TÜRKİYE CUMHURİYETİ
YILDIZ TEKNİK ÜNİVERSİTESİ
BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ

VARYASYONEL OTOKODLAYICI MODELLER İLE EŞ/BENZER ANLAMLI CÜMLE ÜRETİMİ

19011040 — Engin MEMİŞ
20011040 — Elif Sena YILMAZ

BİLGİSAYAR PROJESİ

Danışman
Prof. Dr. Mehmet Fatih AMASYALI

Haziran, 2023

TEŞEKKÜR

Bu süreçte deneyim ve bilgilerini bizden esirgemeyen, değerli zamanını bize ayırarak her zaman bizimle iletişim içerisinde bulunan, bu proje süresince bize yol göstererek yardımda bulunan ve bu sayede projemizi başarılı bir şekilde bitirmemizi mümkün kılan kıymetli danışman hocamız Prof. Dr. Mehmet Fatih AMASYALI'ya teşekkürlerimizi sunarız.

Engin MEMİŞ
Elif Sena YILMAZ

İÇİNDEKİLER

SİMGE LİSTESİ

K	1000
₺	Türk Lirası
%	Yüzde İşareti

KISALTMA LİSTESİ

BERT	Bidirectional Encoder Representations from Transformers
CPU	Central Process Unit
GB	Gigabyte
GPU	Graphics Processing Unit
MB	Megabyte
MLM	Masked Language Modeling
NLP	Natural Language Processing
RAM	Random Access Memory
RNN	Recurrent Neural Network
RNNLM	Recurrent Neural Network Language Modeling
SGD	Stochastic Gradient Descent
SIVAE	Syntax-Infused Variational Autoencoder
SVM	Support Vector Machine
VAE	Variational Autoencoder

ŞEKİL LİSTESİ

TABLO LİSTESİ

ÖZET

VARYASYONEL OTOKODLAYICI MODELLER İLE EŞ/BENZER ANLAMLI CÜMLE ÜRETİMİ

Engin MEMİŞ
Elif Sena YILMAZ

Bilgisayar Mühendisliği Bölümü
Bilgisayar Projesi

Danışman: Prof. Dr. Mehmet Fatih AMASYALI

Veri, günümüzde büyük bir öneme sahip hale gelmiştir ve doğal dil işleme alanında kullanılan veri setlerinin çoğunluğu genellikle İngilizce dilinde oluşturulmaktadır. Türkçe dilindeki veri setleri bu veri setlerine göre daha az miktarda bulunmaktadır ve bu durum Türkçe dilinde gerçekleştirilen doğal dil işleme projeleri için bir engel teşkil etmektedir. Bu engelin kaldırılabilmesi için proje kapsamında Türkçe veri setlerindeki cümlelerden eş/benzer anlamlı cümleler üretilmesi amaçlanmıştır.

Cümle üretiminde kullanılacak olan modelin varyasyonel otokodlayıcılardan faydalanması planlanmıştır. Daha geleneksel olan otokodlayıcılar, verilen cümleyi kodlayıp tekrar cümlenin aynısını oluşturmak için kullanılırken varyasyonel otokadlayıcılar sayesinde verilen cümlenin anlamı korunarak farklı çeşitleri de üretilebilmektedir.

Bu raporda, projenin amacı doğrultusunda çeşitli doğal dil işleme modelleri karşılaştırılıp uygun yöntem ile geliştirelen modelden üretilen cümleler kullanılarak Türkçe veri setlerinin zenginleştirilmesindeki başarısı incelenmiştir.

Anahtar Kelimeler: Doğal dil işleme, varyasyonel otokodlayıcı, cümle üretimi, transformatör, gizli uzay.

ABSTRACT

GENERATING PARAPHRASED SENTENCES WITH VARIATIONAL AUTOENCODER MODELS

Engin MEMİŞ
Elif Sena YILMAZ

Department of Computer Engineering
Computer Project

Advisor: Prof. Dr. Mehmet Fatih AMASYALI

Data has become of great importance nowadays and the majority of data sets used in natural language processing are usually created in English. Turkish language datasets are less abundant compared to these datasets and this situation constitutes an obstacle for natural language processing projects in Turkish language. In order to overcome this obstacle, the project aims to generate sentences with same/similar meaning from sentences in Turkish data sets.

The model to be used in sentence generation is planned to utilize variational autoencoders. While the more traditional autoencoders are used to encode the given sentence and generate the same sentence again, variational autoencoders can be used to generate different variants of the given sentence while preserving its meaning.

In this report, various natural language processing models are compared for the purpose of the project and their success in enriching Turkish datasets is analyzed by using sentences generated from the model developed with the appropriate method.

Keywords: Natural language processing, variational autoencoder, sentence generation, transformer, latent space.

1
Giriş

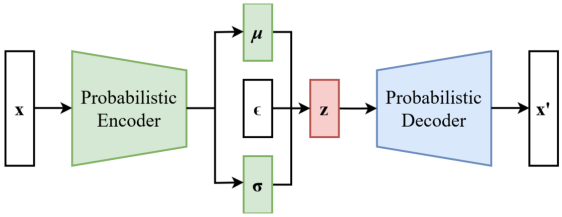
Dil, iletişim için en önemli araçtır. İnsanlar, birbirleriyle fikirlerini, duygularını ve düşüncelerini paylaşmak için dili kullanırlar. Ancak, dilin anlaşılması ve işlenmesi, makineler için insanlara olduğu kadar kolay bir görev değildir. Bu nedenle geliştirilen Doğal Dil İşleme (NLP), insan dilinin makine tarafından anlaşılmasını ve işlenmesini sağlayan bir dizi teknik ve algoritmadan oluşan teknolojidir.

NLP teknolojisi aynı zamanda, insan dilindeki çeşitli dilbilgisi yapılarını anlamak ve çıkarımlar yapmak için kullanılır. Arama motorlarındaki sorgu yanıtları, sosyal medya analizi, metin madenciliği ve daha pek çok alanda NLP teknolojisi kullanılmaktadır.

1.1 Varyasyonel Otokodlayıcılar (VAE)

Varyasyonel otokodlayıcılar (VAE), yapay sinir ağlarına dayalı derin öğrenme yöntemlerinden biridir ve Doğal Dil İşleme (NLP) alanında sıkça kullanılmaktadır. Girdi olarak verilen cümleyi normal otokodlayıcıların aksine gizli uzayda belirli bir olasılıksal dağılımını hesaplayarak temsil eder. Bu sayede cümlelerin çıkışta daha çeşitli şekillerde olmasını sağlayabilir.

Verilen veri kümesindeki örüntüleri ve ilişkileri öğrenmek için kullanılan VAE'lerin, metin verilerindeki dilbilgisi yapılarının anlaşılması ve alternatif cümlelerin oluşturulması gibi görevlerde kullanımı oldukça faydalıdır. Bu nedenle, bu projede de VAE'ler kullanarak eş anlamlı cümlelerin oluşturulması amaçlanmaktadır.



Şekil 1.1 Varyasyonel Otokodlayıcı Yapısı

2
Ön İnceleme

Bu bölümde çalışmamıza uygun çeşitli makaleler incelenmiş ve sunulmuştur.

2.1 Metin Üretimi için *Syntax-Infused* Varyasyonel Otokodlayıcılar

Bu makalede, cümlelerin dil bilgisini iyileştirmek için *syntax* ağaçlarıyla birlikte kullanılan SIVAE'nin etkisi araştırılmıştır. SIVAE, cümleler ve *syntax* ağaçları için ayrı iki *latent* uzay içerir ve iki kodlayıcı ve iki kod çözücüyü barındırır. SIVAE, uzun kısa dönemli bellek mimarileriyle birlikte çalışarak cümleleri ve *syntax* ağaçlarını aynı anda üretir. Yapılan deneyler, SIVAE'nin hem cümlenin yeniden oluşturulması hem de hedeflenen *syntax* değerlendirmeleri üzerinde üretim üstünlüğünü göstermiştir [1].

2.2 *Continuous Space*'ten Cümle Üretimi

Bu makalede, tek tek kelimeler yerine cümlelerin tamamını göz önünde bulunduran bir RNN tabanlı varyasyonel otokodlayıcı modeli araştırılmıştır. RNNLM modeli, cümleleri bütünsel olarak modelleyerek, çeşitli ve iyi oluşturulmuş cümleler üretmeye olanak sağlar. Modelin eksik kelimeleri tamamlama konusunda etkinliği gösterilmiş ancak dil modellemesinde kullanımının olumsuz sonuçları da ortaya konulmuştur [2].

2.3 *Masked Language Modeling* için Varyasyonel Cümle Çoğaltma

Bu makalede, VAE ve GRU yöntemlerini içeren bir varyasyonel cümle çoğaltma yöntemi tanıtılmaktadır. Önerilen yöntem, dilin anlamsal ve *syntax* özelliklerini kodlayan *latent* uzay temsili sayesinde veri artırmanın faydalarından yararlanır. Model, dilin temsili öğrenildikten sonra, GRU'nun ardışık yapısıyla cümleleri *latent* uzayından üretir. Var olan yapılandırılmamış korpusun artırılmasıyla, model pre-training'de MLM'i geliştirir. *Pre-training*'de yöntem, maskelenmiş *token*'lerin tahmin oranını arttırırken, *fine-tuning*'de varyasyonel cümle artırmanın semantik görevlerde ve *syntax* görevlerinde yardımcı olabileceği gösterilmiştir [3].

