T.C.
Uludağ Üniversitesi
Fen Bilimleri Enstitüsü

DERİN ÖĞRENME İLE BEKLENTİ TABANLI DUYGU ANALİZİ

MELEK TURAN

Yüksek Lisans Tezi

DERİN ÖĞRENME İLE BEKLENTİ TABANLI DUYGU ANALİZİ

MELEK TURAN



T.C.
BURSA ULUDAĞ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

DERİN ÖĞRENME İLE BEKLENTİ TABANLI DUYGU ANALİZİ

Melek TURAN
0000-0002-1873-4980

Doç. Dr. Metin BİLGİN
(Danışman)

Dr. Ali DURAN
(İkinci Danışman)

YÜKSEK LİSANS
BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI

BURSA – 2023

Her Hakkı Saklıdır

ÖZET

Yüksek Lisans Tezi

DERİN ÖĞRENME İLE BEKLENTİ TABANLI DUYGU ANALİZİ

Melek TURAN

Bursa Uludağ Üniversitesi
Fen Bilimleri Enstitüsü
Bilgisayar Mühendisliği Anabilim Dalı

Danışman: Doç. Dr. Metin BİLGİN
İkinci Danışman: Dr. Ali DURAN (Millî Eğitim Bakanlığı)

Bu çalışmada, Türkçe metinlerdeki ,belirli öğeler veya unsurları baz alarak, duygusal ifadelerin otomatik tahminlenmesine dayanmaktadır. Bu tekniğin, marka yönetimi, müşteri hizmetleri, reklamcılık ve diğer birçok alanda önemli bir rol oynaması hedeflenmektedir. Türkçe için yapılan beklenti tabanlı duygu analizi çalışmaları henüz sınırlıdır. Bu durum, Türkçe metinlerdeki duygusal ifadeleri tespit etmek için kullanılan yöntemlerle, Türkçe dilinin özelliklerini dikkate alarak etkili sonuçlar üretmeyi önemli kılmaktadır. Çalışmada, belirteç sınıflandırma ve duygu sınıflandırma olmak üzere iki ayrı yöntem kullanılmıştır. Belirteç sınıflandırma yöntemi için, Türkçe için önceden eğitilmiş BERT, ConvBERT, ELECTRA, DeBERTa, DistilBERT olmak üzere 5 farklı modelin karşılaştırmalı analizi gerçekleştirilmiştir. Belirteç sınıflandırma yöntemi ile metinlerdeki ayırıcı unsurların tespit edilmesi, bu unsurların temsil ettiği kategorilerin belirlenmesi sağlanmıştır. Duygu sınıflandırma için önceden eğitilmiş sıralı sınıflandırma modellerinden BERT, ConvBERT, ELECTRA, DistilBERT olmak üzere 4 farklı model kullanılmıştır. Pozitif, negatif ve nötr sınıflarından oluşan restoran (Pontiki vd., 2016) verileri ile eğitim gerçekleştirilmiştir. İki yöntem bir arada kullanarak tasarlanan arayüzde, girdi olarak verilen metin için ilk olarak, belirteç sınıflandırma ile belirteçler seçilmekte, seçilen belirteçler ait olduğu kategoriye göre sınıflandırılmakta, son olarak kategoriye ait duygu sınıfı sıralı sınıflandırma modelleri aracığı ile sunulmaktadır. Eğitimler sonucunda, modeller F1-skor, kesinlik, duyarlılık, doğruluk metriklerine göre değerlendirilmiştir.

Anahtar Kelimeler: Beklenti Tabanlı Duygu Analizi, BERT, ConvBERT, ELECTRA, DeBERTa, DistilBERT, Sıralı Sınıflandırma, Belirteç Sınıflandırma.
2023,viii+73 sayfa

ABSTRACT

MSc Thesis

ASPECT BASED SENTIMENT ANALYSIS WITH DEEP LEARNING

Melek TURAN

Bursa Uludağ University
Graduate School of Natural and Applied Sciences
Department of Computer Engineering

Supervisor: : Assoc. Prof. Dr. Metin BİLGİN
Second Supervisor: Dr. Ali DURAN (Ministry of National Education)

This study is based on the automatic prediction of emotional expressions in Turkish texts, focusing on specific elements or aspects. This technique is expected to play an important role in branding, customer service, advertising and various other fields. There is a limited amount of research into Aspect Based Sentiment Analyzing (ABSA) in Turkish. This highlights the importance of the development of effective methods that take into account the characteristics of the Turkish language for the detection of emotional expressions in Turkish texts.

This study used two methods, token classification and sentiment analysis.The token classification was carried out by using five different models: BERT, ConvBERT, ELECTRA, DeBERTa, and DistilBERT, which were pre-trained for Turkish. The aim of this method is the identification of differentiating elements in the texts and the determination of the categories they represent.. For sentiment analysis, four pre-trained sequential classification models—BERT, ConvBERT, ELECTRA, and DistilBERT—were used. The models were trained using restaurant data (Pontiki et al., 2016) consisting of positive, negative, and neutral classes. In the interface designed by combining these two methods, the input text was first analyzed using aspect-based sentiment classification to select relevant aspects. These aspects were then classified based on their corresponding categories, and the sentiment class specific to each category was determined using sequential classification models. The models were evaluated based on metrics such as F1-score, precision, recall, and accuracy.

Anahtar Kelimeler: Aspect Based Sentiment Analysis, BERT, ConvBERT, ELECTRA, DeBERTa, DistilBERT, Sequantial Classification, Token Classification.
2023,vii+73 page

TEŞEKKÜR

Tez yazım sürecinde, yol gösterici önerileri ve ilham verici tutumu sayesinde beni destekleyen, her konuda kolaylık sağlayarak çalışma sürecimi hızlandıran, karşılaştığım zorluklarda içtenlikle davranarak motivasyon sağlayan, değerli danışman hocalarım Doç. Dr. Metin BİLGİN ve Dr. Ali DURAN'a sonsuz teşekkür ederim.

Çalışmalarım boyunca her zaman yanımda olan, destekleyen, güvenen, maddi, manevi her konuda arkamda duran aileme sonsuz teşekkür ederim.

Bu tez çalışmasının gerçekleştirilmesi için her anlamda gerekli desteği sağlayan Özdilek Ev Tekstil San. ve Tic. AŞ Özveri Ar-Ge Merkezi merkezindeki yöneticilerimize ve tüm şirket çalışanlarına, çok teşekkürler ederim.

Melek TURAN
../…./…..

İÇİNDEKİLER

Sayfa

ÖZET……..i
ABSTRACT……..ii
TEŞEKKÜR…….iii
KISALTMALAR DİZİNİ……..…….vi
ŞEKİLLER DİZİNİ…….vii
ÇİZELGELER DİZİNİ……...…….…...…viii
1.GİRİŞ……..1
2. KAVRAMSAL ÇERÇEVE ……..…….……...2
2.1. Yapay Zekâ……...……..2
2.2. Makine Öğrenimi ……..……4
2.3. Duygu Analizi ……..……..6
2.4. ABSA Çalışmaları ……..…..…..7
2.4.1 Türkçe için ABSA çalışmaları……...…8
2.4.2.Restaurant Veri Seti Üzerinde Gerçekleştirilen Bazı ABSA çalışmaları……...…10
3. MATERYAL VE YÖNTEM……..……13
3.1 Transformatör Mimarisi…….13
3.2. Çalışmada kullanılan modeller……..17
3.2.1. BERT……..... 18
3.2.2. DistilBERT…….... .22
3.2.3 ConvBERT……..23
3.2.4. ELECTRA…… ……24
3.2.5. DeBERTa……...……25
3.2.6. Türkçe için BERTurk Modeli…….……26
3.2.7. Türkçe için DistilBERTurk modeli……..……..……26
3.2.8. Türkçe için ELECTRA modeli…… 26
3.2.9. Türkçe için ConvBERTurk modeli…….…26
3.3. Belirteç Sınıflandırması……...…….……27
3.4. Sıralı Sınıflandırma Modeli…….……..29
3.5. Kullanılan Kütüphaneler……...30
3.5.1. PyTorch……..30
3.5.2. Numpy……... 32
3.5.3. Transformers……..33
3.5.4. Sklearn ...……35
3.5.5. NLTK…… 38
3.5.6. PyQt5 ……..……...……...…39
3.6. Kullanılan Hiperparametreler……41
3.7. Deneysel Çalışmalar……..44
3.7.1. Veri kümesi……. 45
3.7.2. Belirteç sınıflandırma ile kategori tespiti……47

3.7.3. Sıralı sınıflandırma modeli ile duygu analizi……49
3.7.4. Hedef terim, kategori ve duygunun aynı birlikte tespit edilmesi……51
4.BULGULAR…… ……....53
4.1. Belirteç Sınıflandırma ile Kategori Tespitine ait Bulgular……53
4.2. BERT ile Sıralı Sınıflandırma için Elde Edilen Bulgular……..54
4.3. ConvBERT ile Sıralı Sınıflandırma için Elde Edilen Bulgular…….55
4.4. ELECTRA ile Sıralı Sınıflandırma için Elde Edilen Bulgular……..57
4.5. DistilBERT ile Sıralı Sınıflandırma için Elde Edilen Bulgular……..58
4.6. Sonuçlar ……..……...59
5. TARTIŞMA ve SONUÇ……...……...……63
5.1. Gelecek Çalışmalara Yönelik Öneriler……..…64
KAYNAKLAR……66
ÖZGEÇMİŞ…….73

KISALTMALAR DİZİNİ

Kısaltmalar	Açıklama
FBFE	Frekans Tabanlı Özellik Çıkarımı (Frequency Based Feature Extraction)
FBFE-SWS	Duygu Kelimesi Desteği ile Frekans Tabanlı Özellik Çıkarımı (Frequency Based Feature Extraction with Sentiment Word Support)
WSBFE	Web Arama Tabanlı Özellik Çıkarma (Web Search Based Feature Extraction)
ABSA	Beklenti Tabanlı Duygu Analizi (Aspect Based Sentiment Analysis)
MWA	Çok Kelimeli Boyut Çıkarma (Multi-Word Aspect Extraction)
CDF	Koşullu Rastgele Alanlar (Conditional Random Fields)
NLP	Doğal Dil İşleme (Natural Language Processing)
TF-IDF	Dönem Sıklığı-Ters Belge Sıklığı (Term Frequency-Inverse Document Frequency)
LDA	Gizli Dirichlet Tahsisi(Latent Dirichlet Allocation)
NMF	Negatif Olmayan Matris Faktorizasyon (Non-Negative Matrix Factorization)
RNN	Tekrarlayan Sinir Ağı (Recurrent Neural Network)
AEN	Dikkat Kodlayıcı Ağı (Attentional Encoder Network)
DeBERT	Çözülmüş Dikkat Geliştirilmiş BERT ile Kod Çözme (Decoding with Disentangled Attention-Improved BERT)
DPL	Çift Parçalı Sözde Etiketleme (Dual-Granularity Pseudo Labeling)
HAABSA	Beklenti Tabanlı Duygu Analizi için Hibrit Bir Yaklaşım (A Hybrid Approach for Aspect-Based Sentiment Analysis)
MLM	Maskeli Dil Modelleme (Masked Language Modeling)
NSP	Sonraki Cümle Tahmini (Next Sentence Prediction)
BERT	Transformatörlerden Çift Yönlü Kodlayıcı Gösterimleri (Bidirectional Encoder Representations from Transformers)
DistilBERT	Damıtılmış BERT (Distilled BERT)
ConvBERT	Evrişimli BERT (Convolutional BERT)
ELECTRA	Belirteç Değişimlerini Doğru Bir Şekilde Sınıflandıran Bir Kodlayıcıyı Verimli Bir Şekilde Öğrenmek(Efficiently Learning an Encoder that Classifies Token Replacements Accurately)
DeBERTa	Çözülmüş dikkatle kod çözme özelliği geliştirilmiş BERT (Decoding-enhanced BERT with disentangled attention)
SGD	Stokastik Gradyan İnişi (Stochastic Gradient Descent)
AI	Yapay Zekâ (Artificial Intelligent)

ŞEKİLLER DİZİNİ

Sayfa

Şekil 2.1.	Yapay zekâ yaklaşımlarındaki temel değişimler …..……	2
Şekil 2.2.	Turing testi çalışma prensibi…….	3
Şekil 2.3.	ABSA çalışmalarına genel bakış…….….	7
Şekil 3.1.	Transformatör mimarisi……	14
Şekil 3.2.	Çok başlı dikkat mekanizması mimarisi……	15
Şekil 3.3.	Çalışmada kullanılan modellerin kronolojisi…….	17
Şekil 3.4.	BERT modellerinde tranformatör kullanımı…….	18
Şekil 3.5.	BERT modelinin arama sorgusu için kullanımı……..……...	19
Şekil 3.6.	BERT MLM çalışma prensibi…….……..	20
Şekil 3.7.	BERT NSP çalışma prensibi…….……...…….	21
Şekil 3.8.	Distilbert ve BERT modellerinde transformatör katmanının kullanımı.	22
Şekil 3.9.	Skalalı nokta çarpımı öz dikkat ve aralık tabanlı dinamik konvolüsyon	23
Şekil 3.10.	Belirteç sınıflandırma modellerinin çalışma prensibi..……..	28
Şekil 3.11.	Sıralı sınıflandırma modellerinin çalışma prensibi……	29
Şekil 3.12.	Torch kütüphanesinin kullanımı…….……..	31
Şekil 3.13.	Numpy kütüphanesinin kullanımı……..……...	32
Şekil 3.14.	Transformers kütüphanesinin kullanımı……..……..	34
Şekil 3.15.	Sklearn kütüphanesinin kullanımı……..……...	35
Şekil 3.16.	Karmaşıklık matrisinin kullanımı……..	37
Şekil 3.17.	NLTK kütüphanesinin kullanımı……..	39
Şekil 3.18.	Arayüz tasarımı…….	40
Şekil 3.19.	Veri setine ait örnek……...	45
Şekil 3.20.	Duygularına göre cümle sayısı……..	46
Şekil 3.21.	Cümlelerin içerdikleri benzersiz kelime sayıları……...	46
Şekil 3.22.	Veri setinde en çok bulunan terimler (unigram)……	47
Şekil 3.23.	Belirteç sınıflandırma modeli için girdi ve çıktı……	48
Şekil 3.24.	Hedef terim, kategori ve duyguların aynı anda tespit edilmesi…….	51
Şekil 4.1.	BERT kayıp puanı- adım sayısı…….	54
Şekil 4.2.	BERT modeli için karmaşıklık matrisi……..	54
Şekil 4.3.	ConvBERT kayıp puanı- adım sayısı……	55
Şekil 4.4.	ConvBERT modeli için karmaşıklık matrisi…….	56
Şekil 4.5.	ELECTRA kayıp puanı- adım sayısı……	57
Şekil 4.6.	ELECTRA modeli için karmaşıklık matrisi…….	57
Şekil 4.7.	DistilBERT kayıp puanı- adım sayısı……...	58
Şekil 4.8.	DistilBERT modeli için karmaşıklık matrisi…….	59
Şekil 4.9.	Belirteç sınıflandırma modellerin doğruluk puanlarının karşılaştırılması…….	60
Şekil 4.10.	Sıralı sınıflandırma modellerine ait metriklerim karşılaştırılması…….	61
Şekil 4.11.	Sıralı sınıflandırma modellerinin duygu sınıfları için F1 skor karşılaştırması……...	62

ÇİZELGELER DİZİNİ

Sayfa

Çizelge 3.1.	Eğitim ve test verisinin kategorilere göre sayısal dağılımı……...	45
Çizelge 3.2.	Belirteç sınıflandırma etiketlenmiş cümle örneği…….	48
Çizelge 3.3.	Eğitimde kullanılan hiper parametrelerin değerleri……	50
Çizelge 4.1.	Belirteç sınıflandırma modellerinin metriklere göre değerlendirilmesi…….	53
Çizelge 4.2.	BERT modelinin duygu sınıfları üzerinde değerlendirilmesi……..	55
Çizelge 4.3.	ConvBERT modelinin duygu sınıfları üzerinde değerlendirilmesi..	56
Çizelge 4.4.	ELECTRA modelinin duygu sınıfları üzerinde değerlendirilmesi..	58
Çizelge 4.5.	DistilBERT modelinin duygu sınıfları üzerinde değerlendirilmesi.	59

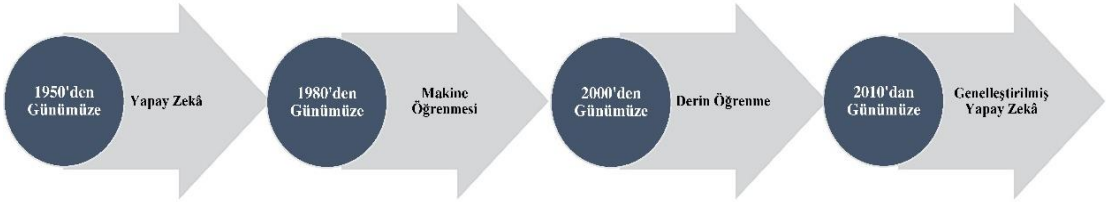
1. GİRİŞ

Günümüzde bireyler, dijital kanallar aracılığıyla, özellikle internet ve sosyal medya platformlarında, günlük olarak yüzlerce hatta binlerce metin mesajı, tweet, yorum ve inceleme gibi yazılı veriye maruz kalmaktadır (Singh vd., 2022). Statista 2023 verilerine göre dünya genelinde 5,3 milyar internet kullanıcısı vardır ve bu kullanıcıların 4.95 milyarı sosyal medya kullanmaktadır. Bu oran toplam dünya nüfusunun %61,4'üne karşılık gelmektedir (Petrosyan, 2023). Sosyal medyada yazılı verilerin büyük bir kısmı, bireylerin görüşlerini, duygularını ve deneyimlerini yansıtan değerlendirmeler ve incelemelerden oluşmaktadır . Bu yazılı verilerin doğru bir şekilde anlaşılması ve analiz edilmesi, marka yönetimi, müşteri hizmetleri, kamusal politika oluşturma gibi birçok alanda kritik bir öneme sahiptir (Calude, 2023). Bu alanda, duygu analizi ve özellikle Beklenti Tabanlı Duygu Analizi (Aspect-Based Sentiment Analysis [ABSA]) son yıllarda oldukça popüler hale gelmiştir. ABSA, bir metnin belirli öğeleri veya unsurları (ürün, hizmet, konu vb.) hakkındaki duygu ifadelerini tespit etmek ve analiz etmek için kullanılan bir doğal dil işleme tekniğidir (Yang vd., 2019). Bu teknik, birçok uygulamada kullanılabilmekte ve özellikle ürün veya hizmet değerlendirmeleri, sosyal medya analizi, pazar araştırması, kullanıcı deneyimi ve marka yönetimi gibi alanlarda etkili sonuçlar vermektedir.

ABSA, klasik duygu analizi ve kelime sınıflandırma yöntemlerinden ayırt edici özelliklere sahiptir. Geleneksel yaklaşımlar, genellikle bir cümlede tek bir duygu ya da sınıflandırma üzerine odaklanırken, ABSA bir cümlede birden çok farklı sınıf ve duyguyu tespit edebilme kapasitesine sahiptir. Bu özellik, özellikle sosyal medya gönderileri, haber metinleri ve müşteri yorumları gibi heterojen veri kaynaklarında karşılaşılan çeşitlilikteki duyguların analizi için son derece değerlidir. Örneğin, tek bir cümle içinde belirli bir hedef kelime için olumlu bir duygu ifade edilirken, başka bir kelime için olumsuz bir duygu ifade edilebilir. Geleneksel yöntemlerin bu tür karışık duygu yapılarıyla başa çıkma konusunda sınırlılıkları bulunmaktadır. ABSA, bu zorluğu aşmak amacıyla tasarlanmıştır. Bu analiz, cümledeki kelimelerin semantik bağlamlarını dikkate alarak, bir cümlede var olan çoklu duygusal tonları doğru bir şekilde belirleme yeteneğine sahiptir.

2. KAVRAMSAL ÇERÇEVE

Bu bölümde, yapay zekâ, makine öğrenimi, duygu analizi ve ABSA çalışmaları üzerinde yapılan araştırmalar değerlendirilmiştir.



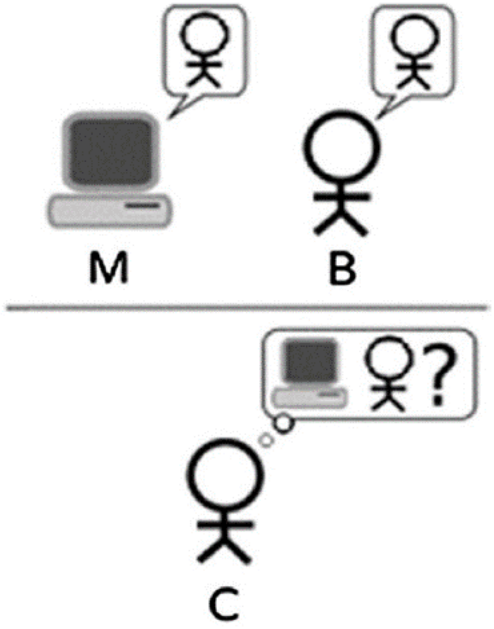
Şekil 2.1. Yapay zekâ yaklaşımlarındaki temel değişimler (Sağıroğlu, 2020, s. 14).

Şekil 2.1'de görüldüğü üzere yapay zekâ çalışmaları, farklı yıllarda değişik konulara yoğunlaşmaktadır. Bu çalışmalar, yapay zekanın ilerlemesine, yeni algoritmalar, yapılar, mimariler, platformlar, öğrenme algoritmaları, uygulamalar ve donanımların geliştirilmesine önemli katkılarda bulunmuştur (Sağıroğlu, 2020). Yapay zekâ makine öğrenimi doğal dil işleme, konuşma, uzman sistemler, planlama optimizasyon, robotik, görü gibi pek çok alanı kapsamaktadır. Bu çalışmada yapay zekânın alt dalı olan doğal dil işlemenin, sınıflandırma ve kümeleme alanında çalışılmış ve duygu sınıflandırma, kategori sınıflandırma konuları incelenmiştir.

2.1. Yapay Zekâ

Yapay zekâ (Artificial Intelligence [AI]) zengin bir geçmişe sahiptir ve genellikle insan zekâsı gerektiren görevleri yerine getirebilen makinelerin geliştirilmesi olarak tanımlanabilir (Poole ve Mackworth, 2010). Yapay zekâ (YZ) kavramı bilgisayar bilimi, felsefe, beşeri bilimler, istatistik, matematik ve sosyal bilimler dahil olmak üzere çeşitli disiplinleri kapsamaktadır (Tariq vd., 2021). Bir diğer tanımda YZ, insanlarda, hayvanlarda ve makinelerde zeki davranışın ne olduğunu inceleyen ve insan yapımı aygıtların nasıl bu tip davranışlar sergileyebileceğini arayan bir bilim dalıdır (Whitby, 2005). Bilişsel işlevler açısından insan davranışını taklit eden makineleri içermektedir (Tariq vd., 2021).

YZ tarihi, 20. yüzyılın ortalarına, araştırmacıların makineleri insan zekâsını taklit edebilecek şekilde programlama fikrini keşfetmeye başladığı döneme dayanmaktadır. Yapay zekâ felsefesini ilk ortaya çıkaran kişi ünlü İngiliz Alan Turing'dir (Pirim 2006).“YZ'nin nihai hedefi nedir?” sorusuna verilen en ünlü cevap, kuşkusuz Turing testi olarak bilinen testtir (Whitby, 2005). Dartmouth konferansından altı yıl önce, 1950 yılında Turing, Mind adlı dergide “Bilgi İşlem Makineleri ve Zekâ (Computing Machinery and Intelligence)” adlı bir makale yayınlamıştır. Bu makalede Turing “Makineler düşünebilir mi?” sorusunu tartışmış ve aksi fikirleri reddetmiştir (Pirim, 2006).



Şekil 2.2. Turing testi çalışma prensibi (Damassino, 2020)

Turing testi çalışma prensibinde A, B ve C olmak üzere gözlemci, insan ve makine bulunur. Turing testinin amacı, Şekil 2.2'de gösterilen gözlemci C, insan B ve makine A ile yaptığı yazılı sohbetlere dayanarak, hangi katılımcının insan olduğunu belirlemeye çalışır. C'nin görevi, doğru tahminde bulunmaktır. Turing testi başarılı kabul edilirse, makine A, insan B gibi düşünme yetisine sahip olduğu kabul edilir. Ancak, Turing testi sadece bir gözlemcinin kısa bir süreliğine bir makine ve bir insanla yazılı sohbet yoluyla yapabileceği bir değerlendirmedir (Damassino, 2020; Levesque, 2017; Shieber, 2004).

YZ kavramı ise, 1956 yılında Dartmouth Konferansı'nda, makinelerin insan benzeri zekâya sahip olabileceği potansiyelini tartışan bir grup bilim insanı tarafından

kullanılmıştır (Poole ve Mackworth, 2010). Bu başlangıçtan günümüze kadar, artan bilgi işlem kapasitesi, gelişen algoritmalar ve daha fazla veriye erişim sayesinde yapay zekâ, önemli ilerlemeler kaydetmiştir.

İlk yapay zekâ yaklaşımları, sembolik akıl yürütme ve mantık temelli sistemlere dayalı olarak tasarlanmıştır ve makinelerin insan benzeri düşünme yeteneklerini taklit edebilmeleri hedeflenmiştir. Ancak, bu klasik yapay zekâ yöntemleri, belirsizlik ve karmaşıklıkla başa çıkabilme konusunda yetersiz kalmışlardır. Daha yakın zamanlarda, algoritmalı yaklaşımların, büyük veri setleri üzerinde model eğitimi yaparak kalıpları tanıma ve tahminlerde bulunma kapasitesi sayesinde makine öğrenimi ve derin öğrenme teknikleri öne çıkmıştır. Bu teknikler, klasik yöntemlerin karşılaştığı zorlukları aşarak yapay zekânın uygulama alanlarını genişletmiştir (Poole ve Mackworth, 2010).

YZ tıp, operasyonel yönetim ve inşaat dahil olmak üzere çeşitli alanlarda uygulama alanı bulmuştur. Tıpta YZ teşhis, tedavi planlaması ve ilaç keşfi için kullanılmaktadır (Tariq vd., 2021). Operasyonel yönetimde yapay zekâ, süreçleri optimize etmek, verimliliği artırmak ve karar almayı geliştirmek için kullanılmıştır (Tariq vd., 2021). İnşaat sektöründe maliyet kontrol analizi ve karar verme için yapay zekâdan yararlanılmıştır (Xie vd., 2022).

Sonuç olarak, YZ makinelerin insan benzeri zekâyla görevleri yerine getirebilmesi için çok disiplinli bir araştırma alanıdır. Bu alanın kökleri 20. yüzyılın ortalarına dayanmakta olup, başlangıçta sembolik akıl yürütme temelli yaklaşımlardan başlamış, günümüzde ise makine öğrenimi ve derin öğrenme yöntemlerine evrilmiştir. YZ, pek çok sektörde yenilikçi bakış açıları ile devrim yaratarak uygulama olanakları sunmuş ve teknolojik ilerlemenin sınırlarını zorlayarak sürekli olarak gelişimini sürdürmektedir.

2.2. Makine Öğrenimi

YZ, verilerden öğreniş ya da öğrenmesin zeki algoritmalarla uğraşan kapsayıcı bir bilimdir. Buna karşın Yapay Öğrenme, YZ'nin alt dalıdır ve verilerden öğrenen algoritmalarla dayalıdır (Husain, 2019). Makine Öğrenimi (Machine Learning [ML]),

bilgisayarların kendi başlarına düşünmesini ve öğrenmesini sağlayan bir yapay zekâ kategorisidir (Alzubi vd., 2018).

