğrenciler için tarafsız ve şeffaf bir değerlendirme sunmuştur. Geliştirilen araç sayesinde Blok Tabanlı Programlama dillerinde BİD becerilerini ölçmekte sınırlı olan değerlendirme araçlarına alternatif bir araç geliştirilmiştir.

## VII. KAYNAKÇA

1. <http://www.robotics.stanford.edu/~ang/courses.html>, [Erişim Tarihi: Mayıs 2019]
2. J. L. Berral – Garcia. “Statistics When and How to Apply, Machine Learning and Deep Learning Techniques”, 20. Uluslararası Şeffaf Optik Ağlar Konferansı (ICTON),SS.1-4, 2018.
3. R. S. Sutton ve A. G. Barto, “Reinforcement learning: An introduction”, MIT Basın, Cambridge, Cilt. 1, 1998.
4. Autar Kaw, “Holistic Numerical Methods Institute”, <http://numericalmethods.eng.usf.edu/nbm>, Simple Linear Regression [Erişim Tarihi: Mayıs 2019]
5. O. Gençoğlu, T. Virtanen ve H. Huttunen “Recognition of acoustic events using deep neural networks” 22. Avrupa Sinyal İşleme Konferansı (EUSIPCO), 2014.
6. Tefvik çoban. “Makine Öğrenme Algoritmaları İle Web Siteleri Tıklamalarının Analizi” İstanbul Beykent üniversitesi, 2011.
7. S. B. Kotsiantis, “Use of machine learning techniques for educational proposes: a decision support system for forecasting students’ grades” Yapay Zeka İnceleme. Cilt. 37, Sayı. 4, SS. 331–344, 2012.
8. Jie Xu ; K.H. Moon ; M. V. D. Schaar. “Kooperatif tasarım sistemlerinde uyumsuzluk çözümünün desteklenmesi”, IEEE Sinyal İşlemede Seçilmiş Konular Dergisi, Cilt: 11 , Sayı: 5 , SS. 742 – 753, 2017.

9. H. Agrawal ve H. Mavani “Student Performance Prediction using Machine Learning”, Uluslararası Mühendislik Araştırma ve Teknoloji Dergisi (IJERT). Cilt.4, Sayı.3, SS.111-113, 2015.
10. “Identifying At-Risk Students Using Machine Learning Techniques: A Case Study with IS 100”, Uluslararası Makine Öğrenimi ve Bilişim Dergisi, Cilt.2, No.4, Sayı.5, SS.476 – 480, 2012.
11. M.Ciolacu, A.F. Tehrani, R. Beer ve H. Popp, “Education 4.0 — Fostering student's performance with machine learning methods”, 2017 IEEE 23. Uluslararası Elektronik Ambalajda Tasarım ve Teknoloji Sempozyumu (SIITME) SS.438-443, 2017.
12. D. Kelly ve B. Tangney, “First Aid for You': getting to know your learning style using machine learning” , SS. 1-3, 2005.
13. J. M. Wing, "Computational thinking", ACM'nin İletişimi konferansı, Cilt. 49, No.3, SS. 33-35, 2006.
14. V.Barr, ve C. Stephenson, “ Bringing computational thinking to K-12: What is involved and what is the role of the computer science education community?” ACM gedikler, 2(1), SS. 48–54, 2011.
15. J.Bilbao, O. García, C.Rebollar, E.Bravo ve C.Varela,” Skills, attitudes and concepts of the Computational Thinking”, Ekonomi ve Eğitim, ISBN: 978-1-61804-369-6.
16. J M.-León, G Robles, ve M Román -González “Computational Thinking Development Assessment Scores with Software Complexity Metrics.” IEEE Global Mühendislik Eğitim Konferansı (EDUCON), 978-1-4673-8633-3/16/, S.S.1040, 2016.

17. Scratch wiki <https://en.scratch-wiki.info/wiki/Blocks> [Eriřim Tarihi: Mayıs 2019]
18. <https://scratch.mit.edu/statistics/> [Eriřim Tarihi: Mayıs 2019]
19. İ. Őimřek, (2013), "Eđitimde Olęme Ve Deęerlendirme Kriterlerine Uygun Web Tabanlı Uzman Sınav Sistemi Geliřtirilmesi", Doktora Tezi İstanbul Üniversitesi,. SS. 25,2013.
20. J M.-León, G Robles ve M Román -González "Dr. Scratch: Automatic Analysis of Scratch Projects to Assess and Foster Computational Thinking." RED-Uzaktan Eđitim Dergisi, SS:: 46, 2015.
21. Moreno J. León ve G. Robles. "Analyze your Scratch projects with Dr. Scratch and assess your computational thinking skills." Scratch konferansı (SS. 12-15), 2015.
22. Moreno, J. a. (2014). "Automatic Detection of Bad Programming Habits in Scratch: A Preliminary Study." (IEEE) Eđitim Konferansında Sınırlar, 2014.
23. <http://www.medicine.ankara.edu.tr/2016/09/04/olcme-degerlendirme-sureci-nedir/> [Eriřim Tarihi: Mayıs 2019]
24. Colaboratory <https://research.google.com/colaboratory/faq.html> [Eriřim Tarihi: Mayıs 2019]
25. Tensorflow, <https://www.tensorflow.org/about> [Eriřim Tarihi: Mayıs 2019]
26. Tensorflow.js, <https://www.tensorflow.org/js/tutorials> [Eriřim Tarihi: Mayıs 2019]
27. E. Aivaloglou, F. Hermans, J.Moreno-Leon, ve G. Robles, "A dataset of scratch programs: scraped, shaped and scored," 2017 IEEE / ACM 14. Uluslararası Madencilik Yazılım Depoları Konferansı (MSR). SS.511-514, 2017.


