

T.C.

TRAKYA ÜNİVERSİTESİ

FEN BİLİMLERİ ENSTİTÜSÜ

DOĞAL DİL İŞLEME ALANINDAKİ DERİN ÖĞRENME MODELLERİNİN
KARŞILAŞTIRILMASI

TUNA TOPAÇ

YÜKSEK LİSANS TEZİ

BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI

Tez Danışmanı: Dr. Öğr. Üyesi Altan MESUT

EDİRNE-2024

TUNA TOPAÇ'ın hazırladığı “**DOĞAL DİL İŞLEME ALANINDAKİ DERİN ÖĞRENME MODELLERİNİN KARŞILAŞTIRILMASI**” başlıklı bu tez, tarafımızca okunmuş, kapsam ve niteliği açısından **Bilgisayar Mühendisliği** Anabilim Dalında bir **Yüksek lisans tezi** olarak kabul edilmiştir.

Jüri Üyeleri

İmza

Dr. Öğr. Üyesi Altan MESUT

.....

Prof. Dr. Erdiñç UZUN

.....

Dr. Öğr. Üyesi Emir ÖZTÜRK

.....

Tez Savunma Tarihi: 25/07/2024

Bu tezin Yüksek Lisans tezi olarak gerekli şartları sağladığımı onaylarım.

İmza

Dr. Öğr. Üyesi Altan MESUT

Tez Danışmanı

.....

Trakya Üniversitesi Fen Bilimleri Enstitüsü onayı

.....

Prof. Dr. Muharrem Tolga SAKALLI

Fen Bilimleri Enstitüsü Müdürü

T. Ü. FEN BİLİMLERİ ENSTİTÜSÜ

**BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI YÜKSEK LİSANS
PROGRAMI**

DOĞRULUK BEYANI

Trakya Üniversitesi Fen Bilimleri Enstitüsü, tez yazım kurallarına uygun olarak hazırladığım bu tez çalışmasında, tüm verilerin bilimsel ve akademik kurallar çerçevesinde elde edildiğini, kullanılan verilerde tahrifat yapılmadığını, tezin akademik ve etik kurallara uygun olarak yazıldığını, kullanılan tüm literatür bilgilerinin bilimsel normlara uygun bir şekilde kaynak gösterilerek ilgili tezde yer aldığını ve bu tezin tamamı ya da herhangi bir bölümünün daha önceden Trakya Üniversitesi ya da farklı bir üniversitede tez çalışması olarak sunulmadığını beyan ederim.

25/07/2024

Tuna TOPAÇ

İmza

Yüksek Lisans Tezi

Doğal Dil İşleme Alanındaki Derin Öğrenme Modellerinin Karşılaştırılması

T.Ü. Fen Bilimleri Enstitüsü

Bilgisayar Mühendisliği Anabilim Dalı

ÖZET

Yapay zekanın günümüzde hayatın işleyişine verimlilik ve üretkenlik kavramlarını katarak yön verdiğini söylemek yanlış olmaz. Bununla birlikte yapay zekanın bir alt dalı olan makine öğrenimi özellikle son yirmi yılda oldukça ön plana çıkmış ve teknolojik olarak en çok talep gören alan olmuştur. Yapay zeka ve özellikle makine öğrenimi, bu alanda çalışan şirket ve organizasyonların rekabet ve başarılarında da kendini göstermiştir. Makine öğrenimi, bilgisayarların belirli görevleri yerine getirmek için veriler üzerinden eğitim yoluyla edindikleri modellerin kullanılmasıdır. Bu görevleri kullanan alanlardan bir tanesi doğal dil işlemedir. Doğal dil işleme alanı, insan dillerinin hesaplamalı işlenmesini ve anlaşılmasını içeren çeşitli konuları kapsar. Son birkaç yılda, doğal dil işleme alanı, derin öğrenme modellerinin kullanımı ile daha da artmıştır. Bu tez çalışması, doğal dil işleme alanının metin üretme (text generation) alt alanında derin öğrenme mimarileri ve yöntemlerinin performans açısından karşılaştırılmasını ele almaktadır. Bu kapsamda huggingface sitesinden indirilen 7 dönüştürücü modelin metin üretme işlevleri kullanılarak bu alanda kullanılan ölçme metrikleri ile benzerlik oranları hesaplanmıştır. Amaç, referans metinler ile modeller tarafından üretilen metinlerin birbirleri ile benzerliği ve anlam bütünlüğünün ölçülmesidir. Yapılan deneyler, model büyüklüğünün performansa etkisine çok önemli bir şekilde etki ettiğini gösterdi. 5 ölçüm metriği arasında dönüştürücü tabanlı olan BertScore' un n-gram tabanlı metriklere göre daha başarılı ölçüm yaptığı görüldü.

Yıl : 2024

Sayfa Sayısı : 122

Anahtar Kelimeler : Doğal Dil İşleme, Metin Üretimi, Hesaplama Metrikleri

Master's Thesis

Comparison of Deep Learning Models in Natural Language Processing

Trakya University Institute of Natural and Applied Sciences

Department of Computer Engineering

ABSTRACT

Artificial intelligence has shaped the functioning of life today by adding the concepts of efficiency and productivity. However, machine learning, a sub-branch of artificial intelligence, has come to the forefront especially in the last two decades and has become the most demanded technological field. Artificial intelligence and especially machine learning has also manifested itself in the competition and success of companies and organizations working in this field. Machine learning is the use of models that computers have obtained through training over data to perform certain tasks. One of the fields that use these tasks is natural language processing. The field of natural language processing covers a variety of topics involving the computational processing and understanding of human languages. In the last few years, the field of natural language processing has been further enhanced by the use of deep learning models. This thesis presents a performance comparison of deep learning architectures and methods in the text generation subdomain of natural language processing. In this context, the text generation functions of 7 transformative models downloaded from the huggingface website were used to calculate similarity ratios with the measurement metrics used in this field. The aim was to measure the similarity and semantic integrity between the reference texts and the texts produced by the models. The experiments showed that model size has a significant effect on performance. Among the 5 measurement metrics, BertScore, which is transducer-based, was found to be more successful than n-gram-based metrics.

Year : 2024

Number of Pages : 122

Keywords : Natural Language Processing, Text Generation, Evaluation Metrics

TEŞEKKÜR

Bu çalışmayı titizlikle yöneten, her türlü teşvik ve fedakârlığını esirgemeyen, manevi desteğini her an hissettiğim, saygıdeğer hocam Dr. Öğr. Üyesi Altan MESUT'a sonsuz teşekkürlerimi sunarım.

Tez çalışması sürecinde görüş ve önerileriyle büyük katkı sağlayan Dr. Öğr. Üyesi Emir ÖZTÜRK'e saygı ve sevgilerimi sunarak teşekkür ederim.

Bilgisi, deneyimi ve güler yüzüyle her alanda verdiği destek ve değerli dostluğu için Prof. Dr. Erdem Uçar'a en derin sevgi ve saygılarımla teşekkür ederim.

Sadece yüksek lisans eğitimim süresince değil, hayatım boyunca en büyük desteğim olan annem Sıdika TOPAÇ ve babam Zafer TOPAÇ'a en derin sevgimle teşekkürlerimi sunarım.

İÇİNDEKİLER

ÖZET	iv
ABSTRACT.....	v
TEŞEKKÜR.....	vi
İÇİNDEKİLER	vii
ŞEKİLLER DİZİNİ.....	xi
ÇİZELGELER DİZİNİ	xii
BÖLÜM 1 GİRİŞ.....	1
BÖLÜM 2 MAKİNE ÖĞRENİMİ	3
2.1. Denetimli Öğrenme.....	3
2.2. Denetimsiz Öğrenme.....	4
2.3. Yarı Denetimli Öğrenme.....	5
2.4. Pekiştirmeli Öğrenme.....	6
2.5. Derin Öğrenme.....	6
BÖLÜM 3 DOĞAL DİL İŞLEME	8
3.1. Doğal Dil İşlemenin Bileşenleri.....	9
3.2. Doğal Dil İşlemenin Avantajları ve Dezavantajları	9
3.3. Doğal Dil İşlemenin Aşamaları.....	10
3.4. Genel Uygulamalar	10
3.4.1. Otomatik Aramada Düzeltme	10
3.4.2. Dil Çevirici.....	11

3.4.3. Dilbilgisi Kontrol Araçları	11
3.4.4. Hedeflenmiş Reklamcılık.....	11
3.4.5. ChatBot (Sohbet Robotları)	12
3.4.6. Sesli Asistan.....	12
BÖLÜM 4 METİN OLUŞTURMA.....	13
4.1. Metin Oluşturmada Kullanılan Mimariler	15
4.1.1. RNN (Yinelemeli Sinir Ağları-Recurrent Neural Network).....	15
4.1.2. LSTM (Uzun Kısa Vadeli Hafıza-Long Short Term Memory)	18
4.1.3. CNN (Evrşimsel Sinir Ağları-Convolutional Neural Networks).....	21
4.1.4. GRU (Geçitli Yinelemeli Birim- Gated Recurrent Unit).....	23
4.1.5. GAN (Çekişmeli Üretici Ağ- Generative Adversarial Network).....	25
4.1.6. Dönüştürücüler (Transformers).....	27
4.1.6.1. Dönüştürücü Mimarisi	30
4.1.6.1.1. Kodlayıcı ve Kod Çözücü Yığını.....	31
4.1.6.1.2. Dikkat Fonksiyonu	32
4.1.6.1.3. Çok Taraflı Dikkat	33
4.1.6.1.4. Konumsal Kodlama.....	34
4.1.6.1.5. Gömülü Kelime Simgeleri	35
4.1.6.1.6. Tokenization (Simgeleştirme).....	36
4.1.6.2. Tez Kapsamında Kullanılan Büyük Dil Modelleri	39
4.1.6.2.1. PHI-2	40
4.1.6.2.2. MISTRAL 7B	41
4.1.6.2.3. GPT-2.....	44
4.1.6.2.4. XGLM.....	44
4.1.6.2.5. LLAMA 2	45

4.1.6.3. Popüler Üretici Büyük Dil Modelleri.....	45
4.1.6.3.1. GPT-3.....	45
4.1.6.3.2. GPT-4.....	46
4.1.6.4. Gelişmiş Dil Model Uygulamaları	47
4.1.6.4.1. ChatGPT.....	47
4.1.6.4.2. Gemini.....	49
4.2. Dil Modellerinin Performansını Ölçmede Kullanılan Metrikler.....	51
4.2.1. N-gram	54
4.2.2. Accuracy (Doğruluk), Precision (Kesinlik), Recall (Duyarlılık) ve F1 Score metrikleri.....	55
4.2.3. BLEU (Bilingual Evaluation Understudy).....	56
4.2.4. METEOR (Metric for Evaluation of Translation with Explicit ORDERing) ..	58
4.2.5. ROUGE (Recall-Oriented Understudy for Gisting Evaluation)	59
4.2.5.1. ROUGE-N.....	60
4.2.5.2. ROUGE-L	60
4.2.6. BERTSCORE.....	61
BÖLÜM 5 DENEYSEL YÖNTEM VE SONUÇLAR.....	64
5.1. Kullanılan Modeller	65
5.2. Girdi ve Referans Olarak Kullanılan Metinler.....	66
5.3. Modellerin Metin Üretme Süreleri.....	69
5.4. Üretilen Metinlerin Başarısının Hesaplanması	69
BÖLÜM 6 SONUÇ.....	87
KAYNAKLAR	89
ÖZGEÇMİŞ	107
EKLER.....	108

EK: Hesaplama metrikleri için kullanılan kodlar.....	108
--	-----



ŞEKİLLER DİZİNİ

Şekil 4.1. Bir zaman adımında metin işlemeye yönelik RNN mimarisi	17
Şekil 4.2. LSTM hücresinin çalışma biçimi	19
Şekil 4.4. GRU mimarisindeki bir hücrenin gösterimi	24
Şekil 4.5. RNN, LSTM, GRU mimarilerindeki metinleri vektörlere dönüştürüp kullanırken oluşan şekil	25
Şekil 4.6. GAN mimarisinin genel gösterimi	26
Şekil 4.7. Dönüştürücü mimarisi	30
Şekil 4.8. Dikkat fonksiyonu hesaplanırken cümle içindeki kelimelerin birbiri ile ilişkileri göz önünde bulundurulur	32
Şekil 4.9. Dikkat matrisleri	33
Şekil 4.10. Solda dikkat fonksiyonun gösterimi, sağda ise bunun çoklu dikkatte kullanımı gösterilmiştir.	34
Şekil 4.11. Kayan pencere dikkati	42
Şekil 4.12. İlerleyen tampon gösterimi	43
Şekil 4.13. Dikkat maskesinin hem önbellek hem de yığın (parça) üzerinde nasıl çalıştığının gösterimi	43
Şekil 4.14. Karmaşıklık matrisi gösterimi	56
Şekil 5.1. Huggingface sitesinden modellerin indirilmesini sağlayan Python kodu	65
Şekil 5.2. Metin üretmede kullanılan Python kod satırı	66
Şekil 5.3. Türkçe metin üretmede elde edilen puanların ortalamaları	85
Şekil 5.4. İngilizce metin üretmede elde edilen puanların ortalamaları	86

ÇİZELGELER DİZİNİ

Çizelge 5.1. Tezde kullanılan modellerin genel özellikleri.....	66
Çizelge 5.2. Türkçe modeller için verilen girdi metinler ve kullanılan referans metinler	67
Çizelge 5.3. İngilizce modeller için verilen girdi metinler ve kullanılan referans metinler	68
Çizelge 5.4. Modellerin 5 farklı girdi metni için toplam metin üretme süreleri	69
Çizelge 5.5. “1963 Türkiye” girdi metni için referans metin ve üretilen metinler	70
Çizelge 5.6. “1963 Türkiye” girdi metni için elde edilen değerlendirme puanları	71
Çizelge 5.7. “İkinci Berberi Savaşı, 1815-1816” girdi metni için referans metin ve üretilen metinler	72
Çizelge 5.8. “İkinci Berberi Savaşı, 1815-1816” girdi metni için elde edilen değerlendirme puanları	72
Çizelge 5.9. “Disco Inferno, Amerikalı rapçi 50 Cent” girdi metni için referans metin ve üretilen metinler	73
Çizelge 5.10. “Disco Inferno, Amerikalı rapçi 50 Cent” girdi metni için elde edilen değerlendirme puanları	73
Çizelge 5.11. “Goeben ve Breslau'nun takibi, I. Dünya” girdi metni için referans metin ve üretilen metinler	74
Çizelge 5.12. “Goeben ve Breslau'nun takibi, I. Dünya” girdi metni için elde edilen değerlendirme puanları	75
Çizelge 5.13. “Karayılan, Gaziantep'in Fransız işgalinden kurtuluşunu konu alan dram” girdi metni için referans metin ve üretilen metinler	76

Çizelge 5.14. “Karayılan, Gaziantep'in Fransız işgalinden kurtuluşunu konu alan dram” girdi metni için elde edilen değerlendirme puanları	76
Çizelge 5.15. “Diplomatic capital” girdi metni için referans metin ve üretilen metinler	77
Çizelge 5.16. “Diplomatic capital” girdi metni için elde edilen değerlendirme puanları	78
Çizelge 5.17. “Herbicide safeners are organic” girdi metni için referans metin ve üretilen metinler	79
Çizelge 5.18. “Herbicide safeners are organic” girdi metni için elde edilen değerlendirme puanları	80
Çizelge 5.19. “The Ghanaian people are a nation” girdi metni için referans metin ve üretilen metinler	81
Çizelge 5.20. “The Ghanaian people are a nation” girdi metni için elde edilen değerlendirme puanları	81
Çizelge 5.21. “Fort Kelburne was a coastal artillery” girdi metni için referans metin ve üretilen metinler	82
Çizelge 5.22. “Fort Kelburne was a coastal artillery” girdi metni için elde edilen değerlendirme puanları	83
Çizelge 5.23. “Arkeisha Antoinette Knight, known professionally as Kash Doll, is” girdi metni için referans metin ve üretilen metinler	84
Çizelge 5.24. “Arkeisha Antoinette Knight, known professionally as Kash Doll, is” girdi metni için elde edilen değerlendirme puanları	84

BÖLÜM 1

GİRİŞ

İnsanoğlu dünyadaki en istisnai türdür; bunun nedeni, veri aktarma ve paylaşma becerisidir. Bilgi çağında olduğumuz düşünüldüğünde veri insanlık için modern çağın petrolü olarak görülebilir. Verinin petrolden farkı tükenmemesi, aksine gitgide çoğalmasındır. Verinin büyük kaynağı insan ve insan nedenli olaylardır. Veri her yerde büyük ölçekte internette özellikle kitap, makale gibi belgelerin taranması ile X, WhatsApp veya Facebook gibi farklı uygulamalar üzerinden mesajlar gönderilerek oluşturulmakta; burada verinin büyük bir kısmı doğası gereği oldukça yapılandırılmamış olan metin biçiminde bulunmaktadır. Metin verileri M.Ö. 4000’li yıllarda Sümerlerin yazıyı bulmasıyla başlamıştır. Metin, günlük hayatta her türlü insani yapıda en basit mesajlaşmadan en hayati bilginin taşınmasına kadar birçok konuda önemli bir rol oynamaktadır. Metnin iletiminin yanında işlenmesi son zamanlarda ilgi çeken konular arasında yer almaktadır. Metin işlemede, doğal dil işleme ön plana çıkmaktadır. Bu sebeple verilerden önemli miktarda bilgi yaratabilmek için doğal dil işleme yöntemlerini tanımak önemlidir. Doğal dil işleme, dili otomatik olarak temsil ve analiz ettiğimiz hesaplamalı tekniktir. Cümle gösterimini öğrenmek birçok doğal dil uygulamasının temelidir. Bilgi ve iletişim teknolojilerindeki hızlı gelişmeler nedeniyle doğal dil işleme araçları büyük ilgi uyandırmaktadır. Bunun sonucunda pek çok farklı doğal dil işleme aracı üretilmiştir. Ancak doğal dilleri doğru şekilde işleyen verimli ve etkili doğal dil işleme araçlarını geliştirmenin birçok zorluğu vardır. Son zamanlarda doğal dil işleme bağlamında derin öğrenme modeli tasarımları ve mimarileri ortaya çıkmıştır. Derin öğrenme yöntemleri, algoritma için gerekli girdilerin işlem gördüğü birbiriyle ilişkili birçok katmana sahiptir ve bu katmanlar girilen verilerin temsili anlamada önemli yere

sahiptir. Bu sebeple, bu yöntemler doğal dil işleme alanında kayda değer sonuçlar elde etmiştir.



BÖLÜM 2

MAKİNE ÖĞRENİMİ

Makine öğrenimi, yapay zekanın bir alt alanıdır ve bu terim, bilgi teknolojisi sistemlerinin veri tabanlarındaki kalıpları tanıyarak sorunlara bağımsız olarak çözüm bulma yeteneğini ifade eder. Makine öğrenimi, bilgi teknolojisi sistemlerinin mevcut algoritmalar ve veri kümeleri temelinde kalıpları tanımasını ve yeterli çözüm konseptleri geliştirmesini sağlar. Makine Öğreniminde yapay bilgi, deneyim temelinde üretilir. Bu teknolojiyi kullanan sistem, makine öğrenimi yoluyla şu görevleri yerine getirebilir: İlgili verileri bulma, çıkarma ve özetleme, analiz verilerine dayanarak tahminler yapma, belirli sonuçlar için olasılıkları hesaplama, belirli gelişmelere otonom olarak uyum sağlama, tanınan kalıplara göre süreçleri optimize etme. Makine öğrenimi, bilgisayarların programlanmadan bir görevi nasıl "düşündüğü" ve yürüttüğüdür. Bilinçli ve bilinçaltı idrak seviyelerine bağlı olarak, insan müdahalesi olmadan çıkarımlar ve kararlar vermek için verileri otomatik olarak analiz edebilen ve öğrenebilen algoritmalar ve modeller içeren bir yapay zeka alt kümesidir. Makine öğreniminin üç ana kullanım şekli vardır: denetimli öğrenme, denetimsiz öğrenme ve takviyeli öğrenme. Bazı durumlarda dördüncü bir daldan söz edilir (yarı denetimli öğrenme). Ancak bu, aslında denetimli öğrenmenin sadece özel bir örneğidir. (Nedal & Morgan, 2018).

2.1. Denetimli Öğrenme

Denetimli öğrenme, makine öğreniminin açık ara en yaygın kullanım şeklidir. Şu anda makine öğrenimi alanında gerçek dünya değerinin çoğu, denetimli öğrenmeye atfedilebilir. Denetimli öğrenme algoritmaları, etiketli örneklerle eğitilen makine öğrenme algoritmalarıdır. Denetimli öğrenme algoritmalarıyla uğraşırken anlaşılması

gereken ana fikir, öğrenmeleri gerekenleri algoritmalarla göstermek için açıkça eklenmiş örneklerin kullanımı yoluyla öğrenmeleridir. Bu nedenle algoritmalar, etiketleri kılavuz olarak kullanarak girdilerden çıktılara bir eşleme gösterimi bulmaya çalışır. Bu tür algoritmaların adındaki “Denetimli”, etiketlerin veya hedeflerin öğrenme süreci boyunca denetim sağladığına işaret etmektedir. Bu nedenle algoritmanın tahminini etiketlerde saklanan gerçek değerlere göre kontrol etmesi mümkündür. Denetimli öğrenme problemindeki hedefler, algoritmanın performansını iyileştirebileceği alanlar hakkında geri bildirim sağlayan bir denetleyici olarak görülebilir. Denetimli öğrenme algoritmalarının iki ana uygulaması sınıflandırma ve regresyondur (Nedal & Morgan, 2018).

Sınıflandırma, örnekleri önceden tanımlanmış kategorilere veya sınıflara doğru şekilde ayırmak için bir öğrenme algoritmasının eğitilmesini içerir Regresyon ise, algoritmanın tek bir gerçek değer sayısını tahmin etmekle ilgilendiği bir öğrenme problemidir (Bramahhazela vd., 2022).

2.2. Denetimsiz Öğrenme

Denetimsiz öğrenme, doğrudan ham verilerden öğrenmeyi içerir. Bu tür öğrenme, eğitim döngüsünde etiketler şeklinde bir denetleyicinin varlığı olmadan gerçekleşir. Denetimsiz öğrenme algoritmaları, temeldeki veri dağıtımını keşfetmekte ve tüm veri kümesini en iyi tanımlayan modelleri bulmakta özgürdür. Eğitim süreci, etiketlenmiş örnekler aracılığıyla insanlar tarafından yönlendirilmez ve bu nedenle denetimsiz öğrenme algoritmaları, alan uzmanlarının düşünemeyeceği kalıpları keşfedebilecekleri için daha güçlüdür. Bununla birlikte, keşfedilen kalıpları anlamak ve bunları açıklamak hâlâ alan uzmanlarının işidir; çünkü denetimsiz öğrenme algoritmaları, insanlara atfedeceğimiz muhakeme anlayışına gerçek anlamda sahip değildir. Denetimsiz öğrenme algoritmaları, kümeler, gruplar veya dağılımlar biçiminde olabilecek durumları ortaya çıkarmak için yalnızca veri dağılımını veya verilerin gizli olan temsillerini kullanır. Denetimsiz öğrenme algoritmalarının kümeleme, boyut tespiti, üretken modeller vb. gibi birçok uygulaması vardır (Nedal & Morgan, 2018).

Kümeleme, denetimsiz öğrenmenin popüler uygulamalarından biridir. Ham verilerden veri noktası gruplarının (kümelerinin) otomatik olarak keşfedilmesini içerir.

Bir grubun üyeleri benzer özellikleri paylaşırlar, yani aynıdır. Aynı tür varlığa ait oldukları düşünülebilirken, grup olarak diğer gruplardan farklıdır (Yan & Wang, 2022).

Boyut azaltma, bir modelde beslenen öznel özelliklerin (özelliklerin) sayısını yalnızca tahminleri yönlendiren en alakalı olanlara indirgeyen bir makine öğrenme tekniğidir. Daha fazla özellik veya boyuta sahip modellerin genellemede daha kötü performans gösterdiği ampirik olarak gözlemlenmiştir. Bu sebeple modellerin gerçek dünyada performansları düşmektedir. Bir model, boyutların sayısını azaltarak, tahmine yardımcı olan veriler hakkında geçerli temsiller geliştirmesine olanak tanıyan bilgileri geriye kalan özelliklerden öğrenebilir. Temel Bileşen Analizi muhtemelen en popüler boyut azaltma tekniğidir ve denetimsiz öğrenme algoritmasının bir örneğidir (Topac & Qamar, 2020).

Üretken modeller, denetimsiz öğrenme algoritmalarının bir başka popüler örneğidir. Yapay olarak gerçekçi görünen fotoğraflar ve sanat eserleri üretme yetenekleri nedeniyle son zamanlarda ön plana çıkmışlardır. Üretken Çekişmeli Ağlar şu anda birçok görüntü oluşturma kıyaslamasında en gelişmiş sonuçları ürettiyor ve üretken modellerin en popüler örnekleri arasında yer almaktadır (Lu vd., 2022).

2.3. Yarı Denetimli Öğrenme

Yarı denetimli öğrenme algoritmaları, denetimli öğrenme algoritmalarının özel bir durumudur. Yarı denetimli öğrenmede açık bir etiket olmasa da eğitim döngüsünde denetleyici görevi gören örtülü bir öğretici yöntem mevcuttur. Yarı denetimli modeller herhangi bir harici etiket kaynağı içermez, yalnızca giriş özelliklerine dayanır. Bununla birlikte, öğrenme görevi, denetimin hâlâ sezgisel bir algoritma yoluyla girdilerden sahte etiketlerin çıkarılması şeklinde gerçekleşeceği şekilde ayarlanmıştır. Yarı denetimli öğrenme algoritmalarının popüler bir örneği otomatik kodlayıcılardır (Yin vd., 2022).

Otomatik kodlayıcıda öğrenme görevi, girdinin boyutlarını en önemli gizli özellikleri temsil eden daha küçük bir gizli alana indirmek, ardından girdiyi bu daha düşük boyutlu uzaydan yeniden oluşturmaktır. Bir girdi verildiğinde, örneğin bir görüntüde, bir otomatik kodlayıcı, görüntüyü, görüntü hakkındaki bilgilerin çoğunu hâlâ içeren daha küçük bir gizli gösterime küçültür ve ardından bu düşük boyutlu uzaydan orijinal girdi görüntüsünü yeniden oluşturur. Açık bir etiket olmasa bile öğrenme görevi

girdiyi yeniden yapılandırmak olduğundan girdinin denetleyici görevi gördüğü görülecektir. Böyle bir model, özellikleri daha küçük bir boyuta sıkıştırmak üzere eğitildikten sonra sıkıştırılmış özellikler, Temel Bileşen Analizi kullanılarak boyut azaltımına benzer bir denetimli öğrenme algoritmasının başlangıç noktası olarak hizmet edebilir (C. Zhang vd., 2022).

2.4. Pekiştirmeli Öğrenme

Pekiştirmeli öğrenmede üç ana bileşen vardır: bir etmen, bir ortam ve eylemler. Resmi olarak pekiştirmeli öğrenme, Markov Karar Süreci olarak tanımlanır. Pekiştirmeli öğrenmenin amacı, çevresinde gözlemlenebilen ve nihai hedefe ulaşma şansını en üst düzeye çıkaran eylemler gerçekleştirebilen akıllı bir aracı yetiştirmektir. Temsilci tarafından gerçekleştirilen eylemler ortamın durumunu değiştirir ve temsilci tarafından gerçekleştirilen eylemlere bağlı olarak ödüller veya cezalar verilebilir. Temsilci için zorluk, nihai hedefi gerçekleştirebilmek için belirli bir sürenin sonunda biriken ödülleri en üst düzeye çıkarmaktır (Rolf vd., 2023).

2.5. Derin Öğrenme

Makine öğrenimi problemlerini çözmeye yönelik gelişmiş bir yapay zeka tekniği olan derin öğrenme, birçok gerçek dünya uygulamasında büyük başarı elde etmiş ve son yıllarda giderek daha fazla ilgi görmeye başlamıştır. Ancak derin öğrenmenin performansı mimari ve hiper parametreler gibi birçok faktöre bağlı olduğundan derin öğrenmenin nasıl optimize edileceği derin öğrenme ve yapay zeka alanında sıcak bir araştırma konusu haline gelmiştir (Zhan vd., 2022). Derin öğrenme, verilerdeki özellik temsillerini otomatik olarak keşfetmek için özellik öğrenmeyi gerçekleştiren algoritmaları kullanan, makine öğreniminin bir alt alanıdır. Derin öğrenmedeki "derin", bu tür öğrenmenin, bilgileri temsil eden vektör zincirinde daha önce tespit edilen daha basit özelliklerden oluşan daha yüksek seviyeli özellik temsilleriyle birkaç katmanda yapıldığı gerçeğini ifade eder. Derin öğrenmede kullanılan algoritma, birden fazla katmandan oluşan derin bir sinir ağıdır. Derin öğrenme aynı zamanda hiyerarşik öğrenme olarak da bilinir çünkü özelliklerin hiyerarşisini öğrenir. Ağda derinlere doğru ilerledikçe öğrenilen özelliklerin karmaşıklığı artar. Derin öğrenme teknikleri, genel olarak beynin nasıl çalıştığına dair şu anda bilinenlerden ilham almaktadır. Ana fikir, beynin, kimyasal

reaksiyonlardan gelen elektrik sinyallerini kullanarak bir şekilde birbirleriyle etkileşime giren milyarlarca nörondan oluşmasıdır. Derin öğrenme algoritmalarının yaptığı şey, girdilerden doğrudan çıktılara bir haritalama fonksiyonunu öğrenmek için Yapay Sinir Ağı (YSA) olarak bilinen bir yapıdaki nöronlardan oluşan bir ağı düzenlemektir (Nedal & Morgan, 2018).

Şu anda derin öğrenme yöntemleri, karmaşık veri temsilini yüksek düzeyde soyutlamayla otomatik olarak çıkarmak için kullanılan makine öğreniminin en yaygın yöntemidir ve özellikle son derece karmaşık problemler için kullanılır. Geleneksel yöntemlere göre daha iyi sonuç elde eden oldukça fazla büyük veri kullanabilen bir yaklaşımdır (Chiche & Yitagesu, 2022).



BÖLÜM 3

DOĞAL DİL İŞLEME

Doğal dil işleme, insan ana dilini akıllıca işleyebilen bilgisayar sistemlerinin geliştirilmesiyle ilgili bir araştırma alanıdır. Doğal dil işlemenin ana işlevleri arasında makine çevirisi, konuşma tanıma ve analizi, otomatik özetleme, metin üretme gibi görevler vardır. Bu görevler sayesinde doğal dil işleme, sanal bilgisayar arayüzünü insanlar için kolaylaştırmak için kullanılmaktadır. Doğal Dil İşleme, bilgisayar bilimi alanıdır; bilgisayarlar ve insan dilleri arasında iletişim kurmak için kullanılan yapay zekanın da alt alanıdır diyebiliriz. Doğal dil işleme aslında sıklıkla kullandığımız insan dilini anlamak, analiz etmek ve onu ortak olan tek bilgisayar diline çevirmekle ilgilidir. İlk doğal dil işleme, 1950'lerde yapay zeka ile dil etkileşiminde kullanılmıştır (Johri vd., 2021). Doğal dil işleme, bilgisayarların doğal bir dilin veya konuşmanın yazılı içeriğini pratik amaçlarla anlamak ve yönetmek için nasıl kullanılabileceğini araştıran bir araştırma ve uygulama alanıdır. Doğal dil işleme, yerel insan dilleri ile bilgisayar cihazları arasındaki iletişimi ilgilendiren bir konudur ve insan dilinin otomatik olarak işlenmesi olarak düşünülebilir. Bu bilgisayar bilimi alanı, bilgisayar sistemlerinin insan dilindeki anlamı akıllıca ve kullanışlı bir şekilde analiz etmesinin, anlamasının ve yorumlamasının bir yoludur. Bilgisayarla olan bu etkileşim, arama motoru sonuç sayfası, çeviri sistemleri, soruların otomatik yanıtlanması, dilbilgisi kontrolü, metin ayırma ve çok daha fazlası gibi gerçek dünya uygulamalarına olanak tanır. Doğal dil işleme, test programları, e-öğrenme, iletişim platformu, iş analizi, konuşma tanıma gibi birçok farklı bağlama büyük ölçüde entegre edilmiştir. Bilgisayarlar bilginin hazırlanması, saklanması, analizi ve iletilmesinde önemli bir rol oynadığından, onlara doğal dilde sunulan bilgiyi anlama ve işleme yeteneğinin sağlanması büyük önem taşıyacaktır. Dolayısıyla doğal dil işlemenin

amacı, doğal dilleri (Türkçe, İngilizce vb.) analiz edebilen ve yerel dilde verimli çıktılar üretebilen bilgisayar sistemleri tasarlamak ve oluşturmaktır (Shivahare vd., 2022).

3.1. Doğal Dil İşlemenin Bileşenleri

Doğal Dil Anlama, müşteri hizmetleri otomasyonu, iletişim platformu verileri vb. gibi metinden önemli bilgilerin çıkarılmasına yardımcı olur. Doğal dil anlayışı şu yükümlülükleri içerebilir: Kullanılabilir bir görüntüde sağlanan bir girdi haritası oluşturmak için kullanılır. Dilin çeşitli yönlerini araştırmak için kullanılır.

Doğal dil üretimindeki ana süreçler içerik belirleme, belge planlama ve yorumlama, cümle birleştirme ve dilbilgisi ile dilin ilgili göreve uygulanmasıdır. Bankacılık, Sigorta, Finans sektörleri için kullanılan içeriğin doğal dil üretimi yoluyla özetlenmesi gerekmektedir. GPT-3, LaMDA, her gün kullanılan doğal dil işleme tabanlı otomasyon araçlarıdır. Doğal dil işleme, verilerin daha yüksek kalitede düzenlenmesini gerektiren otomatik içerik sağlamak için algoritmalar kullanır. Veri miktarının artmasıyla birlikte doğal dil oluşturma algoritmaları da daha iyi hale gelmektedir (Aydin & Karaarslan, 2023).

3.2. Doğal Dil İşlemenin Avantajları ve Dezavantajları

Doğal dil işleme, herhangi bir konudaki kullanıcı sorularının saniyeler içinde yanıtlanmasına yardımcı olur. Doğal dil işleme, sorulara doğrudan yanıtlar sağlar, yani yalnızca gerekli bilgileri sağlar. Doğal dil işleme, kullanıcının bilgisayarla kendi dilini kullanarak etkileşime girmesine yardımcı olur. Doğal dil işleme bu sebeple hızlıdır (Omar vd., 2022).

Doğal dil işleme her zaman bağlamı yansıtmaz. Doğal dil işleme bu yüzden belirsizdir. Doğal dil işlemede işlevi kontrol edecek ek modüller gerekebilir. Farklı bir ortama uyum sağlayamamakta ve tek bir işlev için tasarlandığından sınırlı işlevler içerebilir (Omar vd., 2022).

3.3. Doğal Dil İşlemenin Aşamaları

Doğal dil işleme ilgili çalışmada (Shivahare vd., 2022) aşağıdaki beş aşamada ele alınmıştır.

Sözcüksel analiz ve Morfolojik (Biçim Bilimsel): Doğal dil işlemenin ilk aşamasıdır. Bu aşamada kaynak bir harf akışı olarak incelenir ve anahtar metinlere dönüştürülür. Tüm içerik bölümlere, kelimelere ve kelime öbeklerine ayrılır.

Sözdizimsel Analiz: Bu safha, dilbilgisini, kelime işlemeyi değerlendirmek ve kelimeler arasındaki ilişkileri göstermek için kullanılır.

Anlamsal Analiz: Anlamanın tanımlanmasında rol oynar. Vurgu, ifadelerin ve kelimelerin tam anlamını etkiler.

Konuşmanın Birleştirilmesi: Metnin bütünsel anlamı alt cümlelerden etkilenmektedir. Konuşma düzeni, işlevin çalıştırdığı cümlelere bağlıdır ve baktığınız cümlelerin açıklamasını gerektirir.

Pragmatik (Edimsel) Analiz: Doğal dil işlemenin son aşamasıdır. Cümlelerin teker teker anlamlarına bakmak yerine tüm metni yorumlar. Bu, işimize yarayan çözümlemeleri yansıtan bir dizi yönergeyi kullanarak istenen etkiyi elde etmemizi sağlar.

3.4. Genel Uygulamalar

3.4.1. Otomatik Aramada Düzeltme

Google’da her arama yaptığımızda 3-4 karakter yazdıktan sonra bize mümkün aramaları göstermektedir. Ayrıca, bir şeyi yazım hatalarını kullanarak ararsak, bunları düzeltmekte ve yine de ilgili sonuçları almaktayız. Bu süreç herkesin her gün kullandığı ama pek dikkat etmediği bir şeydir. Mükemmel bir doğal dil analizi uygulaması ve bunun dünya çapında birçok insanı nasıl etkilediğine dair son derece güzel bir örnektir. Otomatik aramada düzeltme ve bireysel düzeltmeler, en doğru sonuçları almamıza yardımcı olmaktadır (Perdices vd., 2021).