Makine öğreniminin geçmişi 20. yüzyılın ortalarına kadar uzanabilir. 1943 yılında McCulloch ve Pitts, makine öğrenimi algoritmalarının geliştirilmesinin temelini oluşturan sinir ağlarının ilk matematiksel modelini önermişlerdir. 1955'te "yapay zekâ" terimi McCarthy vd. tarafından icat edilmiş ve birkaç yıl sonra, 1959'da "makine öğrenimi" terimi Samuel tarafından popüler hale getirilmiştir (Siriborvornratanakul, 2022).

Makine öğrenimi birkaç kategoriye ayrılabilir. Örneğin "gözetimli" öğrenmede bir algoritma önceden belirlenip etiketlenmiş örnekler dizisiyle sunulur ve böylece veriler ve etiketlenmiş sonuçlar arasında bağ kurma ya da sınıflandırma avantajı olur. "Gözetimsiz" öğrenmede ise önceden belirlenmiş ve etiketlenmiş örnekler yoktur sadece etiketlenmemiş ve sınıflandırılmamış veriler vardır. Örneğin her birinde "kedi" ya da "köpek" yazan birçok kedi ve köpek resmi sunulduktan sonra daha sonra karşısına çıkacak kedi ve köpekleri ayırt etme becerisini öğrenmek denetlenen bir öğrenmedir. Buna karşın, denetlenmeyen öğrenmede amaç üstlerinde etiket olmayan geniş bir kedi ve köpek koleksiyonu verildiğinde farkı ayırt etmeyi öğrenmektir (Husain, 2019).

Son yıllarda makine öğreniminde istatistiksel ve olasılıksal yaklaşımlara doğru bir değişim yaşanmış, bu değişime bilgi işlem gücündeki ilerlemeler, büyük veri kümelerinin kullanılabilirliği ve daha karmaşık algoritmaların geliştirilmesi yol açmıştır (Fernández ve Peters 2023).

Makine öğrenimindeki önemli gelişmelerden biri, yapay sinir ağlarının çok katmanlı eğitimine odaklanan bir alt alan olan derin öğrenmenin ortaya çıkması olup, derin öğrenme, bilgisayarlı görme, doğal dil işleme ve konuşma tanıma dahil olmak üzere çeşitli alanlarda dikkate değer bir başarı elde etmiştir (Fernández ve Peters 2023).

Sonuç olarak, derin öğrenme modellerinin verilerin hiyerarşik temsillerini otomatik olarak öğrenebilmelerinin görüntü sınıflandırma, nesne algılama ve dil çevirisi gibi

görevlerde performansın iyileşmesine fayda sağlayacağı ileri sürülebilir.

Makine öğrenimi sağlık, finans, ulaşım ve pazarlama dahil olmak üzere birçok alanda uygulama alanı bulmuş, sağlık hizmetlerinde hastalık teşhisi, tedavi planlaması ve ilaç keşfi için makine öğrenimi algoritmaları kullanılmaktadır (Fernández ve Peters 2023).

Finansta, dolandırıcılık tespiti (Selimoglu ve Yılmaz, 2021), risk değerlendirmesi (Parçali ve Mutlu, 2022) ve algoritmik ticaret (Aylak & Okan, 2021) için makine öğrenimi kullanılmakta, ulaşımda (Sağbaş ve Ballı 2016), otonom sürüş (Dogan vd., 2011) ve trafik tahmini (Rzeszótko ve Nguyen, 2012) için makine öğreniminden yararlanılmaktadır. Pazarlamada müşteri segmentasyonu (Monil vd., 2020) ve kişiselleştirilmiş reklamcılık (Shatnawi ve Mohamed, 2012) için makine öğrenimi algoritmalarından faydalanılmaktadır.

Sonuç olarak, makine öğrenimi, YZ alanının bir alt dalıdır ve bilgisayar sistemlerinin verileri analiz ederek örüntüler ve bilgi çıkarabilmesine izin veren bir teknoloji ve matematiksel bir yaklaşım olarak değerlendirilebilir. Makine öğrenimi, belirli görevleri veya sorunları çözmek için kullanılan algoritmaların geliştirilmesini ve eğitilmesini içerdiği gibi, bu algoritmalar, verilerden öğrenen ve gelecekteki veriler üzerinde tahminlerde bulunabilen, sınıflandırma yapabilen veya kararlar alabilen yapılar olarak özetlenebilir.

2.3. Duygu Analizi

Fikir madenciliği olarak da bilinen duygu analizi, incelemeler, sosyal medya gönderileri veya müşteri geri bildirimleri gibi bir metin parçasında ifade edilen duyguyu veya duygusal tonu analiz etmeyi ve belirlemeyi içeren bir çalışma alanıdır. Amaç, ifade edilen duygunun olumlu, olumsuz veya tarafsız olup olmadığını anlamak ve sınıflandırmaktır.

Türkçe ve diğer diller dahil olmak üzere çeşitli diller için duygu analizine yönelik çalışmalar yapılmıştır (Dehkharghani vd., 2016). Araştırmacılar bu dillere özgü duygu analizi modelleri geliştirmek için farklı yaklaşım ve teknikleri araştırmışlardır.

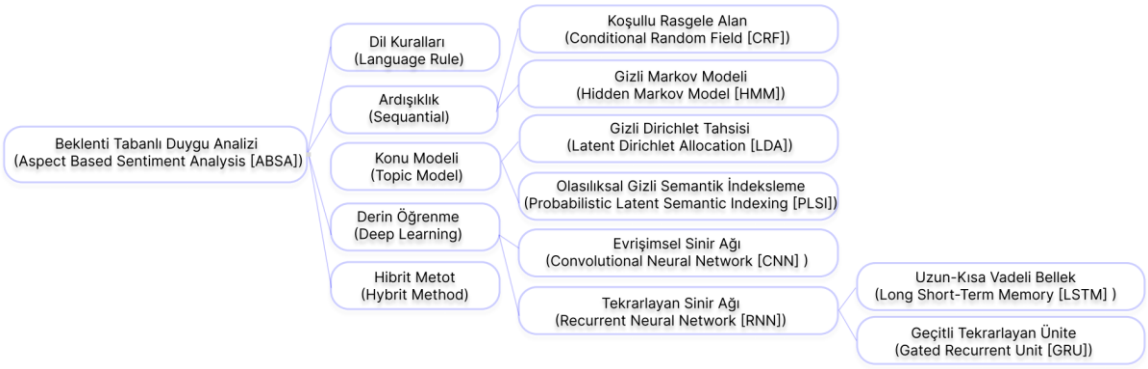
Örneğin, Türkçe duygu analizi üzerine yapılan bir çalışmada (Dehkharghani vd., 2016), araştırmacılar orijinal olarak İngilizce duygu analizi için tasarlanan SentiStrength aracını kutupsallık sözlüğünü Türkçeye çevirerek özelleştirmişlerdir. Bu onlara Türkçe metin belgelerindeki duyguyu analiz etme olanağı sağlamıştır.

Benzer şekilde Kabakus tarafından yapılan başka bir çalışma, evrişimsel sinir ağları ve çift yönlü uzun kısa süreli belleğin birleşimine dayanan Türkçe duygu analizine odaklanmıştır (Kabakus, 2022).

Özetle, duygu analizi çalışmaları farklı diller için genellikle, mevcut araçların ve yöntemlerin belirli bir dile özgü adaptasyonlarını, dil-bağımsız modellemeyi veya transfer öğrenme yöntemlerini kullanarak diğer dillerden bilgi transferini kapsadığı görülmektedir. Bu çalışmaların temel amacının, müşteri geri bildirimleri, sosyal medya takibi ve pazar analizi gibi sektörlerde etkin uygulamaların geliştirilmesini destekleyerek, farklı dillerdeki metinlerde duygu ve tonun doğru bir şekilde analiz edilmesini ve anlaşılmasını sağlamak olduğu söylenebilir.

2.4. ABSA Çalışmaları

Bu bölümde Türkçe için ABSA çalışmaları ve bu çalışmada da kullanılan restoran veri seti (Pontiki vd., 2016, Pontiki vd., 2014), üzerinde ABSA çalışmaları iki ayrı başlıkta incelenecektir.



Şekil 2.3. ABSA çalışmalarına genel bakış (Madhoushi, 2019).

ABSA çalışmaları, dil kuralları, ardışıklık, konu modeli, derin öğrenme ve hibrit metodları içerir. Bu metotların alt dalları Şekil 2.3'de gösterilmektedir. Gösterilen ardışıklık metodu, koşullu rastgele alan (Conditional Random Field [CDF]) ve gizli markov modeli (Hidden Markov Model [HMM]) kapsarken; konu modeli (Topic Model), gizli dirichlet tahsisi (Latent Dirilechlet Allocation [LDA]) ve olasılıksal gizli semantik indekslemeyi (Probabilisitic Latent Semantic Indexing[PLSI]) kapsar. Derin öğrenme ise evrişimsel sinir ağları (Convolutional Neutral Network [CNN]) ve tekrarlayan sinir ağlarını (Reccurrent Neutral Network [RNN]) kapsar. Tekrarlayan sinir ağları, uzun-kısa vadeli bellek (Long Short-Term Memory [LSTM]) ve geçitli tekrarlayan ünite (Gated Recurrent Unit [GRU]) gibi daha özel görevler için özelleştirilmiş yöntemleri barındırmaktadır.

2.4.1 Türkçe için ABSA çalışmaları

Bu bölümde Türkçe metinlerde ABSA çalışmaları incelenecektir.
Kama vd., (2016), formal olmayan metin belgelerinden özniteliklerin çıkarılması için üç adımlı bir yaklaşım önerilmiştir. Bunlar, frekans tabanlı özellik çıkarımı (Frequency Based Feature Extraction [FBFE]), duygu sözcüğü desteği ile birlikte frekans bazlı özellik çıkarımı (Frequency Based Feature Extraction with Sentiment Word Support [FBFE-SWS]) ve web arama tabanlı özellik çıkarımıdır (Web Search Based Feature Extraction [WSBFE]). Bu yöntemde veri setindeki isimlerin ve isim gruplarının frekansları yerine Web'deki frekansları kullanılmaktadır. Kelimelerin hangi kelimelerle birlikte Web'te kullanıldığı analiz edilerek isim grupları tespit edilmiştir. Oluşan bu isim grupları hedef terim çıkarımında kullanılmıştır. Web aracılığıyla cep telefonu modeli için kullanıcı incelemeleri ve yorumlarından oluşan bir veri seti üzerinde çalışılmış, web arama tabanlı özellik çıkarımı (Web Search Based Feature Extraction [WSBFE]) için %69.79 F1-skor, %59,24 kesinlik değeri elde edilmiştir (Kama vd., 2016).

Türkmen vd. (2016), otel yorumları üzerinde çalışmış ABSA tabanlı bir yöntem önermişlerdir. Web'ten toplanan yorumların ön işlenmesinin ardından, puanlama algoritması önererek yönlerin duyarlılık puanının belirlemişlerdir (Türkmen vd., 2016).

Kama vd. (2017), otel yorumları üzerinde çalışmış ABSA tabanlı bir yöntem önermişlerdir. Web'ten toplanan yorumların ön işlenmesinin ardından, puanlama algoritması önererek yönlerin duyarlılık puanının belirlemişlerdir (Kama vd., 2017).

Ekinci vd. (2017), farklı alanlardan ürünlerin Türkçe yorumlarını kullanarak ikili ifadeli özelliklerin otomatik olarak keşfedilmesini sağlayan bir sistem ortaya koymuşlardır. Bu çalışmada, farklı alanlardan ürünlerin Türkçe yorumlarını kullanarak ikili ifadeli özelliklerin otomatik olarak keşfedilmesini sağlayan bir sistem ortaya konmuştur. Sistem, veri setinin düzenlenmesi, n-gram frekanslarının belirlenmesi, n-gram frekanslarının filtrelenmesi, sezgisel kural tabanlı n-gramları, çoklu kelime yönlerinin oluşturulması. Müşteri yorumlarından MWA çıkarımına odaklanılmıştır. %82 kesinlik, %83 duyarlılık puanı elde edilmiştir (Ekinci vd., 2017).

Çetin ve Eryiğit (2018), hedef kategori ve hedef terimin çıkartılması üzerinde çalışmışlardır. İkisinin birlikte tanımlanabilmesi için kelime vektörleri, sözcük analiz bilgilerini baz alan koşullu rasgele alanlara dayanan etiketleme algoritması tasarlamışlardır. Restoran veri seti üzerinde (Pontiki vd., 2016) hedef terim çıkartma, hedef kategori çıkartma ve ikisinin aynı anda çıkarımı üzerine çözüm üretmişlerdir. Hedef kategori ve terimin aynı anda belirlenmesinde %46,7 F1 skoru elde etmişlerdir. Veri kümesini eğitime hazır hale getirmek için ITU NLP Web servisi kullanılmıştır. Hedef kategori tespiti için birçok-sınıflı sınıflandırma problemi ele alınmıştır. Çoklu sınıflandırma olmasının sebebi ise bir cümlede birden fazla duygu ve kategorinin bulunmasıdır. İkinci bir yaklaşım olarak ise dizilim etiketleme yaklaşımı uygulanmıştır (Çetin ve Eryiğit 2018).

Kama vd. (2019), ABSA için bir araç geliştirmiştir. Veri seti olarak çevrimiçi ürün incelemelerini kullanmışlar ve incelemelerden örtük ve açık yönleri frekans tabanlı yöntemle çıkarmışlardır. Yöntem duygu kelime gruplarını cümleden çıkartarak birbiriyle eşleştirmekte ve oluşan kelime gruplarını puanlamaktadır. Oluşan kelime gruplarının yardımıyla, örtük kelime görünüşleri çıkartılmaktadır. Bu çalışma ile F1-Skor %86.75, kesinlik değeri %91,22 olarak ölçülmüştür (Karagoz vd., 2019).

Özkan (2020), yüksek lisans tezinde Türkçe için ABSA çalışması gerçekleştirmiştir. Çalışmada, Akılı telefon hakkında web üzerinden alınan Türkçe incelemeler kullanılmıştır. Performans, fiyat ve kamera özellikleri ile ABSA çalışması gerçekleştirilmiştir. Açık ve kapalı özellik çıkarımı sağlanmış, bu çıkarımlar ilgili hedeflerle birleştirilmiştir. Sözlük tabanlı yaklaşım kullanılmıştır. İncelenen hedefler için en yüksek kesinlik, duyarlılık ve F1 ölçümü değerlerinin sırasıyla %93, %94 ve %93 olduğunu gösterilmiştir (Özkan, 2020).

Salur vd. (2021), Türkçe restoran verisi (Pontiki vd., 2016) üzerinde çalışmışlardır. Metinlerde lemma kelime, ara kelime, ham kelime formatlarından dönem sıklığı ters belge sıklığı (Term Frequency-Inverse Document Frequency [TF-IDF]) , gizli dirichlet tahsisi(Latent Dirichlet Allocation [LDA]), negatif olmayan matris faktorizasyon (Non-Negative Matrix Factorization [NMF]) ve kural tabanlı yöntemler ile görünüş çıkarımı yapılmıştır. Bu dört yöntem ile sonuçları farklı stratejilerde birleştirerek sonuçlar üretmişlerdir. Ham kelimelerde %60,07 F1-skor puanı ile TF-IDF'nin en başarılı yöntem olduğu açıklanmıştır. Ara kelimelerde is %60,07 F1-skor ile hibrit modelin başarılı olduğu sunulmuştur (Salur vd., 2021).

2.4.2 Restoran veri seti üzerinde gerçekleştirilen bazı ABSA çalışmaları

Song vd. (2019), ABSA için gerçekleştirmiş oldukları çalışmada hedeflere yönelik duygu eğilimi araştırmışlardır. Tekrarlayan sinir ağlarının paralelleştirilmesinde ki zorluklara ve uzun vadeli kalıpları hatırlamaya çözüm olarak dikkatli kodlayıcı ağı (Attentional Encoder Network [AEN]) önermişlerdir. Etiket güvenilmezliği sorunu için, kayıp işlevine etiket düzeltme düzenleme (LSR) ekleyerek etiket yumuşatmayı önermişlerdir. Model eğitimi için BERT kullanılmış ve model 3 farklı veri seti üzerinde test edilmiştir. Dikkat Kodlayıcı Ağına iki farklı gömme katmanı eklenmiş ve AEN-Glove ve AEN-Bert olmak üzere iki model elde edilmiştir. Duygu polaritesi olarak pozitif, negatif ve nötr olmak üzere üç kategoriye ayırılmıştır. LSR eklenen AEN-Glove yüksek doğruluk elde ederken LSR'siz Glove düşük doğruluk elde etmiştir. AEN-GloVe ve AEN-GloVe-BiLSTM modellerinin başarılarının birbirine yakın olduğu gözlemlenmiştir. AEN-Bert modeli boyut olarak yüksek olsa da doğruluk oranı olarak başarı elde etmiştir (Song vd., 2019).

Silva ve Marcacini (2021), ABSA için gerçekleştirdikleri çalışmada, metindeki söz dizimsel ve anlamsal özellikleri bilgisayar temsil edebilmek için BERT tabanlı DeBERT modelini önermişlerdir. Ayrıştırılmış bu model basit bir ince ayar stratejisi ile birlikte ABSA'nın veri kümelerinde önceki çalışmalara göre yüksek performans elde etmiştir. Amaç, konum vektörünü içerik vektörüne toplayan her bir bileşen için dikkat ağırlıklarını öğrenmektir. Burada konum vektörü yerleştirme sözdizimsel özellikler oluşturmaktan ve içerik yerleştirme anlamsal özelliklerden sorumludur. Eğitim sırasında sınıflandırma hatası, modelin başlangıç parametrelerini belirlemek için kullanılmaktadır. Duygu sınıflandırma görevi için kategorik çapraz entropi kaybı işlevine sahip standart bir softmax katmanı kullanmaktadırlar. 5936 örnekten oluşan SemEval 2014 Task 4 (Pontiki vd., 2014) laptop ve restoran veri seti kullanılmış, olumlu, olumsuz, nötr olmak üzeri 3 duygu durumu kategorisi üzerine çalışmışlardır. Laptop ve restoran veri seti için sırasıyla, %79,36, %83,42 F1 Skor elde etmişlerdir (Silva ve Marcacini,2021).

Reddy vd., (2020), SemEval 2014 Task 4 (Pontiki vd., 2014), SemEval 2016 Task 5 (Pontiki vd., 2016), SentiHood (Saeidi vd., 2016) veri setleri ile çalışmışlardır. ABSA çalışması için, araştırmacılar BERT-IL ve BERT-IL Finetuned olmak üzere 2 model önermişlerdir. BERT modelinin son 5 katmanının BERT-IL modeli ile birlikte eğitilmesi ile elde BERT-IL Finetuned elde edilir. Daha yüksek performans gösteren bu model daha yüksek işlem maliyeti ve bellek gerektirmektedir. Her iki modeli de 1'e-5 öğrenme oranı ve 0,1 Dropout ile ve 8 parti boyutuyla Adam optimizer ile eğitmişler ve tüm deneyler için yüzde doğruluk puanları raporlamışlardır. SemEval 2016 Task 5 görevi için BERT-IL %80,3, BERT-IL Finetuned %88,7 doğruluk puanı elde etmiştir (Rendy vd., 2020).

Zhang vd. (2022), ABSA için DPL yöntemini önermişlerdir. Vanilla Pseudo-Label methodunu baz alır ve çift yollu bir sisteme genişletir. Model tasarımı BERT'e dayanmaktadır. DPL'i en boy tabanlı duygu analizinde diğer yöntemlerle birleştirilebilmektedir. Çalışma da Relational Graph Attention Network (Bai vd., 2020), Adapter (Rietzler vd., 2019) ve MultiBERT modelleri ile birleştirilen DPL, bahsedilen modeller için %2'ye kadar doğruluk artışı sağlamıştır. Restaurant ve Laptop verisetleri üzerinde çalışılmıştır. Restaurant veriseti üzerinde %89,54 doğruluk oranı elde eden DPL

daha fazla performans kazanımı için tam uyumluluk gösterdiğini kanıtlanmıştır (Zhang vd.,, 2022).

Wallaart ve Frasincar (2019), ABSA için restoran (Pontiki vd., 2016) veri seti üzerinde çalışmışlardır. İki aşamadan oluşan duygu analizi algoritması tasarlanmıştır. Öncelikli olarak duygu tahminlemesi için sözlükleştirilmiş bir alan ontolojisi kullanır, ardından yedekleme algoritması olan döner dikkat mekanizması (Rotatory Attention Mechanism [LCR-Rot]) kullanır. Yedekleme algoritması üzerine çalışma sırasını değiştiren (LCRRot-inv) ve çoklu tekrarlamalara (LCR-Rot-hop) karşı iki özellik de eklenmiştir. Veri kümesi üzerinde ön işleme gerçekleştirilmiş, veri tokenize ve lemmatize edilmiştir. Gömme vektör boyutları 300'dür. GLoVe çerçevesinden önceden eğitilmiş kelime vektörü dağarcığı kullanılmıştır. Ontoloji tabanlı yaklaşım ve sinirsel dikkat modeli makalede tasvir edilmiştir. Görünüme dayalı duygu analizi için bir ontoloji yaklaşımı kullanarak, önceden tanımlanmış sınıfları, bu sınıflar arasındaki ilişkileri kullanarak duyarlılık tahminlenmiştir (Wallaart ve Frasincar, 2019).

Truşcă vd., (2020) ABSA için Hibrit Yaklaşımı (HAABSA) iki yönde geliştirmiştir. Önce, belirli bir metindeki kelime anlam bilimiyle daha iyi başa çıkabilmek için, bağlamsal olmayan sözcük yerleştirmelerini derin bağlamsal sözcük yerleştirmeleriyle değiştirilmiştir. İkinci olarak, girdi verilerini modellemede yöntem esnekliğini artırmak için HAABSA üst düzey temsillerine ekstra bir dikkat katmanı ekleyerek hiyerarşik dikkat kullanılmıştır. İki standart veri seti (SemEval 2015 ve SemEval 2016) kullanarak, önerilen uzantıların ABSA için oluşturulmuş modelin doğruluğunu iyileştirdiğini gösterilmiştir. Çalışma sonucunda doğruluk puanı, 2015 veri seti için %80,3'ten %81,7'ye ve SemEval 2016 veri seti için %86,4'ten %87,0'a yükseltmektedir (Truşcă vd., 2020).

3. MATERYAL VE YÖNTEM

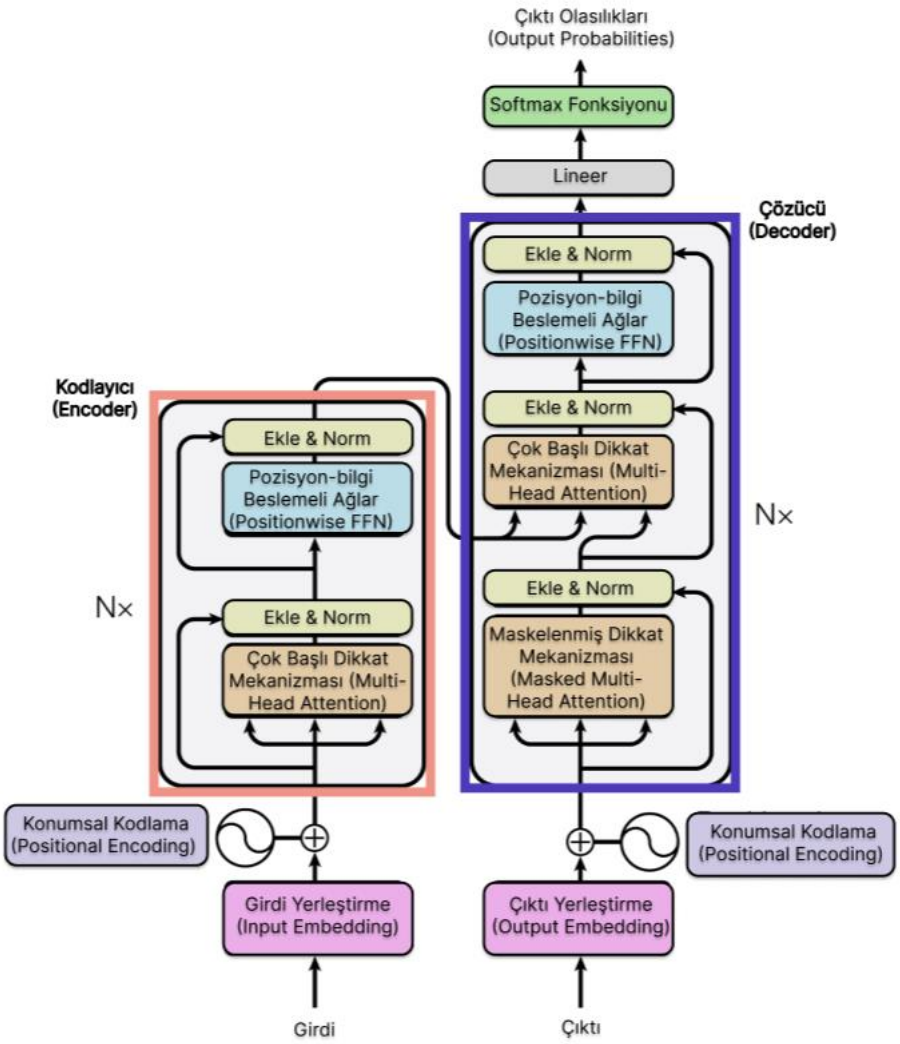
Bu bölümde çalışmada kullanılan yöntemler, modeller, transformatör (transformer) mimarisi, kullanılan Python kütüphaneleri, deneysel çalışmalar, kullanılan teknolojiler anlatılacaktır.

3.1. Transformatör Mimarisi

Transformatör (transformer) modelleri, bir dizideki birbirini etkileyen ve birbirine bağlı olan birbirinden uzak veri öğelerinin bile ince bağlantılarını tespit etmek için, ilgi veya öz-ilgi adı verilen ve hala gelişmekte olan bir dizi matematiksel teknik uygulamaktadır. İlk olarak Google'ın 2017 tarihli bir makalesinde (Vaswani vd., 2017) tanımlanan transformatör modelleri, makine öğrenmesi alanındaki gelişmelerde öncü haline geldiği için "dönüştürücü AI" olarak adlandırılmaktadır. Büyük dil modelleri oluşturmak için geniş bir şekilde kullanılmıştır. Transformatör mimarisi, önceki rekabetçi modellere göre daha hızlı eğitilebilir ve paralel işleme yetenekleriyle öne çıkmaktadır (Vaswani vd., 2017).