3
Fizibilite

Bu bölümde fizibilite çalışması 4 farklı bölümde anlatılmaktadır: teknik fizibilite, ekonomik fizibilite, yasal fizibilite ve zaman fizibilitesi.

3.1 Teknik Fizibilite

Teknik fizibilite iki alt başlık altında açıklanmaktadır: yazılım fizibilitesi ve donanım fizibilitesi.

3.1.1 Yazılım Fizibilitesi

Bu bölümde projede kullanılan geliştirme ortamı ve programlama dilleri anlatılmaktadır.

- **Geliştirme Ortamı**
  Google Colab:
  Colaboratory (kısaca "Colab"), Google Research tarafından sunulan bir üründür. Özellikle makine öğrenimi, veri analizi ve eğitim için uygun olan Colab, ücretsiz GPU erişimi verdiği için tercih edilmiştir [4]. Sonrasında ücretsiz versiyon daha büyük veri setlerinde çalışmak için Colab Pro'ya yükseltilmiştir.

- **Programlama Dili**
  Python:
  Python, veri bilimi ve makine öğrenmesinde yaygın olarak kullanılan nesneye yönelik bir programlama dili olduğu için kullanılmıştır.

3.1.2 Donanım Fizibilitesi

Bu bölümde minimum ve önerilen sistem gereksinimleri verilmiştir.

Tablo 3.1 Minimum Sistem Gereksinimleri

RAM	4 GB
Gerekli Disk Alanı	5 GB
CPU	2.4 GHz
GPU	4 GB

Tablo 3.2 Önerilen Sistem Gereksinimleri

RAM	16 GB
Gerekli Disk Alanı	15 GB
CPU	2.6 GHz
GPU	16 GB

3.2 Ekonomik Fizibilite

Bu bölümde proje için gereken bilgisayarın ve projeyi geliştiren personelin maaşları verilmiştir.

Tablo 3.3 Gerekli Donanım Ücreti

Ürün	**Fiyat**
1x Önerilen Bilgisayar	12000 ₺
1x Önerilen GPU	32000 ₺

Tablo 3.4 Personel Gider Tablosu

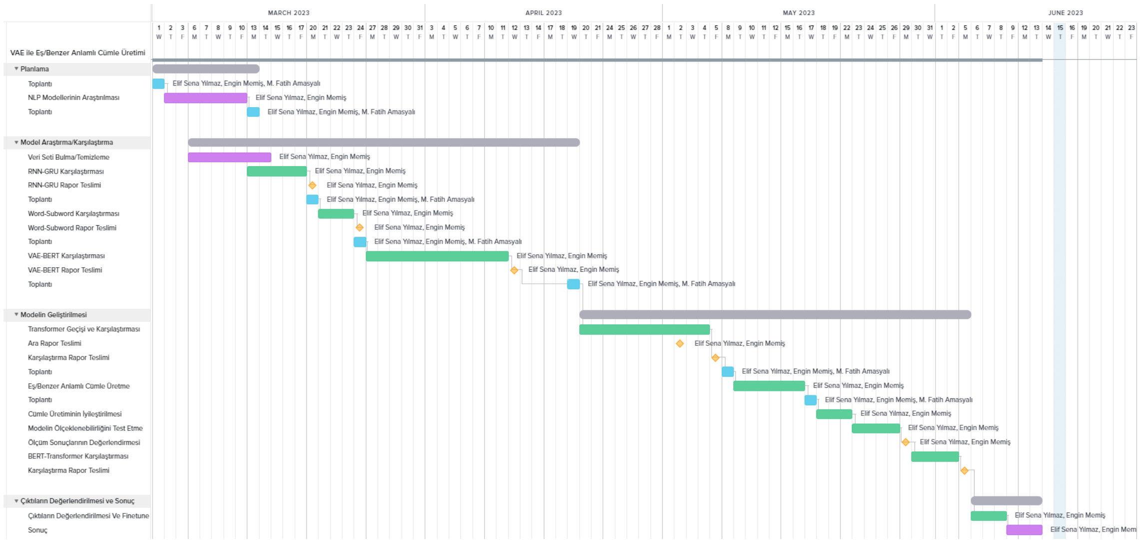
Personel	**Kişi**	**Gün**	**Günlük Maaş**	**Toplam**
Sistem Analisti	2	5	800 ₺	8000 ₺
Yazılım Geliştiricisi	2	27	950 ₺	51300 ₺
Yazılım Test Uzmanı	1	5	600 ₺	3000 ₺
			Toplam Gider	62300 ₺

3.3 Yasal Fizibilite

Proje süresince kullanılan veri setleri halka açık olduğundan dolayı hiçbir patent ve marka hakkı ihlal edilmemiştir. Bunun dışında açık kaynak kodlu ve ücretsiz ürünler kullanılmaktadır. Gerekli kanunlar incelendiğinde projenin önünde herhangi bir hukuki engel bulunmamaktadır. Bu nedenle proje yasaldır.

3.4 İş Gücü ve Zaman Fizibilitesi

Bu bölümde, projenin görev ve zaman dağılımı Şekil 3.1'de gösterilmiştir.



Şekil 3.1 Gantt Şeması

4
Sistem Analizi

4.1 Amaç

Bu projenin amacı, girilen bir cümle ile benzer/aynı anlamlı yeni bir cümle üreten sistem geliştirmektedir. Bu projenin kapsamında, klasik eş/benzer anlamlı cümle üretim mimarilerinde gereken cümle ikililerinden oluşan kümenin gereksinimi, VAE'ler sayesinde eğiticisiz olarak sağlanacaktır. Türkçe veri setlerine katkı sağlamak amacıyla Türkçe metinler üzerinde çalışılacaktır.

4.2 Gereksinim Analizi

Projede yazılan kodun geliştirilmesinde NumPy, Pandas, PyTorch, Transformers ve birkaç basit kütüphane kullanılmıştır. Geliştirme ortamı olarak Anaconda Navigator, Visual Studio Code ve Google Colab seçilmiştir.

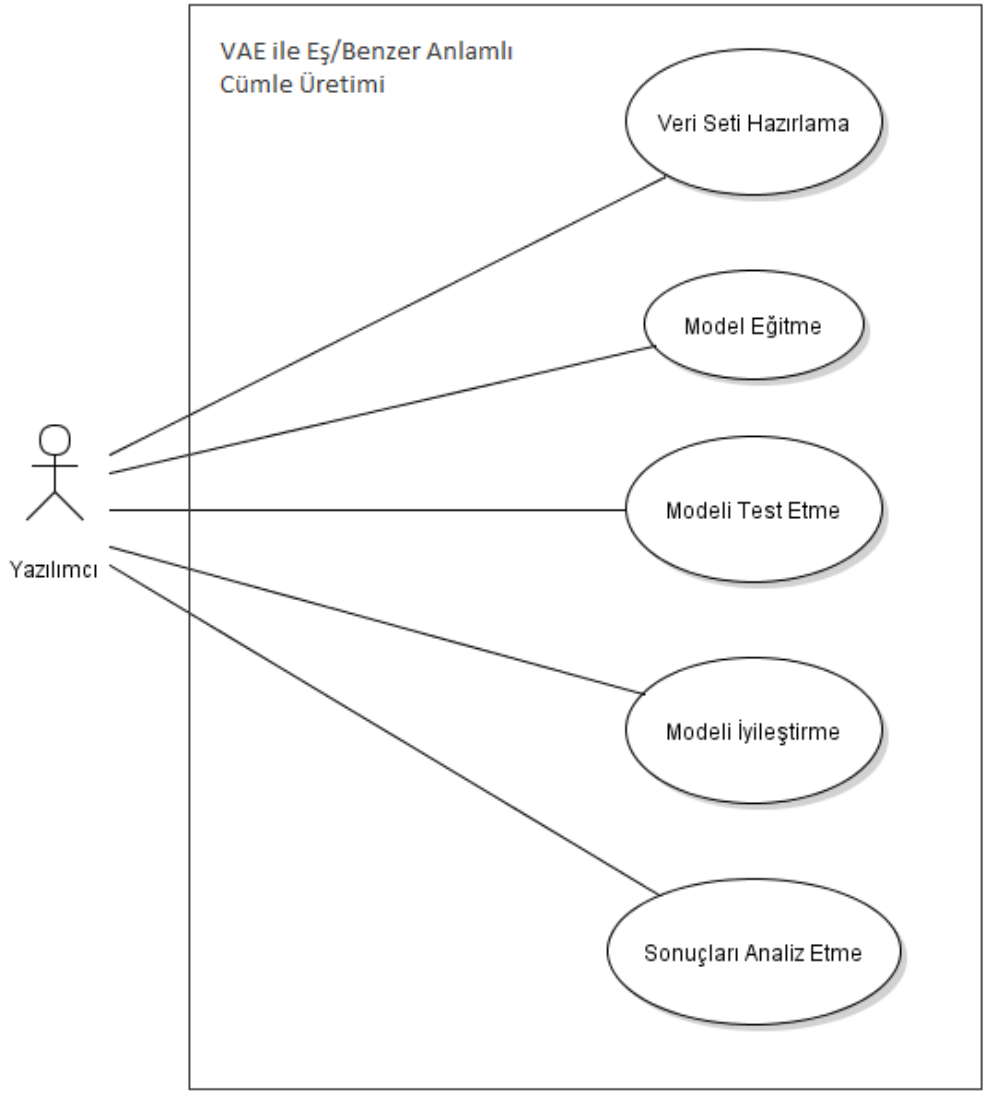
Model seçimi yaparken, karşılaştırma yapmak amacıyla, 100MB ve 10MB'lık, modeli test ederken ise sırasıyla 10MB ve 4MB'lık halka açık Türkçe Vikipedi cümleleri içeren veri setleri kullanılmıştır. Test veri setleri eğitim veri setlerinden farklı cümleler içermektedir.