28. Github.com, 'Derin Öğrenme Kursu', 2018. [Online]. Mevcut: <https://github.com/enggen/Deep-Learning-Coursera> [Erişim: 1 Ocak 2019]
29. Github.com, 'Doğrusal-Regression.ipynb', 2018. [Online]. Mevcut: <https://github.com/jakevdp/PythonDataScienceHandbook/blob/master/notebooks/05.06-Linear-Regression.ipynb> [Erişim: 1 Ocak 2019]




## VIII. EKLER

### EK-A

#### DATASET OF SCRATCH VERİSETİ KULLANIM İZİNİ

**Elif Karakaş** 1 gün önce

Merhaba, yüksek lisans tezi için "Kazı Kazanç, Şekli ve Puanlanmış Bir Veri Kümesi" veri kümesini kullanmak istiyorum. Ne yapmalıyım?

**Jesús Moreno-León** sana 1 gün önce

Merhaba,

Veri kümesinden dosyaları buradan indirebilirsiniz:  
<https://github.com/TUdelftScratchLab/ScratchDataset>

Umarım yararlıdır! Bitirdiğiniz zaman teziniz hakkında bilgi veriniz.

En iyisi,  
Jesús.

Cevapla Okundu olarak işaretle Arşiv sohbet

## EK-B

### Blok tipi çeviri sözlüğü

Dönüştürülmüş değeri	Blok türü	Blok türü	Dönüştürülmüş değeri
0	%	70	mousePressed
1	&	71	mouseX
2	*	72	mouseY
3	+	73	nextCostume
4	-	74	nextScene
5	<	75	not
6	=	76	noteOn:duration:elapsed:from:
7	>	77	penColor:
8	√	78	penSize:
9	answer	79	playDrum
10	append:toList:	80	playSound:
11	backgroundIndex	81	pointTowards:
12	bounceOffEdge	82	putPenDown
13	broadcast:	83	putPenUp
14	call	84	randomFrom:to:
15	changeGraphicEffect:by:	85	readVariable
16	changePenHueBy:	86	rest:elapsed:from:
17	changePenShadeBy:	87	rounded
18	changePenSizeBy:	88	say:
19	changeSizeBy:	89	say:duration:elapsed:from:
20	changeTempoBy:	90	scale
21	changeVar:by:	91	sceneName
22	changeVolumeBy:	92	senseVideoMotion
23	changeXposBy:	93	setGraphicEffect:to:
24	changeYposBy:	94	setLine:ofList:to:
25	clearPenTrails	95	setPenHueTo:
26	color:sees:	96	setPenShadeTo:
27	comeToFront	97	setRotationStyle
28	computeFunction:of:	98	setSizeTo:
29	concatenate:with:	99	setTempoTo:
30	contentsOfList:	100	setVar:to:
31	costumeIndex	101	setVideoState
32	createCloneOf	102	setVideoTransparency
33	deleteClone	103	setVolumeTo:
34	deleteLine:ofList:	104	show
35	distanceTo:	105	showList:
36	doAsk	106	showVariable:
37	doBroadcastAndWait	107	soundLevel
38	doForever	108	stampCostume

39	dolf	109	startScene
40	dolfElse	110	startSceneAndWait
41	doPlaySoundAndWait	111	stopAllSounds
42	doRepeat	112	stopScripts
43	doUntil	113	stringLength:
44	doWaitUntil	114	tempo
45	drum:duration:elapsed:from:	115	think:
46	filterReset	116	think:duration:elapsed:from:
47	forward:	117	timeAndDate
48	getAttribute:of:	118	timer
49	getLine:ofList:	119	timerReset
50	getParam	120	timestamp
51	getUserId	121	touching:
52	getUserName	122	touchingColor:
53	glideSecs:toX:y:elapsed:from:	123	turnLeft:
54	goBackByLayers:	124	turnRight:
55	gotoSpriteOrMouse:	125	volume
56	gotoX:y:	126	wait:elapsed:from:
57	heading	127	whenClicked
58	heading:	128	whenCloned
59	hide	129	whenGreenFlag
60	hideList:	130	whenIReceive
61	hideVariable:	131	whenKeyPressed
62	insert:at:ofList:	132	whenSceneStarts
63	instrument:	133	whenSensorGreaterThan
64	keyPressed:	134	xpos
65	letter:of:	135	xpos:
66	lineCountOfList:	136	ypos
67	list:contains:	137	ypos:
68	lookLike:	138	
69	midInstrument:		



