DÜZCE ÜNİVERSİTESİ | LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ

METİN MADENCİLİĞİ VE DERİN AĞLAR İLE SORU CEVAP SİSTEMİ

HÜSEYİN AVNİ ARDAÇ

DOKTORA TEZİ
BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI

DANIŞMAN
PROF. DR. PAKİZE ERDOĞMUŞ

DÜZCE, 2024

T.C.
DÜZCE ÜNİVERSİTESİ
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ

METİN MADENCİLİĞİ VE DERİN AĞLAR İLE SORU CEVAP SİSTEMİ

Hüseyin Avni ARDAÇ tarafından hazırlanan tez çalışması aşağıdaki jüri tarafından Düzce Üniversitesi Lisansüstü Eğitim Enstitüsü Bilgisayar Mühendisliği Anabilim Dalı'nda **DOKTORA TEZİ** olarak kabul edilmiştir.

Tez Danışmanı

Prof. Dr. Pakize ERDOĞMUŞ
Düzce Üniversitesi

Jüri Üyeleri

Prof. Dr. Pakize ERDOĞMUŞ
Düzce Üniversitesi

Prof. Dr. Devrim AKGÜN
Sakarya Üniversitesi

Doç. Dr. Zehra KARAPINAR ŞENTÜRK
Düzce Üniversitesi

Doç. Dr. Abdullah Talha KABAKUŞ
Düzce Üniversitesi

Dr. Öğr. Üyesi Muhammet Sinan BAŞARSLAN
İstanbul Medeniyet Üniversitesi

Tez Savunma Tarihi: 24/06/2024

BEYAN

Bu tez çalışmasının kendi çalışmam olduğunu, tezin planlanmasından yazımına kadar bütün aşamalarda etik dışı davranışımın olmadığını, bu tezdeki bütün bilgileri akademik ve etik kurallar içinde elde ettiğimi, bu tez çalışmasıyla elde edilmeyen bütün bilgi ve yorumlara kaynak gösterdiğimi ve bu kaynakları da kaynaklar listesine aldığımı, yine bu tezin çalışılması ve yazımı sırasında patent ve telif haklarını ihlal edici bir davranışımın olmadığını beyan ederim.

24 Haziran 2024

Hüseyin Avni ARDAÇ

TEŞEKKÜR

Doktora öğrenimimde ve bu tezin hazırlanmasında gösterdiği her türlü destek ve yardımdan dolayı çok değerli hocam Prof. Dr. Pakize ERDOĞMUŞ'a en içten dileklerimle teşekkür ederim. Tez izleme Kurul üyelerim Prof. Dr. Devrim AKGÜN ve Doç. Dr. Zehra KARAPINAR ŞENTÜRK hocalarıma da emekleri için teşekkür ederim.

Her zaman yanımda olan ve benden desteğini esirgemeyen, başarılarımda moral ve motivasyon katkısı olan annem Fethiye ARDAÇ'a ve babam Mehmet Avni ARDAÇ'a çok teşekkür ederim.

Doktora eğitimim boyunca tüm desteğini veren ve geceli gündüzlü yanımda olan ve bana her zaman güvenen çok değerli eşim Fatma Betül KARA ARDAÇ'a sonsuz teşekkür ederim.

Desteklerini benden esirgemeyen çalışma arkadaşlarıma şükranlarımı sunarım.

24 Haziran 2024 **Hüseyin Avni ARDAÇ**

İÇİNDEKİLER

ŞEKİL LİSTESİ

Sayfa No

ÇİZELGE LİSTESİ

Sayfa No

KISALTMALAR

AI	Artificial Intelligence (Yapay Zekâ)
Adagrad	Adaptive Gradient (Adaptif Gradyan)
Adadelta	Adaptive Delta (Adaptif Delta)
Adam	Adaptive Moment Estimation (Adaptif Moment Tahmini)
AdamW	Adaptive Moment Estimation and Weight Decay (Adaptif Moment Tahmini ve Ağırlık Azalması)
Adamax	Adaptive Maximum (Uyarlanabilir Maksimum)
Adafactor	Adaptive Factor (Uyarlanabilir Faktör)
ANN	Artificial Neural Networks (Yapay Sinir Ağları)
API	Application Programming Interface (Uygulama Programlama Arayüzü)
BERT	Bidirectional Encoder Representations from Transformers (Transformatörlerden Çift Yönlü Kodlayıcı Temsilleri)
BERTBiDuQuA	BERT&BiLSTM Düzce University Question Answering (BiLSTM&BERT Düzce Üniversitesi Soru Cevaplama)
BERTDuQuA	BERT Düzce University Question Answering (BERT Düzce Üniversitesi Soru Cevaplama)
BiLSTM	Bidirectional Long Short-Term Memory (Çift Yönlü Uzun Kısa Süreli Bellek)
BoW	Bag of Words (Kelime Torbası)
CBOW	Continuous Bag of Words (Sürekli Kelime Torbası)
CNN	Convolutional Neural Networks (Konvolüsyonel Sinir Ağları)
CUDA	Compute Unified Device Architecture (Birleşik İşlem Cihaz Mimarisi)
DistilBERT	A Distilled Version of BERT (BERT'in Özütleştirilmiş Bir Versiyonu)
ELMo	Embeddings from Language Model (Dil Modellerinden Gömme Vektörler)
ELU	Exponential Linear Unit (Üstel Doğrusal Birim)
EM	Exact Match (Tam Eşleşme)
GB	Gigabyte (Gigabayt)
GELU	Gaussian Error Linear Unit (Gauss Hatası Doğrusal Birim)

GloVe	Global Vectors (Global Vektörler)
GPU	Graphics Processing Unit (Grafik İşleme Ünitesi)
GPT	Generative Pre-trained Transformer (Üretici Önceden Eğitilmiş Dönüştürücü)
IDF	Inverse Document Frequency (Ters Belge Frekansı)
IR	Information Retrieval (Bilgi Erişim)
JSON	JavaScript Object Notation (JavaScript Nesne Gösterimi)
KB	Knowledge Base (Bilgi Tabanlı)
Lion	Layer-wise Optimizer (Katman bazlı Optimizatör)
LSTM	Long Short-Term Memory (Uzun Kısa Süreli Bellek)
MLM	Masked Language Model (Maskeli Dil Modeli)
MMR	Maximal Marginal Relevance (Maksimal Marjinal Uygunluk)
MRR	Mean Reciprocal Rank (Ortalama Karşılıklı Sıralama)
Nadam	Nesterov-accelerated Adaptive Moment Estimation (Nesterov hızlandırmalı Uyarlanabilir Moment Tahmini)
NLU	Natural Language Understanding (Doğal Dil Anlama)
NLP	Natural Language Processing (Doğal Dil İşleme)
NSP	Next Sentence Prediction (Sonraki Cümle Tahmini)
PReLU	Parametric ReLU (Parametrik ReLU)
ReLU	Rectified Linear Unit (Doğrultulmuş Doğrusal Birim)
RDF	Resource Description Framework (Kaynak Tanımlama Çerçevesi)
Rprop	Resilient Propagation (Dirençli Yayılımı)
RMSprop	Root Mean Square Propagation (Kök Ortalama Kare Yayılımı)
RNN	Recurrent Neural Network (Tekrarlayan Sinir Ağı)
RoBERTa	Robustly Optimized BERT Approach (Dayanıklı Bir Şekilde Optimize Edilmiş BERT Yaklaşımı)
SELU	Scaled Exponential Linear Unit (Ölçeklendirilmiş Üstel Doğrusal Birim)
SGD	Stochastic Gradient Descent (Stokastik Gradyan İnişi)
SP	Semantic Parsing (Anlamsal Ayrıştırma)
SQuAD	Stanford Question Answering Dataset (Stanford Soru Cevaplama Veri Kümesi)
Tanh	Hyperbolic Tangent (Hiperbolik Tanjant)
TF	Term Frequency (Terim Frekansı)

TF-IDF	Term Frequency-Inverse Document Frequency (Terim Frekansı-Ters Belge Frekansı)
THQuAD	Turkish History Question Answering Dataset (Türkçe Tarih Soru Cevaplama Veri Kümesi)
TPU	Tensor Processing Unit (Tensör İşleme Birimi)
Word2Vec	Word Vector (Kelime Vektörü)
QAS	Question Answering System (Soru Cevaplama Sistemleri)

SİMGELER

Σ	Toplam Sembolü
σ	Sigma
α	Adım boyutu veya öğrenme hızı
β_1	Birinci moment kestirimi için üstel azalma oranı
β_2	İkinci moment kestirimi için üstel azalma oranı
θ	Stokastik hedef fonksiyonu
θ_0	İlk parametre vektörü
m_0	Birinci moment vektörü
v_0	İkinci moment vektörü
ε	Sıfıra bölünmeyi önlemek için kullanılan değer
Q	Query Vector (Sorgu Vektörü)
K	Key Vector (Anahtar Vektörü)
V	Value Vector (Değer Vektörü)

ÖZET

METİN MADENCİLİĞİ VE DERİN AĞLAR İLE SORU CEVAP SİSTEMİ

Hüseyin Avni ARDAÇ

Düzce Üniversitesi
Lisansüstü Eğitim Enstitüsü, Bilgisayar Mühendisliği Anabilim Dalı
Doktora Tezi
Danışman: Prof. Dr. Pakize ERDOĞMUŞ

Haziran 2024, 129 sayfa

Günümüzde teknolojinin hızlı gelişimi, insanların yaşam şekillerinde birçok alışkanlığında değişimine sebep olmuştur. Pandeminin de etkisiyle eğitim başta olmak üzere birçok alanda yüz yüze iletişim oldukça azalmıştır. Birçok kurum sektörlerine yönelik verileri kullanmak amacıyla çalışmalar yapmaktadırlar. Gelişen yapay zekâ teknolojileriyle, bu çalışmalar firmalara katma değer kazandırmaktadır. Bu değeri kazandıran çalışma alanlarından biri de yapay zekânın alt dalı olan doğal dil işlemede soru cevaplama sistemleridir. Soru cevaplama sistemleri, kullanıcıların doğal dildeki sorularına cevap verebilme yeteneğine sahip sistemlerdir. Bu sistemler soruları anlama ve doğru cevapları bulma aşamalarından oluşmaktadır. Sorular doğal dil işleme teknikleriyle analiz edilir ve cevaplar bilgi çekme yöntemleriyle ilgili veri kaynaklarından elde edilir. Bu çalışmada, metin madenciliği ve derin ağlar kullanılarak soru cevaplama modelleri tasarlanmıştır. Önceden eğitilmiş İngilizce BERT-base modeli farklı hiperparametrelerle ince ayar tekniği kullanılarak Stanford Soru Cevaplama Veri Seti (SQuADv1.1) ile eğitimi yapılmıştır. Eğitim sonucu literatürde yapılan çalışmalara kıyasla %88,13 F1 skoru ve %80,74 Tam Eşleşme (Exact Match - EM) oranıyla yüksek başarı elde edilmiştir. Ardından Türkçe Tarih Soru Cevaplama Veri Seti (THQuADv1.0) üzerinde iyileştirme çalışması yapılmıştır. Veri setine Düzce Üniversitenin birimleriyle ilgili sorularda eklenerek THQuADv2.0 olarak güncellenmiştir. Önceden eğitilmiş Türkçe BERTurk-base modeli İngilizce modelde elde edilen başarılı hiperparametrelerle ince ayar tekniği kullanılarak THQuADv2.0 veri seti ile eğitimi yapılmıştır. Yapılan eğitim sonucunda Türkçe soru cevaplama için BERTDuQuA (BERT Düzce University Question Answering) modeli oluşturulmuştur. BERTDuQuA modeli %87,10 F1 skoru ve %76,90 EM ile yüksek başarı elde edilmiştir. BERT modeline BiLSTM katmanıda eklenerek yeni bir model oluşturulmuştur. Türkçe modelde elde edilen başarılı hiperparametrelerle ince ayar tekniği kullanılarak THQuADv2.0 veri seti ile eğitimi yapılmıştır. Yapılan eğitim sonucunda Türkçe soru cevaplama için BERTBiDuQuA (BERT&BiLSTM Duzce University Question Answering) modeli oluşturulmuştur. BERTBiDuQuA modeli %88,84 F1 skoru ve %78,43 EM ile yüksek başarı elde edilmiştir.

Anahtar sözcükler: Derin Öğrenme, Doğal Dil İşleme, Metin Madenciliği, Soru Cevaplama Sistemi

ABSTRACT

QUESTION ANSWERING SYSTEM WITH TEXT MINING AND DEEP NETWORKS

Hüseyin Avni ARDAÇ

Düzce University
Graduate School, Department of Computer Engineering
Doctoral Thesis
Supervisor: Prof. Dr. Pakize ERDOĞMUŞ

June 2024, 129 pages

Nowadays, the rapid development of technology has led to changes in many aspects of people's lifestyles. Due to the pandemic, face-to-face communication has decreased significantly, particularly in education. Institutions are now utilizing their data to improve their sectors. The use of artificial intelligence technologies adds value to companies. Question answering systems are a sub-branch of artificial intelligence that add value by answering users' questions in natural language. These systems consist of two stages: understanding the questions and finding the correct answers. The analysis of questions utilises natural language processing techniques, while answers are obtained through information extraction methods from relevant data sources. In this study, the design of question answering models using text mining and deep networks is presented. The pre-trained English BERT-base model was fine-tuned with the Stanford Question Answering Dataset (SQuADv1.1) using various hyperparameters and fine-tuning values. The results of the training show high success rates, with an F1 score of 88.13% and an Exact Match (EM) rate of 80.74%, compared to previous studies in the literature. An improvement study was conducted on the Turkish History Question Answering Dataset (THQuADv1.0), which was subsequently updated to THQuADv2.0 by adding questions related to Düzce University units. The pre-trained Turkish BERTurk-base model was then fine-tuned using the THQuADv2.0 dataset and the successful hyperparameters and fine-tuning values obtained from the English model.The training resulted in the creation of the BERTDuQuA (BERT Duzce University Question Answering) model for Turkish question answering, which achieved high performance with an F1 score of 87.10% and an EM of 76.90%. A novel model was devised by incorporating the BiLSTM layer into the BERT model. Training was conducted with the THQuADv2.0 dataset, utilising the fine-tuning technique with the optimal hyperparameters identified in the Turkish model. The outcome of this training was the BERTBiDuQuA (BERT&BiLSTM Duzce University Question Answering) model, which was developed for Turkish question answering. The BERTBiDuQuA model demonstrated high success, achieving an F1 score of 88.84% and an EM of 78.43%.

Keywords: Deep Learning, Natural Language Processing, Text Mining, Question Answering System

EXTENDED ABSTRACT

QUESTION ANSWERING SYSTEM WITH TEXT MINING AND DEEP NETWORKS

Hüseyin Avni ARDAÇ

Düzce University
Graduate School, Department of Computer Engineering
Doctoral Thesis
Supervisor: Prof. Dr. Pakize ERDOĞMUŞ

June 2024, 129 pages

1. INTRODUCTION

In the contemporary era, a plethora of studies are being conducted on text in various disciplines. Text analysis is a crucial tool in numerous fields, including sentiment analysis, machine translation, text generation, summarization, and question answering. Question answering using text mining and deep networks is a technique that aims to automatically identify answers to questions from textual data, leveraging natural language processing and knowledge extraction techniques. Research in this area is focused on developing various approaches for comprehending textual information, interpreting it in a particular context, and producing accurate responses. Question answering involves automatically providing solutions to queries posed in natural language. These systems constitute a form of information retrieval that incorporates three integral components: comprehension of the posed question, retrieval of relevant data, and derivation of a precise answer. The prevalence of question-answering systems has increased significantly in recent years due to the necessity for companies to reduce expenses and guide clients in a beneficial direction. Unlike conventional information retrieval, a question-answering system directly processes inquiries in natural language. Based on the user's information requirements, the system then provides the most suitable response to the submitted question. The objective of this study is to develop a question-answering system that can accurately respond to content-related inquiries. To achieve this, the model was trained using various hyperparameter values to determine the optimal answer. Transducer-based neural networks were employed in the study to obtain stronger representations with a Self-Attention structure and to handle long-range dependencies. The results were evaluated using EM and F1 score. Furthermore, a web application was developed to demonstrate the efficacy of the models in responding to queries.

2. MATERIAL AND METHODS

We will begin by describing the datasets used to train and test the question-answering system under investigation. We will then discuss BERT and self-attention structures, which are the approaches utilized for question answering. Finally, we will explain our proposed method for question answering in detail.

Two language datasets were utilized in the study. For English question answering, we employed the SQuADv1.1 dataset, developed by Stanford University, which is specifically designed for the question answering domain. SQuADv1.1 is a reading comprehension dataset that contains answers to each question of the paragraphs in Wikipedia articles, comprising 23 215 paragraphs and 107 785 question-answer pairs from 536 Wikipedia articles. The solutions to the questions are not multiple-choice; there is only one answer with the highest likelihood.

THQuADv1.0 was used to answer the Turkish questions. Questions belonging to the guidelines of Düzce University were added to the dataset. The THQuADv1.0 dataset contains 841 titles, 2 701 passages, and 15 554 questions and answers. The Düzce University Guidelines dataset comprises 15 titles, 74 passages, and 510 questions and answers. The two datasets were combined to form the THQuADv2.0 dataset, which contains 856 titles, 2 775 passages, and 16 064 questions and answers.

BERT (Bidirectional Encoder Representations from Transformers) is a language representation model designed for the extraction of pre-trained deep, bidirectional representations. At the level of each network layer, the context is taken into account for each word, with the bidirectional transformers used in this model allowing for this to be achieved on both sides. The pre-trained BERT representations can then, by means of fine-tuning, be used to achieve state-of-the-art performance on a wide variety of tasks.

BiLSTM (Bidirectional Long Short-Term Memory) model is a neural network architecture that combines two LSTM layers, each operating in parallel on sequence data in the forward and reverse directions. This structure provides a more comprehensive understanding of context by utilizing information from both past (previous) and future (next) time frames. BiLSTMs are widely used in language modeling, question answering, text classification, speech recognition, and other natural language processing tasks. The bidirectional flow of information facilitates the model's ability to more effectively learn dependencies and relationships in sequence data, thereby enhancing its performance.

The BERT-base model is a neural network architecture comprising 12 layers, 768 dimensions, and 110 million parameters. The training corpus consists of the BooksCorpus and English Wikipedia corpora, which collectively incorporate 800 million and 2,5 billion words, respectively. The BERTurk-base model comprises 12 layers and 768 dimensions. The model is trained on a filtered and sentence-segmented version of the Turkish OSCAR corpus, a recent Wikipedia dump, various OPUS corpora, and a special corpus provided by Kemal Oflazer. As BERT is trained on large amounts of textual data, it is essential to perform adequate pre-processing of the data to ensure the effective use of BERT.

The proposed system is designed to cultivate two distinct language models in English and Turkish with the objective of effectively responding to user inquiries. The specific actions that have been taken to achieve this objective are outlined below. The English model was subjected to a series of tests, during which various parameters were evaluated, including the learning rate, size, and number of cycles. The most effective hyperparameters were applied using a fine-tuning technique. The English model was trained using the these parameters: epoch values of 10, 20, 30, and 50; learning rates of 1e-05, 2e-05, 3e-05, 4e-05, and 5e-05, and batch sizes of 2, 4, 8, 16, and 32. THQuADv1.0 dataset underwent improvement work. The Turkish spelling conventions were applied, and erroneous data was removed and organized according to standard practices. The dataset was divided into sections to facilitate analysis. The model's ability to generalize was enhanced by incorporating inquiries related to the units of Düzce University into the dataset. Two different Turkish models were created. The first one was created with the basic model BERTurk-base. The second one was created by adding BERTurk-base with BiLSTM. The hyperparameters that optimized as a result of the employed optimizastion task in the English model were applied with a fine-tuning technique. Turkish models was trained using the these hyperparameter values: The following hyperparameters were applied: epochs 10, 20, 30, and 50; learning rates 1e-05, 2e-05, 3e-05, 4e-05, and 5e-05; and batch sizes 2, 4, 8, 16, and 32. A web application was developed using the BERTDuQuA model, which was created by training the BERTurk model using the THQuADv2.0 dataset. With this application, users can find answers to their questions about Düzce University Regulations.

3. RESULTS AND DISCUSSIONS

The objective of this study is to develop an appropriate question-answering system that can address inquiries pertaining to a particular subject area. The initial phase involved the utilization of the SQuADv1.1 dataset and the English BERT-base model to establish the optimal hyperparameters, which were subsequently determined for the targeted model. Statistical analysis revealed that the targeted model exhibited remarkable performance,

achieving an F1 score of 88,13% and an Exact Match (EM) rate of 80,74%, which surpassed the results declared in previous studies. Improvement studies were conducted on THQuADv1.1, the Turkish question and answer dataset, resulting in the removal of erroneous data. To improve generalization capabilities, the model's dataset was augmented by incorporating question and answer data pertaining to Düzce University's policies and departments. To enhance the question answering system training, we employed the BERTurk-base model and utilized a hyperparameter tuning method. The parameters that yielded success results in the English model were then utilized as hyperparameters. This resulted in the creation of our BERTDuQuA model for THQuADv2.0, which exhibited the high performance among Turkish language models with an EM score of 76.90% and F1 score of 87.10% when compared to previous studies. A novel model was devised by incorporating the BiLSTM layer into the BERT model. Training was conducted with the THQuADv2.0 dataset, utilising the fine-tuning technique with the optimal hyperparameters identified in the Turkish model. The outcome of this training was the BERTBiDuQuA model, which was developed for Turkish question answering. The BERTBiDuQuA model demonstrated high success, achieving an F1 score of 88.84% and an EM of 78.43%. This models were successfully integrated into a web application and demonstrated the capability to accurately answer the presented questions.