Modellerin karşılaştırılması için farklı sınıflandırma veri setleri kullanılmıştır. Bunlar haber başlıklarını sınıflandıran, deprem tweetlerini yardım içerikli olup olmama durumuna göre sınıflandıran, tweetlerin duygularını sınıflandıran olmak üzere 3 farklı veri setidir.

Seçtiğimiz modeli eğitirken halka açık olan 1GB boyutunda Türkçe cümleler içeren veri setine geçiş yapılmıştır. Dolayısıyla test veri seti de 100MB olacak şekilde güncellenmiştir. Bu iki veri seti birbirinden farklı cümleler içermektedir.

4.3 Kullanım Senaryosu (Use Case) Diyagramı

Bu bölümde, projenin kullanım senaryosu Şekil 4.1'de gösterilmiştir.



Şekil 4.1 Use Case Diyagramı

5
Sistem Tasarımı

Bu bölümde projenin yazılım tasarımı anlatılmıştır.

5.1 Doğal Dil İşleme için Yapay Sinir Ağları

Yapay sinir ağları, beynin sinirsel yapısını örnek alarak sınıflandırma, tahmin, karar verme, görselleştirme ve benzeri görevleri gerçekleştirmeyi öğrenebilen doğrusal olmayan bir modeldir. Yapay sinir ağı, yapay nöronlardan oluşur ve birbirine bağlı girdi, birden fazla katman içerebilen gizli ve çıktı şeklinde üç katman halinde düzenlenir [5].

5.1.1 *Recurrent Neural Network* (RNN)

RNN'ler olarak bilinen yinelemeli sinir ağları, nöronlar arasındaki bağlantıların döngü oluşturduğu bir yapay sinir ağı çeşididir. Bu, çıktının yalnızca mevcut girdilere değil, aynı zamanda bir önceki adımın nöron durumuna da bağlı olduğu anlamına gelir [5].

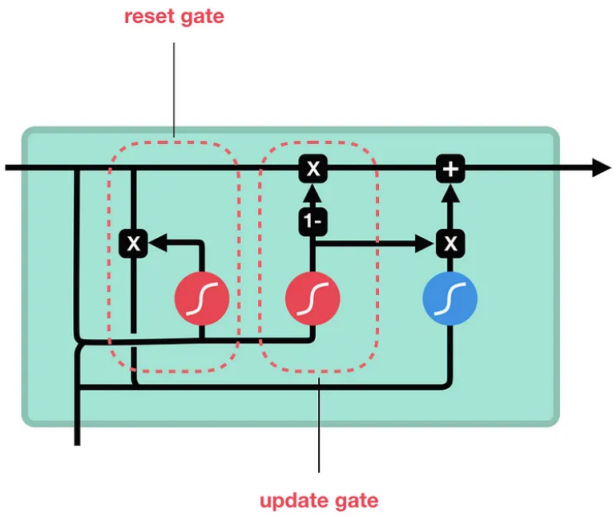
- ***Vanishing Gradient* Sorunu:** Back-propagation ile birlikte ağırlıklar Zincir Kuralı (Chain Rules) üzerinden ayarlandığından bu RNN için kaçınılmaz bir problemdir. Her bir katmandaki ağırlıklar, gradient değerleri geriye doğru adım ilerledikçe katlanarak küçülecek ve sonunda yok olacaktır [6].

5.1.2 *Gated Recurrent Unit* (GRU)

GRU, vanishing-gradient sorununa çözüm olarak ortaya çıkmıştır. Bilgi akışını düzenleyebilecek geçit adı verilen mekanizmalara sahiptir [6].

- **Sıfırlama Geçiti:** Önceki bilgilerin ne kadarının unutulacağına karar vermek için kullanılır.

- **Güncelleme Geçiti:** Hangi bilgilerin saklanıp,hangi bilgilerin atılacağına ya da hangi yeni bilgilerin ekleneceğine karar verir.



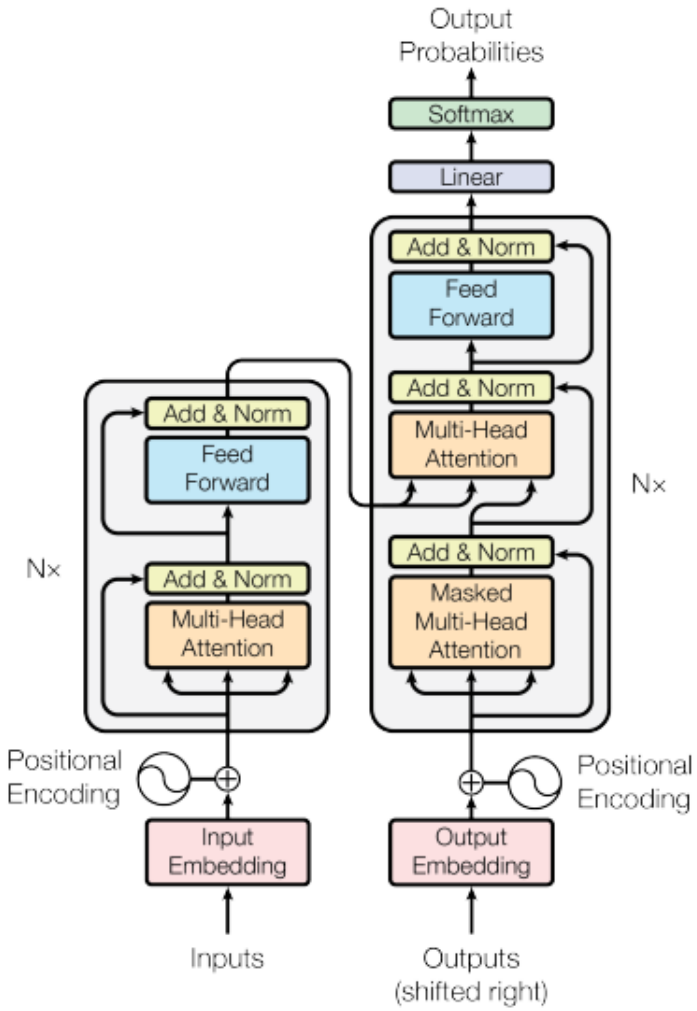
Şekil 5.1 GRU Hücresi ve Geçitleri [7]

5.1.3 *Transformer* (Transformatör)

İlk olarak 2017 yılında Google tarafından tanıtılan transformatörler, NLP alanında yaygın olarak kullanılmaya başlanmıştır. Transformatörlerin önceki modellere göre en önemli avantajı, diğer modeller gibi yinelemeli çalışmadığı için makaleler veya kitaplar gibi uzun biçimli metin girdilerini işleyebilmeleridir. Bu, modelin farklı zamanlarda girdi dizisinin farklı bölümlerine odaklanmasına olanak tanıyan *self-attention* mekanizmalarının kullanılmasıyla gerçekleştirilir [8].

Self-attention, cümledeki kelime çiftleri arasındaki ilişkilere odaklanan bir dizi işleme mekanizmasıdır, bu kelimelerin bir cümlenin başında, sonunda veya ortasında olup olmadığını algılamaz [8].

Transformatör modelinde de bulunan *Multi-head attention* ise birden fazla *self-attention* mekanizmasının paralel olarak kullanılmasını sağlayan ve karmaşık hesaplamalar yapmasına olanak tanıyan bir yapıdır [8].



Şekil 5.2 Transformatör Model Mimarisi [9]

5.1.3.1 BERT

BERT, transformatör mimarisine dayalı bir dil modelidir. BERT hem büyük tek yönlü hem de çift yönlü dil modelleri üzerinde eğitilir. Bu, bir kelimenin bağlamını kendisinden önce ve sonra gelen kelimelere dayanarak öğrenebileceği anlamına gelir. BERT kendi başına birden fazla problemde kullanılabilecek şekilde tasarlanmış bir modeldir. Farklı problemlerde kullanılabilmesi için üstüne ekstra katmanlar eklenmesi gerekmektedir [10].