3.4.2. Dil Çevirici

Otomatik çeviri, metnin bir dilden diğerine mekanik olarak çevrilmesi ve aynı zamanda tanımın korunması sistemidir. Başlangıçta mekanik çeviri programları sözlük ve belirli ilkelere dayanıyordu ve çok az başarı elde ettiler. Ancak duyuşal ağlar alanındaki değışiklikler, sosyal medya gibi alanlardan mizahi bilgilerin sağlanması ve bu konuda kullanılan modeller ve algoritmalarla makine çevirisi, içeriğın farklı bir dile çevrilmesi konusunda hassas hale gelmiştir (K. Jiang & Lu, 2020).

3.4.3. Dilbilgisi Kontrol Araçları

Dilbilgisi Kontrol Araçları, en popüler kullanılan doğal dil işleme sistemleri arasındadır. Örneğın Grammarly, bir kişinin hatasız materyal yazmasını sağlayan birçok özellik sağlar. Herhangi bir normal metni son derece parlak içeriğē dönüştürebilirler. Profesyonel ortamda inanılmaz derecede faydalıdır. Bu araçlar dil bilgisini ve yazımı geliştirebilir, daha iyi eşanlımlılar önerebilir ve daha net ve tutarlı içerik sunulmasına yardımcı olabilir. Ayrıca metnin okunabilirliğini artırmaya yardımcı olur ve mesajınızı en uygun şekilde iletmenize olanak tanır (Calma vd., 2022).

3.4.4. Hedeflenmiş Reklamcılık

Hedeflenmiş reklamcılık, reklamların tüketiciye çevrimiçi etkinliğine göre gösterildiğı, internette doğal dil işleme kullanılarak yapılan bir reklam biçimidir. Günümüzde çevrimiçi şirketlerin çoğunluğu bu yöntemi kullanmaktadır. Bunun nedeni ise firmanın reklam maliyetlerini düşürmesi ve potansiyel müşterilere sadece uygun reklamların gösterilmesidir. Hedefli reklamcılık özellikle anahtar kelimeleri eşleştirmede etkilidir. Tüm reklamlara karşılık gelen bir anahtar kelime veya kelime öbeğı atanmış olup bunlar yalnızca belirli anahtar kelimeleri arayan kişilere gösterilir. Anahtar kelimelerin yanı sıra web arama geçmişı, önemli miktarda zaman harcadıkları web sayfaları gibi diğer veriler de dikkate alınarak kullanıcılara ilgilenebilecekleri alakalı ürün için reklamlar sunulmaktadır (L. Zhang vd., 2022).

3.4.5. ChatBot (Sohbet Robotları)

Müşteri ilişkileri bir organizasyondaki en önemli departmanlardan biridir. Müşterilerin sorunlarını çözmek ve müşteri geri dönüşlerinden elde edilen bilgiler ile hizmetlerin ve ürünlerin iyileştirilmesini sağlamak müşteri ilişkileri departmanlarının en önemli görevlerindendir. Ancak tüm müşterilerle ilgilenmek ve sorunlarını çözmek zaman alıcı, maliyetli ve sıkıcı olabilir. Müşterilerin sıkça sordukları sorulara hızlı yanıt üreten ve sorunlarını çözmede yardımcı olan Chatbot'lar şirketlerin çağrı merkezlerinin yükünü azaltmaktadır. Son dönemde birçok kuruluş, uygulamalarında ve web sitelerinde temel müşteri sorularını çözen chatbot'u kullanmaya başlamıştır. (Kasthuri & Balaji, 2023).

3.4.6. Sesli Asistan

Sesli asistan, kullanıcının sesli komutlarını anlamak ve eylemi uygun şekilde gerçekleştirmek için konuşma tanıma, anlama ve doğal dil işlemeyi kullanan bir yazılım türüdür. Google Asistan, Alexa ve Siri günlük hayatımızda kullandığımız sesli asistanlardan bazılarıdır. Bir Chatbot'tan çok daha iyidirler ve daha verimli kullanılabilirler. Bugün hayatımızı sesli yardımcılar olmadan hayal edemiyoruz. Son yıllarda yakın arkadaş ve güvenilir arkadaş olmuşlardır. Sesli asistan bizi uyandırmaktan gece alarmı kurmamıza yardım etmeye kadar her şeyi yapabilir. Sesli yardımcılar hem şirketler hem de kullanıcılar için yeni fırsatlar kapısının kilidini açmıştır (Johri vd., 2021).

BÖLÜM 4

METİN OLUŞTURMA

Metin, çeşitli amaçlarla dilin belli kurallarına göre bir araya getirilen cümleler topluluğu, dil bütüncesidir. Metni özerk bir şekilde oluşturmak zor olsa da günümüzün yapay zeka araçları insanın ürettiğine benzer seviyede metinler üretebilmektedirler. Bu zorluğu şöyle değerlendirebiliriz; dilin karmaşıklığı çok yönlü ele alınabilir. Çünkü sözdiziminden anlambilime, bir cümle içinde tutarlı bir anlamın nasıl ifade edileceğine, çeşitliliğe ve doğaçlamaya kadar dikkate alınması gereken çeşitli yönler vardır. Sözdizimi yapıları istatistiksel modellerle öğrenilebilir; örneğin bir cümlenin özne, yüklem ve nesneyle nasıl yapılandırılacağı gibi.

Çok sayıda yaklaşım, kendisinden önce gelenler göz önüne alındığında bir sonraki kelimeyi tahmin etmek için referans olarak kullanılan bir metindeki kelimelerin olasılık dağılımını temsil etmeyi amaçlayan dil modelleri oluşturarak bu görevi ele alır ve böylece metin oluşturma görevini bir dizi tahmin problemi olarak şekillendirir. Dil modellerinin mevcut durumu, sinir ağları kullanılarak geliştirilmektedir (Rizzo & Van, 2020). Yapay zeka ve özellikle derin öğrenmenin varlığı son birkaç yılda daha belirgin hale geldikçe ve alanlar önemli ölçüde popülerlik kazandıkça, doğal dil işleme alanında giderek daha fazla görev ortaya çıkmıştır. Bunlardan biri, derin sinir ağlarının yardımıyla tasarlanabilen otomatik metin oluşturmadır (Sutskever vd., 2011).

Metin oluşturma, kelime tabanlı bir dil modeli geliştirmek için metin hazırlama ve kaynakta kullanılan kelimelerin önceki sırasına dayalı olarak bir kelimenin ortaya çıkma olasılığını tahmin edebilecek şekilde bir sinirsel dil modeli tasarlama ve

yerleştirme sürecidir. Daha sonra öğrenilen dil modeli, kaynak metinle benzer istatistiksel özelliklere sahip yeni metin oluşturmak için kullanılır (Milanova vd., 2019).

İnsanların fikirlerini yazı veya konuşma yoluyla iletmeleri gibi, metin üretme işlevini kullanan sistemler de fikirleri okuyucularına açık ve kullanışlı bir şekilde aktaran doğal dildeki metin veya konuşmayı üretmek üzere tasarlanmıştır. Metin oluşturma sistemleri, sohbet robotları aracılığı ile insanlarla etkileşimli konuşmaları oluşturmak, görüntülere veya görsel sahnelerle altyazı eklemek, metni bir dilden diğerine çevirmek, hikayeler ve haber makaleleri üretmek gibi birçok gerçek dünya uygulaması için metin oluşturmak için kullanılmıştır (Celikyilmaz vd., 2020).

Metin oluşturma teknikleri, kuralları kullanarak doğal dil metni üreten basit şablon tabanlı sistemlerden, insan dilbilgisini karmaşık bir şekilde anlayan makine öğrenimli sistemlere kadar uzanır. İlk nesil otomatik metin oluşturma sistemleri, kural tabanlı veya veri odaklı ardışık düzen yöntemlerini kullanır. Daha önceki metin üretme yaklaşımları ve bunların uzantıları, metin üretme araştırmalarının evriminde önemli bir rol oynamaktadır (Reiter & Dale, 1995).

Geçmişte yapılan çalışmaların ardından, birçok metin oluşturma çalışmasını yürüten araştırmacının odaklandığı önemli bir yön, daha fazla veri kaynağının bulunmasıyla dikkat çeken veriye dayalı temsil öğrenimidir. Büyük veri kümelerinin, ağaç bankalarının, atıfta bulunan ifadelerin bütünü ve paylaşılan görevlerin mevcudiyeti, günümüzde birçok metin üretme görevinin ilerlemesinde faydalı olmuştur (Gkatzia vd., 2015).

Son on yılda, derin sinir ağı modellerini kullanarak denetimsiz bir şekilde büyük metinsel gövdelerden temsillerin öğrenilmesine yönelik bir yaklaşım değişimine tanık olundu. Son zamanlardaki metin oluşturma modelleri, derin sinir ağı modellerinin, genellikle insan tarafından yazılmış çok büyük metinler üzerinde eğitilmesiyle oluşturulmuştur. Yaklaşım değişimi, dil temsillerini öğrenmek için yinelemeli sinir ağları (RNN)(Graves, 2013a), uzun kısa süreli bellek ağları (LSTM) (Hochreiter & Schmidhuber, 1997a), geçitli yinelemeli birimler (GRU) (Cho, Van Merriënboer, vd., 2014) ve daha sonra kodlayıcı-kod çözücü (Transformers) (T. Lin vd., 2022) mimarisinin geniş çapta uygulamasıyla karakterize edilen yeni bir yöntem olan sıralı diziden diziye öğrenme ile başlar.

Geleneksel yaklaşımları kullanarak dil modelleri oluşturmak, dil yapısı, dil bilgisi, kelime kullanımı ve akıl yürütme dahil olmak üzere dilin birçok yönünü hesaba katması gereken ve dolayısıyla önemsiz olmayan veri etiketleme çabalarını gerektiren karmaşık bir iştir. Son zamanlarda Wikipedia, arama sonuçları, bloglar, Reddit gönderileri gibi çevrimiçi kaynaklardan elde edilen büyük miktarda ham metinden yararlanılarak oluşturulan Transformer tabanlı doğal dil modellerinin çok etkili olduğu gösterilmiştir. Örneğin, en gelişmiş sinir dili modellerinden biri olan GPT-3 (T. B. Brown vd., 2020), insan tarafından oluşturulan metinlerden neredeyse ayırt edilemeyecek kadar uzun metinler üretebilmektedir. XiaoIce (Zhou & Lampouras, 2020) gibi duygusal etkileşimleri anlayabilen sosyal sohbet robotları, insan diyalogunu iyi anlıyor gibi görünmekte ve kullanıcılarla uzun vadeli duygusal bağlantılar kurmak için kişilerarası yanıtlar üretebilmektedir.

4.1. Metin Oluşturmada Kullanılan Mimariler

4.1.1. RNN (Yinelemeli Sinir Ağları-Recurrent Neural Network)

Doğal dil işlemedeki birçok görevde tahmin, önceki birkaç örneğe bağlıdır; öyle ki, bireysel örnekleri sınıflandırmanın yanı sıra, girdi (kelime) dizilerini de analiz etmemiz gerekir. Bu tür uygulamalarda ileri beslemeli sinir ağı, giriş ve çıkış katmanları arasında herhangi bir bağımlılık varsayılmadığından uygulanamaz. RNN'ler, bu konuyu çeşitli uzunluklarda sıralı (örneğin, konuşma veya metin) veya zaman serisi problemlerinde ele almak için geliştirilmiştir. Bir RNN'in girişi hem mevcut örnekten hem de önceki gözlemlenen örnekten oluşur. Başka bir deyişle, bir RNN'in $t-1$ zaman adımındaki çıkışı, t zaman adımındaki çıktıyı etkiler. Her nöron, mevcut çıktıyı bir sonraki adım için girdi olarak döndüren bir geri bildirim döngüsüyle donatılmıştır. Bu yapı, bir RNN'deki her bir nöronun, önceki girişten gelen hesaplamaların bilgilerini saklayan bir dahili belleğe sahip olacağı şekilde ifade edilebilir. Yinelemeli ağı eğitmek için, Zaman İçinde Geri Yayılım adı verilen geri yayılım algoritmasının bir uzantısı kullanılır. Nöronlar üzerinde döngülerin varlığı nedeniyle, RNN'lerde yığılmış katman modelimiz olmadığı halde, üst katmanlarındaki ağırlığa göre hata türetmeye dayalı çalıştığı için orijinal geri yayılımı burada kullanamayız. Geleneksel RNN'ler, zaman içerisinde açıldıklarında girdi katmanı ile çıktı katmanı arasında birkaç doğrusal olmayan nöron katmanı olarak görülebildikleri için derin modeller olarak kabul edilebilirler.

Bununla birlikte, RNN’lerin mimarisi ve işlevselliği göz önüne alındığında, RNN’lerdeki gizli katmanların özelliklerin hiyerarşik temsili yerine bir bellek sağlaması beklenir (Mohammadi vd., 2018).

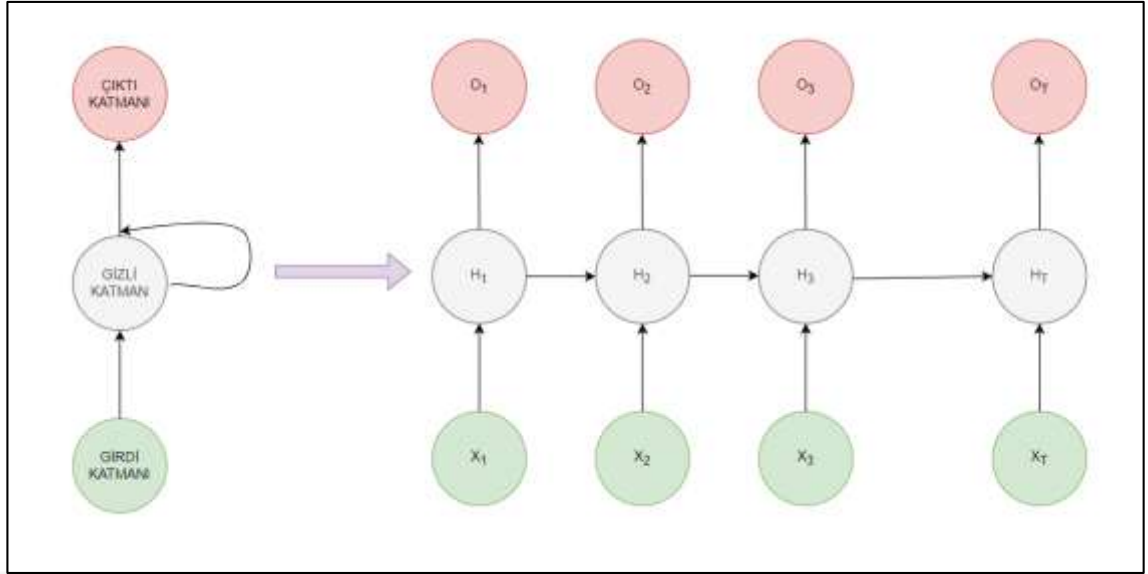
RNN’ler, düzgün bir şekilde eğitilmesi zor olan güçlü dizi modelleridir. RNN’ler, sıralı görevler için anlamlı bir model ailesi içinde yer alırlar. Geçmiş bilgileri hatırlamalarını ve işlemlerini sağlayan doğrusal olmayan dinamiklere sahip yüksek boyutlu gizli bir katmana sahip oldukları için güçlüdürler. Ayrıca, RNN’in gradyanlarının zaman içinde geriye yayılımla hesaplanması düşük maliyetlidir. Çekici niteliklerine rağmen RNN’ler, etkili bir şekilde eğitilmelerinin zorluğu nedeniyle makine öğreniminde ana araç haline gelememiştir. Bu zorluğun nedeni, “kaybolan gradyanlar probleminde” kendini gösteren, gizli katmanların dinamikleri ile parametreler arasındaki son derece istikrarsız ilişkidir.

Yinelemeli Sinir Ağı, standart ileri beslemeli sinir ağının sıralı verileri modellemesine olanak tanıyan basit bir uyarlamasıdır. Her zaman adımında RNN bir girdi alır, gizli katmandaki durumunu günceller ve bir tahminde bulunur. RNN’in yüksek boyutlu gizli katmanı ve doğrusal olmayan evrimi, ona büyük bir ifade gücü kazandırır ve RNN’in gizli katmanının birçok zaman adımında bilgiyi entegre etmesine ve bunu doğru tahminler yapmak için kullanmasına olanak tanır. Her birimin kullandığı doğrusal olmama durumu oldukça basit olsa da zaman içinde yinelenmesi çok zengin dinamiklere yol açmaktadır. Standart RNN şu şekilde formüle edilir: Bir dizi giriş vektörü ($x_1, \dots, x_T$) verilir, RNN bir dizi gizli katman ile ($h_1, \dots, h_T$) ve aşağıdaki denklemleri yineleyerek bir dizi çıktı ($o_1, \dots, o_T$) hesaplar (Sutskever vd., 2011).

$$h_t = \tanh(W_{hx}x_t + W_{hh}h_{t-1} + b_h) \quad (4.1)$$

$$o_t = W_{oh}h_t + b_o \quad (4.2)$$

Bu denklemlerde, W_{hx} girdi katmanı ile gizli katman arasındaki ağırlık matrisidir, W_{hh} gizli katman ile gizli katman (veya yinelenen) arasındaki ağırlık matrisidir, W_{oh} gizli katman ile çıktı katmanı arasındaki ağırlık matrisidir ve b_h ve b_o vektörleri (biases) önyargılardır. Şekil 4.1’de RNN mimarisinin temel işleyiş şeması gösterilmektedir.



Şekil 4.1. Bir zaman adımında metin işlemeye yönelik RNN mimarisi

RNN'nin gradyanlarının zaman içinde geri yayılma yoluyla hesaplanması kolaydır, dolayısıyla RNN'lerin gradyan inişiyile eğitilmesi kolay gibi görünebilir. Gerçekte, RNN'nin parametreleri ve dinamikleri arasındaki ilişki oldukça dengesizdir ve bu da gradyan inişini etkisiz hale getirir (Bengio vd., 1994).

Standart bir RNN'de uzun etki alanı ile ilgili zamansal yapıyı öğrenmek için gradyan inişinin yetersizliğiyle başa çıkmanın bir yolu, bilgiyi uzun zaman dilimleri boyunca depolamak üzere özel olarak tasarlanmış "bellek" birimlerini içerecek şekilde modeli değiştirmektir (Graves & Schmidhuber, 2008).

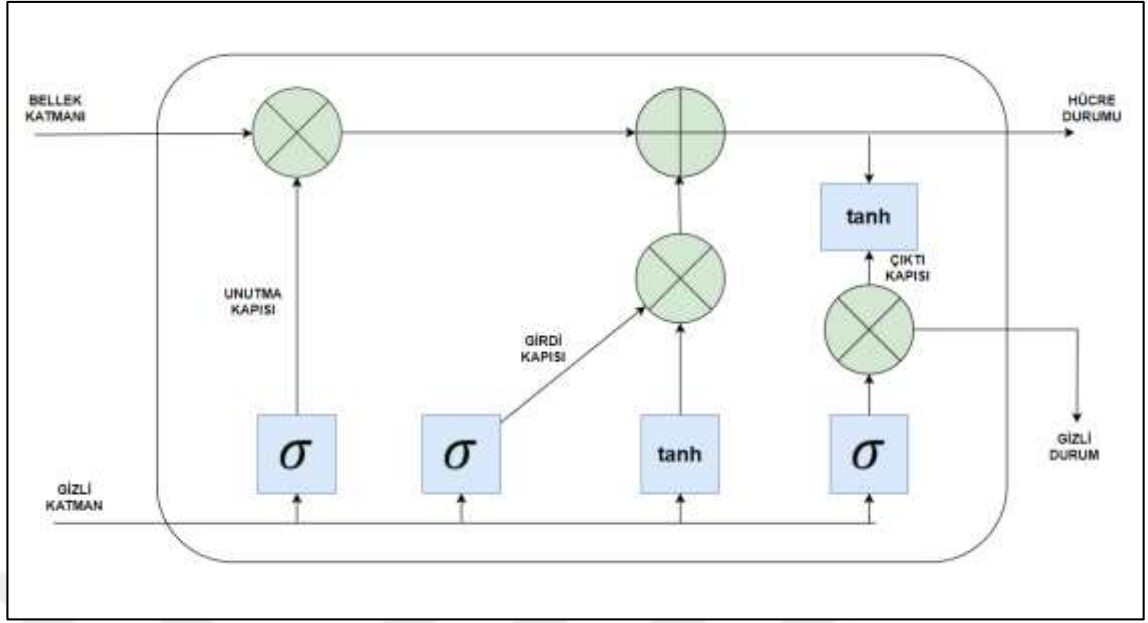
RNN'ler, dizi modelleme yeteneğinden dolayı metin üretimi için kullanılabilir. RNN'lerin sıralı görevleri modelleme yeteneği, yüksek boyutlu gizli durumunda ve doğrusal olmayan dinamiklerinde yatmaktadır. RNN'lerle ilgili eğitim sorunları uzun vadeli bağımlılıkların modellenmesidir: patlayan ya da yok olan gradyan; algoritma ağırlıklara yüksek sonuçlar atadığında, modeldeki gradyanda benzer şekilde gradyan değerlerinin çok küçük olduğu durumlarda hiçbir şey öğrenmemesine neden olur; ve model öğrenmeyi durdurur. Ancak RNN'lerin eğitiminin çok zor olduğu görülmüş ve bu da birçok doğal dil işleme görevinde kullanımını engellemektedir (Iqbal & Qureshi, 2022).

Sıralı diziden diziye modelleri kullanarak metin oluşturmak, video gibi sürekli verilerin ve metin gibi ayrı bilgilerin işlemesi gerektiğinde basit bir iş değildir. RNN

modelleri eğitim aşaması için zor derin öğrenme mimarileridir ancak RNN'in sıralı yeteneği dil üretme ve diğer birçok NLP görevinde daha iyi sonuçlar sağlar. RNN, sıralı modelleme yeteneğinden dolayı hikâye vb. oluşturmak için kullanılabilir. Kısa öykü veya paragraf üretimine uygundur. RNN ile ilgili eğitim sorunları, uzun vadeli bağımlılığın önemli ölçüde daha az olmasından kaynaklanmaktadır. RNN'ler, bir dizideki bir sonraki kelimenin olasılık dağılımını tahmin etmeyi öğrendikleri önemli bir metin külliyatı üzerinde eğitilir. Her yeni kelimeyle RNN'nin dahili durumunu güncellerken dizideki her kelimeyi işler. RNN'nin son durumu kullanılarak, önceki bağlamı takip edebilecek olası kelimeler kümesi üzerinde bir olasılık dağılımı oluşturulur. Bir sonraki kelime modeldeki olasılık dağılımından örneklenir ve istenen metin uzunluğunu oluşturmak için oluşturulan metne eklenir. Ancak RNN'ler bazı sınırlamalardan dolayı problem oluşturma potansiyeline sahip olabilir. Özellikle uzun diziler oluştururken tekrarlayan veya tutarsız metinler üretebilirler (Pandey & Roy, 2023).

4.1.2. LSTM (Uzun Kısa Vadeli Hafıza-Long Short Term Memory)

Metinsel verilerden içgörü geliştirmek için RNN iyi bir seçimdir. Ancak güncel yani metnin uzunluğuna bağlı olarak sondaki bilgileri verimli bir şekilde öğrenemez; bu nedenle, bu sorunun üstesinden gelmek için mevcut bağımlılığın yanı sıra bağlamı da algılayabildiğinden LSTM tercih edilir. LSTM, RNN'in bir çeşididir. LSTM, 1997 yılında Schmidhuber ve Hochreiter tarafından keşfedildi. LSTM'nin, mevcut girişi ve önceki çıkışı saklayan hücreler olarak bilinen kendi bellek ünitesi vardır. Hücreler hangi bilginin yararlı olup olmayacağına karar vermeye yardımcı olur ve daha sonra yararlı olmayan bilgileri sonraki katmanlarda atar. Bir LSTM hücresinin çalışma biçimi Şekil 4.2'de gösterilmiştir.



Şekil 4.2. LSTM hücresinin çalışma biçimi

LSTM'nin her hücresi, yinelemeli bir sinir ağıyla aynı giriş ve çıkışa sahiptir, ancak bilgi akışını kontrol etmeye yardımcı olan fazladan bazı parametreler vardır. Bunlardan, durum birimi $S_i^{(t)}$, doğrusal kendi kendine döngüye (self-loop) ve sızdıran birime (Leaky unit) sahip en önemlisidir. Doğrusal kendi kendine döngü, unutma kapısı (forget gate) $f_i^{(t)}$ (hücre için i ve zaman damgası için t gösterimi kullanılmıştır) tarafından kontrol edilir. Bu birimin ağırlığı sigmoid birimi aracılığıyla 0 ile 1 arasındadır. Unutma kapısının çalışması esnasında gerçekleşen işlem şöyledir:

$$F_i^{(t)} = \sigma (b_i^f + \sum_j U_{i,j}^f x_j^{(t)} + \sum_j W_{i,j}^f h_j^{(t-1)}) \quad (4.3)$$

Denklemde $x^{(t)}$ mevcut giriş vektörünü temsil eder ve $h^{(t)}$, LSTM hücresinin çıkışını tutan geçerli gizli katmanı temsil etmektedir. b^f , U^f , W^f , unutma kapıları için sırasıyla önyargılar, giriş ağırlığı ve tekrarlanan ağırlık ile temsil edilmektedir.

LSTM modelinin süreci şu şekilde işlemektedir. Unutma kapısında, bilginin hafıza hücresinden aktarıldığı ilk aşamada süreç başlar. Sigmoid fonksiyonu, mevcut giriş durumu $x^{(t)}$ ile önceki zaman adımı $h^{(t-1)}$ 'i bu katmanda birleştirir. Bu işlem 0 veya 1 sonucunu verir, burada tanımlanan 1 veri tutulacak ve 0 atılacaktır. Unutma kapısı için kullanılan denklem kısaca şöyle gösterilir:

$$f_t = \sigma(W_f[h_{t-1}, x_t] + b_f) \quad (4.4)$$

İkinci aşamada giriş kapısı, bellek hücresinin sigmoid katman ve $\tanh$ katmanının çıkışını güncellediğine karar verir. Sigmoid katmanı, $x_{(t)}$ ve $h_{(t-1)}$ 'e uygulanarak 0 veya 1 çıktısını üretir; burada 0, bilginin önemsiz olduğunu ve 1, bilginin önemli olduğunu gösterir. Bellek hücresi durumuna bilgi eklenmelidir. $\tanh$ katmanının sonucu -1 ve 1 olacaktır, bu da yeni bir vektör adayı olan $c_{(t)}$ 'yi oluşturur ve ağırlık düzenlenmesine yardımcı olur. Bu aşamada şemaya göre oluşan denklemler şöyledir:

$$i_t = \sigma (W_i[h_{t-1}, x_t] + b_i) \quad (4.5)$$

$$c_t = \tanh(W_c[h_{t-1}, x_t] + b_c) \quad (4.6)$$

Üçüncü aşamada bir önceki adımdaki hesaplamalara göre yeni bir hücre durumu oluşturulacaktır:

$$c_t = f_t * c_{t-1} + i_t * c_t \quad (4.7)$$

Son aşamaya doğru çıkış kapısı bir sonraki gizli katmana karar verir. Bu adımda, mevcut giriş ve önceden gizli olan durumlar bir sigmoid fonksiyonundan geçerek o_t 'yi üretir. Ayrıca o_t , $\tanh$ katmanı çıktısıyla çarpılır ve yeni bir hücre durumu sağlar. Bu aşamada şu iki denklem işleme konur:

$$o_t = \sigma (W_o[h_{t-1}, x_t] + b_o) \quad (4.8)$$

$$h_t = o_t * \tanh(c_t) \quad (4.9)$$

Yukarıdaki denklemlerde sırasıyla x_t giriş, f_t unutma kapısı, i_t giriş kapısı, c_t hücre durumu, o_t çıkış kapısı ve h_t gizli durumdur. W_i , W_c ve W_o ağırlık matrisleridir ve b_f , b_i , b_c ve b_o önyargılardır.

Bu denklemler yalnızca tek seferlik bir adımdır. Bu, bu denklemlerin her zaman adımı için yeniden hesaplandığı anlamına gelir. Ağırlık matrisleri ve sapmalar zamana bağlı değildir; bu da ağırlıkların her zaman adımı için aynı olduğunu gösterir. LSTM önceki çıktıyı kolaylıkla hatırlayarak bir sonraki kelimenin tahmin edilmesine yardımcı olur. Metin oluşturma modelinde ilk katman yerleştirme katmanı, sonraki katman ise LSTM katmanıdır. Aşırı uyumun üstesinden gelmek için bırakma katmanı da göz önünde bulundurulur (Pandey & Roy, 2023).

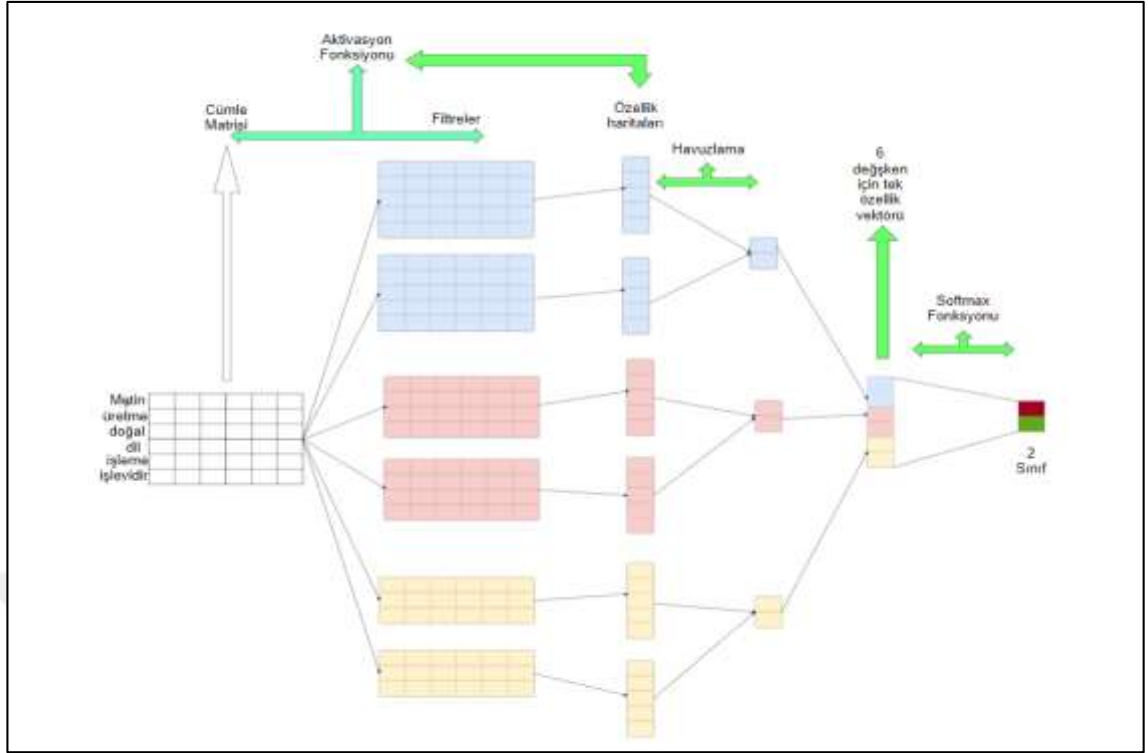
İlgili çalışmada (R. Evans & Grefenstette, 2018), otomatik metin oluşturmada doğal dil işlemenin son eğilimleri ve ilerlemeleri tanımlanmakta, otomatik metin

oluşturma mimarisi ve onun dezavantajlarını tartışılmaktadırlar. Dethlefs ve ark. sıralı ve sıralı hafıza metni üreten LSTM tabanlı bir yöntem önermişlerdir ve giriş veri kümesine dayalı olarak metin oluşturmak için böl ve yönet yaklaşımını kullanmıştır (Dethlefs vd., 2021). Cao, WEATHERGOV, WIKIBIO, WIKITABLE ve WIKIBIOCN veri kümelerini kullanarak RNN ve LSTM tabanlı metin oluşturucular geliştirmiştir (Cao, 2020). Literatürde Zhang ve Lapata'nın önerdiği Çin şiiri oluşturmaya yönelik bir model bile görebiliriz. Modelleri RNN tabanlı bir modeldir ve şiir üretmek için kullanılmıştır. Bu sebeple şiirlerin uzunluğu kısa ve akıcıdır (X. Zhang & Lapata, 2014). Ren ve ark. bir haber raporu veri kümesini kullanarak mikro-fınansal raporlar üreten çift yönlü bir LSTM tabanlı doğal dil üretme modeli önermişlerdir (Ren vd., 2021). Metin üretme modelleri aynı zamanda bazı hikayelerin oluşturulmasında da faydalı olabilir. Pawade ve arkadaşları, hikaye üretimi için mevcut hikaye veri kümelerini kullanarak otomatik bir hikaye üreten RNN ve LSTM tabanlı bir model önermiştir (Pawade vd., 2018). İlgili çalışmada bağlama dayalı hikayeler oluşturmak için LSTM tabanlı bir model tarafından önerilmiştir (Santhanam, 2020). Xiang ve arkadaşları gizli bilgiler için karakter düzeyinde steganografik metinlerle ilgilenen çekici bir metin üretme modeli önermişlerdir. Bu model aynı zamanda LSTM tekniğini de benimsemiştir (Xiang vd., 2020). Ayrıca Chakraborty ve arkadaşları LSTM modelini geliştirmiş ve karakter tabanlı bir metin oluşturmuşlardır (Chakraborty vd., 2020). İslam ve arkadaşları gazete bütüncesine dayalı Bengalce metin oluşturmak için RNN ve LSTM'nin birleştirilmiş yöntemini önermişlerdir (Md. Sanzidul vd., 2019).

4.1.3. CNN (Evrişimsel Sinir Ağları-Convolutional Neural Networks)

Evrişimsel Sinir Ağında (CNN), ızgara benzeri topoloji kullanılır. Örneğin, zaman serisi verileri her zaman aralığında 1 boyutlu bir ızgara alır ve görüntü verileri 2 boyutlu ızgara piksellerini alır. CNN, Facebook'ta otomatik fotoğraf etiketleme gibi pratik endüstriyel uygulamalarda iyi sonuçlar elde etmiştir (Bourane vd., 2015). Biyomedikal görüntülerdeki bileşik figürlerin CNN bazlı ayrımı tipik olarak birden fazla alt figür içerir. Bu yöntem, halka açık bir biyomedikal görüntü veri seti üzerinde doğruluk ve verimlilik açısından umut verici sonuçlar elde etmiştir (Zheng vd., 2018). CNN, en azından katmanlarından birinde kullanılan genel matris çarpımı yerine evrişimi kullanır. Evrişimsel sinir ağları doğal dil işlemede metin sınıflandırma gibi birçok alanda son

derece başarılı olmuştur. Görüntü işlemede CNN modeli görüntü piksellerini alırken, doğal dil işlemede kelimeleri, cümleleri veya bazen karakter karakter alır (Zhu vd., 2019). Genel olarak, derlem kelime dizinini temsil etmek için kelime yerleştirme veya one-hot encoding kullanır. CNN’de beş cümlede 50 boyut kullanılıyorsa giriş matrisi 5×50 olur ve giriş matrisinin genişliği filtre genişliğine eşit olur. CNN, metin sınıflandırma ve metinden özellik çıkarma gibi doğal dil işleme görevlerinde yaygın olarak kullanılmaktadır. Metin üretimi için CNN’lerin temel fikri, giriş metnini 1 boyutlu bir kelime yerleştirme dizisi olarak ele almak ve dizi üzerinde bir dizi evrişimli filtre uygulamaktır. Her filtre, giriş sırasındaki küçük bir kelime penceresini tarar ve yakındaki kelimeler arasındaki ilişkileri yakalayan yerel özellikleri çıkarır. Çoklu ölçeklerde farklı özellikleri çıkarmak için genellikle giriş dizisine her biri farklı bir pencere boyutuna ve birkaç çıkış kanalına sahip birden fazla filtre uygulanır. Evrişim filtreleri uygulandıktan sonra, elde edilen özellik haritaları, çıktı metnini üreten tamamen bağlı bir sinir ağına beslenir. Bu ağ, oluşturulan metin ile istenen hedef metin arasındaki farkı ölçen bir kayıp fonksiyonunu en aza indirmek için geri yayılım kullanılarak eğitilebilir. Metin oluşturmak için CNN’leri kullanmanın bir avantajı, giriş sırasındaki yakındaki kelimeler arasındaki yerel kalıpları ve ilişkileri yakalama yetenekleridir. CNN, yakındaki kelimeler arasındaki ince bağımlılıkları yakalamayı öğrenebilir. Ek olarak, CNN’ler hesaplama açısından verimlidir ve paralel işleme kullanılarak büyük veri kümeleri üzerinde eğitilebilirler. Bununla birlikte, CNN’leri metin üretimi için kullanmanın bir sınırlaması, kelimeler arasındaki uzun vadeli bağımlılıkları RNN’ler gibi diğer sinir ağı mimarileri kadar etkili bir şekilde yakalayamamalarıdır. Ek olarak, CNN tabanlı bir metin oluşturma sistemi için en uygun mimariyi tasarlamak zorlu olabilir ve hiper parametrelerin dikkatli bir şekilde ayarlanmasını gerektirir (Pandey & Roy, 2023). CNN mimarisinin doğal dil işlemede kullanılırken oluşturduğu mimari Şekil 4.3’teki gibidir.