4. CONCLUSION AND OUTLOOK

In future studies, it would be beneficial to improve the data set, increase the number of question-answer pairs, and enhance the architecture. The focus should be on defining new approaches to continuously enhance the performance of existing question answering systems and to produce more effective solutions to their problems. Using domain-specific data sets for models can result in more precise and consistent answers to questions related to that domain. Future research and development in this field should focus on various aspects, including the use of question answering systems in different industrial applications, ethical and security issues, and user experience and interaction. Question and answer systems have significant potential in various sectors, and it is predicted that research in this field will continue to grow.

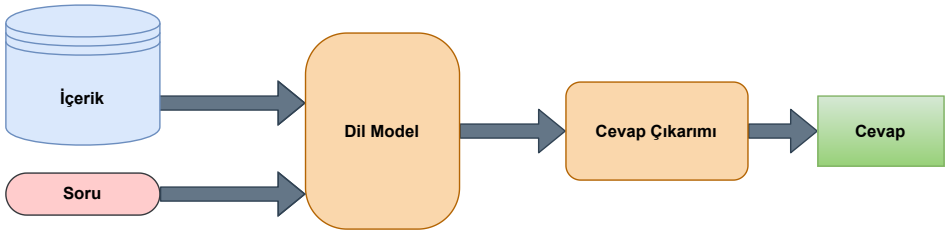
1. GİRİŞ

Bilgisayarlar dili anlama konusunda her zaman zorluklarla karşılaşmıştır. Metinsel girdileri toplayabilir, saklayabilir ve okuyabilirlerken, temel dili yorumlayacak bağlamdan yoksundurlar. Bu sorunu çözmek için, bilgisayarların insan dilini bilgisayar ortamında anlamasını ve yorumlamasını sağlayan Doğal Dil İşleme (Natural Language Processing - NLP) alanı ortaya çıkmıştır. NLP, bilgisayarların insanlar gibi metinleri ve konuşulan kelimeleri okumasını, analiz etmesini, yorumlamasını ve anlamasını sağlamayı amaçlayan yapay zekânın bir alt alanıdır. NLP, bilgisayarların insan dilini anlamasına yardımcı olmak için dilbilim, istatistik ve makine öğrenimini birleştirir.

NPL, derin öğrenmede zorlu bir problemdir çünkü bilgisayarlar ham metinler üzerinden öğrenme gerçekleştiremezler. Bilgisayarın gücünden faydalanmak için kelimelerin bir modele girmeden önce vektörlere dönüştürülmesi gerekir. Elde edilen vektörlere kelime gömme adı verilir. Bu gömmeler daha sonra soru cevaplama, metin oluşturma, isim varlık tanıma veya makine çevirisi gibi istenen görevi çözmek için kullanılabilir. Vektörlerin akıllıca işlenmesi sayesinde model, bazı görevlerde insan yetenekleriyle eşdeğer performans gösterebilir. Vektörleri işlemek için bir doğal dil modeli oluşturmak gerekir. Doğal dil işlemenin soru cevaplama problemini çözmek için kullanılan Transformatörlerden Çift Yönlü Kodlayıcı Temsilleri (Bidirectional Encoder Representations from Transformers - BERT) dil modeli ele alınacaktır.

Günümüzde bilgiye erişmek için en yaygın tercih edilen yöntemlerden biri arama motorlarıdır. Günümüz arama motorları çeşitli çarpıcı özelliklere sahip olmakla birlikte, bilgi bolluğu ve kullanıcıların aradıkları konularla ilgili doğru verilere ulaşamamaları durumunda, kullanıcının sorularına tam ve kesin bir yanıt sağlamada sınırlı kalabilmektedir. Bu nedenle, otomatik olarak doğru ve kesin cevapları sağlayabilen araçlara duyulan ihtiyaç giderek artmaktadır. Bu ihtiyacı karşılamak için geliştirilen Soru Cevaplama Sistemleri (Question Answering System - QAS), NLP alanının bir alt dalı olarak günümüzde başarıyla kullanılmakta ve güncelliğini sürdürmektedir. Günümüzde metinler üzerinde

farklı alanlarda birçok çalışma gerçekleştirilmektedir. Duygu analizi, makine çevirisi, metin üretme, metin özetleme ve soru cevaplama gibi birçok alanda metinlerin analizine gerek duyulmaktadır [1]–[3]. Metin madenciliği ve derin ağlar ile soru cevaplama, doğal dil işleme ve bilgi çıkarma tekniklerinin birleşimini kullanarak metin verilerinden sorulara otomatik olarak cevaplar bulma amacını taşır [4]. Bu alanda yapılan araştırmalar, metinlerdeki bilgilerin anlaşılması, belirli bir bağlamda yorumlanması ve doğru cevapların çıkarılması için çeşitli yöntemlerin geliştirilmesine odaklanmaktadır [5]. Soru cevaplama, doğal dilde sorulan bir soruyu otomatik olarak cevaplama işleminin gerçekleştirilmesiyle oluşur. Bu sistemler soruyu anlama, bilgiyi getirme ve cevap çıkarımını olmak üzere üç bileşeni olan bir bilgi alma türüdür [6]. Soru cevap sistemleri şirketlerin maliyetlerini azaltma ve müşteriyi doğru yönlendirme doğrultusunda ihtiyaç duymasından dolayı oldukça önem kazanmaktadır. Geleneksel bilgi getirim sistemiyle karşılaştırıldığında, soru cevap sistemi kullanıcının doğal dilde sorulan sorusunu almaktadır [4]. Ardından, kullanıcının bilgi ihtiyacına göre, gönderilen soruya karşılık gelen en uygun cevapla yanıt vermektedir [7]. Şekil 1.1'de gösterildiği gibi, bir soru cevap sistemi ilk olarak doğal dilde açıklanan soruyu ve içeriği analiz eder, ardından içeriğe uygun olan cevabı tespit eder son olarak da kullanıcıya cevabı getirmektedir.



Şekil 1.1. Soru cevap sisteminin akışı diyagramı.

QAS, doğal dil platformlarının bilgi tabanları (Knowledge Base - KB) ile etkileşime girmesini sağlar. Bu QAS, sorulan sorulara doğrudan ve daha spesifik cevaplar döndürür. Son yıllarda, Google'ın Bilgi Grafiği [8], YAGO [9], DBPedia [10] ve Freebase [11] gibi KB'in popülerliğinin ve kullanımının artmasıyla birlikte insanlar bu bilgi tabanlarına erişmek için etkili yöntemler aramaya daha fazla ilgi duymaktadır. Bu tür bilgi tabanlarının çoğu veri formatı olarak Kaynak Tanımlama Çerçevesi (Resource Description Framework - RDF) benimser ve ayrıca milyarlarca özne, yüklem, nesne üçlüsü içerir [12].

SPARQL [13], Xcerpt [14], RQL [15] dahil olmak üzere bu tür büyük bilgi tabanlarını sorgulamak için tasarlanmış çeşitli diller vardır. Ancak, bu dillerin öğrenilmesi, sorgu diline, sözdizimine, anlambilimine ve KB ontolojisine aşina olunması gerektiğinden bir sınırlama getirmektedir.

Buna karşın, doğal dili sorgu olarak alan bilgi tabanlı soru cevaplama daha kullanıcı dostu bir çözümdür ve son yıllarda araştırma odağı haline gelmiştir [16]. Soru cevaplama görevi için iki ana araştırma akışı vardır: Anlamsal Ayrıştırma (Semantic Parsing - SP) tabanlı sistemler [17], [18] ve Bilgi Erişim (Information Retrieval - IR) tabanlı sistemler [19]–[22]. SP tabanlı yöntemler, doğal dil sorusunu mantıksal formlar gibi koşullu olarak yapılandırılmış ifadelere dönüştüren bir anlamsal ayrıştırıcı oluşturarak soru cevaplama problemini ele alır [23] ve ardından cevabı elde etmek için bilgi tabanı üzerinde sorguyu çalıştırır. SP tabanlı yöntemler üç modülden oluşmaktadır. Varlık bağlama, bir soruda geçen tüm varlıkları tanır ve her bir varlığı bilgi tabanındaki bir varlığa bağlar. Yüklem eşleme, soru için bilgi tabanındaki aday yüklemleri bulur. Cevap seçimi, aday varlık-yüklem çiftlerini bir sorgu ifadesine dönüştürür ve cevabı elde etmek için bilgi tabanını sorgular. IR tabanlı yöntemler, herhangi bir gramer veya sözlük gerektirmeden şemasından bağımsız olarak herhangi bir bilgi tabanını sorgulayabileceği cevapları ve soruları aynı gömme uzayına eşlemeye odaklanır [19]–[22]. IR tabanlı yöntemler, SP tabanlı yaklaşımlara kıyasla daha esnektir ve daha az denetim gerektirir [17], [18].

Anlamsallığı yakalayabilen gömme tekniklerindeki ilerlemeyle birlikte, NLP [23] ve Doğal Dil Anlama (Natural Language Understanding - NLU) [24] gibi çeşitli alanlarda uygulanmaktadır. Bilgi tabanları-soru cevaplama alanında, IR tabanlı bir çatı altında, gömme tabanlı yaklaşımlar önerilmiştir [21], [25]. Ancak bu yaklaşımlar iki kısıtlamayla karşı karşıyadır. Birincisi, bu modeller tüm bilgi tabanlarının temsilini öğrenmeden farklı bileşenleri ayrı ayrı kodlamaktadır. Dolayısıyla, bileşimsel semantiği küresel bir perspektifte yakalayamazlar. İkincisi, Uzun Kısa Süreli Bellek [26], Çift Yönlü Uzun Kısa Süreli Bellek ve Evrişimli Sinir Ağlarının performansı büyük ölçüde büyük eğitim veri setlerine bağlıdır ve bu da genellikle pratikte mevcut değildir. Son yıllarda, büyük ölçekli denetimsiz derlemler üzerinde önceden eğitilmiş dil modelleri [27]–[29], önceki dilbilimsel bilginin otomatik olarak çıkarılmasındaki avantajlarını göstermiştir ve yukarıdaki sorunlarla başa çıkmanın olası bir yolunu göstermektedir.

Yapılandırılmış veya yapılandırılmamış kaynaklardan bilgi çıkarmak, bir arama motoru veya sosyal medya platformu kullandığımızda günlük bir ihtiyaçtır. Akademik bir çalışma alanı olan IR, metinsel veriler, web sayfaları, videolar, belgeler, veritabanları vb. gibi çeşitli kaynaklardan bilgi çıkarma sorununa cevap aramak için geliştirilmiştir. Bir IR sisteminin amacı, kullanıcının ihtiyacıyla ilgili olan ve kullanıcının bir görevi tamamlamasına yardımcı olan bilgileri içeren belgeleri almaktır. Bu görev, ilgili sayfaları sıralamak, soruları cevaplamak veya belgeleri özetlemek olabilir. Kullanıcılar farklı uygulamalara ihtiyaç duysa da ana husus makinelerin doğal dili anlamasını sağlamaktır.

Soru cevaplama, kullanıcının gönderilen bir sorguya cevap beklediği bir bilgi erişim görevidir. Çoğu bilgi erişim sisteminde olduğu gibi bir belge koleksiyonu döndürmek yerine bir veya daha fazla kaynaktan bilgi toplayarak bir cevap bulmayı amaçlar. Örneğin, bir kullanıcı "Türkiye'nin başkenti neresidir?" diye sorduğunda, "Ankara" şeklinde kesin bir cevap bekler, belgeleri veya ilgili sayfaları okumayı amaçlamaz.

Büyük dil modelleri, doğal dil işleme alanında devrim niteliğinde yenilikler sunarak dil anlayışı ve metin üretimi konularında büyük ilerlemeler sağlamıştır. Google AI tarafından geliştirilen BERT, çift yönlü bağlam kullanarak dilin daha derin anlamlarını öğrenme yeteneği ile öne çıkarken, metin anlama ve soru cevaplama sistemleri gibi görevlerde üstün performans sergilemektedir [29]. Diğer taraftan, OpenAI tarafından geliştirilen GPT (Generative Pre-trained Transformer) serisi, tek yönlü bağlam kullanarak doğal ve akıcı metin üretimi konusunda önemli başarılara imza atmış, yaratıcı yazma ve sohbet botları gibi uygulamalarda güçlü bir araç olarak kendini kanıtlamıştır [27], [30]. Bu modeller, çeşitli doğal dil işleme görevlerinde transfer öğrenme yetenekleri sayesinde geniş bir kullanım yelpazesi sunarak, dil teknolojilerinin evriminde kritik rol oynamaktadır.

Bu çalışmada transformatör tabanlı sinir ağları kullanılmıştır [31]. Transformatör mimarisi ile öz dikkat yapısıyla daha güçlü temsiller elde etmeyi ve uzun menzilli bağımlılıkları ele almayı amaçlanmıştır. Sonuçları değerlendirmek için Tam Eşleşme (Exact Match - EM) ve F1 skoru metrikleri kullanılmıştır.

2. LİTERATÜR TARAMASI

QAS, sorulara doğal dilde yanıtlar sağlar. Aynı bilgi doğal dilde farklı şekilde ifade edilebildiğinden, anlamsal olarak eşdeğer sorularda farklı cevaplar üretmek mümkündür [32]. Bu nedenle, QAS oldukça zor bir NLP araştırma alanıdır. Son yıllarda bilgisayarların herhangi bir konuda doğal dilde sorulan sorulara otomatik olarak cevap verebilmesi için birçok çalışma bu alana odaklanmıştır. Bu çalışmalarda Makine öğrenmesi ve derin öğrenme yöntemleri kullanılmaktadır. Bu bölümde literatürdeki çalışmalar anlatılmıştır.

İngilizce dili çalışmalarında belirgin bir rol oynadığı, literatürde bu alandaki çalışmaların büyük bir kısmının İngilizce dilini temel aldığını gözlemlemekteyiz. İngilizce dilinin kurallarının ve yapısının diğer dillere göre daha basit ve anlaşılır olması, dil işleme ve anlama aşamalarında çalışmaların daha kolay üretilebilmesine imkân tanımaktadır. Ayrıca, İngilizce dilinde oluşturulmuş ve analiz için uygun veri setlerinin internet üzerinden kolayca erişilebilir olması, bu dilde yapılan çalışmaların kolaylıkla artmasına ve gelişmesine olanak sağlamaktadır.

Diğer yandan, Türkçe dilinde yapılan metin madenciliği çalışmalarının literatürde yeterince temsil edilmediği gözlemlenmektedir. Özellikle Türkçe literatürde BERT dil modeli ile yapılan çalışmaların sayısının oldukça sınırlı olduğu belirtilmelidir. Bu tür çalışmaların temel amacı, Türkçe literatür için BERT Türkçe dilinde gerçekleştirilen çalışmalara kaynak sağlamaktır.

Chen vd. [33] tarafından 2021 yılında yapılan çalışmada, BIRD-QA adlı bir IR tabanlı alana özel soru cevap yapısı oluşturmak için veri ön işleme stratejileri ve BERT modeli ile önceden eğitilmiş modeller araştırmışlardır. Bu sistemi bir üniversitenin web sitesi verilerini kullanarak alana özel bir bilgi tabanı oluşturulmuştur. Genişletilmiş BERT ve ALBERT tabanlı modellerin birden çok varyasyonunu uygulamışlardır. Stanford Soru Cevaplama Veri Kümesi SQuADv1.1 ve SQuADv2.0 veri kümelerini kullanarak okuduğunu anlama görevi konusundaki yapıyı doğrulamışlardır. Genişletilmiş ALBERT

tabanlı modelleri %75,4 EM puanı ve %78,8 F1 skoru puanı elde etmiştir. Ayrıca bir üniversite web sitesindeki verileri kullanarak küçük bir soru cevaplama fizibilite testini sunulmuştur.

Tieu vd [34], 2021 yılına ait Otomatik Hukuki Soru Cevaplama Yarışması (Automated Legal Question Answering Competition - ALQAC) kullandıkları yaklaşımı anlatmışlardır. İki veri kümesi kullanmıştır; birincisi kanun veri kümesi, ikincisi ise beyan veri kümesidir. Burada iki bileşenli getirme ve seçim kısmı olan bir model sunulmuştur. Bu yarışmada, Görev 1, Görev 2 ve Görev 3'te sırasıyla %88,07; %71,02 ve %69,89'luk skorlarla tüm görevlerin birincilik ödülünü elde etmişlerdir.

Gupta vd. [35] yapmış oldukları çalışmada, çok alanlı çok dilli soru cevaplamanın zorluklarını değerlendirilmiştir. Kıyaslama için gerekli kaynakları oluşturulmuş ve temel bir model geliştirmişlerdir. Web'den altı farklı alanda 500 makale kullanılmıştır. Bu makaleler, 250 İngilizce belge ve 250 Hintçe belgeden oluşan karşılaştırılabilir bir içerik oluşturmuşlardır. Bu karşılaştırılabilir veriler hem İngilizce hem de Hintçe olmak üzere soru ve cevaplardan oluşan 5 495 soru cevap çiftinden oluşmaktadır. Beklenen cevaba bağlı olarak bir girdi sorusunu kategorilere ayırmak için derin öğrenmeye dayalı bir model geliştirilmiştir. Cevaplar, benzerlik hesaplaması ve sonraki sıralama yoluyla çıkarılmıştır. Soru sınıflandırma modelinin değerlendirilmesi %80,30 doğruluk göstermiştir.

Liu vd. [36] web aramalarına dayalı bir soru cevaplama sistem tasarlamışlardır. İlk soru analizi, ikinci bilgi alma ve üçüncü cevap çıkarma modülü olmak üzere üç modül oluşturmuşlardır. Anahtar kelime ve soru tipi çıkarımını soru analizi modülü kullanılarak yapmışlardır. Daha sonra bilgi erişim modülü ilgili ilk 100 web sayfasını aramış ve taramıştır. Çok sayıda aday cevap seçilmiş ve ilgili puanlarla değerlendirilmiştir. Tüm bu işlemlerden sonra, en yüksek puana sahip en iyi cevap kullanıcılara döndürmeyi başarmışlardır.

Uğurlu vd. [37], COVID-19 ile ilgili bu bilgi kirliliğini ortadan kaldırmak ve insanların sorularına güvenilir kaynaklardan elde edilen bilgilerle doğru cevaplar verebilmek için bir akıllı sanal asistan geliştirilmişlerdir. Geliştirdikleri akıllı sanal asistan iki farklı türde soruya cevap vermektedir. Bunlardan birincisi, COVID-19 ile ilgili halk arasında sıkça sorulan bilgi sorularıdır. İkincisi ise dünyadaki ülkelerin COVID-19 vaka sayısı, ölüm

sayısı, test sayısı, vb. sayısal verilerine dair sorulardır. Geliştirdikleri sanal asistanın, kullanıcıların COVID-19 ile ilgili sorularının büyük çoğunluğuna cevap verebildiği görülmüştür.

Allam ve Haggag [38] metodolojinin üç temel bileşeni soru kategorizasyonu, bilgi alma ve yanıt çıkarımıdır. Doğal Dil İşleme kullanılarak tüm bu bileşenler birlikte çalışır. İlk olarak, işleme prosedürü sırasında bir sorgu analiz edilir. Daha sonra sorular kategorize edilir ve yeniden formüle edilir. Daha sonra belge işleme sırasında veri kümelerinden ve web sitelerinden bilgi alınır. Cevap işleme aşamasında cevap tanımlama, çıkarma ve doğrulama tamamlanır. Yapmış oldukları çalışmada kısa cevap için MRR puanı %55,5 elde etmişlerdir.

Chau vd. [39] ne göre anayasa, kanun, tüzük, yönetmelik, emir, ortak karar, kararname, karar, genelge, ortak genelge ve yönerge yasal belgelerin ayrıldığı kategorilerden bazılarıdır. Her bir belge bu durumda geçerlidir. Bu belgeler soru cevaplama formatı için temel teşkil eder. Bu modelin tasarımında; önce soru işleme, ardından belge alma, cevap seçimi ve cevap çıkarma gelir. Bu soru cevaplama modelini oluşturmak için NLP ve BERT yapısını kullanmışlardır. Bu yaklaşımı kullanarak 0,860 hassasiyet, 0,958 geri çağırma ve 0,906 F1 skoru puanı elde etmişlerdir.

Larson vd. [40] yapmış oldukları çalışmada daha önce cevaplanmış belirli bir soru için benzer bir soru elde etmeye çalışmıştır. Yahoo Questions'ın sağlık kategorisinden yaklaşık 100 000 soru cevaplama veri setini kullanmışlardır. Bir soru sorulduğunda, sistem bunu depolanmış olan sorularla karşılaştırır. Anahtar kelimeler eşleşiyorsa veya soru saklanan soruyla neredeyse aynıysa, çıktıyı en iyi soru olarak verir. Bu sistemde, önceden cevaplanmış en alakalı soruları elde etmek için NLP teknikleri ve IR (Information Retrieval) teknikleri kullanılmaktadır. Bu teknik için bigramlar ve unigramlar kullanılmıştır. Bilgi alma Lucene kullanılarak uygulanmıştır. Yahoo araması için MRR skoru 0,1466 ve kendi sistemleri için MRR skoru 0,2382 olarak elde etmişlerdir.

Biswas vd. [41], Li ve Roth veri setindeki 2 000 soruyu kullanmışlardır. Bu makaledeki sorular üç ana kategoriye ayrılmıştır. Birincisi tanımlama tipi, ikincisi betimleme tipi ve üçüncüsü de olgu tipidir. NLP kullanarak her soruyu ayrı bir "Wh" sorgu sınıfına ayırmışlardır. Böylece, "Nasıl" sonucunun doğruluğu %99,6 olarak elde etmişlerdir.

Ranjan ve Balabantaray [42] yapmış oldukları çalışmada bilgi toplamak ve cevap bulmak için Google kullanmışlardır. Daha gerçekçi bir yanıtla hızlı bir test sunmuşlardır. Soruları gruplandırmak için yalnızca beş kategori kullanılmış ve diğer sorular için belirli yönergeler ve standartlar geliştirilmişlerdir. Bunun için olası yanıtları sıralamalarına yardımcı olması için Wikipedia'yı kullanmışlardır. Wikipedia'yı kullandıklarında doğruluk oranı %62,11 olarak elde etmişlerdir.