5.2 Modellerin Karşılaştırılması

Bu bölümdeki tüm karşılaştırmalar 100MB ve 10MB Türkçe Vikipedi cümleleri içeren veri setleri kullanılarak yapılmıştır.

5.2.1 RNN-GRU Karşılaştırması

GRU'da bulunan geçitlerden dolayı GRU'nun daha yavaş çalışacağı tahmin edilmesine rağmen RNN'in az da olsa daha yavaş çalıştığı ölçülmüştür. Çıktılarının kalitesine bakıldığında RNN'in ürettiği cümlelerin kalitesi GRU'ya göre çok düşük kaldığı gözlemlenmiştir. Bu yüzden RNN 100 MB veri setiyle eğitilmemiştir.

Tablo 5.1 Bir *Epoch* Eğitimi için Gerekli Süreler (RNN-GRU)

	RNN	**GRU**
10 MB	102,80 saniye	95,37 saniye
100 MB	-	4470,34 saniye

5.2.2 GRU Kullanılarak *Word-Subword* Karşılaştırması

Veri setindeki cümleler, modelde kullanılabilmesi için BERTurk'ün [11] 32K *token* sayısına sahip *tokenizer*'ı ile işleme sokulmuştur. *Subword* yapısında veri setindeki toplam *token* sayısı azaldığı için sürede önemli derecede düşüş gözlemlenmiştir.

Tablo 5.2 Bir *Epoch* Eğitimi için Gerekli Süreler (*Word-Subword*)

	Word	**Subword**
10 MB	93.37 saniye	55.10 saniye
100 MB	4470.34 saniye	560.94 saniye

5.2.3 GRU-BERT Temsilleri Karşılaştırması

BERT 100 MB veri setiyle sıfırdan bir *epoch* eğitilmiştir. GRU ise aynı veri setiyle sıfırdan on *epoch* eğitilmiştir. *Epoch* sayılarındaki fark BERT'i eğitmenin çok uzun sürmesinden kaynaklıdır. Bu eğitilen modellerden alınan temsiller ile içerisinde üç farklı sınıflandırma veri setinde karşılaştırılma yapılmıştır. Bunlar üç farklı haber kategorisi bulunan veri seti [12], 5 farklı duygu kategorisi olan Twitter veri seti [13], depreme yardım içerikli olup olmama olarak iki kategorisi olan Twitter veri setidir[14].

Tablo 5.3 Temsillerle Tahmin Edilen Haber Veri Seti Sınıflandırma Başarı Oranları

	SVM	**Logistic Regression**
BERT	%72,2	%73,7
GRU	%86,6	%84,6

Tablo 5.4 Temsillerle Tahmin Edilen Duygu Veri Seti Sınıflandırma Başarı Oranları

	SVM	**Logistic Regression**
BERT	%63,8	%65,5
GRU	%68,2	%65,7

Tablo 5.5 Temsillerle Tahmin Edilen Deprem Veri Seti Sınıflandırma Başarı Oranları

	SVM	**Logistic Regression**
BERT	%88,6	%90,6
GRU	%90,6	%91,3

6
Uygulama

Bu bölümde oluşturulan modelin yapısından detaylı bir şekilde bahsedilip modelin geliştirme aşaması anlatılacaktır.

6.1 Modelin Mimarisi

6.1.1 *Encoder* (Kodlayıcı)

Model transformatör mimarisine uygun olarak tasarlanmış olup aşağıdaki kısımları içerir.

- *Input Embedding*
- *Positional Encoding*
- 6x Katman
  - *Multi-Head Attention*
  - *Normalization Layers*
  - *Position-Wise Feed-Forward*

6.1.2 *Latent Space* (Gizli Uzay)

Verilen cümle *encoder*'dan geçtikten sonra üzerine gürültü eklenerek 16 boyutlu gizli uzayda temsil edilir. Benzer anlmalı cümlelerin bu uzayda birbirine yakın olarak konumlanması beklenir.

6.1.3 *Decoder* (Çözücü)

Yine transformatör mimarisine uygun olarak *decoder* tasarlanmış olup aşağıdaki kısımları içerir. *Encoder*'a girdi olarak verilen cümlenin gizli uzayda temsil ettiği nokta alınıp *decoder*'da katmanlardan geçerek işlem görür.

- *Target Embedding*
- *Positional Encoding*
- 6x Katman
  - *Multi-Head Attention*
  - *Encoder-Decoder Attention*
  - *Normalization Layers*
  - *Position-Wise Feed-Forward*

6.1.4 *Loss Function* (Kayıp Fonksiyonu)

Modelde kayıp fonksiyonu olarak *Cross Entropy* kullanılmıştır.

6.1.5 Eğitim Parametreleri

Aşağıda bu modelin eğitiminde kullanılan bazı önemli parametreler listelenmiştir.

- Batch Size: 64
- Learning Rate: 0.001
- Optimizer: Stochastic Gradient Descent (SGD)

6.2 Modelin Geliştirilmesi

Projede kullanılacak olan transformatör yapısını içeren hazır bir model(BERT), diğer modellerle karşılaştırılmıştır. Bu karşılaştırma sonucunda transformatör yapısını içeren modelin cümle üretiminde daha başarılı olduğu görülüp varyasyonel otokodlayıcı içeren bir transformatör modeli oluşturulmuştur.

Bu bölümde bahsedilen tüm ölçüm sonuçları "Deneysel Sonuçlar" bölümünde detaylı olarak verilecektir.

Yeni oluşturulan model ve diğer modellerin 100 MB veri setiyle ilk eğitimi yapılmıştır. Diğer modellerle aynı seviyeye gelip gelmediğini test etmek için çeşitli sınıflandırma veri setleri üzerinde başarıları ölçülmüştür.

Modelin başarısı beklenen düzeye ulaştığı için eğitim veri seti 10 kat arttırılarak 1 GB boyutuna getirilip 1 epoch eğitildiğinde ürettiği cümleler ile 100 MB veri setiyle 10 epoch eğitildiğinde ürettiği cümlelerin kalitesine ve eğitim süresine bakılarak

ölçeklenebilirliği test edilmiştir. Bunu yaparken 1 GB veri seti RAM boyutunun yetersiz olmasından dolayı bütün bir şekilde eğitime sokulamamıştır. Bu sorunu çözmek için veri seti 200 MB'lık 5 parçaya bölünmüş, her parçanın eğitim sonucu bir sonraki parçanın eğitimi için *checkpoint* olarak kullanılmıştır. Bu karşılaştırma sonucunda, eğitim ikisinde de yaklaşık 8 saat sürmüş fakat cümlelerin kalitesinin 1 GB ile eğitilen modelde biraz daha iyi olduğu gözlemlenmiştir. Bu sonuçlardan sonra eğitimin 1 GB veri setiyle devam etmesine karar verilmiştir.

Model ile oluşturulan cümlenin verilen cümle ile tamamen aynı cümle değil de benzer bir cümle olması için cümle üretimi gürültü eklenerek yapılmaktadır. Fakat oluşturulan model cümlelerin aynısını üretecek kadar gelişmediği için cümle üretiminden gürültü etkisinin çıkarılması uygun görülmüştür.

7
Deneysel Sonuçlar

Bu bölümde modelin geliştirilmesi süreci boyunca yapılan ölçümlerin sonuçları verilecektir.

7.1 Oluşturulan Modelin Diğer Modellerle Karşılaştırılması

Geliştirilen modelin başarısını ölçmek için önceki modeller (GRU, BERT) ile bu yeni model, sınıflandırma içeren üç farklı veri seti üzerinde denenmiştir. Bu veri setleri; beş farklı duygu kategorisi bulunan *tweet* veri seti, üç farklı haber kategorisi bulunan haber cümleleri veri seti ve depreme yardım içerikli ya da değil olarak iki farklı kategori içeren *tweet* veri setleridir.

GRU 100 MB veri setiyle on *epoch*, BERT 100 MB veri setiyle bir *epoch*, transformatör 100 MB veri setiyle bir *epoch* eğitildikten sonra oluşan modeller kullanılmıştır. *Epoch* sayısındaki farklılık eğitim sürelerin çok uzun olmasından dolayıdır.

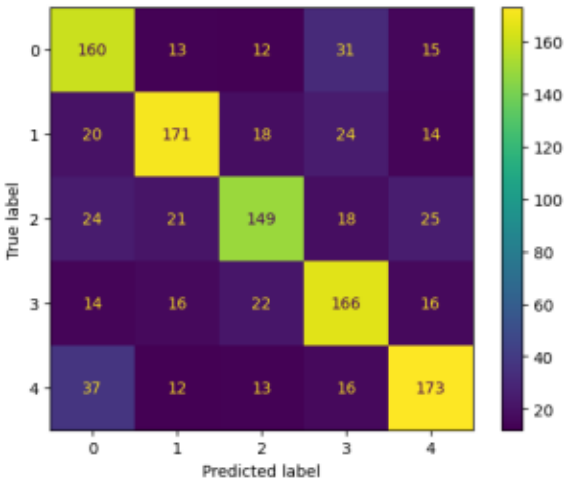
7.1.1 Duygu Sınıflandırma Veri Seti

Bu veri setinde kızgın, korku, mutlu, üzgün, sürpriz olmak üzere beş farklı kategori bulunmaktadır.