Transformatör mimarisi,

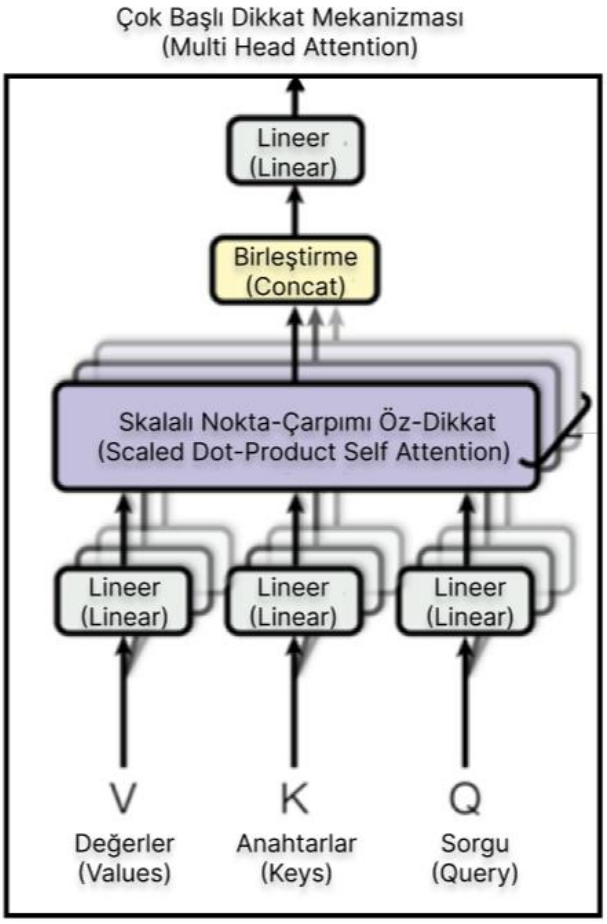
- Dikkat mekanizması (attention mechanism),
- Çok başlı dikkat (multi-head attention),
- Konumsal kodlama (position encoding),
- Kodlayıcı-çözücü mimarisi (encoder-decoder architecture),
- Pozisyon-bilgi beslemeli ağlar (position-wise feed-forward networks),
- Katman normalleştirme (layer normalization),
- Lineer ve softmax katmanından (linear and softmax layer) oluşmaktadır.



Şekil 3.1. Transformatör mimarisi (Vaswani vd., 2017).

Transformatör mimarisinde, bileşenler arasındaki akış, genellikle bir kodlayıcı (encoder) ve bir çözücü (decoder) olmak üzere iki ana kısma ayrılmaktadır. Transformatör mimarisi Şekil 3.1'de gösterilmektedir. Şekildeki kodlayıcı (encoder) kısmında, ilk olarak girdi bulunmaktadır. Girdi**,** genellikle bir cümle, metin veya başka bir sıralı veri olabilir. Sonrasında konumsal kodlama (positional encoding), giriş verisi üzerindeki her kelimenin veya elemanın konumunu ifade etmek için konumsal kodlamalar eklenmesi, modelin sıralı bilgileri daha iyi işlemesine yardımcı olmaktadır. Konumsal kodlama eklenmiş giriş verisi, bir dizi kodlayıcı katmanından geçmektedir. Her bir katmanda, öz-dikkat (self-attention) mekanizması, giriş verisinin belirli bağlamı ve ilişkileri dikkate almasını sağlamaktadır. Öz-dikkat (self-attention), çok başlı dikkat (multi-head attention)

mekanizmasının temel bir bileşeni olarak kodlayıcı (encoder) kısmında kullanılıp çok başlı dikkat (multi-head attention), birden fazla başlık (head) içeren bir dikkat mekanizmasını ifade etmektedir. Bu başlıklar, aynı girdiyi farklı ağırlıklarla dikkate alarak modelin farklı özelliklere odaklanmasını sağlamaktadır (Vaswani vd., 2017).



Şekil 3.2. Çok başlı dikkat mekanizması mimarisi (Vaswani vd., 2017).

Çok başlı dikkat mekanizması bir adım daha ileri giderek öz-dikkat mekanizmasını çoklu başlıklar (heads) kullanarak genişletmektedir. Her bir dikkat başlığı, farklı öğrenilmiş ağırlıklara sahiptir ve bu başlıkların çıktıları birleştirilerek genel bir çıktı elde edilmektedir. Bu, modelin farklı bağlam ve özelliklere daha esnek bir şekilde dikkat etmesini sağlamaktadır. Yani, her bir dikkat başlığı, öz-dikkat mekanizmasını farklı şekillerde uygulamakta ve farklı özelliklere odaklanmasını sağlamaktadır (Du vd., 2020).

Çok başlı dikkat mekanizması mekanizmasın mimarisi Şekil 3.2'de gösterilmektedir. Gösterilen mimari de anahtar (key), sorgu (query), ve değer (value) vektörlerini

kullanmaktadır. Her bir dikkat başlığının içerdiği öz-dikkat mekanizmalarının ağırlıklandırma işlemlerini gerçekleştirmek üzere kullanılır. Key vektörleri, dikkat ağırlıklarını belirlemek için kullanılmaktadır. Her bir dikkat başlığı için öğrenilen key vektörleri, sorgu vektörleri ile benzerlik ölçümü yapmaktadır. Query vektörleri, dikkat ağırlıklarını belirlemek ve değer vektörlerini dikkat ağırlıklarıyla ağırlıklı bir şekilde toplamak için kullanılmaktadır. Value vektörleri, dikkat ağırlıklarıyla ağırlıklı bir şekilde toplanacak olan değerler olup, bu vektörlere lineer dönüşüm uygulanmaktadır. Bu işlemler sayesinde Skalalı Nokta Çarpımı Öz Dikkat (Scaled Dot-Product Self Attention) mekanizmasının her kelimenin diğer kelimelerle olan ilişkilerini öğrenmesi ve geniş bağlamı ele alabilmesi sağlanmaktadır. Skalalı Nokta Çarpımı Öz Dikkat (Scaled Dot-Product Self Attention) özellikle transformatör (transformer) modellerinde kullanılan bir öz-dikkat (self-attention) mekanizmasıdır (Shen vd., 2021). Son aşamada, her bir dikkat başlığından elde edilen ağırlıklar birleştirilmekte (concatenate) ve ardından bir lineer dönüşüm (linear transformation) uygulanarak final çıktı elde edilmektedir.

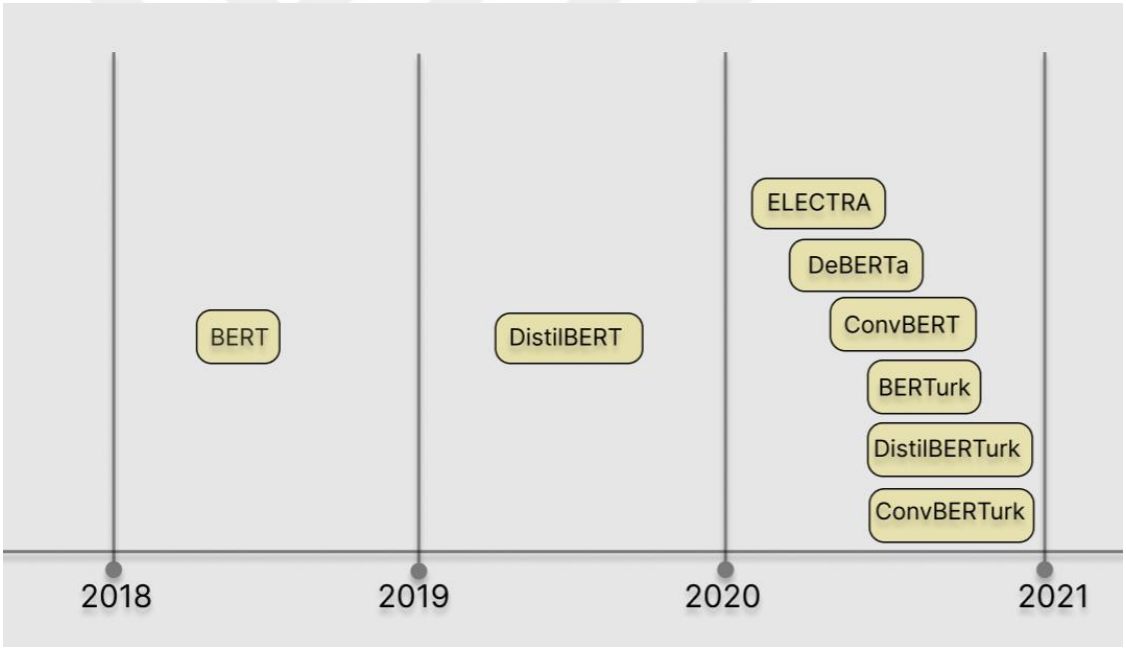
Her kodlayıcı katmanının çıktıları, bir katman normalleştirme işleminden geçirilmekte ve ardından ileri beslemeli ağlar tarafından işlenmektedir. Bu adımlar, modelin daha karmaşık özellikleri öğrenmesine yardımcı olmaktadır. Katman normalleştirme, bir sinir ağı katmanının çıktılarını normalleştirmek için kullanılan bir tekniktir (Zhang vd., 2020). Bu normalleştirme işlemi, her bir örnek içinde bir katmanın tüm birimlerine aynı normalleştirme faktörlerini uygulamaktadır. Kodlayıcı katmanlarını geçtikten sonra elde edilen çıkış, bir çözücü (decoder) katmanına iletilmektedir.

Çözücü (decoder) kısmında, çözücü katmanlarında, maskeli çok başlı dikkat mekanizması (masked multi-head attention) kullanılarak çıkışa dikkat uygulanmaktadır (Vaswani vd., 2017). Bu, özellikle dil çevirisi gibi sıralı veri görevlerinde gelecekteki bilgileri kullanmamak için önem arz etmektedir. Ardından katman normalleştirme (Layer Normalization) işlemine geçilmektedir. Her çözücü katmanının çıktıları, bir kez daha katman normalleştirme işleminden geçirilmekte ve ardından ileri beslemeli ağlar (feed-forward networks) tarafından işlenmektedir. Çözücü katmanlarını geçtikten sonra, modelin çıktısını oluşturan son lineer ve softmax katmanına geçilir. Bu katman, modelin son çıkışını oluşturmaktadır.

Transformatör (transformer), mimarisi, transfer öğrenme için oldukça uygundur ve büyük dil modelleri ile kullanılmaktadır. Çeşitli doğal dil işleme görevlerinde, özellikle dil anlama, metin sınıflandırma, dil çevirisi ve metin oluşturma gibi alanlarda başarıyla uygulanmaktadır.

3.2. Çalışmada Kullanılan Modeller

Bu bölümde, çalışmada kullanılan modeller hakkında genel bilgi, modellerin kullandıkları bazı teknikler ve bu teknikler ile eğitilen Türkçe ön eğitimli modeller sunulmuştur.

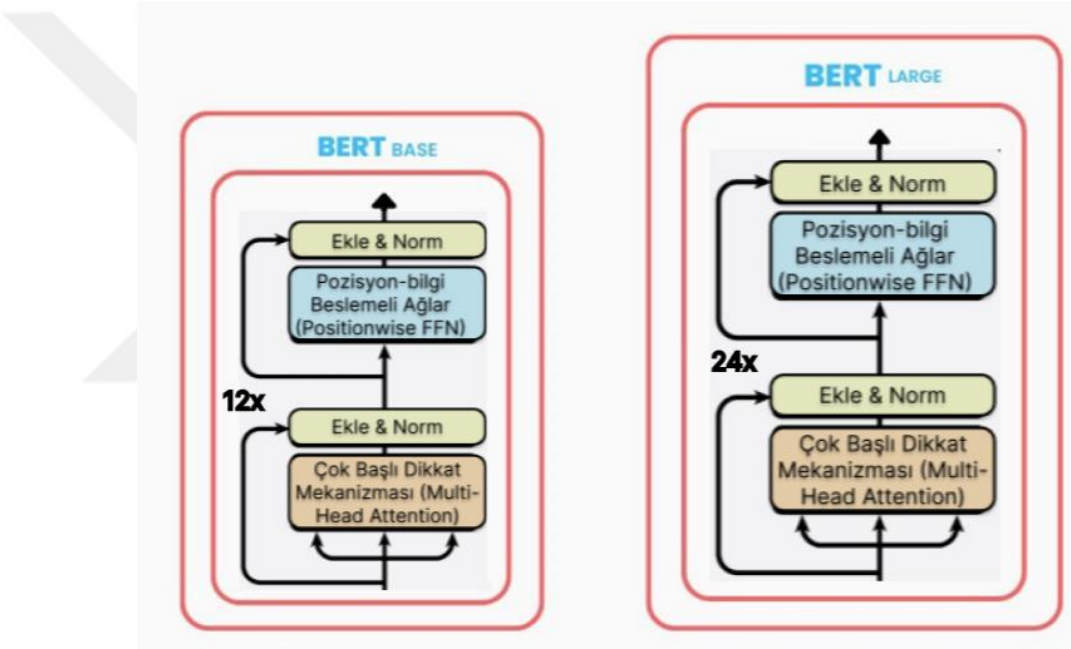


Şekil 3.3. Çalışmada kullanılan modellerin kronolojisi.

Çalışmada kullanılan modeller yayınlanma sırası olarak; BERT (Devlin vd., 2018), DistilBERT (Sanh vd., 2019), ELECTRA (Clark vd., 2020), DeBERTa (He vd., 2020), ConvBERT (Jiang vd., 2020)'tir. Ardından 2020 yılında Türkçe ön eğitimli, ConvBERTurk, BERTurk, DistilBERTurk, ELECTRA modelleri tanıtılmıştır (Schweter 2020). Yayınlanma yıllarına göre modeller Şekil 3.3'te gösterilmektedir.

3.2.1. BERT

Transformatörden Çift Yönlü Kodlayıcı Temsilleri (BERT), önceden eğitilmiş bir dil modelidir ve 2018'de Google tarafından geliştirilmiştir. Bert modeli, transformatör mimarisini kullanmaktadır. Fakat transformatör mimarisinin tamamını kullanmaz. BERT, önceden eğitilmiş bir dil modelidir ve aslında transformatörün sadece kodlayıcı (encoder) kısmını içermektedir (Devlin vd., 2018). Kodlayıcı, giriş verilerini bir dizi öznitelik vektörüne dönüştürmek için kullanılmaktadır.



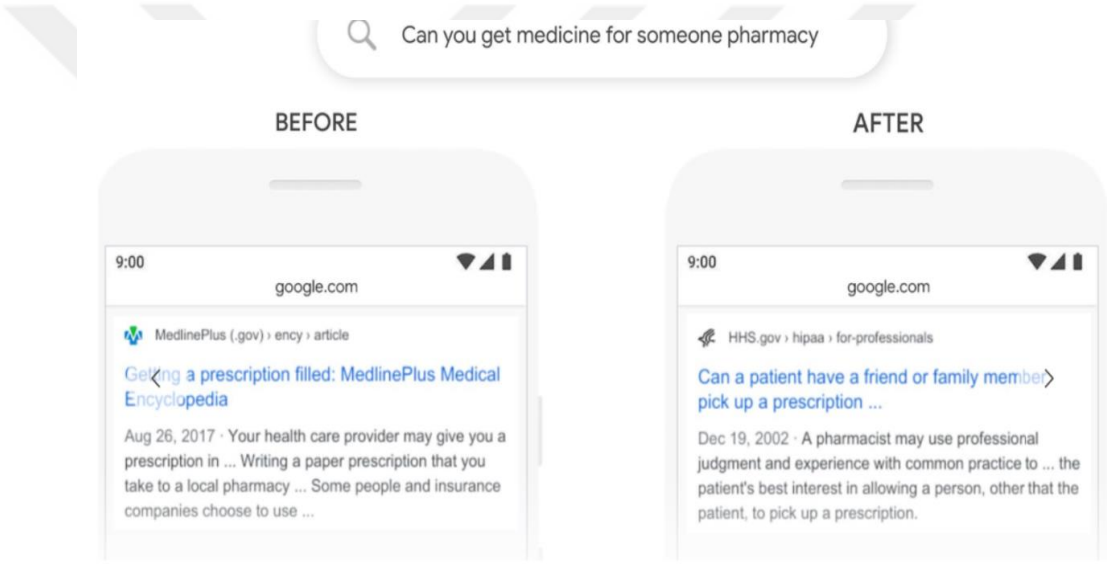
Şekil 3.4. BERT modellerinde tranformatör kullanımı.

BERT'in iki ana modeli BERT temel (base) ve BERT büyük (large) modelidir. BERT temel (base) modeli 12 transformatör katmanı bulundurmaktadır (Joshi vd., 2019). Bu transformatörler çift yönlü olarak çalışmaktadır. BERT büyük large modeli ise 24 adet transformatör katmanı bulundurmaktadır (Şekil 3.4).

BERT, geniş bir öğrenme veri kümesi ile eğitilmekte ve önceden eğitilmiş modelin bu eğitim verileri üzerindeki anlayışı, daha spesifik görevlerdeki performansını artırmaktadır. Belirteç tanıma, adlandırılmış varlık tanıma, soru cevaplama gibi doğal dil

işleme görevlerinde kullanılabilir. Model, hem sol hem de sağ bağlamı aynı anda dikkate alan bir yapıya sahiptir ve bu sayede, özellikle kelime anlamlarının belirlenmesi gibi zorlu doğal dil işleme görevlerinde etkili sonuçlar vermektedir (Devlin vd., 2018).

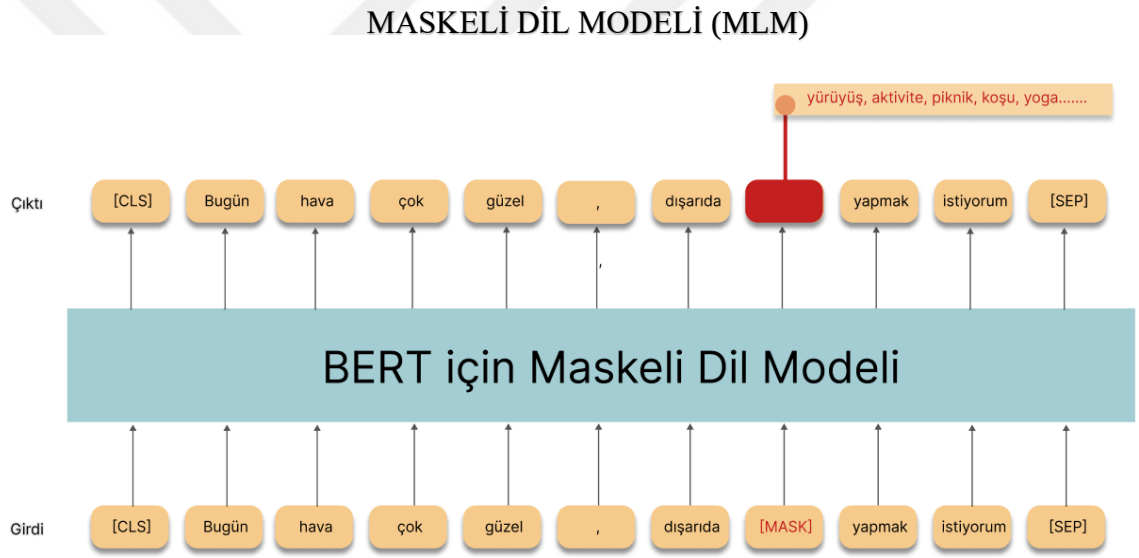
Google'ın BERT öncesi arama motorlarında kullandığı algoritmalar, Panda (2011), Penguin (2012), Hummingbird (2013), Pigeon (2014), Mobile-Friendly Update (2015), RankBrain (2015), Fred (2017), BERT (2019), Core Algorithm Updates (devam ediyor) şeklindedir. 2015'te yayınlanan RankBrain, sorguların amacını anlamaya yönelik bir algoritmayken, 2017'de yayınlanan Fred, düşük kaliteli içeriye sahip siteleri filtrelemeyi hedef almaktadır (Hoş, 2023).



Şekil 3.5. BERT modelinin arama sorgusu için kullanımı (Urhan, 2020).

BERT ile arama sorgularında değişen gelişime ait örnek Şekil 3.5'te gösterilmektedir. Şekilde yapılan sorguda kullanıcı başka birisi eczaneden ilaç alınıp alınamayacağını sorgulamış ve BERT öncesinde bu sorgu karşılığında, reçete doldurma hakkında bilgi içeren bir sonuçla karşılaşmıştır. BERT sonrasında ise algoritma, kullanıcının sorusunu anlamış, arkadaş ve aile bireyleri için reçete ile ilaç alınması hakkında sonuç sunmuştur. Bu da BERT algoritmasının doğal dil işleme alanındaki başarısını ortaya koymaktadır. BERT modeli çift yönlü transformatörler kullanmasının yanında maskeli dil modelleme (Masked Language Model [MLM]) ve sonraki cümle tahmini (Next Sentence Predition [NSP]) isminde iki teknik kullanmaktadır (Devlin vd., 2018).

Maskeli dil modelleme (Masked Language Model [MLM]), dil modellerinde yaygın olarak kullanılan bir tekniktir. Bu teknik, bir dil modeline verilen bir metnin bir kısmını (kelime, harf vb.) maskeleyerek, modelin eksik olan kısım hakkında tahmin yapmasını sağlamaktadır (Sinha vd., 2021). Bu yöntem genellikle ön eğitim sürecinde kullanılmaktadır. Örneğin, BERT modeli, ön eğitim aşamasında, bir metinde rastgele seçilen kelimelerin %15'ini maskeleyerek, geri kalan kelimeleri kullanarak maskelemelerin yerlerini tahmin etmeye çalışmaktadır. Bu şekilde, model dilin yapısını, kelime ilişkilerini ve dilin doğal özelliklerini öğrenmektedir.



Şekil 3.6. BERT maskeli dil modeli çalışma prensibi.

Maskeli dil modelleme (Masked Language Model [MLM]) bir kelimeyi maskeleyerek tahmin edilmesi prensibine dayanmaktadır. Modelin çalışma prensibi Şekil 3.6.'da gösterilmektedir. "Bugün hava çok güzel, dışarıda [MASK] yapmak istiyorum", cümlesinde maskelenmiş kısma uygun kelimelerin getirilmesi maskeli dil modeli tarafından gerçekleştirilmektedir.



Şekil 3.7. BERT sonraki cümle tahmini çalışma prensibi.

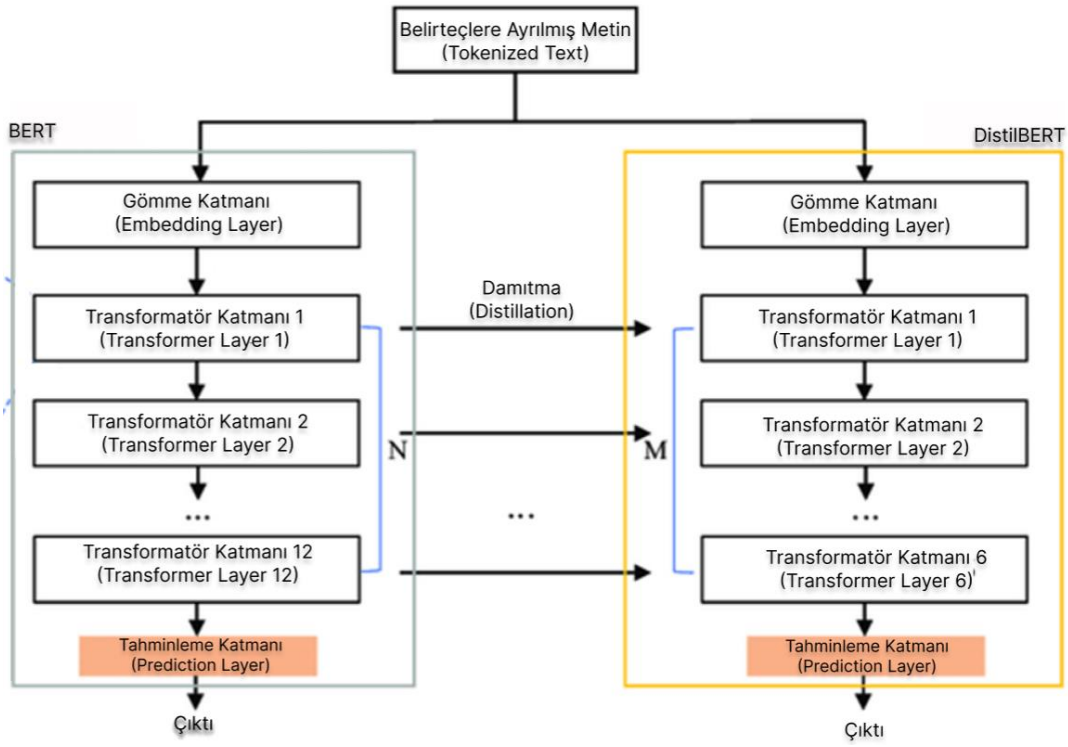
Sonraki cümle tahmini (Next Sentence Predition [NSP]), dil modellerinde kullanılan bir diğer ön eğitim tekniğidir. Bu teknik, bir metnin bir cümlesini alarak, sonrasında gelen bir cümleye ait olup olmadığını tahmin etmeyi amaçlamaktadır. NSP tekniği, ön eğitim aşamasında dil modellerine önceden verilen metinlerdeki cümle ilişkilerini anlamalarını öğretmektedir. Bu sayede, model daha sonra gelen girdilerde iki cümle arasındaki bağı anlayabilir ve daha doğru tahminler yapabilmektedir (Sun vd., 2021).

Metnin ilk cümlesini baz alarak ikinci cümleyi tahmin eden bu sistemin çalışma prensini Şekil 3.7'de gösterilmektedir. "Bugün hava çok güzel, dışarı çıkıp biraz yürüyebiliriz." cümlesine devam niteliğinde olabilecek alternatifler algoritma tarafından tahmin edilmektedir. Örneğin; "Seninle birlikte yürümek bana da iyi gelir, hadi birlikte çıkalım." cümlesi mantıklı bir devam cümlesi olarak düşünülebilir. Ancak, "Bugün hava çok güzel, dışarı çıkıp biraz yürüyebiliriz." cümlesini takiben gelen "Yıldızların ışıltısı gökyüzünü aydınlatıyordu." ifadesi, bağlamsal olarak uyumsuz bir devam niteliği taşımaktadır. Eğitim sürecinde, cümleler ikili gruplar halinde incelenir ve ikinci cümlenin, ilk cümleye bağlamsal bir devam olup olmadığına karar verilmeye çalışılır. Cümle çiftlerinin bulunduğu veri seti içerisinde ikinci cümleler rastgele yer değiştirilerek modelin uyumu test edilir. Model, bu eğitim sürecinde optimal sonuçlara ulaşmayı amaçlayarak kayıp fonksiyonunun değerini en aza indirmeye çalışmaktadır.

3.2.2. DistilBERT

DistilBERT, Hugging Face tarafından geliştirilmiştir. DistilBERT'in BERT modeline kıyasla daha küçük, hızlı ve hafif bir versiyonudur. BERT gibi önceden eğitilmiştir, ancak daha küçük bir modeldir. Bu öğrenme stratejisi, "öğrenci-öğretmen" prensibine dayanarak, büyük ve karmaşık bir modelin bilgisini daha küçük bir modele aktarılarak daha hafif bir model elde edilmesini sağlamaktadır. Transformatör mimarisinde kullanılan işlemlerin çoğu (doğrusal katman ve katman normalleştirme), modern doğrusal cebir çerçevelerinde yüksek düzeyde optimize edilmiştir (Sanh vd., 2019).

DistilBERT, sadece bir kısmı kullanılarak eğitilen verilerle, BERT'ten daha az parametreye sahip olmasına rağmen, benzer performans göstermektedir. Ayrıca, DistilBERT daha hızlı ve daha az bellek tüketen bir modeldir, bu nedenle, daha küçük işlemci gücüne sahip sistemlerde veya mobil cihazlarda doğal dil işleme görevleri için kullanılabilir (Sanh vd., 2019).