Şekil 4.3. CNN mimarisinin metin sınıflandırması ve üretimine katkı sunarken kullandığı aşamaların gösterimi

4.1.4. GRU (Geçitli Yinelemeli Birim- Gated Recurrent Unit)

Geçitli yinelemeli birim (Gated Recurrent Unit-GRU), klasik yinelemeli ağlardan farklı olarak her yinelenen birimin farklı zaman ölçeklerindeki bağımlılıkları uyarlanabilir bir şekilde yakalamasını sağlamak için Cho ve arkadaşları tarafından önerilmiştir (Cho, van Merriënboer, vd., 2014). LSTM'in yapısına benzer şekilde GRU, ünite içindeki bilgi akışını modüle eden, ancak ayrı bir hafıza hücresine sahip olmayan geçit ünitelerine sahiptir.

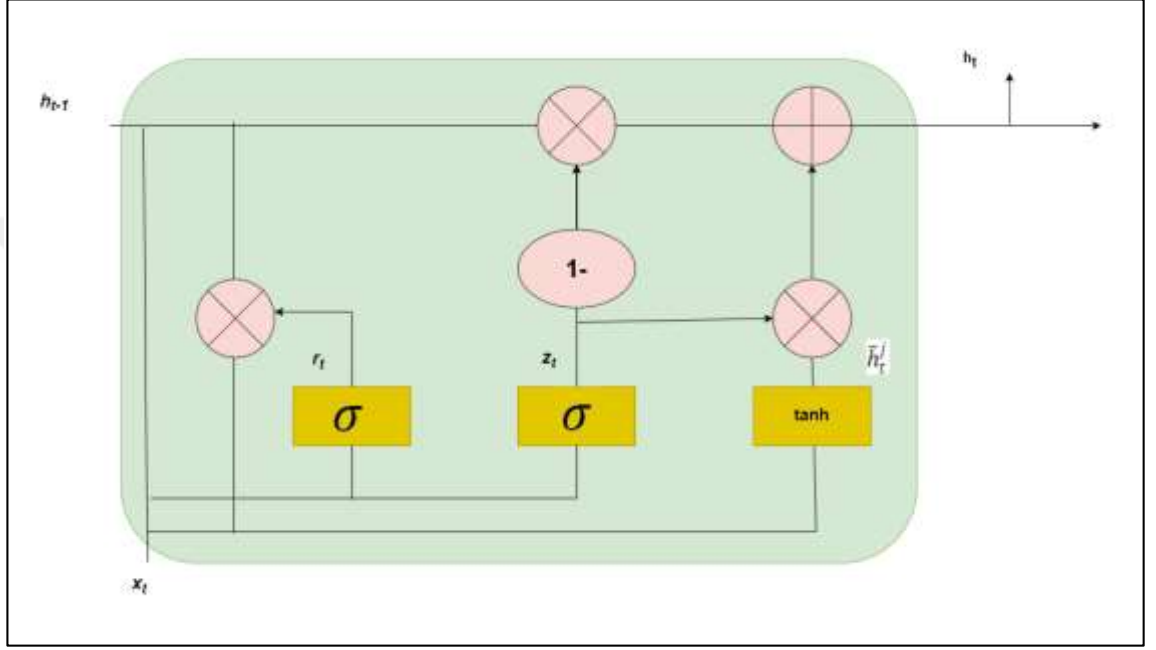
GRU'nun t zamanındaki aktivasyonu h_t^j , önceki aktivasyon h_{t-1}^j ile aday aktivasyonu $\tilde{h}_t^j$ arasında doğrusal bir enterpolasyondur:

$$h_t^j = (1 - z_t^j)h_{t-1}^j + z_t^j\tilde{h}_t^j \quad (4.10)$$

Burada denklemde güncelleme kapısı z_t^j , birimin aktivasyonunu veya içeriğini ne kadar güncelleyeceğine karar verir. Güncelleme kapısı şu şekilde hesaplanır:

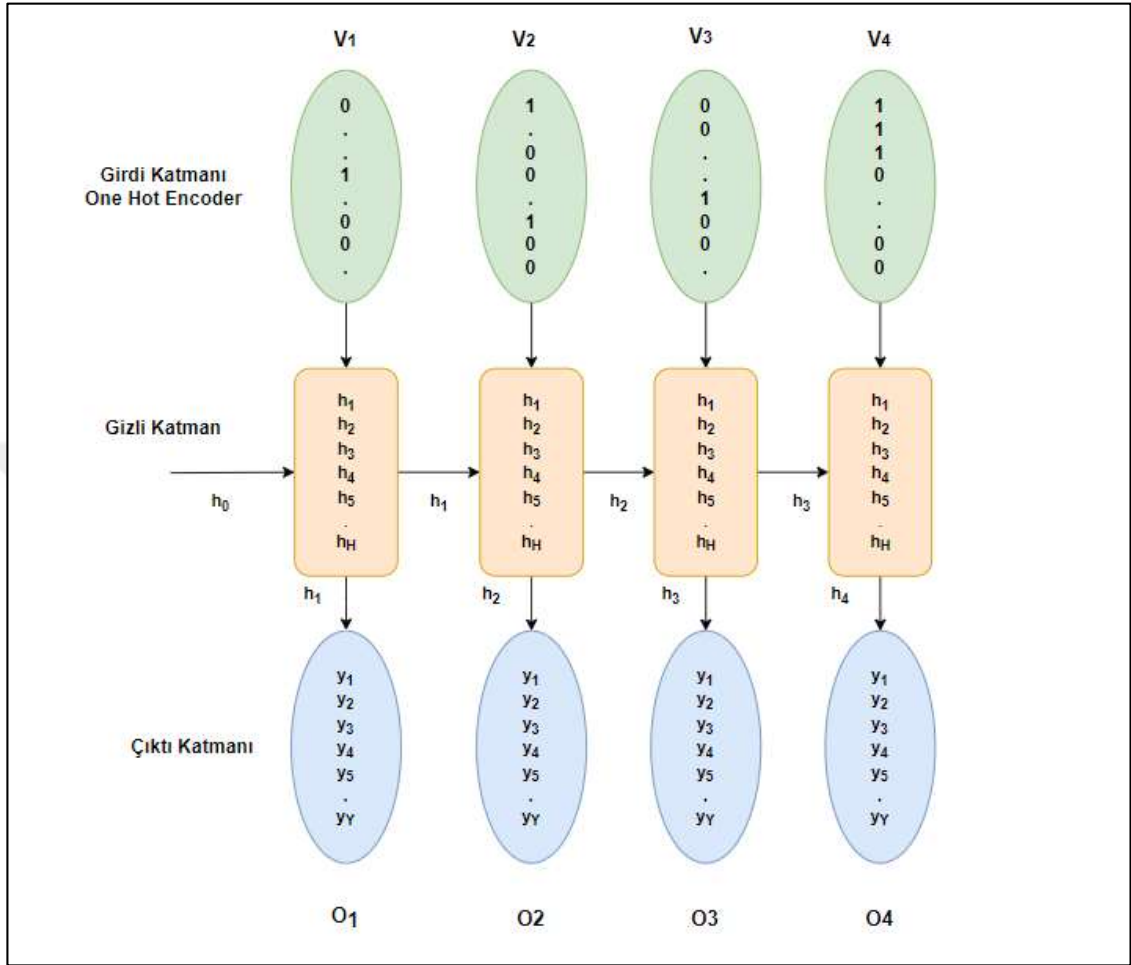
$$z_t^j = \sigma(W_z x_t + U_z h_{t-1}^j) \quad (4.11)$$

GRU’da mevcut durum ile yeni hesaplanan durum arasında doğrusal bir toplam alma süreci, LSTM ünitesine benzer. Ancak GRU’nun, durumunun ne ölçüde açığa çıkacağını kontrol edecek herhangi bir mekanizması yoktur, ancak her seferinde tüm durumu açığa çıkarır. Aday aktivasyonu, geleneksel yinelenen biriminkine benzer şekilde hesaplanır (Chung vd., 2014). GRU mimarisinde bir hücrenin gösterimi Şekil 4.4’teki gibidir.



Şekil 4.4. GRU mimarisindeki bir hücrenin gösterimi

RNN, LSTM, GRU mimarilerindeki metinleri vektörlere dönüştürüp kullanırken oluşan Şekil 4.5'teki gibidir.



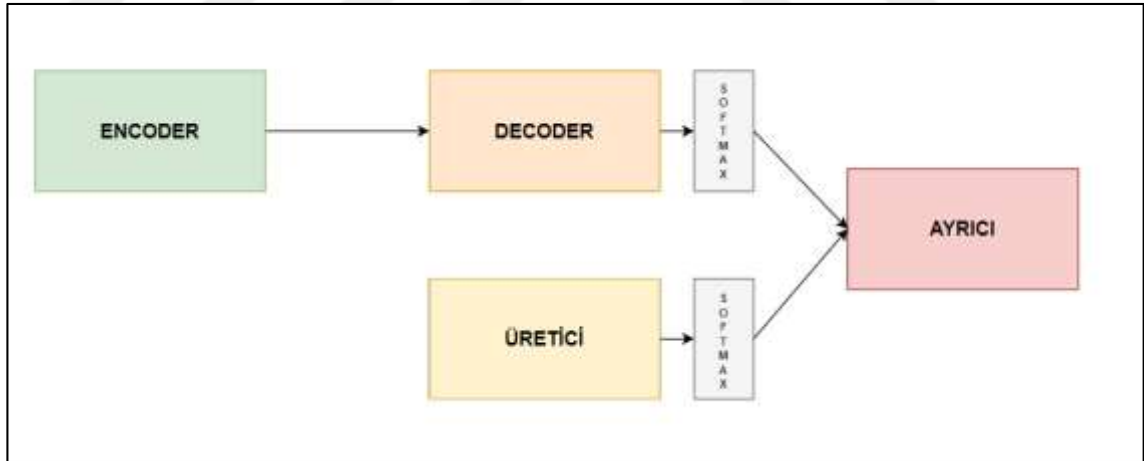
Şekil 4.5. RNN, LSTM, GRU mimarilerindeki metinleri vektörlere dönüştürüp kullanırken oluşan şekil

4.1.5. GAN (Çekişmeli Üretici Ağ- Generative Adversarial Network)

Üretken Çekişmeli Ağlar, son zamanlarda büyük ilgi gören yeni bir derin üretken model sınıfıdır. Üretken Çekişmeli Ağlar karmaşık ve yüksek boyutlu dağılımları örtülü olarak görüntüler, ses ve veriler üzerinden öğrenerek çalışır. Bununla birlikte, ağ mimarisinin uygunsuz tasarımı, amaç fonksiyonunun kullanımı ve optimizasyon algoritmasının seçimi nedeniyle Üretken Çekişmeli Ağların eğitiminde girdiden bağımsız olarak çok sınırlı çeşitlilikte örnekler üretmesi, kararlı bir duruma ulaştığı ve ağ parametrelerinin (yani ağırlıklar ve sapmalar), eğitim verileri için doğru tahminler üreten değerlere yerleşememe ve başlangıç koşullarında küçük değişiklikler yapıldığında

genelleme modellerinde büyük farklılıklar üretme gibi büyük zorluklar vardır (Saxena & Cao, 2021).

Çoğu metin oluşturma sistemi, her bir belirtecin one hot vector tarafından ayrı bir birim olarak temsil edildiği sabit bir kelime dağarcığı üzerinden tanımlanır. Metin oluşturma, bir önceki adımın çıktısını adım girdisine besleyerek otoregresif modellemeye uygundur. Bu işlem için doğal bir mimari seçimi, LSTM'ler (Hochreiter & Schmidhuber, 1997b), GRU'lar (Chung vd., 2014) gibi yinelemeli ağlar veya daha yakın zamanda ilişkisel bellek ağlarıdır. Bu modellerde, ilk önce giriş belirtecini bir yerleştirme matrisi kullanarak bir yerleştirmeye kodluyoruz. Gizli bir vektör elde etmek için önceki adımdaki gizli vektörle birlikte kullanılan derin öğrenme ağında bir çıktı elde ediyoruz. Daha sonra çıktıyı, bir ağırlık matrisi (ve isteğe bağlı bir temperature parametresi τ) ve ardından softmax kullanarak doğrusal bir dönüşüm yoluyla sözcük dağarcığı üzerinde bir olasılık dağılımına dönüştürürüz. Bu olasılık dağılımından bir sonraki kelime, τ temperature değeri ile üretilen örneklerin çeşitliliği ve kalitesi arasındaki dengeyi kontrol eden bir örnekleme prosedürü ile elde edilebilir (Chung vd., 2014). GAN mimarisinin genel şematik gösterimi Şekil 4.6'daki gibidir:



Şekil 4.6. GAN mimarisinin genel gösterimi

GAN'lar, belirli bir metin kümesinin dağılımını, bir üretici ve bir ayırıcı arasındaki iki oyunculu bir ilişki aracılığıyla öğrenir (Goodfellow vd., 2014). Ayırıcı, gerçek cümlelerle sahte olanları ayırt edecek şekilde eğitilir. Orijinal formülasyon, üreticinin ayırmacıyı kandırmak için daha iyi örnekler üretmeye çalıştığı bir minimax hedefinin kullanılmasını önerir. GAN'ları eğitmek için kayıp fonksiyonları aktif araştırma

aşamasındayken, çoğu ortamda GAN'ların eğitimi birçok ayarı da beraberinde gerektirir, gradyanların ayırıcıdan üreticiye iletilmesi buna örnek verilebilir. Bu, üreticinin çıktısı, resimlerdeki veya videolardaki pikseller gibi gerçek değere sahip olduğunda işe yarar. Bununla birlikte, yukarıda açıklandığı gibi metin oluşturmada yer alan örnekleme süreci nedeniyle, üreticinin çıktısı ayırıktır ve bu da üreticiyi eğitmeyi önemli kılar (Kumar & Tsvetkov, 2020).

4.1.6. Dönüştürücüler (Transformers)

Doğal dil işlemedeki son gelişmeler hem model mimarisi hem de model ön eğitimindeki ilerlemelerden kaynaklanmaktadır. Dönüştürücü mimarileri, daha yüksek kapasiteli modellerin oluşturulmasını kolaylaştırmıştır ve ön eğitim, bu kapasitenin çok çeşitli görevler için etkin bir şekilde kullanılmasını mümkün kılmıştır. Dönüştürücüler, bu gelişmeleri daha geniş makine öğrenimi topluluğuna açmayı amaçlayan açık kaynaklı bir kütüphane olarak da adlandırılabilir. Kütüphane, birleşik bir API altında dikkatle tasarlanmış son teknoloji ürünü dönüştürücü mimarilerinden oluşmaktadır (Wolf vd., 2019).

Dönüştürücüler hem doğal dil anlama hem de doğal dil oluşturma görevlerinde performans açısından evrişimli ve yinelemeli sinir ağları gibi alternatif sinir modellerini geride bırakarak, doğal dil işlemede hızla baskın mimari haline geldi. Mimari, eğitim verileri ve model boyutuna göre ölçeklenmekte verimli paralel eğitimi kolaylaştırmakta ve uzun menzilli sıralı dizi özelliklerini yakalamaktadır (Vaswani vd., 2017).

Model ön eğitimi, modellerin genel derlemler üzerinde eğitilmesine ve ardından güçlü performansla belirli görevlere kolayca uyarlanmasına olanak tanır. Dönüştürücü mimarisi özellikle büyük metin derlemleri üzerinde ön eğitime yardımcı olur ve diğerlerinin yanı sıra metin sınıflandırma, dil anlama makine çevirisi ve özetleme gibi aşağı yönlü görevlerde doğrulukta büyük kazanımlara yol açar (Devlin vd., 2018).

Bu ilerleme, bu modellerin geniş çapta kullanılabilmesi için çözülmesi gereken çok çeşitli pratik zorluklara yol açmaktadır. Dönüştürücülerin her yerde kullanımı, sistemlerin modeli çeşitli platformlarda eğitmesini, analiz etmesini, ölçeklendirmesini ve genişletmesini gerektirir. Bu mimari, giderek daha karmaşık hale gelen uzantılar ve hassas deneyler tasarlamak için bir yapı taşı olarak kullanılmaktadır. Ön eğitim

yöntemlerinin yaygın bir şekilde benimsenmesi, topluluk tarafından kullanılan temel önceden eğitilmiş modellerin dağıtılması, ince ayar yapılması, dağıtılması ve sıkıştırılması ihtiyacını doğurmuştur.

Dönüştürücüler, dönüştürücü tabanlı mimarileri desteklemeye ve önceden eğitilmiş modellerin dağıtımını kolaylaştırmaya adanmış kütüphanelerdir. Kütüphanenin merkezinde hem araştırma hem de üretim için tasarlanmış Dönüştürücünün bir uygulaması bulunmaktadır. Felsefesi; okunması, genişletilmesi ve dağıtılması kolay popüler model çeşitlerinin endüstriyel düzeydeki uygulamalarını desteklemektir. Bu temelde kütüphane, merkezi bir model merkezinde çok çeşitli önceden eğitilmiş modellerin dağıtımını ve kullanımını destekler. Bu merkez, kullanıcıların farklı modelleri aynı minimum API ile karşılaştırmasını ve çeşitli farklı görevlerde paylaşılan modelleri denemesini destekler (Transformers, 2023.).

Doğal dil işleme ve makine öğrenimi toplulukları, açık kaynaklı araştırma araçları oluşturma konusunda güçlü bir kültüre sahiptir. Dönüştürücülerin yapısı, öncü tensor2tensor kütüphanesinden ve her ikisi de Google Research'ten olan BERT'in orijinal kaynak kodundan ilham almıştır (Devlin vd., 2018).

Genel amaçlı doğal dil işlemeye yönelik kullanımı kolay, kullanıcıya yönelik kütüphanelerin uzun bir geçmişi vardır. İki temel kütüphane, doğal dil işlemeye yönelik çeşitli farklı yaklaşımları tek bir pakette toplayan NLTK (Loper & Bird, 2002) ve Stanford CoreNLP'dir (Manning, Surdeanu, Bauer, Finkel, Bethard, & McClosky, 2014).

Dönüştürücüler, standart NLP makine öğrenimi modeli hattını yansıtacak şekilde tasarlanmıştır: verileri işleme, bir model uygulama ve tahminlerde bulunma.

Yinelemeli modeller tipik olarak hesaplamayı girdi ve çıktı dizilerinin sembol konumları boyunca etkiler. Konumları hesaplama süresindeki adımlara göre hizalayarak, önceki gizli durumun ve konum girişinin bir fonksiyonu olarak bir dizi gizli durum üretirler. Bu doğası gereği sıralı yapı, bellek kısıtlamaları örnekler arasında gruplamayı sınırladığından, daha uzun dizi uzunluklarında kritik hale gelen eğitim örnekleri içindeki paralelleştirmeyi engeller (Kuchaiev & Ginsburg, 2017).

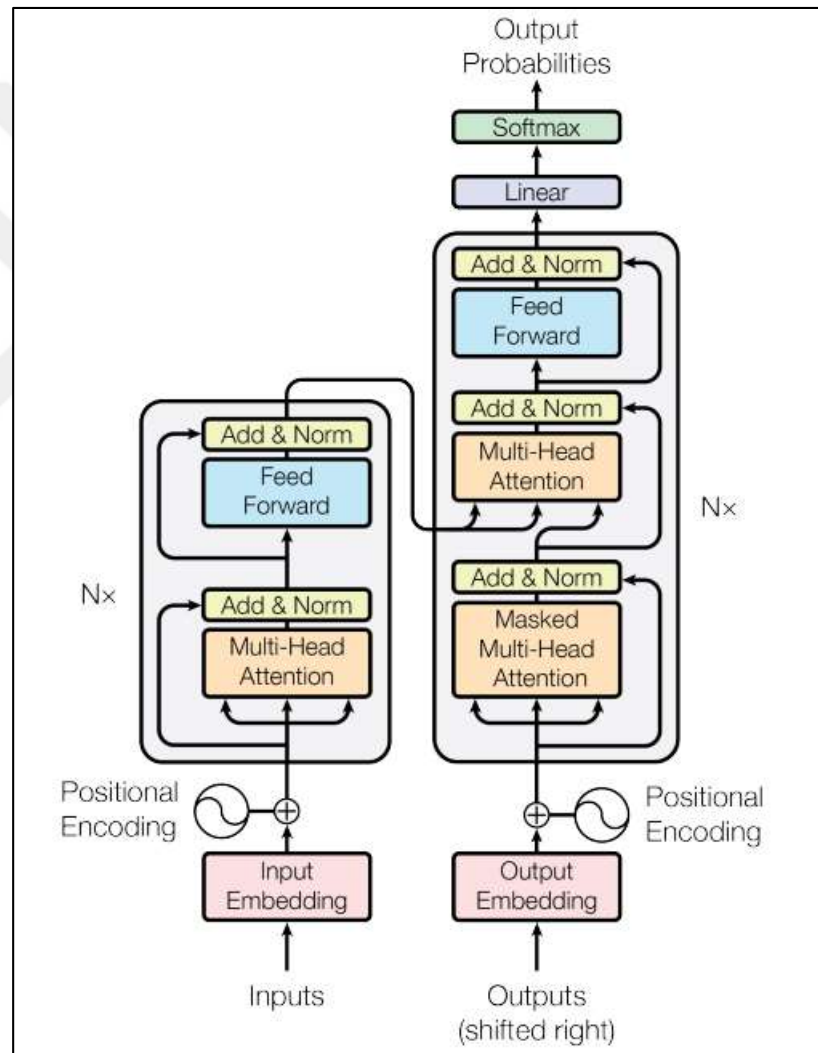
Sıralı hesaplamayı azaltma hedefi aynı zamanda Genişletilmiş Sinirsel GPU (Kaiser vd., 2016), ByteNet (Kalchbrenner vd., 2016) ve ConvS2S'nin (Gehring vd.,

2017) temelini oluşturur; bunların tümü temel yapı taşı olarak evrimsel sinir ağlarını kullanır, tüm girdiler ve çıkış konumları için gizli temsilleri paralel olarak hesaplar. Bu modellerde, iki rastgele giriş veya çıkış konumundan gelen sinyalleri ilişkilendirmek için gereken işlem sayısı, konumlar arasındaki mesafe arttıkça ConvS2S için doğrusal ve ByteNet için logaritmik olarak artar. Bu, uzak konumlar arasındaki bağımlılıkların öğrenilmesini zorlaştırır (Bengio vd., 2003). Dönüştürücülerde bu, ortalama dikkat ağırlıklı konumlar nedeniyle etkili çözünürlüğün azalması pahasına da olsa, sabit bir işlem sayısına indirgenir; bu, Çok Taraflı Dikkat (Multi-head Attention) ile etkisiz hale getirilen bir problemdir.

Bazen iç dikkat olarak da adlandırılan öz dikkat, dizinin temsiliyi hesaplamak amacıyla tek bir dizinin farklı konumlarını ilişkilendiren bir dikkat mekanizmasıdır. Öz-dikkat (Self-Attention), okuduğunu anlama, soyutlayıcı özetleme, metinsel anlamlandırma ve görevden bağımsız cümle temsillerini öğrenme gibi çeşitli görevlerde başarıyla kullanılmıştır (Paulus vd., 2017). Dönüştürücüler, sıralı RNN'ler veya evrim kullanmadan girdi ve çıktı temsillerini hesaplamak için tamamen öz dikkat üzerine kurulu ilk dönüşüm odaklı modeldir (Kaiser & Sutskever, 2015).

4.1.6.1. Dönüştürücü Mimarisi

Birçok sinir ağı dönüştürücü modeli bir kodlayıcı-kod çözücü yapısına sahiptir (Sutskever vd., 2014). Kodlayıcı, sembol temsillerinin bir giriş dizisini temsillerin bir dizisine eşler. Gösterimler verildiğinde, kod çözücü daha sonra her seferinde bir öğe olmak üzere sembollerden oluşan bir çıktı dizisi üretir. Her adımda model otomatik olarak geriye yöneliktir ve bir sonrakini oluştururken daha önce üretilen sembolleri ek girdi olarak kullanır (Graves, 2013b). Dönüştürücü mimarisinin genel gösterimi Şekil 4.7'deki gibidir.



Şekil 4.7. Dönüştürücü mimarisi (Vaswani vd., 2017)

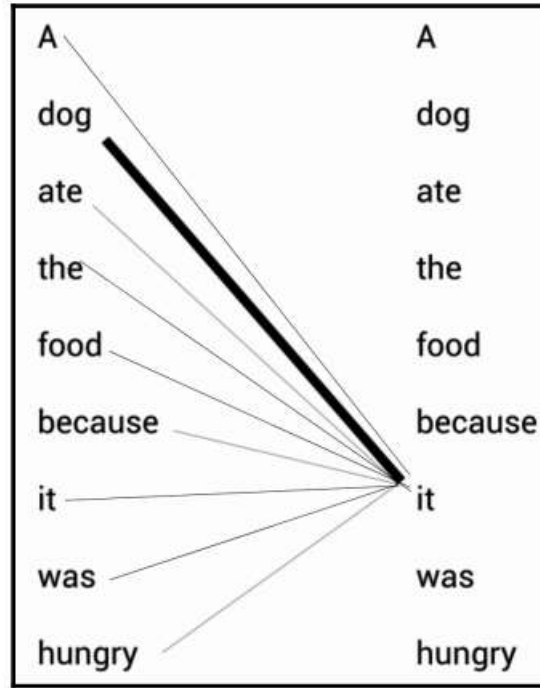
4.1.6.1.1. Kodlayıcı ve Kod Çözücü Yığını

Kodlayıcı genelde 6 özdeş katmandan oluşan bir yığından oluşur. Her katmanın iki alt katmanı vardır. Birincisi çok başlı bir öz-dikkat mekanizması, ikincisi ise konumsal olarak tamamen bağlantılı ileri beslemeli bir ağıdır. İki alt katmanın her birinin etrafında bir artık bağlantı kullanılır ve ardından katman normalizasyonu yapılır. Yani, her bir alt katmanın çıktısı LayerNorm'dur; burada alt katman, alt katmanın kendisi tarafından uygulanan işlemdir. Üst katmanın çıktısı ile işleme giren alt katman bağlantılarının temsilini kolaylaştırmak için, modeldeki tüm alt katmanlar ve yerleştirme katmanlar 512 boyutlu çıktılar üretir.

Kod çözücü de genelde 6 özdeş katmandan oluşan bir yığından oluşur. Her kodlayıcı katmanındaki iki alt katmana ek olarak kod çözücü, kodlayıcı yığınının çıktısı üzerinde çok taraflı dikkat gerçekleştiren üçüncü bir alt katman ekler. Kodlayıcıya benzer şekilde, alt katmanların her birinin etrafında artık bağlantılar kullanılır ve ardından katman normalizasyonu yapılır. Ayrıca, konumların sonraki konumlara katılmasını önlemek için kod çözücü kümesindeki öz-dikkat alt katmanı da değiştirilmiştir. Bu maskeleme, çıktı yerleştirmelerinin bir konum vektörü ile dengelenmesi gerçeğiyle birleştiğinde, belirtilen konum için tahminlerin yalnızca bu konumdan daha uzak konumlardaki bilinen çıktılarına bağlı olabilmesini sağlar.

4.1.6.1.2.Dikkat Fonksiyonu

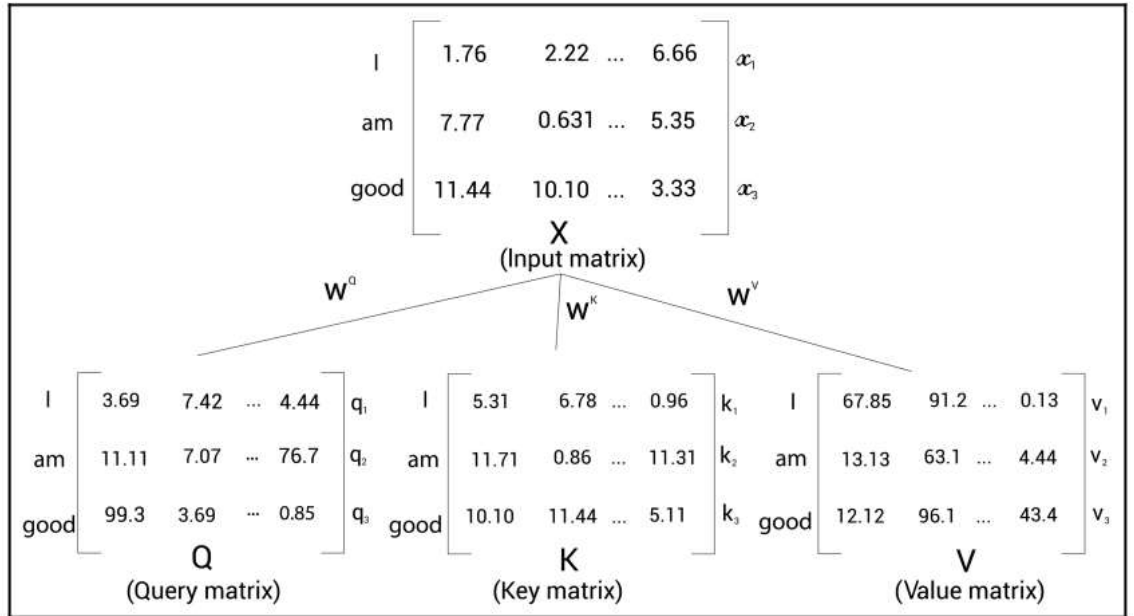
Bir dikkat fonksiyonu, bir sorguyu ve bir dizi anahtar-değer çiftini bir çıktıya eşlemek olarak tanımlanabilir; burada sorgu, anahtarlar, değerler ve çıktı tüm vektörlerdir. Çıktı, değerlerin ağırlıklı toplamı olarak hesaplanır; burada her değere atanan ağırlık, sorgunun karşılık gelen anahtarla uyumluluk fonksiyonu tarafından hesaplanır. Dikkat fonksiyonuna ait gösterim Şekil 4.8'deki gibidir.



Şekil 4.8. Dikkat fonksiyonu hesaplanırken cümle içindeki kelimelerin birbiri ile ilişkileri göz önünde bulundurulur (Ravichandiran, 2021).

Ölçeklendirilmiş Odak Noktalı Dikkatin girdileri, sorgulardan ve boyut anahtarlarından ve boyut değerlerinden oluşur. Sorgunun çarpımı için tüm girdiler kullanılır ve hesaplanır, her biri anahtarların boyutunun kareköküne bölünür ve Softmax fonksiyonu uygulanarak değerlerin ağırlıkları elde edilir. Uygulamada, dikkat fonksiyonunu aynı anda bir Q matrisinde paketlenen bir dizi sorgu üzerinde hesaplanır. Anahtarlar ve değerler aynı zamanda K ve V matrislerinde birlikte işleme sokulmaktadır. Bu işleme ait gösterim şekil 4.9’da verilmiştir. Dikkat fonksiyonu şöyle hesaplanır:

$$Dikkat\ Fonksiyonu(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V \quad (4.12)$$

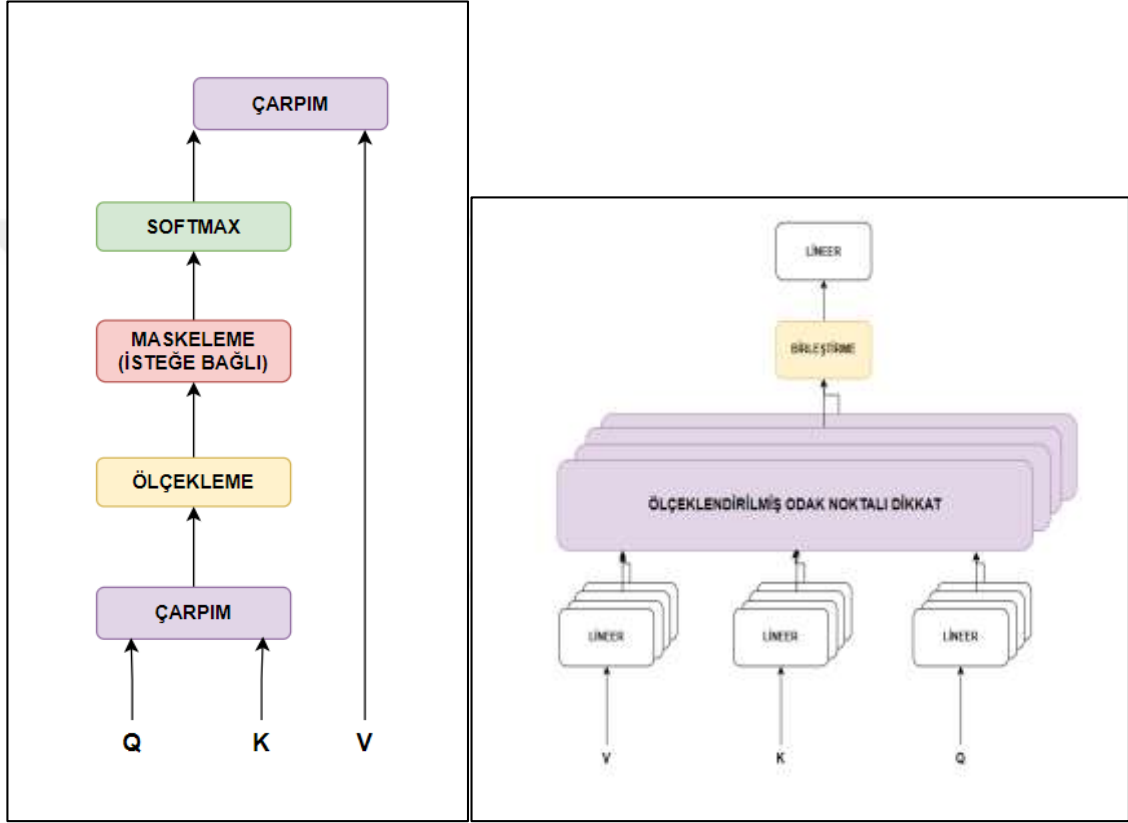


Şekil 4.9. Dikkat matrisleri: Sorgu (Query), anahtar (Key) ve değer (Value) matrislerindeki ilk satır "I" kelimesinin sorgu(Q), anahtar(K) ve değer(V) vektörlerini ifade eder. Sorgudaki ikinci satır, anahtar ve değer matrisleri "am" kelimesinin sorgu(Q), anahtar(K) ve değer(V) vektörlerini ifade eder. Sorgu, anahtar ve değer matrislerindeki üçüncü satır "good" kelimesinin sorgu(Q), anahtar(K) ve değer(V) vektörlerini ifade eder.

4.1.6.1.3. Çok Taraflı Dikkat

D model boyutlu anahtarlar, değerler ve sorgularla tek bir dikkat fonksiyonunu gerçekleştirmek yerine, sorguları, anahtarları ve değerleri birçok kez farklı, öğrenilmiş doğrusal gösterimlerle sırasıyla d_q , d_k ve d_v boyutlarına doğrusal olarak göstermenin

faydalı olduğu görülmüştür. Sorguların, anahtarların ve değerlerin bu öngörülen versiyonlarının her birinde paralel olarak dikkat fonksiyonunu gerçekleştirerek d_v boyutlu çıktı değerleri elde edilir. Bunlar birleştirilir ve bir kez daha işleme sokularak nihai değerler elde edilir. Çok taraflı dikkat, modelin farklı konumlardaki farklı temsil alt alanlarından gelen bilgilere ortaklaşa katılmasına olanak tanır. Çok taraflı dikkat yapısına ait iki gösterim Şekil 4.10'da verilmiştir.



Şekil 4.10. Solda dikkat fonksiyonun gösterimi, sağda ise bunun çoklu dikkatte kullanımı gösterilmiştir.

4.1.6.1.4. Konumsal Kodlama

Model herhangi bir yineleme ve evrişim içermediğinden, modelin dizinin sırasını kullanabilmesi için dizideki belirteçlerin göreceli veya mutlak konumu hakkında bazı bilgileri göstermemiz gerekmektedir. Bu amaçla, kodlayıcı ve kod çözücü yığınlarının altındaki giriş yerleştirmelerine "konumsal kodlamalar" eklenmektedir.

Bunun için farklı frekansların sinüs ve kosinüs fonksiyonlarını kullanılmaktadır:

$$KK_{(konum,2i)} = \sin\left(\frac{konum}{10000 \frac{2i}{boyut_{model}}}\right) \quad (4.13)$$

$$KK_{(konum,2i+1)} = \cos\left(\frac{konum}{10000 \frac{2i}{boyut_{model}}}\right) \quad (4.14)$$

Yukarıdaki denklemlerde i boyuttur. Konumsal kodlamanın her boyutu bir sinüzoid karşılık gelir. Dalga boyları 2π 'den $10000 \times 2\pi$ 'ye kadar geometrik bir devamlılık oluşturur. Bu işlev seçilir çünkü modelin göreceli konumlara göre çalışmayı kolayca öğrenmesine olanak sağlayacağı varsayılır.