## EK-C

Model eğitilmesi yapılmadan önce, veri setlerinin Coloborotory'e alınması aşamasında kullanılan kod komutları;

```
//CSV dosyalarını okuma
deger-verisi = pd.read-csv('deger-clearCSV', usecols = ['p-ID', 'block-type'])
deger-verisi.columns = ['p-ID', 'block-type']
//Sadece kullanılacak olan proje numarası ve blok tipi alanını belirleme

sutun-listesi = sorted(deger-verisi['block-type'].unique())
cevriyen-sutun = dict(enumerate(sutun-listesi))
ceriyen-sutun-rehber = {'-'+(value): key for key, value in cevriyen-sutun.items()}
ceriyen-sutun-rehber['p-ID'] = 'p-ID'
//Meta verilerinden proje numarası ve 7 alandan birini seçme

meta-veri = pd.read-csv('meta-clearCSV')
col-list = ['p-ID', 'Abstraction']
meta-veri = meta-veri[col-list]
meta-veri.describe()
// Proje ve Meta veri setlerini birleştirme

deger-data = pd.merge(deger-data, meta-veri, on='p-ID')
```

## EK-D

Derin Sinir Ağları yöntemiyle model eğitilme kodları

```

ozellik_satiri_karma = tf.feature_column.categorical_column_with_hash_bucket(
    key = "block_type",
    hash_bucket_size = 139)
karma_feature_column = tf.feature_column.indicator_column(ozellik_satiri_karma)
columns = [ karma_feature_column ]
blok_tipi = birlesim1['block_type'].tolist()
p_ID = birlesim1['p_ID'].tolist()
features = []
features.append(blok_tipi)
items = len(features[0])
labels = [round(x) for x in birlesim1['Abstraction']] #etike
etiketi_kodlamasi = True
    print ("Etiketlerin boyutu: ", len(labels))
    print ("Özelliklerin boyutu: ", len(features[0]))
from sklearn.preprocessing import LabelBinarizer
if not etiketi_kodlamasi:
    kodla = LabelBinarizer()
    kodla.fit(labels)
    labels = kodla.transform(labels)
    labels = labels.astype(np.float32)
    etiketi_kodlamasi = True
print ("etiket sayısı: ", len(labels))
print ("özellik sayısı: ", len(features[0]))
    tarin_sayisi = int(0.8*items)
    feature_testi = [x[0:-tarin_sayisi] for x in features]
    features = [x[0:tarin_sayisi] for x in features]
    labels_test = labels[0:-tarin_sayisi]
    labels = labels[0:tarin_sayisi]
    p_ID_test = p_ID[0:-tarin_sayisi]
    p_ID = p_ID[0:tarin_sayisi]
test_giris = tf.estimator.inputs.numpy_input_fn(
    x={'block_type': np.array(feature_testi[0])},
    y=np.array(labels_test),
    num_epochs=1,
    shuffle=False)
LT_g = labels_test
ID_g = p_ID_test
num_hidden_units = [512, 32]
klas_sayisi = 4
model = tf.estimator.DNNClassifier(feature_columns=columns,hidden_units=[512,
256],optimizer=lambda:

```

```
tf.train.AdamOptimizer(learning_rate=tf.train.exponential_decay(
learning_rate=0.1,global_step=tf.train.get_global_step(),
decay_steps=10000,decay_rate=0.96)),activation_fn=tf.nn.relu,n_classes=klas_sayisi
, model_dir="./kontrolnoktasi/")
    model.train(input_fn=train_giris, steps=20000)
    result = model.evaluate(input_fn=test_giris)
print("Sınıflama doğruluğu: {0:.2%}".format(result["accuracy"]))
    predictions = model.predict(input_fn=giris_tahmini)
    cls = [p['classes'] for p in predictions]
    cls_pred = np.array(cls, dtype='int').squeeze()
results = pd.DataFrame()
results['p_ID']=ID_g
results['cls_pred'] = cls_pred
results['LT_g'] = LT_g
print(results.groupby(['p_ID', 'cls_pred', 'LT_g']).count())
```