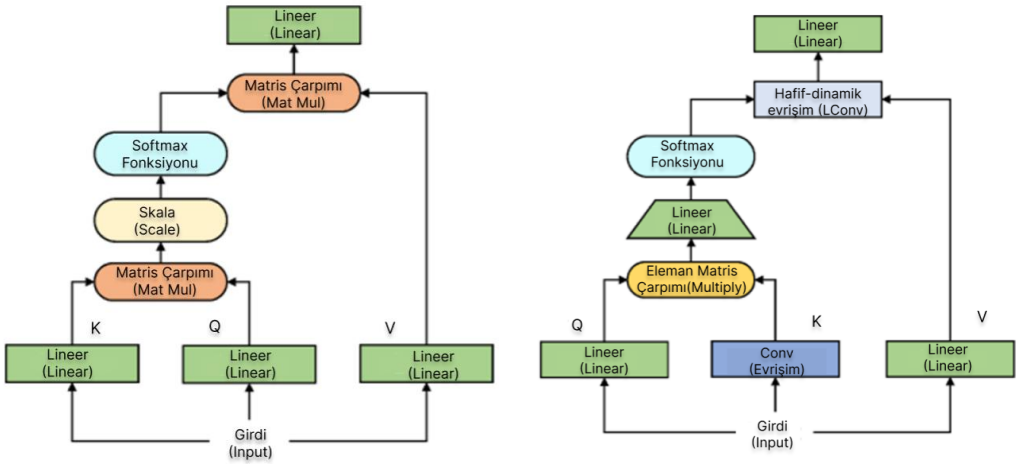
Şekil 3.8. Distilbert ve BERT modellerinde transformatör katmanının kullanımı (Adel vd., 2022).

DistilBERT modelinde (Distilled BERT), orijinal BERT modeline kıyasla daha az transformatör katmanı bulunmaktadır. DistilBERT modeli genellikle 6 veya 12 transformatör katmanı kullanır transformatör katmanının modellere göre kullanımı Şekil 3.8.'de gösterilmektedir. Bu sayede, DistilBERT daha az hesaplama gücü gerektirir ve daha hızlı çalışabilir. Ayrıca, daha az parametreye sahip olduğu için daha az bellek kullanır, bu da modelin hafifletilmiş bir versiyonunu oluşturmaktadır (Adel vd., 2022).

DistilBERT ile yapılan deneylerde, BERT modeline kıyasla daha düşük doğruluk oranlarına sahip olduğunu, ancak yine de yeterli doğru sonuçları ürettiğini göstermektedir. Sonuç olarak, DistilBERT, yüksek performanslı bir doğal dil işleme modeli olmasına rağmen BERT'ten daha küçük ve daha hızlıdır (Sanh vd., 2019).

3.2.3. ConvBERT

ConvBERT, dinamik evrişim işlemlerini birleştirerek popüler doğal dil işleme modeli BERT'nin performansını artıran bir mimaridir (Jiang vd., 2020). ConvBERT, Facebook AI tarafından geliştirilen bir modeldir.



Şekil 3.9. Skalalı nokta çarpımı öz dikkat ve aralık tabanlı dinamik konvolüsyon (Jiang vd., 2020).

BERT'ten farklı olarak, ConvBERT, hem sıralı hem de evrişimli yapılar kullanarak daha hızlı bir şekilde önceden eğitilir. BERT mimarisinin belirli sınırlamalarının üstesinden gelmek üzere geliştirilmiştir. BERT modeli, öz-dikkat (self-attention) mekanizmasında

skalalı nokta çarpımı öz dikkat (scaled dot-product self attention) kullanırken, ConvBERT (Convolutional BERT) ise, geleneksel BERT modeline kıyasla farklı bir yaklaşım benimsemektedir. ConvBERT, konvolüsyonel katmanları içerir ve özellikle dil modellerinde kullanılan öz-dikkat mekanizmasının yerine bir aralık tabanlı dinamik konvolüsyon (span-based dynamic convolution) mekanizması kullanmaktadır (Jiang vd., 2020). Her iki mekanizmanın da mimarisi Şekil 3.9'da gösterilmektedir. Skalalı nokta çarpımı öz dikkat mekanizmasında iki vektör arasındaki nokta çarpımını alarak benzerlik ölçüsü elde edilmektedir. Bu, iki vektörün ne kadar benzer olduğunu belirlemeye yardımcı olmaktadır. Nokta-çarpımı sonuçlarını ağırlıklarını bir sabit ile çarparak (genellikle karekök boyutu kadar olan bir skalayla), gradyan patlaması veya kaybolması gibi problemleri önlemeye yardımcı olmaktadır. Nokta çarpımı sonuçlarına softmax fonksiyonu uygulanmaktadır. Her kelimenin temsilini oluşturmak için, softmax ağırlıklarıyla ağırlıklı toplam alınmaktadır. Aralık tabanlı dinamik konvolüsyon (span-based dynamic convolution) ise belirlenen aralık (span) içindeki kelimelerle özel bir konvolüsyon işlemi gerçekleştirilmekte ve her bir kelime, span içindeki diğer kelimelerle ağırlıklandırılarak özel bir temsil elde edilmektedir. Bu mekanizmada oluşturulan iki vektör arasında eleman bazlı matris çarpımı gerçekleştirlmektedir. Matris çarpımı sonuçlarına softmax fonksiyonu uygulanmaktadır. Hafif dinamik evrişim (Light-weight and dynamic convolution [LCov]), tüm ağırlıkları birbirine bağlayan hafif, derinlik açısından evrişimi ifade etmektedir. İki modelin temel farkı girdi aşamasından sonra uygulanan konvolüsyon işlemidir diyebiliriz.

Adlandırılmış varlık tanıma görevlerinde ConvBERT, BERT'i geride bırakmıştır. Aynı zamanda BERT'ten daha az parametre ile çalıştığı halde daha iyi sonuçlar elde ettiği gözlemlenmiştir (Jiang vd., 2020).

3.2.4. ELECTRA

ELECTRA, dil modelleri için yeni bir önbellekleme yaklaşımıdır. BERT ve GPT gibi mevcut önbellekleme yöntemleri modelleri birer üreteç olarak eğitmektedir. Diğer bir ifade ile modelleri, bir cümledeki eksik kelimeleri tahmin etmeye veya verilen bir ipucuna dayanarak yeni metinler oluşturmaya yönelik eğitilmektedir (Clark vd., 2020).

ELECTRA modeli bir üreteç yerine bir ayırıcı olarak eğitildiği yeni bir önbellekleme yaklaşımı önermektedir. Yani, eksik kelime tahmin etmek veya yeni metin oluşturmak yerine, model gerçek ve sahte girdiler arasındaki farkı belirlemek için eğitilmiştir (Chi vd., 2021).

Yazarlar, bunu "değiştirilmiş simge tespiti" adı verilen yeni bir önbellekleme göreviyle başarmışlardır. Bu görevde, model, bir cümledeki "gerçek" bir simge ile bir üreteç modeli tarafından değiştirilmiş bir "sahte" simge arasındaki farkı belirlemek için eğitilmiştir. Bu yaklaşımın temel avantajı, yazarların belirttiği gibi, daha verimli bir önbellekleme sağlamasıdır. Çünkü ayırıcı modelin sadece gerçek veya sahte girdi ayırımı yapması gerektiği için üreteç modelden daha büyük miktarda veriyle eğitilebilir (Clark vd., 2020).

3.2.5. DeBERTa

Microsoft Research tarafından geliştirilen bir modeldir. BERT gibi önceden eğitilmiştir, ancak hem yoğun hem de verimli bir mimariye sahiptir. Ayrıca, çok dilli modelleri desteklemesiyle de öne çıkmaktadır (He vd., 2020). DeBERTa'nın en önemli özelliği, "çözümleme" adı verilen bir tekniktir. Bu teknik, DeBERTa 'nın BERT'ten farklı olarak, kodlama ve çözümleme aşamalarında farklı dikkat mekanizmalarını kullanmasına olanak tanımaktadır. Bu da modelin daha iyi bir anlama ve daha yüksek performans elde etmesine yardımcı olmaktadır (He vd., 2020). Ayrıca DeBERTa, BERT'in dikkat mekanizmalarının daha da geliştirilmiş bir versiyonunu kullanmaktadır. Bu geliştirilmiş dikkat mekanizması, "ayrıştırılmış dikkat" olarak adlandırılır ve farklı görevlere yönelik farklı dikkat ağırlıkları kullanarak daha iyi bir sonuç elde etmektedir (He vd., 2020).

DeBERTa'nın performansı, birçok NLP görevinde diğer dil modellerinden daha yüksektir. Örneğin, GLUE (General Language Understanding Evaluation) benchmark testinde, DeBERTa 'nın 4,4 milyon parametresiyle BERT'ten %1,5 daha iyi performans gösterdiği görülmüştür. Sonuç olarak, DEBERTA, BERT temelli bir dil modeli için yenilikçi bir yaklaşım sunmaktadır (Martin vd., 2022).

3.2.6. Türkçe için BERTurk Modeli

BERTurk modeli Türk OSCAR corpus, Wikipedia dökümü , OPUS corpora ve Kemal Oflazer tarafından sağlanan özel bir corpus ile eğitilmiştir. 35 GB boyutta corpus ile Google'ın TensorFlow Araştırma Bulutu (TFRC) sayesinde, TPU v3-8 üzerinde kasalı bir modeli 2 milyon adımda eğitilmiştir (Schweter, 2020).

3.2.7. Türkçe için DistilBERTurk Modeli

DistilBERTurk, Türkçe için topluluk güdümlü kasalı damıtılmış bir BERT modelidir. 7 GB corpus ile DistilBERTurk resmi Hugging Face uygulaması ile 5 gün boyunca 4 RTX 2080 TI üzerinde eğitilmiştir (Schweter, 2020).

3.2.8. Türkçe için ELECTRA Modeli

ELECTRA, kendi kendini denetleyen dil temsili öğrenimi için bir yöntemdir. ELECTRA modelleri, bir GAN'ın ayrımcısına benzer şekilde, başka bir sinir ağı tarafından üretilen "gerçek" giriş belirteçleri ile "sahte" giriş belirteçlerini ayırt edecek şekilde eğitilmiştir (Schweter, 2020).

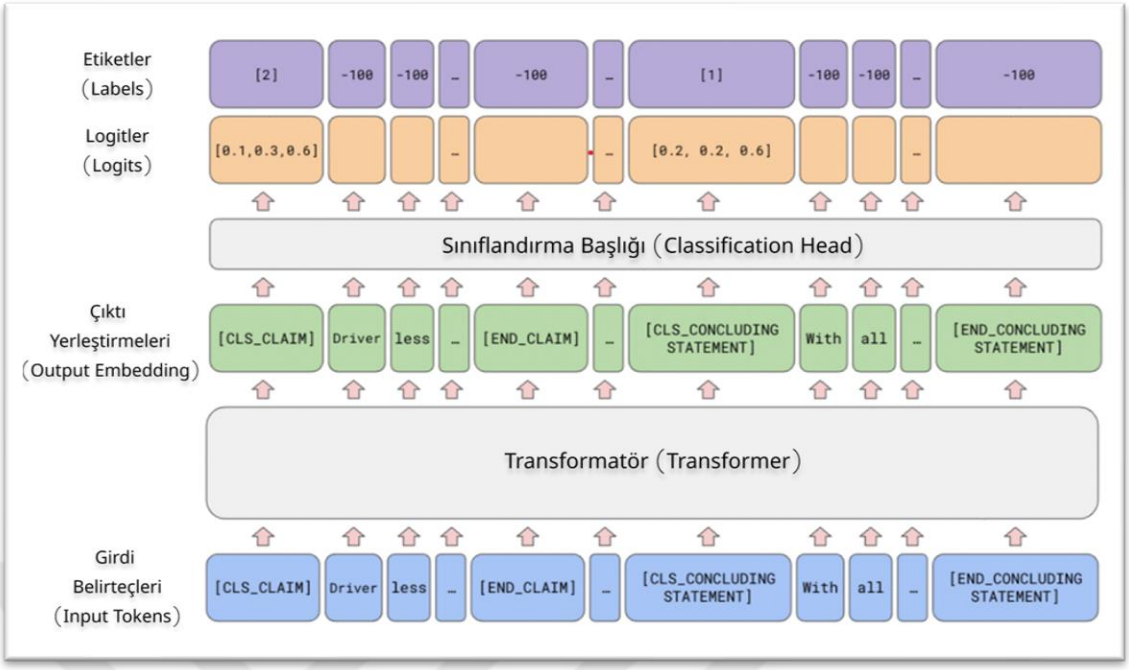
3.2.9. Türkçe için ConvBERTurk modeli

AI2 ekibinden çok dilli mC4 derleminin yakın zamanda yayınlanan Türkçe kısmı üzerinde eğitilmiştir ve 242 GB corpus kullanılmıştır. ELECTRA temel modeline ek olarak, mC4 korpusunun Türkçe kısmında bir ConvBERT modeli de eğitilmiştir. TPU'da 1M adım eğitilmiştir (Schweter, 2020).

3.3. Belirteç Sınıflandırması

Belirteç sınıflandırması, belirteçleri anlamlarına veya işlevlerine göre önceden tanımlanmış sınıflara ayırma işlemidir. Belirteçler, doğal dil işleme görevlerinde kullanılan sözcükler veya tümcecikler gibi ayrı metin birimleridir. Belirteç sınıflandırması, duyarlılık analizi, adlandırılmış varlık tanıma ve konuşma parçası etiketleme gibi çeşitli uygulamalarda kullanılmaktadır (Yuan vd., 2021). Belirteç sınıflandırmasına bir örnek, metindeki deyimsel ifadelerin tanımlanmasıdır. Salton ve ekibi, deyim belirteci sınıflandırmasına göre tahmine dayalı özellikleri kodlayan dağıtılmış temsiller oluşturmak için düşünceyi atlama vektörleri (skip-thought vectors) kullanımını araştırmışlardır. Bu temsilleri kullanan sınıflandırıcıların, deyim belirteci sınıflandırmasındaki en son teknolojiye kıyasla rekabetçi performansa sahip olduğunu göstermişlerdir (Salton vd., 2016). Belirteç sınıflandırmasının başka bir örneği, yasal argüman madenciliğidir. Xu ve Ashley , belirli yasal argüman öğelerinin daha doğru tanımlanması için yasal metinlerdeki belirteçleri kelime düzeyinde sınıflandırmak için bir Longformer modeli kullanmışlardır. Belirteç düzeyinde sınıflandırmanın bu görevde cümle düzeyinde sınıflandırmadan daha iyi performans gösterdiğini bulmuşlardır (Xu ve Ashley, 2022) .

Belirteç sınıflandırması, diyalog eylemleri olarak cümlelerin sınıflandırılmasında da kullanılabilir. O'Shea ve ekibi, işlev sözcüklerinde bulunan yapısal bilgilere dayalı olarak cümlelerin diyalog eylemleri olarak sınıflandırılması için yeni bir teknik önermektedir. İşlev sözcüklerini sayısal belirteçlerle değiştirerek ve her bir içerik sözcüğünü standart bir sayısal joker simgeyle değiştirerek göze çarpan özellikleri çıkarmak için karar ağaçlarını kullanmaktadır (O'Shea vd., 2010). Sonuç olarak belirteç sınıflandırması, belirteçleri anlamlarına veya işlevlerine göre önceden tanımlanmış sınıflara ayırma işlemi olup duygu analizi, adlandırılmış varlık tanıma ve konuşma parçası etiketleme gibi doğal dil işlemede çeşitli uygulamalara sahiptir. Belirteç sınıflandırması için düşünceyi atlama vektörleri (skip-thought vectors), karar ağaçları gibi farklı teknikler kullanılabilir.

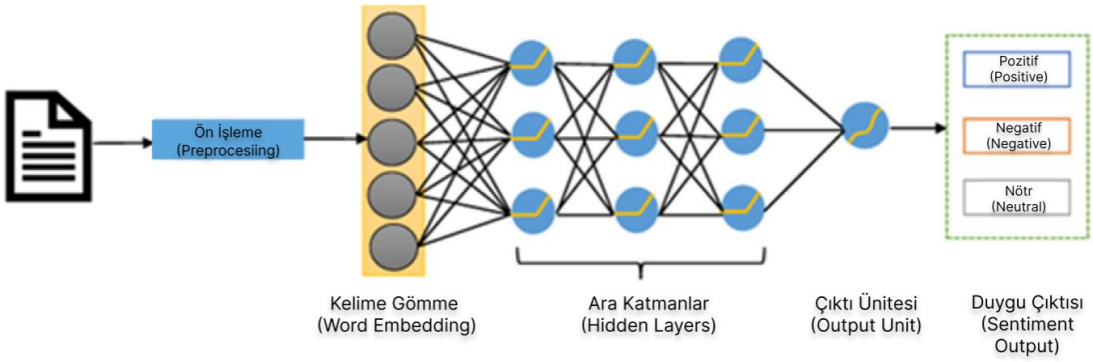


Şekil 3.10. Belirteç sınıflandırma modellerinin çalışma prensibi (Broad 2022).

Belirteç sınıflandırmada kullanılan girdi belirteçleri (Input tokens) veri setine ait metinlerin küçük parçalara ayrılmış halidir. Özellikle bir kelime veya alt cümle olabilirler. Model metni girdi belirteçlerine ayırdıktan sonra model her birini ayrı ayrı işlenmektedir. Sisteme ait mimari Şekil 3.10'da gösterilmektedir. CLS_CLAIM, bir argümanın başlangıcını temsil eden bir belirteci ve argümanın kendisini ifade eden bir vektörü birleştirirken, END_CONCLUDING_STATEMENT, bir argümanın sonunu temsil eden bir belirteci ve argümanın sonunu ifade eden bir vektörü birleştirmektedir. Girdi belirteçleri hazır hale getirildikten sonra, derin öğrenme mimarisi olan transformatör (transformer) işlemine tabii tutulmaktadır. Ardından çıktı yerleştirmeleri (output embeddings) oluşur. Çıktı yerleştirmeleri, modelin çıkışı olarak verilen herhangi bir öğenin bir vektör temsilidir. Sınıflandırma başlığı (classification head) ise bir modeldeki son katmanlardan biridir ve modelin sınıflandırma yapmasına yardımcı olmaktadır. Sınıflandırma başlığı, çıktı katmanından gelen bilgileri işler ve sonuç olarak sınıf tahminleri üretir. Son aşamada, logit ve etiket (label) üretilmektedir. Üretilen logitler, modelin çıkışında hesaplanan sayısal değerlerdir ve sınıf olasılıklarını tahmin etmek için kullanılmaktadır. Etiketler ise bu olasılıklar sonucunda, yüksek olasılık veren sınıfı hedef çıktı olarak göstermek için kullanılmaktadır.

3.4. Sıralı Sınıflandırma Modeli

Doğal dil işleme için Sıralı Sınıflandırma Modeli, sıralı cümle sınıflandırması için kullanılan bir model türüdür. Bu model, çevreleyen cümleler içindeki bağlamsal bilgileri kullanan hiyerarşik bir sıralı etiketleme ağına dayanmaktadır (Munkhdalai ve Yu 2017). Model, cümleleri göründükleri bağlamı dikkate alarak bir sırayla sınıflandırmak için tasarlanmıştır. Sıralı Sınıflandırma Modeli, duygu analizi, sahte haber tespiti ve kısa metin sınıflandırması gibi çeşitli NLP görevlerinde kullanılmıştır (Li vd., 2021, Prasanna vd., 2018, Zhoa vd., 2021). Model, bu görevlerde en gelişmiş sonuçları gösterdi ve NLP topluluğunda geniş çapta benimsendi. Sıralı Sınıflandırma Modeli, sıralı verileri içeren Doğal dil işleme görevleri için güçlü bir araçtır ve sınıflandırma görevlerinin doğruluğunu artırma potansiyeline sahiptir.



Şekil 3.11. Sıralı sınıflandırma modellerinin çalışma prensibi (Chandio 2022).

Sıralı sınıflandırmada cümleler ön işleme (preprocessing) işleminden sonra, kelime gömme (word embedding) işlemi gerçekleşmektedir. Sıralı sınıflandırma mimarisi Şekil 3.11'de gösterilmiştir. Ön işleme sonrası kelime gömme, bir dil işleme teknolojisi yöntemidir ve bir kelimenin anlamsal temsili için matematiksel bir vektör oluşturur. Kelimeler genellikle bir kelime sözlüğünden çıkarılır ve bir vektör uzayında temsil edilir. Temsilleri oluşturulan kelimeler, yapay sinir ağları ile eğitilir. Ara katmanlar (hidden layers), yapay sinir ağlarında birbirine bağlı sinir hücrelerinden oluşan katmanlardır. Bu katmanlar, girdi katmanından çıktı katmanına kadar olan işleme aşamalarında bilgi işleme ve öğrenme gerçekleştirirler ve verilerin daha karmaşık özelliklerini ve ilişkilerini

öğrenebilmektedirler. Ara katmanların ardından çıkış katmanında olasılıkları hesaplanan vektörler için etiketler oluşturulmaktadır. Bu etiketler sayısal değerler taşımakta ve bu sayısal değerler pozitif, negatif, nötr gibi sınıfları temsil etmektedir.

3.5. Kullanılan Kütüphaneler

Çalışma Python dilinde gerçekleştirilmiş olup kullanılan bazı önemli kütüphaneler bu bölümde açıklanmaktadır.

3.5.1. PyTorch

PyTorch, 2017 yılında Meta AI tarafından Python programlama dilinde yazılmış açık kaynaklı bir makine öğrenimi kütüphanesidir (Paszke vd., 2017). PyTorch, araştırmacıların ve mühendislerin derin öğrenme modelleri oluşturmasını, eğitmesini ve dağıtmasını kolaylaştıran birçok araç sağlar. PyTorch'un en önemli özelliklerinden biri, dinamik hesaplama grafikleridir (Neubig vd., 2017). Bu grafikler, modelin işlem yaparken veriye göre değişen grafiğini oluşturur. Otomatik türevleme (automatic differentiation) özelliği sayesinde, model eğitiminde kullanılan gradyan hesaplamalarını otomatik olarak yapar (Paszke vd., 2017). Otomatik türevleme, bir fonksiyonun türevini hesaplamak için kullanılan bir yöntemdir. PyTorch'un autograd modülü, tensorler üzerindeki işlemlerin türevlerini otomatik olarak hesaplar ve gradyanı tutar. Bu sayede, model eğitimi sırasında gradyanın hesaplanması işlemi otomatik olarak yapılır ve daha hızlı bir eğitim süreci elde edilir. (Paszke vd., 2017).

Bu çalışmada eğitim için, pytorch kütüphanesi kullanılmıştır. İnput ids, segment ids, input mask ve label id’ler torch kütüphanesi aracılığı ile tensörlere dönüştürüldüler. Ardından verinin modele uygun hale getirlmesi için TensorDataset sınıfı kullanılmıştır. TensorDataset, PyTorch kütüphanesinde kullanılan bir sınıftır. Bu sınıf, modellerin eğitim ve doğrulama verilerini yüklemek için kullanılmıştır. Verilerin daha kolay yönetilmesini ve modellere daha kolay yüklenmesini sağlaması amacıyla tercih edilmiştir. BERT tabanlı modellerin eğitimi ve tahminleri sırasında kullanılan bu veri özellikleri aşağıda açıklanmıştır.

Input IDs: BERT tabanlı modelin girdi olarak kabul edeceği her bir kelimenin benzersiz bir kimliği olarak kullanılmıştır. Bu kimlikler, belirli bir alt kelime sözlüğüne dayalı olarak oluşturulmuş ve her kelime için bir tamsayı olarak atanmıştır.

Segment IDs: BERT, iki farklı metni işleyebilen çift metin sınıflandırma görevleri gibi bazı görevlerde kullanılmak üzere tasarlanmıştır (Devlin vd., 2018). Bu tür görevlerde, girdi olarak iki ayrı metin parçası alınır ve bunların hangi metne ait olduğunu belirlemek için segment kimlikleri kullanılmaktadır. Örneğin, 1 segment kimliği birinci metni, 0 segment kimliği ise ikinci metni temsil edebilir.

Input Mask: BERT tabanlı modelin dikkate alması gereken girdi kelime parçalarını belirlemek için kullanılan bir maskeleme mekanizması olarak kullanılmıştır. Input mask, gereksiz kısımları işaretleyerek BERT modelinin gereksiz bölümleri dikkate almamasına yardımcı olması amacıyla kullanılmıştır.

Label IDs: Modelin sınıflandırma görevlerinde tahmin edeceği etiketin benzersiz kimliği olarak kullanılmıştır. Örneğin, bir duygusal analiz görevinde, etiketler olumlu, olumsuz veya nötr etiketleri için 0,1,2 label id'leri kullanılmıştır. Bu, modelin sınıflandırma görevinde doğru sonuçlar vermesi için gereklidir.

```
all_input_ids = torch.tensor([f.input_ids for f in eval_features])
all_segment_ids = torch.tensor([f.segment_ids for f in eval_features])
all_input_mask = torch.tensor([f.input_mask for f in eval_features])
all_label_ids = torch.tensor([f.label_id for f in eval_features])
eval_data = TensorDataset(all_input_ids, all_segment_ids,
                          all_input_mask, all_label_ids)
```

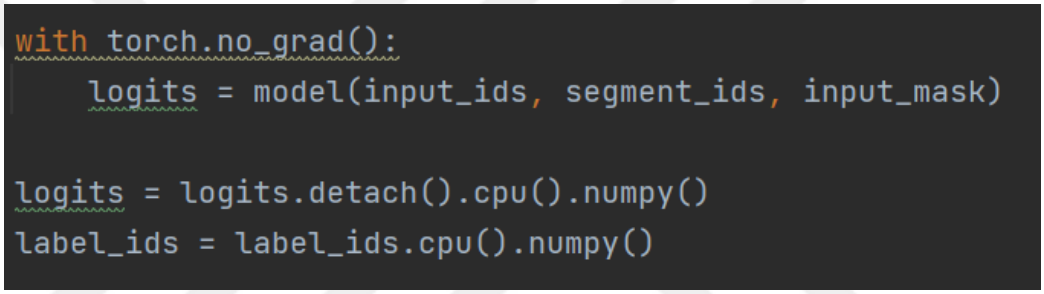
Şekil 3.12. Torch kütüphanesinin kullanımı.

Bu özellikler, modelin eğitimi ve tahminleri sırasında verilerin nasıl işlendiğini ve modelin nasıl öğrendiğini etkiler. Bu nedenle, veri hazırlama sürecinde bu özelliklerin doğru şekilde ayarlanması önemlidir. Şekil 3.12'de, çalışmada içerisinde torch kütüphanesinin kullanımı gösterilmiştir.

3.5.2. Numpy

NumPy, Python programlama dilinde bilimsel hesaplama ve veri analizi yapmak için kullanılan bir kütüphanedir. NumPy, büyük veri setleri ve matrisler üzerinde hızlı ve etkili bir şekilde işlem yapmayı sağlayan çok boyutlu dizileri (ndarray) içeren bir veri yapısı sunar (Van Der Walt vd., 2011).

NumPy, bilimsel hesaplama için birçok fonksiyon ve araç içermektedir. Bu araçlar arasında matematiksel fonksiyonlar, rastgele sayı üretimi, Fourier dönüşümleri, lineer cebir işlemleri ve diğer birçok bilimsel hesaplama işlemi yer alır. Bu araçlar makine öğrenimi uygulamalarında sıklıkla kullanılmaktadır.



Şekil 3.13. Numpy kütüphanesinin kullanımı.

Logits.detach() metodu, PyTorch gibi derin öğrenme çerçevelerinde kullanılan bir yöntem olup , bir tensorun hesaplanan gradient değerlerini hesaplamadan, tensorun bir kopyasını döndürür. Logits.detach() numpy ile kullanımına ilişkin kod örneği Şekil 3.13'te gösterilmiştir. Bu metot özellikle eğitim sırasında modelin parametrelerini güncellemek yerine, önceden eğitilmiş bir modelin çıktılarını almak için kullanılmıştır. Çalışmada önceden eğitilmiş modeller kullanılmış ve bu yöntemle model çıktıları numpy kütüphanesi ile hızlı şekilde içe aktarılmıştır.