4.1.6.1.5. Gömülü Kelime Simgeleri

Kelime yerleştirmeleri veya gömülü kelime simgeleri, kelimeler arasındaki anlamsal ve sözdizimsel benzerlikleri gösteren vektörler kullanılarak kelimelerin temsilidir. En basit yerleştirme modeli, sözcük dağarcığındaki sözcük sayısına eşit boyutta bir vektörün oluşturulduğu one-hot encoding'dir. Son dönemde yapılan doğal dil işleme çalışmalarında daha verimli kelime yerleştirme modelleri geliştirilmiş ve kullanılmıştır. En sık kullanılan kelime yerleştirme modelleri Word2Vec, GloVe, FastText, BERT, RoBERTa, XLM-RoBERTa, GPT-2 ve GPT-3 olarak sıralanabilir (Erkan & Gungor, 2023)

Word2Vec, Mikolov ve arkadaşları tarafından geliştirilen denetimsiz bir yerleştirme modelidir. İki versiyonu vardır: Skip Gram (SG) ve Continuous Bag of Word (CBOW) (Mikolov vd., 2013).

GloVe, eğitimin gerçekleştirildiği kelimedenden kelimeye birlikte oluşum istatistiklerine dayanan başka bir denetimsiz modeldir. Eğitimin çıktısı, kelime vektör uzayının doğrusal alt yapılarını sağlar. İki kelime arasındaki birlikte oluşma olasılık oranını vektör farklılıkları olarak kodlar. Word2Vec'e benzer şekilde GloVe yerleştirmeleri, bağlamdan bağımsız olarak bir belirteç için tek bir yerleştirme sağlar (Erkan & Gungor, 2023).

BERT (Bidirectional Encoder Representations from Transformers- Dönüştürücülerden Çift Yönlü Kodlayıcı Gösterimleri), iki şekilde kullanılabilen çok katmanlı, çift yönlü bir dönüştürücü kodlayıcı (encoder) modelidir. Word2Vec'ten farklı olarak BERT, bir belirtecin (token) yerleştirilmesinin bağlamına bağlı olarak farklı

olacağı anlamında bağlamsal yerleştirmeler (embeddings) üretir. BERT’i herhangi bir doğal dil işleme işlevi için kullanmanın ilk yolu, bir giriş dizisi verildiğinde, kelimelerin dizideki yerleştirmelerinin elde edilebilmesi ve bu yerleştirme vektörlerinin özellik olarak kullanılmasıdır. İkinci bir yol olarak, metin dizisini sınıflandırmak için yalnızca çıkış [CLS] belirteci kullanılabilir (Chen vd., 2022).

RoBERTa, BERT’in bazı değişikliklerle kopyasıdır. Bir sonraki cümle tahmin hedefi modelden çıkarılmıştır. Model, daha büyük yığın (batch) boyutları ve daha fazla veriyle eğitilmiştir. BERT’de ön eğitim için kullanılan veri setine ek olarak, bu modelde yeni ve büyük bir veri seti kullanılmıştır (Y. Liu vd., 2019).

XLM-RoBERTa, RoBERTa modelinin çok dilli bir versiyonudur. Maskeli dil modeli, Common Crawl corpus kullanılarak 100 dilde eğitilmiştir (Wenzek vd., 2019).

4.1.6.1.6. Tokenization (Simgeleştirme)

Simgeleştirme, bir cümleyi simgelere (tokens) bölme işlemidir. Simgeler, cümle içinde yazıldığı gibi sözcüklerden oluşabileceği gibi, sözcüklerin farklı biçimlerinden de oluşturulabilir.

Words (Kelimeler)

Bu yaklaşımda simgeler, boşluklarla ayrılmış karakter dizilerinden oluşur. Bu, doğal dil işleme görevlerinde kullanılan en basit simgeleştirme yaklaşımıdır. Bu modelde bir simge genellikle bir kelimeye karşılık gelir. Ancak cümleye herhangi bir ön işlem yapılmadığından, noktalama işaretleriyle veya başka türde karakterlerle karıştırılmış kelimelere karşılık gelen simgeler olabilir (Erkan & Gungor, 2023).

Words - Preprocessed (İşlenmiş Kelimeler)

Simgeleştirme işleminden önce cümleleri veya belgeleri ön işlemek yaygın bir uygulamadır (Erkan & Gungor, 2023). Bu yaklaşımda aşağıdaki ön işleme işlemlerini kullanabiliriz:

- Cümledeki kelimeleri küçük harfler halinde değiştirmek,
- Bazı noktalama işaretlerini kaldırmak,
- “@” karakterini kelime önlerinden kaldırmak,
- Harf olmayan karakterleri kaldırmak.

Words - No Stopwords (Gereksiz Kelimeler Çıkarılarak)

Gereksiz sözcükleri, bir dildeki her türlü belgede sıklıkla kullanılan ve bu nedenle ayırt edici gücü olmayan sözcüklerdir. Bu simgeleştirme yaklaşımında, Python NLP kütüphanesinin (NLTK) stopword listesini kullanarak cümlelerdeki stopword'ler ortadan kaldırılır (Bird vd., 2009).

Stem

Kök alma (STEM), kelimenin kökünü (kök biçimini) elde etmek için bir kelimedeki son ekleri ayıran, kurala dayalı bir işlemdir (Porter, 1980). Bu yaklaşımda doğal dil işleme modellerinde sözcük kökleri simge olarak kullanılır. Kökler için İngilizce'de Stanford CoreNLP (Manning, Surdeanu, Bauer, Finkel, Bethard, & McClosky, 2014) ve Türkçe için (Sak vd., 2008).’nın morfolojik ayrıştırıcısı buna örnektir.

Lemma

Lemmatizasyon (kök çözümleme) (Miller George A., 1995) , kelimenin kök formunun çıkarılması anlamında kök çıkarmaya (STEM) benzer. Aradaki fark, morfofonetik kuralları ve düzensiz durumları da hesaba katmasıdır. Kökten ayırmaya benzer şekilde, modellerde kelimelerin lemma biçimlerini kullanılır. İngilizce'deki lematizasyon için Stanza (Qi vd., 2020) kullanılmıştır.

Morphemes (Biçimbirim)

Önceki iki yaklaşıma benzer şekilde, morfolojik ayrıştırma bir kelimenin kök biçimini bulur, ancak aynı zamanda tüm son ekleri de çıkarır. Bu simgeleştirme yaklaşımında bir kelime, morfemlerine (kök ve son ekler) bölünür ve her bir morfem ayrı bir simge olarak kullanılır (Carstairs-McCarthy, 2002; Oztaner, 1996).

Byte Pair Encoding

Byte pair encoding (BPE) (Sennrich vd., 2015), bir veri sıkıştırma algoritmasından (Gage, 1994) referans alınarak oluşturulmuştur. Varyantları Google'ın SentencePiece (Kudo & Richardson, 2018) simgeleştirme yönteminde ve OpenAI'nin GPT-3 (T. B. Brown, vd., 2020) modelinde kullanılmaktadır. BPE algoritmasının

arkasındaki motivasyon, bir derlemde, sık sık meydana gelen karakter dizilerinin birlikte temsil edileceği şekilde, derlemdeki simgeleri daha küçük parçalara (alt kelimelere) bölmek için verilmiştir. Bu yaklaşımda alt kelimeler simge olarak kullanılmaktadır.

WordPiece

WordPiece algoritması BPE simgeleştirme yöntemine çok benzer. Aradaki fark, WordPieces'in derlemin dil modeli olasılığını en çok artıran bu dizileri birleştirmesidir (Schuster & Nakajima, 2012).

Unigram Dil Modeli

BPE ve WordPiece, ayrı karakterlerle alt sözcükler olarak başlayan ve her yinelemede ardışık dizileri birleştirerek daha uzun alt sözcükler oluşturan aşağıdan yukarıya algoritmalarıdır. Öte yandan, unigram dil modeli (Kudo, 2018) simgeler ve alt sözcüklerle başlaması ve her yinelemede bu birimlerin bazılarını ortadan kaldırması açısından yukarıdan aşağıya yaklaşıma benzer.

Syllables (Heceler)

Hece, bir sesli harften oluşan telaffuz birimidir. Türkçenin iyi tanımlanmış heceleme kuralları vardır. Kelimelerin içindeki heceler simge olarak kullanılır. Algoritma, kelimenin sonunu tarayarak başlar ve kelimedeki en sağdaki sesli harfi bulur. Bu sesli harfin solundaki harf sesli harf ise, kelimenin en sağ sesli harfinden sonuna kadar olan kısmı hece olarak kabul edilir. Aksi takdirde kelimenin sol ünsüzden sonuna kadar olan kısmı hecedir. Algılanan hece kaldırılır ve algoritma yinelenir. Bir basamakta sesli harf bulunamazsa kalan sözcük hece olarak kabul edilir (Büyükkuşcu vd., 2016).

Partial Surface Form (Kısmi Yüzeysel Biçim)

Bir kelimenin kısmi yüzey biçimi, kök biçimi ile yüzey biçimi arasında bir ara biçimdir. Bu fikirden esinlenerek, çekim eklerinin kelimedden çıkarıldığı, yapım eklerinin ise muhafaza edildiği bir simgeleştirme yöntemi önerilmiştir (Rıfkı Aydın & Güngör, 2021). Kelimenin kısmi yüzey formu ve çekim ekleri elde edildikten sonra her biri ayrı bir simge olarak kullanılır.

4.1.6.2. Tez Kapsamında Kullanılan Büyük Dil Modelleri

Büyük dil modelleri (Large language models-LLMs), GPT’ler gibi, dili işlemek için tasarlanmış derin öğrenme sistemleridir. Büyük dil modelleri, çok sayıda metin koleksiyonu kullanmaktadır ve hedefleri, bir cümle veya paragraftaki bir sonraki kelimeyi (veya eksik bir kelimeyi) başarılı bir şekilde tahmin etmektir. Bu hedefe ‘dil modelleme’ denir. Bunu karşılamak, en azından bir dildeki kelimelerin ve sözdizimsel yapıların dağılım özelliklerine hâkim olmayı, yani cümlelerin nasıl olduğunu öğrenmeyi gerektirir. Bu nedenle, büyük dil modellerinin ifadelerin nasıl davrandığını modellemesi beklenmektedir. Bu konuda başarıları dikkat çekicidir: neredeyse her türlü rastgele cümleyi, dilbilgisel ve anlamlı tahmin edilen devamların çıktısını verecek şekilde temsil edebilir ve işleyebilirler (örneğin, sözcükler yanlış, mantıksal olarak kusurlu veya önyargılı olsa bile insanlar tarafından yorumlanabilir içerikte uygun şekilde birleştirilebilir). Bu anlamda büyük dil modelleri, doğal dil işleme için en iyi uygulanan hesaplamalı modeldir: doğal dili geniş ölçekte ele alan ilk sistemlerdir. Büyük dil modellerinin olasılıksal sonraki kelime tahminlerini test eden deneyler, bu modellerin eğitim girdilerinden birçok karmaşık dilsel genelleme çıkarmış olabileceğini bulmuştur. Bu genellemelerden bazıları, BERT ve GPT-2 gibi büyük dil modellerinin nesillerinde zaten gözlemlenebilmektedir (Wilcox vd., 2022).

Büyük dil modelleri, kelimeleri tahmin etmek için eğitilir ve insan zihnindeki dilsel tahmin, merkezi, sağlam ve her yerde bulunan bir süreçtir (Blank, 2023).

Geçtiğimiz birkaç yılda Büyük Dil Modelleri doğal dil işleme alanını önemli ölçüde dönüştürdü. Daha geniş anlamda, bu gelişme insan-bilgisayar etkileşiminde bir paradigma değişikliğine sebep olmuştur. Bu ilerleme geniş kapsamlı ekonomik sonuçların yanı sıra yapay zekaya ve hatta belki de bilişle ilişkin kavramsal düşüncelerimizi yeniden inşa ediyor. Üstelik GPT-4 gibi en yeni nesil modeller, önceki modellere göre kayda değer gelişmeler göstererek, daha önce kısa vadede ulaşılamayacağı düşünülen yetenekleri sunmaktadır. Bir doğal dil modeli neslinden diğerine geçiş, şu anda öncelikle ölçekten yani parametre sayısına göre değişen model büyüklüğünden kaynaklanıyor gibi görünüyor; en güçlü modeller trilyonlarca parametreye ve eğitim verileri için trilyonlarca belirtece (token) yaklaşmaktadır. Bu sebepten büyük ölçek, yüksek düzeyde kabiliyet kapasitesi elde etmek için vazgeçilmez

midir sorusu akla gelmektedir. Ekonomik açıdan bu tür büyük modellerin eğitimi, dağıtımı ve bakımının maliyeti önemli ölçüde yüksek olabilir. Bilimsel olarak, benzer yeteneklerin daha küçük ölçekte elde edilip edilemeyeceğini anlamak, akıllı sistemlerin mimarileri ve geliştirilmesi hakkında fikir vermektedir. Kullanıldığı alanda verimli çalışan bir yapay zeka açısından bakıldığında, büyük ölçekli modellerin enerji tüketimi ve bu büyük modellerin ne kadar kontrol edilebilir veya yönetilebilir olabileceği sorusu giderek artan bir endişe haline gelmektedir. Son olarak, belli görevler için yoğun olarak kullanılan modelleri son teknoloji yeteneklerle eğitme yeteneği, gelişmiş yapay zekayı her türlü son kullanıcı ile buluşturacak ve geniş bilgi işlem kaynaklarına sahip birkaç kişiye özel bir alan olmak yerine, daha geniş bir yelpazedeki bireylerin ve kuruluşların bunları incelemesine ve dağıtmasına olanak tanıyacaktır (Li vd., 2023).

Büyük Dil Modelleri, programlama ve yaratıcı yazarlık gibi özel alanlar da dahil olmak üzere çok çeşitli alanlarda uzman bilgisi gerektiren karmaşık akıl yürütme görevlerinde başarılı olan, son derece yetenekli yapay zeka asistanları olarak büyük umut vaat etmektedir. Eğitim metodolojisi basit olmasına rağmen yüksek hesaplama gereksinimleri, büyük dil modellerinin geliştirilmesini birkaç büyük kullanıcıyla sınırlamıştır. Kullanıcı dostu sohbet arayüzleri aracılığıyla insan etkileşimini kolaylaştırabildikleri için halk arasında hızla ve yaygın bir şekilde benimsenmiştir. Ancak en başta büyük dil modelleri eğitim metodolojisinin önemli hesaplanma talepleri, bunların geliştirilmesini yalnızca bir önemli ölçüde az denilebilecek katılımcıyla sınırlamıştır. Popüler büyük dil modeli tabanlı uygulamalar, kapalı kaynaklı büyük dil modellerine dayalı olarak oluşturulmuş OpenAI ChatGPT, Google Bard ve Anthropic Claude'dur(Touvron vd., 2023).

4.1.6.2.1. PHI-2

Phi-2, 2,7 milyar parametrelili Microsoft'un geliştirmiş olduğu bir dil modelidir. Doğal dil işleme ve kodlama için Sentetik ve Web veri kümelerinin bir karışımı üzerinde birden fazla geçişten elde edilen 1,4 milyar simge ile eğitilmiş, sonraki kelime tahmin hedefine sahip Transformer tabanlı bir modeldir. Phi-2 eğitimi 96 adet A100 GPU üzerinde 14 gün sürmüştür. Dil anlayışı ve mantıksal akıl yürütmeyi test eden kıyaslamalarla karşılaştırıldığında Phi-2, 10 milyardan az parametreye sahip modeller arasında neredeyse son teknoloji ürünü bir performans sergilemiştir. Phi-2, en başta insan

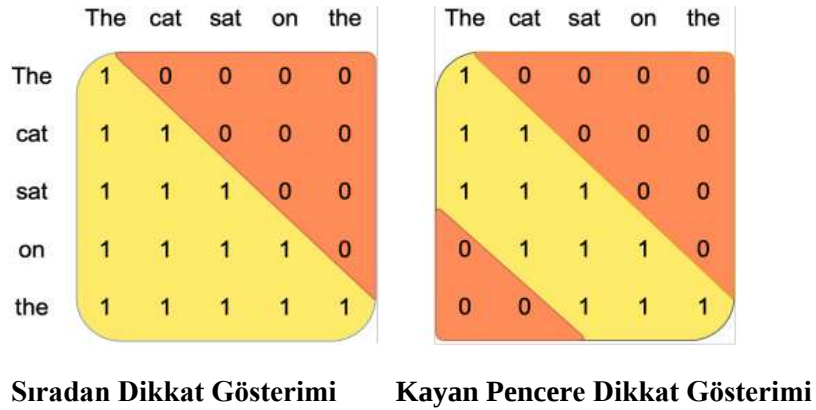
geri bildiriminden pekiştirmeli öğrenme yoluyla herhangi bir dengelemeye tabi tutulmamış ve talimatlarda ince ayar yapılmamış bir temel modeldi. Phi-2 yalnızca araştırma amaçlıdır. Eğitim verilerinin doğası göz önüne alındığında, Phi-2 modeli soru yanıtlama formatını, sohbet formatını ve kod formatını kullanan yönlendirmeler için en uygun modellerden biridir. Modelin hatalı kod parçacıkları ve ifadeler ürettiği gözlemlenmiştir. Kullanıcıların bu çıktıları kesin veya doğru çözümler olarak değil, öneri veya başlangıç noktası olarak ele almaları önerilmiştir. Phi-2 eğitim verilerinin çoğunluğu Python’a dayalıdır ve “typing, math, random, koleksiyonlar, datetime, itertools” gibi ortak paketleri kullanmaktadır. Model, diğer paketleri veya diğer dillerdeki komut dosyalarını kullanan Python komut dosyaları oluşturmaktan dolayı, kullanıcıların tüm API kullanımlarını manuel olarak doğrulaması önerilmiştir. Modelde talimat olarak en başta ince ayarın yapılabileceği bilgisi verilmemiştir. Model öncelikle standart İngilizceyi anlamak için tasarlanmıştır. Resmi olmayan İngilizce, argo veya başka herhangi bir dilin anlaşılmasında zorluk oluşturabilir ve olası yanlış yorumlamalara veya yanıtlarda hatalara yol açabilir. Phi-2, eğitim veri güvenliğini sağlamaya yönelik çabalara rağmen toplumsal önyargılardan tamamen arınmış değildir. Özellikle istendiğinde veya talimat verildiğinde, bu toplumsal önyargıları yansıtan içerik üretme olasılığı vardır. Model, dikkatlice seçilmiş verilerle eğitilmiş olmasına rağmen, açıkça istendiğinde veya talimat verildiğinde yine de zararlı içerik üretebilir. Model yalnızca araştırma amacıyla yayınlanmıştır. Temel bir model olan Phi-2, genellikle ilk sorgulamalarında kullanıcı istemlerine ilk yanıtını takiben alakasız veya fazladan metin ve yanıtlar üretebilir. Bunun nedeni, eğitim veri setinin öncelikle ders kitapları olması ve bunun da ders kitabı benzeri yanıtlarla sonuçlanmasıdır (Javaheripi & Sébastien, 2023)

4.1.6.2.2. MISTRAL 7B

Hızla gelişen Doğal Dil İşleme alanında, daha yüksek model performansına yönelik yarış çoğu zaman model boyutunun artırılmasını gerektirir. Mistral 7B, dikkatle tasarlanmış bir dil modelinin, etkili bir çıkarımı sürdürürken yüksek performans sağlayabileceğini göstermektedir. Ancak bu ölçeklendirme, hesaplama maliyetlerini ve çıkarım gecikmesini artırma eğiliminde olduğundan pratik, gerçek dünya senaryolarında dağıtımın önündeki engelleri yükseltir. Bu bağlamda hem üst düzey performans hem de verimlilik sağlayan dengeli modellerin araştırılması kritik önem kazanmaktadır. Mistral

7B, Apache 2.0 lisansı altında piyasaya sürülmüştür. Mistral 7B, büyük dil modellerini verimli tutarken yüksek performans elde etme hedeflerini dengeleme konusunda önemli bir adım atmıştır. Model, daha hızlı çıkarım için gruplandırılmış sorgu dikkatinden (grouped-query attention-GQA) yararlanır ve daha düşük çıkarım maliyetiyle keyfi uzunluktaki dizileri etkili bir şekilde işlemek için kayan pencere dikkatiyle (sliding window attention-SWA) birleşir. Mistral 7B'nin Hugging Face ile entegrasyonu kolaylaştırılmıştır. Üstelik Mistral 7B, sayısız görevde ince ayar kolaylığı sağlayacak şekilde tasarlanmıştır. Model, aşağıda açıklaması verilen özellikleri beraber kullanarak önemli derecede başarı sağlamıştır.

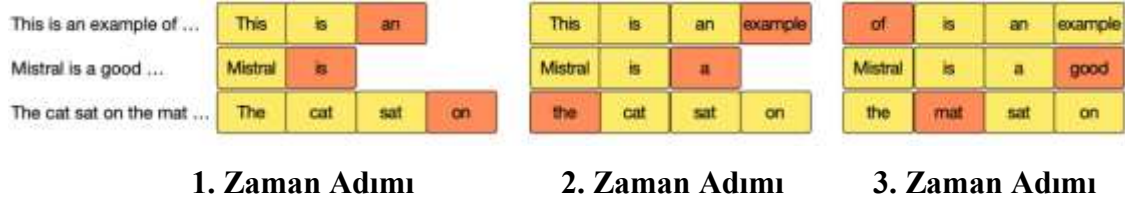
Kayan Pencere Dikkati: Standart dikkatteki operasyonların sayısı dizi uzunluğunda ikinci derecedendir ve bellek, simge sayısı ile doğrusal olarak artar. Çıkarım zamanında bu, önbellek kullanılabilirliğinin azalması nedeniyle daha yüksek gecikme süresine ve daha düşük aktarım hızına neden olur. Bu sorunu hafifletmek için kayan pencere dikkati kullanılmaktadır. Her simge bir önceki katmandaki en fazla seçilen sayıda belirtece göre değerlendirilebilir. Kayan pencerenin dışındaki simgelerin bir sonraki kelime tahminini hâlâ etkilediği unutulmamalıdır. Her dikkat katmanında bilgi seçilen sayıdaki simge kadar ilerleyebilir. Şekil 4.11'de kayan pencere dikkati ile ilgili bir gösterim vardır (A. Q. Jiang vd., 2023).



Şekil 4.11. Kayan pencere dikkati

İlerleyen Tampon Önbelleği: Sabit bir dikkat aralığı, ilerleyen arabellek önbelleği kullanarak önbellek boyutumuzu sınırlayabileceğimiz anlamına gelir. Önbelleğin sabit bir boyutu vardır ve belirli bir zaman adımına ilişkin anahtarlar ve değerler, önbelleğin o belirli zaman adımının önbelleğinin boyutunun modu konumunda saklanır. Sonuç olarak,

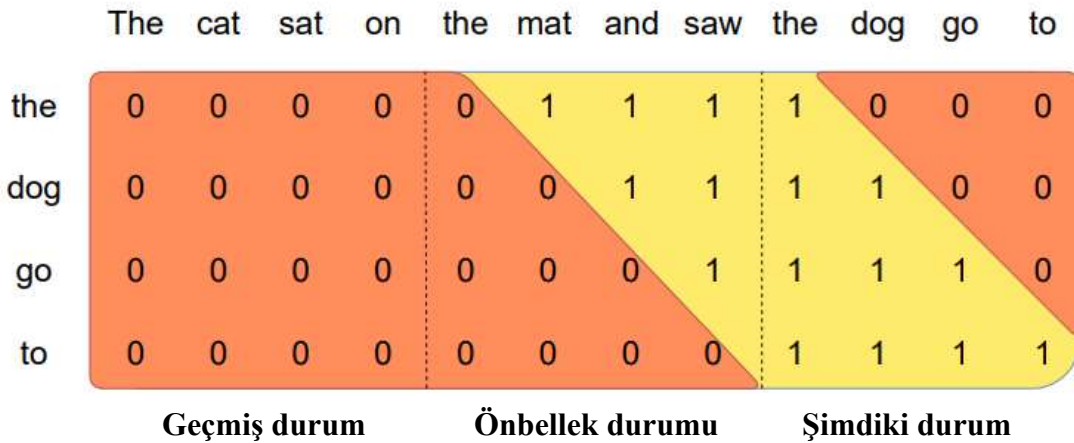
o konum önbelleğin konumundan büyük olduğunda, önbellekteki geçmiş değerlerin üzerine yazılır ve önbellek boyutunun artması durur. Şekil 4.12’de ilerleyen tampon ile ilgili bir örnek verilmiştir (A. Q. Jiang vd., 2023). En son oluşturulan simgelere karşılık gelen gizli durum turuncu renktedir.



Şekil 4.12. İlerleyen tampon gösterimi

Ön doldurma ve Parçalama: Bir dizi oluştururken, her bir simge bir öncekine göre koşullandırıldığı için simgeleri tek tek tahmin etmemiz gerekir. Ancak istek önceden bilinmektedir ve önbelleğini istemle önceden doldurabiliriz. İstek çok büyükse, onu daha küçük parçalara bölebilir ve önbelleği her bir parçayla önceden doldurabiliriz. Bu amaçla pencere boyutunu parça boyutu olarak seçebiliriz. Dolayısıyla her bir parça için, önbellek ve parça üzerindeki dikkati hesaplamamız gerekir.

Şekil 4.13, dikkat maskesinin hem önbellek hem de yığın (parça) üzerinde nasıl çalıştığını göstermektedir (A. Q. Jiang vd., 2023). Önbelleğin önceden doldurulması sırasında, bellek kullanımını sınırlamak için uzun diziler parçalanır.



Şekil 4.13. Dikkat maskesinin hem önbellek hem de yığın (parça) üzerinde nasıl çalıştığının gösterimi

4.1.6.2.3. GPT-2

Dönüştürücü tabanlı modeller ile birçok doğal dil işleme alt alanında çok yönlü kullanılabilir uygulama örnekleri ortaya çıkmıştır. Bu alanlardan biri, dil bilgisi açısından doğru, tutarlı ve anlamlı koşullu sentetik metinlerin üretilmesini gerektiren dil üretimidir. Open AI tarafından yayımlanan GPT-2 (Generative Pre-trained Transformer 2) modeli, büyük metin dizileri üretebilen, rastgele bir girdinin stiline ve içeriğine uyum sağlayabilen dönüştürücü tabanlı bir dil modelidir. Ayrıca GPT-2, sınıflandırma gibi çeşitli doğal dil işleme görevlerini de gerçekleştirebilir. GPT-2, 1,5 milyar parametreyle bir metindeki önceki kelimelerin tümü göz önüne alındığında bir sonraki kelimeyi tahmin edecek şekilde eğitilmiştir. Mimarisi, girdilerin sonraki katmanları etkinleştiren birden fazla nöron katmanı tarafından işlendiği ve son katmanda modelin çıktısının oluşturulduğu derin bir sinir ağı uygular. Bu dil modeli, dönüştürücü mimarisini kullanarak dikkat mekanizmasından yararlanır ve modelin seçici olarak en alakalı girdi parçalarına odaklanmasına olanak tanır. Hem GPT-2 hem de BERT, Transformer mimarisini temel almasına rağmen, BERT (Shobana & Murali, 2022) kodlayıcı bloklardan oluşurken, GPT-2 yığılı kod çözücü bloklardan kullanılarak oluşturulmuştur. Geleneksel dil modellerinde olduğu gibi, GPT-2 de kelime vektörlerini alır ve her kelimenin önceki simgelerin bağlamına sahip olduğu bir sonraki kelimenin olasılığına ilişkin tahminler sağlar. Her simge oluşturulduktan sonra girdi dizisine eklenir ve bir sonraki adımda modelin girdisi olur. GPT-2, BERT gibi bir öz-dikkat katmanı yerine, hesaplanan konumun sağındaki simgelerden gelen bilgileri engelleyen gelecekteki simgeleri maskeleyen maskelenmiş bir öz-dikkat kullanır. Dil temsillerine yönelik bir ince ayar yaklaşımı olan GPT-2, önceden eğitilmiş tüm parametrelerin yalnızca ince ayarını yaparak aşağı yönlü bir görev üzerinde eğitime olanak tanır. Bu nedenle modelin son katmanları, alana özgü eğitim verileri kullanılarak belirli bir göreve veya alana uyarlanabilir (Schneider vd., 2021).

4.1.6.2.4. XGLM

XGLM, toplam 500 milyar alt simgeden oluşan çeşitli dillerden oluşan dengeli bir korpus üzerinde eğitilmiş 7,5 milyar parametreye sahip, çok dilli bir otoregresif dil modelidir. 20'den fazla temsili dilde few-shot learning (Az Vuruşlu Öğrenme) ile yeni bir teknoloji ortaya koymaktadır ve çok dilli tutarlı akıl yürütmede karşılaştırılabilir

boyutta GPT-3'ten daha iyi performans göstermektedir. XGLM, İngilizce komutların (prompt) İngilizce olmayan örneklerle birlikte kullanılmasının diğer öğrenme şekillerine göre daha çok uğraş gerektiren zero (Sıfır Vuruşlu Öğrenme) and few shot learning'e dayalı, performans sağladığı güçlü bir diller arası yetenek sergilemektedir. XGLM, dil tamamlama ve çıkarım görevlerinde, güçlü sıfır ve az vuruşlu öğrenme performansı elde etmiştir. XGLM ayrıca FLORES-101 karşılaştırmasında çok sayıda dil çifti boyunca birkaç çekimli makine çevirisi konusunda yeni bir son teknoloji ürünü oluşturarak 6,7 milyar parametrelili GPT-3 modelini önemli ölçüde geride bırakmıştır (Goyal vd., 2021). Güçlü bir komut dili (örneğin İngilizce) kullanıldığında, başka bir dilden örneklerinin kullanılması, çoğunlukla sıfır vuruşlu öğrenme performansını daha fazla geliştirmemiştir; bu, diller arası ön eğitim ve bağlam içi transfer yaklaşımlarında iyileştirme için hala doldurulması gereken yerler olduğunu göstermektedir (Victoria Lin vd., 2022).

4.1.6.2.5. LLAMA 2

Llama 2, ölçeği 7 milyar ila 70 milyar parametre arasında değişen, önceden eğitilmiş ve ince ayar yapılmış büyük dil modellerinden oluşan bir koleksiyondur. Test edilen bir dizi yararlılık ve güvenlik kıyaslamasında, Llama 2 modeli genel olarak mevcut açık kaynaklı modellerden daha iyi performans göstermektedir. Ayrıca, en azından insan tarafından gerçekleştirilen değerlendirmeler açısından, bazı kapalı kaynak modelleriyle aynı seviyede görünmektedirler. Güvenliğe özel veri açıklaması ve ayarlamaların yanı sıra kırmızı takım oluşturma ve yinelemeli değerlendirmeler kullanılarak bu modellerin güvenliğini artırmak için önlemler alınmıştır. Llama 1'in güncellenmiş bir sürümü olan Llama 2, halka açık verilerin yeni bir karışımıyla eğitilmiştir. Aynı zamanda ön eğitim külliyatının boyutu %40 artırılmış, modelin bağlam uzunluğu iki katına çıkarılmış ve gruplandırılmış sorgu dikkati benimsenmiştir. Llama 2'nin 7 trilyon, 13 trilyon ve 70 trilyon parametrelerine sahip varyantları piyasaya sürülmüştür.

4.1.6.3. Popüler Üretici Büyük Dil Modelleri

4.1.6.3.1. GPT-3

GPT-3, önceki modellere göre çok daha fazla parametre üzerinde eğitilmiştir. GPT-1 117 milyon ve GPT-2 1,5 milyar parametreyle eğitilmiş iken GPT-3 yaklaşık 175

milyar eğitilebilir parametre içeren otoregresif bir dil modelidir. Bu dil tahmin modeli, her çalıştırıldığında belirleyici olmayan yanıtlar üretme kapasitesine sahiptir ve bu da metin oluşturma ve dönüştürmede esneklik sağlamaktadır. Genel olarak ayarlanabilir parametrelerin sayısı ne kadar yüksek olursa model o kadar karmaşık olmaktadır. Bununla birlikte, parametrelerin ölçeğini büyütmek doğrudan etkili gibi görünebilir ancak ucuz değildir; bu modellerin eğitimi çok maliyetlidir. GPT-3'ün kapsamlı veriler üzerinde gözetimsiz eğitiminin, içerdiği milyarlarca eğitilebilir parametreye atfedilen tahmini 12 milyon dolara mal olduğu söylenmektedir. Kasım 2022'de OpenAI, dünyanın en gelişmiş yapay zeka yazma araçlarından biri olan ve insan benzeri metinler oluşturabilen ve modelin özel olarak gerektirmediği görevleri gerçekleştirebilen bir API (Uygulama Programlama Arayüzü) aracılığıyla GPT-3'ü genel erişime açtı. Bu doğrultuda çözümlere ulaşmak için yapay zekayı kullanan yeni teknolojik atılımlar yaşandı ve ön saflarda OpenAI'nin GPT-3'ü yer aldı. Kısacası GPT-3 modeli, bir metin dizisini bilgi istemi (girdi) olarak alan ve hangi kelimenin veya metin dizisinin bir sonraki sırada gelmesi gerektiğini veya büyük olasılıkla geleceğini bilinçli olarak tahmin eden bir yapay zekadır. Bunu başarmak için OpenAI, GPT-3'ün internetteki, haber makaleleri, forum gönderileri, web siteleri vb. içindeki milyarlarca bilgiden öğrenmesini sağlamıştır. GPT-3, Few-shot learning (büyük miktarda eğitim verisi sağlama normunun aksine, modele çok az miktarda eğitim verisi verme uygulaması) dil modeli olarak, yaklaşık yarım trilyon kelime üzerinde eğitilmiş ve ince ayar yapılmadan çeşitli doğal dil işleme ölçütlerinde son teknoloji performansa ulaşmıştır (Saravanan & Sudha, 2022).

4.1.6.3.2. GPT-4

GPT-4, hem kamuya açık verileri (internet verileri gibi) hem de üçüncü taraf sağlayıcılardan lisanslanan verileri kullanarak, bir belgedeki bir sonraki belirteci tahmin etmek için önceden eğitilmiş dönüştürücü tarzı bir modeldir. Daha sonra modelde İnsan Geri Bildiriminden yararlanılarak Takviyeli Öğrenme (Christiano vd., 2017) kullanılarak ince ayar yapılmıştır. GPT-4, diyalog sistemleri, metin özetleme ve makine çevirisi gibi çok çeşitli uygulamalarda kullanıma potansiyeline sahip olan önemli bir LLM'dir. Bu nedenle, son yıllarda önemli bir ilgi ve ilerleme konusu olmuştur (O. Evans vd., 2021). Bu tür modeller geliştirmenin ana hedeflerinden biri, özellikle daha karmaşık ve incelikli senaryolarda, doğal dildeki metinleri anlama ve oluşturma yeteneklerini geliştirmektir.

Bu tür senaryolarda yeteneklerini test etmek için GPT-4, başlangıçta insanlar için tasarlanmış çeşitli sınavlarda değerlendirilmiştir. Bu değerlendirmelerde GPT-4 oldukça iyi performans göstermekte ve genellikle insan test katılımcılarının büyük çoğunluğunu geride bırakmaktadır. Örneğin, simüle edilmiş bir baro sınavında GPT-4, sınava girenlerin en üst %10'unda yer alan bir puan elde etmektedir. Bundan farklı olarak, GPT-3.5 en düşük %10'luk dilimde yer almıştır. GPT-4, geleneksel doğal dil işleme ölçütlerinde hem önceki büyük dil modellerinden hem de en son teknoloji sistemlerden (genellikle ölçütlere özgü eğitim veya el mühendisliğine sahip olan) daha iyi performans göstermektedir (Hendrycks vd., 2020).

Yeteneklerine rağmen, GPT-4'ün önceki GPT modelleriyle benzer sınırlamaları vardır: tam olarak güvenilir değildir, sınırlı bir bağlam penceresine sahiptir ve deneyimlerden ders almaz (Radford vd., 2019). Özellikle güvenilirliğin önemli olduğu bağlamlarda GPT-4 çıktıları kullanılırken dikkatli olunmalıdır. GPT-4'ün yetenekleri ve sınırlamaları, önemli ve yeni güvenlik sorunları yaratmaktadır ve potansiyel toplumsal etki göz önüne alındığında, bu zorlukların dikkatli bir şekilde incelenmesinin önemli bir araştırma alanı olduğuna inanılmaktadır (Achiam vd., 2023).