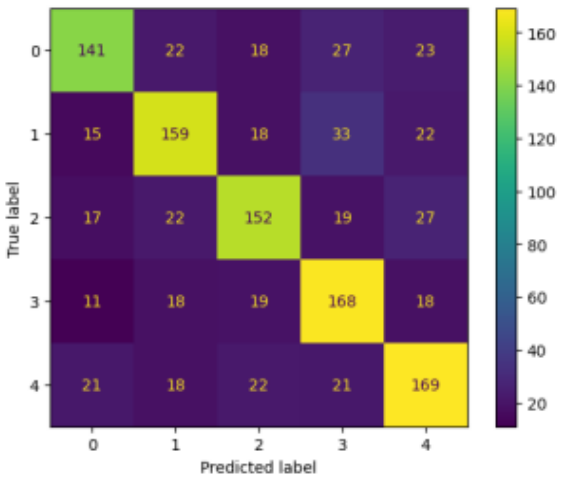
Modellere ait ölçümlerin başarı oranları Tablo 7.1'de gösterilmiştir. Sonrasında bu başarı oranlarına ait karışıklık matrisleri verilmiştir.

Tablo 7.1 Duygu Veri Setinde Modellerin Başarı Oranları

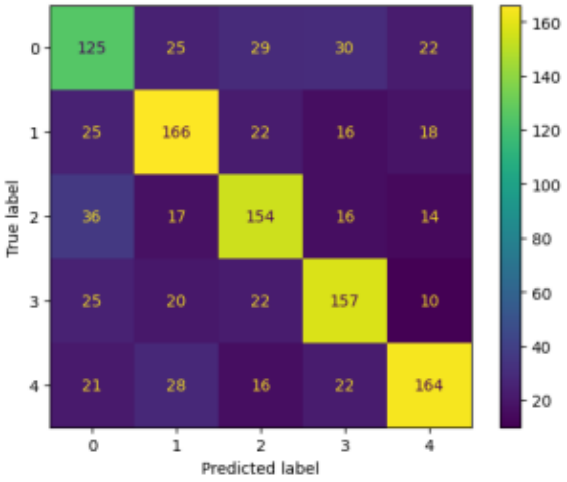
	SVM	**Logistic Regression**
GRU	%68,2	%65,7
BERT	%63,8	%65,5
Transformatör	%68,3	%67,3



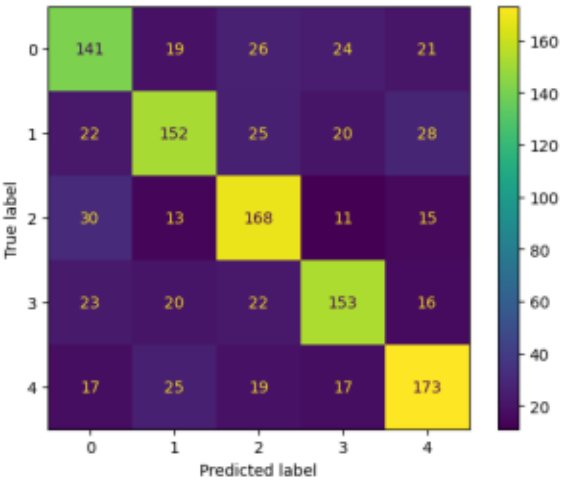
Şekil 7.1 GRU-SVM Karışıklık Matrisi



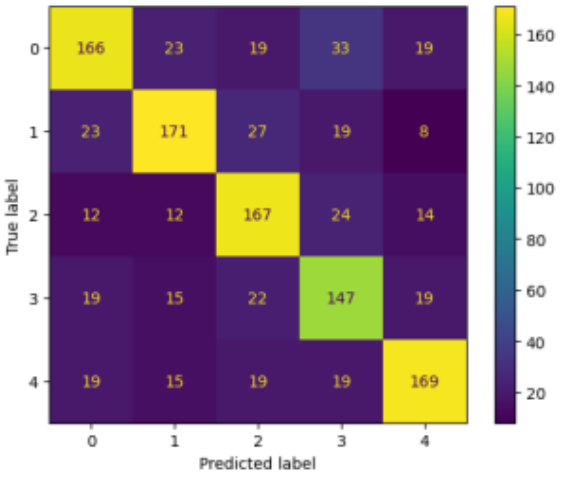
Şekil 7.2 GRU-Log. Reg. Karışıklık Matrisi



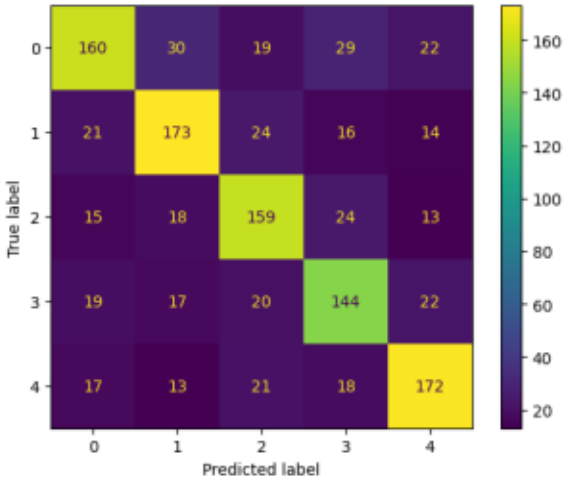
Şekil 7.3 BERT-SVM Karışıklık Matrisi



Şekil 7.4 BERT-Log. Reg. Karışıklık Matrisi



Şekil 7.5 Transformatör-SVM Karışıklık Matrisi



Şekil 7.6 Transformatör-Log. Reg. Karışıklık Matrisi

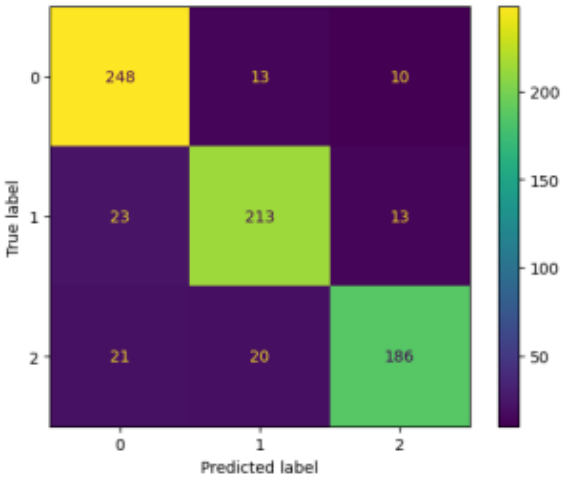
7.1.2 Haber Cümleleri Sınıflandırma Veri Seti

Bu veri setinde siyaset, spor, teknoloji olmak üzere üç farklı kategori bulunmaktadır.

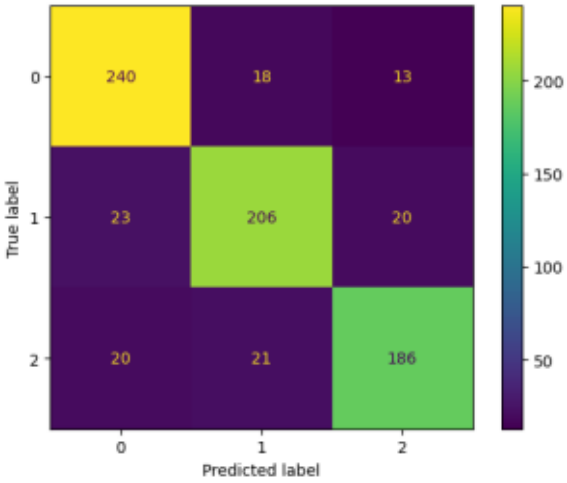
Modellere ait ölçümlerin başarı oranları Tablo 7.2'de gösterilmiştir. Sonrasında bu başarı oranlarına ait karışıklık matrisleri verilmiştir.