Day ve Kuo [43], Delta Okuduğunu Anlama Veri Kümesi (DRCD) kullanılmıştır. Bu çalışmada BERT modeli, sınıflandırma için Q-EAT ve AT modelleri ile kullanmışlardır. Bu sistem %77,44 EM puanı elde etmişlerdir.

Dodiya ve Jain [44] yapmış oldukları sistemde bazı kurallar ve kalıplar kullanılarak sorular sınıflandırılmıştır. Soru cevaplama sistemi tamamen tıbbi sorulara dayandığı için tıbbi alanla ilgili cevapları çıkarmıştır. Sorular açıklama, sayısal, konum ve insan kategorileri olarak sınıflandırılmıştır. Doğruluk soru türüne göre hesaplamışlardır: "Ne" için doğruluk oranı %55,33 olarak elde etmişlerdir.

Luo vd. [45] yapmış oldukları çalışmada, varlık bağlama ve ilişki algılama modellerini kullanarak tek ilişkili soru cevaplama (SR-QA) için BERT tabanlı bir teknik sunmuştur. Varlık bağlantısı için BERT'i önceden eğitmişler ve gürültüyü filtrelemek için sezgisel bir yaklaşım kullanmışlardır. Ek olarak, şu anda kullanılan ilişki tanıma teknikleri, anlamlarını belirlemeden önce sorguyu ve aday gerçeği bağımsız olarak modellemişlerdir. Bu modelin Basit Sorular veri kümesi üzerindeki doğruluğu %80,9 olarak elde etmişlerdir.

Devanshi ve Rebecca [46] yapmış oldukları çalışmada, soru cevaplama alanındaki zorlukları ele almışlardır. Amaç, kapalı bir alan faktoid soru cevap sistemi geliştirmek ve bunun için doğal dil işleme yöntemlerini kullanmışlardır. Soru ve cevap adayları oluşturmak için desen eşleme ve bilgi çekme kullanmışlardır. Sorular ve cevaplar, benzerlikleri değerlendirmek için bir özellik uzayına dönüştürmüşlerdir. Model, soru ve cevap cümleleri arasındaki ilişkiyi öğrenmek için evrimsel bir sinir ağı kullanarak aday cevapları sıralamışlardır. Önemli bir nokta olarak, model manuel özellik mühendisliği veya dil özgü veriye ihtiyaç duymadan farklı alanlara uyarlanabilir. TREC QA veri kümesi üzerindeki eğitim ve test sonuçları, bu yaklaşımın umut verici başarı metrikleri sunduğunu göstermişlerdir.

Nikita vd. [47], BERT modeli ve Google Dialogflow ile entegre edilmiş kullanıcı dostu bir Soru Cevaplama Modeli (QAM) tasarlamışlardır. QAM, büyük veri kümesine dayalı sorulara kullanıcılara doğru cevaplar sağlamak amacıyla geliştirmişlerdir. BERT modeli ve sohbet botu, webhook ve API kullanarak birbirleriyle etkileşim sağlamışlardır. Kullanıcı, Dialogflow sohbet botu ile etkileşime girdiğinde, niyetleri eşleştirmiş ve talebi BERT modeline göndermiştir. Sonuç olarak, BERT modeli, sohbet botuna cevap verir ve son kullanıcıya yanıt vermiştir. Bu çalışma, müşteri deneyimini geliştirmek ve okuma anlama görevlerini daha verimli hale getirmek için QAM kullanımını önermişlerdir.

Zhang ve arkadaşları [48], önceden eğitilmiş BERT modelini temel alan sözdizimi güdümlü bir ağ (SG-Net) modeli önermiştir. Ayrıştırma ağacı yapısı ile modele açık sözdizimsel kısıtlamalar ekleyerek daha iyi kelime temsili elde etmeyi amaçlamışlardır. SG-Net modelleri SQuADv2.0 için %85,1 EM ve %87,9 F1 skoruna ulaşmıştır.

Zhang ve arkadaşları [49] SemBERT adı verilen semantik duyarlı bir model önermiştir BERT modeline dayanmaktadır. Modellerinin yapısı anlamsal bir yapıdan oluşur rol etiketleyici, bir dizi kodlayıcı ve anlamsal entegrasyon bileşenleri. Onlar ile aynı ağırlıkları ve ince ayar prosedürlerini kullanarak modeli analiz etmiştir. BERT modeli. SQuADv2.0 sonuçları incelendiğinde, SemBERT model %80,9 EM ve %83,6 F1 skoruna ulaşmıştır.

Hu ve diğerleri [50] dikkat okuyucu yöntemini geliştirmiş ve Güçlendirilmiş Anımsatıcı Okuyucu modelini önermiştir. İlk aşamada, çok turlu hizalama mimarilerinde dikkat eksikliği ve fazlalığı sorunlarını azaltan dikkat mekanizmasını kurmuşlardır. Daha sonra yakınsama bastırma sorununu ele almak için dinamik-kritik takviyeli öğrenme yaklaşımını geliştirdiler. Önerdikleri model sonucunda, bu model SQuADv1.1 için %88,5 F1 skorunu elde etmişlerdir.

Liu ve diğerleri [51], dikkat modelleri için iki önemli şekilde bir faz iletkeni (PhaseCond) çerçevesi önermiştir. Bu çerçeve, bir dikkat katmanları yığını ve bilgi akışını düzenleyen bir iç veya dış füzyon katmanları yığını uygulayan birden fazla aşamadan oluşur. Ayrıca, soru ve pasaj gömme katmanlarını farklı bakış açılarından kodladılar ve PhaseCond için nokta çarpımı dikkat işlevini geliştirdiler. PhaseCond çerçevesi ile SQuADv1.1 için %84 F1 skoruna ulaşmışlardır.

Pan ve diğerleri [52], geçmiş dikkat yöntemlerinde kullanılan vektörlerin sorgu cümlesindeki anahtar kelimelerin ağırlığını hafife alması nedeniyle Hafıza Ağı ile Çok Katmanlı Gömme (MEMEN) sinir ağı mimarisi geliştirmiştir. Modelin kodlama katmanında, kelimelerin sözdizimsel ve anlamsal bilgisi için klasik skip-gram modelini kullanmışlardır. Ayrıca metinlerdeki önemli bilgileri yakalamak için bir hafıza ağı önermişlerdir. Bu bellek ağını sorgu ve parçacığın tam yönlendirme eşleşmesinden türetmişlerdir. Önerdikleri MEMEN modeli ile SQuADv1.1 için %82,66 F1 skoru elde etmişlerdir.

Xiong ve arkadaşları [53], geleneksel crossentropy kaybını pekiştirmeli öğrenme ve eğitilmiş bir kelime örtüşme ölçüsü ile birleştiren hibrit bir hedef önermiştir. Hedef için, değerlendirme ölçütü ile optimizasyon hedefi arasındaki uyumsuzluğu gidermek için kelime örtüşmesinden kaynaklanan ödülleri kullandılar. Ayrıca dinamik ortak dikkat ağlarını da geliştirdiler. Deneyler sonucunda model, soru türleri ve girdi uzunlukları arasında uzun sorular için iyi performans gösterdi. Analiz aşamasında, bu model SQuADv1.1 için %78,9 EM ve %86 F1 skoruna ulaşmışlardır.

Huang ve diğerleri [54], kelime yerleştirme seviyesinden en üst seviye temsile kadar tüm bilgileri kullanan bir yaklaşımın soruyu cevaplamada daha başarılı olacağını düşünmüştür. Bu nedenle FusionNet adında bir model önermişlerdir. Tüm temsil katmanlarında sinir ağları kullanan modellerin öğrenilmesi zor olduğundan, bu katmanları kullanarak daha az eğitim yükü ile bir dikkat puanlama fonksiyonu elde etmişlerdir. Bu dikkat fonksiyonunu çok seviyeli bağlam katmanlarında kullandılar. SQuADv1.1 üzerinde yapılan analiz sonucunda %85,9 F1 skoru puanı elde etmişlerdir.

Liu ve diğerleri [55] QA için stokastik cevap ağı adı verilen çok adımlı bir sinir ağı modeli önermiştir. Modelin eğitimi sırasında muhakeme adımlarının sayısını değiştirmişlerdir. Ayrıca, cevap üretme modülündeki son katman tahminlerinde stokastik bırakma işlemi uygulanmıştır. Kod çözme sırasında, son adım yerine tüm adımlardaki tahminlerin ortalamasını dikkate alan cevaplar üretmişlerdir. Önerdikleri model, birbirini takip eden adımlarda tahmini iyileştirirken, her adım hala aynı cevabı üretmek için eğitilmektedir. Deneysel çalışmalar sonucunda modelin sağlamlığını ve veri kümeleri için

doğruluk değerini önemli ölçüde artırdığını göstermişlerdir. SQuADv1.1 için sonuçlar incelendiğinde %86,49 F1 skoru değeri ölçülmüştür.

Salant ve Berant [56], bağlamın okuduğunu anlama görevi üzerindeki etkisini araştırmıştır. Bunun için bağlamsal ve bağlamsal olmayan temsilleri ayırarak bağlam kullanımının olumlu etkisini inceleyen bir nöral modül önermişlerdir. Bu modül ile bağlamsal ve bağlamsal olmayan temsiller arasında geçiş yaparak jeton gömme işlemini gerçekleştirmişlerdir. Önerilen modülün önceden eğitilmiş bir dil modeline eklenmesiyle bağlamsal bilginin modele aktarılması sağlanmıştır. Modüllerini SQuADv1.1 için geliştirme setinde analiz etmişler ve %84 F1 skoru elde etmişlerdir.

Hu ve arkadaşları [57], okuduğunu anlama görevi için cevaplanmamış soruların belirlenmesinde kullanışlı bir oku-sonra-doğrula sistemi önermiştir. Önerdikleri sistem, sorular için aday cevapları çıkarmanın yanı sıra cevapsız sorular için olasılıkları da hesaplamaktadır. Ayrıca sistemde, girdi metnindeki pasaj ve soruya göre aday cevabın gerekli olup olmadığına karar veren bir cevap doğrulayıcı kullanmışlardır. Önerdikleri sistemi SQuADv2.0 üzerinde analiz etmişler ve %74,2 F1 skoru elde etmişlerdir.

Wang ve Jiang [58], okuduğunu anlama modellerinin insanların bilgilerinin sinir ağları ile entegrasyonunu araştırmıştır. Her bir pasaj-soru çiftinden anlamsal bağlantıları çıkarmak için WordNet kullanan bir yöntem önermişlerdir. Ayrıca, çıkarılan bu bilgileri kullanabilen Bilgi Destekli Okuyucu adlı uçtan uca bir model geliştirmişlerdir. Modellerin analiz aşamasında SQuADv1.1 için %83,5'lik bir F1 skoru değeri vermiştir.

Sorunun yanıtlanabilir olup olmadığını belirlemek için Sun ve diğerleri [59] uçtan uca öğrenilebilen U-Net birleşik modelini önermiştir. U-Net modeli üç bileşenden oluşmaktadır: aday cevapları tahmin eden bir cevap işaretleyicisi, cevap olmadığında bir metin aralığı seçmeyen bir cevapsız işaretleyicisi ve sorulara cevap verilmediğini gösteren bir cevap doğrulayıcısı. Geliştirdikleri evrensel düğüm sonucunda soru ve bağlam geçişini tek bir bitişik belirteç ile elde etmişlerdir. Bu düğüm hem soru hem de pasaj üzerinde ilerletilerek sorunun cevaplana bilirliği öğrenilmektedir. U-Net modelini SQuADv2.0 üzerinde test etmişler ve %72,6 F1 skoru elde etmişlerdir.

Ram ve arkadaşları [60] yinelemeli aralık seçimi adı verilen yeni bir ön eğitim aşaması önermiştir. Birden fazla tekrar eden aralık kümesine sahip snippet modele verildiğinde, bir aralık hariç her kümedeki tüm tekrar eden aralıkları maskelerler. Bu işlemle, maskelenen her bir aralık için pasajdaki doğru aralığı seçmeyi amaçlamışlardır. Maskelenen açıklıklar, yanıt aralığını belirlemek için özel bir işaretleyici ile değiştirilmiş ve sistem işaretleyicilerle eğitilmiştir. Bu eğitim aşamasını SQuADv2.0 için test etmişler ve %72,7 F1 skoru ölçmüşlerdir.

Rohit vd. [61], SQuAD veri kümesi temel alınarak soru cevaplama alanında makine öğrenimi mimarileri kullanmışlardır. İlk model GloVe ile unsupervised öğrenme kullanmış ve çift yönlü-LSTM ile eğitmişlerdir. Yaptıkları eğitimin doğruluğu %64,93; test doğruluğu %60,33 olmuştur. İkinci modelde ise InferSent kullanılmış ve XG Boost ile Multinomial Logistik Regresyon algoritmaları kullanılarak eğitmişlerdir. Yaptıkları eğitimin doğruluğu %70,02; testte %66,03 başarısını elde etmişlerdir.

Chakraborty ve arkadaşları [62], biyomedikal alandaki araştırmacılar için literatür taraması sağlayan metin tabanlı bir veri madenciliği aracı önermiştir. Bunun için QA için BERT modeline dayalı nöral tabanlı derin bağlamsal bir model geliştirmişlerdir. Eğitim aşamasında biyomedikal alandaki en büyük veri kümelerinden biri olan BREATHE7 veri kümesinden faydalanmışlardır. Analiz sonucunda QA ince ayar görevlerinde en gelişmiş sonuçları elde etmişlerdir.

Yoon ve arkadaşları [63], önceden eğitilmiş bir biyomedikal dil modeli olan BioBERT'in biyomedikal sorular üzerindeki başarısını analiz etmiştir. Sorular bilgi, liste ve evet/hayır sorularından oluşmaktadır. Modelin başarısının ölçüldüğü 7. BioASQ Yarışması'nda (Görev 7b, Aşama B) en iyi performansı göstermişlerdir.

Farklı soru cevaplama uygulamalarına genelleme yapabilmek için Su ve diğerleri [64] görevler arasında paylaşılan temsili öğrenebilen bir yapı önermiştir. Önceden eğitilmiş bir dil modeline uygulamışlardır. Modellerinin başarısı birçok veri kümesinde ince ayar yapılarak ölçülmüştür. Analiz sonucunda, %56,59 EM ve ortalama %68,98 F1 skoru ile önerdikleri önceden eğitilmiş modelin BERT modelinden daha başarılı olduğunu göstermişlerdir.

Beltagy ve arkadaşları [65], BERT'e dayalı önceden eğitilmiş bir LM olan SciBERT'i önermiştir, bilimsel veri eksikliğini gidermek için. Ön eğitim aşamasını şu konularda gerçekleştirdiler geniş birçok alanlı bilimsel yayın topluluğu. BERT ile karşılaştırıldığında, önerdikleri model oldukça başarılı olmuştur.

Zhou ve arkadaşları [66], topluluk odaklı bir çevrimiçi soru cevaplama web sitesi olan topluluk soru cevaplamada (CQA) cevap seçimi için bir tekrarlayan konvolüsyonel sinir ağı (RCNN) önermiştir. Önerdikleri yöntem, hem soru ile cevap arasındaki anlamsal eşleşmeyi hem de cevap dizesine gömülü anlamsal korelasyonları yakalamak için konvolüsyonel sinir ağlarını tekrarlayan sinir ağı ile birleştirmektedir. Analiz sonucunda, RCNN'nin temel model üzerinde iyileştirme sağladığını göstermişlerdir.

Martinez-Gil ve diğerleri [67], çoktan seçmeli soruların otomatik olarak yanıtlanması için yeni bir yöntem önermiştir. Hukuk sorularından oluşan bir veri kümesi üzerinde uyguladıkları ampirik değerlendirme sonucunda, önerilen yöntemin olumlu etkisini göstermişlerdir.

Esposito ve diğerleri [68], belgelerden ilgili cümlelerin alınmasını iyileştirmek için sözlüksel kaynaklara ve kelime gömülmelerine dayalı karma bir Sorgu Genişletme yaklaşımı önermiştir. İlk olarak, MultiWordNet'ten sorudaki ilgili terimlerin eş anlamlılarını ve hipernonimlerini almışlar ve bunları kullanılan belge koleksiyonu ile bağlamsallaştırmışlardır. Son olarak, Word2Vec'in üzerine inşa edilen bir anlamsal benzerlik metriği ile elde edilen küme, ifadelere ve sorunun anlamına göre sıralandı ve filtrelendi.

Yeh ve Chen [69], soru cevaplama sistemi için QAInfomax adı verilen ve modellerin öğrenme sırasında verilerdeki yüzeysel önyargılara yakalanmamasına yardımcı olmayı amaçlayan alternatif bir yaklaşım önermiştir. Bunun için paragraflar, sorular ve cevaplar arasındaki karşılıklı bilgiyi maksimize etmişlerdir. Önerilen QAInfomax, Adversarial-SQuAD veri kümesi üzerinde ek eğitim verisi olmadan en iyi performansı elde etmiştir.

Carrino ve diğerleri [70], SQuADv1.1'i otomatik olarak İspanyolcaya çevirmek için Translate Align Retrieve yöntemini geliştirmiştir. Daha sonra bu veri kümesini ince ayarlı

çoklu BERT dil modelini eğitmek için kullandılar. Sonuç olarak, modelleri diller arasında kullanılan MLQA ve XQuAD kıyaslama ölçütleriyle analiz ettiler. Analizleri sonucunda, İspanyolca MLQA derleminde %68,1 F1 skoru ve İspanyolca XQuAD derleminde %77,6 F1 skoru elde etmişlerdir.

SQuAD'den esinlenen Möller ve arkadaşları [71], COVID-19'daki makaleleri kullanarak 2019 soru/cevap çiftinden oluşan bir COVID-QA veri kümesi oluşturmuştur. SQuAD üzerinde ince ayarı yapılan RoBERTa modelini COVID-QA ile eğiterek analizi gerçekleştirmişlerdir. Deneysel çalışmalar sonucunda COVID-QA için %59,53 F1 skoru elde etmişlerdir.

Farsça soru cevaplama veri kümelerinin eksikliği nedeniyle, Abadani ve diğerleri [72] SQuADv2.0'ı çevirerek oluşturulan Farsça Soru Cevaplama Veri Kümesini (ParSQuAD) elde etmişlerdir. Çeviriyi manuel ve otomatik veri kümeleri olmak üzere iki versiyonda oluşturmuşlardır. BERT, ALBERT ve çok dilli BERT modelleri ile her iki veri kümesini de analiz etmişler ve manuel için %56,66; otomatik için %70,84 F1 skoruna ulaşmışlardır.

Chen J ve arkadaşları [73], BERT 'in alan bilgisini genişletmek için yarı yapısal bilgileri kullanan konu tabanlı bir alan uyarlama yöntemi önermektedir. Deneysel sonuçlar, müşteri sağlık sorularının yanıtlanmasında %4,8'lik bir doğruluk artışı da dahil olmak üzere biyomedikal NLP görevlerinde önemli gelişmeler olduğunu göstermektedir.

Arezoo Saedi ve arkadaşları [74] yapmış oldukları çalışmada Tüketici Sağlık Sorularını doğru bir şekilde sınıflandırmak için deneyler yapılmıştır. Sorular, belirli özellikler, kelime gömme, cümle gömme ve ince ayar tabanlı tekniklerle temsil edilmiştir. Yapılan deneyler sonucunda, BERT tabanlı modelin Genetik ve Nadir Hastalıklar (GARD) üzerinde %86, Yahoo! Answers tabanlı veri setinde ise %80,4 doğruluk oranına sahip olduğu tespit edilmiştir. Ayrıca, A Robustly Optimized BERT Pretraining Approach (RoBERTa) büyük modele dayalı yöntem, GARD veri setinde %86,2 doğruluk oranıyla BERT-büyük modele göre daha iyi performans sergilemiştir.

Gupta ve arkadaşları [75] yaptıkları çalışmada, doğal dil işleme (NLP) yöntemlerinden MLM ve NSP ile birlikte BERT tokenization ve YAKE tekniği kullanılarak mevcut ÇSY odaklarını belirlemek amacıyla Forbes'in Hindistan'ın 2021'deki en iyi şirketlerinin

sürdürülebilirlik raporlarının tematik bir analizi yapılmıştır. Deney %98 doğrulukla tahminle sonuçlanmıştır.

Kotstein ve arkadaşları [76] doğal dil sorgularıyla eşleşen bir sözdizimi yapısı içinde Web API öğelerini tanımlamak için bir Transformer modeli önermişlerdir. Bu model, Transformer tabanlı soru cevaplama ile anlamsal arama görevini çözer ve önerdikleri yaklaşımın farklı görevlerde uygulanabilirliğini göstermiştir. 2 321 OpenAPI dokümantasyonundan örneklerle farklı veri kümeleri hazırlarız ve önceden eğitilmiş BERT modellerini bu iki göreve göre ince ayarlamışlardır. İnce ayarlı modelinin genelleştirilebilirliğini ve sağlamlığını değerlendirdiler ve parametre eşleştirme için %81,95 ve uç nokta bulma görevi için %88,44 doğruluk elde etmişlerdir.

Liu ve arkadaşları [77], mineral bilgisi edinmek için bilgi grafiğine dayalı bir Çince mineral soru ve cevaplama sistemi önerilmişlerdir. Oluşturulan bilgi grafiği 22 568 varlık ve 91 699 ilişkiye sahiptir. Sistemin doğruluk oranı %91,2'dir.

Kazemi ve arkadaşları [78] yaptıkları araştırmada, Farsça haber makaleleri için bir Soru Cevaplama (QA) sistemi tasarlamayı amaçlamaktadır. Farsça haber alanında en çok Ne ve Kim sorularının, en az ise Neden ve Hangi sorularının yer aldığını göstermektedir. FarsNewsQA, Farsça haberlerle ilgili soruları cevaplamak için BERT, ParsBERT ve ALBERT modellerini kullanarak geliştirilmiştir. En iyi versiyonu, Stanford Üniversitesi'nin İngilizce SQuAD veri kümesindeki QA sistemi ile karşılaştırılabilir %75,61'lik bir F1 skoru sunmaktadırlar.