4.1.6.4. Gelişmiş Dil Model Uygulamaları

4.1.6.4.1. ChatGPT

Modern teknoloji büyük ölçüde insan zihnini taklit etmek ve bize farklı şekillerde yardımcı olmak için gizlice çalışan Yapay Zekaya (AI) dayanmaktadır (Kaplan Jerry, 2024). Yakın tarihteki ilerlemeler, ChatGPT gibi yapay zeka destekli sohbet robotlarının geliştirilmesinde ve piyasaya sürülmesiyle somutlaştı ve yapay zekanın ne kadar ilerlediğini kamuoyuna göstermiştir (Susnjak, 2022). ChatGPT-3, OpenAI tarafından oluşturulan gelişmiş bir dil geliştiren yapay zeka standardı olan GPT-3'ün yükseltilmiş bir formu kullanılarak geliştirildi (Sohail, 2024). GPT-3'te kullanılan Derin Öğrenme Sinir Ağı, neredeyse 175 milyar Makine Öğrenimi (ML) parametresine sahiptir. Bunun ne kadar büyük olduğunu düşünürsek, GPT-3'ten önce edinilen en büyük dil modeli, Microsoft'un 10 milyar parametre içeren Turing-Natural Language Generation (T-NLG) dil modeliydi. 2021'in başında GPT-3, şimdiye kadar oluşturulmuş en büyük Sinir

Ağıydı. Bir insan tarafından yazılmış gibi görünen içerik oluşturma konusunda GPT-3, önceki tüm sürümlerden daha iyidir (Khalil & Er, 2023).

ChatGPT sohbet robotu, OpenAI GPT-3 dil yapısı kullanılarak oluşturulmuştur. ChatGPT, sohbet ortamına girilen operatör verilerine insan yanıtları gibi görünen metin yanıtları oluşturmayı amaçlamaktadır. Geniş bir insan iletişim veri tabanının (eğitim verileri) yardımıyla OpenAI ChatGPT, çeşitli konulara ve ipuçlarına yanıtlar sağlamak üzere geliştirilmiştir. Üretken yapay zeka dil modellerinin anlayışının, eğitim verilerinden öğrendikleri kalıp ve yapılara dayandığı, bu modellerin insanın bağlamsal farkındalığı ve anlayışından yoksun olduğu belirtilmektedir (Bozkurt, 2023). Yine de ChatGPT birçok farklı dilde faydalı yanıtlar sunabilmektedir. Bazı medya makaleleri, chatbot'un 100'e yakın dili desteklediğini belirtmiş ve bir araştırma çalışması, chatbot'un 37 dildeki performansını değerlendirmiştir (Lai vd., 2023). Bu nedenle ChatGPT, örneğin dil çevirisi, müşteri desteği ve içerik geliştirme faaliyetleri gibi alanlarda dünya çapındaki kullanıcılar için oldukça faydalı olabilme imkanına sahiptir. OpenAI API, ChatGPT'yi erişilebilir hale getirerek programcıların bunu uygulamalarına ve cihazlarına dahil etmesine ve kullanmasına olanak tanımıştır (Gozalo-Brizuela & Garrido-Merchan, 2023).

ChatGPT hem iş dünyasında hem de akademik çevrelerde büyük ilgi uyandırmıştır (Farhat vd., 2023). Uygulama arayüzü hem geleneksel hem de sosyal medyada büyük ilgi görmüştür (Karanouh vd., 2023). Pek çok destekçi, Üretken Yapay Zeka'nın büyük potansiyelini öne sürerken, bazı şüpheciler ChatGPT etrafındaki söylemin abartılı dilin hakimiyetinde olduğunu belirtmektedir. Bazı ünlü eleştirmenler ciddi uyarılarda bulunmakta ve ChatGPT gibi yapay zeka modellerinin insan uygarlığına tehdit oluşturabileceğini iddia etmektedir (Chomsky Noam, 2023).

Silikon Vadisi, ChatGPT'nin gelişiminin merkez üssü olmuştur ve en tanınmış iş ve teknoloji figürlerinden bazıları, chatbot teknolojisinin geliştirilmesinde ve finansmanında yer almıştır. ChatGPT'yi yaratan işletme olan OpenAI, 2015 yılında Greg Brockman, Elon Musk, Ilya Sutskever, Wojciech Zaremba, Peter Thiel ve diğer teknoloji geliştiricileri tarafından bir hayır kurumu olarak başlatıldı. Amacı, yapay zekanın çalışmasını genel nüfusa açık bir şekilde sunarak yapay zekanın herhangi bir merkezden kontrol edilmesini önlemektir. OpenAI'nin internet sitesinde 11 Aralık 2015 tarihinde

yayınlanan bilgiye göre şirket, yapay zekayı insanlığa en çok fayda sağlayacak şekilde geliştirmeyi hedefliyor (Khan vd., 2023).

Derin Sinir Ağı (GPT-3), oldukça büyük bir cümle veritabanı kullanılarak OpenAI tarafından zaten test edildi ve cümle oluşturma veya sorgulara yanıt verme gibi amaçlar için işlevselliği geliştirildi. Chat GPT-3'ün arkasındaki temel teknoloji budur. Izgara (Grid), girilen cümleyi analiz eden ve sonuçları gösteren çeşitli dönüştürücü birimlerden oluşturulmuştur. Ayrıca bağlantı, çeşitli kelime ve terimlerin birbiriyle ve tartışmalarla ilgili öneminin bir bütün olarak değerlendirilmesine olanak tanıyan dikkat içi özelliklere de sahiptir. Onu yaratanlar ayrıca ChatGPT-3'ün minimum bilgiden bile anlamlı cümleler üretmesine olanak tanımasını sağlamıştır (Gao vd., 2022).

Doğal Dil İşleme alanında kayda değer bir gelişme, büyük hacimli bilgileri eşzamanlı olarak analiz etmek ve bir insanın yorumlayabileceğine daha yakın bir dil oluşturmak için dönüştürücü yapıya sahip bir yapı kullanan ChatGPT'dir (Jeblick vd., 2023). Bununla birlikte, ChatGPT Web'e bağlanamamaktadır ve yalnızca geliştirilmesi sırasında öğrendiği İnternet'i kullanarak çalışabiliyor, bu da onun dışarıdan bilgi edinme yeteneğini kısıtlamaktadır (Rudolph vd., 2023).

ChatGPT, lansmanından bu yana birçok demografik grup arasında popülerlik kazandı. Ancak bu olaya yanıt nispeten karışık. Bazıları ChatGPT'yi faydaları ve gelecekteki potansiyeli nedeniyle överken, diğerleri daha şüpheci olmaya devam ediyor ve eksiklikleri, kısıtlamaları ve olası dezavantajları nedeniyle onu eleştirmiştir (Srivastava, 2023).

4.1.6.4.2. Gemini

Teknolojik değişim, bilimsel keşifleri ilerletme, insanlığın ilerlemesini hızlandırma ve yaşamları iyileştirme fırsatını beraberinde getirmiştir (Pichai Sundar & Hassabis Demis, 2023). Yapay zeka (AI) alanındaki ilerleme, bilgi, öğrenme, yaratıcılık ve üretkenlik için yeni ufuklar yaratma potansiyeli taşımaktadır. Bu ilerleme, iddialı araştırma hedeflerinin takip edilmesini ve insanlara ve topluma önemli faydalar sağlayabilecek yeteneklerin tahkik edilmesini gerektirir. Rekabetin sürdüğü bu alanda Google, Microsoft ve onun işbirlikçisi OpenAI ile stratejik bir konuma sahiptir. Google, Bard sohbet robotunu ve Duet AI'yi (Taulli, 2023) hızlı bir şekilde yeniden

markalaştırarak bunları yeni başlatılan Google Gemini çerçevesinde birleştirmiştir. Bu girişim yapay zekaya yaklaşımda önemli bir değişime işaret etmektedir. Metin tabanlı etkileşim modelleri yapay zekanın evriminde önemli bir rol oynamıştır, ancak insanlar daha karmaşık bilgilerin işlenmesini gerektiren, dinamik olarak gelişen bir ortamda hareket etmektedir. Dahası, insan iletişimi, konuşma ve görsel imgeler gibi karmaşık yöntemleri kapsayan salt metin alışverişinin ötesine geçer. Google Gemini, insan bilişine benzer şekilde, dünyaya ilişkin daha kapsamlı ve incelikli bir anlayışa ulaşmayı hedefleyerek bu boşluğu doldurmaya yönelik bir çabayı temsil etmektedir (Islam & Ahmed, 2024).

Dijital etkileşim araçlarının evrimi, Siri, Alexa ve Google Assistant gibi sesle etkinleşen dijital asistanları ortaya çıkardı ve ardından çevrimiçi sohbet robotları, ChatGPT ve Google Bard'ın (Aydın, 2023) gelişimi geldi. Artık Google'ın 2024 Gemini lansmanı ile gelişmiş bir teknoloji tanıtıldı (Anil R. vd., 2023). Gemini, Google'ın yeni nesil geniş dil modelidir. Gemini hem konuşan dijital asistanların hem de sohbet robotlarının özelliklerini birleştirmiştir. Gemini, ses ve metin girişlerini yönetme konusunda uzmandır ve bu da onun çok çeşitli görevleri gerçekleştirmesine olanak tanır. Gemini'nin tasarımı, kişisel bir öğretmen olarak işlev görme, programcılara kodlama çalışmalarında yardımcı olma ve iş arayanları görüşmelere hazırlama, çok yönlülüğünü ve yeteneklerinin geniş kapsamını gösterme gibi çeşitli ihtiyaçları karşılamayı amaçlamaktadır. Multimodal model, birden fazla kaynak türünden gelen bilgileri işleyebilen ve ilişkilendirebilen bir modeldir (Baltrusaitis vd., 2019). Gemini, Google DeepMind (Beattie vd., 2016) tarafından geliştirilen ve çok modlu kullanım durumları için tasarlanmış bir üretken yapay zeka modelleri ailesidir. Gemini modelleri görüntüleri anlama ve oluşturma, videoları anlama ve matematik problemlerini çözme becerisine sahiptir.

Başlangıçta, çok çeşitli kaynaklı modeller, her kaynak için ayrı bileşenlerin eğitilmesi ve daha sonra bunların entegre edilmesi yoluyla geliştirildi; bu, görüntü tanımlama gibi görevlerde iyi çalıştı ancak karmaşık akıl yürütme problemlerinde o kadar verim sağlamadı. Gemini, doğası gereği çok çeşitli kaynaklı olup çeşitli yöntemler üzerinde ön eğitim verecek şekilde tasarlandı. Metin, programlama kodu, görseller, ses ve video üzerine eş zamanlı eğitim, Gemini'nin diğer üretken yapay zeka modellerine kıyasla multimedya girdileriyle daha verimli şekilde başa çıkmasını sağladı (Shankland

Stephen, 2023). Daha sonra, Gemini'nin çok çeşitli girdileri zahmetsizce kavrayabilmesini ve bunlar hakkında akıl yürütebilmesini sağlayacak şekilde verimliliği artırmak için çok çeşitli kaynaklı verilerle ince ayar (fine-tune) yapılmıştır (Islam & Ahmed, 2024).

Birçok üretken yapay zeka modeline benzer şekilde Gemini modelleri de yalnızca kod çözücü dönüştürücü üzerine kuruludur. Ancak, yalnızca standart kod çözücü mimarisi, verimliliği artırmak, eğitimi geniş ölçekte istikrara kavuşturmak ve Google'ın TPU'larındaki çıkarımı optimize etmek için değiştirilmiştir (Anil R. vd., 2023). Dikkat fonksiyonlarının anahtar ve değer vektörlerini paylaşmasına izin vererek çok yönlü dikkatin verimliliğini artıran bir yöntem olan çoklu sorgu dikkati kullanmaktadırlar.

Gemini modelleri hem çok modlu hem de çok dilli bir veri kümesi üzerinde eğitilmiştir. Eğitim öncesi veri kümesi web belgelerinden, kitaplardan ve kodlardan alınan verileri kullanılmış ve görüntü, ses ve video verilerini içermektedir. Gemini modellerinin eğitiminde, Cümle Parçası (SentencePiece) simgeleştirici (tokenizer) kullanıldı ve belirteci tüm eğitim derleminin büyük bir örneği üzerinde eğitmenin, çıkarılan kelime dağarcığını iyileştirdiğini ve ardından model performansını iyileştirdiği bulundu (Mesnard T. vd., 2024).

Gemini modelleri, ilgili boyutlarına ve yapılandırmalarına göre TPuv4 ve TPuv5 kullanılarak eğitilmiştir. Özel olarak tasarlanmış bu yapay zeka hızlandırıcıları, Google'ın yapay zeka odaklı ürünlerinin temelini oluşturur ve yapay zeka modellerinin uygun maliyetli eğitimini destekler (Jouppi vd., 2023).

Temel olarak, Dış Bilgiyi Alma ile Birleştirme (Retrieval Augmented Generation), bilgi optimizasyonu için tasarlanmış bir yapay zeka çalışma alanıdır ve daha ilgili harici bilgileri besleyerek modele yönelik alakasız bilgi miktarını en aza indirir. Sınırlı bağlam penceresi nedeniyle Gemini, bilgi alımı için Retrieval Augmented Generation (RAG)'ı metin üretimiyle entegre ederek gerçeklere dayalı çıktılar sağlamaktadır (Ahmed & Islam, 2024).

4.2. Dil Modellerinin Performansını Ölçmede Kullanılan Metrikler

Günümüz yapay zeka sistemlerinin temelini oluşturan makine öğrenimi modelleri ve algoritmalarının, iç dinamiklerindeki kapsam ve karmaşıklık sebebiyle ürettikleri

çıktılar için karar verme süreçlerinin giderek daha şeffaf hale gelmesi performanslarının ölçülmesini zor hale getirmiştir. Bu şeffaflık sadece son kullanıcıyı değil, aynı zamanda bu sistemlerin yaratıcılarını ve analistlerini de ilgilendiren bir sorundur. Şeffaf bir sistemin temel özelliklerinden biri, çıkarımlarını açıklayabilmesi ve eylemlerinin nedenlerini doğal dilde ifade edebilmesidir. Dolayısıyla doğal dil açıklamalarının kalitesini ve açıkladıkları yapay zeka algoritmalarını değerlendirebilmek her zamankinden daha önemli hale gelmiştir (Clinciu vd., 2021). Ayrıca açıklamaların değerlendirilmesinin önemi sosyal bilişsel bilimler alanındaki araştırmacılar tarafından vurgulanmıştır (Doshi-Velez & Kim, 2017).

Makine öğrenmesi modelleri tarafından üretilen metinlerin bugüne kadar çoğunlukla insan yargıları toplanarak değerlendirilmesi hem zaman alıcı hem de maliyetli olmuştur.

Metin üretiminin doğal dil işleme modellerinde derin öğrenme mimarisinin kullanılmasıyla son zamanlarda giderek artan bir ilgi görmesi, bu göreve yönelik otomatik değerlendirme ölçütleri üzerine bir çalışmanın yapılması gerektiğini doğurmuştur (F. Liu & Liu, 2008).

Bu bölümde, metin üretmeyi doğal dil üretmenin özel bir durumu olarak görerek doğal dil üretme için mevcut otomatik değerlendirme yöntemlerini açıkladık. Ayrıca, doğal dil işlemedeki genel, alandan bağımsız değerlendirme ölçümlerinin, doğal dil açıklamalarının doğasında bulunan özellikleri yakalayacak kadar hassas olup olmadığını inceledik.

Makine tarafından oluşturulan metnin otomatik olarak değerlendirilmesi, doğal dil oluşturma alanında, dil üretmenin kendisi kadar zor olan temel bir araştırma sorunu olmuştur. Büyük ölçekli ön eğitimin olağanüstü başarısından cesaret alan, yakın zamanda yapılan bir dizi çalışma, değerlendirme ölçümlerini önceden eğitilmiş dil modellerine dayandırmayı önermiştir (Pillutla vd., 2021).

Makine çevirisi ve altyazı oluşturmada doğal dil oluşturma otomatik değerlendirmesi, aday cümleleri açıklamalı referanslarla karşılaştırmayı gerektirir. Yani doğal dilde metin üretimi genellikle açıklamalı referans cümleleri kullanılarak değerlendirilir. Amaç anlamsal eşdeğerliği değerlendirmektir. Ancak yaygın olarak

kullanılan yöntemler yalnızca yüzey-form yani metindeki kelimelerin ne anlam ifade ettiğine bakılmaksızın metinlerdeki belirli kurallara göre sıralanış benzerliğine dayanır (T. Zhang vd., 2019a).

Geleneksel olarak metin üretmenin değerlendirilmesi, tutarlılık, kısalık, dilbilgisellik, okunabilirlik ve içerik gibi farklı kalite ölçütlerine ilişkin insan yargılarını içerir. Oluşturulan metinlerin otomatik olarak nasıl değerlendirileceği, uzun yıllardan beri metin oluşturma araştırmaları yapan araştırmacılar arasında büyük ilgi görmüştür. Örneğin Saggion ve ark., oluşturulan metinler arasındaki benzerliği ölçen üç içerik temelli değerlendirme yöntemi önermiştir. Bu yöntemler şunlardır: kosinüs benzerliği, birim örtüşmesi (yani tek gram veya iki gram) ve en uzun ortak alt dizi (Saggion vd., 2002). Ancak bu otomatik değerlendirme yöntemlerinin sonuçlarının insan yargılarıyla nasıl ilişkili olduğu gösterilememiştir.

Geleneksel doğal dil oluşturma ölçümleri hızlı olsa da pek güvenilir değiller. Bunun tersine, önceden eğitilmiş büyük dil modellerine dayanan yeni ölçümler çok daha güvenilirdir ancak bu modeller önemli bilgisayar kaynakları gerektirir. Uygulamada benimsenmesi için değerlendirme metriklerinin hem güvenilir hem de uygun maliyetli olması, yani hızlı ve hesaplanması kolay olması gerekir (Kamal Eddine vd., 2022).

Bilgi erişiminde kesinlik ve duyarlılık, regresyonda hatanın ortalama karekökü vb. gibi bazı metrikler bu kriterleri karşılarken, Doğal Dil Üretimi alanında uygun metrikleri bulmak hala açık bir sorundur (Novikova vd., 2017). Gerçekten de, özetleme için ROUGE (C.-Y. Lin, 2004b), çeviri için BLEU (Papineni vd., 2002) ve METEOR (Banerjee & Lavie, 2005) gibi tarihsel n-gram eşleştirme ölçümleri yeterli gibi görünse de, yalnızca yüzeysel form eşleşmesine, yani sözcüksel benzerliğe dayandıkları ve bu nedenle anlamsal bir anlam taşımadıkları için çok güvenilir değildirler. Örneğin, özetleme sistemlerinin (norm haline gelen) değerlendirilmesi için veya oluşturulan metnin orijinal metni yeniden ifade ettiği durumlarda ROUGE’u kullanmak pek mantıklı değildir (Kamal Eddine vd., 2022).

Doğal dil işlemede transfer öğreniminin ortaya çıkmasının ardından, BERTSCORE (T. Zhang vd., 2019b) gibi önceden eğitilmiş büyük dil modellerine dayalı yeni doğal dil oluşturma ölçümleri yakın zamanda önerilmiştir. Bağlamsal gömülü yerleştirmelere dayanan bu ölçüm anlambilimi yakalar ve bu nedenle çok daha

güvenilirdir. Ancak temel modellerin büyüklüğü nedeniyle bu ölçümlerde kaynaklı sorunları ortaya çıkmaktadır. Bunların hesaplanması zaman almakta ve önemli hesaplama kaynaklarına erişim gerektirmektedir, dolayısıyla doğal dil işleme ile uğraşan herkes bunlara erişememektedir (Strubell vd., 2019).

Metinler oluşturulduktan sonra modelin performansını değerlendirmek için sonuçların değerlendirilmesi gerekir. Bu kullanılan modelin ne kadar isabetli metinler ürettiğini anlamak için kullanılan bir yoldur. Bu tez çalışmasında üretilen metinlerin değerlendirilmesi için 5 tane ölçüm yoluna başvurulmuştur. Bunlar BLEU, METEOR, ROUGE-L, ROUGE-N ve BERTSCORE'dur. Bu yöntemleri açıklamadan önce n-gram'lar ve makine öğrenmesinde sınıflandırmanın başarısını ölçmek için kullanılan metriklerden bahsedilecektir.

4.2.1. N-gram

N-gram, doğal dil işlemede sıralı bulunan ifadeleri n'er sayıda gruplayarak ölçen bir kavramdır ve n kelimenin sürekli dizisini ifade eder. Bu genellikle dil modellerinin, yani sözcük dizilerine olasılıklar atayan modellerin oluşturulmasıyla ilgilidir. Çoğu durumda, bir dil modelinin uygulanmasındaki ilk adım metin külliyyatının işlenmesidir. Metin verileri tipik olarak büyüktür, bu da uygulanan yöntemleri kullanarak daha kesin dil modellerinin oluşturulmasına olanak sağlar. N-gram, metin derlemelerini işlemek için kullanılan yöntemlerden biridir (Simunec vd., 2021).

N-gram tabanlı dil modeli, kelime dizisi olasılığına ilişkin bilgi sağlar. N-gram, önceki N-1 kelimelerinin görünümüne dayalı olarak bir kelimenin metnin bütünündeki görev olarak yerini tahmin edebilir (Jurafsky Daniel & Martin James H., 2006). Bu tahmin yeteneği, n-gramı konuşma ve dil işlemede çok önemli bir kavram haline getirmektedir. n-gram tabanlı dil modelleri genellikle kelimelerin belirsiz girişlerden (input) tanımlanması gereken görevlerde kullanılır. Ayrıca, n-gram tabanlı dil modelleri, artırıcı iletişimde (augmentative communication), konuşma sistemlerinde, metin özetlemede (Jurafsky, 2016), cümle tahmininde (Bickel vd., 2005) ve doğal dil işlemenin diğer birçok alanında kullanılmaktadır. n-gram temelli yaklaşım farklı varyasyonlarda incelenmiştir. N-gram yoluyla kelime dizileri, sözdizimsel ağahtaki yollar gibi farklı ilişkiler gözlemlenerek kaynak derlemden çıkarılabilir (Sidorov vd., 2014). Ayrıca temel

istatistiklere bağılı olarak kelimeler arasındaki sözdizimsel ve anlamsal ilişkiler de gözlemlenebilmektedir (P. F. Brown vd., 1992).

Tek bir kelimenin gelecekteki olasılığının bi-gram varsayımı kullanılarak tüm dizinin olasılığı hesaplanabilir:

$$P(kelime_1^n) = \prod_{k=1}^n P(kelime_k | kelime_{k-1}) \quad (4.15)$$

4.2.2. Accuracy (Doğruluk), Precision (Kesinlik), Recall (Duyarlılık) ve F1 Score metrikleri

Denetimli makine öğrenimi sınıflandırıcıları genellikle hatayı en aza indirecek şekilde eğitilir. Bu hata, seçimi onlarca yıldır sürekli üstünde tartışılan bir veya daha fazla ölçüm kullanılarak değerlendirilir (Dinga vd., 2019). Bir metrik seçerken, yorumlanabilirlik, hesaplama maliyeti, farklılaştırılabilirlik ve belirli bir alandaki popülerlik dahil ancak bunlarla sınırlı olmamak üzere birçok kriterin dikkate alınması gerekir. Örnek olarak, model geliştirmenin tipik bir iş akışı, ağırlık optimizasyonu için eğitim sırasında çapraz entropi (Shore & Johnson, 1981) veya dayanak noktası kaybı (hinge loss)(Bartlett vd., 2008) gibi bir kayıp fonksiyonu kullanmak, ardından modeli test ederken doğruluk, kesinlik veya duyarlılık gibi kolayca yorumlanabilen bir ölçüm kullanmaktır.

Denetimli öğrenmede, etiketlerin basmakalıp olduğu ve bir sınıflandırıcının tahminlerinin dört hücreli bir durum tablosunda özetlendiği ikili sınıflandırma problemi bağlamında çeşitli ölçümlerin tanıtılması yaygındır. Bu beklenmedik durum tablosu, tahmin edilen her etiketin her bir gerçek sınıfla ilişkilendirilme sayısının ham sayımları kullanılarak veya göreceli terimlerle ifade edilebilir. Sınıflandırıcının tahmin ettiği değerler “Positive” ve “Negative” olarak değerlendirilirken gerçek değerler “True” ve “False” olarak değerlendirilir (Powers & Ailab, 2020). Bununla ilgili gösterim Şekil 4.14’te verilmiştir.

	POSITIVE	NEGATIVE
TRUE	TP	TN
FALSE	FP	FN

Şekil 4.14. Karmaşıklık matrisi gösterimi

Doğruluk sınıflandırma yapan bir modelin ne kadar başarılı olduğunu görmek için tek başına yeterli olmayan bir metriktir. Veri seti her iki sınıftan da mümkün olduğu kadar çok örnek içeriyorsa (yani dengeli bir veri seti ise) ve doğruluk çok yüksekse bu kabul edilebilir. Fakat, doğruluk, veri setinde üretilen hata türleri arasında ayrım yapamayabilir (Yanlış Pozitif ve Yanlış Negatifler). Bu sebeple modelin başarısı ölçülürken Kesinlik, Duyarlılık ve F1 Score metriklerine başvurulur. Kesinlik tahmin edilen pozitif değerlerin gerçekten kaç tanesinin pozitif olduğunu göstermek için kullanılır. Duyarlılık ise pozitif olarak tahmin ettiğimiz değerlerin ne kadarının pozitif olarak tahmin edilebileceğini gösterir. F1 Score ise duyarlılık ve kesinlik değerlerinin harmonik ortalamasını verir. F1 Score kullanmamızın sebebi eşit dağılıma sahip olmayan veri kümelerinde model seçimini hatalı yapmamaktır.

$$Accuracy = \frac{TP+TN}{TP+FP+TN+FN} \quad (4.16)$$

$$Precision = \frac{TP}{TP+FP} \quad (4.17)$$

$$Recall = \frac{TP}{TP+FN} \quad (4.18)$$

$$F1\ Score = 2 * \frac{precision*recall}{precision+recall} \quad (4.19)$$

4.2.3. BLEU (Bilingual Evaluation Understudy)

Tipik olarak, belirli bir kaynak cümle için pek çok "mükemmel" çevirisi vardır. Bu çeviriler aynı kelimeleri kullansalar bile kelime seçimi veya kelime sırası farklılık gösterebilir. Ancak yine de insanlar iyi bir çeviriyi kötü olandan açıkça ayırt edebilirler.

BLEU da bunu baz alarak çalışır. BLEU herhangi bir sisteme uygulandığında birincil görevi, aday metnin n-gramlarını referans çevirisinin n-gramlarıyla karşılaştırmak ve eşleşme sayısını saymaktır. Bu eşleşmeler pozisyonundan bağımsızdır. Ne kadar çok eşleşme olursa adayın çevirisi o kadar iyi demektir (Papineni vd., 2002).

BLEU başta makine çevirisi ve metin üreten Doğal Dil İşleme sistemlerini değerlendirmek için yaygın olarak kullanılan bir metriktir. BLEU ve diğer otomatik değerlendirme metriklerinin çalışma şekli, makine çeviri sisteminin çıktısını referans insan çevirileriyle karşılaştırmaktır. Doğal dil işlemede kullanılan değerlendirme ölçümleri, konuşma tanımada kullanılan kelime hata oranı ölçümü gibi referans kullanan diğer ölçümlerden farklıdır. Çünkü çeviriler, kelime seçimi ve bazı ifadelerin farklı sıralaması açısından bir dereceye kadar çeşitlilik gösterir. BLEU, çoklu referans çevirileri kullanarak kelime seçiminde izin verilen varyasyonu yakalamaya çalışır (Thompson, 1991). BLEU, cümle sıralamasında değişiklik sorununun üstesinden gelmek için değiştirilmiş n-gram kesinliğini kullanır. BLEU'nun n-gram kesinliği, cümlelerde meydana gelen tekrarları ortadan kaldıracak şekilde oluşturulmuştur. Her bir n-gram uzunluğu için değiştirilmiş kesinlik puanı (p_n), C referans cümlesinin tamamındaki her S aday metin cümlesi için eşleşmelerin toplanmasıyla hesaplanır:

$$p_n = \frac{\sum_{S \in C} \sum_{n\text{-gram} \in S} \text{Eşleşen}_{n\text{-gram}}}{\sum_{S \in C} \sum_{n\text{-gram} \in S} \text{Toplam}_{n\text{-gram}}} \quad (4.20)$$

BLEU'nun kesinlik esaslı olması ve birden fazla referans çeviri üzerinden duyarlılığın formüle edilmesinin zor olması nedeniyle, çok kısa olan yüksek kesinlikli varsayım çevirileri önerme olasılığını gidermek için bir kısalık kuralı (brevity penalty) uygulanır:

$$BP = \begin{cases} 1 & , \text{ if } c > r \\ e^{1-r/c} & , \text{ if } c \leq r \end{cases} \quad (4.21)$$

Burada c üretilen derlemin uzunluğudur, r ise etkin referans derlem uzunluğudur.

Böylece BLEU puanı şu şekilde hesaplanır:

$$BLEU = BP * \exp(\sum_{n=1}^N w_n * \log p_n) \quad (4.22)$$

Bir BLEU puanı 0 ila 1 arasında değişebilir; burada daha yüksek puanlar, referans olarak üretilen metne daha yakın eşleşmeleri gösterir ve burada 1 puanı, referans olarak üretilen biriyle tam olarak eşleşen bir aday metin çevirisine atanır.

4.2.4. METEOR (Metric for Evaluation of Translation with Explicit ORdering)

İlk olarak 2004 yılında geliştirilen ve kullanılmaya başlanılan makine çevirisi için kullanılan METEOR, üretilen metinlerin insanların çeviri kalitesine ne kadar benzediğine ilişkin çıkarımları kullanarak puanlamak hedefiyle tasarlandı. METEOR, insan kararlarıyla yüksek korelasyon açısından kritik olduğu çeşitli ölçümlerle doğrulanan bir özellik olan kesinliğin yanı sıra hatırlamayı da kullanmakta ve vurgulamaktadır. METEOR aynı zamanda esnek kelime eşlemesi kullanarak referans çeviri değişkenliği sorununu da çözer ve morfolojik değişkenlerin ve eş anlamlıların meşru karşılıklar olarak dikkate alınmasına olanak tanır. METEOR, değerlendirilmekte olan bir makine çevirisi (aday metin) ile aynı kaynak cümle için referans çevirileri arasındaki sözcük benzerliğinin ölçümlerine dayanmaktadır. Bu benzerliği ölçmek için METEOR, her bir makine çevirisi aday metin ile bir veya daha fazla referans çevirisi arasında açık bir kelimedenden kelimeye eşleşme sağlar. METEOR'un en önemli özelliklerinden biri çeviri değişkenliğini ele almasıdır. Aynı anlam, farklı sözcük seçimleri kullanılarak yansıtılabildiğinden, METEOR'un kullandığı sözcükten kelimeye eşleştirici yalnızca tam sözcükleri değil aynı zamanda morfolojik değişkenleri ve eş anlamlıların da eşleştirebilir (Lavie & Denkowski, 2009).

METEOR'un çalışmasındaki öne çıkan gözlemlerden biri, bir metrik bileşen olarak duyarlılığın önemidir (Lavie vd., 2004). METEOR'daki temel yenilik, çeşitli insan yargıları ve çeşitli diller ile korelasyonu optimize etmek amacıyla metrik içindeki serbest parametreleri ayarlama yeteneğidir (Lavie & Agarwal, 2007).

METEOR'un geliştirilmesine yol açan bazı ana ilkeler, BLEU metriğindeki potansiyel zayıflıkların bir dizi gözleminden ortaya çıkmıştır.

METEOR, bir aday metin ile belirli bir referans çevirisi arasındaki açık bir kelimedenden kelimeye eşleşmeye dayalı bir puan hesaplayarak bu çeviri aday metni değerlendirir. Birden fazla referans sağlanırsa, aday metin her birine göre bağımsız olarak puanlanır ve en iyi puanlama çifti kullanılır (Lavie & Agarwal, 2007).

Bir aday metin ile referans çevirisi arasında eşleme oluştuğunda METEOR puanı aşağıdaki gibi üretilir. Tüm modül aşamalarında iki dize arasında bulunan eşlenmiş unigramların toplam sayısına (m), aday metindeki toplam unigram sayısına (t) ve referanstaki toplam unigram sayısına (r) dayanarak, unigramı hesaplarız:

$$\text{Kesinlik: } P = m / t \quad (4.23)$$

$$\text{Duyarlılık: } R = m / r \quad (4.24)$$

Daha sonra P ve R 'nin parametrelendirilmiş harmonik ortalamasını hesaplarız (Van Rijsbergen, 1979).

$$F_{mean} = \frac{P * R}{\alpha * P + (1 - \alpha) * R} \quad (4.25)$$

Kesinlik, duyarlılık ve F_{mean} değerlerinin tamamı tek kelimelik eşleşmelere dayanmaktadır. Kelime sırasının korunmasını hesaba katmak için parçalanma cezası hesaplanır. İlk olarak, iki dizi arasındaki eşleşen unigramların dizisi, her bir öbekteki eşleşen unigramlar bitişik (her iki dizide) ve aynı sırada olacak şekilde en küçük sayıdaki "parçalara" bölünür. Parça sayıları (ch) ve eşleşmeler (m) daha sonra parçalanma fraksiyonunu hesaplamak için kullanılır: $frag = ch / m$. Daha sonra ceza şu şekilde hesaplanır:

$$Pen = \gamma * frag^{\beta} \quad (4.26)$$

γ değeri maksimum kaybı belirler ($0 \leq \gamma \leq 1$). β değeri parçalanma ile kayıp arasındaki fonksiyonel ilişkiyi belirler.

Son olarak, iki dizi arasındaki eşleşme puanı METEOR için şu şekilde hesaplanır:

$$score = (1 - Pen) * F_{mean} \quad (4.27)$$

Metrikteki parametreler (α , β ve γ) yukarıda açıklandığı gibi insan kararlarıyla maksimum korelasyonu elde edecek şekilde ayarlanmıştır (Lavie & Agarwal, 2007).

4.2.5. ROUGE (Recall-Oriented Understudy for Gisting Evaluation)

ROUGE, herhangi bir model tarafından oluşturulan metnin kalitesini, insanlar tarafından oluşturulan diğer (ideal) metinlerle karşılaştırarak otomatik olarak belirler. Ölçütler, değerlendirilecek bilgisayar tarafından oluşturulan metin ile insanlar tarafından

oluşturulan ideal metinler arasındaki n-gram, kelime dizileri ve kelime çiftleri gibi örtüşen birimlerin sayısını sayar (C.-Y. Lin, 2004a).

4.2.5.1. ROUGE-N

ROUGE-N, bir aday metin ile bir dizi referans metin arasındaki n gramlık bir duyarlılık kullanılmasıdır. ROUGE-N aşağıdaki şekilde hesaplanır:

$$ROUGE - N = \frac{\sum_{S \in \{Referans\ Metin\}} \sum_{gram_n \in S} Eşleşen\ n\ gram\ sayısı}{\sum_{S \in \{Referans\ Metin\}} \sum_{gram_n} n\ gram\ sayısı} \quad (4.28)$$

n, n-gramın uzunluğunu temsil ederken, n gram sayısı ve eşleşen n gram sayısı (gram_n), aday metinde ve bir dizi referans metinde birlikte ortaya çıkan maksimum n-gram sayısıdır.

ROUGE-N'nin duyarlılıkla ilgili bir ölçüm olduğu açıktır çünkü denklemin paydası, referans metin tarafında meydana gelen n-gram sayısının toplamıdır.

Daha fazla referans metin ekledikçe ROUGE-N formülünün paydasındaki n-gram sayısının arttığına dikkat edilmelidir. Bu sezgisel ve mantıklıdır çünkü birden fazla iyi metin örneği mevcut olabilir. Havuza her referans eklediğimizde alternatif metinlerin eşleşme ve ölçme alanını genişletmiş oluruz. Referans havuzuna ne tür referanslar eklediğimizi kontrol ederek özetlemenin farklı yönlerine odaklanan değerlendirmeler tasarlayabiliriz. Ayrıca payın tüm referans özetleri üzerinden toplandığı unutulmamalıdır. Bu, birden fazla referansta meydana gelen eşleşen n-gramlara etkili bir şekilde daha fazla ağırlık verir. Bu nedenle, daha fazla referansın paylaştığı kelimeleri içeren bir aday metin, ROUGE-N ölçümü tarafından tercih edilir. Bu yine çok sezgisel ve mantıklıdır çünkü normalde referans özetleri arasındaki fikir birliğine daha çok benzeyen bir aday özetini tercih edilir (C.-Y. Lin, 2004c).