## EK-E

Proje veri setindeki blok tipinin sayısal verilere dönüştürme aşamasında kullanılan kodlar;

```
// Blok tipi verilerinin sayısal ifadelere çevirme
sutun-listesi = sorted(deger-verisi["block-type"].unique())
cevrilen-sutun = dict(enumerate(sutun-listesi))
cerilen-sutun-rehber = {'-'+(value): key for key, value in cevrilen-sutun.items()}
cerilen-sutun-rehber['p-ID'] = 'p-ID'
// veriler küçük parçalara bölünerek işlem yapılıyor
n-rows = deger-verisi.shape[0]
datasets = []
old-dataset = pd.DataFrame()
for i in range(1,101):
    dataset = deger-verisi.iloc[int(0.01*(i-1)*n-rows):int(0.01*(i)*n-rows), :]
    dataset = pd.get_dummies(dataset, columns = ["block-type"], prefix = [""])
    dataset.columns = [cerilen-sutun-rehber[x] for x in dataset.columns]
    dataset = dataset.groupby(['p-ID']).sum().reset_index()
    datasets.append(dataset)
deger-data = pd.concat(datasets)
del deger-verisi
// Verilerin Turşuya alınması
tursu-dosyasi = 'deger-abstraction.pickle'
if not os.path.isfile(tursu-dosyasi):
    print('Tursu kaydediliyor...')
    try:
        with open('deger-abstraction.pckle', 'wb') as pfile:
            pckle.dump(
                {
                    'deger-abstraction': deger-data
                },
                pfile, pckle.HIGHEST_PROTOCOL)
    except Exception as e:
        raise
    print('Abstraction turşuya atıldı')
tursu-dosyasi = 'deger-abstraction.pckle'
with open(tursu-dosyasi, 'rb') as k:
    tursu-veri = pckle.load(k)
    deger-data = tursu-veri['deger-abstraction']
    del tursu-veri
print('VERİLER VE MODÜLLER EKLENDİ.')
// kategorik değerlere göre Abstraction meta veri alanın için etiket benzeri tahmin vektörü
labels = df[Abstraction]
p-ID = df['p-ID']
```

```
df = df.drop(columns = ['Abstraction, 'p-ID'])
from sklearn.preprocessing import LabelBinarizer
is-labels-encd = False
if not is-labels-encd
    encodr = LabelBinarizer()
    encodr.fit(labels)
    labels = encodr.transfrm(labels)
    is-labels-encd = True
labels.shape
// veri setinin eğitim, doğrulama ve test için ayrılması
// eğitim ve test seti ayırma
egitim-seti = int(0.8*deger-data.shape[0])
test-seti = deger-data.shape[0] - egitim-seti
test-deger-data = deger-data[0:-egitim-seti]
deger-data = deger-data[0:egitim-seti]
test-etiketi = labels[0:-egitim-seti]
tekli-test-deger-data = np.reshape(test-deger-data.iloc[0], (1, len(test-deger-
data.iloc[0])))
tekli-test-etiketi = np.reshape(test-etiketi [0], (1, len(test-labels[0])))
labels = labels[0:egitim-seti]
// eğitim setini bölme
egitim-seti = int(0.9*deger-data.shape[0])
valid-deger-data = deger-data[0:-egitim-seti]
deger-data = deger-data[0:egitim-seti]
valid-etiketi = labels[0:-egitim-seti]
labels = labels[0:egitim-seti]
```

## EK-F

Abstraction alanını tahmin için kullanılan Model Eğitimi kodları;

```
// kategorik değerlere göre Abstraction meta veri alanın için etiket benzeri
tahmin vektörü
labels = df[Abstraction]
p-ID = df['p-ID']
df = df.drop(columns = ['Abstraction, 'p-ID'])
with tf.Session() as session:
    tf.global-variables-initializer().run()
    print("Başladı")
    for step in range( lp-num):
        offset = np.random.randint(0, labels.shape[0] - batch-size - 1)
        batch-data = deger-data.iloc[offset:(offset + batch-size ), :]
        batch-labels = labels[offset:(offset + batch-size ), :]
        feed-dict = {tf-train-dataset : batch-data,
tf-tran-labels : batch-labels}
        -, l, predictons = session.run([optimizr, loss, train-predicton],
Feed-dict =feed-dict )
    if (step % 500 == 0):
        print("Asamada-yığın-kaybı {0}: {1}".format(step, l))
        print("Yığın-dogrulugu: {:.1f}%".format(
accuracy(predictons, batch-labels)))
        print("Gecerlilik-dogrulugu : {:.1f}%".format(
accuracy(valid-predicton.eval(), valid-labels)))
        print("\nTest-doğruluğu : {:.1f}%".format(
accuracy(test-predicton.eval(), test-labels)))
        tf-Abstraction-predicton = tf.identity(tf-Abstraction-predicton, name =
"tf-Abstraction-predicton")
        print("\nTest-doğruluğu : {:.1f}%".format(
accuracy(tf-Abstraction-predicton.eval(), tekli-test-etiketi)))
        tf.saved-model.simple-save(session, "./Abstraction", inputs = {'batch-
data': tf-Abstraction-dataset, 'tf-Abstraction-weights': weights, 'tf-Abstraction-
biases': biases}, outputs = {'tf-Abstraction-predicton': tf-Abstraction-
predicton})
        np.savetxt("W-AbstractionCSV", (session.run(weights)), delimiter = ",")
        np.savetxt("b-AbstractionCSV", session.run(biases), delimiter = ",")
```