Wang vd. [79], Bilgi Tabanlı Soru Cevaplama (KBQA) sistemleri için gelen gürültülü soruların sağlamlığını değerlendirmek ve geliştirmek amacıyla bir yöntem ve model önerilmektedir. Orijinal SimpleQuestions veri kümesine dayalı olarak 29 farklı gürültülü soru içeren bir veri kümesi oluşturulmuştur. Deneysel sonuçlar, modelin bu veri kümelerinde %78,1 ortalama doğruluk elde ettiğini ve diğer modellere kıyasla daha iyi performans gösterdiğini göstermektedir.

Tohma ve arkadaşları [80] yaptıkları çalışmada, Türkçe soru cevaplama sistemleri için duygusal performansı artırmak amacıyla bir hibrit duygu analizi modeli önermektedir. Metin ön işleme adımları kullanılarak Türkçe soru cevaplama duygu veri kümesi

hazırlanmıştır. Vektörleştirilmiş ve genişletilmiş soru cevaplama metin vektörleri makine öğrenimi algoritmaları ile eğitilip test edilmektedir. Bu yöntem kullanılarak duygu analizinde %91,05 doğruluk değerine ulaşılmaktadır.

Wang ve arkadaşları [81] yaptıkları çalışmada, araştırma sınırlamasını doldurmak için terim işlevine duyarlı bir anahtar kelime atıf ağı önermektedir. Anahtar kelimelerin korelasyon yapısını temsil etmek için bir soru-yöntem terim atıf ağı oluşturmuşlardır. Daha sonra, bilim haritalama analizindeki üstünlüğünü doğrulamak için oluşturulan ağın topoloji özelliklerini, soru-yöntem çift taraflı ağını ve bilgi topluluğu yapısını araştırmışlardır. Etkinliğini değerlendirmek için Association for Computing Machinery (ACM) dijital kütüphanesinden toplanan 299 567 konferans bildirisinden oluşan bir veri kümesi kullanılmıştır. Sonuçlar, BERT 'e dayalı terim fonksiyonu tanımlama modelinin %90 F1 skoruna ulaştığını göstermektedir.

Kim vd. [82] yaptıkları çalışmada, ASP problemine çözüm yöntemi önerilmiştir. Mevcut giriş biçimi olan soru-paragraf, üç bağımsız girişe dönüştürülmüştür: soru, paragraf ve birleştirilmiş RoBERTa çıkışları. Birleştirilmiş matris daha sonra iki matrise dönüştürülmüş ve ko-dikkat kullanarak makine okuma anlama algoritması önerilmiştir. Modelin performansı, anlama yeteneği ve tasarım verimliliği değerlendirmek ve analiz etmek için bir ayrım çalışması yapılmıştır. Deneysel sonuçlara göre, önerilen yöntem, mevcut yöntemlere kıyasla EM'yi %0,9 ve F1 skorunu %1 artırmıştır.

Çizelge 2.1'de literatür taraması sonucunda BERT modeli ve türevleninin soru cevaplama görevindeki performansını çeşitli metrik ve veri seti kümeleri üzerinde karşılaştıran çalışmaların özetini sunmaktadır. Farklı veri kümeleriyle yapılan deneylerin sonuçlarını, BERT'in doğruluk oranları ve diğer performans ölçütlerini içermektedir. Çizelge 2.1'in sağladığı bilgiler, araştırmacılara BERT ve türevlerinin çeşitli veri kümelerindeki etkinliği hakkında kapsamlı bir bakış sağlamaktadır ve gelecekteki çalışmalar için bir referans noktası olarak hizmet edebilir.

Çizelge 2.1. Literatürdeki ilgili çalışmaların bir karşılaştırması.

No	Yazar / Yıl	Metrikler	Model / Yapı	Veri Seti
1	Larson vd. [40] / 2006	Ortalama Karşılıklı Sıralama (Mean Reciprocal Rank - MRR) skoru 0,2382	Good-Turing bigram dil modeli	Yahoo Questions'ın sağlık kategorisinden yaklaşık 100 000 soru cevap çifti
2	Allam ve Haggag [38] / 2012	%55,5 MRR	Kural tabanlı yaklaşımla soru sınıflandırma	TREC8 derleminin 200 sorusu
3	Liu vd.[36] / 2014	0.5741 Maksimal Marjinal Uygunluk (Maximal Marginal Relevance - MMR)	SVM model	Kişi, yer, zaman, miktar, tanımla ve diğer soru tipi şeklinde, yapay 300 soru oluşturmuşlar
4	Biswas vd. [41] / 2014	%99,6 Doğruluk Oranı	Kural tabanlı yaklaşımla soru sınıflandırma	Li ve Roth veri setindeki 2 000 soru
5	Ranjan ve Balabantaray [42] / 2016	%62,11 Doğruluk Oranı	Bilgi temelli açık alan	Li ve Roth veri setindeki 2 000 soru
6	Dodiya ve Jain [44] / 2016	%55,33 Doğruluk Oranı	Kural tabanlı yaklaşımla soru sınıflandırma	500 tıbbi soru
7	Hu vd. [50] / 2017	%88,5 F1 Skoru	Dinamik-kritik pekiştirmeli öğrenme	SQuADv1.1
8	Liu vd. [51] / 2017	%84,0 F1 Skoru	Çok katmanlı dikkat modelleri mimarisi	SQuADv1.1
9	Pan vd. [52] / 2017	%82,66 F1 Skoru	Hafıza Ağı ile Çok Katmanlı Gömme (MEMEN) sinir ağı mimarisi	SQuADv1.1
10	Xiong vd. [53] / 2017	%78,9 EM ve %86 F1 Skoru	Derin kalıntı dikkat kodlayıcılı dinamik dikkat ağları (DCN)	SQuADv1.1
11	Huang vd. [54] / 2017	%85,9 F1 Skoru	FusionNet adlı yeni bir sinir yapısı	SQuADv1.1
12	Liu vd. [55] / 2017	%86,49 F1 Skoru	Çok adımlı bir sinir ağı modeli	SQuADv1.1

Çizelge 2.1. (devam) Literatürdeki ilgili çalışmaların bir karşılaştırması.

No	Yazar / Yıl	Metrikler	Model / Yapı	Veri Seti
13	Salant ve Berant [56] / 2017	%84 F1 Skoru	RNN tabanlı yeniden birleştirme	SQuADv1.1
14	Hu vd. [57] / 2018	%74,2 F1 Skoru	ELMo (Embeddings from Language Models)	SQuADv1.1 ve SQuADv2.0
15	Wang ve Jiang [58] / 2018	%83,5 F1 Skoru	Makine Okuduğunu Anlama (Machine Reading Comprehension - MRC) modeli	SQuADv1.1
16	Sun vd. [59] / 2018	%72,6 F1 Skoru	Birleşik kodlama, çok seviyeli dikkat, nihai füzyon ve tahmin	SQuAD2.0
17	Gupta vd. [35] / 2018	%80,30 Doğruluk Oranı	Soru sınıflandırma modeli CNN ve RNN ile	Web'den altı farklı alanda 500 makale
18	Zhou vd. [66] / 2018	Macro-F1 %58,77	Tekrarlayan konvolüsyonel sinir ağı (RCNN)	SemEval2015
19	Martinez-Gil vd. [67] / 2019	%65 Doğruluk Oranı	Güçlendirilmiş Çoktan Seçmeli Soru Cevaplama	Oxford Üniversitesi yayınlarındaki hukuki sorulardan oluşan bir veri kümesi
20	Su vd. [64] / 2019	%56,59 EM ve %68,98 F1 Skoru	Multi-task XLNet-large	SQuAD, NewsQA, TriviaQA, SearchQA, HotpotQA, ve NaturalQuestions
21	Zhang vd. [48] / 2019	%85,1 EM ve %87,9 F1 Skoru	BERT	SQuAD 2.0 ve ReAding Comprehension Dataset From Examinations (RACE)
22	Zhang vd. [49] / 2019	%83,6 F1 Skoru ve %80,9 EM	BERT	MRC benchmark dataset, SQuADv2.0

Çizelge 2.1. (devam) Literatürdeki ilgili çalışmaların bir karşılaştırması.

No	Yazar / Yıl	Metrikler	Model / Yapı	Veri Seti
23	Yoon vd. [63] / 2019	%82,5 F1 Skoru	BioBERT	SQuADv2.0
24	Beltagy vd. [65] / 2019	Ortalama Bert Ham %73,58; Bert Finetunig %77,16; SCIBERT Ham %76,01; SCIBERT Finetuning %79,27	BERT	BC5CDR, JNLPBA, NCBI-disease, EBM-NLP, GENIA, ChemProt, SciERC, SciERC, ACL-ARC Paper Field
25	Yeh ve Chen [69] / 2019	%88,6 F1 Skoru	BERT	Adversarial-SquAD
26	Carrino vd. [70] / 2019	MLQA derleminde %68,1 F1-skoru ve XQuAD derleminde %77,6 F1-skoru	BERT	SQuAD-es v1.1
27	Esposito vd. [68] / 2020	%0,90 MMR ve Doğruluk %0,81	Sözlüksel kaynaklara ve kelime gömülmelerine dayanan hibrit QE (Hybrid query expansion)	Sözcüksel Yanıt Türünden (LAT) oluşan bir veri kümesi
28	Möller vd. [71] / 2020	%59,53 F1 Skoru	RoBERTa	COVID-QA veri kümesi
29	Uğurlu vd. [37] / 2020	Tasarlanan akıllı asistan sorulara cevap verebilmiştir	Kelime örtüşüm (word overlap) yöntemi	Koronavirüs Bilim Kurulu açıklamalarından 54 soru cevap çifti ve Worldometer'da yayınlanan veriler
30	Chau vd. [39] / 2020	0,860 hassasiyet, 0,958 geri çağırma ve 0,906 F1 Skoru	BERT	Anayasa, kanun, tüzük, yönetmelik, emir, ortak karar, kararname, karar, genelge, ortak genelge ve yönerge yasal belgeler alanlarından yaklaşık 250 000 soru cevap çifti

Çizelge 2.1. (devam) Literatürdeki ilgili çalışmaların bir karşılaştırması.

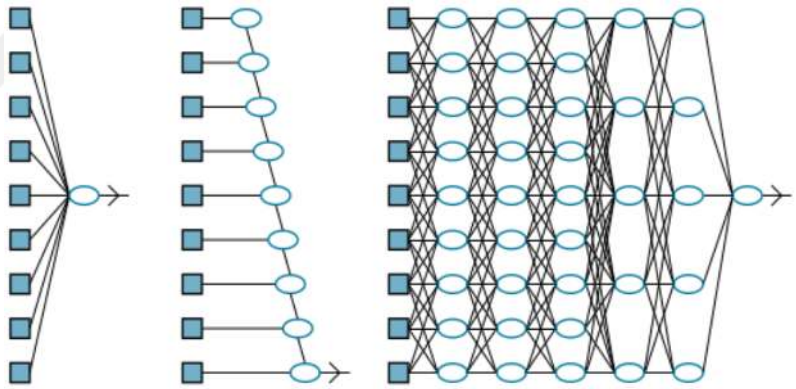
No	Yazar / Yıl	Metrikler	Model / Yapı	Veri Seti
31	Luo vd. [45] / 2020	%80,9 Doğruluk Oranı	BERT	SimpleQuestions, Freebase'deki 108 442 soru ve cevap.
32	Day ve Kuo [43] / 2020	%77,44 EM	BERT	Delta Okuduğunu Anlama Veri Kümesi (DRCD)
33	Chakraborty vd. [62] / 2020	%80,85 EM Ve %83,96 F1 Skoru	BERT nöral tabanlı derin bağlamsal model	BREATHE7 veri kümesi ve SQuADv2.0
34	Abadani vd. [72] / 2021	manuel için %56,66; otomatik için %70,84 F1 Skoru	BERT, ALBERT ve çok dilli BERT	Farsça Soru Cevaplama Veri Kümesi (ParSQuAD)
35	Chen vd. [33] / 2021	%75,4 EM ve %78,8 F1 Skoru	BERT ve ALBERT-base	SQuADv1.1 ve SQuADv2.0
36	Tieu vd. [34] / 2021	Görev 1 için %88,07 Makro F2, Görev 2 için %69,89 Doğruluk Oranı, Görev 3 için %71,02 Doğruluk Oranı	BERT	16 kanun ve 2279 kanun maddesi
37	Devanshi ve Rebecca [46] / 2021	0,8253 MMR	NLP örüntü eşleştirme ve bilgi alma yöntemleri	Text REtrieval Conference (TREC) QA
38	Nikita vd. [47] / 2021	Başarılı olmuştur	BERT	Conversational Question Answering (CoQA) dataset
39	Ram vd. [60] / 2021	%72,7 F1 Skoru	SpanBERT-base	SQuADv2.0
40	Rohit vd. [61] / 2021	İlk model eğitimin doğruluğu %64,93; test doğruluğu %60,33; İkinci model doğruluğu %70,02; testte %66,03	RNN bidirectional LSTM model, ikinci model Multinomial Logistic regression ve XGBoost	SQuAD
41	Chen J vd. [73] / 2023	Mediqu için doğruluk %71,91, PıbMedQA için doğruluk %58,2	BERT, RoBERTa, SpanBERT, BioBERT ve PubMedBERT	MEDIQA-2019 ve PubMedQA

Çizelge 2.1. (devam) Literatürdeki ilgili çalışmaların bir karşılaştırması.

No	Yazar / Yıl	Metrikler	Model / Yapı	Veri Seti
42	Arezoo Saedi vd. [74] / 2023	%86,00 Doğruluk Oranı (BERT), %86,20 Doğruluk Oranı (RoBERTa)	BERT, RoBERTa	Genetik ve Nadir Hastalıklar (GARD) üzerinde %86,00, Yahoo! Answers tabanlı veri seti
43	Liu vd. [77] / 2023	%91,2 Doğruluk Oranı	BERT	Ulusal Mineral Kaya ve Fosil Numune Kaynakları Altyapısı ve Acta Mineralogica Sinica verileri
44	Kazani vd. [78] / 2023	%75,61 F1 Skoru	BERT, ParsBERT ve ALBERT	FarsNewsQA, SQuAD
45	Wang vd. [79] / 2023	%78,1 Doğruluk Oranı	BERT	SimpleQuestions veri kümesi
46	Tohma vd. [80] / 2023	%91,05 Doğruluk Oranı	BERT	SCD Dataset, DS1 Dataset ve Sentimentset Dataset
47	Wang vd. [81] / 2023	%90 F1 Skoru	BERT	Bilgisayar Makineleri Derneği dijital kütüphanesi
48	Kim vd. [82] / 2023	%89,8 EM ve %95,6 F1 Skoru	RoBERTa, BERT	SQuADv1.1
49	Gupta vd. [75] / 2024	%98 Doğruluk Oranı	BERT	Fortune 500 Hint kuruluşları listesinden BT kuruluşlarını seçmiş ve sürdürülebilirlik raporları
50	Kotstein vd. [76] / 2024	%81,95 ve %88,44 Doğruluk Oranı	BERT	2 321 OpenAPI dokümantasyonundan alınan örnekler

3. DERİN ÖĞRENME

Derin öğrenme, NLP'de en yaygın kullanılan makine öğrenimi biçimidir. 1980'lerde araştırmacılar tarafından yapay sinir ağları geliştirilmiştir. Yapay sinir ağları çok sayıda temel makine öğrenimi modelini tek bir ağda birleştirir. Beyne benzer şekilde, basit makine öğrenimi modelleri nöron olarak adlandırılır. Bu nöronlar katmanlar halinde organize edilir. Derin bir sinir ağı birçok katman içerir. Derin öğrenme, derin sinir ağı modellerini kullanan bir makine öğrenimi tekniğidir [83]. En soldaki katman giriş katmanı, en sağdaki katman çıkış katmanı ve arasında kalan katmanlar ise gizli katmanlar olarak adlandırılmaktadır. Şekil 3.1'de sırasıyla sıg bir model, kısa yollara sahip bir model ve birbirleriyle etkileşimli birçok yola sahip Derin Sinir Ağları model yapılarına dair örnekler şematik olarak gösterilmiştir.



Şekil 3.1. Basitten karmaşığa sinir ağları modeli örnekleri ([84]).

Derin sinir ağları önemli miktarda veri ve bilgi işlem gücü gerektirir. Bu nedenle, derin sinir ağı kullanımı ile çalışırken bu faktörleri göz önünde bulundurmak önemlidir. Ancak transfer öğrenimi, önceden eğitilmiş bir derin sinir ağının çok daha az eğitim verisi ve bilgi işlem yükü ile yeni bir görevi yerine getirmek üzere daha fazla eğitilmesine olanak tanır. Transfer öğrenmenin en basit şekli ince ayar olarak bilinir. Bu süreç, modelin Wikipedia gibi büyük bir genel veri kümesi üzerinde eğitilmesini ve ardından ilgili NLP görevi için etiketlenmiş daha küçük bir göreve özgü veri kümesi üzerinde eğitilmesini içerir [85]. İnce ayar veri kümeleri, yalnızca birkaç yüz veya hatta onlarca eğitim örneğinden oluşan son derece küçük olabilir. Bu, ince ayar eğitiminin tek bir CPU'da birkaç dakika

içinde tamamlanmasına olanak tanır. Transfer öğrenimi, derin öğrenme modellerinin farklı platformlarda ve farklı görevler için kullanılmasını kolaylaştırmıştır [86].

Son zamanlarda, çeşitli diller ve veri kümeleri için ön eğitim kombinasyonlarıyla eğitilmiş derin öğrenme modelleri sunan sağlayıcılar çoğalmaktadır. Bu önceden eğitilmiş modeller indirilebilir ve çok çeşitli hedef görevler için ince ayar yapılabilir [86].

Tez çalışma kapsamında THQuADv2.0 veri kümesi üzerinde soru cevaplama için önerilen derin öğrenme modelinde ön eğitimli BERT metin temsili kullanılmıştır. Ön eğitimli metin temsilleriyle oluşturulan modele ait deneyler sonuçlar başlığında anlatılmıştır.

3.1. DERİN ÖĞRENME MODELLERİ

Derin öğrenme modelleri, yapay sinir ağları gibi karmaşık yapılara sahip makine öğrenme algoritmalarıdır ve eğitim için genellikle büyük miktarlarda veri gerektirir. Bu modeller, veri modellerini otomatik olarak öğrenerek karmaşık görevleri gerçekleştirebilir. Derin öğrenme özellikle resim, ses veya metin gibi yüksek boyutlu ve karmaşık veri türleri için etkilidir.

Temel olarak derin öğrenme modelleri, veritabanlarında gizli kalıpları ve ilişkileri bulmak için matematiksel işlemleri kullanır. Bu modeller tipik olarak her biri daha yüksek düzeyde bir temsil oluşturmak için giriş verilerini işleyen birbirine bağlı katmanlardan oluşur. Derin öğrenme modelleri, genellikle yapay sinir ağları olarak adlandırılan büyük ve karmaşık ağlardır.

Derin öğrenme modelleri, sınıflandırma, regresyon, görüntü tanıma ve dil işleme gibi karmaşık sorunları çözmek için özel olarak tasarlanmıştır. Bu modeller verilere dayalı olarak öğrenir; bu da, daha fazla veriyle eğitildikçe performanslarının genellikle arttığı anlamına gelir.

3.1.1. Konvolüsyonel Sinir Ağları

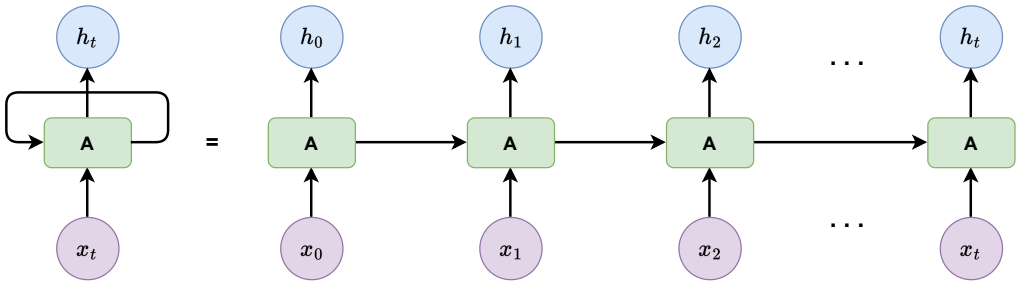
Konvolüsyonel Sinir Ağları (Convolutional Neural Networks - CNN), esas olarak görüntü işleme ve bilgisayarlı görme alanlarında kullanılan bir tür sinir ağıdır [87]. CNN'ler görüntülerdeki özelliklerin düzeyini öğrenmek için sinir ağlarını kullanır [88]. Bu

katmanlar, bir görüntünün küçük bölümlerini tarayarak, köşe panelleri gibi düşük seviyeli özelliklerden daha karmaşık özelliklere kadar derin öğrenme oluşturarak özellik haritaları oluşturur [89]. Bir CNN mimarisi tipik olarak doğrulama birimlerinden, işlem birimlerinden, kümelemeden ve entegrasyondan oluşur [90]. Doğrulama katmanı, filtreleme yoluyla giriş verilerinin belirli kısımlarını kaldırarak ağın görüntü tanıma, sınıflandırma ve nesne algılama gibi görevlerde yüksek performans elde etmesine olanak tanır [91].

3.1.2. Tekrarlayan Sinir Ağı

Tekrarlayan Sinir Ağı (Recurrent Neural Network - RNN) sıralı verileri işlerken kullanılır. Her bir kelime sırayla RNN'e verilir ve çıktı olarak bir vektör elde edilir. Elde edilen bu vektör ile sonraki kelimenin vektörü tekrar RNN'e girdi olarak verilir. Bu işlem cümle sonuna kadar devam eder. Sadece önceki kelimelere baktıkları için, RNN'ler bağlam oluşturmada iyi sonuçlar vermezler. Ayrıca RNN sinir ağı eğitilirken donanımı çok optimize kullanamazlar. Bu problemlere çözüm bulan Dönüştürücü dil modelleri geliştirilmiştir [92].