Tablo 7.2 Haber Veri Setinde Modellerin Başarı Oranları

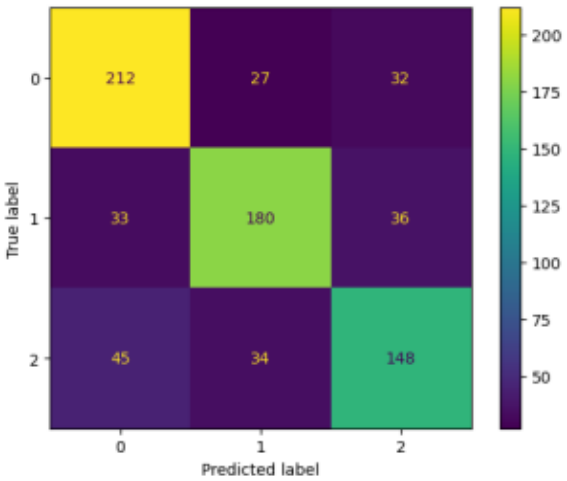
	SVM	**Logistic Regression**
GRU	%86,6	%84,6
BERT	%72,2	%73,7
Transformatör	%68,1	%66,1



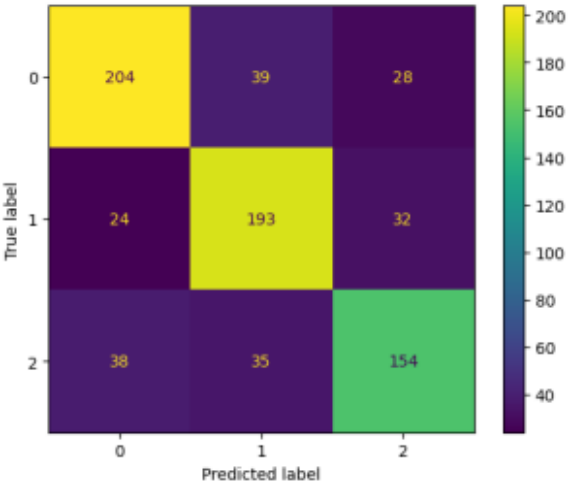
Şekil 7.7 GRU-SVM Karışıklık Matrisi



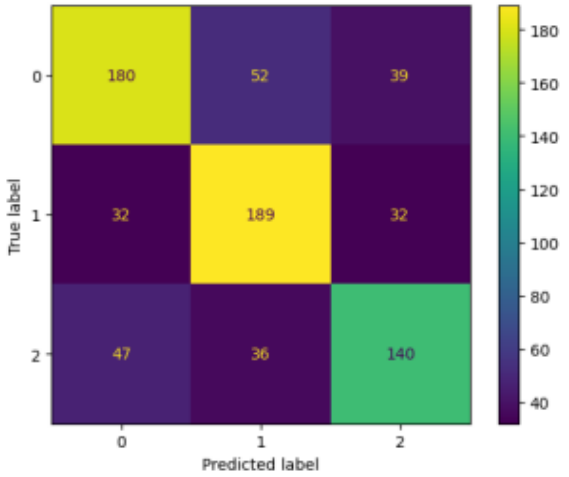
Şekil 7.8 GRU-Log. Reg. Karışıklık Matrisi



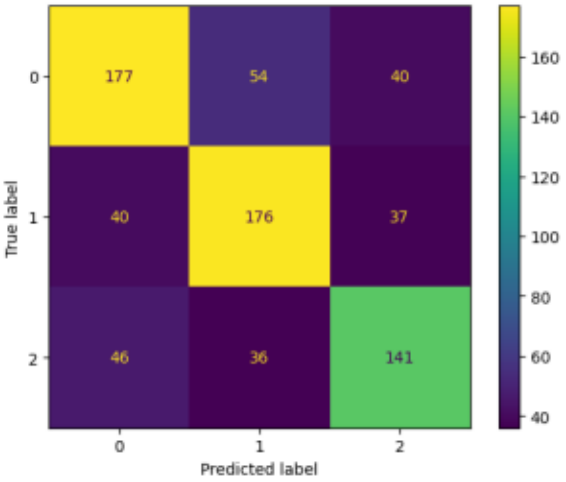
Şekil 7.9 BERT-SVM Karışıklık Matrisi



Şekil 7.10 BERT-Log. Reg. Karışıklık Matrisi



Şekil 7.11 Transformatör-SVM Karışıklık Matrisi



Şekil 7.12 Transformatör-Log. Reg. Karışıklık Matrisi

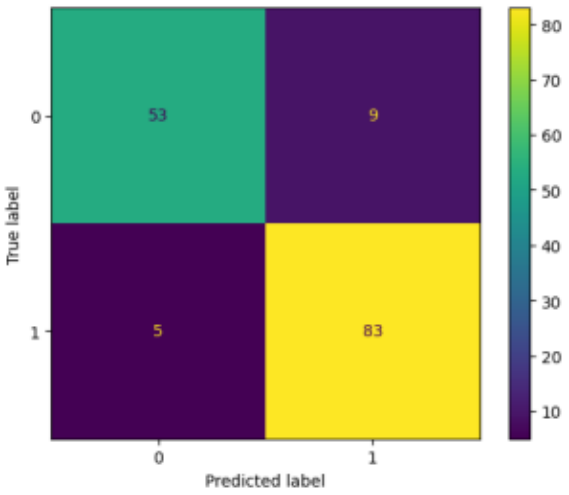
7.1.3 Deprem *Tweet*leri Veri Seti

Bu veri setinde yardım içerikli olma ve olmama olmak üzere iki farklı kategori bulunmaktadır.

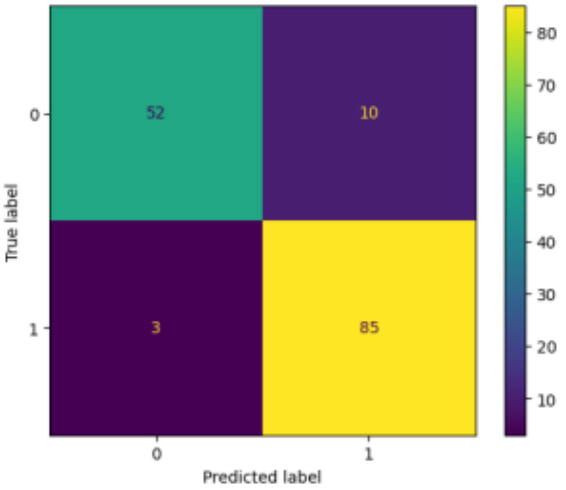
Modellere ait ölçümlerin başarı oranları Tablo 7.3'de gösterilmiştir. Sonrasında bu başarı oranlarına ait karışıklık matrisleri verilmiştir.

Tablo 7.3 Deprem *Tweet*leri Veri Setinde Modellerin Başarı Oranları

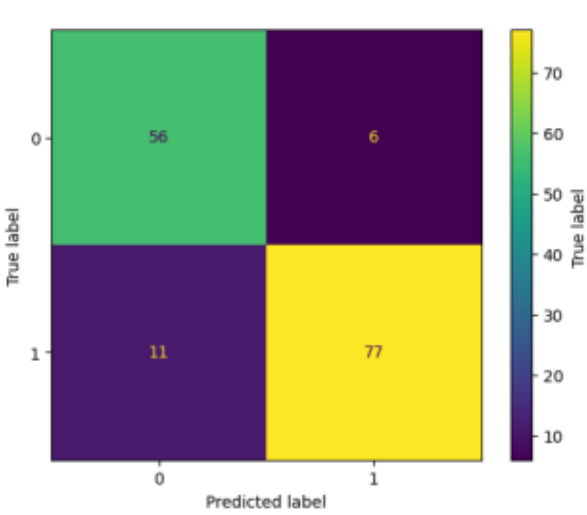
	SVM	**Logistic Regression**
GRU	%90,6	%91,3
BERT	%88,6	%90,6
Transformatör	%86	%88



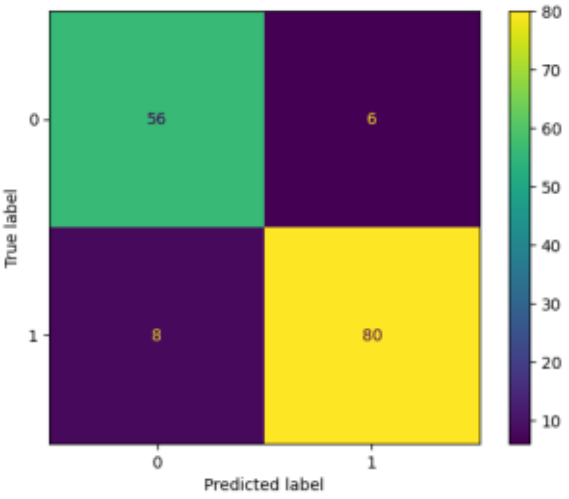
Şekil 7.13 GRU-SVM Karışıklık Matrisi



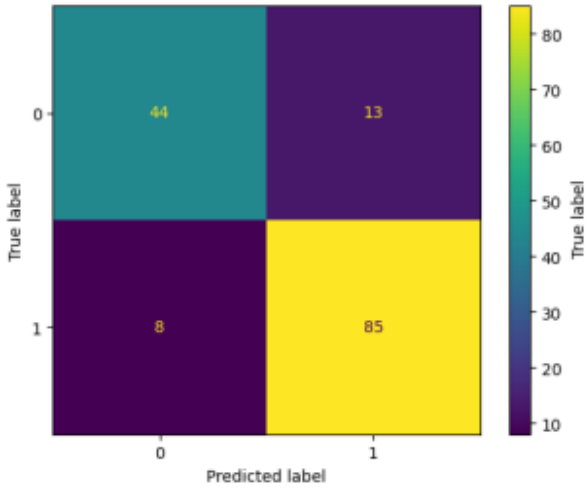
Şekil 7.14 GRU-Log. Reg. Karışıklık Matrisi



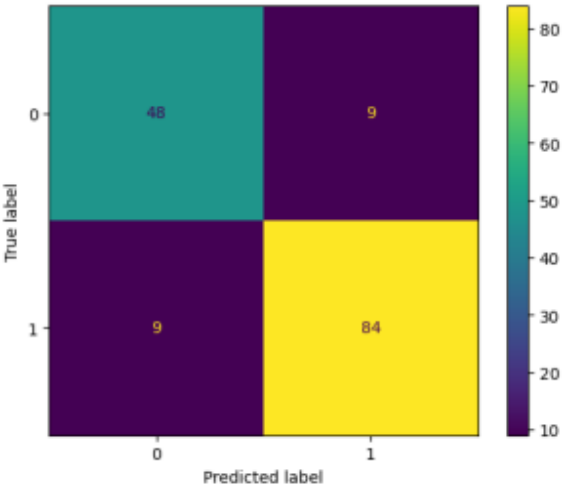
Şekil 7.15 BERT-SVM Karışıklık Matrisi



Şekil 7.16 BERT-Log. Reg. Karışıklık Matrisi



Şekil 7.17 Transformatör-SVM Karışıklık Matrisi



Şekil 7.18 Transformatör-Log. Reg. Karışıklık Matrisi

7.2 Modelin Ölçeklenebilirliği

Aşağıda modelin 1 GB 1 *epoch* ve 100 MB 10 *epoch* eğitimi sonucunda üretilen cümlelerden bazı örnekler gösterilmiştir.