3.5.3. Transformers

Transformers kütüphanesi, PyTorch veya TensorFlow gibi derin öğrenme çerçeveleri üzerinde çalışan bir Python kütüphanesidir. Transformers, önceden eğitilmiş Transformer tabanlı dil modellerini yüklemenize, inşa etmenize ve kullanmanıza olanak tanımaktadır (Rothman, 2021). Bu modeller, BERT, GPT-2, RoBERTa, T5, DistilBERT gibi birçok popüler modeli içermektedir. Bu modeller, geniş bir dil modeli eğitimi kümesi üzerinde önceden eğitilmiştir ve daha spesifik bir görev için ince ayar yapmak için kullanılabilmektedir. Transformers kütüphanesi, ayrıca model eğitimi için kolay bir kullanıcı arayüzü sağlamaktadır.

Transformer model ise İlk olarak Google'ın 2017 tarihli bir makalesinde (Vaswani vd., 2017), tanımlanan transformer modelleri, bugüne kadar icat edilmiş en yeni ve en güçlü model yapılarından biridir. Vaswani ve ekibi, makalede dikkat mekanizmasının kullanımını öneren ve önceden tanımlanmış dil işleme algoritmalarından daha iyi sonuçlar veren bir yapay zekâ modeli olan "Transformer"ı tanıtmışlardır (Vaswani vd., 2017).

Transformer modeli, birbirine bağlı iki modülden oluşur. Encoder, her bir kelimeyi bir vektöre dönüştürür ve ardından bu vektörleri bir dizi katman boyunca ileterek, her katmanda farklı bir şekilde işleyerek özetlemektedir. Decoder, hedeflenen çıktıyı oluşturmak için, Encoder tarafından oluşturulan vektörleri kullanır ve ayrıca, Encoder'ın ürettiği vektörlerin hangilerinin önemli olduğunu belirlemek için dikkat mekanizmasını kullanmaktadır (Vaswani vd., 2017).

Transformer modeli, önceki dil modelleri gibi, girdi metninin bir vektör temsilini oluşturur, ancak bu temsil oluşturma işlemi, önceki modellerde olduğu gibi bir "rekürsif sinir ağı" kullanmak yerine tamamen "dikkat" mekanizmasına dayanır. Bu, modelin tüm girdi metnini aynı anda işleyebilmesini ve daha önceki dil modellerinden çok daha hızlı olmasını sağlamaktadır (Vaswani vd., 2017).

```
from transformers import AutoTokenizer
from transformers import ConvBertForSequenceClassification
from transformers import get_linear_schedule_with_warmup
from transformers import AdamW

tokenizer =AutoTokenizer.from_pretrained("dbmdz/convbert-base-turkish-cased")
model = ConvBertForSequenceClassification.from_pretrained("dbmdz/convbert-base-turkish-cased",
                                                          num_labels=len(label_list))

optimizer = AdamW(model.parameters(),
                  lr=5e-5,
                  eps=1e-8
                  )
total_steps = len(train_dataloader) * epochs
scheduler = get_linear_schedule_with_warmup(optimizer,
                                            num_warmup_steps=0,
                                            num_training_steps=total_steps)
```

Şekil 3.14. Transformers kütüphanesinin kullanımı.

Çalışmada transformers kütüphanesinin kullanıldığı kod satırı Şekil 3.14'de gösterilmektedir. Kullanılan AutoTokenizer, AdamW, Get linear Schedule with warmup, SequenceClassification kütüphane fonksiyonların kullanım amacı aşağıda açıklanmıştır.

AutoTokenizer: Hugging Face kütüphanesi içindeki bir araçtır ve otomatik olarak veri setlerini belirli bir model için uygun hale getirmek için kullanılır. Bu işlemde, veri setindeki metinler belirli bir formata dönüştürülür, örneğin, token'lara bölünür ve maskelenir.

AdamW: bir tür optimizasyon algoritmasıdır ve özellikle derin öğrenme modellerinde sıkça kullanılır. AdamW, Adam optimizasyon algoritmasının bir türevidir ve ağırlıkları daha iyi bir şekilde güncellemek için L2 düzenlileştirme ile birleştirilir.

Get linear Schedule with warmup: bir öğrenme oranı planlama aracıdır ve ağırlıkların güncellenme sıklığını belirler. Bu yöntemde, öğrenme oranı, eğitim sürecinin başında yavaş yavaş artar ve daha sonra sabit bir seviyede kalır.

SequenceClassification: bir tür doğal dil işleme (NLP) görevidir ve metin verilerinin sınıflandırılmasını amaçlar. Bu görevde, veri setindeki metinler belirli bir sınıfa (örneğin, olumlu veya olumsuz) atanır.

3.5.4. Sklearn

Scikit-learn, Python programlama dilinde kullanılan açık kaynaklı bir makine öğrenimi kütüphanesidir. Bu kütüphane, sınıflandırma, regresyon, kümeleme, boyut azaltma ve model seçimi gibi birçok makine öğrenimi görevini kolaylaştıran araçlar ve algoritmalar içermektedir (Bisong, 2019).

Scikit-learn, Numpy ve Scipy gibi diğer popüler Python kütüphaneleri ile uyumlu olarak çalışır ve bu nedenle geniş bir veri işleme, görselleştirme ve manipülasyonu için uygun bir ekosisteme sahiptir. Ayrıca, hızlı ve verimli çalışma için Cython ile derlenmiş C kodunu kullanmaktadır.

Bu kütüphane, öğrenme algoritmalarının yanı sıra, özellik seçimi, model seçimi, veri ön işleme ve model değerlendirmesi gibi makine öğrenimi sürecinin diğer aşamalarında da kullanılacak araçlar ve yöntemler içermektedir. Çalışma içerisinde sklearn kütüphanesinin kullanımına ilişkin örnek kod satırı Şekil 3.15'te gösterilmektedir.

```
from sklearn.metrics import f1_score, precision_score, recall_score
from sklearn.metrics import accuracy_score
from sklearn.metrics import confusion_matrix, ConfusionMatrixDisplay
f_score = f1_score(full_label_ids, prediction_scores, average='macro')
precision = precision_score(full_label_ids, prediction_scores, average='macro')
recall = recall_score(full_label_ids, prediction_scores, average='macro')
acc = accuracy_score(full_label_ids, prediction_scores)
print("F-Score: ", f_score,"Recall: ", recall, "Precision: ", precision,'acc:',acc)
cm = confusion_matrix(full_label_ids, prediction_scores)
cmd_obj = ConfusionMatrixDisplay(cm, display_labels=['Pozitif', 'Negatif','Nötr'])
cmd_obj.plot()
cmd_obj.ax_.set(title='Karmaşıklık Matrisi',xlabel='Tahmin', ylabel='Gerçek')
```

Şekil 3.15. Sklearn kütüphanesinin kullanımı.

Sklearn kütüphanesi ile modellerin karşılaştırılmasında kullanılan metrikler aşağıda tanımlanmıştır. Metriklerin hesaplanmasında doğru pozitif (True Pozitive [TP]), doğru negatif (True Negative [TN]), yanlış pozitif (False Positive [NP]), ve yanlış negatif (False Negative [FN]) olmak üzere dört temel hesaplama kullanmaktadır.

TP, modelin pozitif bir sınıfı doğru bir şekilde tahmin ettiği durumken TN, Modelin negatif bir sınıfı doğru bir şekilde tahmin ettiği durumların hesaplanmasını içermektedir. FP ise modelin negatif bir sınıfı pozitif olarak yanlış bir şekilde tahmin ettiği durumu, FN, modelin pozitif bir sınıfı negatif olarak yanlış bir şekilde tahmin ettiği durumları içermektedir.

Kesinlik (Precision) : Kesinlik, gerçek pozitiflerin sayısının pozitif tahminlerin toplam sayısına bölümüdür. Modelin, tahmin edilen tüm pozitif vakalar arasından pozitif vakaları doğru bir şekilde belirleme yeteneğini ölçmektedir. Kesinlik değeri hesaplanırken TP ve FP sonuçları kullanılmaktadır. Bu sonuçlar kullanılarak kesinlik değerinin hesaplanması denklem 3.1'de gösterilmektedir.

$$Kesinlik\ (Precision) = \frac{TP}{TP + FP} \tag{3.1}$$

Duyarlılık (Recall) : Duyarlılık, gerçek pozitiflerin sayısının gerçek pozitiflerin toplam sayısına bölümüdür. Doğru tahmin edilip edilmediklerine bakılmaksızın, modelin tüm pozitif vakaları doğru bir şekilde belirleme yeteneğini ölçmektedir. Duyarlılık değeri hesaplanırken TP ve FN sonuçları kullanılır. Bu sonuçlar kullanılarak duyarlılık değerinin hesaplanması denklem 3.2'de gösterilmektedir.

$$Duyarlılık\ (Recall) = \frac{TP}{TP + FN} \tag{3.2}$$

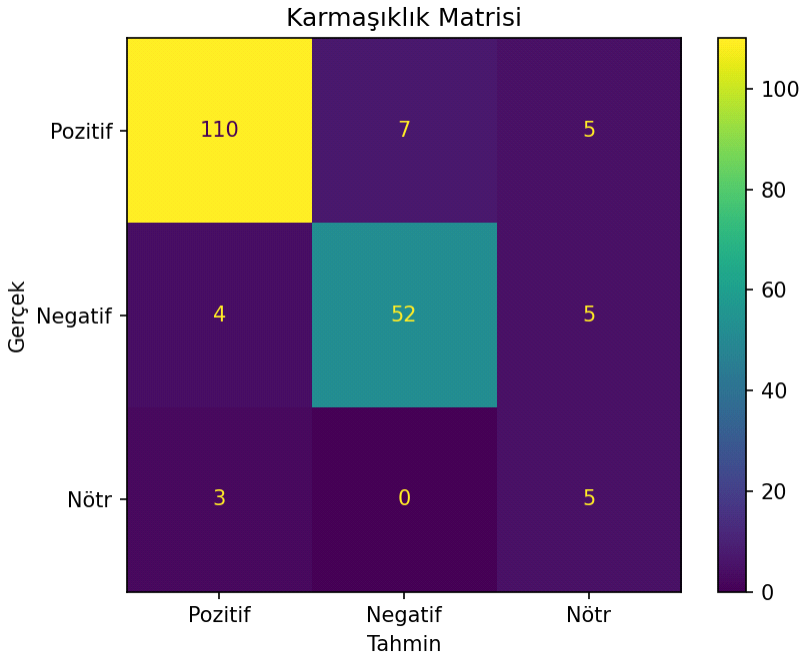
Doğruluk (Accuracy): Doğruluk, doğru tahmin sayısının toplam tahmin sayısına bölümüdür. Bir modelin tahminlerinin genel doğruluğunu ölçmektedir. Doğruluk değerinin hesaplanmasında kullanılan denklem 3.3'te gösterilmektedir.

$$Doğruluk\ (Accuracy) = \frac{TP + TN}{TP + FN + TN + FP} \tag{3.3}$$

F1 puanı (F1 Score) : F1 puanı, kesinlik ve duyarlılığın harmonik ortalamasıdır. Hem kesinliği hem de geri çağırmayı hesaba katan bir modelin doğruluğunun bir ölçüsüdür ve özellikle sınıflar dengesiz olduğunda model performansını değerlendirmek için önemli bir metriktir. F1 puanı hesaplanırken kesinlik (precision) ve duyarlılık (recall) değerleri kullanılır. F1 puanı hesaplanmasında kullanılan denklem 3.4'de gösterilmektedir.

$$F1\ Puanı(F1\ Score) = 2 * \frac{kesinlik * duyarlılık}{kesinlik + duyarlılık} \tag{3.4}$$

Karışıklık matrisi (Confusion Matrix) : Karışıklık matrisi, gerçek etiketleri tahmin edilen etiketlerle karşılaştırarak bir sınıflandırma modelinin performansını özetleyen bir tablodur. Gerçek pozitiflerin, gerçek negatiflerin, yanlış pozitiflerin ve yanlış negatiflerin sayısını göstermektedir. Karmaşıklık matrisinin kullanımına ait örnek görsel Şekil 3.16'da sunulmuştur.



Şekil 3.16. Karmaşıklık matrisinin kullanımı.

Karışıklık matrisi görüntüsü (Confusion Matrix Display): Karışıklık matrisi görüntüsü, sonuçların yorumlanmasını kolaylaştıran bir karışıklık matrisinin görsel bir temsilidir. Sayıları genellikle renk kodlu bir biçimde gösterir ve kesinlik, hatırlama ve F1 puanı gibi ek ölçümler içerebilir.

Bu fonksiyonlar dışında eğitim sonuçlarını değerlendirmek için, makro ortalama (macro avg) ve ağırlıklı ortalama (weighted avg) hesaplanmıştır.

Makro ortalama (Macro avg): Her sınıfın doğruluk oranını ayrı ayrı hesaplar ve ardından tüm sınıfların doğruluk oranlarının ortalamasını alır. Bu, tüm sınıfların performansına eşit ağırlık verir.

Ağırlıklı ortalama (Weighted avg): Her sınıfın doğruluk oranını ayrı ayrı hesaplar ve ardından her sınıfın örnek sayısına göre ağırlıklandırılarak bir ağırlıklı ortalama hesaplar. Bu, sınıfların örnek sayısına göre performansına ağırlık verir ve daha büyük sınıfların performansı daha fazla etki eder.

3.5.5. NLTK

Doğal dil araç seti (Natural Language Toolkit [NLTK]) doğal dil işleme konusunda bir Python kütüphanesidir. NLTK, öğrenme, sınıflandırma, veri görselleştirme ve metin madenciliği gibi birçok NLP görevi için kullanılabilen açık kaynaklı bir araçtır (Bird, 2006).

NLTK, doğal dil verilerini okumak, işlemek, analiz etmek ve manipüle etmek için önceden tanımlanmış araçlar ve algoritmalar sunmaktadır. NLTK'nin içerisinde, dilbilgisi analizi, kelime dağılım analizi, anlamsal analiz, kelime kökü çıkarma, dil modelleme ve belge sınıflandırma gibi birçok araç ve algoritma bulunmaktadır. Bu araçlar, metin verilerinin daha iyi anlaşılmasına, özetlenmesine ve sınıflandırılmasına yardımcı olmaktadır. NLTK kütüphanesi çalışma içerisinde gerekli olmayan sözcüklerin (stop word) veri setinden temizlenmesi için kullanılmıştır.

```
import nltk
from nltk.stem import WordNetLemmatizer
from nltk.tokenize import word_tokenize
nltk.download("punkt")
nltk.download('wordnet')
nltk.download('omw-1.4')
nltk.download('stopwords')
stop=set(stopwords.words('turkish'))
lem=WordNetLemmatizer()
for news in text:
    words=[w for w in word_tokenize(news) if (w not in stop)]
    words=[lem.lemmatize(w) for w in words if len(w)>2]
```

Şekil 3.17. NLTK kütüphanesinin kullanımı.

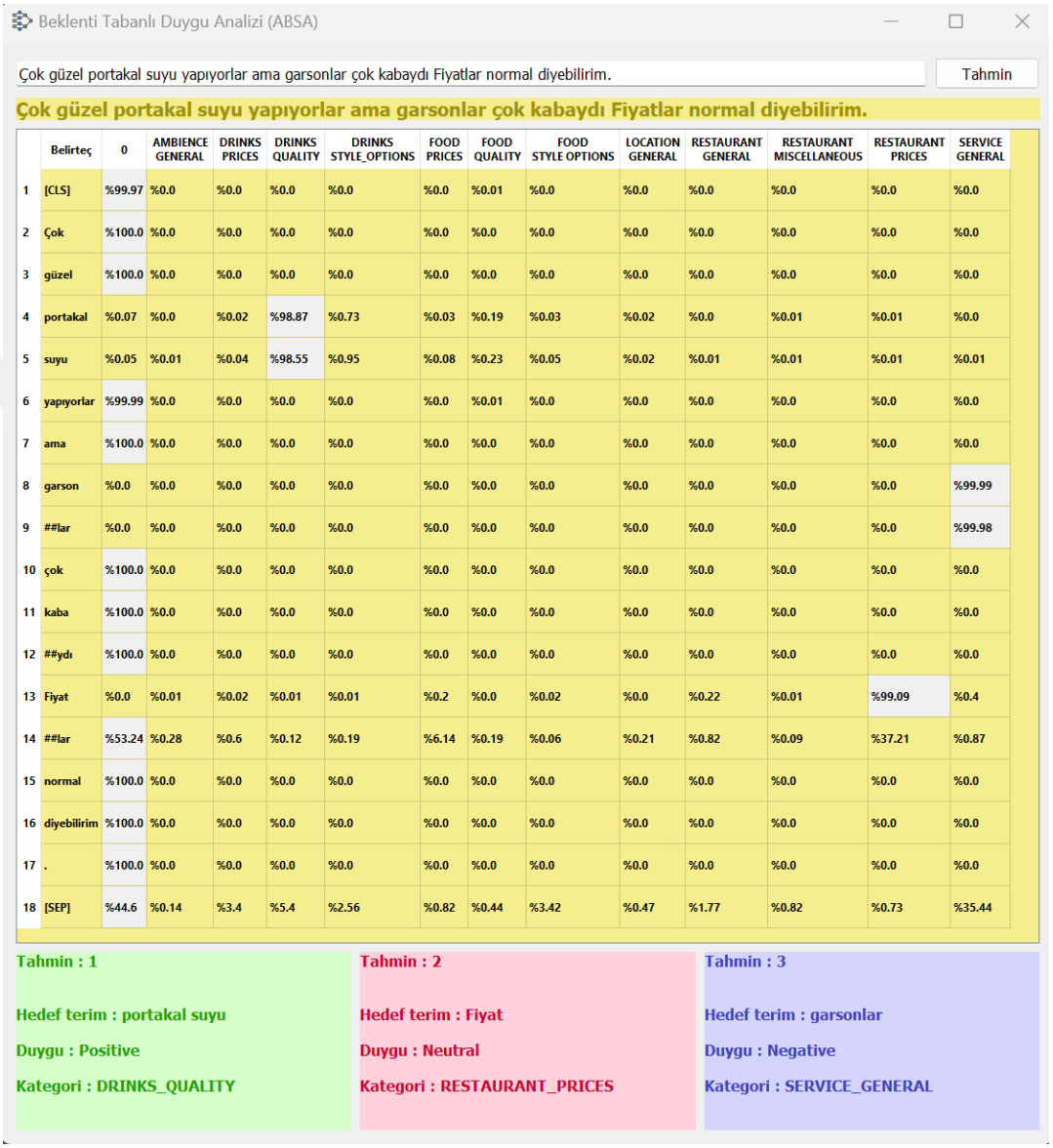
Yapılan çalışmada NLTK kütüphanesinin çalışma içerisinde kullanımına örnek Şekil 3.17'de gösterilmektedir. Kod satırında kullanılan WordNetLemmatizer, kelime çıkarma işlemini gerçekleştiren bir araçtır. Bu araç, bir kelimenin kökünü (lemma) bulmak için kullanılmaktadır, Word Tokenize ise, bir metin dizesini kelimelere ayırarak bir liste olarak döndürmektedir. Bu fonksiyon, kelime ayırma işlemi sırasında özel karakterleri, noktalama işaretlerini ve diğer ayrıntıları da dikkate alır. Cümleler kelimelere ayrıldıktan sonra WordNetLemmatizer fonksiyonu ile eklerden ayrılmışlardır. Ardından gereksiz kelimeler listesi ile eşleşen kelimeler veri setinden atılmıştır.

3.5.6. PyQt5

Python programlama dili, geniş bir kullanıcı kitlesine sahip popüler bir dil olmasının yanı sıra birçok farklı alanlarda kullanılmaktadır. Bu nedenle, Python programcıları sıklıkla kullanıcı arayüzleri, grafikler ve veri görselleştirmesi gibi GUI (Graphical User Interface) araçlarının kullanımına ihtiyaç duymakadır. PyQt5, Python programlama dilinde GUI geliştirmek için kullanabilecek bir framework sağlamaktadır (Willman, 2021).

PyQt5 ile birçok farklı GUI öğesi, widget'lar ve araçlar sunulmaktadır. PyQt5, düğmeler, metin kutuları, etiketler, sekmeler, menüler, diyalog kutuları, çizim araçları, grafikler ve daha birçok araç ile gelmektedir.

Bu çalışmada pyqt5 framework, ABSA için kullanıcı arayüzü oluşturmak amacıyla kullanılmıştır. Çalışma için pyqt5 ile hazırlanan arayüz Şekil 3.18'deki gösterilmektedir.



	Belirteç	0	AMBIENCE GENERAL	DRINKS PRICES	DRINKS QUALITY	DRINKS STYLE_OPTIONS	FOOD PRICES	FOOD QUALITY	FOOD STYLE OPTIONS	LOCATION GENERAL	RESTAURANT GENERAL	RESTAURANT MISCELLANEOUS	RESTAURANT PRICES	SERVICE GENERAL
1	[CLS]	%99.97	%0.0	%0.0	%0.0	%0.0	%0.0	%0.01	%0.0	%0.0	%0.0	%0.0	%0.0	%0.0
2	Çok	%100.0	%0.0	%0.0	%0.0	%0.0	%0.0	%0.0	%0.0	%0.0	%0.0	%0.0	%0.0	%0.0
3	güzel	%100.0	%0.0	%0.0	%0.0	%0.0	%0.0	%0.0	%0.0	%0.0	%0.0	%0.0	%0.0	%0.0
4	portakal	%0.07	%0.0	%0.02	%98.87	%0.73	%0.03	%0.19	%0.03	%0.02	%0.0	%0.01	%0.01	%0.0
5	suyu	%0.05	%0.01	%0.04	%98.55	%0.95	%0.08	%0.23	%0.05	%0.02	%0.01	%0.01	%0.01	%0.01
6	yapıyorlar	%99.99	%0.0	%0.0	%0.0	%0.0	%0.0	%0.01	%0.0	%0.0	%0.0	%0.0	%0.0	%0.0
7	ama	%100.0	%0.0	%0.0	%0.0	%0.0	%0.0	%0.0	%0.0	%0.0	%0.0	%0.0	%0.0	%0.0
8	garson	%0.0	%0.0	%0.0	%0.0	%0.0	%0.0	%0.0	%0.0	%0.0	%0.0	%0.0	%0.0	%99.99
9	##lar	%0.0	%0.0	%0.0	%0.0	%0.0	%0.0	%0.0	%0.0	%0.0	%0.0	%0.0	%0.0	%99.98
10	çok	%100.0	%0.0	%0.0	%0.0	%0.0	%0.0	%0.0	%0.0	%0.0	%0.0	%0.0	%0.0	%0.0
11	kaba	%100.0	%0.0	%0.0	%0.0	%0.0	%0.0	%0.0	%0.0	%0.0	%0.0	%0.0	%0.0	%0.0
12	##ydı	%100.0	%0.0	%0.0	%0.0	%0.0	%0.0	%0.0	%0.0	%0.0	%0.0	%0.0	%0.0	%0.0
13	Fiyat	%0.0	%0.01	%0.02	%0.01	%0.01	%0.2	%0.0	%0.02	%0.0	%0.22	%0.01	%99.09	%0.4
14	##lar	%53.24	%0.28	%0.6	%0.12	%0.19	%6.14	%0.19	%0.06	%0.21	%0.82	%0.09	%37.21	%0.87
15	normal	%100.0	%0.0	%0.0	%0.0	%0.0	%0.0	%0.0	%0.0	%0.0	%0.0	%0.0	%0.0	%0.0
16	diyebilirim	%100.0	%0.0	%0.0	%0.0	%0.0	%0.0	%0.0	%0.0	%0.0	%0.0	%0.0	%0.0	%0.0
17	.	%100.0	%0.0	%0.0	%0.0	%0.0	%0.0	%0.0	%0.0	%0.0	%0.0	%0.0	%0.0	%0.0
18	[SEP]	%44.6	%0.14	%3.4	%5.4	%2.56	%0.82	%0.44	%3.42	%0.47	%1.77	%0.82	%0.73	%35.44

Şekil 3.18. Arayüz tasarımı.

Arayüzden girdi olarak alınan cümle veya cümleler, tahmin butonuna basıldığında ABSA gerçekleştirmekte ve verilen cümlelere ait hedef terim, belirteçlerin tahmin skorları, duygu, kategori çıktılarını ekran üzerinde göstermektedir.

Belirteçlerin skorları yüzdelik olarak hesaplanmıştır. Tahminlenen logit değerleri tensörlerden, numpy array'e dönüştürülmüş, ardından softmax fonksiyonu ile 0-1 arasında değerleri belirlenmiş, 100 ile çarpılarak yüzdelik skorlar elde edilmiştir. Belirtece ait en yüksek skor, hangi kategori sınıfına ait olduğunu göstermektedir. 0 ile temsil edilen belirteçler, önceden belirlenen herhangi bir kategori sınıfına atanamadığını ifade etmektedir. Örneğin Şekil 3.18'de "Fiyat" kelimesi %99 ile RESTAURANT PRICES sınıfına ait olduğu tahminlenmiş, "Fiyat" kelimesine ek olarak gelen "lar" ise, %37,21 RESTAURANT PRICES, %53,24 hiçbir kategoriye ait olmadığı tahminlenmiştir. "garson" kelimesine eklenen "lar" eki ise, model tarafından anlamlandırılmış ve SERVICE GENERAL kategori sınıfına dahil edilmiştir. Bunun gibi örnekler için veri setinin zenginleştirilmesi, tahminleme başarısını artıracaktır.

3.6. Kullanılan Hiperparametreler

Grup boyutu (Batch size): Grup boyutu, yapay sinir ağlarında kullanılan bir kavramdır ve ağın eğitimi sırasında kaç örneğin aynı anda işleneceğini belirlemektedir. Bir batch, ağa verilen veri kümesinin bir parçasıdır ve birden fazla örnek içermektedir. Bu örnekler ağa bir arada verilmekte ve ağ, her bir örnek için ayrı ayrı hesaplanan kayıp fonksiyonu değerlerinin ortalamasını kullanarak geriye doğru yayılımını gerçekleştirmektedir. Grup boyutu değeri, birçok faktörden etkilenmekte ve ağın performansını önemli ölçüde etkilemektedir. Küçük grup boyutu değerleri, ağın daha hızlı öğrenmesini sağlamakta ancak hesaplama işlemleri daha yavaş gerçekleşmektedir. Büyük grup boyutu değerleri ise hesaplama işlemlerini hızlandırmakta ancak ağın öğrenme hızını azaltabilmektedir. Ayrıca, daha büyük grup boyutu değerleri, daha fazla bellek kullanımı gerektirmektedir. Bu çalışmada gpu belleğinin maksimum kapasitesine uygun batch değeri belirlenmiştir.

Adım Sayısı (Epoch): Adım sayısı terimi, makine öğrenmesi ve yapay sinir ağları gibi alanlarda kullanılan bir kavramdır. Bir eğitim veri kümesindeki tüm örneklerin ağ tarafından bir kez işlenmesiyle oluşan bir adımı ifade etmektedir.

Yani bir adım (epoch), veri kümesindeki tüm örneklerin ağ tarafından işlenmektedir. Örneğin, bir veri kümesinde 10.000 örnek varsa ve eğitim sırasında grup boyutu (batch size) değeri 100 ise, bir adımda (epoch) 100 batch işlenecektir. Bu durumda 10.000

örneğin tamamının ağ tarafından işlenmesi için 100 adım gerekmektedir.