## EK-G

Parallelism alanını tahmin için kullanılan Model Eğitimi kodları;

```
// kategorik değerlere göre Parallelism meta veri alanının için etiket benzeri
tahmin vektörü
labels = df[Parallelism]
p-ID = df['p-ID']
df = df.drop(columns = ['Parallelism, 'p-ID'])
with tf.Session() as session:
    tf.global-variables-initializer().run()
    print("Başladı")
    for step in range( lp-num):
        offset = np.random.randint(0, labels.shape[0] – batch-size - 1)
        batch-data = deger-data.iloc[offset:(offset + batch-size ), :]
        batch-labels = labels[offset:(offset + batch-size ), :]
        feed-dict = {tf-train-dataset : batch-data,
tf-tran-labels : batch-labels}
        -, l, predictons = session.run([optimizr, loss, train-predicton],
Feed-dict =feed-dict )
    if (step % 500 == 0):
        print("Asamada-yığın-kaybı {0}: {1}".format(step, l))
        print("Yığın-dogrulugu: {:.1f}%".format(
accuracy(predictons, batch-labels)))
        print("Gecerlilik-dogrulugu : {:.1f}%".format(
accuracy(valid-predicton.eval(), valid-labels)))
        print("\nTest-doğruluğu : {:.1f}%".format(
accuracy(test-predicton.eval(), test-labels)))
        tf-Parallelism-predicton = tf.identity(tf-Parallelism-predicton, name =
"tf-Parallelism-predicton")
        print("\nTest-doğruluğu : {:.1f}%".format(
accuracy(tf-Parallelism-predicton.eval(), tekli-test-etiketi)))
        tf.saved-model.simple-save(session, "./Parallelism", inputs = {'batch-data':
tf-Parallelism-dataset, 'tf-Parallelism-weights': weights, 'tf-Parallelism-biases':
biases}, outputs = {'tf-Parallelism-predicton': tf-Parallelism-predicton})
        np.savetxt("W-ParallelismCSV", (session.run(weights)), delimiter = ",")
        np.savetxt("b-ParalleismCSV", session.run(biases), delimiter = ",")
```

## EK-H

Logic alanını tahmin için kullanılan Model Eğitimi kodları;

```
// kategorik değerlere göre Logic meta veri alanın için etiket benzeri tahmin
vektörü
labels = df[Logic]
p-ID = df['p-ID']
df = df.drop(columns = ['Logic, 'p-ID'])
with tf.Session() as session:
    tf.global-variables-initializer().run()
    print("Başladı")
    for step in range( lp-num):
        offset = np.random.randint(0, labels.shape[0] – batch-size - 1)
        batch-data = deger-data.iloc[offset:(offset + batch-size ), :]
        batch-labels = labels[offset:(offset + batch-size ), :]
        feed-dict = {tf-train-dataset : batch-data,
tf-tran-labels : batch-labels}
        -, l, predictons = session.run([optimizr, loss, train-predicton],
Feed-dict =feed-dict )
    if (step % 500 == 0):
        print("Asamada-yığın-kaybı {0}: {1}".format(step, l))
        print("Yığın-dogrulugu: {:.1f}%".format(
accuracy(predictons, batch-labels)))
        print("Gecerlilik-dogrulugu : {:.1f}%".format(accuracy(valid-
predicton.eval(), valid-labels)))
        print("\nTest-doğruluğu : {:.1f}%".format(
accuracy(test-predicton.eval(), test-labels)))
        tf-Logic-predicton = tf.identity(tf-Logic-predicton, name =
"tf-Logic-predicton")
        print("\nTest-doğruluğu : {:.1f}%".format(
accuracy(tf-Logic-predicton.eval(), tekli-test-etiketi)))
        tf.saved-model.simple-save(session, "./Logic", inputs = {'batch-data': tf-
Logic-dataset, 'tf-Logic-weights': weights, 'tf-Logic-biases': biases}, outputs =
{'tf-Logic-predicton': tf-Logic-predicton})
        np.savetxt("W-LogicCSV", (session.run(weights)), delimiter = ",")
        np.savetxt("b-LogicCSV", session.run(biases), delimiter = ",")
```



## EK-I

Synchronization alanını tahmin için kullanılan Model Eğitimi kodları;

```
// kategorik değerlere göre Synchronization meta veri alanını için etiket
benzeri tahmin vektörü
labels = df[Synchronization]
p-ID = df['p-ID']
df = df.drop(columns = ['Synchronization, 'p-ID'])
with tf.Session() as session:
    tf.global_variables_initializer().run()
    print("Başladı")
    for step in range( lp-num):
        offset = np.random.randint(0, labels.shape[0] - batch-size - 1)
        batch-data = deger-data.iloc[offset:(offset + batch-size ), :]
        batch-labels = labels[offset:(offset + batch-size ), :]
        feed-dict = {tf-train-dataset : batch-data,
                    tf-tran-labels : batch-labels}
        -, l, predictons = session.run([optimizr, loss, train-predicton],
        Feed-dict =feed-dict )
    if (step % 500 == 0):
        print("Asamada-yığın-kaybı {0}: {1}".format(step, l))
        print("Yığın-dogrulugu: {:.1f}%".format(
            accuracy(predictons, batch-labels)))
        print("Gecerlilik-dogrulugu : {:.1f}%".format(
            accuracy(valid-predicton.eval(), valid-labels)))
        print("\nTest-dogruluğu : {:.1f}%".format(
            accuracy(test-predicton.eval(), test-labels)))
        tf-Synchronization-predicton = tf.identity(tf-Synchronization-predicton,
        name = "tf-Synchronization-predicton")
        print("\nTest-dogruluğu : {:.1f}%".format(
            accuracy(tf-Synchronization-predicton.eval(), tekli-test-etiketi)))
        tf.saved-model.simple-save(session, "./Synchronization", inputs = {'batch-
        data': tf-Synchronization-dataset, 'tf-Synchronization-weights': weights, 'tf-
        Synchronization-biases': biases}, outputs = {'tf-Synchronization-predicton':
        tf-Synchronization-predicton})
        np.savetxt("W-SynchronizationCSV", (session.run(weights)), delimiter =
        ",")
        np.savetxt("b-SynchronizationCSV", session.run(biases), delimiter = ",")
```