RNN'lerde kullanılan Kodlayıcı-Kod Çözücü (Encoder-Decoder) yapılarının gelişmesiyle, makine dili çevirisi, metinlerin sınıflandırılması gibi NLP görevlerinde alınan sonuçlar yeni bir boyut kazanmıştır [66]. Kodlayıcı ve Kod Çözücü, birer RNN yapısıdır. Bu yapıda öncelikle kelime gömme adı verilen metin temsilleri oluşturulmaktadır. Kodlayıcı bölümünde RNN eğitilir ve bu eğitimin çıktısı olan gizli katman, kod çözücüye aktarılmaktadır. Kod çözücüden üretilen çıktı, probleme göre değişkenlik gösterebilmektedir. Şekil 3.2'de mimarisi gösterilmektedir [93].



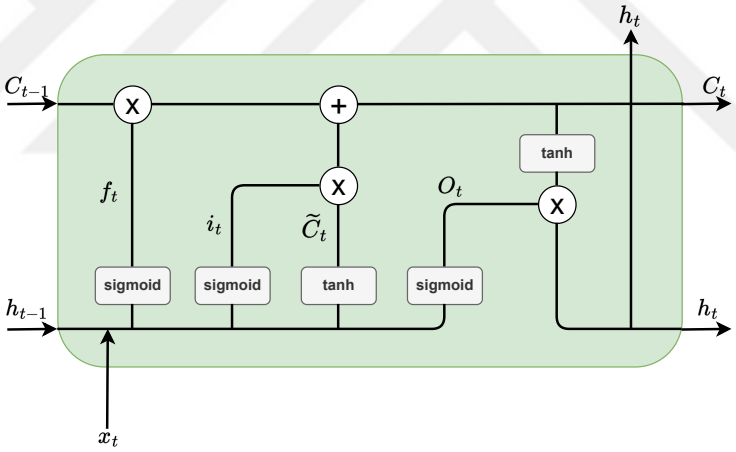
Şekil 3.2. RNN mimarisi ([93]).

3.1.3. Uzun Kısa Süreli Bellek

Günümüzde, 1997 yılında Hochreiter ve Schmidhuber [94]'in tanıttıgı, çok derin yapılardan kaçınmak için çok fazla yinelemeyi engelleyen kapıların bulunduğu LSTM adı verilen yinelemeli yapı yaygın bir şekilde kullanılmaktadır. Özellikle zaman serisi verileri ve dogal dil işleme gibi sıralı verileri modellemek için geliştirilmiştir.

RNN, uzun vadeli bağımlılıklar nedeniyle eğitim sırasında gradyanların patlaması veya kaybolması sorunları yaşayabilir. Bu sebeple LSTM, RNN'nın uzun süreli bağımlılıkları ögrenmedeki zorluklarını bertaraf etmek için tasarlanmıştır [95].

Sekil 3.3'te hücre hafızası, giriş unutma ve çıkış kapıları içeren bir LSTM biriminin yapısı verilmiştir. Bu birimlerin oluşturdugu LSTM ağları ile eğitim sürecinde önemli bilgiler uzun vadeli olarak saklanmakta, gereksiz bilgiler ise kapı mekanizmaları ile ihmal edilmektedir [96].

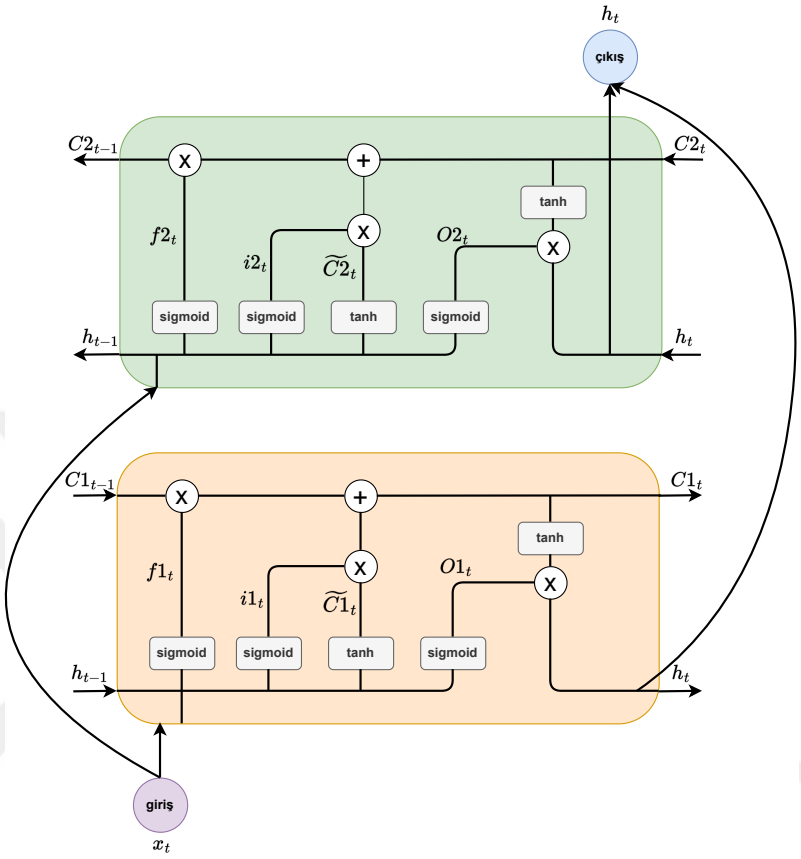


Şekil 3.3. LSTM mimarisi ([97]).

3.1.4. Çift Yönlü Uzun Kısa Süreli Bellek

Çift Yönlü Uzun Kısa Süreli Bellek (Bidirectional Long Short Term Memory - BiLSTM) [98], [99], ardışık verilerin analizi için kullanılan bir yapay sinir ağı mimarisidir. Bu mimari, geleneksel LSTM'nin ötesine geçer ve veriyi hem ileri hem de geri yönde işleyen iki paralel LSTM ağını içerir. BiLSTM, bu çift yönlü işlemle hem geçmiş hem de gelecek bağlamını aynı anda yakalar. Bu çift yönlü işleme sayesinde, BiLSTM, ardışık verilerin içerdiği uzun vadeli bağımlılıkları daha iyi modelleyebilir. Hem geçmiş hem de gelecek bilgisini dikkate alarak, zaman içindeki ilişkileri daha iyi anlama yeteneğine sahiptir.

Bu özellik, Doğal Dil İşleme, konuşma tanıma, zaman serisi analizi ve benzeri birçok uygulamada daha iyi performans elde etmek için kullanılır. Şekil 3.4'te BiLSTM'nin mimarisini göstermektedir.



Şekil 3.4. BiLSTM mimarisi ([100]).

3.2. AKTİVASYON FONKSİYONLARI

Her bir nöron yapısı girişler toplamını bir aktivasyon fonksiyonundan geçirerek oluşan çıktı bilgisini çok katmanlı yapıda etkileşimde bulundugu diğer nöronlara iletir. Kullanılan aktivasyon fonksiyonu oluşan bilgiyi karakteristigine göre filtreleyerek öğrenme işlemine katkıda bulunur. Bu özelliğiyle aktivasyon fonksiyonu nöronun davranışını belirleyen en önemli parçalardan biridir. Aşagıda ANN en çok kullanılan aktivasyon fonksiyonlarına değinilerek karakterleri hakkında bilgi verilmiştir [101], [102].

Bu aktivasyon fonksiyonları, derin öğrenme modellerinin performansını optimize etmek için çeşitli durumlarda kullanılabilir. Hangi fonksiyonun en uygun olduğunu belirlemek, modelin yapısı, veri seti ve çözülmesi gereken problem gibi faktörlere bağlıdır.

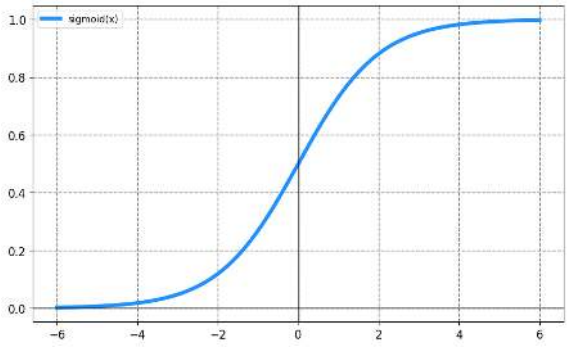
Çıktı katmanlarında kullanacağımız fonksiyonu seçmek gizli katmana göre biraz daha kolay çünkü yapacağımız tahmine göre uygun aktivasyon fonksiyonumuzu kullanıyoruz. Ara katmanlarda ise kullanacağımız fonksiyonu deneme yanılma yöntemi ile seçiyoruz. Modelimizi farklı aktivasyon fonksiyonlarını kullanarak eğitip en iyi performans aldığımız Softmax aktivasyon fonksiyonu ve ReLU aktivasyon fonksiyonu bu çalışmada kullanılmıştır.

3.2.1. Sigmoid Aktivasyon Fonksiyonu

Sigmoid aktivasyon fonksiyonu doğrusal olmayan karmaşık yapıları öğrenme yetisi ve her zaman (0,1) aralığında sonuç üretmesi sebebiyle de ikili olasılıksal sınıflandırma problemlerinde avantaj sağlaması, bu aktivasyon fonksiyonunu öğrenme sürecinde gizli katmanlarda en sık kullanılan aktivasyon fonksiyonularından biri haline getirmiştir ve Denklem (3.1) ile hesaplanır [101], [103].

$$S(x) = \frac{1}{(1+e^{-x})} \tag{3.1}$$

Şekil 3.5'teki sigmoid fonksiyonu grafiğinde $(-\infty, +\infty)$ değerlerine yaklaştıkça x değerlerinin değişimin y değerlerini çok küçük oranlarda etkilediği görülmektedir. Bu durum kaybolan gradyanlar (vanishing gradient) sorununu doğurarak öğrenme sürecini negatif anlamda etkileyebilmektedir.



Şekil 3.5. Sigmoid aktivasyon fonksiyonu.

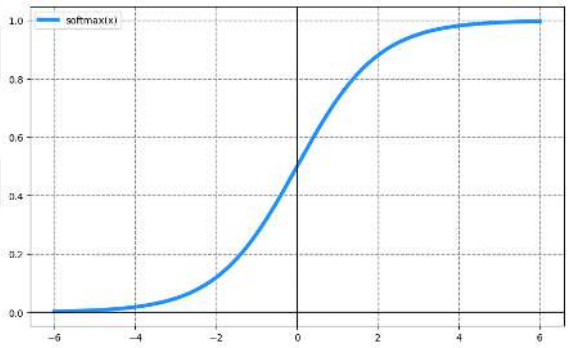
3.2.2. Softmax Aktivasyon Fonksiyonu

Softmax aktivasyon fonksiyonu, özellikle çok sınıflı sınıflandırma problemlerinde kullanılan ve her sınıf için olasılık dağılımı sağlayan bir fonksiyondur. Softmax

Aktivasyon Fonksiyonu sigmoid fonksiyonuna çok benzemektedir. Sigmoid aktivasyon fonksiyonundan farklı olarak özellikle derin öğrenme modellerinin çıkış katmanında tercih edilmektedir. Aynı şekilde (0,1) aralığında değerler üretmesi o sınıfa aitliğe dair olasılıksal bir yorum çıkarımını sağlamaktadır. Denklem (3.2) ile hesaplanır [101].

$$\sigma(z)_i = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}} \tag{3.2}$$

Softmax aktivasyon fonksiyonu, tüm sınıflar için olasılıkların toplamının 1 olmasını sağlar. Bu özellik, çıktıların kolayca karşılaştırılabilmesini ve yorumlanabilmesini sağlar. Softmax fonksiyonu, özellikle sinir ağlarının çıktı katmanında, her sınıfa ait tahmin edilen olasılıkları elde etmek için kullanılır. Şekil 3.6'da Softmax aktivasyon fonksiyonu grafiği göstermektedir.



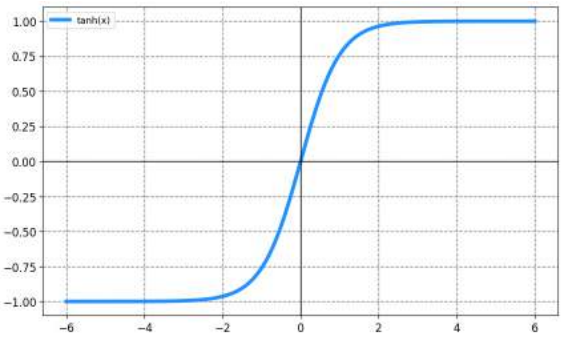
Şekil 3.6. Softmax aktivasyon fonksiyonu.

3.2.3. Hiperbolik Tanjant Aktivasyon Fonksiyonu

Hiperbolik Tanjant (Hyperbolic Tangent - tanh) aktivasyon fonksiyonu sigmoid fonksiyonuna benzemesinin yanında sigmoid fonksiyonundan farklı olarak (-1,1) aralığında değer üretir. Geriye yayılım (backpropagation) değeri daha yüksek olduğundan daha hızlı öğrenme sağlar fakat aynı sigmoid aktivasyon fonksiyonunda olduğu gibi kaybolan gradyanlar sorunu bu fonksiyonda da bulunmaktadır ve Denklem (3.3) ile hesaplanır [101].

$$tanh(x) = \frac{e^{x} - e^{-x}}{e^{x} + e^{-x}} \tag{3.3}$$

Şekil 3.7'de Hiperbolik tanjant fonksiyonu, sigmoid fonksiyonuna benzer, ancak çıktı aralığı -1 ile 1 arasındadır. Bu özellik, özellikle sinir ağlarında kullanıldığında, sıfır merkezli çıktılar sağladığı için bazı öğrenme algoritmalarının daha hızlı ve etkili olmasını sağlar.

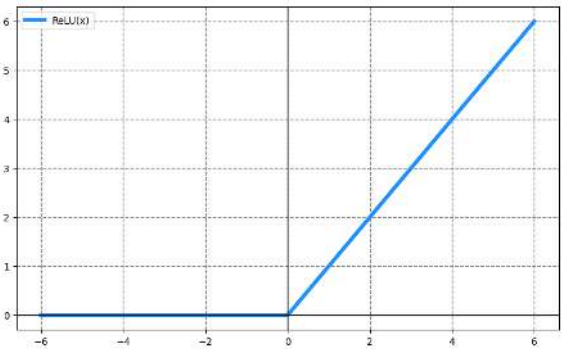


Şekil 3.7. Hiperbolik Tanjant aktivasyon fonksiyonu.

3.2.4. ReLU Aktivasyon Fonksiyonu

Şekil 3.8'de ReLU (Rectified Linear Unit) Aktivasyon Fonksiyonu $[0, +\infty)$ aralığında değerler üretmektedir. Sigmoid ve hiperbolik tanjant aktivasyon fonksiyonları ile karşılaştırıldığında hesaplama maliyetinin daha az olması günümüzde, özellikle de derin öğrenme modellerinde, oldukça yaygın olarak kullanılmasını sağlamıştır [103]. Denklem (3.4) ile hesaplanır [103]. Şekil 3.8'de ReLU aktivasyon fonksiyonu grafiği gösterilmektedir.

$$ReLU(x) = max(0, x) \tag{3.4}$$

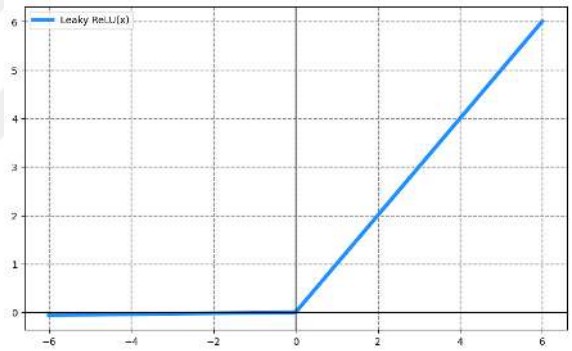


Şekil 3.8. ReLU aktivasyon fonksiyonu.

3.2.5. Sızıntı ReLU Aktivasyon Fonksiyonu

Sızıntı (Leaky) ReLU aktivasyon fonksiyonu, ReLU'nun bir varyantıdır ve negatif değerler için küçük bir eğim tanımlar. Bu sayede, nöronlar tamamen kapanmaz. ReLU 'nun varyantı olduğu için onun sahip olduğu tüm avantajlara sahiptir. Denklem (3.5)'te α genellikle küçük bir pozitif sayı, örneğin 0.01'dir. Dying ReLU problemini hafifletir. Negatif değerlerde küçük bir eğim sağlayarak nöronların tamamen kapanmasını önler. Bundan dolayı ölü nöron sonunu ortadan kaldırmak için geliştirilmiş bir fonksiyondur diyebiliriz. Denklem (3.5) ile hesaplanır [104]. Şekil 3.9'da Leaky ReLU aktivasyon fonksiyonu grafiği gösterilmektedir.

$$\text{Leaky ReLU}(x) = \begin{cases} x & \text{if } x \geq 0 \\ \alpha x & \text{if } x < 0 \end{cases} \tag{3.5}$$

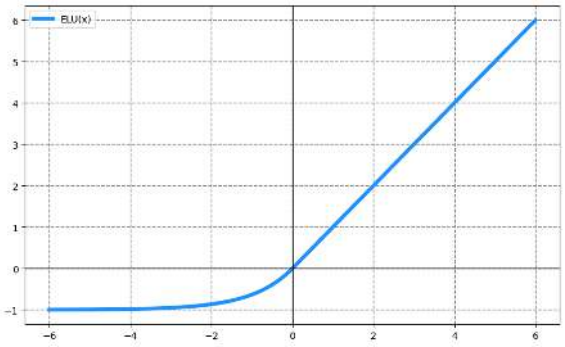


Şekil 3.9. Leaky ReLU aktivasyon fonksiyonu.

3.2.6. Üstel Doğrusal Birim Aktivasyon Fonksiyonu

Üstel Doğrusal Birim (Exponential Linear Unit - ELU) aktivasyon fonksiyonu, ReLU'nun negatif değerlerdeki yavaşlığını çözmek için geliştirilmiş bir aktivasyon fonksiyonudur. Negatif girdilerde yavaş azalır. Denklem (3.5)'te α genellikle 1 olarak alınır. Negatif değerler için yavaş azalır, bu da negatif girdilerde bile bilgi akışını sağlar. Ortalama aktivasyon değerlerini 0'a yaklaştırarak öğrenmeyi hızlandırır ve ağırlıkların dengelenmesini sağlar. Sigmoid ve tanh'a göre gradyanların vanishing problemine daha az eğilimlidir. Hesaplama karmaşıklığı ReLU'ya göre daha fazladır, bu da bazı durumlarda hesaplama maliyetini artırabilir. Denklem (3.6) ile hesaplanır [105]. Şekil 3.10'da ELU aktivasyon fonksiyonu grafiği gösterilmektedir.

$$\text{ELU}(x) = \begin{cases} x & \text{if } x \geq 0 \\ \alpha\left(e^{x} - 1\right) & \text{if } x < 0 \end{cases} \tag{3.6}$$

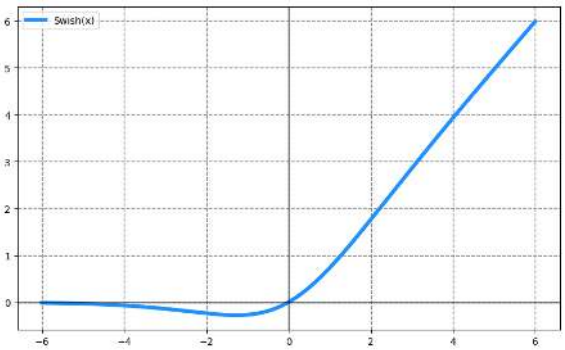


Şekil 3.10. ELU aktivasyon fonksiyonu.

3.2.7. Swish Aktivasyon Fonksiyonu

Swish, sigmoid fonksiyonunun kendisi ile çarpımı olarak tanımlanır. Bu, sigmoid fonksiyonunun yumuşaklık özelliklerini ve ReLU'nun doğrusal olmayan avantajlarını birleştirir. Bu da daha derin ağlarda daha iyi performans ve daha hızlı öğrenme sağlar. Parametrik olmayan yapısı nedeniyle, belirli durumlarda özel ayarlama gerektirebilir. Denklem (3.7) ile hesaplanır [106]. Şekil 3.11'de Swish aktivasyon fonksiyonu grafiği gösterilmektedir.

$$\text{Swish}(x) = x \cdot \sigma\left(x\right) = \frac{x}{1 + e^{-x}} \tag{3.7}$$



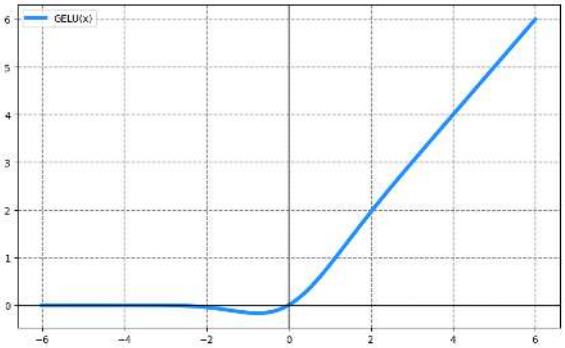
Şekil 3.11. Swish aktivasyon fonksiyonu.