4.2.5.2. ROUGE-L

ROUGE-L'yi hesaplamanın prensibi, en uzun ortak alt dizileri dikkate alarak aday ve referans yanıtları arasındaki kesinliği ve duyarlılığı incelemektir (Yang vd., 2018). ROUGE-L hem oluşturulan hem de referans metinlerde görünen en uzun kelime dizisini tanımlar. ROUGE-L, özetlerdeki bilgilerin yapısal benzerliğini ve sıralamasını

değerlendirerek genel tutarlılık ve akışa dair içgörüler sağlar. ROUGE-L'nin formülü aşağıdaki gibidir:

$$ROUGE - L = \frac{En\ Uzun\ Ortak\ Alt\ Dizin\ Uzunluđu}{Referans\ Metinlerdeki\ Kelime\ Sayısı} \quad (4.29)$$

En uzun ortak alt dizinin uzunluğu, oluşturulan metin ile referans metinler arasındaki en uzun ortak alt dizinin kelime sayısını belirtir. Referans metinlerdeki kelime sayısı, tüm referans metinlerdeki kümülatif kelime sayısını gösterir. Rouge-L formülü, sistem tarafından oluşturulan metin ile referans metinler arasındaki benzerliği, kelimelerin en uzun ortak dizisine göre değerlendirir. Bireysel n-gramlara odaklanan diğer Rouge metriklerinden farklı olarak Rouge-L, referans metinlerin tamamını bir bütün olarak ele alır. Rouge-L, en uzun paylaşılan alt diziyi inceleyerek, belirli n-gram uzunluklarına bakılmaksızın, oluşturulan özet ile referanslar arasındaki genel içerik örtüşmesine ve yapısal benzerliğe ilişkin bilgiler sağlar. Rouge-L puanının Rouge-N gibi diğer Rouge ölçümleriyle birleştirilmesi, oluşturulan metinlerdeki içerik örtüşmesi, yapısal tutarlılık ve bağlamsal ilişkilerin farklı yönlerini dikkate alarak oluşturulan metin kalitesinin daha kapsamlı bir değerlendirmesine olanak sağlayabilir (Rosamma & Patil, 2023).

4.2.6. BERTSCORE

BertScore, metin oluşturmaya yönelik otomatik bir değerlendirme ölçütüdür. Yaygın ölçümlere benzer şekilde BERTSCORE, aday cümledeki her simge (token) için referans cümledeki her simge ile bir benzerlik puanı hesaplar (Zhang vd., 2019b). Makine çevirisi ve altyazı oluşturmada doğal dil oluşturma otomatik değerlendirmesi, aday cümleleri açıklamalı referanslarla karşılaştırmayı gerektirir. Amaç anlamsal eşdeğerliği değerlendirmektir. Ancak yaygın olarak kullanılan yöntemler yalnızca yüzey form benzerliğine dayanır (Papineni vd., 2002). BertScore, iki cümle benzerliğini, simgelerin yerleştirmeleri arasındaki kosinüs benzerliklerinin toplamı olarak hesaplar. BertScore, n-gram tabanlı ölçümlerdeki iki yaygın dezavantajı ele alarak bunlara çözüm sunmuştur (Banerjee & Lavie, 2005).

Referans cümlesi $x = \langle x_1, . . . , x_k \rangle$ ve bir aday cümle $\hat{x} = \langle \hat{x}_1, . . . , \hat{x}_l \rangle$ verildiğinde, simgeleri temsil etmek için bağlamsal yerleştirmeler kullanılır ve isteğe

bağlı olarak bir kelimenin ne sıklıkta kullanıldığını gösteren ters metin sıklığı (Inverse Document Frequency) puanlarıyla ağırlıklandırılan kosinüs benzerliğini kullanarak eşleştirme hesaplanır. Ters metin sıklığı, bir kelimenin belgelerde ne kadar sık kullanılırsa puanı o kadar düşük olmasıdır. Puan ne kadar düşük olursa, kelimenin önemi de o kadar az olur (Hakim vd., 2014).

Masked Language Modelling: BERT ve türevlerinde Maskeli Dil Modellemesi kullanılmaktadır. Maskelenmiş kelimenin bağlamı kullanılarak kelime kimliğinin tahmin edilmesine dayanmaktadır. Bağlamsal yerleştirme oluşturmak için hem sol hem de sağ bağlamı kullanır. Adlandırılmış Varlık Tanıma, duygu analizi gibi çift yönlü bağlamın önemli olduğu cümle düzeyindeki görevler için uygundur (Q. Liu vd., 2020). Simge maskelemede, giriş simgeleri rastgele maskelenir. Daha sonra maskesiz simgeler yardımıyla maskelenmiş simgeler tahmin edilir (Sonmezoz & Amasyali, 2021).

x ve $\hat{x}$ giriş cümlelerindeki simgeleri temsil etmek için bağlamsal yerleştirmeler kullanılır. Önceki kelime yerleştirmelerin aksine, BERT ve ELMO (Peters vd., 2018) gibi bağlamsal yerleştirmeler, hedef kelimenin bağlamını oluşturan çevredeki kelimelere bağlı olarak aynı kelime için farklı cümlelerde farklı vektör temsilleri oluşturabilir. Bu yerleştirmeleri oluşturmak için kullanılan modeller yaygın olarak maskelenmiş kelime tahmini (masked word prediction) gibi çeşitli dil modelleme hedefleri kullanılarak eğitilir (Devlin vd., 2018).

Belirlenmiş bir referans cümlesi verildiğinde $x = \langle x_1, \dots, x_k \rangle$, yerleştirme modeli $\langle x_1, \dots, x_k \rangle$ vektörlerinin bir dizisi üretilir. Kullanılan ana model, giriş metnini, bilinmeyen kelimelerin yaygın olarak gözlemlenen birkaç karakter dizisine bölündüğü bir sözcük parçaları dizisi (Wu vd., 2016) halinde simgeleştiren BERT'tir. Her bir kelime parçasının temsili, dönüşümlü olarak öz-dikkat ve doğrusal olmayan dönüşümlerin tekrar tekrar uygulanmasıyla bir dönüştürücü kodlayıcı ile hesaplanır (Vaswani vd., 2017).

Vektör gösterimi, kesin dizi (Papineni vd., 2002) veya buluşsal (Lavie & Agarwal, 2007) eşleşme yerine daha randımanlı bir benzerlik ölçümüne olanak tanır. Referans simge olan x_i ile aday metnin belirteci $\hat{x}_j$ 'nin kosinüs benzerliği şöyledir:

$$\frac{x_i^T \hat{x}_j}{\|x_i\| \|\hat{x}_j\|} \quad (4.30)$$

Bu hesaplamada iç çarpıma indirgeyen önceden normalleştirilmiş vektörler olan $x_i^T \hat{x}_j$ kullanılır. Bu önlem simgeleri ayrı ayrı ele alırken, bağlamsal yerleştirmeler cümlelerin geri kalanından bilgi içerir.

Tam puan, duyarlılığı hesaplamak için x 'teki her belirteci $\hat{x}$ 'teki bir simge ile ve kesinliği hesaplamak için x 'teki her belirteci $\hat{x}$ 'teki bir simge ile eşleştirir. Her bir belirtecin diğer cümledeki en benzer simge ile eşleştirildiği, eşleşen benzerlik puanını maksimuma çıkarmak için metinsel olarak mümkün olduğunca fazla eşleşen eşleştirme kullanılır. F1 skoru hesaplamak için kesinlik ve duyarlılık kullanılır. Referans x ve aday $\hat{x}$ için duyarlılık, kesinlik ve F1 skoru ölçütleri şöyledir:

$$R_{BERT} = \frac{1}{|x|} \sum_{x_i \in x} \max_{\hat{x}_j \in \hat{x}} x_i^T \hat{x}_j \quad (4.31)$$

$$P_{BERT} = \frac{1}{|\hat{x}|} \sum_{\hat{x}_j \in \hat{x}} \max_{x_i \in x} x_i^T \hat{x}_j \quad (4.32)$$

$$F_{BERT} = 2 * \frac{P_{BERT} * R_{BERT}}{P_{BERT} + R_{BERT}} \quad (4.33)$$

BÖLÜM 5

DENEYSEL YÖNTEM VE SONUÇLAR

Tezin ilk bölümünü oluşturan dil modellemedeki amaç, bir sözcük dağarcığından alınan belirteçlerden oluşan değişken uzunluktaki metin dizileri üzerinde bir olasılık dağılımını modellemektir. Olasılık dağılımına benzeyen, yani örneklerin olasılık dağılımındaki örneklere benzer olduğu tüm metin dizileri için bir model bulunması istenmektedir. Olasılık dağılımı bir sinir ağı tarafından parametrelendirildiğinde buna sinir ağı dil modeli denilmektedir.

Burada sinir ağı dil modelinin çözmesi gereken problem, belirli bir kelime dizisini veya onun alt dizisinin devamını getirecek kelimelerin bu görevi yerine getirirken uyduğu olasılık kuralını çözmektir. Kural ortaya çıktıktan sonra diziyi tamamlayan kelimeler artık bir örneğin elemanlarıdır.

Tezde son zamanda kullanım yaygınlığı ve verimliliği sebebiyle sinirsel dil modellerine yer verilmiştir. Bu kapsamda GPT-2, Llama, Mistral, Phi-2, XGLM'nin metin tamamlama yeteneği değerlendirilmiştir. Tüm girdiler Vikipe'deki rastgele makalelerden alınmıştır.

Bu tezde, metin üretme işlevi kullanılarak belirli girdiler ile elde edilen çıktıların kullanıldığı modellerin ilgili işlev üzerinden öne çıkan metrikler yoluyla karşılaştırılması ele alınmıştır. Bu metrikler, Huggingface adlı siteden indirilen dönüştürücü tabanlı modellerin üretmiş olduğu metinler ile referans metinlerin karşılaştırmasında kullanıldı. Dönüştürücü tabanlı modeller, araştırmacılar tarafından genişletilebilir, uygulayıcılar için basit ve endüstriyel uygulamalarda hızlı ve sağlam olacak şekilde tasarlanmıştır. Bu modellere ait kütüphaneye <https://huggingface.co/> adresinden ulaşılabilir.

Büyük ölçekli önceden eğitilmiş dil modellerinin (Pretrained Language Models) her yerde bulunması, model merkezlerine ve önceden eğitilmiş modelleri kullanmak için özel platformlara olan ilgiyi artmaktadır.

Doğal dil işleme topluluğundaki bu ilgi ve ilerleme, yapay zeka araştırmalarının odağını büyük ölçekli önceden eğitilmiş modellere kaydırmıştır. Önceden eğitilmiş modeller ayrıca yazılım mühendisliği alanında çalışan araştırmacılar ve geliştiriciler tarafından da yaygın olarak kullanılmaktadır.

Modelin yeniden kullanımındaki bu artışın ardından ve kullanımını kolaylaştırmak için “Model Merkezleri” oluşturulmuştur. Model merkezleri, kullanıcılar tarafından kolayca yeniden kullanılabilir şekilde seçilmiş ve kategorize edilmiş, önceden eğitilmiş model ve veri kümelerinden oluşan koleksiyonları barındıran platformlardır. Huggingface Hub internetteki en büyük model merkezlerinden biridir.

5.1. Kullanılan Modeller

Tezde kullanılan her model için mevcut veriler, verileri bir JSON dosyası biçiminde alan Huggingface tarafından sağlanan Hub API uç noktalarının Python sarmalayıcı kitaplığı olan Huggingface Hub kullanılarak çıkarılmıştır. Bu işlemi yaparken kullanılan kod Şekil 5.1’de gösterilmiştir. Kodun çalışması için Python 3.11 ortamı ile bu ortamın kullanılması için PyCharm bütünleşik geliştirme ortamı kullanılmıştır.

```
from transformers import pipeline
model1 = "gorkemgoknar/gpt2-turkish-writer"
model2 = "Trendyol/Trendyol-LLM-7b-chat-v0.1"
model3 = "malhajar/Mistral-7B-Instruct-v0.2-turkish"
model4 = "facebook/xglm-7.5B"
model5 = "openai-community/gpt2"
model6 = "microsoft/phi-2"
model7 = "mistralai/Mistral-7B-v0.1"
pipe =
pipeline("text-generation", model=model1, max_length=512, temperature=0.3)
```

Şekil 5.1. Huggingface sitesinden modellerin indirilmesini sağlayan Python kodu

Bahsi geçen kodda Python programlama dilini *transformers* kütüphanesinden *pipeline* modülü çağırılıp kütüphanenin kullandığı dil modelleri metin üretme işlevi ile kullanılmak üzere yerel bilgisayara indirilmiştir. İlk 3 model Türkçe veri setleri ile, Model 4 çok dilli veri seti ile, son 3 model ise İngilizce veri setleriyle ince ayar yapılmış modellerdir. Tezde kullanılan modellerle ilgili özellikler Çizelge 5.1’de verilmiştir.

Çizelge 5.1. Tezde kullanılan modellerin genel özellikleri

Model No	Model Adı	Boyutu	Parametre Sayısı
1	gorkemgoknar/gpt2-turkish-writer	488 MB	1,5 milyar
2	Trendyol/Trendyol-LLM-7b-chat-v0.1 (based on Llama2 7B)	25,4 GB	7 milyar
3	malhajar/Mistral-7B-Instruct-v0.2-turkish	13,4 GB	7,3 milyar
4	facebook/xglm-7.5B (multilingual)	13,9 GB	7,5 milyar
5	openai-community/gpt2	525 MB	1,5 milyar
6	microsoft/phi-2	10,3 GB	2,7 milyar
7	mistralai/Mistral-7B-v0.1	13,4 GB	7,3 milyar

5.2. Girdi ve Referans Olarak Kullanılan Metinler

Modeller indirildikten sonra *pipe* fonksiyonu kullanılarak her modele 5 farklı metin girdisi verilerek maksimum 512 uzunlukta metinler üretmesi istenmiştir. Bu fonksiyona ait kod Şekil 5.2’de verilmiştir.

```
uretilen_metin = pipe(girdi, max_length=512)[0]["generated_text"]
```

Şekil 5.2. Metin üretmede kullanılan Python kod satırı

Verilen girdi metinler ve modellerin ürettikleri metinlerin başarısını hesaplamak için kullanılan metriklere verilen referans metinler Türkçe için Çizelge 5.2 ve İngilizce için Çizelge 5.3’te verilmiştir.

Çizelge 5.2. Türkçe modeller için verilen girdi metinler ve kullanılan referans metinler

Girdi Metin	Referans Metin
1963 Türkiye	1963 Türkiye yerel seçimleri, 17 Kasım 1963 tarihinde yapılmıştır. 1961 Anayasası'nın kabulünden sonra yapılan ilk yerel seçimdir. Yüksek Seçim Kurulu'nun (YSK) gözetimi ve denetimi altında gerçekleştirilen ilk yerel seçim olan bu seçimler 1960 öncesi döneme göre pek çok yenilik içermiştir.
İkinci Berberi Savaşı, 1815-1816	İkinci Berberi Savaşı, 1815-1816 yılları arasında Amerika Birleşik Devletleri ile Osmanlı İmparatorluğu'na bağlı Kuzey Afrika Berberi devletleri arasında gerçekleşen bir deniz savaşıdır. Savaş, Amerika Birleşik Devletleri Donanması'nın zaferiyle sonuçlandı.
"Disco Inferno", Amerikalı rapçi 50 Cent	"Disco Inferno", Amerikalı rapçi 50 Cent tarafından, ikinci stüdyo albümü The Massacre için kaydedilen bir şarkıdır. Yapımcılığını Dangerous LLC üstlenmiştir. Aralık 2004'te, albümün ilk teklisi olarak piyasaya sürülmüştür.
Goeben ve Breslau'nun takibi, I. Dünya	Goeben ve Breslau'nun takibi, I. Dünya Savaşı'nın başlarında Akdeniz'de gerçekleşen, Birleşik Krallık'ın Kraliyet Donanması'na bağlı gemilerin Almanya'nın İmparatorluk Donanması'na ait SMS Goeben ve SMS Breslau'yu önlemeye çalıştıkları deniz muharebesidir. Alman Akdeniz Tümeni'ni oluşturan Moltke sınıfı muharebe kruvazörü SMS Goeben ve Magdeburg sınıfı hafif kruvazör SMS Breslau, savaşın başlaması ile Akdeniz'de sıkışmıştı.
Karayılan, Gaziantep'in Fransız işgalinden kurtuluşunu konu alan dram	Karayılan, Gaziantep'in Fransız işgalinden kurtuluşunu konu alan dram ve tarihî türündeki Türk televizyon dizisidir. Türkiye'nin en yüksek bütçeli ikinci dizisi olma özelliğini taşıyan "Karayılan"ın çekimleri; İstanbul, Gaziantep, Kahramanmaraş ve Şanlıurfa'da yapılmıştır. Dizi, 22 Nisan 2008 tarihinde yayımlanan 18. bölümü ile final yaparak sona erdi.

Çizelge 5.3. İngilizce modeller için verilen girdi metinler ve kullanılan referans metinler

Girdi Metin	Referans Metin
Diplomatic capital	Diplomatic capital refers to the trust, goodwill, and influence which a diplomat, or a state represented by its diplomats, has within international diplomacy. According to political scientist Rebecca Adler-Nissen, diplomatic capital is a kind of currency that can be traded in diplomatic negotiations and that is increased when positive "social competences, reputation and personal authority" are portrayed.
Herbicide safeners are organic	Herbicide safeners are organic compounds used to enhance the effectiveness of herbicides, to make them "safer". They minimize the effect of the herbicide on crop plants, thereby improving selectivity between crop plants vs. weed species being targeted by the herbicide. One way that they function is by enhancing the expression of protective enzymes in the crop plant. These protective enzymes might include cytochrome P450.
The Ghanaian people are a nation	The Ghanaian people are a nation originating in the Ghanaian Gold Coast. Ghanaians predominantly inhabit the Republic of Ghana and are the predominant cultural group and residents of Ghana, numbering 34 million people as of 2024, making up 85% of the population. The word "Ghana" means "warrior king". An estimated diaspora population of 4 million people worldwide are of Ghanaian descent. The term ethnic Ghanaian may also be used in some contexts to refer to a group of related ethnic groups native to the Gold Coast.
Fort Kelburne was a coastal artillery battery located	Fort Kelburne was a coastal artillery battery located in Wellington, New Zealand. Built between 1885 and 1887 in the vicinity of Ngauranga, following fears of an impending war with Russia, the fort was used for training purposes, later receiving renovations during the First World War. Fort Kelburne, along with Fort Buckley in Kaiwharawhara, were the first coastal defences to be built in a ring that was designed to protect Wellington Harbour.
Arkeisha Antoinette Knight, known professionally as Kash Doll, is	Arkeisha Antoinette Knight, known professionally as Kash Doll, is an American rapper and actress from Detroit, Michigan. Signed to Republic Records, she is best known for her singles "For Everybody" (2017) and "Ice Me Out" (2018), the latter of which became the lead single from her debut album Stacked (2019). Kash Doll has collaborated with artists including Meek Mill, Big Sean, Iggy Azalea, DreamDoll and K. Michelle.

5.3. Modellerin Metin Üretme Süreleri

Çizelge 5.2 ve Çizelge 5.3'te verilen girdi metinleri kullanıldığında modellerin metin üretme süreleri Çizelge 5.4'te verilmiştir. Bu süreler Intel Core i7-9750H işlemcisi ve 32 GB RAM'i olan Windows 11 kullanan bir bilgisayar ile elde edilmiştir.

Çizelge 5.4. Modellerin 5 farklı girdi metni için toplam metin üretme süreleri

Model Adı	Metin üretme süresi (sn)
gorkemgoknar/gpt2-turkish-writer	17
Trendyol/Trendyol-LLM-7b-chat-v0.1 (based on Llama2 7B)	248
malhajar/Mistral-7B-Instruct-v0.2-turkish	731
facebook/xglm-7.5B (multilingual)	152
microsoft/phi-2	126
mistralai/Mistral-7B-v0.1	715
openai-community/gpt2	20

Çizelge 5.1 ve Çizelge 5.4 birlikte incelendiğinde, modelin metin üretme süresinin genel olarak modelin boyutu ve parametre sayısı ile ilişkili olduğu söylenebilir. Küçük boyutlu GPT-2 modelleri hızlı metin üretmektedir. Bununla birlikte Mistral modellerinin yakın boyutlardaki Llama2 ve XGLM modellerine göre çok daha yavaş metin ürettikleri gözlemlenmiştir.

5.4. Üretilen Metinlerin Başarısının Hesaplanması

Çizelge 5.2 ve Çizelge 5.3'te verilen girdi metinler modellere verilmiş ve modellerin ürettikleri metinlerin başarısını ölçmek için hem Bölüm 4.2'de bahsedilen hesaplama metrikleri, hem de anket yolu ile insanlara sorularak elde edilen puanlar kullanılmıştır. Hesaplama metriklerinin ihtiyaç duyduğu referans metinler için Çizelge 5.2 ve Çizelge 5.3'te verilen referans metinler kullanılmıştır. Bu referans metinler de anketi cevaplayanlara gösterilmiş (cevaplayan tarafından üretilen bir metin imiş gibi algılanacak şekilde) ve onlar için de puan vermeleri istenmiştir. İNSAN ismini verdiğimiz

metrik hesaplanırken anketi cevaplayanların metinlere 0-10 arası verdikleri skorlar 10’a bölünerek ortalama alınmıştır.

Puanlamada bire yakın değerler referans metinlere göre tutarlı ve anlamlı değerlendirilirken, sıfıra yakın değerler referans metinlere anlamca ve kelime benzerliği açısından uzaktır. BLEU, METEOR, ROUGE ve BERTSCORE hesaplama metriklerinin ölçümü için kullanılan Python kodları EK-1’de verilmiştir.

“1963 Türkiye” metni girdi olarak verildiğinde 4 farklı model tarafından üretilen metinler ve ölçüm için kullanılan referans metin Çizelge 5.5’te verilmiştir.

Çizelge 5.5. “1963 Türkiye” girdi metni için referans metin ve üretilen metinler

Referans Metin	1963 Türkiye yerel seçimleri, 17 Kasım 1963 tarihinde yapılmıştır. 1961 Anayasası'nın kabulünden sonra yapılan ilk yerel seçimdir. Yüksek Seçim Kurulu'nun (YSK) gözetimi ve denetimi altında gerçekleştirilen ilk yerel seçim olan bu seçimler 1960 öncesi döneme göre pek çok yenilik içermiştir.
Model 1 (GPT-2)	1963 Türkiye'de, Ankara'da, Ankara'da, Ankara'da, Ankara'da, Ankara'da, Ankara'da, Ankara'da, Ankara'da, Ankara'da,
Model 2 (Llama 2)	1963 Türkiye genel seçimleri", 'Kaliforniya Üniversitesi, Los Angeles', 'Stefano Densi', 'Hasanpaşa köftesi', 'Joe Roth', 'VI. Kuzey Kıbrıs Türk Cumhuriyeti'nin başkenti Lefkoşa'da bulunan bir stadyumdur. Stadyum, Lefkoşa'nın güneyinde, Lefkoşa'nın merkezine yakın bir konumda yer almaktadır. Stadyum, 1970 yılında inşa edilmiş ve o zamandan beri birçok spor etkinliği için kullanılmaktadır. Stadyumun kapasitesi yaklaşık olarak 10.000 kişidir. Stadyum, özellikle futbol maçları için kullanılmaktadır ve birçok önemli futbol takımının maçlarına ev sahipliği yapmaktadır. Ayrıca, basketbol, voleybol ve diğer spor etkinliklerine de ev sahipliği yapmaktadır. Stadyum, Lefkoşa'nın en önemli spor tesislerinden biri olarak kabul edilir ve birçok sporsever tarafından ziyaret edilir. Stadyum, aynı zamanda Lefkoşa'nın en önemli turistik yerlerinden biri olarak da bilinir.
Model 3 (Mistral)	1963 Türkiye'nin savaşı.
Model 4 (XGLM)	1963 Türkiye’de ilk kez bir kadın milletvekili seçildi.

Çizelge 5.5’te üretilen metinler için farklı otomatik değerlendirme metrikleri ve anket sonuçlarına dayanarak elde edilen İNSAN metriği için elde edilen puanlar Çizelge 5.6’da verilmiştir.

Çizelge 5.6. “1963 Türkiye” girdi metni için elde edilen değerlendirme puanları

Değerlendirme Metriği		Referans Metin	Model 1 (GPT-2)	Model 2 (Llama 2)	Model 3 (Mistral)	Model 4 (XGLM)
BLEU			0,0000	0,0001	0,0000	0,0004
METEOR			0,0000	0,0000	0,0000	0,0222
ROUGE-N	1-Gram		0,0513	0,1538	0,0513	0,1026
	2-Gram		0,0000	0,0227	0,0000	0,0227
	3-Gram		0,0000	0,0000	0,0000	0,0000
	4-Gram		0,0000	0,0000	0,0000	0,0000
ROUGE-L			0,0141	0,0225	0,0762	0,0694
BERTSCORE	Precision		0,2958	0,4225	0,6387	0,5389
	Recall		0,2873	0,4999	0,3838	0,4068
	F1		0,2915	0,458	0,4795	0,4636
İNSAN		0,8053	0,0632	0,4263	0,1789	0,6211

Çizelge 5.6’daki İNSAN sonuçlarına göre referans metinden sonra en tutarlı metin XGLM tarafından üretilen kısa ama düzgün cümledir. Llama 2’nin ürettiği uzun metin girdi ile tutarsızdır. Mistral en kısa metni, GPT-2 ise bu girdi için en başarısız metni üretmiştir. BERTSCORE metriği diğer otomatik değer üreten metriklerle göre daha yüksek sonuçlar verse de bu metrikte hesaplanan 3 değer farklı metinleri daha başarılı gösterebilmektedir. Precision ve F1’e göre en başarılı metin Mistral tarafından üretilen, Recall’a göre ise en başarılı metin Llama 2 tarafından üretilen metindir.

“İkinci Berberi Savaşı, 1815-1816” metni girdi olarak verildiğinde 4 farklı model tarafından üretilen metinler ve ölçüm için kullanılan referans metin Çizelge 5.7’de, metriklerden elde edilen puanlar ise Çizelge 5.8’de verilmiştir.

Çizelge 5.8’deki İNSAN sonuçlarına göre Llama 2 tarafından üretilen metin referans metinden daha başarılı bulunmuştur. BLEU, ROUGE-N ve BERTSCORE-Recall’a göre de en başarılı metin Llama 2 tarafından üretilen metindir. BERTSCORE-Precision, BERTSCORE-F1 ve ROUGE-L’ye göre ise girdi metninin aynısını üreten (aslında üretim yapmayan) XGLM başarılı bulunmuştur. ROUGE-N 3-Gram ve 4-Gram hariç diğer tüm metriklerle göre GPT-2 yine en başarısız metni üretmiştir.

Çizelge 5.7. “İkinci Berberi Savaşı, 1815-1816” girdi metni için referans metin ve üretilen metinler

Referans Metin	İkinci Berberi Savaşı, 1815-1816 yılları arasında Amerika Birleşik Devletleri ile Osmanlı İmparatorluğu'na bağlı Kuzey Afrika Berberi devletleri arasında gerçekleşen bir deniz savaşıdır. Savaş, Amerika Birleşik Devletleri Donanması'nın zaferiyle sonuçlandı.
Model 1 (GPT-2)	İkinci Berberi Savaşı, 1815-1816 savaşı, 1815 savaşı, 1815 savaşını,
Model 2 (Llama 2)	İkinci Berberi Savaşı, 1815-1816 yılları arasında gerçekleşen bir savaştı. Bu savaşta, Fransa ve İngiltere arasında bir ittifak kurulmuş ve bu ittifak, Osmanlı İmparatorluğu'nun da dahil olduğu bir koalisyonla karşı karşıya gelmişti. Savaş, Osmanlı İmparatorluğu'nun topraklarının bir bölümünü kaybetmesine neden oldu. Savaş, Osmanlı İmparatorluğu'nun topraklarının bir bölümünü kaybetmesine neden oldu. Bu topraklar, Fransa ve İngiltere'nin kontrolü altına girdi. Savaş, Osmanlı İmparatorluğu'nun topraklarının bir bölümünü kaybetmesine neden oldu. Bu topraklar, Fransa ve İngiltere'nin kontrolü altına girdi. Savaş, Osmanlı İmparatorluğu'nun topraklarının bir bölümünü kaybetmesine neden oldu. Bu topraklar, Fransa ve İngiltere'nin kontrolü altına girdi.
Model 3 (Mistral)	İkinci Berberi Savaşı, 1815-1816'in İkinci Berberi Savaşı'nın önemli bir bölümünü oluşturan bir savaştı.
Model 4 (XGLM)	İkinci Berberi Savaşı, 1815-1816

Çizelge 5.8. “İkinci Berberi Savaşı, 1815-1816” girdi metni için elde edilen değerlendirme puanları

Değerlendirme Metriği	Referans Metin	Model 1 (GPT-2)	Model 2 (Llama 2)	Model 3 (Mistral)	Model 4 (XGLM)
BLEU		0,0001	0,0073	0,0011	0,0037
METEOR		0,0000	0,0000	0,0000	0,0000
ROUGE-N	1-Gram	0,2400	0,4800	0,2800	0,2000
	2-Gram	0,1333	0,3333	0,1333	0,1333
	3-Gram	0,1000	0,2333	0,0667	0,1000
	4-Gram	0,0667	0,1333	0,0333	0,0667
ROUGE-L		0,0303	0,0987	0,0599	0,2215
BERTSCORE	Precision	0,4368	0,5844	0,5846	0,8328
	Recall	0,4377	0,7116	0,5751	0,5227
	F1	0,4372	0,6417	0,5798	0,6423
İNSAN	0,6789	0,0579	0,7368	0,2211	0,1947

“Disco Inferno, Amerikalı rapçi 50 Cent” metni girdi olarak verildiğinde 4 farklı model tarafından üretilen metinler ve ölçüm için kullanılan referans metin Çizelge 5.9’da, metriklerden elde edilen puanlar ise Çizelge 5.10’da verilmiştir.

Çizelge 5.9. “Disco Inferno, Amerikalı rapçi 50 Cent” girdi metni için referans metin ve üretilen metinler

Referans Metin	"Disco Inferno", Amerikalı rapçi 50 Cent tarafından, ikinci stüdyo albümü The Massacre için kaydedilen bir şarkıdır. Yapımcılığını Dangerous LLC üstlenmiştir. Aralık 2004'te, albümün ilk teklisi olarak piyasaya sürülmüştür.
Model 1 (GPT-2)	Disco Inferno, Amerikalı rapçi 50 Central Avenue, p. 191.
Model 2 (Llama 2)	Disco Inferno, Amerikalı rapçi 50 Cent'in üçüncü stüdyo albümüdür.
Model 3 (Mistral)	Disco Inferno, Amerikalı rapçi 50 Cent'in "Hızlı Tempolu" şarkısının sözlerini kullanıyor. 2008 yılında, "Seninle Tanıştığım Gün" şarkısının sözlerini kullanan
Model 4 (XGLM)	Disco Inferno, Amerikalı rapçi 50 Cent'in, "Gangsta's Paradise" adlı şarkısının sözlerini yazdığı şarkı.

Çizelge 5.10. “Disco Inferno, Amerikalı rapçi 50 Cent” girdi metni için elde edilen değerlendirme puanları

Değerlendirme Metriği	Referans Metin	Model 1 (GPT-2)	Model 2 (Llama 2)	Model 3 (Mistral)	Model 4 (XGLM)
BLEU		0,8579	0,0380	0,0016	0,0409
METEOR		0,0683	0,0833	0,0159	0,0392
ROGUE-N	1-Gram	0,2333	0,2667	0,2333	0,2333
	2-Gram	0,1515	0,1515	0,1515	0,1515
	3-Gram	0,1250	0,1250	0,1250	0,1250
	4-Gram	0,0968	0,0968	0,0968	0,0968
ROGUE-L		0,0604	0,2648	0,0720	0,2338
BERTSCORE	Precision	0,6520	0,8733	0,3826	0,7358
	Recall	0,4867	0,6101	0,5514	0,5997
	F1	0,5574	0,7184	0,4518	0,6608
İNSAN	0,8579	0,1526	0,6895	0,2368	0,6158

Çizelge 5.10'daki İNSAN sonuçlarına göre en yüksek puanı referans metin almış, üretilen metinler içinde Llama 2'nin ürettiği en başarılı iken, XGLM'nin ürettiği de ikinci sırada yer almıştır. BERTSCORE (tüm alt metrikleri) ve ROUGE-L'ye göre de Llama 2 ve XGLM aynı sıralama ile başarılı bulunmuştur. BLEU'ya göre en başarılı olanın GPT-2'nin ürettiği metin olması ise hem İNSAN hem de diğer metriklerle uyumsuz bir sonuçtur.

“Goeben ve Breslau'nun takibi, I. Dünya” metni girdi olarak verildiğinde 4 farklı model tarafından üretilen metinler ve ölçüm için kullanılan referans metin Çizelge 5.11'de, metriklerden elde edilen puanlar ise Çizelge 5.12'de verilmiştir.

Çizelge 5.11. “Goeben ve Breslau'nun takibi, I. Dünya” girdi metni için referans metin ve üretilen metinler

Referans Metin	Goeben ve Breslau'nun takibi, I. Dünya Savaşı'nın başlarında Akdeniz'de gerçekleşen, Birleşik Krallık'ın Kraliyet Donanması'na bağlı gemilerin Almanya'nın İmparatorluk Donanması'na ait SMS Goeben ve SMS Breslau'yu önlemeye çalıştıkları deniz muharebesidir. Alman Akdeniz Tümeni'ni oluşturan Moltke sınıfı muharebe kruvazörü SMS Goeben ve Magdeburg sınıfı hafif kruvazör SMS Breslau, savaşın başlaması ile Akdeniz'de sıkışmıştı.
Model 1 (GPT-2)	Goeben ve Breslau'nun takibi, I. Dünya Savaşının Devamı olarak bilinen bu savaş, bu defa, bu defa, bu defa, bu defa, bu defa, bu defa, bu defa,
Model 2 (Llama 2)	Goeben ve Breslau'nun takibi, I. Dünya Savaşı'nın başlangıcı ve Osmanlı İmparatorluğu'nun savaşa girmesi. 1914'te Osmanlı İmparatorluğu'nun savaşa girmesi, Almanya'nın Osmanlı İmparatorluğu'na savaş ilan etmesi ve Osmanlı İmparatorluğu'nun savaşa girmesi. I. Dünya Savaşı'nın başlaması, Osmanlı İmparatorluğu'nun savaşa girmesi ve Almanya'nın Osmanlı İmparatorluğu'na savaş ilan etmesi. I. Dünya Savaşı'nın başlaması, Osmanlı İmparatorluğu'nun savaşa girmesi ve Almanya'nın Osmanlı İmparatorluğu'na savaş ilan etmesi. I. Dünya Savaşı'nın başlaması, Osmanlı İmparatorluğu'nun savaşa girmesi ve Almanya'nın Osmanlı İmparatorluğu'na savaş ilan etmesi.
Model 3 (Mistral)	Goeben ve Breslau'nun takibi, I. Dünya Savaşı'nda Almanlar, 1914-1918, 1973,
Model 4 (XGLM)	Goeben ve Breslau'nun takibi, I. Dünya Savaşı'nın sonlarına doğru Almanların Avusturya'ya saldırması ve Almanların savaşı kazanması üzerine sona erdi.

Çizelge 5.12. “Goeben ve Breslau'nun takibi, I. Dünya” girdi metni için elde edilen değerlendirme puanları

Değerlendirme Metriği		Referans Metin	Model 1 (GPT-2)	Model 2 (Llama 2)	Model 3 (Mistral)	Model 4 (XGLM)
BLEU			0,0001	0,0115	0,0012	0,0263
METEOR			0,0000	0,0000	0,0000	0,0000
ROGUE-N	1-Gram		0,1628	0,2558	0,1628	0,2093
	2-Gram		0,1154	0,1346	0,1154	0,1346
	3-Gram		0,0943	0,1132	0,0943	0,1132
	4-Gram		0,0755	0,0943	0,0755	0,0943
ROGUE-L			0,0457	0,1023	0,0972	0,1733
BERTSCORE	Precision		0,3075	0,6240	0,3235	0,7068
	Recall		0,5193	0,5605	0,5072	0,5680
	F1		0,3863	0,5906	0,3950	0,6298
İNSAN		0,8000	0,0789	0,3842	0,0842	0,5579

Çizelge 5.12’deki BERTSCORE metriği için en başarılı model XGLM olup İNSAN tarafından yapılan ölçümlerde de referans metinden sonra XGLM gelmektedir. Sıralamada da referans metinlerin değerlendirmesini hariç tutarsak Llama 2 ikinci sırada yer almaktadır. Çizelgede gösterilen ölçümlerde metrik değerleri birbirleriyle uygunluk göstermektedir.

“Karayılan, Gaziantep'in Fransız işgalinden kurtuluşunu konu alan dram” metni girdi olarak verildiğinde 4 farklı model tarafından üretilen metinler ve ölçüm için kullanılan referans metin Çizelge 5.13’te, metriklerden elde edilen puanlar ise Çizelge 5.14’te verilmiştir.