Adım değeri, eğitim sırasında ağın performansını önemli ölçüde etkilemektedir. Daha yüksek adım değerleri, ağın daha uzun süre eğitilmesini sağlamakta ve daha iyi sonuçlar elde edilmektedir. Daha düşük adım değerleri ise ağın daha az eğitilmesini sağlamakta ve daha hızlı sonuç alınmaktadır. Ancak, daha az epoch değeri kullanmak ağın tamamını öğrenmesine engel olma riski taşımaktadır.

Adım değeri, ağın eğitiminde birçok faktörden etkilenir. Bu faktörler arasında veri kümesinin boyutu, grup boyutu (batch size) değeri, ağın karmaşıklığı, öğrenme oranı ve kullanılan optimizasyon algoritması yer almaktadır. Bu faktörlerin her biri, ağın optimum adım değerini (epoch) etkilemekte ve optimum değeri bulmak genellikle deneme yanılma yöntemi ile elde edilmektedir.

Öğrenme puanı (Learning rate): Öğrenme puanı (learning rate), makine öğrenmesi ve yapay sinir ağlarında kullanılan bir hiperparametredir. Öğrenme puanı, ağın her bir adımında güncellenecek olan ağırlıkların ne kadar değiştirileceğini belirlemektedir.

Bir yapay sinir ağı, bir dizi katmandan oluşur ve her katmanda ağırlıklar bulunmaktadır. Bu ağırlıklar, ağın veriler üzerindeki tahminlerini hesaplamak için kullanılmaktadır. Eğitim sürecinde, ağa verilen veriler kullanılarak kayıp (hata) hesaplanmakta ve bu kaybın azaltılması için ağırlıklar güncellenmektedir. Bu güncelleme, öğrenme puanı ve gradyan inişi gibi teknikler kullanılarak yapılabilmektedir.

Öğrenme puanı, ağın güncellenmesindeki adım büyüklüğünü belirlemektedir. Eğer öğrenme puanı çok yüksekse, ağırlıkların güncellenmesi çok büyük adımlarla gerçekleşmekte ve bu ağın minimum hataya ulaşmasını zorlaştırabilmektedir. Öte yandan, eğer öğrenme puanı çok düşükse, ağın minimum hataya ulaşması çok yavaş olmaktadır.

Öğrenme puanı, her veri seti ve problem için farklı olabilmektedir. Bu değer, genellikle deneme yanılma yoluyla belirlenmekte ve ağın performansını önemli ölçüde etkilemektedir. Doğru bir öğrenme puanı seçmek, ağın hızlı bir şekilde eğitilmesini ve optimum performans elde etmesini sağlayabilmektedir.

Optimizasyon (Optimizer): Transformatör **(**transformers) için yaygın olarak kullanılan optimizasyonlardan biri, "Adam" olarak adlandırılan adaptif momentum yöntemidir. Adam, gradyan iniş (gradient descent) optimizasyon yöntemlerinin bir türüdür ve optimize edilecek fonksiyonun hesaplanan gradyanını kullanarak modelin ağırlıklarını güncellemektedir.

Adam optimizasyonu, her bir ağırlığın farklı bir öğrenme oranına (learning rate) sahip olmasına izin vermekte ve ağırlıkların birikimli gradyanlarına da dikkat ederek, ağırlıkların güncellenmesinde momentum hesaplamaları yapmaktadır. Bu özellikleri sayesinde, Adam optimizasyonu, ağın eğitim sürecinde daha hızlı ve daha istikrarlı sonuçlar verebilmektedir.

Diğer optimizasyonlar arasında Skolastik Gradyan İnişi (Stochastic Gradient Descent [SGD]), "Adagrad", "Adadelta" ve "RMSProp" gibi yöntemler bulunur. Optimizer seçimi, ağın özelliklerine ve eğitim verilerine göre değişebilmektedir.

Maksimum cümle uzunluğu: Modellerin eğitiminde, maksimum cümle uzunluğu önemlidir çünkü modelin performansını etkileyebilmektedir.

Maksimum cümle uzunluğu, modelin girdi olarak kabul edebileceği en uzun cümle boyutunu belirlemektedir. Örneğin, 512 belirteçlik (token) bir maksimum cümle uzunluğu belirlenmişse, model sadece 512 belirtece (token) kadar olan cümleleri kabul eder ve daha uzun cümleleri keser. Bu, modelin bellek kullanımını sınırlamakta ve eğitim sürecini hızlandırabilmektedir. Ancak, maksimum cümle uzunluğunun çok küçük bir değere ayarlanması durumlarında, veri kaybı oluşabilmektedir. Bu yüzden bu değerin doğru tanımlanması önem taşımaktadır.

Isınma oranı (Warm up proportion): Isınma oranı, özellikle büyük doğal dil işleme modellerinde sıklıkla kullanılan bir eğitim tekniğidir ve öğrenme oranı (learning rate) eğitim başlangıcında küçük bir değerle başlayarak zamanla artırılmaktadır. Bu artışın ne kadar süreceği ise ısınma oranı tarafından belirlenmektedir ve genellikle %5 ila %20 arasında bir değer seçilmektedir. Örneğin, eğer eğitimde 10.000 adım yapılacaksa ve %10'luk bir ısınma oranı seçilirse, ilk 1.000 adım boyunca öğrenme oranı yavaş yavaş arttırılmaktadır. Sonraki 9.000 adımda ise sabit bir öğrenme oranı kullanılmaktadır. Bu teknik, modelin başlangıçta daha düşük öğrenme oranlarıyla eğitilerek stabil bir eğitim süreci geçirmesini ve daha sonra öğrenme oranının artmasıyla daha verimli bir şekilde öğrenmesini sağlamaktadır. Aksi takdirde, öğrenme oranının çok hızlı artması, modelin aşırı uyum sağlamasına (overfitting) ve eğitim sürecinde istenmeyen dalgalanmalara neden olabilmektedir.

Epsilon: Epsilon Adam optimizasyon algoritmasında kullanılan bir parametredir. Adam, gradient tabanlı bir optimizasyon algoritması olup ve öğrenme puanını (learning rate) dinamik olarak ayarlayarak eğitim sürecini hızlandırmayı hedeflemektedir. Epsilon değeri, Adam algoritmasında kullanılan bir hesaplama aşamasında sıfıra bölünme hatasını önlemek için kullanılmaktadır.
Epsilon değeri, modelin eğitiminde önemlidir çünkü bu değer, optimizasyon algoritmasının doğru bir şekilde çalışmasını sağlamakta ve eğitim sürecinde hataları önlemektedir.

3.7. Deneysel Çalışmalar

Bu bölümde veri kümesi hakkında genel bilgiler, gerçekleştirilen belirteç sınıflandırma ile kategori tahminlenmesine ait kullanılan yöntemler ve deney sonuçları, Sıralı sınıflandırma modeli ile duygu analizine ait kullanılan yöntemler ve deneylerin sonuçları sunulacaktır. Tüm deneyler Monster Tulpar T5 V21.7 üzerinde, NVIDIA GeForce RTX 3070 ekran kartı üzerinde gerçekleştirilmiştir.

3.7.1. Veri kümesi

Bu çalışmada veri kümesi olarak eğitim için SemEval-2016 ABSA restoran eğitim verisi (Pontiki vd., 2016), test içinse SemEval-2016 ABSA restoran test (Pontiki vd., 2016), + SemEval-2016 ABSA restoran deneme verisi (Pontiki vd., 2016) kullanılmıştır.

```
<text>Manzara sahane evet ama servis rezalet.</text>
<Opinions>
   <Opinion target="servis" category="SERVICE#GENERAL" polarity="negative" from="24" to="30" />
   <Opinion target="Manzara" category="AMBIENCE#GENERAL" polarity="positive" from="0" to="7" />
</Opinions>
```

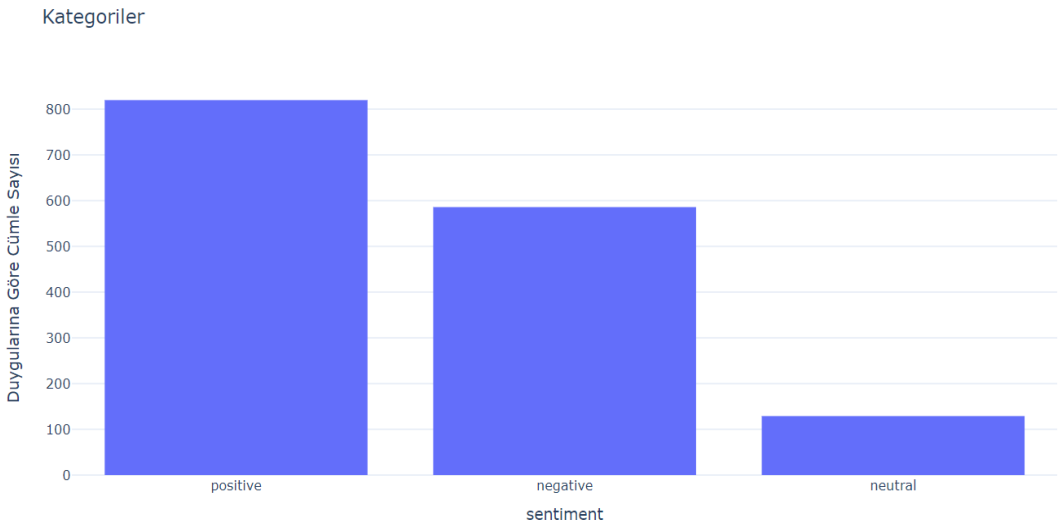
Şekil 3.19. Veri setine ait örnek.

Eğitim verisi için 1535 cümle kullanılmıştır. Şekil 3.19'da gösterildiği gibi, bir cümle birden fazla hedef terim, kategori ve duyguya ait olabilmektedir. Toplam 12 farklı kategori bulunmaktadır. Test verisi için 191 cümle kullanılmıştır. Test ve eğitim verisinin kategori bazında sayısal dağılımı Çizelge 3.1'de gösterilmiştir.

Çizelge 3.1. Eğitim ve test verisinin kategorilere göre sayısal dağılımı.

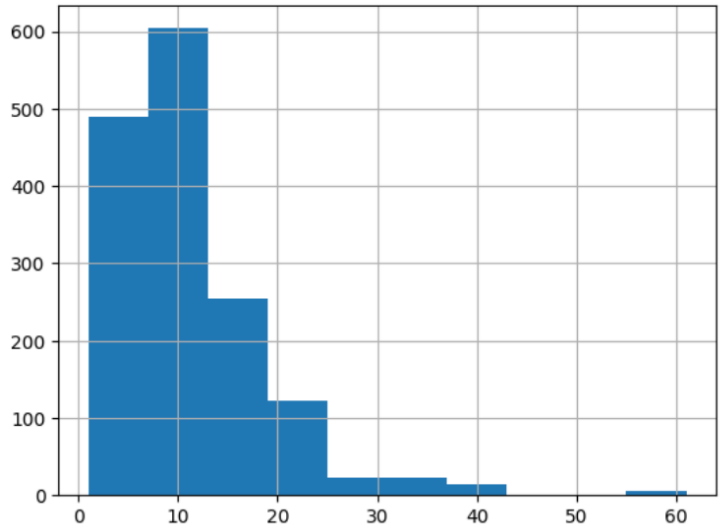
Kategori	Eğitim verisi sayısı	Test verisi sayısı
FOOD#QUALITY	446	77
AMBIENCE#GENERAL	277	31
SERVICE#GENERAL	235	25
RESTAURANT#GENERAL	228	22
FOOD#STYLE_OPTIONS	123	10
RESTAURANT#PRICES	65	9
DRINKS#QUALITY	51	6
LOCATION#GENERAL	39	4
DRINKS#STYLE_OPTIONS	30	1
FOOD#PRICES	31	4
DRINKS#PRICES	9	1
RESTAURANT#MISCELLANEOUS	1	1

Eğitim verisinde 1535 cümlenin 820'si pozitif, 586'i negatif, 129 nötr duygu içermektedir. Veri setinin duygulara göre sayısal dağılımı Şekil 3.20'de gösterilmiştir.



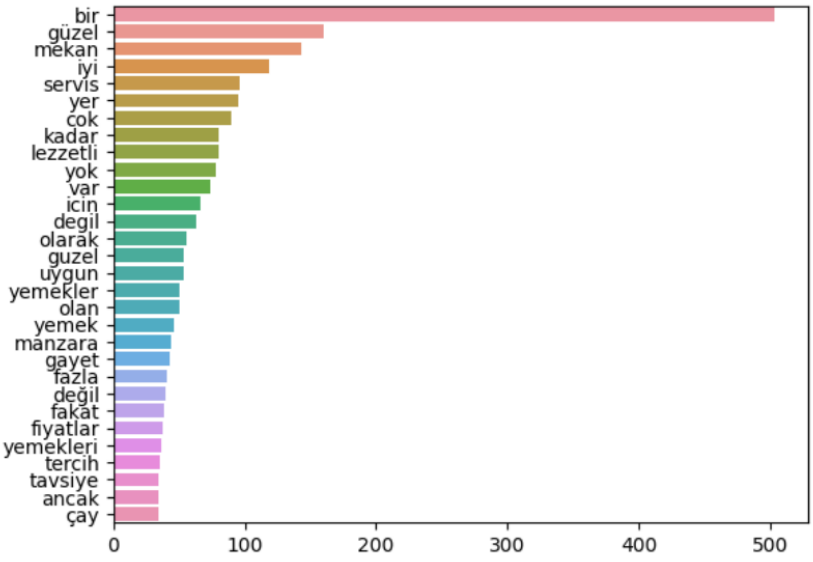
Şekil 3.20. Duygularına göre cümle sayısı.

Cümlelerin belirli oranda benzersiz kelime içermesi kategorilere ayırma işleminde önemli rol oynamaktadır. Sınıfların ayrışmasında belirleyici kelime vektörleri oluşturmaktadır. Şekil 3.21'de cümlelerin içerdiği benzersiz kelime sayısının cümle sayısına göre dağılımı gösterilmektedir. Veri setindeki cümlede benzersiz kelime sayısı ortalaması 9 kelime olarak hesaplanmıştır.



Şekil 3.21. Cümlelerin içerdikleri benzersiz kelime sayıları.

Gelecek olan her kelimenin olasılığı kelimenin frekansı ile ilişkilidir. Kelimenin kullanım sıklığı bize öncesinde ve sonrasında gelen kelimelerinde tahmini hakkında fikir vermektedir. Ağırlıklı olarak servis, mekan, lezzet, yemek, manzara gibi kelimeleri içeren cümlelerin daha fazla olduğu Şekil 3.22'de görülmektedir. Bu analiz, eğitim sonrasında bu terimleri içeren cümlelerin daha başarılı tahminleneceği ile ilgili bize ipuçları vermektedir.



Şekil 3.22. Veri setinde en çok bulunan terimler (unigram).

3.7.2. Belirteç sınıflandırma ile kategori tespiti

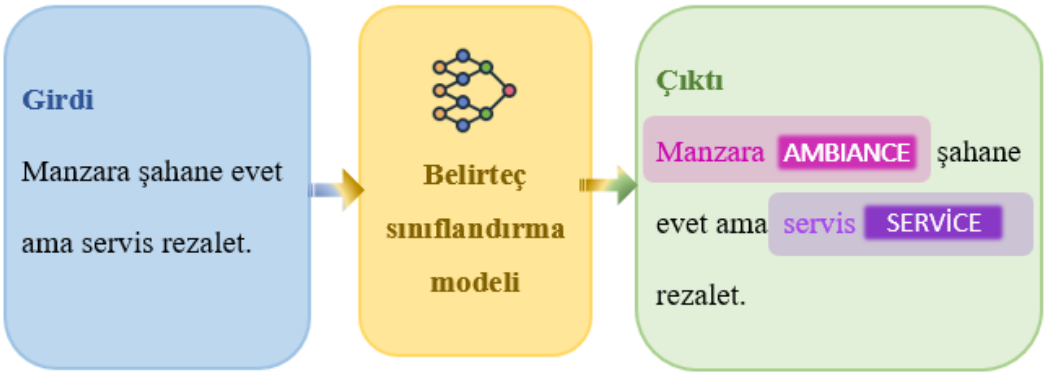
Kategori tahminlemesini gerçekleştirmek için belirteç sınıflandırma yöntemi kullanılmıştır. Sınıflandırma için gerçekleştirilen adımlar aşağıdaki gibidir.

- Veri seti cümleler, kategoriler, hedef terimler olmak üzere 3 sütun içeren DataFrame'e dönüştürülmüştür.
- Cümleler belirteçlere ayrılmıştır.
- Ayrılan belirteçler içerisinden hedef terimler tespit edilmiştir.
- Hedef terim içeren belirteçlerin yerine ilgili kategorinin ismi yazılmıştır.
- Kategori içeren belirteçlerin haricinde tüm belirteçlere 0 değeri atanmıştır. İşleme ait örnek Çizelge 3.2'de gösterilmiştir.

Çizelge 3.2. Belirteç sınıflandırma etiketlenmiş cümle örneği.

Cümle : Manzara güzel ,yemekler başarılı ama servis berbat. Belirteçler : [Manzara], [güzel], [yemekler], [başarılı], [ama], [servis], [berbat]	
Manzara	AMBIENCE#GENERAL
çok	0
güzel	0
yemekler	FOOD#QUALITY
başarılı	0
ama	0
servis	SERVICE#GENERAL
berbat	0

- Veri, modelin maksimum giriş uzunluğundan fazla olmayacak şekilde kırpılmıştır.
- Ön işleme işlevini tüm veri kümesine uygulamak için haritalama işlevi kullanılmıştır.
- Ön işleme uygulanan belirteçler ve etiketler, modele girdi olarak verilmiştir.
- Metriklere göre öğrenmenin tamamlandığı noktada eğitim sonlandırılmıştır.
- Çıktı adlandırılmış varlıklara sahip açıklamalı metindir.



Şekil 3.23. Belirteç sınıflandırma modeli için girdi ve çıktı.

Gerçekleştirilen sınıflandırma işlemine ait girdi ve çıktılara ilişkin örnek Şekil 3.23'te gösterilmektedir.

Belirteç sınıflandırması için BERT, ConvBERT, ELECTRA, DeBERTa, DistilBERT olmak üzere 5 farklı model kullanılmış ve karşılaştırılmıştır. Tüm eğitimler Python dilinde gerçekleştirilmiştir. Modellerin hiperparametreleri, batch size (grup boyutu) 16, learning rate (öğrenme puanı) $2e^{-5,}$ epoch (adım sayısı) 20 olarak kullanılmıştır.

3.7.3. Sıralı sınıflandırma modeli ile duygu analizi

Cümle içindeki hedef terimlerin bağlı oldukları duyguları tespit etmek için sequantial model (sıralı sınıflandırma modeli) kullanılmıştır. Eğitim için kullanılan adımlar aşağıdaki gibidir.

- Veri seti içerisindeki cümleler, hedef terimler ve duygular json formatına dönüştürülmüştür.
- Veriler içeri aktarılarak, modele uygun girdiler haline getirildi.
- Tokenizer ve modeller tanımlandı.
- Hiperparametreler belirlendi. Belirlenen hiperparametreler tüm modeller için ortaktır.
- Beş farklı model ile eğitim gerçekleştirildi.
- Modellere ait kayıp fonksiyonu grafikleri matplotlib aracılığı ile çizdirildi.
- Logit ve label sonuçları kaydedildi.
- Test verisi üzerinde modeller test edilerek, karmaşıklık matrisleri ve sonuç metrikleri incelendi.

Sıralı sınıflandırma ile duygu analizi için BERT, DistilBERT, ConvBERT, ELECTRA olmak üzere toplam 4 model kullanılmıştır.

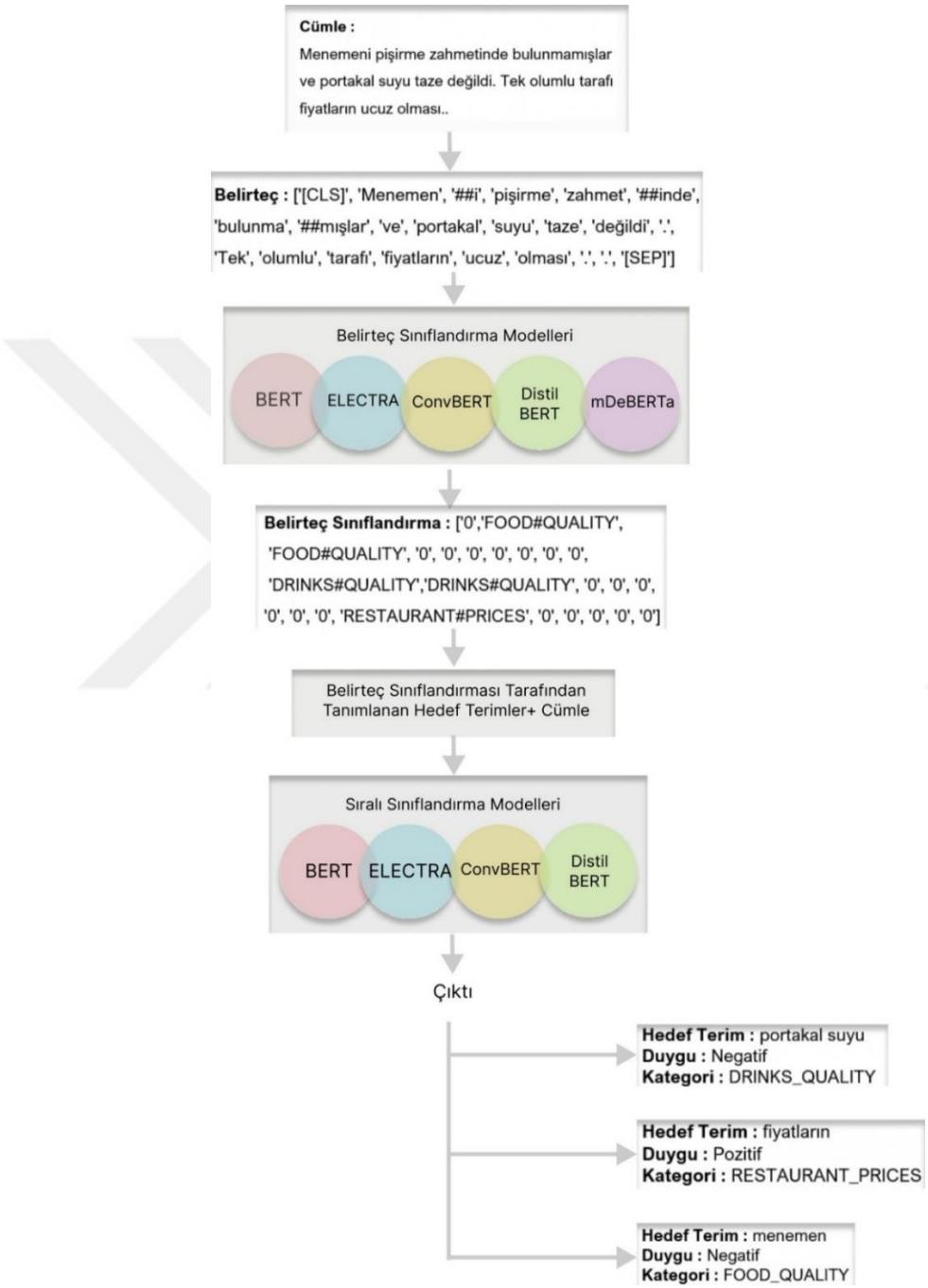
Çizelge 3.3. Eğitimde kullanılan hiperparametrelerin değerleri.

Parametre İsimleri	**Parametreler**
Batch size (Grup boyutu)	16
Epoch (adım sayısı)	16
Learning rate (Öğrenme puanı)	5e -5
Optimizer	AdamW
Maksimum cümle uzunluğu	200 karakter
Warm up proportion(Isınma oranı)	0.1
Epsilon	1e -8

Tüm eğitimler Python dili ile gerçekleştirilmiştir. Eğitim için kullanılan bazı parametreler Çizelge 3.3'te gösterilmiştir. Gerçekleştirilen eğitim sonuçları kesinlik, duyarlılık, F1-Skor, doğruluk metriklerine göre değerlendirilmiştir.

3.7.4. Hedef terim, kategori ve duygunun aynı birlikte tespit edilmesi

Hedef terim, kategori ve duyguların birlikte tespit edilmesi için gerçekleştirilen adımlar Şekil 3.24'te gösterilmektedir.



Şekil 3.24. Hedef terim, kategori ve duyguların aynı anda tespit edilmesi.

Cümleler belirteçlere ayrılmasının ardından, belirteç sınıflandırma modellerden birisi ile tahminlenir ve belirteçlere karşılık gelen kategoriler (AMBIENCE GENERAL, DRINKS

PRICES, DRINKS QUALITY, DRINKS STYLE OPTIONS, FOOD PRICES, FOOD QUALITY, FOOD STYLE OPTIONS, LOCATION GENERAL, RESTAURANT GENERAL, RESTAURANT MISCELLANEOUS, RESTAURANT PRICES, SERVICE GENERAL) belirlenir. Belirli bir kategoriye ait olduğu belirlenen belirteçler, hedef terimler olarak kaydedilir. Belirlenen hedef terimler, cümle ile birlikte sıralı sınıflandırma modellerinden birisine modele uygun girdi haline dönüştürülerek gönderilir. Sıralı sınıflandırma modeli girdi olarak verilen cümle ve hedef terim üzerinde duygu analizi gerçekleştirir.

4. BULGULAR

Bu bölümde belirteç sınıflandırma ve sıralı sınıflandırma modellerinin eğitimleri sonucunda elde edilen bulguların kesinlik, duyarlılık, F1 skor, doğruluk, makro ortalama, ağırlıklı ortalama gibi bazı metriklere göre yüzdelik skorları ve grafikleri sunulacaktır.

4.1. Belirteç Sınıflandırma ile Kategori Tespitine ait Bulgular

Belirteç sınıflandırma ile kategori tespitinde BERT ConvBERT, ELECTRA, DeBERTa, DistilBERT olmak üzere 5 farklı model kullanılmış ve karşılaştırılmıştır.

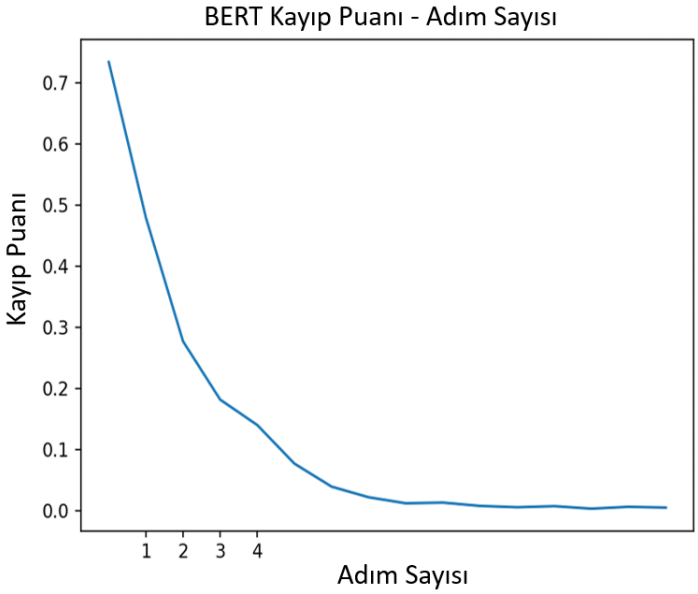
Belirteç sınıflandırma ile kategori tespiti için gerçekleştirilen eğitimin kesinlik, duyarlılık, F1-Skor, doğruluk metriklerine göre sonuçları Çizelge 4.1'de gösterilmektedir. Çizelgede kullanılan kesinlik, duyarlılık, F1 skor, eğitim doğruluğu değerleri test verisine ait değerlerdir. Test doğruluğu, modellerin hiç görmediği test verileri üzerindeki doğruluğuna ait yüzdelik değerlerdir.