3.2.8. Gauss Hatası Doğrusal Birim Aktivasyon Fonksiyonu

Gauss Hatası Doğrusal Birim (Gaussian Error Linear Unit - GELU), giriş değerini Gaussian dağılım kullanarak hesaplayan bir aktivasyon fonksiyonudur. GELU, özellikle Transformer tabanlı modellerde ve BERT gibi doğal dil işleme modellerinde yaygın olarak kullanılır. Bu fonksiyon, belirli bir giriş değeri için belirli bir Gaussian dağılımı altında aktivasyon olup olmayacağına karar verir. Denklem (3.8)'de $\phi(x)$, standart normal dağılımın kümülatif dağılım fonksiyonudur ve genellikle Denklem (3.9)'a göre hesaplanır. Non-lineer aktivasyon sağlar, bu da derin öğrenme modellerinin daha karmaşık ilişkileri öğrenmesini sağlar. Yumuşak geçişler sayesinde vanishing gradient problemini minimize eder. Derin dil modellerinde performans artışı sağlar. Hesaplaması daha karmaşıktır ve bu da eğitim süresini artırabilir [107]. Şekil 3.12'de GELU aktivasyon fonksiyonu grafiği gösterilmektedir.

$$\text{GELU}(x) = x \cdot \phi(x) \tag{3.8}$$

$$\phi(x) = 0.5\left(1 + erf\left(\frac{x}{\sqrt{2}}\right)\right) \tag{3.9}$$



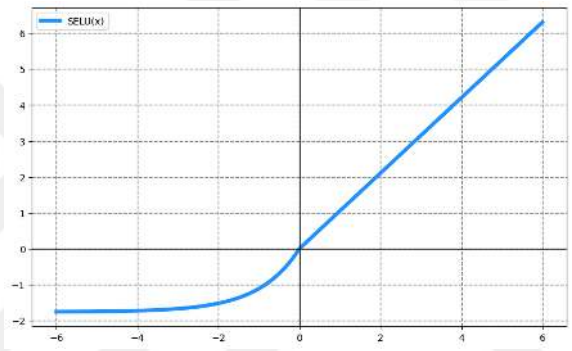
Şekil 3.12. GELU aktivasyon fonksiyonu.

3.2.9. Ölçeklendirilmiş Üstel Doğrusal Birim Aktivasyon Fonksiyonu

Ölçeklendirilmiş Üstel Doğrusal Birim (Scaled Exponential Linear Unit - SELU), ELU'nun bir varyantıdır ve ağın kendi kendine normalleşmesini sağlayan bir ölçeklendirme faktörü içerir. Bu ölçeklendirme, ağırlıkların ve aktivasyonların normalizasyonunu sağlar ve ağın

derinliğinin artmasına olanak tanır. Denklem (3.10)'da λ ve α, sabitlerdir. Genellikle $\lambda = 1.0507$ ve $\alpha = 1.67326$ olarak alınır. Otomatik normalizasyon sağlar, bu da derin ağlarda daha hızlı ve stabil öğrenme sağlar.Derin ağların daha etkili şekilde öğrenmesini ve genelleştirme kapasitesini artırır. Hesaplama olarak biraz daha karmaşıktır. Spesifik parametre değerlerine bağlı olduğu için belirli durumlarda ayarlama gerektirebilir [108]. Şekil 3.13'te SELU aktivasyon fonksiyonu grafiği gösterilmektedir.

$$\text{SELU}(x) = \lambda \begin{cases} x & \text{if } x \geq 0 \\ \alpha\,(e^x - 1) & \text{if } x < 0 \end{cases} \tag{3.10}$$

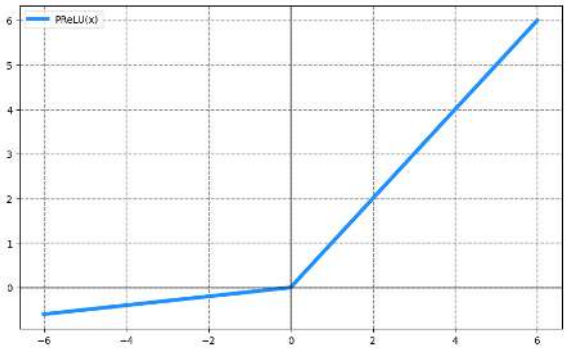


Şekil 3.13. SELU aktivasyon fonksiyonu.

3.2.10. Parametrik ReLU Aktivasyon Fonksiyonu

Parametrik ReLU (Parametric ReLU - PReLU), Leaky ReLU'nun bir varyantıdır ve negatif girişler için eğim parametresi öğrenilir. Bu, modelin belirli görevler için optimize edilmesine olanak tanır. Denklem (3.11)'de $\alpha = 1.67326$ öğrenilen bir parametredir. Negatif değerlerdeki eğim parametresi öğrenilir, bu da daha esnek ve adaptif bir aktivasyon fonksiyonu sağlar. Dying ReLU problemini azaltır. Performansı artırabilir ve bazı görevlerde ReLU'dan daha iyi sonuçlar verebilir. Ekstra parametrelerin öğrenilmesi gerektiği için hesaplama maliyeti artar. Çok büyük negatif girişlerde hala bazı öğrenme zorlukları yaşanabilir [109]. Şekil 3.14'te PReLU aktivasyon fonksiyonu grafiği gösterilmektedir.

$$\text{SELU}(x) = \lambda \begin{cases} x & \text{if } x \geq 0 \\ \alpha\,(e^x - 1) & \text{if } x < 0 \end{cases} \tag{3.11}$$



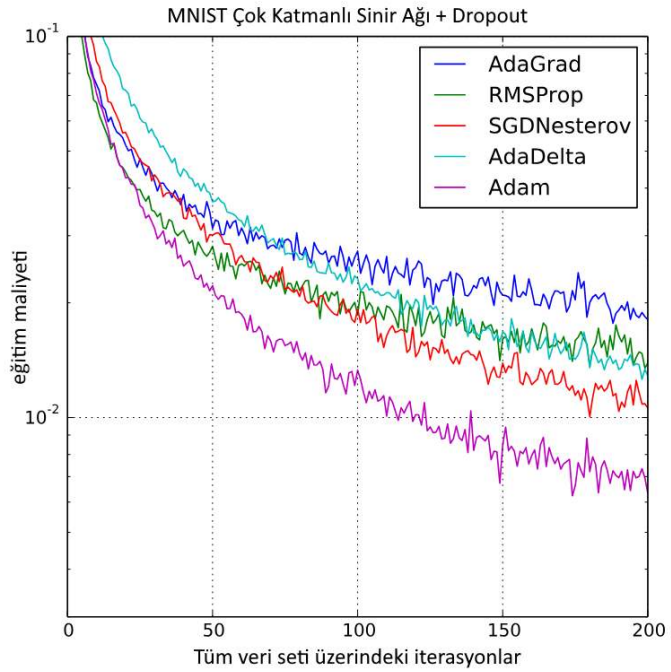
Şekil 3.14. PReLU aktivasyon fonksiyonu.

3.3. OPTİMİZASYON YÖNTEMLERİ

Optimizasyon, çeşitli değişkenlere bağlı olarak değişen bir amaç fonksiyonunu verilen kısıtlayıcı şartlar altında en optimum (en küçük veya en büyük) yapan değerlerin bulunması olarak tanımlanır [110]. Özellikle mühendislik problemleri başta olmak üzere, işletme, ekonomi ve tasarım problemlerinde optimizasyona ihtiyaç duyulur. Derin ağlarda optimizasyon algoritmaları ise gerçek değerler ve parametreler ile o an bulunan değerler arasındaki hatayı en küçük yapacak şekilde parametreleri güncelleyerek hatayı minimize etmeyi amaçlar. Bu algoritmalardan bazıları şu şekilde listelenebilir:

- Adagrad
- Adadelta
- Stokastik Gradyon İnişi
- Rprop ve RMSprop
- Adam ve AdamW
- Adamax
- Nadam
- Adafactor
- Lion

Optimizasyon algoritmalarının karşılaştırılması Şekil 3.15'te verilmiştir.



Şekil 3.15. Optimizasyon algoritmaları çok katmanlı algılayıcı eğitiminin karşılaştırılması ([111]).

3.3.1. Adagrad Optimizasyon Algoritması

Adagrad, her parametre için ayrı ayrı öğrenme hızları tutarak daha nadir rastlanan özellikler için daha büyük öğrenme hızları ve daha sık rastlanan özellikler için daha küçük öğrenme hızları kullanır. Bu algoritma, özellikle seyrek veriye sahip durumlarda çok etkilidir çünkü nadir rastlanan özelliklerin öğrenme hızını artırarak daha hızlı öğrenmelerini sağlar. Denklem (3.12) ile hesaplanır. NLP'de seyrek kelime vektörleri ve özellikler üzerinde çalışırken Adagrad'in adaptif öğrenme hızı büyük avantaj sağlar [112].

$$\theta_{t+1} = \theta_t - \frac{\eta}{\sqrt{G_t + \varepsilon}} \odot g_t \tag{3.12}$$

- θ parametreleri temsil eder.
- η öğrenme hızıdır.
- g_t t zamanında gradyanı ifade eder.
- G_t her zaman adımında gradyanların karesinin kümülatif toplamıdır.
- ε sayısal kararlılık için küçük bir sayıdır.

3.3.2. Adadelta Optimizasyon Algoritması

Adadelta, Adagrad'ın bir geliştirmesidir. Adagrad, her parametre için öğrenme oranını, o parametre için hesaplanan tüm gradyanların karelerinin toplamının tersine orantılı olarak ayarlayarak çalışır. Bu, sık güncellenen parametrelerin daha az, nadiren güncellenen parametrelerin ise daha fazla öğrenmesini sağlar. Ancak, bu süreçte öğrenme oranları zamanla çok küçülerek modelin öğrenmesini durdurabilir [113].

Adadelta, bu sorunu çözmek için geçmiş kare gradyanların birikimini sınırlı bir pencere içinde tutar ve bu birikimin ortalamasını kullanır. Bu, geçmişteki tüm kare gradyanları biriktirmek yerine, belirli bir pencere içindeki ortalamayı alarak daha dengeli bir öğrenme oranı sağlar. Denklem (3.13) ile hesaplanır.

$$\begin{aligned} E[g^2]_t &= \rho E[g^2]_{t-1} + (1-\rho) g_t^2 \\ \Delta\theta_t &= -\frac{\sqrt{E[\Delta\theta^2]_{t-1}+\varepsilon}}{\sqrt{E[g^2]_t+\varepsilon}} g_t \\ \theta_{t+1} &= \theta_t + \Delta\theta_t \end{aligned} \tag{3.13}$$

3.3.3. Stokastik Gradyon İnişi (SGD) Optimizasyon Algoritması

SGD, klasik gradyan inişi algoritmasının bir varyasyonudur ve her iterasyonda tüm veri kümesi yerine rastgele seçilen bir veri örneği üzerinden gradyan hesaplaması yapar. Bu, hesaplama maliyetini azaltırken, daha gürültülü bir gradyan güncellemesi sağlar. Gürültü, modele daha iyi genelleme yeteneği kazandırabilir. Denklem (3.14) ile hesaplanır. Derin öğrenmede büyük veri setlerinde ve derin sinir ağlarının eğitiminde yaygın olarak kullanılır [114].

$$\theta_{t+1} = \theta_t - \eta \cdot g_t \tag{3.14}$$

- θ parametreleri temsil eder.
- η öğrenme hızıdır.
- g_t t zamanında rastgele seçilen bir veri örneği üzerinden hesaplanan gradyandır.

3.3.4. Rprop Optimizasyon Algoritması

Rprop, gradyanların yalnızca işaretini kullanarak, adım büyüklüklerini her parametre için adaptif olarak belirler. Bu sayede, gradyan büyüklüğünün etkisi ortadan kaldırılır ve yalnızca yön bilgisi kullanılır. Bu, ağın daha kararlı ve hızlı bir şekilde öğrenmesini sağlar. Denklem (3.15) ile hesaplanır. Özellikle derin sinir ağlarının eğitiminde yaygın olarak kullanılır [115].

$$\Delta_{ij}^{(t)} = \begin{cases} \eta^{+}\Delta_{ij}^{(t-1)} & \text{if } \frac{\partial E}{\partial w_{ij}}^{(t-1)} \cdot \frac{\partial E}{\partial w_{ij}}^{(t)} > 0 \\ \eta^{-}\Delta_{ij}^{(t-1)} & \text{if } \frac{\partial E}{\partial w_{ij}}^{(t-1)} \cdot \frac{\partial E}{\partial w_{ij}}^{(t)} < 0 \\ \Delta_{ij}^{(t-1)} & \text{else} \frac{\partial E}{\partial w_{ij}}^{(t-1)} \cdot \frac{\partial E}{\partial w_{ij}}^{(t)} = 0 \end{cases} \tag{3.15}$$

- $\Delta_{ij}^{(t)}$ adım büyüklüğünü ifade eder.
- η^{+} ve η^{-} adım büyüklüğünün artış ve azalış oranlarıdır.
- $\frac{\partial E}{\partial w_{ij}}$ ağırlık matrisinin gradyanıdır.

3.3.5. RMSprop Optimizasyon Algoritması

RMSprop (Root Mean Square Propagation), Adagrad algoritmasını geliştirmek amacıyla ortaya çıkmıştır. Her parametre için ayrı bir hareketli ortalama kullanarak, öğrenme hızını dinamik olarak ayarlar. Bu, Adagrad'ın dezavantajı olan öğrenme hızının çok küçülmesi sorununu çözmeyi amaçlar. Denklem (3.16) ile hesaplanır. Sinir ağlarının eğitiminde ve pekiştirmeli öğrenme problemlerinde yaygın olarak kullanılır [116].

$$\theta_{t+1} = \theta_t - \frac{\eta}{\sqrt{E[g^2]_t + \varepsilon}} \cdot g_t \tag{3.16}$$

- $E[g^2]_t$ gradyanların karesinin üstel hareketli ortalamasıdır.
- η öğrenme hızıdır.
- g_t t zamanında gradyandır.
- ε sayısal kararlılık için küçük bir sayıdır.

3.3.6. Adam ve AdamW Optimizasyon Algoritması

İlk olarak 2014 yılında Kingma ve Ba tarafından ortaya atılan Adam (Adaptive Moment Estimation) [111] optimizasyon algoritmasında popüler iki optimizasyon yönteminin avantajları birleştirilmiştir. Adagrad'ın seyrek gradyanlarla başa çıkma yeteneği ve RMSProp'un sabit olmayan hedeflerle başa çıkma yeteneği birleştirilerek algoritmanın hızlı, uygulaması kolay ve makine öğrenimi alanındaki çok çeşitli dışbükey olmayan optimizasyon problemlerine uygun olduğu belirtilmiştir. Adam optimizasyon algoritması, eğitim verilerine dayalı ağ ağırlıklarını güncellemek için klasik stokastik gradyan iniş yerine kullanılabilen bir yöntemdir. Ağırlıklar güncellenirken aşağıdaki parametreler bazı varsayılan değerleri ile kullanılır. Adam optimizasyon algoritması Denklem (3.17) ile hesaplanır.

$$
\begin{aligned}
m_t &= \beta_1 m_{t-1} + (1-\beta_1) g_t \\
v_t &= \beta_2 v_{t-1} + (1-\beta_2) g_t^2 \\
\hat{m}_t &= \tfrac{m_t}{1-\beta_1^t} \\
\hat{v}_t &= \tfrac{v_t}{1-\beta_2^t} \\
\theta_{t+1} &= \theta_t - \tfrac{\eta \hat{m}_t}{\sqrt{\hat{v}_t}+\varepsilon}
\end{aligned}
\tag{3.17}
$$

- m_t birinci momentin (ortalamanın) hareketli ortalamasıdır.
- v_t ikinci momentin (varyansın) hareketli ortalamasıdır.
- β_1 ve β_2 hareketli ortalama katsayılarıdır.
- η öğrenme hızıdır.
- ε sayısal kararlılık için küçük bir sayıdır.

Adam algoritmasına ek olarak, ağırlık çürümesi (weight decay) eklenir. Böylelikle AdamW (Adaptive Moment Estimation and Weight Decay) optimizasyon algoritması Denklem (3.18) ile hesaplanır [117].

$$
\begin{aligned}
m_t &= \beta_1 m_{t-1} + (1-\beta_1) g_t \\
v_t &= \beta_2 v_{t-1} + (1-\beta_2) g_t^2 \\
\hat{m}_t &= \tfrac{m_t}{1-\beta_1^t} \\
\hat{v}_t &= \tfrac{v_t}{1-\beta_2^t} \\
\theta_{t+1} &= \theta_t - \left(\tfrac{\eta \hat{m}_t}{\sqrt{\hat{v}_t}+\varepsilon} + \lambda \theta_t \right)
\end{aligned}
\tag{3.18}
$$

- λ ağırlık çürümesi katsayısıdır.

Adam algoritması çok çeşitli sinir ağı modellerinin eğitiminde yaygın olarak kullanılır. Önceden eğitilmiş modellerin yeniden eğitilmesi sürecinde etkilidir. Pekiştirmeli öğrenme problemlerinde kullanımı yaygındır. NLP'de büyük ve karmaşık modellerin eğitiminde kullanılır. Görüntü işleme ve nesne tanıma gibi görevlerde yaygın olarak kullanılır.

AdamW optimizasyon algoritması [117], Adam optimizasyon algoritmasının bir genişlemesi olarak ortaya çıkmıştır. Adam, derin öğrenme modellerinde sıkça kullanılan bir optimizasyon algoritmasıdır. Özellikle büyük ve karmaşık modellerin eğitiminde etkili olan bu algoritmalar, adaptif öğrenme hızları ve moment tahminleri sayesinde hızlı ve kararlı bir öğrenme süreci sağlarlar. AdamW, ek olarak ağırlık çürümesi ile modelin genelleme yeteneğini artırarak aşırı uyumu önlemeye yardımcı olur.

AdamW'nin temel fikri, ağırlık bozulmasını (weight decay) daha etkin bir şekilde ele almak ve modelin genel performansını artırmaktır. Ağırlık bozulması, modelin karmaşıklığını kontrol etmek ve aşırı uyumu önlemek için kullanılan bir regülerleştirme tekniğidir. Ancak, Adam optimizasyonunda, ağırlık bozulması doğrudan ağırlıklara eklendiğinden, momentumun ağırlıkları güncellemedeki etkisini azaltabiliyordu. Bu durum, modelin performansını olumsuz etkileyebiliyordu.

AdamW, bu sorunu çözmek için ağırlık bozulmasını daha doğru bir şekilde uygular. Özellikle, ağırlık bozulması terimi sadece modelin ağırlıklarını güncellerken eklenir, momentum kısmını etkilemez. Böylece, ağırlık bozulmasının etkisi daha tutarlı hale gelir ve modelin genel performansı iyileştirilir.

Adam'ın AdamW'ye göre avantajları [117]:

- Adam, daha yaygın kullanıldığı için literatürde ve pratikte geniş bir kabul görmüştür.
- Eğitim sırasında hiperparametre ayarlaması daha az karmaşıktır.
- Adam, adaptif öğrenme hızı ve momentum kombinasyonu sayesinde hızlı bir başlangıç yapabilir.
- İlk birkaç iterasyonda daha hızlı yakınsama sağlar, bu da erken durdurma kriterlerinin belirlenmesinde avantaj sağlar.
- Gürültülü ve değişken verilerde daha kararlı güncellemeler yapar.
- Özellikle çevrimiçi öğrenme ve gerçek zamanlı uygulamalarda avantajlıdır.

AdamW'nin Adam'a göre avantajları [117]:

- AdamW, ağırlık çürümesi uygulayarak aşırı uyumu daha etkili bir şekilde önler.
- Modelin test verisi üzerindeki performansını artırır ve genelleme yeteneğini geliştirir.
- AdamW, daha düzenli ağırlık güncellemeleri ile modelin daha iyi genelleme yapmasını sağlar.
- AdamW, eğitim sürecinde daha tutarlı ve stabil performans gösterir.
- Ağırlık çürümesi sayesinde optimizasyon sürecinde daha güvenilir sonuçlar elde edilir.

Derin Öğrenmede, Adam hızlı yakınsama yeteneği nedeniyle derin sinir ağlarının ilk aşamalarında avantaj sağlar. Ayrıca, geniş bir yelpazede kullanılan bir optimizasyon algoritması olduğu için, başlangıç olarak tercih edilir. AdamW ise aşırı uyum riskinin yüksek olduğu büyük ve derin modellerde tercih edilir. Modelin genelleme performansını artırır. Doğal dil işleme de, Adam dil modellerinin ve metin sınıflandırma gibi görevlerin eğitiminde hızlı yakınsama sağlar. AdamW ise dil modellerinin genelleme yeteneğini artırmak için kullanılır, özellikle büyük dil modellerinde tercih edilir.

Her iki algoritmanın da belirli avantajları ve kullanım alanları vardır. Tercih, genellikle veri kümesinin yapısına, modelin karmaşıklığına ve eğitim sürecinde karşılaşılan spesifik zorluklara bağlı olarak yapılır. Bu çalışmada ise AdamW algoritması kullanılmıştır.

3.3.7. Adamax Optimizasyon Algoritması

Adamax, Adam algoritmasının bir uzantısıdır. Adam, gradyanların birinci ve ikinci momentlerinin hareketli ortalamalarını kullanarak adaptif öğrenme oranları hesaplar. Ancak, bazı durumlarda bu hareketli ortalamalar çok büyük değerlere ulaşabilir ve bu da öğrenme sürecini olumsuz etkileyebilir [111]. Adamax, L2 normu yerine Lα normunu kullanarak bu sorunu çözer. Lα normu, bir vektördeki en büyük mutlak değeri ifade eder. Denklem (3.19) ile hesaplanır.

$$\begin{aligned} m_t &= \beta_1 m_{t-1} + (1-\beta_1) g_t \\ u_t &= \max(\beta_2 u_{t-1}, |g_t|) \\ \hat{m}_t &= \frac{m_t}{1-\beta_1^t} \\ \theta_{t+1} &= \theta_t - \frac{\eta}{u_t} \hat{m}_t \end{aligned} \tag{3.19}$$

3.3.8. Nadam Optimizasyon Algoritması

Nadam (Nesterov-accelerated Adaptive Moment Estimation), Adam ve Nesterov Momentumu'nun birleşimidir. Adam, gradyanların birinci ve ikinci momentlerinin hareketli ortalamalarını kullanarak öğrenme oranlarını adaptif hale getirir. Nesterov Momentumu ise, her adımda gelecekteki gradyanları tahmin ederek daha doğru güncellemeler yapmayı sağlar [118].