## EK-J

FlowControl alanını tahmin için kullanılan Model Eğitimi kodları;

```
// kategorik değerlere göre FlowControl meta veri alanın için etiket benzeri
tahmin vektörü
labels = df[FlowControl]
p-ID = df['p-ID']
df = df.drop(columns = ['FlowControl, 'p-ID'])
with tf.Session() as session:
    tf.global_variables_initializer().run()
    print("Başladı")
    for step in range( lp-num):
        offset = np.random.randint(0, labels.shape[0] – batch-size - 1)
        batch-data = deger-data.iloc[offset:(offset + batch-size ), :]
        batch-labels = labels[offset:(offset + batch-size ), :]
        feed-dict = {tf-train-dataset : batch-data,
                    tf-tran-labels : batch-labels}
        -, l, predictions = session.run([optimizir, loss, train-predicton],
        Feed-dict=feed-dict )
    if (step % 500 == 0):
        print("Asamada-yığın-kaybı {0}: {1}".format(step, l))
        print("Yığın-dogrulugu: {:.1f}%".format(
            accuracy(predictions, batch-labels)))
        print("Gecerlilik-dogrulugu : {:.1f}%".format(
            accuracy(valid-predicton.eval(), valid-labels)))
        print("\nTest-dogruluđu : {:.1f}%".format(
            accuracy(test-predicton.eval(), test-labels)))
        tf-FlowControl-predicton = tf.identity(tf-FlowControl-predicton, name =
        "tf-FlowControl-predicton")
        print("\nTest-dogruluđu : {:.1f}%".format(
            accuracy(tf-FlowControl-predicton.eval(), tekli-test-etiketi)))
        tf.saved-model.simple-save(session, "./FlowControl", inputs = {'batch-
        data': tf-FlowControl-dataset, 'tf-FlowControl-weights': weights, 'tf-
        FlowControl-biases': biases}, outputs = {'tf-FlowControl-predicton': tf-
        FlowControl-predicton})
        np.savetxt("W-FlowControlCSV", (session.run(weights)), delimiter = ",")
        np.savetxt("b-FlowControlCSV", session.run(biases), delimiter = ",")
```

## EK-K

UserInteractivity alanını tahmin için kullanılan Model Eğitimi kodları;

```
// kategorik değerlere göre UserInteractivity meta veri alanın için etiket benzeri
tahmin vektörü
labels = df[UserInteractivity]
p-ID = df['p-ID']
df = df.drop(columns = ['UserInteractivity', 'p-ID'])
with tf.Session() as session:
    tf.global_variables_initializer().run()
    print("Başladı")
    for step in range( lp-num):
        offset = np.random.randint(0, labels.shape[0] – batch-size - 1)
        batch-data = deger-data.iloc[offset:(offset + batch-size ), :]
        batch-labels = labels[offset:(offset + batch-size ), :]
        feed-dict = {tf-train-dataset : batch-data,
                    tf-tran-labels : batch-labels}
        -, l, predictions = session.run([optimizir, loss, train-predicton],
        Feed-dict=feed-dict )
    if (step % 500 == 0):
        print("Asamada-yığın-kaybı {0}: {1}".format(step, l))
        print("Yığın-dogrulugu: {:.1f}%".format(
            accuracy(predictions, batch-labels)))
        print("Gecerlilik-dogrulugu : {:.1f}%".format(
            accuracy(valid-predicton.eval(), valid-labels)))
        print("\nTest-doğruluğu : {:.1f}%".format(
            accuracy(test-predicton.eval(), test-labels)))
        tf-UserInteractivity-predicton = tf.identity(tf-UserInteractivity-predicton,
        name = "tf-UserInteractivity-predicton")
        print("\nTest-doğruluğu : {:.1f}%".format(
            accuracy(tf-UserInteractivity-predicton.eval(), tekli-test-etiketi)))
        tf.saved-model.simple-save(session, "./UserInteractivity", inputs = {"batch-
        data": tf-UserInteractivity-dataset, 'tf-UserInteractivity-weights': weights, 'tf-
        UserInteractivity-biases': biases}, outputs = {'tf-UserInteractivity-predicton': tf-
        UserInteractivity-predicton})
        np.savetxt("W-UserInteractivityCSV", (session.run(weights)), delimiter =
        ",")
        np.savetxt("b-UserInteractivityCSV", session.run(biases), delimiter = ",")
```