Çizelge 4.1. Belirteç sınıflandırma modellerinin metriklere göre değerlendirilmesi (%).

Model	Kesinlik	Duyarlılık	F1 Skor	Test Doğruluğu
BERT	89,47	94,44	91,89	92,11
ConvBERT	88,89	94,12	91,43	91,67
ELECTRA	88,24	96,77	92,31	90,20
DeBERTa	86,96	95,24	90,91	88,24
DistilBERT	86,96	95,24	90,91	86,67

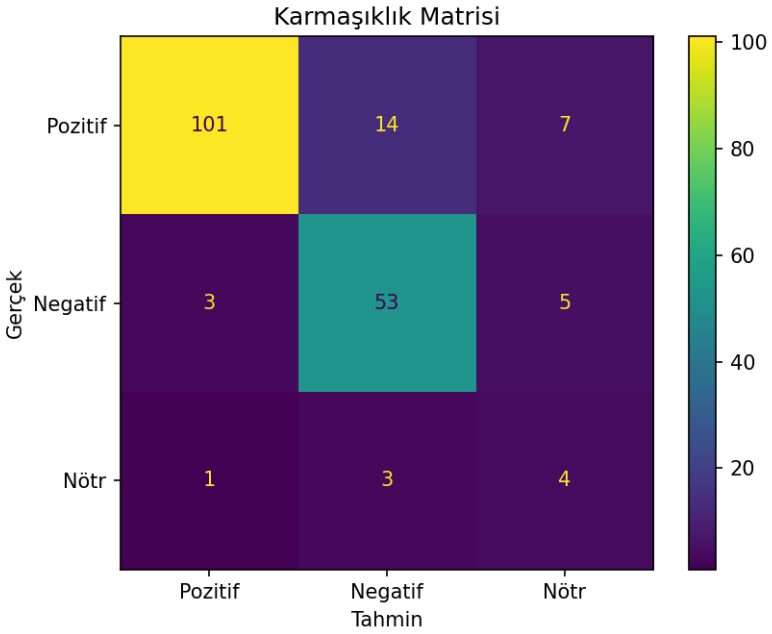
4.2. BERT ile Sıralı Sınıflandırma için Elde Edilen Bulgular

Sıralı sınıflandırma ile duygu analizinin, BERT modeli ile gerçekleştirilmesi sonucunda, BERT kayıp puanı - adım sayısı grafiği Şekil 4.1'de gösterilmektedir.



Şekil 4.1. BERT kayıp puanı- adım sayısı.

Sıralı sınıflandırma ile duygu analizinin, BERT modeli ile gerçekleştirilmesi sonucunda, karmaşıklık matrisi Şekil 4.2'de sunulmaktadır.



Şekil 4.2. BERT modeli için karmaşıklık matrisi.

Model performansını değerlendirmek için bazı metrikler kullanılmıştır. Eğitim sonucunda test verisi üzerinde, %67,74 F1-Skor, %73,22 duyarlılık, %65,63 kesinlik ve %82,72 doğruluk puanı elde edilmiştir.

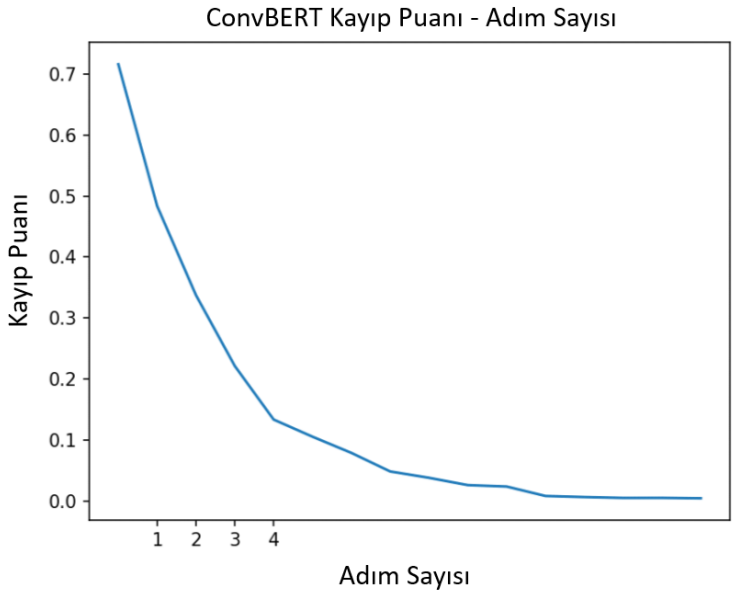
Pozitif, negatif ve nötr olmak üzere üç sınıfta gerçekleştirilen eğitimin, sınıflara göre başarısı, detaylı olarak metriklerle değerlendirilmiş ve Çizelge 4.2'de sunulmuştur.

Çizelge 4.2. BERT modelinin duygu sınıfları üzerinde değerlendirilmesi(%).

Metrikler	Pozitif	Negatif	Nötr	Makro Ort.	Ağırlıklı Ort.
Kesinlik	96,19	75,71	25,00	65,63	86,66
Duyarlılık	82,78	86,88	50,00	73,22	82,72
F1-Skor	88,98	80,91	33,33	67,74	84,07

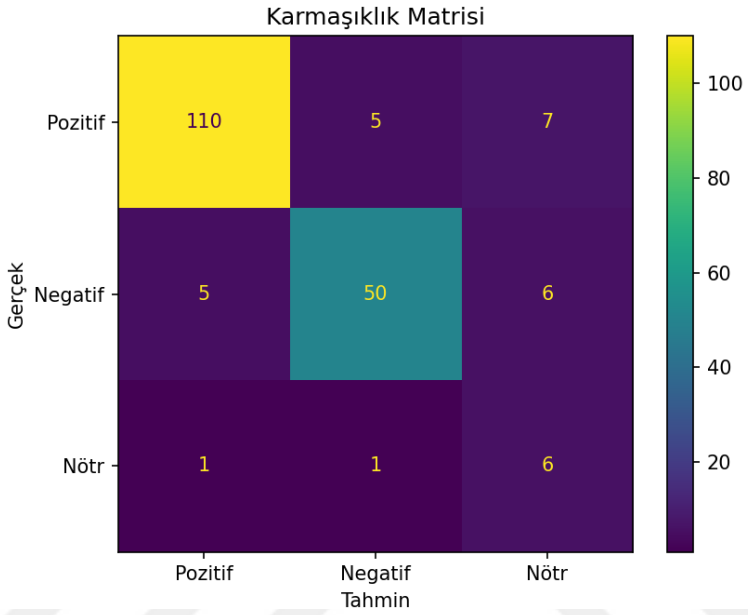
4.3. ConvBERT ile Sıralı Sınıflandırma için Elde Edilen Bulgular

Sıralı sınıflandırma ile duygu analizinin, ConvBERT modeli ile gerçekleştirilmesi sonucunda, ConvBERT kayıp puanı - adım sayısı grafiği Şekil 4.3'te gösterilmektedir.



Şekil 4.3. ConvBERT kayıp puanı- adım sayısı.

Sıralı sınıflandırma ile duygu analizinin, ConvBERT modeli ile gerçekleştirilmesi sonucunda, karmaşıklık matrisi Şekil 4.4'te sunulmaktadır. Eğitim sonucunda test verisi üzerinde, %74,11 F1-Skor, %82,37 duyarlılık, %71,89 kesinlik ve %86,91 doğruluk puanı elde edilmiştir.



Şekil 4.4. ConvBERT modeli için karmaşıklık matrisi.

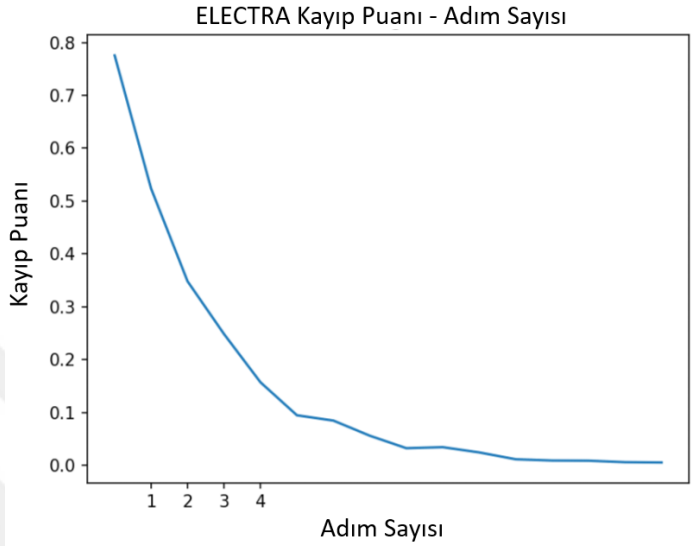
Pozitif, negatif ve nötr olmak üzere üç sınıfta gerçekleştirilen eğitimin, sınıflara göre başarısı, detaylı olarak metriklerle değerlendirilmiş ve Çizelge 4.3'te sunulmuştur.

Çizelge 4.3. ConvBERT modelinin duygu sınıfları üzerinde değerlendirilmesi (%)

Metrikler	Pozitif	Negatif	Nötr	Makro Ort.	Ağırlıklı Ort.
Kesinlik	94,82	89,28	31,57	71,89	90,40
Duyarlılık	90,16	81,96	75,00	82,37	86,91
F1-Skor	92,43	85,47	44,44	74,11	88,20

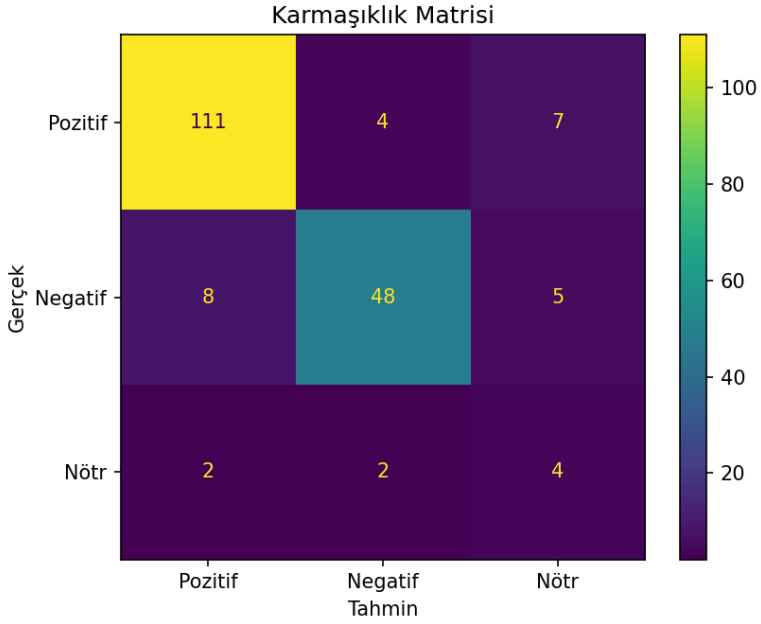
4.4. ELECTRA ile Sıralı Sınıflandırma için Elde Edilen Bulgular

Sıralı sınıflandırma ile duygu analizinin, ELECTRA modeli ile gerçekleştirilmesi sonucunda, ELECTRA kayıp puanı - adım sayısı grafiği Şekil 4.5'te gösterilmektedir.



Şekil 4.5. ELECTRA kayıp puanı- adım sayısı.

ELECTRA sıralı sınıflandırma modeli ile Eğitim sonucunda test verisi üzerinde, %69,38 F1-Skor, %73,22 duyarlılık, %68,54 kesinlik ve %85,34 doğruluk puanı elde edilmiştir.



Şekil 4.6. ELECTRA modeli için karmaşıklık matrisi.

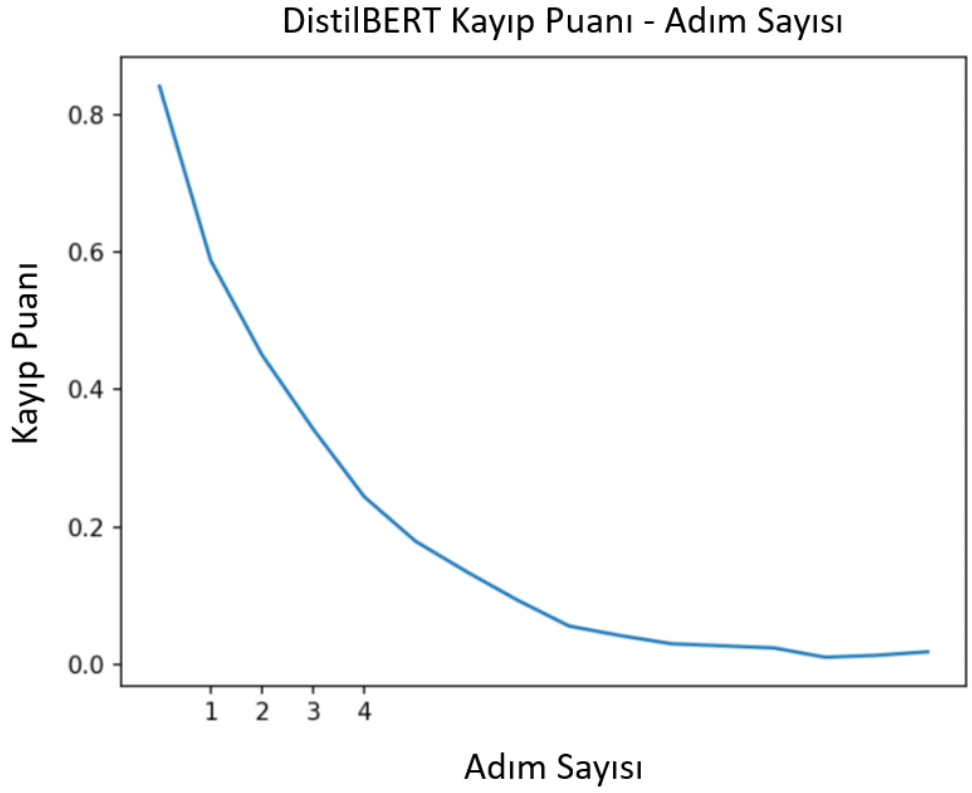
Sıralı sınıflandırma ile duygu analizinin, ELECTRA modeli ile gerçekleştirilmesi sonucunda, karmaşıklık matrisi Şekil 4.6'te sunulmaktadır. Pozitif, negatif ve nötr olmak üzere üç sınıfta gerçekleştirilen eğitimin, sınıflara göre başarısı, detaylı olarak metriklerle değerlendirilmiş ve Çizelge 4.4'te sunulmuştur.

Çizelge 4.4. ELECTRA modelinin duygu sınıfları üzerinde değerlendirilmesi(%)

Metrikler	Pozitif	Negatif	Nötr	Makro Ort.	Ağırlıklı Ort.
Kesinlik	91,73	88,88	25,00	68,54	88,03
Duyarlılık	90,98	78,68	50,00	73,22	85,34
F1-Skor	91,35	83,47	33,33	69,38	86,41

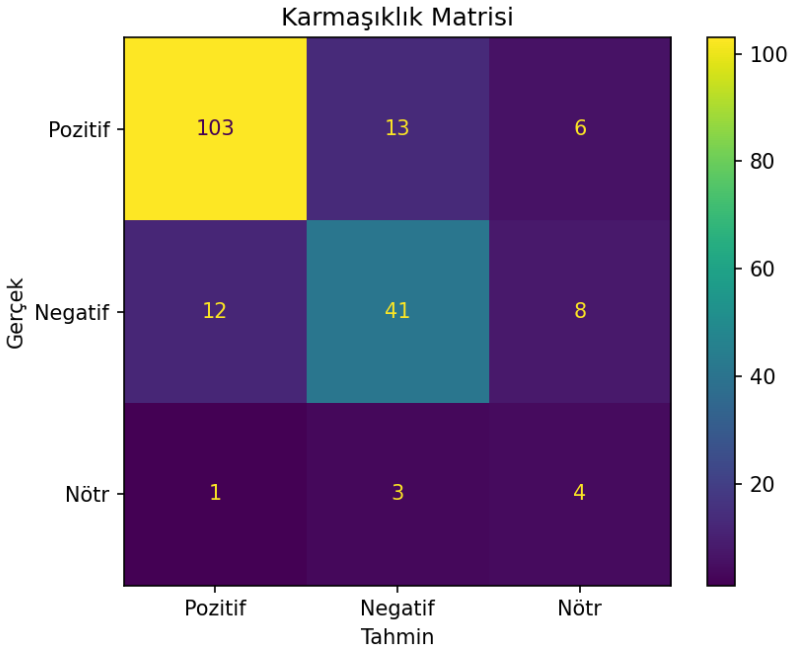
4.5. DistilBERT ile Sıralı Sınıflandırma için Elde Edilen Bulgular

Sıralı sınıflandırma ile duygu analizinin, DistilBERT modeli ile gerçekleştirilmesi sonucunda, DistilBERT kayıp puanı - adım sayısı grafiği Şekil 4.7'te gösterilmektedir.



Şekil 4.7. DistilBERT kayıp puanı- adım sayısı.

Sıralı sınıflandırma ile duygu analizinin, DistilBERT modeli ile gerçekleştirilmesi sonucunda, karmaşıklık matrisi Şekil 4.8'te sunulmaktadır. Eğitim sonucunda test verisi üzerinde, %62,27 F1-Skor, %67,21 duyarlılık, %60,98 kesinlik ve %77,48 doğruluk puanı elde edilmiştir.



Şekil 4.8. DistilBERT modeli için karmaşıklık matrisi.

Pozitif, negatif ve nötr olmak üzere üç sınıfta gerçekleştirilen eğitimin, sınıflara göre başarısı, detaylı olarak metriklerle değerlendirilmiş ve Çizelge 4.5'te sunulmuştur.

Çizelge 4.5. DistilBERT modelinin duygu sınıfları üzerinde değerlendirilmesi(%)

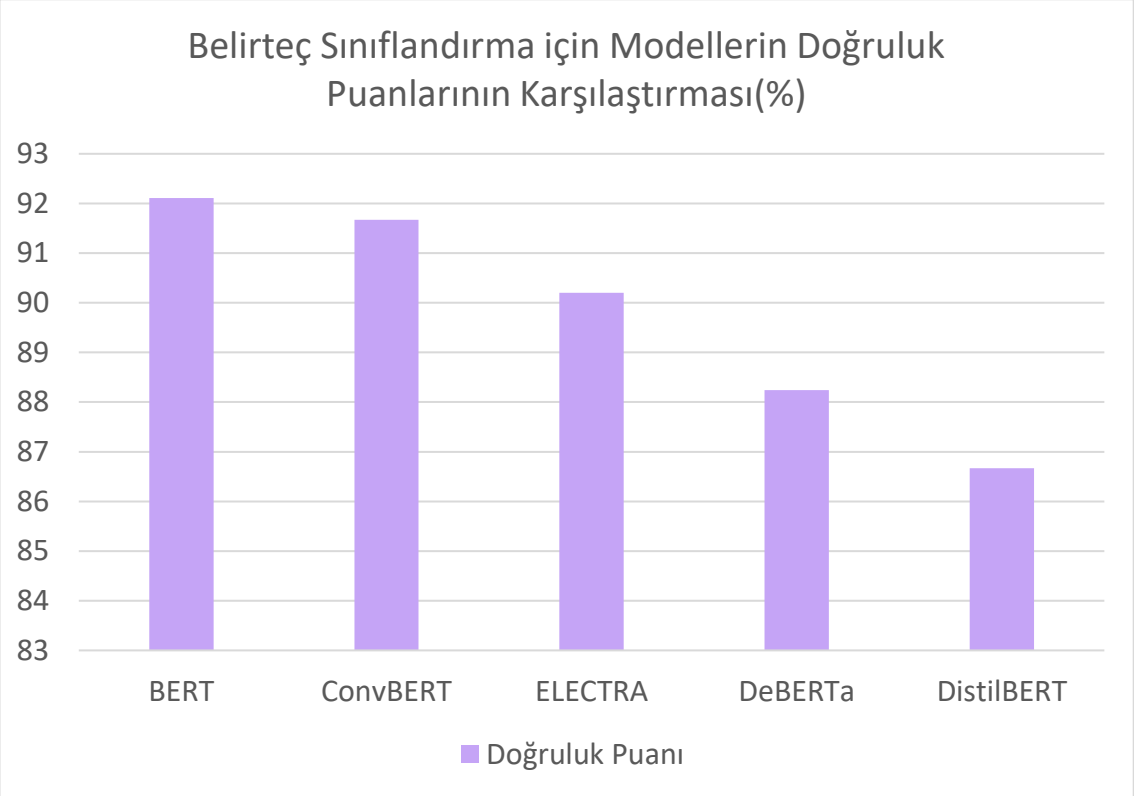
Metrikler	Pozitif	Negatif	Nötr	Makro Ort.	Ağırlıklı Ort.
Kesinlik	88,79	71,92	22,22	60,98	80,61
Duyarlılık	84,42	67,21	50,00	67,21	77,48
F1-Skor	86,55	69,49	30,76	62,27	78,76

4.6. Sonuçlar

Gerçekleştirilen çalışmada, belirteç sınıflandırma ve sıralı sınıflandırma modeli olarak iki yöntem kullanılmıştır. Belirteç sınıflandırma yönteminde BERT, ConvBERT,

ELECTRA, DeBERTa, DistilBERT olmak üzere 5 farklı yöntem kullanılmıştır. Bu modellerin test verisi üzerinde, sırasıyla yüzdelik doğruluk puanı, %92,11; %91,67; %90,20; %88,24; %86,67 olarak elde edilmiştir. En yüksek doğruluk puanına ulaşan BERT modeli çalışma için tercih edilmiştir. Bu sonuç, BERT modelinin bu çalışma için en iyi seçim olduğunu gösterir.

Doğruluk puanı, bir sınıflandırma modelinin doğru tahminlerinin toplam sayısının, tüm tahminlerin sayısına bölünmesiyle hesaplanır. Bu nedenle, doğruluk puanı bir modelin genel performansını gösterir. Modellere ait doğruluk puanının karşılaştırılmasına ait grafik Şekil 4.9' da sunulmuştur.



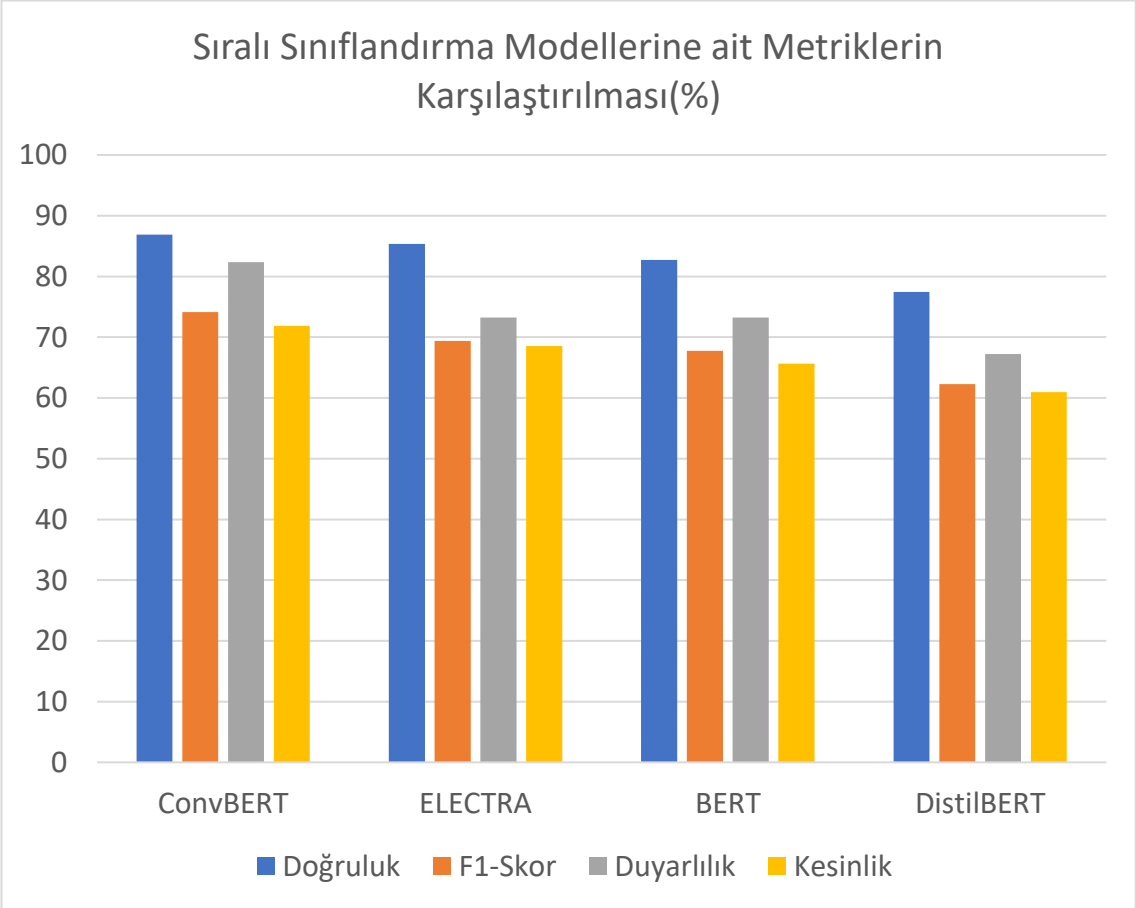
Şekil 4.9. Belirteç sınıflandırma modellerin doğruluk puanlarının karşılaştırılması.

Yapılan çalışmada, farklı sıralı sınıflandırma modelleri karşılaştırılmış ve bu modellerin başarısı ölçülmüştür. Çalışmada ConvBERT, ELECTRA, BERT ve DistilBERT modelleri kullanılmış ve bu modellerin doğruluk puanları karşılaştırılmıştır. Doğruluk puanları sırasıyla, %86,91; %85,34; %82,72; %77,48 olarak elde edilmiştir. Elde edilen

sonuçlara göre, sıralı sınıflandırma görevinde en başarılı model ConvBERT olarak belirlenmiştir ve bu model %86,91 doğruluk puanı elde etmiştir.