Nadam, bu iki yöntemi birleştirerek, her adımda hem adaptif öğrenme oranlarını hem de gelecekteki gradyanları hesaba katar. Denklem (3.20) ile hesaplanır.

$$\begin{aligned} m_t &= \beta_1 m_{t-1} + (1-\beta_1) g_t \\ v_t &= \beta_2 v_{t-1} + (1-\beta_2) g_t^2 \\ \hat{m}_t &= \frac{m_t}{1-\beta_1^t} \\ \hat{v}_t &= \frac{v_t}{1-\beta_2^t} \\ \theta_{t+1} &= \theta_t - \frac{\eta}{\sqrt{\hat{v}_t}+\varepsilon} (\hat{m}_t + \beta_1 \frac{1-\beta_1^t}{1-\beta_1^{t+1}} g_t) \end{aligned} \tag{3.20}$$

3.3.9. Adafactor Optimizasyon Algoritması

Adafactor, büyük dil modelleri gibi büyük parametre setleriyle çalışmak üzere optimize edilmiş bir algoritmadır. Adafactor, bellek kullanımını azaltmak için faktörleştirilmiş ikinci moment tahminleri kullanır. Bu, modelin bellek gereksinimlerini önemli ölçüde azaltır ve büyük modellerin daha verimli bir şekilde eğitilmesini sağlar [119]. Adafactor, ayrıca adaptif öğrenme oranlarını hesaplarken hem birinci hem de ikinci moment tahminlerini kullanır. Bu, modelin daha hızlı ve daha doğru öğrenmesini sağlar. Denklem (3.21) ile hesaplanır.

$$\begin{aligned} E[g^2]_t &= \rho E[g^2]_{t-1} + (1-\rho) g_t^2 \\ \hat{v}_t &= \tfrac{1}{d} \textstyle\sum_{i=1}^{d} E[g^2]_t \\ \theta_{t+1} &= \theta_t - \tfrac{\eta}{\sqrt{\hat{v}_t + \varepsilon}} g_t \end{aligned} \tag{3.21}$$

3.3.10. Lion Optimizasyon Algoritması

Lion (Layer-wise learning rate optimizer), her katmanda farklı öğrenme oranları kullanarak derin öğrenme modellerinin performansını artırmayı amaçlar. Bu yöntem, özellikle derin ağlar için daha hassas güncellemeler yapmayı sağlar. Her katmanın öğrenme oranını ayrı ayrı ayarlayarak, modelin farklı bölümlerinin daha verimli bir şekilde öğrenmesini sağlar [120]. Lion, özellikle derin sinir ağlarının eğitimi sırasında öğrenme oranlarının hassas ayarlanması gereken durumlarda etkili olabilir. Bu, modelin genel performansını ve öğrenme hızını artırabilir. Denklem (3.22) ile hesaplanır.

$$\eta_l = \frac{\eta_0}{\sqrt{1+l}} m_t^l = \beta m_{t-1}^l + (1-\beta) g_t^l \theta_{t+1}^l = \theta_t^l - \eta_l m_t^l \tag{3.22}$$

4. METİN MADENCİLİĞİ VE METİN ÖNİŞLEME

Metin Madenciliği (Text Mining), metinlerden çeşitli çıkarımlar yapmak için doğal dil işleme tekniklerini kullanma sürecidir. Düzensiz elektronik metin verilerinden yapılandırılmış bilgi elde etmeyi içerir. Bu sürecin amacı metinlerden yapılandırılmış veri çıkarmaktır. Örneğin, metinlerin kümelenmesi ve sınıflandırılması, metinlerden ilgi alanının belirlenmesi, soru cevaplama, metin özetleme ve varlık ilişkisi modellemesi hedeflenen görevlerdir [121].

Metin ön işleme, istenmeyen kelimelerin ve özel karakterlerin çıkarılmasını içeren daha kolay metin madenciliği için hazırlık sürecidir. NLP görevlerinden önce verileri hazırlamak için kullanılır.

Metin ön işleme, metin madenciliği için hazırlık sürecidir ve istenmeyen kelimelerin, gereksiz özel karakterlerin çıkarılması gibi adımları içerir. Bu süreç, NLP görevlerinden önce verileri temizlemek ve düzenlemek için kullanılır. Örneğin, metinlerdeki yazım hataları düzeltilir, büyük-küçük harf farklılıkları giderilir, kök veya köklenmiş biçimler elde edilir. Bu ön işleme adımları, NLP modellerinin daha doğru ve etkili çalışmasına yardımcı olur.

4.1. METİN ÖN İŞLEME TEKNİKLERİ

Kullanılan metnin özelliğine ve alındığı kaynağa göre farklı metin ön işleme işlemleri mevcuttur. Bu çalışmada, harf dışındaki tüm sembollerin metinden çıkarılması, tüm harflerin küçük harfe çevrilmesi ve kelimelerin kök hallerine getirilmesi yöntemleri kullanılmıştır.

Belirteçlere ayırma (Tokenization), bir cümle veya belge gibi ham metni bir dizi jetona ayırma işlemidir. Belirteçlere ayırma tipik olarak NLP veri ön işlemenin ilk adımıdır ve sonraki işlem adımlarında atomik birimler olarak ele alınan yinelenen metin dizilerinin

tanımlanmasını içerir. Bu belirteçler kelimeler, morfem olarak bilinen alt kelime birimleri (’na-’ gibi ön ekler veya ’-yor’ gibi son ekler gibi) veya tek tek karakterler olabilir.

Köklendirme (Stemming) ve kök çözümleme (Lemmatizasyon), biçimbirimler dilin en küçük anlamlı öğeleridir, sözcüklerden daha küçüktür. Örneğin, "namümkündü" kelimesinde ön ek olarak ’na-’, kök olarak ’mümkün’ ve son ek olarak geçmiş zaman eki ’-idi’ vardır. Kökleştirme ve kök çözümlemede, kelimeler kök formlarına ("namümkün" + geçmiş) eşleştirilir. Kökleştirme ve kök çözümleme, derin öğrenme modellerinde çok önemli adımlardır. Ancak, bu modeller genellikle eğitim verilerinden öğrenilir, bu nedenle açık köklendirme veya kök çözümleme adımları gerekli değildir.

Durak kelimeleri (Stop words) ise kelimelerin bağlam ve anlam açısından önemli roller oynadığı için metinde bırakılması gerekilmektedir. Özellikle duygu analizi ve dil modellemesi gibi görevlerde, durak kelimeleri metnin anlamını etlilemektedir. Örneğin, "çok", "bile" gibi kelimelerin metinlere olumsuz anlam katarak cümlenin anlamını tamamen değiştirebilir. Ayrıca, bazı dil modelleri ve doğal dil işleme algoritmaları, dilin doğallığını korumak ve cümle yapılarını doğru bir şekilde analiz etmek için bu kelimelere ihtiyaç duyar. Bu nedenle, belirli bağlamlarda durak kelimelerinin çıkarılmaması, metnin anlam bütünlüğünü koruyarak daha doğru ve zengin analizler yapılmasını sağlayabilir.

4.2. METİN TEMSİL YÖNTEMLERİ

Dil modelleri olarak da bilinen metin temsil yöntemleri, belirli bir bağlamda metinleri anlamlandırmaya çalışan kelime grupları üzerindeki olasılık dağılımlarıdır. Bir cümlede meydana gelen belirli bir kelime dizisinin olasılığını belirlemek için çeşitli istatistiksel ve olasılıksal teknikler kullanılır. Dil modelleri, soru cevaplama ve makine çevirisi de dahil olmak üzere NLP uygulamalarında yaygın olarak kullanılmaktadır.

Makinelerin metin verilerini işleyebilmesi için öncelikle bu verilerin onlar için anlaşılabilir hale getirilmesi gerekir. Kelimeleri vektörler halinde sayısallaştırmak ve vektör uzayında temsil etmek için kullanılan çeşitli metin temsil yöntemleri vardır. Bu teknik aynı zamanda metin vektörleştirme olarak da adlandırılır.

4.2.1. Kelime Torbası

Kelime Torbası (Bag of Words - BoW), metin madenciliği ve doğal dil işleme alanlarında sıklıkla kullanılan temel bir metin temsil yöntemidir. Bu yöntem, metin verilerini sayısal bir vektör formatında temsil etmek için kullanılır. Belgelerin kelime sırası korunmaz, sadece kelime sayıları dikkate alınır [92]. Öncelikle, bir belgedeki her kelimenin varlığını veya sıklığını bir vektör içinde kodlar. Örneğin, bir belgede geçen her kelime için kelime dağarcığı içinde bir indeks atanır ve belgenin vektörü, her kelimenin belgedeki sıklığına göre ilgili indekslerde değer alır [122]. Belgelerdeki kelimelerin sıklığını gösteren BOW, sınıflandırıcı tarafından bir dizi özelliğe sahip bir öğrenme modeli oluşturmak için kullanılır. Bu yöntem ile veri kümesindeki tüm kelimelerin ve bu kelimelerin kaç kez geçtiği öğrenilir [123].

BoW'ün avantajlarından biri, basit ve hızlı bir şekilde uygulanabilmesidir. Metinler arasındaki benzerlikleri ve farklılıkları göstermek için etkili bir yöntemdir. Ancak, BoW kelime sırasını ve kelimenin bağlamını dikkate almaz, bu da metnin anlam karmaşıklığını tam olarak yansıtmada kısıtlamalar getirir [124].

BoW, temel bir başlangıç noktası olmasına rağmen, modern doğal dil işleme uygulamalarında daha gelişmiş yöntemler (örneğin, Word Embeddings) ile birlikte kullanılarak metinlerin daha derin anlamsal yapılarını yakalamak için genişletilebilir [125].

4.2.2. Terim Frekansı-Ters Belge Frekansı

Terim Frekansı-Ters Belge Frekansı (Term Frequency-Inverse Document Frequency - TF-IDF), belirli bir belgedeki kelimelerin göreli sıklığını ve veri kümesindeki bu kelimelerin tersine oranını belirleyerek çalışır. Bir terimin belgedeki önemini gösteren istatistiksel yöntemle hesaplanan ağırlık faktörüdür [126]. TF, Luhn'un [127], IDF ise Jones'un [128] çalışmalarıyla ortaya çıkardıkları iki ayrı yöntemdir.

TF, bir kelimenin belgede kaç defa geçtiğini belirtir. Bir kelimenin belgede geçme sayısı, belgedeki toplam kelime sayısına bölünerek bulunur. Diğer bir tanımlamayla bir belgedeki

terim ağırlıklarını hesaplamak için kullanılan yöntemdir. Denklem (4.1)'de TF yöntemine ait denklem görülmektedir.

$$TF(i,j) = \frac{j \text{ belgesinde } i \text{ terim sayısı}}{j \text{ belgesindeki toplam kelime}} \tag{4.1}$$

IDF ise toplam belge sayısının, ilgili kelimenin geçtiği belge sayısına bölünmesiyle elde edilir. Bu işlem sonucunda çıkan değer sıfıra yaklaştıkça kelimenin birçok yerde geçtiği anlaşılır. Bir değerine yakınsa da daha az geçtiği anlaşılır. Bu da ilgili kelimenin belgede önem derecesi olan IDF değerini gösterir. IDF, Denklem (4.2)'de görülmektedir [129]. TF-IDF skoru, Denklem (4.3)'te verildiği gibi hesaplanır.

$$IDF(i) = log\left(\frac{\text{Toplam belge}}{\text{belgede gecen } i \text{ terimi}}\right) \tag{4.2}$$

$$j = TF(i,j) * IDF(i) \tag{4.3}$$

4.2.3. Word2Vec

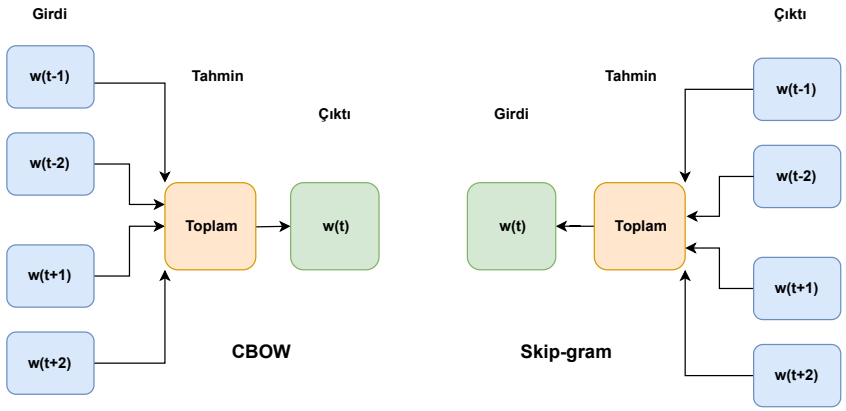
Sinir ağlarını kullanan en popüler metin temsil tekniklerden biri de Word2Vec'tir. Tomas Mikolov ve ekibi tarafından geliştirilmiştir. Word2Vec'in Gram Atlama (Skip-Gram) ve Sürekli Kelime Torbası (Continuous Bag of Words-CBOW) olmak üzere iki yöntemi vardır. Mikolov [125] ve ekibinin geliştirdiği kelime gömme yöntemi olan Word2Vec'in, üzerinde yapılan çalışmalarla GloVe, fastText gibi yeni yöntemler geliştirilmiştir.

CBOW yöntemi, merkez kelimenin etrafındaki kelimelerden merkezi kelimeyi tahmin etmeye çalışmaya dayanır. CBOW yöntemi küçük veri kümelerinde daha iyi tahmin yapmaktadır. Öte yandan birden fazla anlamı olan kelimeleri anlamakta zorlanır [126].

CBOW yöntemi, geçmiş kelimelerin yanı sıra gelecekteki kelimelerde de kullanılan bir mimaridir. CBOW için amaç fonksiyonu Denklem (4.4)'te verilmiştir [130]:

$$J_\theta = \frac{1}{T} = \sum_{t=1}^{T} log\, p\left(w_t | w_{t-n}, \cdots\cdots, w_{t-1}, w_{t+1}, \cdots\cdots, w_{t+n}\right) \tag{4.4}$$

CBOW yönteminde, pencerenin ortasındaki kelimeyi tahmin etmek için bağlamın dağıtılmış temsilleri kullanılır [131]. CBOW yönteminin yapısına ilişkin görsel, Şekil 4.1'de gösterilmektedir [125].



Şekil 4.1. Word2Vec mimarisi ([125]).

Skip-Gram yönteminde, pencere boyutu merkezinde olan kelimeler girdi olarak alınıp pencere boyutu merkezinde olmayan kelimeler çıktı olarak tahmin edilmeye çalışılmaktadır. CBOW'da olduğu gibi merkez kelimeyi tahmin etmek için çevreleyen kelimeleri kullanmak yerine, bu kelimeleri tahmin etmek için merkezi kelime kullanılır. Skip-Gram amaç fonksiyonu, Denklem (4.5)'te verilen Skip-Gram için hedefi üretmek üzere, hedef kelimenin solundaki ve sağındaki n kelimeyi çevreleyen n'nin logaritmik olasılıklarının toplamıdır [125]:

$$J_\theta = \frac{1}{T} = \sum_{t=1}^{T} \sum_{t=1}^{T} log\, p\left(w_{j+1} | w_t\right) \tag{4.5}$$

Skip-Gram yöntemi, bir kelimenin etrafında bulunan kelimeleri tahmin etmeye çalışır. Bu yöntem, daha büyük verilerle daha iyi çalışmaktadır. Aynı zamanda Skip-Gram yöntemi, CBOW'a göre iki veya daha fazla anlamı olan kelimeleri anlamakta daha başarılıdır.

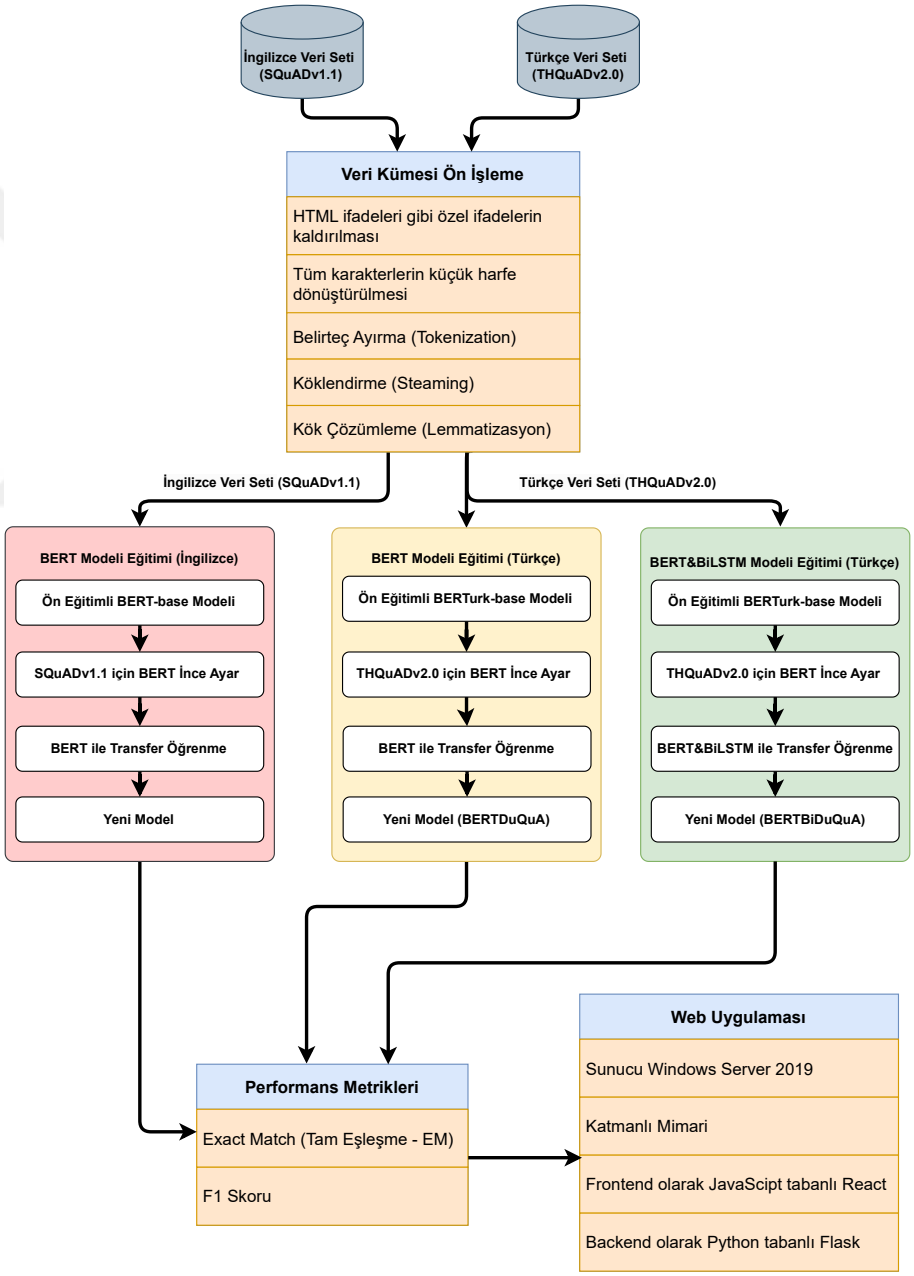
4.3. DOĞAL DİL İŞLEME

NLP, bilgisayarların insan dilini anlamasını, üretmesini ve işlemesini sağlayan yapay zekânın bir alt dalıdır. Son yıllarda, makine öğrenimi, özellikle de verilerden öğrenerek genelleme yapan bir yapay zekâ dalı olan derin öğrenme yaygınlaşmıştır. Derin öğrenme, büyük veri kümelerindeki örüntüleri tanımlayabilen bir makine öğrenimi türüdür. Günlük hayatta, NLP tekniklerini ve teknolojilerini kullanan Siri, Copilot ve Google Assistant gibi sanal asistanlar son kullanıcılar aracılığıyla NLP ile etkileşime girmektedir. Bu asistanlar kullanıcının isteğini anlar ve NLP kullanarak doğal dilde yanıt verir. İnsan dilini anlayabilen, üretebilen veya işleyebilen bilgisayarlı varlıklar oluşturmaya odaklanan bir mühendislik alanıdır [132].

NLP, metin işleme, bilgi çıkarma, konuşma tanıma, soru cevaplama, otomatik çeviri, sesten metne çeviri, metinden konuşmaya çeviri, başlık sınıflandırma ve duygu analizi gibi çeşitli alt görevleri kapsayan geniş bir çalışma alanıdır [86]. Uygulamaları, sohbet robotları ve arama motoru optimizasyonundan sosyal medya analizi, içerik yönetimi gibi çok çeşitlidir. NLP için makine öğrenimi, derin öğrenme, önceden eğitilmiş modeller, transfer öğrenimi ve metin ön işleme tekniklerinin kullanımı açıklanmaktadır.

5. MATERYAL VE YÖNTEM

Bu bölümde tez çalışmasında faydalanılan uygulamalar, programlara dilleri ve kütüphanelerine, veri setlerine, mimariler ve değerlendirmeler için kullanılan ölçütlerine yer verilmiştir. Şekil 5.1'de yapılan işlemlerin diyagram hali yer almaktadır.



Şekil 5.1. Tez akış diyagramı.

5.1. PROGRAMLAMA DİLLERİ, GELİŞTİRME ORTAMLARI VE KÜTÜPHANELER

Bu bölümde tez çalışmasında kullanılan programlama dilleri, geliştirme ortamları ve kütüphanelerden bahsedilecektir.