Verilen Cümle:	Türkiye'nin büyük bir otomotiv sanayisi vardır.
100 MB - 10 Epoch:	Türkiye'nin büyük bir a,si vardır.
1 GB - 1 Epoch:	Türkiye'nin büyük bir'-si vardır.

Verilen Cümle:	Mısır Tarihi boyunca 190 Kral hüküm sürmüştür.
100 MB - 10 Epoch:	ni vardır boyuncaen 1) 1.
1 GB - 1 Epoch:	Mısır a boyunca 190yın vetür.

Verilen Cümle:	Bu yolun ortalarına doğru, bozkıra açılan bir kapı vardır.
100 MB - 10 Epoch:	Bu : ortalarına doğru,in 2a'bir'vardır.
1 GB - 1 Epoch:	Bu de ortalarına doğru, diye aa 6 bir ne vardır.

Verilen Cümle:	Fiziksel teorilerin geçerliliği bilimsel metot ile test edilir.
100 MB - 10 Epoch:	e da lerinin liğien ile na edilir.
1 GB - 1 Epoch:	için lerin niliği ancak on ile = edilir.

Verilen Cümle:	Bu bilim dalı bilimsel bilginin yanı sıra özel yetenek ve önsezi gerektirir.
100 MB - 10 Epoch:	Bu " iki "n yanı sıra özel üzerine vetsezi sonra.
1 GB - 1 Epoch:	Bu bilim (ancak'yanı sıra özeln ve önsezi ki.

7.3 Gürültünün Cümle Üretimine Etkisi

Aşağıda bazı cümlelerin modelin 1 GB 1 *epoch* eğitimden sonra gürültülü ve gürültüsüz şekilde üretilen karşılıkları verilmiştir.

Verilen Cümle:	Gazetede yayınlanmayan buna rağmen, sizin bildiğiniz başka şeyler var mı
Gürültüsüz:	da buna rağmen , başka şeyler var mı
Gürültülü:	(ya rağmen , tini başka şeyler var mı

Verilen Cümle:	Geç Eski Taş Çağı'nda Kuzey Afrika'nın kurak iklimi giderek ısındı ve kuraklaştı.
Gürültüsüz:	Geç Eski birmı'nda Kuzey Afrika'nın ve iklimi nin olarakdı vee olan.
Gürültülü:	Avrupa Eski bir onunı'nda Kuzey Afrika'nın gibi iklimi nin olarakdı ve gibi).

Verilen Cümle:	Her şey diğer organizmalar ve çevreyle etkileşim içerisindedir.
Gürültüsüz:	Her şey diğer dan 18 ve biryle ' içerisindedir.
Gürültülü:	Her şey diğer (18 ve çevreylema adı.

7.4 Eğitimde *Epoch* Sayısının Üretilen Cümle Kalitesine Etkisi

Her *epoch* eğitimi yaklaşık 8 saat sürmüş olup, her *epoch* sonunda modelden cümle üretilmiştir. Bu cümleler incelendiğinde her *epoch* sonunda cümle kalitelerinde iyileşmeler gözlenmiştir. Aşağıda bu cümlelerden bazı örnekler gösterilmiştir.

Verilen Cümle:	Takımı baskı altına almaya kimsenin hakkı yok.
1 Epoch:	dagn altına inl yok.
2 Epoch:	Kar baskı altına almaya - hakkı yok.
3 Epoch:	Takımı baskı altına almaya kimsenin hakkı yok.

Verilen Cümle:	Türkiye'de en çok aboneye sahip olan spor kulübü dergisidir.
1 Epoch:	Türkiye'de en çok :ye sahip olanleragndir.
2 Epoch:	Türkiye'de en çoklardanye sahip olan spor kulübügndir.
3 Epoch:	Türkiye'de en çok 3ye sahip olan spor kulübü dergisidir.

Verilen Cümle: Günümüzde en çok yetiştirilen tarım ürünleri arasında pirinç, mısır ve buğday yer almaktadır.
1 Epoch: a en çok -ilen tarım Ocak arasında başladı, 2007 ve eski yer almaktadır.
2 Epoch: Günümüzde en çok yetiştirilen tarıman arasında, 2007 vet yer almaktadır.
3 Epoch: Günümüzde en çok yetiştirilen tarım ürünleri arasında enerji, 2007 ve Se yer almaktadır.

Verilen Cümle: Hukuk biliminde biçim, öncelikler ve ilkeler doğrultusunda bazı sistemler ortaya çıkmıştır.
1 Epoch: C büyük. (ği içinla 'tir bir tanım vardır).
2 Epoch: Hukuk 2010de biçim, :r ve sonra doğrultusunda bazılara ortaya çıkmıştır.
3 Epoch: Hukuk kısade biçim, öncelikler vele doğrultusunda bazıtu ortaya çıkmıştır.

Verilen Cümle: Doktor Mortimer gözlük camlarının altında gözlerini kırpıştırarak şaşkınlıkla baktı.
1 Epoch: ilenineer vardır :larının altındadıarakki olan W.
2 Epoch: Aynı Mortimer vardır :larının altında.ıştırarak, olan baktı.
3 Epoch: Doktor Mortimerg camlarının altında gözlerini sonraıştırarakkilıkla baktı.

8
Performans Analizi

Bu bölümde; performans ölçümü için, oluşturduğumuz model ile sınıflandırma veri setlerindeki cümlelerin eşini veya benzerini üreterek veri seti zenginleştirilip tekrar teste sokulduktan başarı oranlarının nasıl değiştiği incelenmiştir.

Performans analizi için cümle üretimi ve zenginleştirme üç şekilde gerçekleştirilmiştir. Bu yöntemler aşağıdaki gibidir.

- **%100 Üretim - %50 Veri Setine Ekleme:** Veri setindeki her cümle için bu cümleye karşılık gelen benzer cümle oluşturulduktan sonra bu cümleler loss oranlarına bakılarak başarı oranına göre sıralanmıştır. Sonrasında bu sıradan cümlelerin orijinal veri setinin yarısı kadar olan kısmı alınıp orijinal veri setine eklenmiştir, yani sınıflandırma veri seti yeni üretilen cümlelerle 1,5 katına çıkarılmıştır.
- **%100 Üretim - %100 Veri Setine Ekleme:** Veri setindeki her cümle için bu cümleye karşılık gelen benzer cümle oluşturulduktan sonra cümlelerin tamamı orijinal veri setine eklenmiştir, yani sınıflandırma veri seti yeni üretilen cümlelerle 2 katına çıkarılmıştır.
- **%500 Üretim - %50 Veri Setine Ekleme:** Veri setindeki her cümle için bu cümleye karşılık gelen 5 tane benzer cümle oluşturulduktan sonra bu cümleler loss oranlarına bakılarak başarı oranına göre sıralanmıştır. Sonrasında bu sıradan cümlelerin orijinal veri setinin yarısı kadar olan kısmı alınıp orijinal veri setine eklenmiştir, yani sınıflandırma veri seti yeni üretilen cümlelerle 1,5 katına çıkarılmıştır.

Model; daha uzun bir süre boyunca daha büyük bir veri setiyle eğitilmediği için veri setlerindeki cümle tarzlarının eğitim veri setindeki cümle tarzlarından farklılığı ve cümle üretilirken oluşan rastgelelik yüzünden her cümle için başarılı bir karşılık üretememektedir. Son belirttiğimiz yöntemde, bu rastgeleliğin etkilerinin her

cümleye karşılık olarak 5 tane cümle üretildikten sonra bunların arasından sadece en başarılıların alınması sayesinde azaltılması planlanmıştır.