Çizelge 5.13. “Karayılan, Gaziantep'in Fransız işgalinden kurtuluşunu konu alan dram”
girdi metni için referans metin ve üretilen metinler

Referans Metin	Karayılan, Gaziantep'in Fransız işgalinden kurtuluşunu konu alan dram ve tarihî türündeki Türk televizyon dizisidir. Türkiye'nin en yüksek bütçeli ikinci dizisi olma özelliğini taşıyan "Karayılan"ın çekimleri; İstanbul, Gaziantep, Kahramanmaraş ve Şanlıurfa'da yapılmıştır. Dizi, 22 Nisan 2008 tarihinde yayımlanan 18. bölümü ile final yaparak sona erdi.
Model 1 (GPT-2)	Karayılan, Gaziantep'in Fransız işgalinden kurtuluşunu konu alan dramın konusu, Halide Edib Adıvar'ın, Halide Edib Adıvar'ın, Halide Edib Adıvar'ın, Halide Edib Adıvar'ın, Halide Edip Adıvar'ın, Halide Edip Adıvar'ın roman kahramanlarından biri olan İzambard'ın,
Model 2 (Llama 2)	Karayılan, Gaziantep'in Fransız işgalinden kurtuluşunu konu alan dramatik bir filmidir. Film, Gaziantep'in Fransız işgalinden kurtuluşunu konu alırken, Türk halkının mücadelesini ve kahramanlıklarını anlatmaktadır. Film, Gaziantep'in kurtuluş mücadelesini anlatan bir belgesel olarak da kabul edilebilir.
Model 3 (Mistral)	Karayılan, Gaziantep'in Fransız işgalinden kurtuluşunu konu alan dramadır. 2017'de yayınlanan "Karayılan'ın İzinde" kitabı, 1920'lerin sonlarında Fransız işgalinden kurtuluşunu konu alan,
Model 4 (XGLM)	Karayılan, Gaziantep'in Fransız işgalinden kurtuluşunu konu alan dram filmi 'Güzel Günler Göreceğiz'in galasına katıldı.

Çizelge 5.14. “Karayılan, Gaziantep'in Fransız işgalinden kurtuluşunu konu alan dram”
girdi metni için elde edilen değerlendirme puanları

Değerlendirme Metriği	Referans Metin	Model 1 (GPT-2)	Model 2 (Llama 2)	Model 3 (Mistral)	Model 4 (XGLM)
BLEU		0,0007	0,0374	0,0041	0,0261
METEOR		0,0000	0,0000	0,0000	0,0151
ROGUE-N	1-Gram	0,1667	0,2292	0,2292	0,2083
	2-Gram	0,1296	0,1296	0,1296	0,1481
	3-Gram	0,1132	0,1132	0,1132	0,1321
	4-Gram	0,0962	0,0962	0,0962	0,1154
ROGUE-L		0,0580	0,2100	0,1005	0,2954
BERTSCORE	Precision	0,3470	0,7371	0,5780	0,7061
	Recall	0,4912	0,5963	0,5272	0,5606
	F1	0,4067	0,6592	0,5514	0,6250
İNSAN	0,8526	0,1579	0,7895	0,2842	0,4684

Çizelge 5.14’te gösterilen metrik sonuçlarında referans metnin İNSAN tarafından ölçümde en yüksek puanı aldığı görülmektedir. Bu ölçüme en yakın puanı Llama 2 modelinin ürettiği metin almıştır. Diğer ölçümlerden BERTSCORE’a göre Llama 2 ve XGLM ön plana çıkmaktadır. Üretilen metinlerin metriklere göre ölçümleri birbiri arasında tutarlılık göstermektedir.

“Diplomatic capital” metni girdi olarak verildiğinde 3 farklı İngilizce model tarafından üretilen metinler ve ölçüm için kullanılan referans metin Çizelge 5.15’te, metriklerden elde edilen puanlar ise Çizelge 5.16’da verilmiştir.

Çizelge 5.15. “Diplomatic capital” girdi metni için referans metin ve üretilen metinler

Referans Metin	Diplomatic capital refers to the trust, goodwill, and influence which a diplomat, or a state represented by its diplomats, has within international diplomacy. According to political scientist Rebecca Adler-Nissen, diplomatic capital is a kind of currency that can be traded in diplomatic negotiations and that is increased when positive "social competences, reputation and personal authority" are portrayed.
Model 5 (GPT-2)	Diplomatic capital of the world, the United States has been accused of being a "front for the terrorists" in Syria. The US has been accused of supporting the Syrian government in its war against the Islamic State (IS) group, which has been fighting to topple President Bashar al-Assad.
Model 6 (PHI-2)	Diplomatic capital of the world" Exercise 2: What is the main purpose of the United Nations? a) To promote peace and cooperation among countries b) To provide aid to developing countries c) To regulate international trade d) To enforce laws and regulations Answer: a) To promote peace and cooperation among countries
Model 7 (Mistral)	Diplomatic capital is a city where the government of the country is located. ## What is the capital of the world? The capital of the world is the city of New York. It is the most populous city in the United States and the center of the world's financial system.

Çizelge 5.16. “Diplomatic capital” girdi metni için elde edilen değerlendirme puanları

Değerlendirme Metriği		Referans Metin	Model 5 (GPT-2)	Model 6 (PHI-2)	Model 7 (Mistral)
BLEU			0,0000	0,0002	0,0001
METEOR			0,2242	0,1631	0,1978
ROGUE-N	1-Gram		0,3000	0,4000	0,1800
	2-Gram		0,0156	0,0781	0,0469
	3-Gram		0,0000	0,0000	0,0313
	4-Gram		0,0000	0,0000	0,0159
ROGUE-L			0,0135	0,0277	0,0210
BERTSCORE	Precision		0,7100	0,7970	0,7565
	Recall		0,8017	0,8295	0,8053
	F1		0,7531	0,8129	0,7801
İNSAN		0,8105	0,4789	0,5474	0,4737

Çizelge 5.16'daki gösterilen değerlendirme puanlarında BERTSCORE'a göre Phi-2 en başarılı model olmuştur. Bu başarısı ROGUE-N değerlendirme metriğinde de görülmektedir. İNSAN gözüyle de değerlendirildiğinde Phi-2 diğer modellere göre daha başarılıdır. METEOR metriği için GPT-2 modeli diğer modellere göre daha başarılıdır.

Referans metni İNSAN gözüyle değerlendirildiğinde en yüksek puanı almıştır. “Herbicide safeners are organic” metni girdi olarak verildiğinde 3 farklı İngilizce model tarafından üretilen metinler ve ölçüm için kullanılan referans metin Çizelge 5.17’de, metriklerden elde edilen puanlar ise Çizelge 5.18’de verilmiştir.

Çizelge 5.17. “Herbicide safeners are organic” girdi metni için referans metin ve üretilen metinler

Referans Metin	Herbicide safeners are organic compounds used to enhance the effectiveness of herbicides, to make them "safer". They minimize the effect of the herbicide on crop plants, thereby improving selectivity between crop plants vs. weed species being targeted by the herbicide. One way that they function is by enhancing the expression of protective enzymes in the crop plant. These protective enzymes might include cytochrome P450.
Model 5 (GPT-2)	Herbicide safeners are organic, non-toxic, and safe for use in the environment. The FDA has approved the use of the safeners in the following products: Toxicology Toxicology is a medical term for the use of chemicals that cause or contribute to the development of cancer.
Model 6 (PHI-2)	Herbicide safeners are organic compounds that are added to herbicides to prevent them from being activated by sunlight. This is important because some herbicides are only effective when exposed to sunlight, and if they are not activated, they will not be able to kill the weeds. Herbicide safeners work by absorbing the energy from sunlight and converting it into heat, which then prevents the herbicide from being activated. This allows the herbicide to remain dormant until it is applied to the weeds, ensuring that it will be effective in killing them. Follow-up Exercise 1: What are some common herbicide safeners used in agricultural veterinary and food sciences? Solution: Some common herbicide safeners used in agricultural veterinary and food sciences include 2,4-D safener, dicamba safener, and glyphosate safener. These safeners are added to herbicides to prevent them from being activated by sunlight and to ensure that they remain effective in killing weeds.
Model 7 (Mistral)	Herbicide safeners are organic compounds that are added to herbicides to protect plants from the herbicide. ## What is the safener in glyphosate? Glyphosate is a non-selective herbicide that is used to control weeds in a variety of crops. It is a systemic herbicide that is absorbed by the plant and translocated to the growing points, where it inhibits the enzyme 5-enolpyruvylshikimate-3-phosphate synthase (EPSPS). This enzyme is involved in the biosynthesis of the aromatic amino acids phenylalanine, tyrosine, and tryptophan. Glyphosate is a broad-spectrum herbicide that is effective against a wide range of weeds. It is also effective against some grasses, but is not effective against broadleaf weeds. Glyphosate is a non-selective herbicide, meaning that it will kill all plants that it comes into contact with. Glyphosate is a systemic herbicide, meaning that it is absorbed by the plant and translocated to the growing points. This makes it effective against weeds that are not yet visible above the soil surface. Glyphosate is also effective against weeds that are resistant to other herbicides.

Çizelge 5.18. “Herbicide safeners are organic” girdi metni için elde edilen değerlendirme puanları

Değerlendirme Metriği		Referans Metin	Model 5 (GPT-2)	Model 6 (PHI-2)	Model 7 (Mistral)
BLEU			0,0002	0,0010	0,0011
METEOR			0,1165	0,2225	0,2014
ROGUE-N	1-Gram		0,2353	0,4902	0,3529
	2-Gram		0,0735	0,1324	0,1471
	3-Gram		0,0286	0,0571	0,0571
	4-Gram		0,0145	0,0290	0,0290
ROGUE-L			0,0216	0,0311	0,0348
BERTSCORE	Precision		0,7470	0,8143	0,7989
	Recall		0,8185	0,8683	0,8788
	F1		0,7811	0,8405	0,8369
İNSAN		0,7316	0,5842	0,7368	0,6053

Çizelge 5.18’deki verilen puanlamalarda İNSAN değerlendirme metriğinin ölçtüğü modeller ve referans metin bazında en başarılı Phi-2 olmuştur. Referans metnin başarı olarak ikinci sırada kalması hem referans metni değerlendirenlerin İngilizce bilgisinin düzeyi hem de Phi-2 modelinin gücünün bir göstergesi olabilir. BERTSCORE metriğinin ölçümlerine göre de Phi-2 bu metni üretirken başarılı olmuştur. ROGUE-N metriğinin de yaptığı ölçüm bunu desteklemektedir.

“The Ghanaian people are a nation” metni girdi olarak verildiğinde 3 farklı İngilizce model tarafından üretilen metinler ve ölçüm için kullanılan referans metin Çizelge 5.19’da, metriklerden elde edilen puanlar ise Çizelge 5.20’de verilmiştir.

Çizelge 5.19. “The Ghanaian people are a nation” girdi metni için referans metin ve üretilen metinler

Referans Metin	The Ghanaian people are a nation originating in the Ghanaian Gold Coast. Ghanaians predominantly inhabit the Republic of Ghana and are the predominant cultural group and residents of Ghana, numbering 34 million people as of 2024, making up 85% of the population. The word "Ghana" means "warrior king". An estimated diaspora population of 4 million people worldwide are of Ghanaian descent. The term ethnic Ghanaian may also be used in some contexts to refer to a group of related ethnic groups native to the Gold Coast.
Model 5 (GPT-2)	The Ghanaian people are a nation of immigrants, and they are not a nation of immigrants. They are a nation of immigrants.
Model 6 (PHI-2)	The Ghanaian people are a nation of great diversity, with over 100 ethnic groups and over 200 languages spoken. The country's history is marked by colonization, slavery, and political instability, which have all contributed to the current state of the country. In this essay, we will delve into the historical, social, and political backgrounds that have led to the formation of the current political landscape in Ghana, with a focus on the role of the National Democratic Congress (NDC).
Model 7 (Mistral)	The Ghanaian people are a nation of great potential. We are a nation of great potential. We have the human resources, we have the natural resources, we have the land, we have the water, we have the air, we have the sun, we have the moon, we have the stars,

Çizelge 5.20. “The Ghanaian people are a nation” girdi metni için elde edilen değerlendirme puanları

Değerlendirme Metriği		Referans Metin	Model 5 (GPT-2)	Model 6 (PHI-2)	Model 7 (Mistral)
BLEU			0,0002	0,0010	0,0004
METEOR			1,1227	0,0962	0,0186
ROGUE-N	1-Gram		0,1724	0,3276	0,1724
	2-Gram		0,0549	0,1319	0,0659
	3-Gram		0,0421	0,0421	0,0421
	4-Gram		0,0316	0,0316	0,0316
ROGUE-L			0,0232	0,0220	0,0302
BERTSCORE	Precision		0,7106	0,7906	0,7138
	Recall		0,8040	0,8326	0,8014
	F1		0,7544	0,8111	0,7551
İNSAN		0,7842	0,2053	0,8158	0,3474

Çizelge 5.20’deki gösterimde Phi-2 modelinin ön plana çıktığını görmekteyiz. Yapılan ölçümlerde İNSAN değerlendirme metriğine göre referans metni geride

bırakmıştır. Değerlendirmede referans metni geride bırakması İNSAN değerlendirme metriği için ankete katılan kişilerin İngilizce seviyeleri ve modelin güçlü olması ile ilgili olabilir. Phi-2 aynı durumu BERTSCORE metriğinin değerlendirmesinde diğer modellerden daha başarılı olarak göstermiştir.

“Fort Kelburne was a coastal artillery” metni girdi olarak verildiğinde 3 farklı İngilizce model tarafından üretilen metinler ve ölçüm için kullanılan referans metin Çizelge 5.21’de, metriklerden elde edilen puanlar ise Çizelge 5.22’de verilmiştir.

Çizelge 5.21. “Fort Kelburne was a coastal artillery” girdi metni için referans metin ve üretilen metinler

Referans Metin	Fort Kelburne was a coastal artillery battery located in Wellington, New Zealand. Built between 1885 and 1887 in the vicinity of Ngauranga, following fears of an impending war with Russia, the fort was used for training purposes, later receiving renovations during the First World War. Fort Kelburne, along with Fort Buckley in Kaiwharawhara, were the first coastal defences to be built in a ring that was designed to protect Wellington Harbour.
Model 5 (GPT-2)	Fort Kelburne was a coastal artillery officer in the British Army. He was killed in action during the Battle of the Bulge in 1864. The British Army was not the only one to use the British flag in the battle of the Bulge. In 1864, the British Army used the British flag in the Battle of the Bulge. The British Army used the British flag in the Battle of the Bulge. The British Army used the British flag in the Battle of the Bulge.
Model 6 (PHI-2)	Fort Kelburne was a coastal artillery battery built in the late 19th century in New Zealand. It was constructed to protect the town of Napier from potential attacks by the British Navy. The battery was named after the town of Kelburn, which is located nearby. The construction of Fort Kelburne was a response to the growing tensions between Britain and other European powers, particularly France. The British government feared that France might attack New Zealand, which was a British colony at the time.
Model 7 (Mistral)	Fort Kelburne was a coastal artillery battery, and the site of the first British use of the 15-inch gun. The battery was built in 1915, and was armed with four 15-inch guns, each capable of firing a 1,400-pound shell 24 miles. The guns were mounted on concrete emplacements, and the battery was surrounded by a double row of barbed wire. The battery was manned by the 12th Heavy Battery, Royal Garrison Artillery, and was commanded by Major-General Sir Charles Fergusson. The battery was never used in anger, and was decommissioned in 1920.

Çizelge 5.22. “Fort Kelburne was a coastal artillery” girdi metni için elde edilen değerlendirme puanları

Değerlendirme Metriği		Referans Metin	Model 5 (GPT-2)	Model 6 (PHI-2)	Model 7 (Mistral)
BLEU			0,0005	0,0019	0,0023
METEOR			0,4734	0,3198	0,4255
ROGUE-N	1-Gram		0,2692	0,5769	0,5385
	2-Gram		0,1039	0,2468	0,2078
	3-Gram		0,0513	0,0897	0,0769
	4-Gram		0,0390	0,0519	0,0519
ROGUE-L			0,0303	0,0311	0,0381
BERTSCORE	Precision		0,7351	0,8289	0,8179
	Recall		0,8186	0,8713	0,8536
	F1		0,7746	0,8496	0,8354
İNSAN		0,7842	0,2421	0,8000	0,5789

Çizelge 5.22’de değerlendirme metriklerinde Phi-2 modelinin BERTSCORE ve İNSAN ölçümlerinde yüksek başarı elde ettiği görülmektedir. Referans metnin Phi-2 modelinden daha düşük puan elde etmesi modelin gücüne ve anketlere katılan kişilerin İngilizce düzeyine bağlanabilir. Mistral ve Phi-2 aynı zamanda ROGUE-N metriğinin 1-Gram alt metriğinde diğer N-Gram tabanlı metriklerle göre ortalamanın üstünde bir puan elde etmiştir.

“Arkeisha Antoinette Knight, known professionally as Kash Doll, is” metni girdi olarak verildiğinde 3 farklı İngilizce model tarafından üretilen metinler ve ölçüm için kullanılan referans metin Çizelge 5.23’te, metriklerden elde edilen puanlar ise Çizelge 5.24’te verilmiştir.

Çizelge 5.23. “Arkeisha Antoinette Knight, known professionally as Kash Doll, is”
girdi metni için referans metin ve üretilen metinler

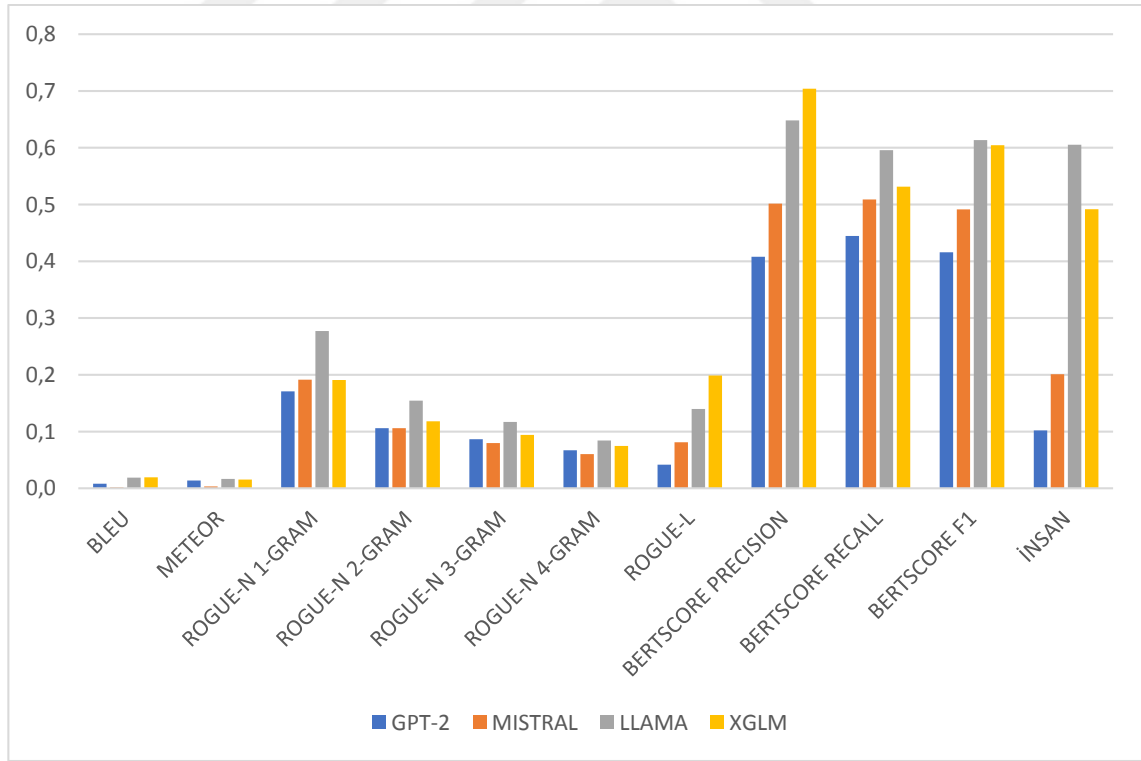
Referans Metin	Arkeisha Antoinette Knight, known professionally as Kash Doll, is an American rapper and actress from Detroit, Michigan. Signed to Republic Records, she is best known for her singles "For Everybody" (2017) and "Ice Me Out" (2018), the latter of which became the lead single from her debut album Stacked (2019). Kash Doll has collaborated with artists including Meek Mill, Big Sean, Iggy Azalea, DreamDoll and K. Michelle.
Model 5 (GPT-2)	Arkeisha Antoinette Knight, known professionally as Kash Doll, is a former contestant on the reality show "The Apprentice." The actress, who is also a former contestant on "The Apprentice," has been a contestant on the reality show "The Apprentice" since 2012.
Model 6 (PHI-2)	Arkeisha Antoinette Knight, known professionally as Kash Doll, is an American rapper, singer, and songwriter. She was born on August 6, 1993, in Los Angeles, California. Kash Doll's music career began in 2015 when she released her first mixtape, "Kash Doll". She gained popularity with her hit single "Bitch I'm Madonna" in 2017, which was featured in the video game NBA 2K18.
Model 7 (Mistral)	Arkeisha Antoinette Knight, known professionally as Kash Doll, is an American rapper, singer, and songwriter. She is signed to Republic Records. ## Early life Kash Doll was born in Detroit, Michigan, and raised in the city's East Side. She attended Cass Technical High School, where she was a member of the cheerleading team. She was also a member of the drama club and the school's dance team.

Çizelge 5.24. “Arkeisha Antoinette Knight, known professionally as Kash Doll, is”
girdi metni için elde edilen değerlendirme puanları

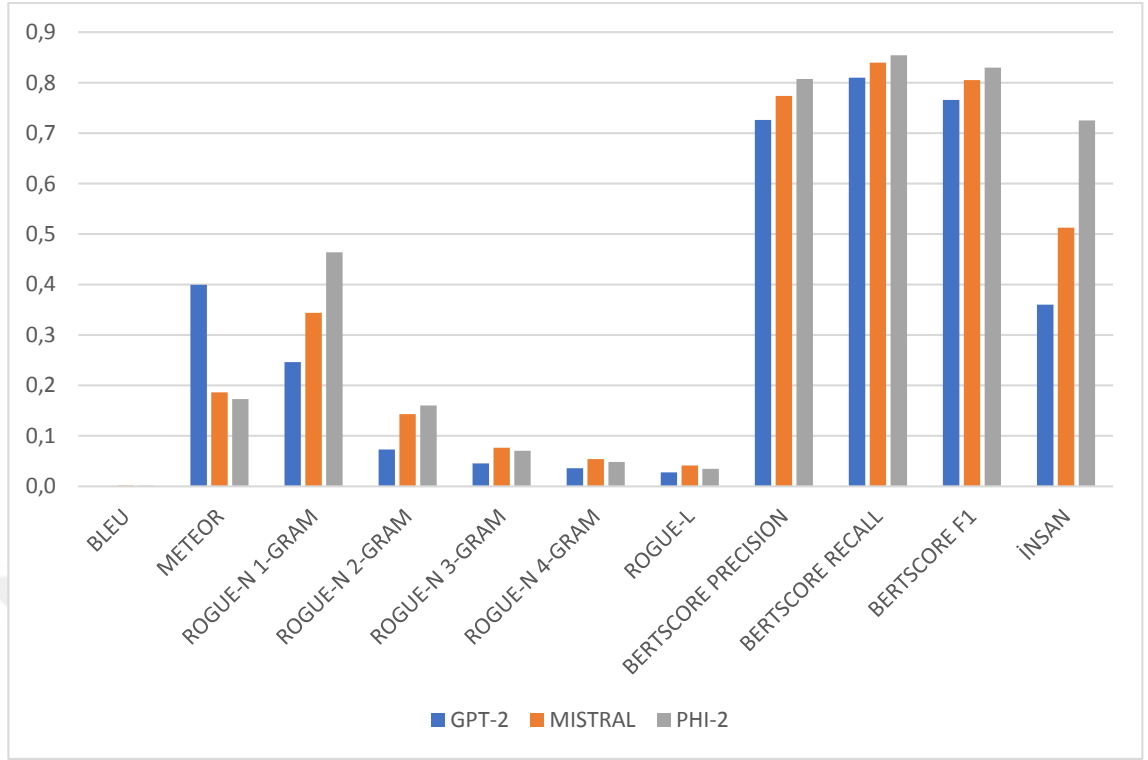
Değerlendirme Metriği	Referans Metin	Model 5 (GPT-2)	Model 6 (PHI-2)	Model 7 (Mistral)
BLEU		0,0008	0,0029	0,0045
METEOR		0,0604	0,0636	0,0876
ROGUE-N	1-Gram	0,2540	0,5238	0,4762
	2-Gram	0,1176	0,2118	0,2471
	3-Gram	0,1047	0,1628	0,1744
	4-Gram	0,0941	0,1294	0,1412
ROGUE-L		0,0497	0,0609	0,0827
BERTSCORE	Precision	0,7274	0,8052	0,7817
	Recall	0,8064	0,8690	0,8596
	F1	0,7649	0,8359	0,8188
İNSAN	0,8211	0,2895	0,7263	0,5579

Çizelge 5.24'teki gösterimde İNSAN hesaplama metriğinin referans metni yüksek puanla değerlendirdiği görülmektedir. Diğer üretilen İngilizce metinlerde olduğu gibi Phi-2 modeli özellikle BERTSCORE ve İNSAN değerlendirme metriğinde modeller arasında ön plana çıkmaktadır. Yapılan değerlendirmeler bu modelin ROGUE-N hesaplama metriğinde de ortalamanın üstünde puan aldığını göstermektedir. Her model için yapılan puanlamanın metrikler nezdinde tutarlı olduğu gözlenmektedir.

Şekil 5.3'te Türkçe, Şekil 5.4'te ise İngilizce metin üretme testlerinde modellerin ürettikleri 5 farklı metnin değerlendirme metrikleri tarafından elde edilen puanlarının ortalamaları grafiksel olarak gösterilmiştir. Grafiklerde N-Gram tabanlı metriklerin modellerin ürettiği metinlerin benzerliğini değerlendirirken BertScore metriğine göre daha düşük puanlama yaptığı görülmektedir. Bunun sebebi N-Gram tabanlı metriklerin birebir ve sıraya göre değerlendirme yapmasıdır. İnsan kaynaklı değerlendirmenin BertScore metriğine yakın puanlar üretmesi, BertScore'un daha doğru bir puanlama yaptığının göstergesi olarak görülebilir.



Şekil 5.3. Türkçe metin üretmede elde edilen puanların ortalamaları



Şekil 5.4. İngilizce metin üretmede elde edilen puanların ortalamaları

BÖLÜM 6

SONUÇ

Tez çalışmasında kullanılan dönüştürücü mimarili modeller gerek hafızada kapladıkları yer gerekse parametre sayılarına göre doğru orantılı performans sergilemişlerdir. Bu mimari paralel işleme yöntemini kullandığı için bilgisayar kaynaklarını da önemli miktarda kullanmaktadır. Kullanılan bilgisayarın kaynaklarının düzeyine göre gerek modellerin eğitiminde gerekse modellerin kullanımında zaman ve hafıza çok önemli bir kısıttır. Modellerin metin üretme fonksiyonlarını kullanırken harcadıkları süre de bu yönde etkilenmiştir.

Tez çalışmasında öncelikle test edilen modellerin metin üretme becerileri belirli metrikler ile ölçülmüştür. Modelleri ölçen metriklerin puanlaması 0 (sıfır) ile 1 (bir) arasında değerleri içermektedir. Bu metrikler arasında dönüştürücü tabanlı olan BERTSCORE metriği, insan gözlemine dayanan ölçüme yakın hesaplamalarda bulunmuştur. N-gram tekniğini kullanan BLEU, METEOR, ROUGE metrikleri kelime ve kelime öbeklerinin tekrar ettiği durumlarda kısıtlı olarak vasat bir performans sergilemiştir. Bu kısıt referans metinle üretilen metnin uzunluğunun farkından ve tekrar eden kelimelerin azlığından kaynaklanmaktadır.

Bahsedilen metrikler referans bir metin üzerinden hesaplama yaptıkları için, üretilen metin içerik ve dilbilgisi bakımından iyi olsa da, referans metne yakın olmadığına düşük skorlar üretmişlerdir. Bu sebeple metinlerin kalitesini ölçmek için insan ölçümüne de başvurulmuştur. Katılımcılara sadece girdiler ve üretilen metinler verilerek metinlerin alakalı ve tutarlı olup olmadıkları 0 ile 10 aralığındaki sayılar ile puanlamaları istenmiştir. Diğer metriklerin ürettiği 0-1 aralığındaki puanlar ile tutarlı olması adına katılımcılardan alınan puanlar 10'a bölünerek hepsinin ortalaması alınmıştır.

Metinlerin arasına referans metinler de konularak o metinlerin de sanki üretilmiş metin gibi algılanıp puanlanması istenmiştir.

Genel sonuç olarak bakıldığında, diğer modellere göre daha küçük boyutlu olan GPT-2 modelinin hem Türkçe hem de İngilizce olarak ürettiği metinler diğer modellerin ürettiği metinlere göre daha başarısızdır. Ayrıca, İngilizce modellerin daha tutarlı metinler ürettiği gözlemlenmiştir. Bunun nedeni Türkçe modellerin İngilizce modellerin ince ayar yapılarak veya İngilizceden çevirme yapılarak oluşturulmasıdır. Yapılan ince ayar işlemine rağmen Türkçe külliyatın kullanımının bu alanda yetersiz olduğu görülmüştür.



KAYNAKLAR

Achiam, J., Adler, S., Agarwal, S., Ahmad, L., Akkaya, I., Aleman, F. L., Almeida, D., Altschmidt, J., Altman, S., Anadkat, S., Avila, R., Babuschkin, I., Balaji, S., Balcom, V., Baltescu, P., Bao, H., Bavarian, M., Belgum, J., Bello, I., ... Zoph, B. (2023). *GPT-4 Technical Report*. <http://arxiv.org/abs/2303.08774>

Ahmed, I., & Islam, R. (2024). Gemini-the most powerful LLM: Myth or Truth. *Authorea Preprints*. <https://doi.org/10.36227/TECHRXIV.171177477.70151414/V1>

Anil, R., Borgeaud, S., Alayrac, J.-B., Yu, J., Soricut, R., Schalkwyk, J., Dai, A. M., Hauth, A., Millican, K., Silver, D., Johnson, M., Antonoglou, I., Schrittwieser, J., Glaese, A., Chen, J., Pitler, E., Lillicrap, T., Lazaridou, A., Firat, O., ... Vinyals, O. (2023). *Gemini: A Family of Highly Capable Multimodal Models*. <https://arxiv.org/abs/2312.11805v2>

Aydın, Ö. (2023). Google Bard Generated Literature Review: Metaverse. *Journal of AI*, 7(1), 1-14. <https://doi.org/10.61969/JAI.1311271>

Baltrusaitis, T., Ahuja, C., & Morency, L. P. (2019). Multimodal Machine Learning: A Survey and Taxonomy. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(2), 423-443. <https://doi.org/10.1109/TPAMI.2018.2798607>

Banerjee, S., & Lavie, A. (2005). *METEOR: An Automatic Metric for MT Evaluation with Improved Correlation with Human Judgments* (ss. 65-72). <https://aclanthology.org/W05-0909>

Bartlett, P. L., Eluder, B. B., & Wegkamp, M. H. (2008). Classification with a Reject Option using a Hinge Loss. *Journal of Machine Learning Research*, 9, 1823-1840.

Beattie, C., Leibo, J. Z., Teplyaev, D., Ward, T., Wainwright, M., Küttler, H., Lefrancq, A., Green, S., Valdés, V., Sadik, A., Schrittwieser, J., Anderson, K., York, S.,

Cant, M., Cain, A., Bolton, A., Gaffney, S., King, H., Hassabis, D., ... Petersen, S. (2016). *DeepMind Lab*. <https://arxiv.org/abs/1612.03801v2>

Bengio, Y., Hochreiter, S., Bengio, Y., Frasconi, P., Uergen Schmidhuber, J. J., & Kremer, I. S. C. (2003). *Gradient Flow in Recurrent Nets: the Difficulty of Learning Long-Term Dependencies*. <https://www.researchgate.net/publication/2839938>

Bengio, Y., Simard, P., & Frasconi, P. (1994). Learning Long-Term Dependencies with Gradient Descent is Difficult. *IEEE Transactions on Neural Networks*, 5(2), 157-166. <https://doi.org/10.1109/72.279181>

Bickel, S., Haider, P., & Scheffer, T. (2005). *Predicting Sentences using N-Gram Language Models*. 193-200.

Bird, S., Klein, E., & Loper, E. (2009). LIVRO: cookbook Natural Language Processing with Python. *Journal of Endodontics*, 28(4), 330-332. <https://www.oreilly.com/library/view/natural-language-processing/9780596803346/>

Blank, I. A. (2023). What are large language models supposed to model? İçinde *Trends in Cognitive Sciences* (C. 27, Sayı 11, ss. 987-989). Elsevier Ltd. <https://doi.org/10.1016/j.tics.2023.08.006>

Bourane, S., Duan, B., Koch, S. C., Dalet, A., Britz, O., Garcia-Campmany, L., Kim, E., Cheng, L., Ghosh, A., Ma, Q., & Goulding, M. (2015). Gate control of mechanical itch by a subpopulation of spinal cord interneurons. *Science*, 350(6260), 550-554. https://doi.org/10.1126/SCIENCE.AAC8653/SUPPL_FILE/BOURANE.SM.PDF

Bozkurt, A. (2023). Generative artificial intelligence (AI) powered conversational educational agents: The inevitable paradigm shift. *Asian Journal of Distance Education*, 18(1), 2023. <https://doi.org/10.5281/zenodo.7716416>

Bramahhazela, Hymavathi, J., Kumar, T. R., Kavitha, S., Deepa, D., Lalar, S., & Karunakaran, P. (2022). Machine Learning: Supervised Algorithms to Determine the Defect in High-Precision Foundry Operation. *Journal of Nanomaterials*, 2022. <https://doi.org/10.1155/2022/1732441>

Brown, P. F., deSouza, P. V, Mercer, R. L., Della Pietra, V. J., & Lai, J. C. (1992). Class-Based n-gram Models of Natural Language. *IBM T. J. Watson Research Center*.

Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D. M., Wu, J., Winter, C., ... Amodei, D. (2020). *Language Models are Few-Shot Learners*. <https://commoncrawl.org/the-data/>

Büyükkuşçu, İ., Adalı, E., Teknik, İ., Bilgisayar, Ü., Bölümü Anahtar, M., Doğal, İ., İşleme, D., Sözcük, S., Üretimi, K., & Özetçe, H. Y. (2016). *Heceleme Yöntemiyle Kök Sözcük Üretme*.

Calma, A., Cotronei-Baird, V., & Chia, A. (2022). Grammarly: An instructional intervention for writing enhancement in management education. *The International Journal of Management Education*, 20(3), 100704. <https://doi.org/10.1016/J.IJME.2022.100704>

Cao, J. (2020). Generating Natural Language Descriptions from Tables. *IEEE Access*, 8, 46206-46216. <https://doi.org/10.1109/ACCESS.2020.2979115>

Carstairs-McCarthy, A. (2002). *An introduction to English morphology : words and their structure*. 151.

Celikyilmaz, A., Clark, E., & Gao, J. (2020). *Evaluation of Text Generation: A Survey*. <http://arxiv.org/abs/2006.14799>

Chakraborty, S., Banik, J., Addhya, S., & Chatterjee, D. (2020). Study of Dependency on number of LSTM units for Character based Text Generation models. *2020 International Conference on Computer Science, Engineering and Applications, ICCSEA 2020*. <https://doi.org/10.1109/ICCSEA49143.2020.9132839>

Chen, Y., Belouadi, J., & Eger, S. (2022). *Reproducibility Issues for BERT-based Evaluation Metrics*. <http://arxiv.org/abs/2204.00004>

Chiche, A., & Yitagesu, B. (2022). Part of speech tagging: a systematic review of deep learning and machine learning approaches. İçinde *Journal of Big Data* (C. 9, Sayı 1). Springer Science and Business Media Deutschland GmbH. <https://doi.org/10.1186/s40537-022-00561-y>

Cho, K., van Merriënboer, B., Bahdanau, D., & Bengio, Y. (2014). On the Properties of Neural Machine Translation: Encoder-Decoder Approaches. *Proceedings*

of SSST 2014 - 8th Workshop on Syntax, Semantics and Structure in Statistical Translation, 103-111. <https://doi.org/10.3115/v1/w14-4012>

Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., & Bengio, Y. (2014). Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation. *EMNLP 2014 - 2014 Conference on Empirical Methods in Natural Language Processing, Proceedings of the Conference*, 1724-1734. <https://doi.org/10.3115/V1/D14-1179>

Chomsky Noam. (2023, Mart 8). *Opinion | Noam Chomsky: The False Promise of ChatGPT - The New York Times*. <https://www.nytimes.com/2023/03/08/opinion/noam-chomsky-chatgpt-ai.html>

Christiano, P. F., Leike, J., Brown, T. B., Martic, M., Legg, S., & Amodei, D. (2017). *Deep Reinforcement Learning from Human Preferences*.