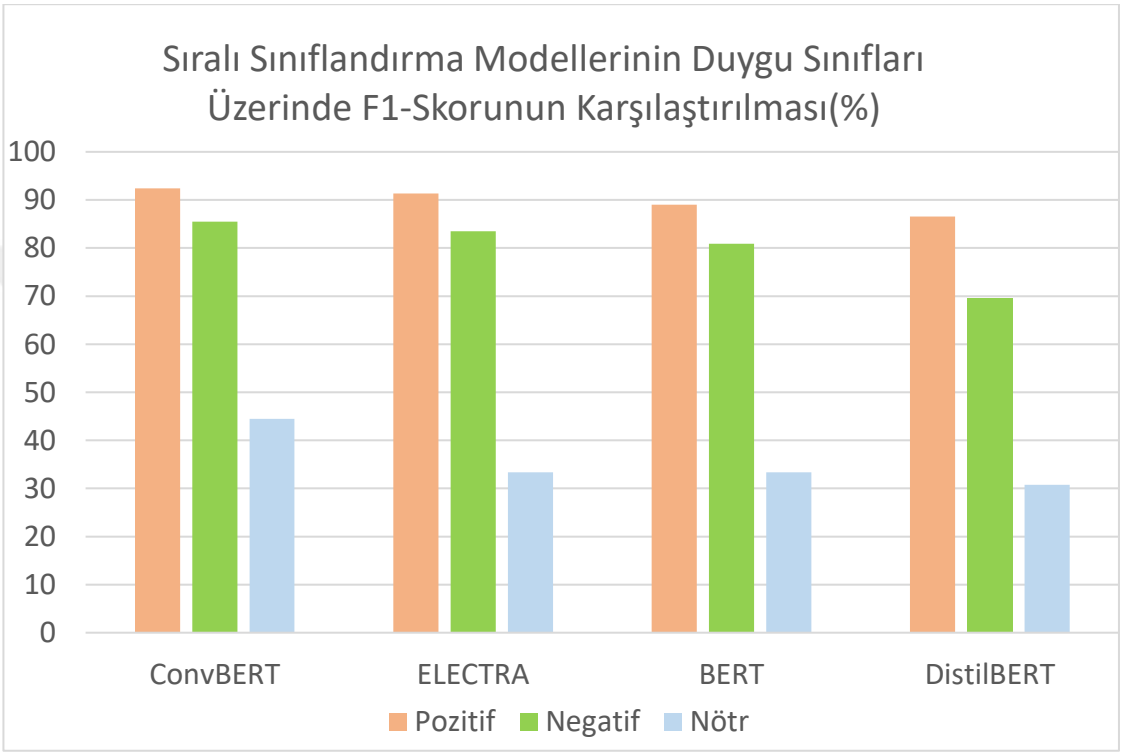
Ayrıca, modellerin performansını ölçmek için farklı metrikler kullanılmıştır. Bu metrikler arasında doğruluk, duyarlılık, kesinlik, F1 skoru gibi ölçümler yer almaktadır. Karşılaştırmalı analiz sonucunda, ConvBERT modelinin en yüksek doğruluk puanına sahip olduğu görülmüştür.

Modellerin performansına ait metriklerin karşılaştırılması Şekil 4.10'da gösterilmiştir. Bu grafikte, ConvBERT modelinin diğer modellere göre daha yüksek F1 skoru elde ettiği ve diğer modellere göre daha iyi bir performans sergilediği görülmektedir.



Şekil 4.10. Sıralı sınıflandırma modellerine ait metriklerin karşılaştırılması.

Sıralı sınıflandırma modellerinin duygu sınıfları için F1skor karşılaştırmasına ilişkin Şekil 4.11'de gösterilen analiz, sıralı sınıflandırma modellerinin duygu sınıfları üzerindeki başarısını göstermektedir. ConvBERT modeli, tüm duygular için en başarılı model olurken, DistilBERT modeli diğer modellere göre daha az başarılı olduğu görülmüştür.



Şekil 4.11. Sıralı sınıflandırma modellerinin duygu sınıfları için F1skor karşılaştırması.

- Modeller üzerinde ince ayar gerçekleştirilerek daha başarılı sonuçlar elde edilebilir.
- Önceden eğitilmiş modeller yerine probleme yönelik yeni modeller eğitilebilir.
- Mevcut modeller yeni verilerin etiketlemesinde kullanılarak çalışma kapsamı genişletilebilir.

5. TARTIŞMA ve SONUÇ

Bu çalışma, Türkçe metinlerde belirteç sınıflandırma ve sıralı sınıflandırma problemlerine odaklanan kapsamlı bir analiz sunmaktadır. Literatür incelendiğinde Türkçe metinlerde bu tür çalışmaların sınırlı sayıda olduğu görülmektedir. Bu kapsamda bu araştırma, Türkçe metin sınıflandırma alanında önemli bir katkı sağlaması beklenmektedir.

Belirteç sınıflandırma ile duygu tanımada BERT, ConvBERT, ELECTRA, DeBERTa, DistilBERT modelleri ile karşılaştırmalı analizde sırasıyla, %92,11; %91,67; %90,20; %88,24; %86,67 doğruluk puanları elde edilmiştir. DistilBERT modeller arasında en hafif versiyon olduğundan, modeli daha küçük boyuta sıkıştırmayı hedeflemektedir, bu yüzden doğruluk oranının en düşük olduğu görülmüştür.

Bu çalışmada, farklı sıralı sınıflandırma modelleri karşılaştırılmış ve bu modellerin başarısı ölçülmüştür. Bu kapsamdaConvBERT, ELECTRA, BERT ve DistilBERT modelleri kullanılmış ve bu modellerin doğruluk puanları karşılaştırılmıştır. Doğruluk puanları sırasıyla, %86,91; %85,34; %82,72; %77,48 olarak elde edilmiştir. Bu sonuçların, ConvBERT'in en yüksek doğruluk puanına sahip olduğunu ortaya koymuştur. ConvBERT, hem evrişim (convolutional) hem de dikkat mekanizması tabanlı özellikleri içeren bir model olduğundan sıralı sınıflandırma görevinde daha başarılı olduğu söylenebilir.

Duygu sınıfı tahminleme sonuçlarına göre, modeller pozitif duyguları tanıma konusunda daha başarılı olurken, nötr duygu sınıfında daha düşük bir başarı oranı görülmektedir. Bu durum, F1 skorunun ortalama değerinin düşük olmasına neden olmaktadır. Veri kümesinin detaylı analizi incelendiğinde, nötr duygu sınıfındaki verilerin sayısının az olmasının bu duruma sebep olduğu öngörülebilir. Bu durum, modellerin nötr duyguları doğru bir şekilde sınıflandırmakta zorlandıklarını ve başarı oranlarının düşük kaldığını göstermektedir.

5.1. Gelecek Çalışmalara Yönelik Öneriler

- Bu çalışmanın sonuçlarına dayanarak, Türkçe metinlerde belirteç sınıflandırma ve sıralı sınıflandırmaya odaklanan daha geniş çaplı çalışmalar yapılabilir.

- Farklı belirteç sınıflandırma modelleri ve sıralı sınıflandırma modelleri üzerinde daha detaylı analizler gerçekleştirilebilir.

- Diğer dil kaynaklarıyla karşılaştırmalı çalışmalar yaparak, Türkçe'nin diğer dillerle olan benzerliklerini ve farklılıklarını inceleyerek çoklu dil çalışmaları gerçekleştirilebilir.

- Duygu sınıfları eşit sayıda bulunacak şekilde veri sayısının artırılabilir.

- Mevcut çalışmalar genellikle genel amaçlı Türkçe metinleri üzerinde yapılmıştır. Gelecekte, belirli sektörlere veya özel alanlara odaklanan Türkçe ABSA çalışmaları gerçekleştirilebilir.

- Türkçe ABSA çalışmaları, anlamsal etiketleme ve öneri sistemleri gibi alanlarda kullanılabilir.

- Gelecekteki araştırmalar, duygu analizi algoritmalarını makine çevirisi sistemleriyle entegre etmeyi inceleyebilir. Bu, sadece kelimeleri çevirerek değil, aynı zamanda çeviri sürecinde genellikle yok olan aktarılmak istenen duygusal tonu da aktararak metinlerin çeviri yeterliklerini ya da kabul edilebilirlik düzeylerini geliştirebilir. Bunun müşteri hizmetleri, uluslararası iletişim ve edebiyat çevirisinde otomatik çevirilerin kalitesini artırmaya yönelik etkileri olabilir.

- Gelecekteki araştırmalarda gerçek zamanlı duygu analizi geri bildirimi sağlayarak toplum çevirmenlerine yardımcı olabilecek araçlar geliştirilebilir. Bu, özellikle konuşmacıların duygusal durumunu anlamanın çok önemli olduğu tıbbi, hukuki

veya sosyal hizmetler çevirisi gibi hassas bağlamlarda faydalı olabilir.

- Gelecekteki araştırmalar, duyguların kültürler arasında nasıl farklı şekilde ifade edildiğini ve algılandığını anlamaya odaklanabilir. Bu, kültürel açıdan daha duyarlı çeviri araçlarının ve hizmetlerinin geliştirilmesine yol açarak çok uluslu şirketlerde, küresel pazarlamada ve uluslararası ilişkilerde iletişimin artırabilir.

- Duygu analizini kullanarak farklı dillerdeki müşteri geri bildirimlerini analiz etmek için araştırma yapılabilir. Bu, çok uluslu şirketlere farklı bölgelerdeki müşteri duyguları hakkında daha derin bilgiler sağlayarak daha hedefli ve etkili marka yönetimi stratejilerine yardımcı olabilir.

- Çevirmenlerin metinlerdeki ve konuşmalardaki duygusal nüansları daha iyi anlamalarına ve aktarmalarına yardımcı olmak için duygu analizini kullanan eğitim programları veya eğitim yazılımları oluşturulabilir. Bu, özellikle edebi ve yaratıcı eserlerde daha incelikli ve etkili çevirilere yol açabilir.

- Daha kişiselleştirilmiş ve duygusal olarak yankı uyandıran reklam içeriği oluşturmada duygu analizinin kullanımı araştırılabilir. Şirketler, metinlerin duygusal bağlamını anlayarak pazarlama stratejilerini hedef kitlelerinin duygusal durumlarına ve tercihlerine daha iyi uyum sağlayacak şekilde uyarlayabilirler.

- Gelecekteki araştırmalarda farklı dillerdeki müşteri etkileşimlerinin gerçek zamanlı izlenmesinde duygu analizinin uygulanması incelenebilir. Bu, şirketlerin müşteri endişelerini anında tespit etmesine ve ele almasına olanak tanıyarak müşteri memnuniyetini ve hizmet kalitesini artırabilir.

- Çeviride duygu analizi kullanmanın etik hususlarını ve kültürel etkilerini araştırmak faydalı sonuçlar verebilir. Bu, bu teknolojilerin önyargılarını ve sınırlamalarını anlamayı ve daha eşitlikçi ve kültürel açıdan duyarlı çeviri araçları geliştirilmesine katkı sağlayabilir.

KAYNAKLAR

Adel, H., Dahou, A., Mabrouk, A., Abd Elaziz, M., Kayed, M., El-Henawy, I. M., ... & Amin Ali, A. (2022). Improving crisis events detection using distilbert with hunger games search algorithm. *Mathematics*, *10*(3), 447. https://doi.org/10.3390/math10030447

Alzubi, J., Nayyar, A., & Kumar, A. (2018). Machine learning from theory to algorithms: an overview. *Journal of Physics: Conference Series,* 1142(012012). https://doi.org/10.1088/1742-6596/1142/1/012012

Aylak, B. L., Okan, O., & Yazıcı, K. (2021). Yapay zeka ve makine öğrenmesi tekniklerinin lojistik sektöründe kullanımı. *El-Cezeri*, *8*(1), 74-93. https://doi.org/10.31202/ecjse.776314

Bai, X., Liu, P., & Zhang, Y. (2020). Investigating typed syntactic dependencies for targeted sentiment classification using graph attention neural network. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, *29*, 503-514.

Bird, S. (2006, July). NLTK: The natural language toolkit. In *Proceedings of the COLING/ACL 2006 Interactive Presentation Sessions*, 69-72.

Bisong, E. (2019). *Building machine learning and deep learning models on Google cloud platform* . Apress.

Broad, N. (2022). Token classification approach [fpe]. 27 Nisan 2023 tarihinde https://www.kaggle.com/code/nbroad/token-classification-approach-fpe adresinden erişildi.

Calude, A. S. (2023). *The linguistics of social media: An introduction*. Routledge.

Chi, Z., Huang, S., Dong, L., Ma, S., Zheng, B., Singhal, S., ... & Wei, F. (2021). Xlm-e: Cross-lingual language model pre-training via electra. *arXiv preprint arXiv:2106.16138*.

Cetin, F. S., & Eryiğit, G. (2018). Türkçe hedef tabanlı duygu analizi için alt görevlerin incelenmesi–hedef terim, hedef kategori ve duygu sınıfı belirleme. *Bilişim Teknolojileri Dergisi*, *11*(1), 43-56. https://doi.org/10.17671/gazibtd.325865

Clark, K., Luong, M. T., Le, Q. V., & Manning, C. D. (2020). Electra: Pre-training text encoders as discriminators rather than generators. *arXiv preprint arXiv:2003.10555*.

Damassino, N. (2020). The questioning Turing test. *Minds and Machines*, *30*(4), 563-587. https://doi.org/10.1007/s11023-020-09551-6

Du, Y., Pei, B., Zhao, X., & Ji, J. (2020). Deep scaled dot-product attention based domain adaptation model for biomedical question answering. *Methods*, *173*, 69-74. https://doi.org/10.1016/j.ymeth.2019.06.024

Dehkharghani, R., Yanikoglu, B., Saygin, Y., & Oflazer, K. (2017). Sentiment analysis in Turkish at different granularity levels. *Natural Language Engineering*, *23*(4), 535-559. https://doi.org/10.1017/S1351324916000309

Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Dogan, Ü., Edelbrunner, J., & Iossifidis, I. (2011, December). Autonomous driving: A comparison of machine learning techniques by means of the prediction of lane change behavior. In *2011 IEEE International Conference on Robotics and Biomimetics* (pp. 1837-1843). IEEE.

Ekinci, E., Türkmen, H., & Omurca, S. İ. (2017). Multi-word aspect term extraction using Turkish user reviews. *International Journal of Computer Engineering and Information Technology*, *9*(1), 15-23. https://ijceit.org/published/volume9/issue1/3Vol9No1.pdf

He, P., Liu, X., Gao, J., & Chen, W. (2020). Deberta: Decoding-enhanced bert with disentangled attention. *arXiv preprint arXiv:2006.03654*.

Hoş, S.,(2023, Mart). BERT algoritması nedir? Yapay zekanın search ekosistemindeki hakimiyeti, örneklerle bert algoritması. 28 Nisan 2023 tarihinde https://www.hosting.com.tr/blog/bert/ adresinden erişildi.

Husain, A. (2019). *Duyarlı makine: Yapay zekânın olgunluk çağı* (Çev. D. Dalgakıran). Siyah Kitap

Jiang, Z. H., Yu, W., Zhou, D., Chen, Y., Feng, J., & Yan, S. (2020). Convbert: Improving bert with span-based dynamic convolution. *Advances in Neural Information Processing Systems*, *33*, 12837-12848.

Joshi, M., Levy, O., Weld, D. S., & Zettlemoyer, L. (2019). BERT for coreference resolution: Baselines and analysis. *arXiv preprint arXiv:1908.09091*.

Kabakus, A. T. (2022). A novel COVID-19 sentiment analysis in Turkish based on the combination of convolutional neural network and bidirectional long-short term memory on Twitter. *Concurrency and Computation: Practice and Experience*, *34*(22), e6883.

Kama, B., Ozturk, M., Karagoz, P., Toroslu, I. H., & Kalender, M. (2017, November). Analyzing implicit aspects and aspect dependent sentiment polarity for aspect-based sentiment analysis on informal Turkish texts. In *Proceedings of the 9th International Conference on Management of Digital EcoSystems (pp.134-141). New York, NY, USA: ACM* https://doi.org/10.1145/3167020.3167041

Kama, B., Ozturk, M., Karagoz, P., Toroslu, I. H., & Ozay, O. (2016). A web search enhanced feature extraction method for aspect-based sentiment analysis for Turkish informal texts. In *Big Data Analytics and Knowledge Discovery: 18th International Conference, DaWaK 2016, Porto, Portugal, September 6-8, 2016, Proceedings 18* (pp. 225-238). Springer International Publishing.

Karagoz, P., Kama, B., Ozturk, M., Toroslu, I. H., & Canturk, D. (2019). A framework for aspect based sentiment analysis on turkish informal texts. *Journal of Intelligent Information Systems*, *53*, 431-451. https://doi.org/10.1007/s10844-019-00565-w

Levesque, H. J. (2017). *Common sense, the Turing test, and the quest for real AI*. Mit Press.

Li, C., Peng, X., Peng, H., Li, J., & Wang, L. (2021, August). TextGTL: Graph-based transductive learning for semi-supervised text classification via structure-sensitive interpolation. *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence*, IJCAI-21, edited by Zhi-Hua Zhou, 2680–86. International Joint Conferences on Artificial Intelligence Organization. https://doi.org/10.24963/ijcai.2021/369.

Madhoushi, Z., Hamdan, A. R., & Zainudin, S. (2019). Aspect-based sentiment analysis methods in recent years. *Asia-Pacific Journal Of Information Technology And Multimedia*, *7*(2), 79-96.

Martin, C., Yang, H., & Hsu, W. (2022, July). KDDIE at SemEval-2022 Task 11: Using DeBERTa for Named Entity Recognition. *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)* (pp. 1531-1535).

Monil, P., Darshan, P., Jecky, R., Vimarsh, C., & Bhatt, B. R. (2020). Customer segmentation using machine learning. *International Journal for Research in Applied Science and Engineering Technology (IJRASET)*, *8*(6), 2104-2108.

Munkhdalai, T. & Yu, H. (2017). Neural Tree Indexers for Text Understanding. ArXiv (Cornell University). https://doi.org/10.18653/v1/e17-1002

Neubig, G., Goldberg, Y., & Dyer, C. (2017). On-the-fly Operation Batching in Dynamic Computation Graphs. Neural Information Processing Systems, 30, 3971–3981.

O'Shea, J., Bandar, Z., & Crockett, K. (2010). A machine learning approach to speech act classification using function words. In *Agent and Multi-Agent Systems: Technologies and Applications: 4th KES International Symposium, KES-AMSTA 2010, Gdynia, Poland, June 23-25, 2010, Proceedings. Part II 4* (pp. 82-91). Springer Berlin Heidelberg.

Özkan, D. (2020). *Aspect-based sentiment analysis in Turkish.*[Yüksek Lisans Tezi, Atılım University]. https://tez.yok.gov.tr/UlusalTezMerkezi/tezSorguSonucYeni.jsp

Parçali, B., & Mutlu, F. (2022). Comparison of the machine learning algorithms in breast cancer risk assessment with the Gail Model: Methodological Study. *Turkiye Klinikleri Journal of Biostatistics*, *14(*1), 32–44. https://doi.org/10.5336/biostatic.2021-85755

Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., ... & Lerer, A. (2017). *Automatic differentiation in pytorch.* In NIPS Workshop, 2017. URL https://openreview.net/pdf?id=BJJsrmfCZ.

Petrosyan, A. (2023, Ekim). Internet users in the world 2023. 8 Kasım 2023 tarihinde https://www.statista.com/statistics/617136/digital-population-worldwide/ adresinden erişildi.

Pirim, A. G. H. (2006). Yapay zeka. *Yaşar Üniversitesi E-Dergisi*, *1*(1), 81-93.

https://dergipark.org.tr/en/download/article-file/179113

Pontiki, M., Galanis D., Pavlopoulos J., Papageorgiou H. , Androutsopoulos I, and Manandhar S. (2014). SemEval-2014 Task 4: Aspect Based Sentiment Analysis. In *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, (pp. 27–35). Association for Computational Linguistics.

Pontiki, M., Galanis, D., Papageorgiou, H., Androutsopoulos, I., Manandhar, S., AL-Smadi, M., ... & Eryiğit, G. (2016). Semeval-2016 task 5: Aspect based sentiment analysis. In *ProWorkshop on Semantic Evaluation (SemEval-2016)* (pp. 19-30). Association for Computational Linguistics.

Poole, D. L., & Mackworth, A. K. (2010). *Artificial Intelligence: foundations of computational agents*. Cambridge University Press.

Prasanna, P. L., & Rao, D. R. (2018). Text classification using artificial neural networks. *International Journal of Engineering & Technology, 7*(1.1), 603-606.

Reddy, N., Singh, P., & Srivastava, M. M. (2020). BERT Understand Sentiment? Leveraging Comparisons Between Contextual and Non-Contextual Embeddings to Improve Aspect-Based Sentiment Models. *arXiv preprint arXiv:2011.11673*.

Rietzler, A., Stabinger, S., Opitz, P., & Engl, S. (2019). Adapt or get left behind: Domain adaptation through bert language model finetuning for aspect-target sentiment classification. *arXiv preprint arXiv:1908.11860*.

Rothman, D. (2021). *Transformers for Natural Language Processing: Build innovative deep neural network architectures for NLP with Python, PyTorch, TensorFlow, BERT, RoBERTa, and more*. Packt Publishing Ltd.

Rzeszótko, J., & Nguyen, S. H. (2012). Machine learning for traffic prediction. *Fundamenta Informaticae*, *119*(3-4), 407-420. https://doi.org/10.3233/FI-2012-745

Saeidi, M., Bouchard, G., Liakata, M., & Riedel, S. (2016). Sentihood: Targeted aspect based sentiment analysis dataset for urban neighbourhoods. *arXiv preprint arXiv:1610.03771*.

Sağbaş, E. A., & Ballı, S. (2016). Akıllı telefon algılayıcıları ve makine öğrenmesi kullanılarak ulaşım türü tespiti. *Pamukkale Üniversitesi Mühendislik Bilimleri Dergisi, 22*(5), 376-383. https://doi.org/10.5505/pajes.2015.63308

Sağıroğlu, M. (2020). Yapay zekâ ve ötesi. İçinde Ş. Sağırloğlu & M. U. Demirezen (Eds.),. *Yapay zekâ ve büyük veri: Teknolojiler, yaklaşımlar ve uygulamalar* (s. 1-60). Nobel Akademik Yayıncılık.

Salton, G. D., Ross, R., & Kelleher, J. D. (2016). Idiom Token Classification using Sentential Distributed Semantics. Arrow - TU Dublin (Technological University Dublin). https://doi.org/10.18653/v1/p16-1019

Salur, M. U., AYDIN, İ., & Jamous, M. (2021). An ensemble approach for aspect term extraction in Turkish texts. *Pamukkale Üniversitesi Mühendislik Bilimleri Dergisi*, *28*(5), 769-776.

Sánchez-Fernández, I., & Peters, J. M. (2023). Machine learning and deep learning in medicine and neuroimaging. *Annals of the Child Neurology Society*, *1*(2), 102–122. https://doi.org/10.1002/cns3.5

Sanh, V., Lysandre Debut, Julien Chaumond, & Wolf, T. (2019). DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. ArXiv (Cornell University).

Schweter, S. (2020). BERTurk - BERT models for Turkish. https://doi.org/10.5281/zenodo.3770924

Selimoglu, M., & Yilmaz, A. (2021). Kredi kartı dolandırıcılık tespitinin makine öğrenmesi yöntemleri ile tahmin edilmesi. *Beykent Üniversitesi Fen Ve Mühendislik Bilimleri Dergisi*, *13*(2), 28-33.

Shatnawi, M., & Mohamed, N. (2012, March). Statistical techniques for online personalized advertising: A survey. In *Proceedings of the 27th Annual ACM Symposium on Applied Computing* (pp. 680-687).

Shen, Z., Zhang, M., Zhao, H., Yi, S., & Li, H. (2021). Efficient attention: Attention with linear complexities. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision* (pp. 3531-3539).

Shieber, S. M. (Ed.). (2004). *The Turing test: Verbal behavior as the hallmark of intelligence*. Mit Press.

Silva, E. H., & Marcacini, R. M. (2021, July). Aspect-based sentiment analysis using BERT with Disentangled Attention. In *Proceedings of the 8th ICML Workshop on Automated Machine Learning (AutoML 2021), Virual Event* (pp. 23-24).

Singh, S. S., Srivastva, D., Kumar, A., & Srivastava, V. (2022). NN-LP-CF: neural network based link prediction on social networks using centrality-based features. In *Deep Learning for Social Media Data Analytics* (pp. 27-42). Springer International Publishing.

Sinha, K., Jia, R., Hupkes, D., Pineau, J., Williams, A., & Kiela, D. (2021). Masked language modeling and the distributional hypothesis: Order word matters pre-training for little. *arXiv preprint arXiv:2104.06644*.

Siriborvornratanakul, T. (2022). Human behavior in image-based Road Health Inspection Systems despite the emerging AutoML. *Journal of Big Data*, *9*(1), 96. https://doi.org/10.1186/s40537-022-00646-8

Song, Y., Wang, J., Jiang, T., Liu, Z., & Rao, Y. (2019). Attentional encoder network for targeted sentiment classification. *arXiv preprint arXiv:1902.09314*.

Sun, Y., Zheng, Y., Hao, C., & Qiu, H. (2021). NSP-BERT: A Prompt-based Zero-Shot Learner Through an Original Pre-training Task--Next Sentence Prediction. *arXiv preprint*

arXiv:2109.03564.

Tariq, M. U., Poulin, M., & Abonamah, A. A. (2021). Achieving operational excellence through artificial intelligence: Driving forces and barriers. *Frontiers in Psychology*, *12*, 686624. https://doi.org/10.3389/fpsyg.2021.686624

Truşcă, M. M., Wassenberg, D., Frasincar, F., & Dekker, R. (2020). A hybrid approach for aspect-based sentiment analysis using deep contextual word embeddings and hierarchical attention. In *Web Engineering: 20th International Conference, ICWE 2020, Helsinki, Finland, June 9–12, 2020, Proceedings 20* (pp. 365-380). Springer International Publishing.

Türkmen, H., Omurca, S. I., & Ekinci, E. (2016). An aspect based sentiment analysis on Turkish hotel reviews. *Girne American University Journal of Social and Applied Sciences*, *6*(2), 12-15.

Urhan, M., (2020, Haziran). BERT algoritması nedir? Örneklerle BERT algoritması. 3 Kasım 2023 tarihinde https://zeo.org/tr/kaynaklar/blog/bert-algoritmasi-nedir adresinden erişildi.

Van Der Walt, S., Colbert, S. C., & Varoquaux, G. (2011). The NumPy array: A structure for efficient numerical computation. *Computing in Science & Engineering*, *13*(2), 22-30. https://doi.org/10.1109/MCSE.2011.37.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is all you need. *Advances in neural information processing systems*, *30*.

Wallaart, O., & Frasincar, F. (2019). A hybrid approach for aspect-based sentiment analysis using a lexicalized domain ontology and attentional neural models. In *The Semantic Web: 16th International Conference, ESWC 2019, Portorož, Slovenia, June 2–6, 2019, Proceedings 16* (pp. 363-378). Springer International Publishing.

Whitby, B. (2005). *Yapay zekâ: Yeni başlayanlar için kılavuz* (Çev. Ç. Karabağlı). İletişim.

Willman, J (2021). *Modern PyQt: create GUI applications for project management, computer vision, and data analysis*. Apress.Xie, H., Ge, Y., & Yi, J. (2022). Cost Control Analysis of construction projects based on wireless communication and artificial intelligence decisions. *Wireless Communications and Mobile Computing*, *2022*.

Xu, H., & Ashley, K. (2022). Multi-granularity Argument Mining in Legal Texts. *arXiv preprint arXiv:2210.09472*.

Yang, C., Zhang, H., Jiang, B., & Li, K. (2019). Aspect-based sentiment analysis with alternating coattention networks. *Information Processing & Management*, 56(3), 463-478.

Yuan, L., Chen, Y., Wang, T., Yu, W., Shi, Y., Jiang, Z. H., ... & Yan, S. (2021). Tokens-to-token vit: Training vision transformers from scratch on imagenet. In *Proceedings of*

the IEEE/CVF international conference on computer vision (pp. 558-567).

Zhang, Q., Lu, H., Sak, H., Tripathi, A., McDermott, E., Koo, S., & Kumar, S. (2020, May). Transformer transducer: A streamable speech recognition model with transformer encoders and rnn-t loss. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (pp. 7829-7833). IEEE.

Zhang, Y., Zhang, M., Wu, S., & Zhao, J. (2022). Towards Unifying the Label Space for Aspect-and Sentence-based Sentiment Analysis. *arXiv preprint arXiv:2203.07090*.

Zhao, K., Huang, L., Song, R., Shen, Q., & Xu, H. (2021). A sequential graph neural network for short text classification. *Algorithms, 14*(12), 352. https://doi.org/10.3390/a14120352