Cümle üretimlerinin hepsi 1 GB 3 *epoch* eğitim sonrasında oluşturduğumuz modelle yapılmıştır. Sınıflandırma başarılarını ölçerken kendi modelimizin yanında karşılaştırma yapmak amacıyla internette hazır bulunan Türkçe 35 GB veri setiyle önceden eğitilmiş BERT modeli [11] kullanılmıştır. Aşağıdaki tablolarda üç sınıflandırma veri seti için de üstteki işlemler uygulandıktan sonra bulunan sonuçlar gösterilmektedir.

Tablo 8.1 Duygu Veri Setinde Cümle Üretimi Sonucunda Ölçülen Başarı Oranları

	Transformatör		**BERT**	
	SVM	**Log. Reg.**	**SVM**	**Log. Reg.**
Orijinal Veri Seti	%68,1	%63,6	%42	%59,2
%100 Üretim - %50 Veri Setine Ekleme	%67,2	%61,8	%43	%56
%100 Üretim - %100 Veri Setine Ekleme	%63,2	%57,1	%43	%51,7
%500 Üretim - %50 Veri Setine Ekleme	%76,4	%65,6	%48,7	%59,5

Tablo 8.2 Haber Veri Setinde Cümle Üretimi Sonucunda Ölçülen Başarı Oranları

	Transformatör		**BERT**	
	SVM	**Log. Reg.**	**SVM**	**Log. Reg.**
Orijinal Veri Seti	%67	%67	%82,3	%90,4
%100 Üretim - %50 Veri Setine Ekleme	%70,2	%67,2	%83	%89
%100 Üretim - %100 Veri Setine Ekleme	%70,4	%64,6	%78,4	%84,4
%500 Üretim - %50 Veri Setine Ekleme	%76	%71,4	%81,3	%91,5

Tablo 8.3 Deprem *Tweet*leri Veri Setinde Cümle Üretimi Sonucunda Ölçülen Başarı Oranları

	Transformatör		**BERT**	
	SVM	**Log. Reg.**	**SVM**	**Log. Reg.**
Orijinal Veri Seti	%88,6	%86,6	%92,6	%79,3
%100 Üretim - %50 Veri Setine Ekleme	%89,2	%87,3	%88	%93
%100 Üretim - %100 Veri Setine Ekleme	%89,7	%90,4	%75	%93
%500 Üretim - %50 Veri Setine Ekleme	%95,4	%95	%93,3	%95,5

Tablolar incelendiğinde birinci ve ikinci yöntemde düşüş ya da artış yaşanması önceden de belirtildiği gibi modelin cümle üretirken her zaman başarılı olamamasıdır.

Birinci yöntemde üretilen cümlelerin üçüncü yöntemden daha az olması sebebiyle, ikinci yöntemde ise üretilen tüm cümlelerin veri setine tamamen geri eklenmesi sebebiyle bu satırlardaki ölçümlerde düşüş ya da bazen az da olsa bir artış gözlemlenmesi beklenen bir değişimdir.

Son yöntem ile zenginleştirilen bütün veri setlerine baktığımızda sınıflandırma başarı yüzdesinde orijinal haline göre gözle görülebilir bir artış bulunması bu proje kapsamında önemli bir sonuçtur.

9
Sonuç

Bu proje kapsamında doğal dil işlemede yaygın olarak kullanılan çeşitli modeller karşılaştırılmış ve bunların sonuçları tablolar halinde kaydedilmiştir. Karşılaştırmalar sonucunda varyasyonel otokodlayıcı kullanan transformatör mimarisinde bir model oluşturulmuştur.

Oluşturulan model çeşitli parametreleri üzerine test edilerek geliştirilmiştir. Ardından Türkçe veri setlerindeki cümlelerden eş/benzer anlamlı cümleler üretilip bu veri setlerinin zenginleştirilmesi sağlanmıştır. Zenginleştirilen veri setleriyle sınıflandırma başarıları ölçüldüğünde, orijinal veri setinin sonuçlarına göre başarı oranında artış görülmüştür. Bu sonuçlar projenin hedefini destekleyen nitelikte olup daha da artırılabilmesi mümkün görülmektedir.

Modelin 1GB veri setiyle eğitilirken RAM boyutundaki yetersizlikten dolayı parçalara bölünerek eğitilmesi, fazla uzun cümlelerin işlenme süresi uzun olduğu için sadece ilk 60 tokeninin alınması, modeli Google'ın ücretli olarak sunduğu yüksek kaliteli GPU'ları kullanılarak 1 epoch eğitiminin bile yaklaşık 8 saat sürmesi bize bu süreçte bazı zorluklar yaratmıştır.

Elimizdeki kısıtlı imkanlar yüzünden modelin istenilen şartlarda eğitilememesinden dolayı oluşturulan cümlelerin hepsi kaliteli cümleler değildir. Daha büyük ve çeşitli cümleler içeren bir veri setiyle, *epoch* sayısı artırılarak, eğitilen modelden üretilen cümlelerin kalitesinin yükselmesi sağlanabilir. Bu sayede veri setleri daha çok miktarda ve başarılı cümleler ile zenginleştirilebileceği düşünülmektedir.

Ayrıca kullanıcının modele dosya şeklinde verdiği cümlelerin eş/benzer anlamlı oluşturulan cümle çıktılarını yine dosya şeklinde alabilmesini sağlayan bir demo kodu yazılmıştır.

Referanslar

[1] X. Zhang, Y. Yang, S. Yuan, D. Shen, and L. Carin, "Syntax-infused variational autoencoder for text generation," *arXiv preprint arXiv:1906.02181*, 2019.

[2] S. R. Bowman, L. Vilnis, O. Vinyals, A. M. Dai, R. Jozefowicz, and S. Bengio, "Generating sentences from a continuous space," *arXiv preprint arXiv:1511.06349*, 2015.

[3] M. Ş. Bilici and M. F. Amasyali, "Variational sentence augmentation for masked language modeling," in *2021 Innovations in Intelligent Systems and Applications Conference (ASYU)*, IEEE, 2021, pp. 1–5.

[4] Google. "Colaboratory nedir?" (no date), [Online]. Available: `https://research.google.com/colaboratory/intl/tr/faq.html` (visited on 04/29/2023).

[5] D. Monsters, *7 types of artificial neural networks for natural language processing*, Sep. 2017. [Online]. Available: `https://medium.com/@datamonsters/artificial-neural-networks-for-natural-language-processing-part-1-64ca9ebfa3b2`.

[6] M. VARER, *Rnn, lstm ve gru modellerinin incelemesi*, May 2020. [Online]. Available: `https://medium.com/@mcvarer/rnn-lstm-ve-gru-modellerinin-incelemesi-f59a73499edb`.

[7] M. Phi, *Illustrated guide to lstm's and gru's: A step by step explanation*, Jun. 2020. [Online]. Available: `https://towardsdatascience.com/illustrated-guide-to-lstms-and-gru-s-a-step-by-step-explanation-44e9eb85bf21`.

[8] N. Pogeant, *Transformers-the nlp revolution*, Dec. 2022. [Online]. Available: `https://medium.com/mlearning-ai/transformers-the-nlp-revolution-5c3b6123cfb4`.

[9] A. Vaswani *et al.*, "Attention is all you need," 2017. [Online]. Available: `https://arxiv.org/pdf/1706.03762.pdf`.

[10] K. T. Uçar, *Bert modeli ile türkçe metinlerde sınıflandırma yapmak*, Jan. 2022. [Online]. Available: `https://medium.com/@toprakucar/bert-modeli-ile-t%C3%BCrk%C3%A7e-metinlerde-s%C4%B1n%C4%B1fland%C4%B1rma-yapmak-260f15a65611`.

[11] *Dbmdz/bert-base-turkish-cased · hugging face*, Feb. 2020. [Online]. Available: `https://huggingface.co/dbmdz/bert-base-turkish-cased`.

[12] [Online]. Available: `http://www.kemik.yildiz.edu.tr/veri_kumelerimiz.html`.

[13] A. Guven, *Turkish tweets dataset*, 2021. [Online]. Available: https://www.kaggle.com/datasets/anil1055/turkish-tweet-dataset.

[14] Ü. Küçüktaş, *Turkey earthquake relief tweets dataset*, 2023. [Online]. Available: https://www.kaggle.com/datasets/ulkutuncerkucuktas/turkey-earthquake-relief-tweets-dataset.

Özgeçmiş

BİRİNCİ ÜYE

İsim-Soyisim: Engin MEMİŞ
Doğum Tarihi ve Yeri: 30.03.2000, İstanbul
E-mail: engin.memis@std.yildiz.edu.tr
Telefon: 0534 244 87 84
Staj Tecrübeleri: Yıldız Teknik Üniversitesi Olasılıksal Robotik Araştırma Grubu

İKİNCİ ÜYE

İsim-Soyisim: Elif Sena YILMAZ
Doğum Tarihi ve Yeri: 21.05.2001, Ankara
E-mail: sena.yilmaz4@std.yildiz.edu.tr
Telefon: 0551 251 35 54
Staj Tecrübeleri: -

Proje Sistem Bilgileri

Sistem ve Yazılım: Windows İşletim Sistemi, Python, Google Colab.
Gerekli RAM: 16 GB
Gerekli Disk: 5 GB