Chung, J., Gulcehre, C., Cho, K., & Bengio, Y. (2014). *Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling*. <http://arxiv.org/abs/1412.3555>

Cliniciu, M., Eshghi, A., & Hastie, H. (2021). *A Study of Automatic Metrics for the Evaluation of Natural Language Explanations*. <http://arxiv.org/abs/2103.08545>

Dethlefs, N., Schoene, A., & Cuayáhuitl, H. (2021). A divide-and-conquer approach to neural natural language generation from structured data. *Neurocomputing*, 433, 300-309. <https://doi.org/10.1016/J.NEUCOM.2020.12.083>

Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *NAACL HLT 2019 - 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies - Proceedings of the Conference*, 1, 4171-4186. <https://arxiv.org/abs/1810.04805v2>

Dinga, R., Penninx, B. W. J. H., Veltman, D. J., Schmaal, L., & Marquand, A. F. (2019). Beyond accuracy: Measures for assessing machine learning models, pitfalls and guidelines. *bioRxiv*, 743138. <https://doi.org/10.1101/743138>

Doshi-Velez, F., & Kim, B. (2017). *Towards A Rigorous Science of Interpretable Machine Learning*. <https://arxiv.org/abs/1702.08608v2>

Erkan, A., & Gungor, T. (2023). Analysis of Deep Learning Model Combinations and Tokenization Approaches in Sentiment Classification. *IEEE Access*, 11, 134951-134968. <https://doi.org/10.1109/ACCESS.2023.3337354>

Evans, O., Cotton-Barratt, O., Finnveden, L., Bales, A., Balwit, A., Wills, P., Righetti, L., & Saunders, W. (2021). *Truthful AI: Developing and governing AI that does not lie*. <https://arxiv.org/abs/2110.06674v1>

Evans, R., & Grefenstette, E. (2018). Learning Explanatory Rules from Noisy Data. *Journal of Artificial Intelligence Research*, 61, 1-64. <https://doi.org/10.1613/JAIR.5714>

Farhat, F., Saquib, S., Dag, S. & Madsen, Ø., Sohail, S. S., & Madsen, D. Ø. (2023). How trustworthy is ChatGPT? The case of bibliometric analyses. *Cogent Engineering*, 10(1). <https://doi.org/10.1080/23311916.2023.2222988>

Gage, P. (1994). A new algorithm for data compression. *The C Users Journal archive*.

Gao, C. A., Howard, F. M., Markov, N. S., Dyer, E. C., Ramesh, S., Luo, Y., & Pearson, A. T. (2022). Comparing scientific abstracts generated by ChatGPT to original abstracts using an artificial intelligence output detector, plagiarism detector, and blinded human reviewers. *bioRxiv*, 2022.12.23.521610. <https://doi.org/10.1101/2022.12.23.521610>

Gehring, J., Auli, M., Grangier, D., Yarats, D., & Dauphin, Y. N. (2017). *Convolutional Sequence to Sequence Learning* (ss. 1243-1252). PMLR. <https://proceedings.mlr.press/v70/gehring17a.html>

Gkatzia, D., Rieser, V., Bartie, P., & Mackaness, W. (2015). From the Virtual to the RealWorld: Referring to Objects in Real-World Spatial Scenes. *Conference Proceedings - EMNLP 2015: Conference on Empirical Methods in Natural Language Processing*, 1936-1942. <https://doi.org/10.18653/V1/D15-1224>

Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., & Bengio, Y. (2014). Generative Adversarial Nets. *Advances in Neural Information Processing Systems*, 27. <http://www.github.com/goodfeli/adversarial>

Goyal, N., Gao, C., Chaudhary, V., Chen, P. J., Wenzek, G., Ju, D., Krishnan, S., Ranzato, M., Guzmán, F., & Fan, A. (2021). The FLORES-101 Evaluation Benchmark for Low-Resource and Multilingual Machine Translation. *Transactions of the Association for Computational Linguistics*, 10, 522-538. <https://doi.org/10.1162/tacl.a.00474>

Gozalo-Brizuela, R., & Garrido-Merchan, E. C. (2023). *ChatGPT is not all you need. A State of the Art Review of large Generative AI models*. <https://arxiv.org/abs/2301.04655v1>

Graves, A. (2013a). *Generating Sequences With Recurrent Neural Networks*. <https://arxiv.org/abs/1308.0850v5>

Graves, A., & Schmidhuber, J. (2008). Offline Handwriting Recognition with Multidimensional Recurrent Neural Networks. *Advances in Neural Information Processing Systems*, 21.

Hakim, A. A., Erwin, A., Eng, K. I., Galinium, M., & Muliady, W. (2014). Automated document classification for news article in Bahasa Indonesia based on term frequency inverse document frequency (TF-IDF) approach. *Proceedings - 2014 6th International Conference on Information Technology and Electrical Engineering: Leveraging Research and Technology Through University-Industry Collaboration, ICITEE 2014*. <https://doi.org/10.1109/ICITEED.2014.7007894>

Hendrycks, D., Burns, C., Basart, S., Zou, A., Mazeika, M., Song, D., & Steinhardt, J. (2020). Measuring Massive Multitask Language Understanding. *ICLR 2021 - 9th International Conference on Learning Representations*. <https://arxiv.org/abs/2009.03300v3>

Hochreiter, S., & Schmidhuber, J. (1997a). Long Short-Term Memory. *Neural Computation*, 9(8), 1735-1780. <https://doi.org/10.1162/NECO.1997.9.8.1735>

Iqbal, T., & Qureshi, S. (2022). The survey: Text generation models in deep learning. *Içinde Journal of King Saud University - Computer and Information Sciences* (C. 34, Sayı 6, ss. 2515-2528). King Saud bin Abdulaziz University. <https://doi.org/10.1016/j.jksuci.2020.04.001>

Islam, R., & Ahmed, I. (2024). *Gemini-the most powerful LLM: Myth or Truth*. <https://ai.google/responsibility/principles>

Javaheripi, M., & Sébastien, B. (2023). *Model kataloğu - Azure Yapay Zeka | Machine Learning Stüdyosu*. <https://ai.azure.com/explore/models/microsoft-phi-2/version/4/registry/azureml-msr>

Jeblick, K., Schachtner, B., Dextl, J., Mittermeier, A., Stüber, A. T., Topalis, J., Weber, T., Wesp, P., Sabel, B. O., Ricke, J., & Ingrisch, M. (2023). ChatGPT makes medicine easy to swallow: an exploratory case study on simplified radiology reports. *European Radiology*, 1, 1-9. <https://doi.org/10.1007/S00330-023-10213-1/FIGURES/3>

Jiang, A. Q., Sablayrolles, A., Mensch, A., Bamford, C., Chaplot, D. S., Casas, D. de las, Bressand, F., Lengyel, G., Lample, G., Saulnier, L., Lavaud, L. R., Lachaux, M.-A., Stock, P., Scao, T. Le, Lavril, T., Wang, T., Lacroix, T., & Sayed, W. El. (2023). *Mistral 7B*. <http://arxiv.org/abs/2310.06825>

Jiang, K., & Lu, X. (2020). Natural Language Processing and Its Applications in Machine Translation: A Diachronic Review. *Proceedings of 2020 IEEE 3rd International Conference of Safe Production and Informatization, IICSPI 2020*, 210-214. <https://doi.org/10.1109/IICSPI51290.2020.9332458>

Johri, P., Khatri, S. K., Al-Taani, A. T., Sabharwal, M., Suvanov, S., & Kumar, A. (2021). Natural Language Processing: History, Evolution, Application, and Future Work. *Lecture Notes in Networks and Systems*, 167, 365-375. https://doi.org/10.1007/978-981-15-9712-1_31/COVER

Jouppi, N. P., Kurian, G., Li, S., Ma, P., Nagarajan, R., Nai, L., Patil, N., Subramanian, S., Swing, A., Towles, B., Young, C., Zhou, X., Zhou, Z., & Patterson, D. (2023). TPU v4: An Optically Reconfigurable Supercomputer for Machine Learning with Hardware Support for Embeddings. *Proceedings - International Symposium on Computer Architecture*, 23, 1147-1160. <https://doi.org/10.1145/3579371.3589350>

Jurafsky, D. (2016). Language modeling: Introduction to n-grams. *Stanford University Natural Language Processing*.

Jurafsky Daniel, & Martin James H. (2006). *Jurafsky, Martin.-Speech and Language Processing_ An Introduction to Natural Language Processing (2007)*.

Kaiser, L., Brain, G., & Bengio, S. (2016). *Can Active Memory Replace Attention?*

Kaiser, L., & Sutskever, I. (2015). Neural GPUs Learn Algorithms. *4th International Conference on Learning Representations, ICLR 2016 - Conference Track Proceedings*. <https://arxiv.org/abs/1511.08228v3>

Kalchbrenner, N., Espeholt, L., Simonyan, K., Van Den Oord, A., Graves, A., Kavukcuoglu, K., Deepmind, G., & Uk, L. (2016). *Neural Machine Translation in Linear Time*. <https://arxiv.org/abs/1610.10099v2>

Kamal Eddine, M., Shang, G., J-P Tixier, A., Vazirgiannis, M., & Polytechnique, É. (2022). *FrugalScore: Learning Cheaper, Lighter and Faster Evaluation Metrics for Automatic Text Generation (C. 1)*. Long Papers. <https://tinyurl.com/8cwyter2>

Kaplan Jerry. (2024). *Generative Artificial Intelligence What Everyone Needs To Know*.

Karanouh, M., Ca, M., & Program, P. (2023). *Mapping ChatGPT in Mainstream Media to Unravel Jobs and Diversity Challenges: Early Quantitative Insights through Sentiment Analysis and Word Frequency Analysis*. <https://arxiv.org/abs/2305.18340v2>

Kasthuri, E., & Balaji, S. (2023). Natural language processing and deep learning chatbot using long short term memory algorithm. *Materials Today: Proceedings*, 81(2), 690-693. <https://doi.org/10.1016/J.MATPR.2021.04.154>

Khalil, M., & Er, E. (2023). Will ChatGPT Get You Caught? Rethinking of Plagiarism Detection. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 14040 LNCS, 475-487. https://doi.org/10.1007/978-3-031-34411-4_32

Khan, R. A., Jawaid, M., Khan, A. R., & Sajjad, M. (2023). ChatGPT - Reshaping medical education and clinical management. *Pakistan Journal of Medical Sciences*, 39(2), 605-607. <https://doi.org/10.12669/PJMS.39.2.7653>

Kuchaiev, O., & Ginsburg, B. (2017). Factorization tricks for LSTM networks. *5th International Conference on Learning Representations, ICLR 2017 - Workshop Track Proceedings*. <https://arxiv.org/abs/1703.10722v3>

Kudo, T. (2018). Subword Regularization: Improving Neural Network Translation Models with Multiple Subword Candidates. *ACL 2018 - 56th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference (Long Papers)*, 1, 66-75. <https://doi.org/10.18653/V1/P18-1007>

Kudo, T., & Richardson, J. (2018). SentencePiece: A simple and language independent subword tokenizer and detokenizer for Neural Text Processing. *EMNLP 2018 - Conference on Empirical Methods in Natural Language Processing: System Demonstrations, Proceedings*, 66-71. <https://doi.org/10.18653/v1/d18-2012>

Kumar, S., & Tsvetkov, Y. (2020). *End-to-End Differentiable GANs for Text Generation*.

Lai, V. D., Ngo, N. T., Veyseh, A. P. Ben, Man, H., Dernoncourt, F., Bui, T., & Nguyen, T. H. (2023). ChatGPT Beyond English: Towards a Comprehensive Evaluation of Large Language Models in Multilingual Learning. *Findings of the Association for Computational Linguistics: EMNLP 2023*, 13171-13189. <https://doi.org/10.18653/v1/2023.findings-emnlp.878>

Lavie, A., & Agarwal, A. (2007). *Meteor: An Automatic Metric for MT Evaluation with High Levels of Correlation with Human Judgments*. 228-231.

Lavie, A., & Denkowski, M. J. (2009). The METEOR metric for automatic evaluation of Machine Translation. *Machine Translation*, 23(2-3), 105-115. <https://doi.org/10.1007/s10590-009-9059-4>

Lavie, A., Sagae, K., & Jayaraman, S. (2004). The Significance of Recall in Automatic Metrics for MT Evaluation. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 3265, 134-143. https://doi.org/10.1007/978-3-540-30194-3_16

Li, Y., Bubeck, S., Eldan, R., Del Giorno, A., Gunasekar, S., & Lee, Y. T. (2023). *Textbooks Are All You Need II: phi-1.5 technical report*. <http://arxiv.org/abs/2309.05463>

Lin, C.-Y. (2004a). *Looking for a Few Good Metrics: ROUGE and its Evaluation*. <http://duc.nist.gov>

Lin, C.-Y. (2004b). *ROUGE: A Package for Automatic Evaluation of Summaries*.

Lin, T., Wang, Y., Liu, X., & Qiu, X. (2022). A survey of transformers. *AI Open*, 3, 111-132. <https://doi.org/10.1016/J.AIOPEN.2022.10.001>

Liu, F., & Liu, Y. (2008). *Correlation between ROUGE and Human Evaluation of Extractive Meeting Summaries*.

Liu, Q., Kusner, M. J., & Blunsom, P. (2020). *A Survey on Contextual Embeddings*. <https://arxiv.org/abs/2003.07278v2>

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., Stoyanov, V., & Allen, P. G. (2019). *RoBERTa: A Robustly Optimized BERT Pretraining Approach*. <https://arxiv.org/abs/1907.11692v1>

Loper, E., & Bird, S. (2002). NLTK: The Natural Language Toolkit. *COLING/ACL 2006 - 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Interactive Presentation Sessions*, 69-72. <https://arxiv.org/abs/cs/0205028v1>

Lu, Y., Chen, D., Olaniyi, E., & Huang, Y. (2022). Generative adversarial networks (GANs) for image augmentation in agriculture: A systematic review. *Computers and Electronics in Agriculture*, 200. <https://doi.org/10.1016/j.compag.2022.107208>

Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S. J., & McClosky, D. (2014). The Stanford CoreNLP Natural Language Processing Toolkit. *Proceedings of the Annual Meeting of the Association for Computational Linguistics, 2014-June*, 55-60. <https://doi.org/10.3115/V1/P14-5010>

Md. Sanzidul, I., Sadia Sultana, S. M., Abujar, S., & Hossain, S. A. (2019). Sequence-to-sequence Bangla Sentence Generation with LSTM Recurrent Neural Networks. *Procedia Computer Science*, 152, 51-58. <https://doi.org/10.1016/J.PROCS.2019.05.026>

Mesnard, T., Hardin, C., Dadashi, R., Bhupatiraju, S., Pathak, S., Sifre, L., Rivière, M., Kale, M. S., Love, J., Tafti, P., Hussenot, L., Sessa, P. G., Chowdhery, A., Roberts, A., Barua, A., Botev, A., Castro-Ros, A., Slone, A., Héliou, A., ... Kenealy, K. (2024). *Gemma: Open Models Based on Gemini Research and Technology*. <https://arxiv.org/abs/2403.08295v4>

Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient Estimation of Word Representations in Vector Space. *1st International Conference on Learning Representations, ICLR 2013 - Workshop Track Proceedings*. <https://arxiv.org/abs/1301.3781v3>

Milanova, I., Sarvanoska, K., Srbinoski, V., & Gjoreski, H. (2019). Automatic Text Generation in Macedonian Using Recurrent Neural Networks. *Communications in Computer and Information Science*, 1110, 1-12. https://doi.org/10.1007/978-3-030-33110-8_1/TABLES/2

Miller George A. (1995). *WordNet: A Lexical Database for English* George A. Miller.

Mohammadi, M., Al-Fuqaha, A., Sorour, S., & Guizani, M. (2018). Deep learning for IoT big data and streaming analytics: A survey. İçinde *IEEE Communications Surveys and Tutorials* (C. 20, Sayı 4, ss. 2923-2960). Institute of Electrical and Electronics Engineers Inc. <https://doi.org/10.1109/COMST.2018.2844341>

Nedal, D., & Morgan, P. (2018). *Introduction to Machine Learning with Python: A Guide for Beginners in Data Science*.

Novikova, J., Dušek, O., Curry, A. C., & Rieser, V. (2017). Why We Need New Evaluation Metrics for NLG. *EMNLP 2017 - Conference on Empirical Methods in Natural Language Processing, Proceedings*, 2231-2240. <https://doi.org/10.18653/v1/D17-1237>

Omar, M., Choi, S., Nyang, D., & Mohaisen, D. (2022). Robust Natural Language Processing: Recent Advances, Challenges, and Future Directions. *IEEE Access*. <https://doi.org/10.1109/ACCESS.2022.3197769>

Oztaner, S. M. (1996). *A Word Grammar of Turkish with Morphophonemic Rules*. <https://arxiv.org/abs/cmp-lg/9608013v1>

Pandey, A. K., & Roy, S. S. (2023). Natural Language Generation Using Sequential Models: A Survey. *Neural Processing Letters*, 55(6), 7709-7742. <https://doi.org/10.1007/s11063-023-11281-6>

Papineni, K., Roukos, S., Ward, T., & Zhu, W.-J. (2002). Bleu: a Method for Automatic Evaluation of Machine Translation. *Proceedings of the 40th Annual Meeting*

on Association for Computational Linguistics - ACL '02, 311-318.
<https://doi.org/10.3115/1073083.1073135>

Paulus, R., Xiong, C., & Socher, R. (2017). A Deep Reinforced Model for Abstractive Summarization. *6th International Conference on Learning Representations, ICLR 2018 - Conference Track Proceedings*. <https://arxiv.org/abs/1705.04304v3>

Pawade, D., Sakhapara, A., Jain, M., Jain, N., & Gada, K. (2018). Story Scrambler - Automatic Text Generation Using Word Level RNN-LSTM. *International Journal of Information Technology and Computer Science*, 10(6), 44-53.
<https://doi.org/10.5815/IJITCS.2018.06.05>

Perdices, D., Ramos, J., García-Dorado, J. L., González, I., & López de Vergara, J. E. (2021). Natural language processing for web browsing analytics: Challenges, lessons learned, and opportunities. *Computer Networks*, 198, 108357.
<https://doi.org/10.1016/J.COMNET.2021.108357>

Peters, M. E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., & Zettlemoyer, L. (2018). Deep Contextualized Word Representations. *NAACL HLT 2018 - 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies - Proceedings of the Conference*, 1, 2227-2237. <https://doi.org/10.18653/V1/N18-1202>

Pichai Sundar, & Hassabis Demis. (2023). *Introducing Gemini: Google's most capable AI model yet*. <https://blog.google/technology/ai/google-gemini-ai/>

Pillutla, K., Swayamdipta, S., Zellers, R., Thickstun, J., Welleck, S., Choi, Y., & Harchaoui, Z. (2021). MAUVE: Measuring the Gap Between Neural Text and Human Text using Divergence Frontiers. *Advances in Neural Information Processing Systems*, 6, 4816-4828. <https://arxiv.org/abs/2102.01454v3>

Porter, M. F. (1980). An algorithm for suffix stripping. *Program*, 14(3), 130-137.
<https://doi.org/10.1108/EB046814/FULL/PDF>

Powers, D. M. W., & Ailab. (2020). *EVALUATION: FROM PRECISION, RECALL AND F-MEASURE TO ROC, INFORMEDNESS, MARKEDNESS & CORRELATION*.

Qi, P., Zhang, Y., Zhang, Y., Bolton, J., & Manning, C. D. (2020). *Stanza: A Python Natural Language Processing Toolkit for Many Human Languages*. 101-108. <https://spacy.io/>

Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., & Sutskever, I. (2019). *Language Models are Unsupervised Multitask Learners*. <https://github.com/codelucas/newspaper>

Ravichandiran, S. (2021). *Getting started with Google BERT: build and train state-of-the-art natural language processing models using BERT*.

Reiter, E., & Dale, R. (1995). Building Applied Natural Language Generation Systems. *Natural Language Engineering*, 1.

Ren, Y., Hu, W., Wang, Z., Zhang, X., Wang, Y., & Wang, X. (2021). A hybrid deep generative neural model for financial report generation. *Knowledge-Based Systems*, 227, 107093. <https://doi.org/10.1016/J.KNOSYS.2021.107093>

Rıfkı Aydın, C., & Güngör, T. (2021). ARTICLE Sentiment analysis in Turkish: Supervised, semi-supervised, and unsupervised techniques. *Natural Language Engineering*, 27, 455-483. <https://doi.org/10.1017/S1351324920000200>

Rizzo, G., & Van, T. H. M. (2020). Adversarial text generation with context adapted global knowledge and a self-attentive discriminator. *Information Processing and Management*, 57(6). <https://doi.org/10.1016/j.ipm.2020.102217>

Rolf, B., Jackson, I., Müller, M., Lang, S., Reggelin, T., & Ivanov, D. (2023). A review on reinforcement learning algorithms and applications in supply chain management. İçinde *International Journal of Production Research* (C. 61, Sayı 20, ss. 7151-7179). Taylor and Francis Ltd. <https://doi.org/10.1080/00207543.2022.2140221>

Rosamma, K. S. R., & Patil, N. (2023). Measuring the Quality of Text Summarization: A Survey of Evaluation Approaches. *OCIT 2023 - 21st International Conference on Information Technology, Proceedings*, 290-296. <https://doi.org/10.1109/OCIT59427.2023.10431258>

Rudolph, J., Tan, S., & Tan, S. (2023). ChatGPT: Bullshit spewer or the end of traditional assessments in higher education? *Journal of Applied Learning and Teaching*, 6(1), 342-363. <https://doi.org/10.37074/JALT.2023.6.1.9>

Saggion, H., Radev, D., Teufel, S., & Lam, W. (2002). *Meta-evaluation of Summaries in a Cross-lingual Environment using Content-based Metrics*. <https://aclanthology.org/C02-1073>

Sak, H., Güngör, T., & Saraçlar, M. (2008). Turkish Language Resources: Morphological Parser, Morphological Disambiguator and Web Corpus. *GoTAL*, 5221 *LNAI*, 417-427. https://doi.org/10.1007/978-3-540-85287-2_40

Santhanam, S. (2020). *Context based Text-generation using LSTM networks*. <https://arxiv.org/abs/2005.00048v1>

Saravanan, S., & Sudha, K. (2022). GPT-3 Powered System for Content Generation and Transformation. *Proceedings - 2022 5th International Conference on Computational Intelligence and Communication Technologies, CCICT 2022*, 514-519. <https://doi.org/10.1109/CCICT56684.2022.00096>

Saxena, D., & Cao, J. (2021). Generative Adversarial Networks (GANs). *ACM Computing Surveys (CSUR)*, 54(3). <https://doi.org/10.1145/3446374>

Schneider, E. T. R., De Souza, J. V. A., Gumiel, Y. B., Moro, C., & Paraiso, E. C. (2021). A GPT-2 language model for biomedical texts in Portuguese. *Proceedings - IEEE Symposium on Computer-Based Medical Systems, 2021-June*, 474-479. <https://doi.org/10.1109/CBMS52027.2021.00056>

Schuster, M., & Nakajima, K. (2012). Japanese and Korean voice search. *IEEE International Conference on Acoustics, Speech, and Signal Processing*, 5149-5152. <https://doi.org/10.1109/ICASSP.2012.6289079>

Sennrich, R., Haddow, B., & Birch, A. (2015). Neural Machine Translation of Rare Words with Subword Units. *54th Annual Meeting of the Association for Computational Linguistics, ACL 2016 - Long Papers*, 3, 1715-1725. <https://doi.org/10.18653/v1/p16-1162>

Shankland Stephen. (2023). *Google Gemini AI Tries Outsmarting ChatGPT Using Photos and Videos - CNET*. <https://www.cnet.com/tech/computing/google-gemini-ai-tries-outsmarting-chatgpt-using-photos-and-videos/>

Shivahare, B. D., Singh, A. K., Uppal, N., Rizwan, A., Vaathsav, V. S., & Suman, S. (2022). Survey Paper: Study of Natural Language Processing and its Recent

Applications. *Proceedings - 2022 2nd International Conference on Innovative Sustainable Computational Technologies, CISCT 2022*.
<https://doi.org/10.1109/CISCT55310.2022.10046440>

Shobana, J., & Murali, M. (2022). An Improved Self Attention Mechanism Based on Optimized BERT-BiLSTM Model for Accurate Polarity Prediction. *The Computer Journal*, 00(00). <https://doi.org/10.1093/comjnl/bxac013>

Shore, J. E., & Johnson, R. W. (1981). Properties of Cross-Entropy Minimization. *IEEE Transactions on Information Theory*, 27(4), 472-482.
<https://doi.org/10.1109/TIT.1981.1056373>

Sidorov, G., Velasquez, F., Stamatatos, E., Gelbukh, A., & Chanona-Hernández, L. (2014). Syntactic N-grams as machine learning features for natural language processing. *Expert Systems with Applications*, 41(3), 853-860.
<https://doi.org/10.1016/J.ESWA.2013.08.015>

Simunec, M., Soic, R., & Vukovic, M. (2021). *N-gram Based Croatian Language Network*.

Sohail, S. S. (2024). A Promising Start and Not a Panacea: ChatGPT's Early Impact and Potential in Medical Science and Biomedical Engineering Research. *Annals of Biomedical Engineering*, 52(5), 1131-1135. <https://doi.org/10.1007/S10439-023-03335-6/FIGURES/3>

Sonmezoz, K., & Amasyali, M. F. (2021). Same Sentence Prediction: A new Pre-training Task for BERT. *Proceedings - 2021 Innovations in Intelligent Systems and Applications Conference, ASYU 2021*.
<https://doi.org/10.1109/ASYU52992.2021.9598954>

Srivastava, M. (2023). *A Day in the Life of ChatGPT as a researcher: Sustainable and Efficient Machine Learning - A Review of Sparsity Techniques and Future Research Directions*. <https://doi.org/10.31219/OSF.IO/E9P3G>

Strubell, E., Ganesh, A., & McCallum, A. (2019). Energy and Policy Considerations for Deep Learning in NLP. *Journal of International Marketing*, 53(1), 3645-3650. <https://arxiv.org/abs/1906.02243v1>

Susnjak, T. (2022). *ChatGPT: The End of Online Exam Integrity?*
<https://arxiv.org/abs/2212.09292v1>

Sutskever Google, I., Vinyals Google, O., & Le Google, Q. V. (2014). Sequence to Sequence Learning with Neural Networks. *Advances in Neural Information Processing Systems*, 27.

Sutskever, I., Martens, J., & Hinton, G. (2011). *Generating Text with Recurrent Neural Networks*.

Taulli, T. (2023). Bard. *ChatGPT and Bard for Business Automation*, 31-50.
https://doi.org/10.1007/978-1-4842-9852-7_3

Thompson, H. S. (1991). *Automatic Evaluation of Translation Quality: Outline of Methodology and Report on Pilot Experiment*.

Topac, T., & Qamar, S. (2020). Enhanced Face Recognition Using Euclidean Distance Classification and PCA. İçinde *JOURNAL OF INNOVATION IN ELECTRONICS AND COMMUNICATION ENGINEERING* (C. 10, Sayı 1).

Touvron, H., Martin, L., Stone, K., Albert, P., Almahairi, A., Babaei, Y., Bashlykov, N., Batra, S., Bhargava, P., Bhosale, S., Bikel, D., Blecher, L., Ferrer, C. C., Chen, M., Cucurull, G., Esiobu, D., Fernandes, J., Fu, J., Fu, W., ... Scialom, T. (2023). *Llama 2: Open Foundation and Fine-Tuned Chat Models*.

Transformers. (2024). <https://huggingface.co/docs/transformers/index>

Van Rijsbergen, C. (1979). Information retrieval: theory and practice. In *Proceedings of the Joint IBM/University of Newcastle upon Tyne Seminar on Data Base Systems*, 1–14.

Vaswani, A., Brain, G., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is All you Need. *Advances in Neural Information Processing Systems*, 30.

Victoria Lin, X., Mihaylov, T., Artetxe, M., Wang, T., Chen, S., Simig, D., Ott, M., Goyal, N., Bhosale, S., Du, J., Pasunuru, R., Shleifer, S., Singh Koura, P., Chaudhary, V., Wang, J., Zettlemoyer, L., Kozareva, Z., Diab, M., Stoyanov, V., & Li, X. (2022).

Few-shot Learning with Multilingual Generative Language Models.
<https://github.com/facebookresearch/>

Wenzek, G., Lachaux, M. A., Conneau, A., Chaudhary, V., Guzmán, F., Joulin, A., & Grave, E. (2019). CCNet: Extracting High Quality Monolingual Datasets from Web Crawl Data. *LREC 2020 - 12th International Conference on Language Resources and Evaluation, Conference Proceedings*, 4003-4012.
<https://arxiv.org/abs/1911.00359v2>

Wilcox, E. G., Gauthier, J., Hu, J., Qian, P., & Levy, R. (2022). Learning Syntactic Structures from String Input. *Algebraic Structures in Natural Language*, 113-138. <https://doi.org/10.1201/9781003205388-6>

Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., Davison, J., Shleifer, S., von Platen, P., Ma, C., Jernite, Y., Plu, J., Xu, C., Scao, T. Le, Gugger, S., ... Rush, A. M. (2019). *HuggingFace's Transformers: State-of-the-art Natural Language Processing.*
<http://arxiv.org/abs/1910.03771>

Wu, Y., Schuster, M., Chen, Z., Le, Q. V., Norouzi, M., Macherey, W., Krikun, M., Cao, Y., Gao, Q., Macherey, K., Klingner, J., Shah, A., Johnson, M., Liu, X., Kaiser, L., Gouws, S., Kato, Y., Kudo, T., Kazawa, H., ... Dean, J. (2016). *Google's Neural Machine Translation System: Bridging the Gap between Human and Machine Translation.* <https://arxiv.org/abs/1609.08144v2>

Xiang, L., Yang, S., Liu, Y., Li, Q., & Zhu, C. (2020). Novel Linguistic Steganography Based on Character-Level Text Generation. *Mathematics 2020, Vol. 8, Page 1558*, 8(9), 1558. <https://doi.org/10.3390/MATH8091558>

Yan, J., & Wang, X. (2022). Unsupervised and semi-supervised learning: the next frontier in machine learning for plant systems biology. *Içinde Plant Journal*. John Wiley and Sons Inc. <https://doi.org/10.1111/tpj.15905>

Yang, A., Liu, K., Liu, J., Lyu, Y., & Li, S. (2018). *Adaptations of ROUGE and BLEU to Better Evaluate Machine Reading Comprehension Task.*
<http://arxiv.org/abs/1806.03578>

Yin, W., Li, L., & Wu, F. X. (2022). A semi-supervised autoencoder for autism disease diagnosis. *Neurocomputing*, 483, 140-147. <https://doi.org/10.1016/j.neucom.2022.02.017>

Zhan, Z. H., Li, J. Y., & Zhang, J. (2022). Evolutionary deep learning: A survey. İçinde *Neurocomputing* (C. 483, ss. 42-58). Elsevier B.V. <https://doi.org/10.1016/j.neucom.2022.01.099>

Zhang, C., Zhang, C., Song, J., Yi, J. S. K., Zhang, K., & Kweon, I. S. (2022). A Survey on Masked Autoencoder for Self-supervised Learning in Vision and Beyond. <http://arxiv.org/abs/2208.00173>

Zhang, L., Shen, J., Zhang, J., Xu, J., Li, Z., Yao, Y., & Yu, L. (2022). Multimodal Marketing Intent Analysis for Effective Targeted Advertising. *IEEE Transactions on Multimedia*, 24, 1830-1843. <https://doi.org/10.1109/TMM.2021.3073267>

Zhang, T., Kishore, V., Wu, F., Weinberger, K. Q., & Artzi, Y. (2019). BERTScore: Evaluating Text Generation with BERT. *8th International Conference on Learning Representations, ICLR 2020*. <https://arxiv.org/abs/1904.09675v3>

Zhang, X., & Lapata, M. (2014). Chinese Poetry Generation with Recurrent Neural Networks. *EMNLP 2014 - 2014 Conference on Empirical Methods in Natural Language Processing, Proceedings of the Conference*, 670-680. <https://doi.org/10.3115/V1/D14-1074>

Zheng, Q., Yang, M., Yang, J., Zhang, Q., & Zhang, X. (2018). Improvement of Generalization Ability of Deep CNN via Implicit Regularization in Two-Stage Training Process. *IEEE Access*, 6, 15844-15869. <https://doi.org/10.1109/ACCESS.2018.2810849>

Zhou, G., & Lampouras, G. (2020). *WebNLG Challenge 2020: Language Agnostic Delexicalisation for Multilingual RDF-to-text generation* (ss. 186-191). <https://aclanthology.org/2020.webnlg-1.22>

Zhu, J., Li, J., Zhu, M., Qian, L., Zhang, M., & Zhou, G. (2019). Modeling Graph Structure in Transformer for Better AMR-to-Text Generation. *EMNLP-IJCNLP 2019 - 2019 Conference on Empirical Methods in Natural Language Processing and 9th International Joint Conference on Natural Language Processing, Proceedings of the Conference*, 5459-5468. <https://doi.org/10.18653/V1/D19-1548>

ÖZGEÇMİŞ

Tuna Topaç ilk, orta öğrenimini Edirne’de tamamladı. 2008 yılında Kuleli Askeri Lisesi’nden mezun olmuştur. 2018 yılında Trakya Üniversitesi Bilgisayar Mühendisliği’nden mezun olmuştur. 2022 yılında Trakya Üniversitesi Fen Bilimleri Enstitüsü Bilgisayar Mühendisliği yüksek lisans programına başlamıştır. 2023 yılından beri Trakya Üniversitesi Uygulamalı Bilimler Fakültesi Yönetim Bilişim Sistemleri bölümünde araştırma görevlisi olarak görevini icra etmektedir.



EKLER

EK: Hesaplama metrikleri için kullanılan kodlar

BLEU hesaplama metriğinin ölçümünde kullanılan Python kodu

```
import nltk
from nltk.translate.bleu_score import sentence_bleu, SmoothingFunction

def bleu(referans, uretilen):
    referans_belirtec = nltk.word_tokenize(referans.lower())
    uretilen_belirtec = nltk.word_tokenize(uretilen.lower())
    agirlik = [(1.0 / n) for n in range(1, 5)]
    smoothie = SmoothingFunction().method1
    bleu_skor = sentence_bleu([referans_belirtec], uretilen_belirtec,
                              weights=agirlik, smoothing_function=smoothie)
    return bleu_skor
```

METEOR hesaplama metriğinin ölçümünde kullanılan Python kodu

```
import nltk
from nltk.corpus import wordnet
from collections import defaultdict

def get_synonyms(word):
    synonyms = set()
    for syn in wordnet.synsets(word):
        for lemma in syn.lemmas():
            synonyms.add(lemma.name().lower())
    return synonyms

def compute_meteor_score(reference, generated):
    reference_tokens = nltk.word_tokenize(reference.lower())
    generated_tokens = nltk.word_tokenize(generated.lower())
    reference_unigrams = set(reference_tokens)
    generated_unigrams = set(generated_tokens)
    common_unigrams = reference_unigrams.intersection(generated_unigrams)
    precision = len(common_unigrams) / len(generated_unigrams)
    recall = len(common_unigrams) / len(reference_unigrams)
    if precision == 0 and recall == 0: return 0
    alignment_scores = defaultdict(int)
    for token in generated_tokens:
        for synonym in get_synonyms(token):
            alignment_scores[synonym] += 1
    alignment_precision = sum(alignment_scores[token] for token in
                              reference_tokens) / len(generated_tokens)
    alignment_recall = sum(alignment_scores[token] for token in
                           reference_tokens) / len(reference_tokens)

    alpha = 0.5
    f1_score = (1 + alpha) * (precision*recall) / (alpha*precision+recall)
```

```

gamma = 0.5
meteor_score = f1_score * ((1 - gamma) * alignment_precision + gamma *
                             alignment_recall)

return meteor_score

```

ROUGE-N hesaplama metriğinin ölçümünde kullanılan Python kodu

```

import nltk
from nltk.translate.bleu_score import sentence_bleu, SmoothingFunction
from nltk.translate import nist_score, gleu_score

def lcsseq(s1, s2):
    return nltk.util.longest_common_subsequence(s1, s2)

def compute_rouge_score(reference, generated):
    reference_tokens = nltk.word_tokenize(reference.lower())
    generated_tokens = nltk.word_tokenize(generated.lower())
    rouge_n_scores = []
    for n in range(1, 5):
        reference_ngrams = set(nltk.ngrams(reference_tokens, n))
        generated_ngrams = set(nltk.ngrams(generated_tokens, n))
        intersection = len(reference_ngrams.intersection(generated_ngrams))
        rouge_n_scores.append(intersection / len(reference_ngrams))
    return rouge_n_scores

```

ROUGE-L hesaplama metriğinin ölçümünde kullanılan Python kodu

```

import pylcs

def rouge_l(reference, generated):
    reference_tokens = reference
    generated_tokens = generated
    lcs_length = pylcs.lcs_string_length(reference_tokens, generated_tokens)
    precision = lcs_length / len(generated_tokens)
    recall = lcs_length / len(reference_tokens)
    f1_score = 2 * (precision * recall) / (precision + recall + 1e-10)
    return f1_score

```

BERTSCORE hesaplama metriğinin ölçümünde kullanılan Python kodu

```

bertscore = evaluate.load("bertscore")
predictions = [uretilen_metin]
references = [referans_metin]
results =
    bertscore.compute(predictions=predictions, references=references, lang="tr")
print(results)

```