## EK-L

DataRepresentation alanını tahmin için kullanılan Model Eğitimi kodları;

```
// kategorik değerlere göre DataRepresentation meta veri alanın için etiket
benzeri tahmin vektörü
labels = df[DataRepresentation]
p-ID = df['p-ID']
df = df.drop(columns = ['DataRepresentation, 'p-ID'])
with tf.Session() as session:
    tf.global_variables_initializer().run()
    print("Başladı")
    for step in range( lp-num):
        offset = np.random.randint(0, labels.shape[0] – batch-size - 1)
        batch-data = deger-data.iloc[offset:(offset + batch-size ), :]
        batch-labels = labels[offset:(offset + batch-size ), :]
        feed-dict = {tf-train-dataset : batch-data,
                    tf-tran-labels : batch-labels}
        -, l, predictons = session.run([optimizr, loss, train-predicton],
        Feed-dict=feed-dict )
    if (step % 500 == 0):
        print("Asamada-yığın-kaybı {0}: {1}".format(step, l))
        print("Yığın-dogrulugu: {:.1f}%".format(
            accuracy(predictons, batch-labels)))
        print("Gecerlilik-dogrulugu : {:.1f}%".format(
            accuracy(valid-predicton.eval(), valid-labels)))
        print("\nTest-doğruluğu : {:.1f}%".format(
            accuracy(test-predicton.eval(), test-labels)))
        tf-DataRepresentation-predicton = tf.identity(tf-DataRepresentation-
        predicton, name = "tf-DataRepresentation-predicton")
        print("\nTest-doğruluğu : {:.1f}%".format(
            accuracy(tf-DataRepresentation-predicton.eval(), tekli-test-etiketi)))
        tf.saved-model.simple-save(session, "./DataRepresentation", inputs =
        {'batch-data': tf-DataRepresentation-dataset, 'tf-DataRepresentation-weights':
        weights, 'tf-DataRepresentation-biases': biases}, outputs = {'tf-
        DataRepresentation-predicton': tf-DataRepresentation-predicton})
        np.savetxt("W-DataRepresentationCSV", (session.run(weights)), delimiter
        = ",")
        np.savetxt("b-DataRepresentationCSV", session.run(biases), delimiter =
        ",")
```

## EK-M

Makine Öğrenmesi Modeli eğitilirken kullanılan 139 blok tipi verisinin, Scratch uygulamasındaki karşılığı

SIRA NO	SCRATCH PROGRAMINDAKİ ADI	BLOK TİPİ ALANINDAKİ ADI	SIRA NO	SCRATCH PROGRAMINDAKİ ADI	BLOK TİPİ ALANINDAKİ ADI
1	Adım Git (ileri git)	forward:	71	Listeye eleman ekleme	append:toList:
2	Derece dön (Sola dön)	turnLeft:	72	Listenin elemanını silme	deleteLine:ofList:
3	Derece dön (Sağa dön)	turnRight:	73	Listenin belirli sırasına eleman ekleme	insert:at:ofList:
4	Yönüne Dön	heading:	74	Listenin elemanını değiştir	setLine:ofList:to:
5	Yönü	heading	75	Listenin kaçınıcı elemanı	getLine:ofList:
6	Fare okuna doğru dön (Yönüne Doğru dön)	pointTowards:	76	Listenin Uzunluğu	lineCountOfList:
7	Saniyede X ve Y noktasına git	gotoX:y:	77	Listesinde Şunu İçeriyor mu? (Contains)	list:contains:
8	Fare okuna git	gotoSpriteOrMouse:	78	Listeyi Göster	showList:
9	Saniyede X ve Y ye Süzül	glideSecs:toX:y:elapsed:from:	79	Listeyi Gizle	hideList:
10	X kordinatında konumunu artır	changeXposBy:	80	Başlat tuşuna basılınca (Tıklanınca)	whenGreenFlag
11	Y kordinatında konumunu artır	changeYposBy:	81	Tuşa basınca başlat	whenKeyPressed
12	X konumunu değiştir	xpos:	82	Kuklaya Tıklanınca başlat	whenClicked
13	Y konumu değiştir	ypos:	83	Ses şiddeti büyük olunca başlat	whenSensorGreater Than
14	X konumunu	xpos	84	Dekor olunca başlat	whenSceneStarts
15	Y konumunu	ypos	85	Haberi Geline (whenIReceive)	whenIReceive
16	Ekrandan Çıktıysa Sek	bounceOffEdge	86	Haberini Sal (yayınla / broadcast )	broadcast:
17	Kuklanın Şekli değişsin	setRotationStyle	87	Haberini Sal ve bekle (yayınla ve bekle /doBroadcastAndWait)	doBroadcastAnd Wait
18	Hello de 2 saniye boyunca (ekrana yaz 2 saniye boyunca)	say:duration:elapsed:from:	88	Saniye bekle (wait)	wait:elapsed:from:

19	Hello de	say:	89	tekrarla (doRepeat)	doRepeat
20	Diye düşün iki saniye boyunca (süre boyunca ekrana yaz)	think:duration:elapsed:from:	90	Sürekli Tekrarla (doForever)	doForever
21	Diye düşün (ekrana yaz)	think:	91	Eğer .. İse(IF )	doIf
22	Görün	show	92	Eğer ... İse ... Değilse (If Else)	doIfElse
23	Gizlen	hide	93	Olana kadar bekle (doWaitUntil)	doWaitUntil
24	Arkaplan numarası	backgroundIndex	94	Olana kadar tekrarla (doUntil)	doUntil
25	Kılığına geç	lookLike:	95	Hepsini durdur( tüm döngüleri durdur)	stopScripts
26	Sonraki Kılık	nextCostume	96	İkiz olarak başlatıldığında ( Klon oluşturulduğunda)	whenCloned
27	Dekoruna geç ( Sahneye geç)	startScene	97	İkizini yarat( Klonunu yarat)	createCloneOf
28	Dekoruna geç ve bekle ( Sahneye geç ve bekle )	startSceneAndWait	98	İkizini Sil ( Klonunu Sil)	deleteClone
29	Sonraki dekor	nextScene	99	Fareye yada kenra değdi mi?	touching:
30	Renk etkisini arttır	changeGraphicEffect:by:	100	rengine değdi mi?	touchingColor:
31	Renk etkisi sıfır olsun	setGraphicEffect:to:	101	Rengi rengine değdi mi?	color:sees:
32	Görsel Etiketleri Temizle	filterReset	102	Fareye olan mesafe	distanceTo:
33	Birim büyüt	changeSizeBy:	103	Diye sor ve bekle	doAsk
34	Büyüklüğü %100 yap	setSizeTo:	104	Yanıt	answer
35	Üste çık (kuklayı sahnede üste çıkar)	comeToFront	105	Tuşa basılı mı?	keyPressed:
36	Alet olarak ayarla(ses)	midiInstrument:	106	Fareye basılı mı?	mousePressed
37	Katman alta in	goBackByLayers:	107	Farenin x si	mouseX
38	Kılık no	costumeIndex	108	Farenin y si	mouseY
39	Dekorun adı	sceneName	109	Ses şiddeti	soundLevel
40	Büyüklük (kuklanın büyüklüğü)	scale	110	Video Üzerindeki Hareket	senseVideoMotion
41	Sesini çal	playSound:	111	Videoyu Aç - Kapat	setVideoState

42	Sesini bitene kadar çal	doPlaySoundAnd Wait	112	Video Saydamlığını Değiştir	setVideo Transparency
43	Tüm sesleri durdur	stopAllSounds	113	Süre ölçer	timer
44	Davulunu vuruş çal	playDrum	114	Süre Ölçeri sıfırla	timerReset
45	Vuruş sus (sesi sustur)	rest:elapsed:from :	115	Değerini Ata	getAttribute:of
46	Notasını Vuruş çal(Notanın vuruşu )	noteOn:duration: elapsed: from:	116	Şimdiki Tarih (Yıl Ay Gün Saat)	timeAndDate
47	Çalgıyı seç (yap)	instrument:	117	Geçensüre	timestamp
48	Sesi belirli birim yükselt	changeVolumeBy:	118	Kullanıcı Kimliği (kullanıcı ID)	getUserId
49	Ses şiddetini % üzerindenayarla	setVolumeTo:	119	Kullanıcı Adı	getUserName
50	Ses şiddetini arttır	volume	120	Boyunca davulu çal	drum:duration :elapsed:from:
51	Tempoyu arttır	changeTempoBy:	121	Toplama	+
52	Tempo dk da ses vuruş sayısı	setTempoTo:	122	Çıkarma	-
53	tempo	tempo	123	Çarpma	*
54	Temizle	clearPenTrails	124	Bölme	∕
55	İz Bırak	stampCostume	125	Rastgele Sayı Tut(Random Number)	randomFrom: to:
56	Kalemi bastır	putPenDown	126	Küçüktür	<
57	Kalem kaldır	putPenUp	127	Eşittir	=
58	Kalem Rengi	penColor:	128	Büyüktür	>
59	Kalem Tonunu Arttır	changePenHueBy :	129	Ve	&
60	Kalem Tonu	setPenHueTo:	130	Veya	
61	Kalem Tonunu Arttır	changePenShade By:	131	Değil	not
62	Kalem tonunu yap	setPenShadeTo:	132	İki kelimeyi birleştir	concatenate: with:
63	Kalem Kalınlığını Arttır	changePenSizeBy:	133	Kelimenin kaçınıcı harfi	letter:of:
64	Kalem Kalınlığını değiştir	penSize:	134	Kelime uzunluğu	stringLength:
65	Değişkeni oku	readVariable	135	Mod değerini al	%
66	Değişkeni değiştir	setVar:to:	136	Değeri Yuvarla	rounded
67	Değişkeni Arttır	changeVar:by:	137	İşlev hesapla(karekök mutlak değer aşağı veya yukarı yuvarla - log - sin - cos - tan -cos)	computeFunction:of:

68	Değişkeni Göster	showVariable:	138	Bir özel Taş oluştur (Fonksyon tanımla)	getParam
69	Değişkeni gizle	hideVariable:	139	Özel taşı kullan (fonksyonu çağır)	call
70	Listeyi oku	contentsOfList:			





## IX. ÖZGEÇMİŞ

Elif KARAKAŞ, 2013 yılında Doğu Akdeniz Üniversitesi Bilgisayar ve Öğretim Teknolojileri Öğretmenliği bölümünden mezun olmuştur. 2016 yılında İstanbul Okan Üniversitesi tezli yüksek lisans eğitimine başlamıştır.

İş hayatı, 2013 ile 2016 yılları arasında özel bir firmada yazılım destek biriminde kullanıcı desteği vermiştir. 2016 dan bu yana ilk ve ortaokul seviyesindeki öğrencilere Robotik ve Kodlama eğitimi vermektedir.