5.1.1. Python

Python programlama dili, çok sayıda hazır araç ve kütüphaneye sahip olması nedeniyle NLP projelerinde yaygın olarak kullanılmaktadır. Java, Go, JavaScript, C++, Julia ve R gibi diğer üst düzey diller de nesne yönelimli programlama yetenekleri ve benzer araç ve kütüphanelere sahip olmaları nedeniyle sıklıkla tercih edilmektedir [133]. Tez çalışmasında Python programlama dilinin 3.8.16 sürümü kullanılmıştır.

5.1.2. Google Colab

Google Colab, Google Cloud üzerinde çalışan ücretsiz bir Jupyter Not Defteri ortamıdır. Google Colab, kullanıcılarına yaptıkları çalışmaları kolayca paylaşma ve çoğaltma ve bulut ortamında saklama imkanı tanımaktadır [134]. Google Colab, güçlü GPU'lara ücretsiz erişim imkanı sağlar ve bu sayede çalışma sürelerini kısaltabilmektedir. Bu özelliklere ek olarak, Google Drive ve Google Sheets gibi diğer Google hizmetleriyle entegrasyonu sayesinde veri alma ve gönderme işlemlerini kolaylaştıran özelliklere sahiptir [135]. Tez çalışmasında hiperparametre optimizasyon uygulaması colab ortamında gerçekleştirilmiştir.

5.1.3. PyTorch

PyTorch [136] sıklıkla kullanılan ücretsiz derin öğrenme çerçevesidir. Python dilinin yanında farklı diller aracılığıyla da kullanılabilir. Önceden yazılan hazır bileşenlerden oluşan büyük kütüphaneler, bu altyapının içerisinde yer alır. Grafik işlemci birimi (GPU), hızlandırıcılar içeren yüksek performanslı bilgi işlem altyapılarını destekler [137]. Tez çalışmasında PyTorch 1.10.2 versiyonu kullanılmıştır.

5.1.4. HuggingFace

Çok sayıda ve birbirinden farklı ön eğitimli, çoğunluğu derin öğrenmeye dayalı NLP modelleri barındıran bir girişimci ekip tarafından sunulan bir ortamdır. Bu ortam, NLP görevleri üzerine çalışacak araştırmacılara hızlı bir şekilde yol gösteren ve kullanıma hazır bir yazılım sağlar [138]. Çalışma kapsamında HuggingFace de bulunan BERT-based ve BERTurk-based modelleri kullanılmıştır.

5.1.5. PyCharm

PyCharm, JetBrains tarafından geliştirilen, Python programlama dili için kapsamlı bir Entegre Geliştirme Ortamı (IDE) sunan popüler bir yazılım aracıdır [139]. 2010 yılında piyasaya sürülen PyCharm, özellikle kod tamamlama, hata ayıklama, test etme ve versiyon kontrolü gibi özelliklerle yazılım geliştirme sürecini büyük ölçüde kolaylaştırır. PyCharm'ın temel özellikleri arasında akıllı kod editörü, entegre hata ayıklayıcı, test koşucusu, ve çeşitli Python kütüphaneleri ve frameworkleri ile uyumluluk bulunur [140]. Tez çalışmasında yazılım geliştirme ortamı olarak kullanılmıştır.

5.1.6. React

React, Facebook tarafından geliştirilen ve ilk kez 2013 yılında piyasaya sürülen bir açık kaynaklı JavaScript kütüphanesidir [141]. React, geliştiricilere web uygulamalarını daha modüler ve yeniden kullanılabilir bileşenler halinde tasarlama imkanı tanır, bu da büyük ve karmaşık uygulamaların yönetimini kolaylaştırır. React'in popülaritesi, geniş topluluk desteği ve zengin ekosistemi ile desteklenmektedir, bu da onu modern web geliştirme için güçlü bir araç haline getirir [142]. Tez çalışmasında React 18.2.0 versiyonu kullanılmıştır.

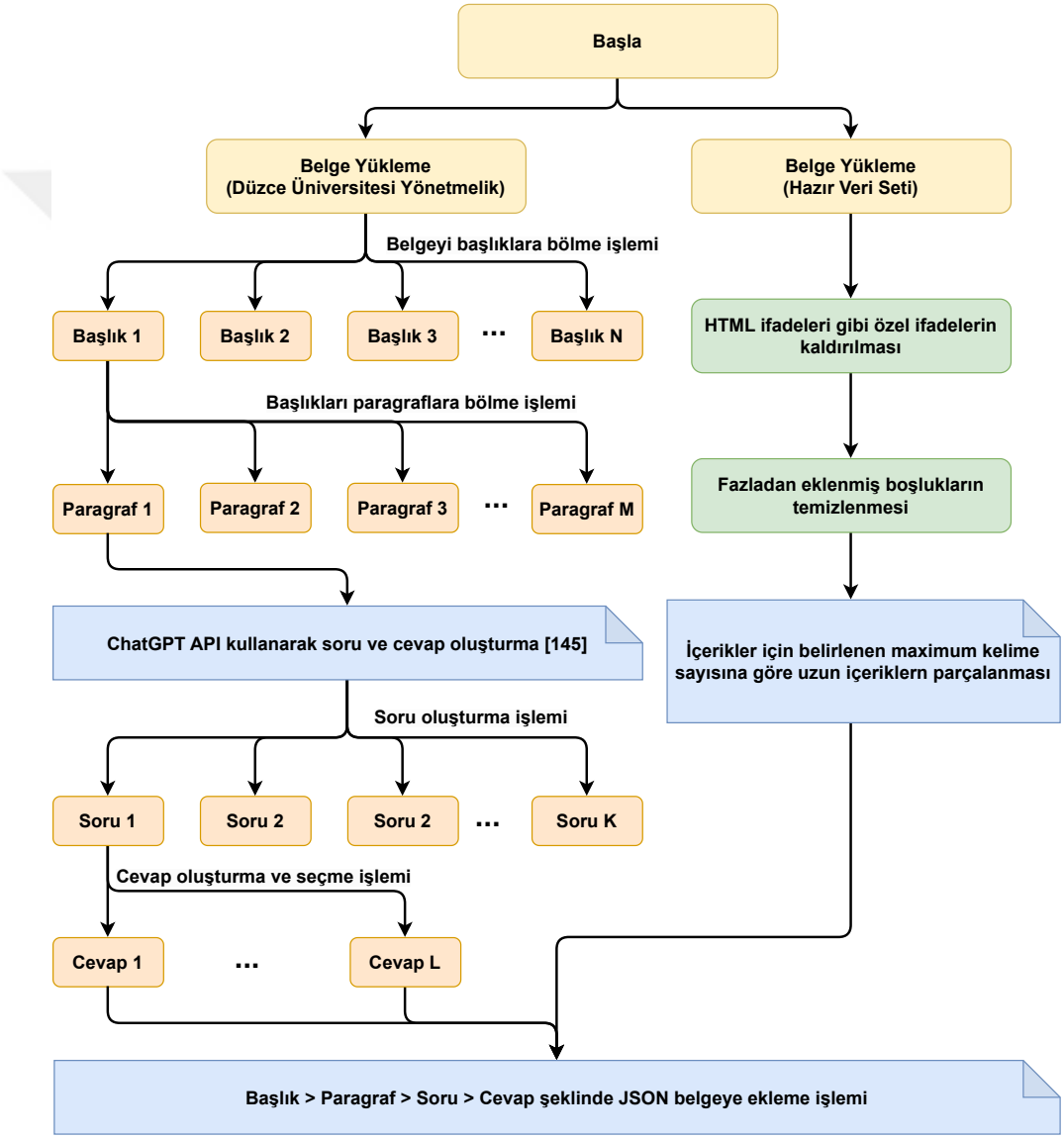
5.1.7. Flask

Flask, Python ile yazılmış hafif ve esnek bir mikro web frameworküdür. 2010 yılında Armin Ronacher tarafından başlatılan Flask, basit ve genişletilebilir yapısı ile öne çıkar [143]. Flask, "micro" terimiyle minimal çekirdek işlevselliğe sahip olduğunu, ancak genişletilebilir bir yapıda olduğunu ifade eder. Flask, WSGI (Web Server Gateway

Interface) uyumluluğu sayesinde çeşitli web sunucuları ile sorunsuz bir şekilde çalışır [144]. Tez çalışmasında Flask 3.0.3 sürümü kullanılmıştır.

5.2. VERİ SETİ

Çalışmada iki farklı dile göre veri seti kullanılmıştır. Yeni veri seti oluştururken Şekil 5.2'de akış diyagramına göre bir ara uygulama geliştirilmiştir. Bu uygulama ile kolay bir şekilde veri seti hazırlanabilmektedir.



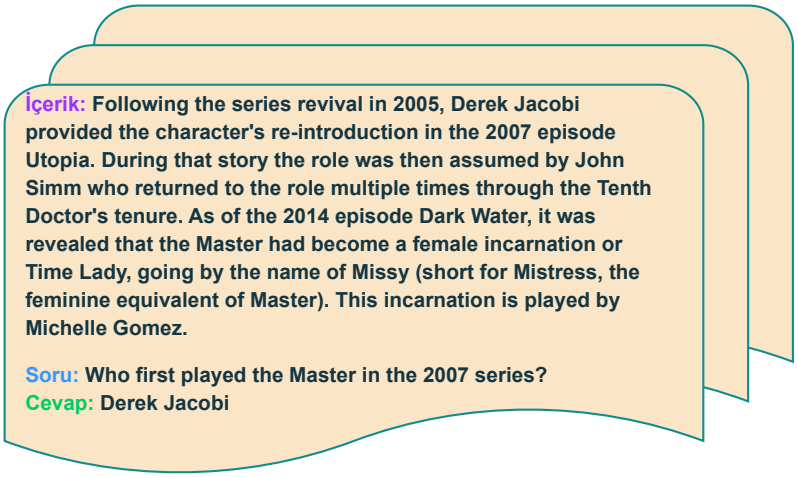
Şekil 5.2. Yönetmeliklerden ara bir uygulama ile veri seti oluşturma işlemi.

ChatGPT [145] ile bir belgeden SQuADv1.1 formatında soru ve cevap oluşturmak, belgenin paragraflara ayrılması, paragraflardaki cümlelere dayalı soruların oluşturulması ve cevapların belirlenmesi süreçlerini içerir. Sorular "Kim?", "Ne?", "Nerede?", "Ne zaman?",

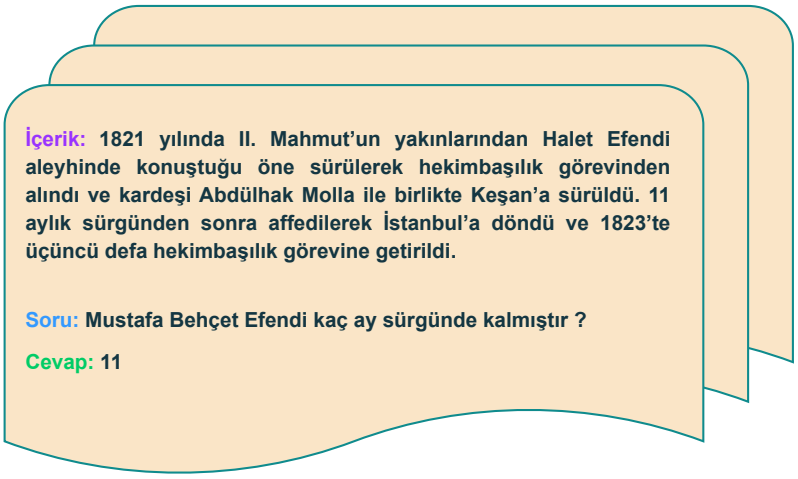
"Nasıl?" ve "Neden?" gibi temel bilgi isteyen türlerdedir. Sorulara cevap olarak, paragrafta doğrudan bulunabilecek metin parçaları seçilir. Cevapların tekli, çoklu veya alternatif cevaplar olarak seçilmesi insan denetimine bırakılmaktadır. ChatGPT'nin mantıklı cevaplar oluşturma yeteneğine rağmen cevapların kesin doğruluğu için insan denetiminden geçmesi gerekmektedir. Sorular ve cevaplar, SQuADv1.1 formatına uygun şekilde JSON formatında düzenlenir.

İngilizce soru cevap için Stanford Üniversitesi'nin soru cevaplama alanı üzerine oluşturduğu SQuADv1.1 veri seti kullanılmıştır. SQuADv1.1, Wikipedia makalelerindeki paragraflara ait her bir sorunun cevabının bulunduğu okuduğunu anlama veri setidir. Wikipedia'dan elde edilmiş 536 makale için 23 215 paragraf ve 107 785 soru cevap çifti içermektedir. Soruların cevapları çoktan seçmeli değildir, en iyi olasılığa sahip tek cevap bulunmaktadır [23].

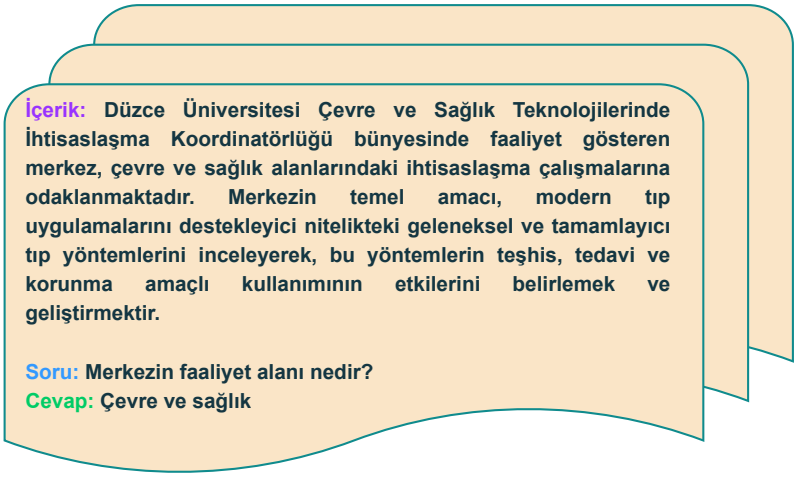
Türkçe soru cevaplama için ise düzenlenmiş THQuADv1.0 kullanılmıştır. Veri setine Düzce Üniversitesi yönergelerine ait sorular eklenmiştir. THQuADv1.0 veri seti; 841 başlık, 2 701 pasaj ve 15 554 soru ve cevap içermektedir. Düzce Üniversitesi yönerge veri seti ise 15 başlık 74 pasaj ve 510 soru ve cevap içermektedir. İki veri setinin birleşmiş haliyle toplamda 856 başlık 2 775 pasaj ve 16 064 soru ve cevap seti olarak THQuADv2.0 oluşturulmuştur. Veri setlerine ait herhangi bir paragrafa ait sorular ve cevaplar Şekil 5.3, Şekil 5.4 ve Şekil 5.5'te yer almaktadır.



Şekil 5.3. SQuADv1.1 veri seti için soru cevap örneği.



Şekil 5.4. THQuADv1.0 veri seti için soru cevap örneği.



Şekil 5.5. THQuADv2.0 veri seti için soru cevap örneği.

Düzce Üniversitesi yönetmelikler kısmından alınan veriler aşağıda listelenmektedir. Bunlara ek olarak Düzce Üniversitesi web sitesinde bulunan tarihçe, genel bilgi ve kuruluş metinlerinin de soru cevap başlıkları olarak eklenmiştir. Burada bulunan yönetmelikler bir veya birden fazla kısa paragraflar haline getirilmiştir. Oluşturulan bu paragraflardan soru çıkarılmış ve cevapları etiketlenmiştir. Şekil 5.6'da verilerin JSON a nasıl eklendiğine dair bir görsel yer almaktadır.

- Düzce Tarihçe
- Düzce Genel Bilgi
- Düzce Üniversitesi Kuruluş
- Geleneksel ve Tamamlayıcı Tıp (GTT) Uygulama ve Araştırma Merkezi Yönetmeliği

- Bitinya Arkeolojisi Uygulama ve Araştırma Merkezi Yönetmeliği
- Radyo-Televizyon Uygulama ve Araştırma Merkezi Yönetmeliği
- Dil, Tarih ve Kültürel Zenginlikleri Uygulama ve Araştırma Merkezi Yönetmeliği
- Döner Sermaye İşletmesi Yönetmeliği
- Kariyer Geliştirme ve Mezun İzleme Uygulama Araştırma Merkezi Yönetmeliği
- Organ Nakli Eğitim, Uygulama Ve Araştırma Merkezi Yönetmeliği
- Ahşap İşleri Uygulama Ve Araştırma Merkezi Yönetmeliği
- Lisans Eğitim–Öğretim ve Sınav Yönetmeliği
- Fındık Uygulama ve Araştırma Merkezi Yönetmeliği
- Kadın Çalışmaları Uygulama ve Araştırma Merkezi Yönetmeliği
- Tıp Fakültesi Eğitim Öğretim ve Sınav Yönetmeliği
- Uzaktan Eğitim Uygulama ve Araştırma Merkezi Yönetmeliği
- Bağımlılıkla Mücadele Uygulama ve Araştırma Merkezi Yönetmeliği
- Tarımsal Atıkların Endüstriye Geri Kazanımı Uygulama Ve Araştırma Merkezi Yönetmeliği


```
"version": "v2.0",
"data": [
  {
    "title": "Geleneksel ve Tamamlayıcı Tıp (GTT) Uygulama ve Araştırma Merkezi Yönetmeliği",
    "paragraphs": [
      {
        "qas": [
          {
            "question": "Merkezin faaliyet alanı nedir?",
            "id": 1,
            "answers": [
              {
                "answer_start": 118,
                "text": "Çevre ve sağlık."
              }
            ]
          },
          {
            "question":  "Merkezin temel amacı ne? ",
            "id": 2,
            "answers": [
              {
                "answer_start": 464,
                "text": "Geleneksel tıp yöntemlerini inceleyerek, modern tıp uygulamalarını desteklemek."
              }
            ]
          }
        ]
      "context": "Düzce Üniversitesi Çevre ve Sağlık Teknolojilerinde İhtisaslaşma Koordinatörlüğü bünyesinde ...
```


Şekil 5.6. Eklenen verilerin JSON örneği.

Türkçe veri seti üzerinde yapılan tüm işlemler ise aşağıda listelenmiştir.

- Çizelge 5.1'de içerikleri uzun olan yapılar parçalara bölünerek daha kolay işlenmesi sağlanmıştır.
- Çizelge 5.2'de veri setinde bulunan bozuk yapıdaki veriler düzeltilmiştir.
- Çizelge 5.3'te tek cevaplı yapılar İngilizce veri seti (SQuADv1.1)'indeki gibi çok cevaplı olacak şekilde düzenlenmiştir.
- Çizelge 5.4'te soru çeşitliliğini arttırmak için Düzce Üniversitesi yönetmelikleriyle ilgili ise 15 başlık 74 pasaj ve 510 soru ve cevap eklenmiştir.

Çizelge 5.1. İçerikleri uzun olan yapılar parçalara ayrılması.

Durum	Veri
Eski	**İçerik:** İstanbul'un Fethi, 6 Nisan-29 Mayıs 1453 tarihleri arasındaki kuşatmanın sonucunda Osmanlı Padişahı II. Mehmed komutasındaki birliklerin Bizans İmparatorluğu'nun başkenti İstanbul'u ele geçirmesiyle son buldu. İstanbul, daha önce de defalarca kuşatılmıştı; VII.-VIII. asırlarda Emeviler ve Abbasiler tarafından kuşatıldı ancak başarısız olundu. Osmanlılar da şehri daha önce kuşatmıştı, Matheos Kantakuzinos'un Bizans tahtına geçmesini sağlamışlar ve ödül olarak Çimpe Kalesi'ni alarak Rumeli'de ilk kez toprak kazanmışlardı. Rumeli'ye geçişle beraber bölgede sınırları genişleyen Osmanlılar ilk kez I. Bayezid komutasında 1395 yılında İstanbul'u kuşattı. Bazı kaynaklarda ise 1391 tarihli farklı bir kuşatmadan söz edilmektedir. I. Bayezid'in kuşatmasında mancınıklar kullanıldı, kuşatma üzerine Macar Krallığı günümüz Bulgaristan topraklarına taarruz etti ve İstanbul kuşatması sonlandırıldı. Ertesi yıl kuşatma tekrar başladı ve bu sefer deniz bağlantısını tümüyle koparmak için Anadolu Hisarı inşa edildi. Bizans imparatorunun ateşkes talebi üzerine kuşatma kaldırıldı. Ankara Savaşı'yla beraber Osmanlı Devleti Fetret Devri'ne girdi..Bu dönemde Bayezid'in oğullarından Musa Çelebi tarafından 1412 yılında İstanbul tekrar kuşatıldı. Musa Çelebi, kargaşanın Bizans yüzünden olduğuna ve bazı rakip şehzadelerin Bizans tarafından desteklendiğine inanıyordu. Ancak rakip şehzadelerden I. Mehmed'in harekete geçmesi sebebiyle bu kuşatma da kaldırıldı. Dördüncü kuşatma ise II. Murad döneminde oldu; II. Murad elçiler göndererek Düzmece Mustafa'nın desteklenmemesini talep etti ancak karşılık bulamadı. İsyan ile uğraşan II. Murat, Şehzade Mustafa'ya yardım ettiğine inandığı için Bizans İmparatorluğu'nun üzerine yürüdü ve kuşatma başladı. Bizans İmparatoru VII. İoannis'in Karadeniz kıyılarındaki bazı toprakları ve haraç vermeyi teklif etmesiyle bu kuşatma da kaldırıldı. II. Mehmed tahta geçtiğinde etrafı bütünüyle sarılmış bir şehirle karşı karşıyaydı.

Çizelge 5.1. (devam) İçerikleri uzun olan yapılar parçalara ayrılması.

Durum	Veri
Yeni	**İçerik 1:** İstanbul'un Fethi, 6 Nisan-29 Mayıs 1453 tarihleri arasındaki kuşatmanın sonucunda Osmanlı Padişahı II. Mehmed komutasındaki birliklerin Bizans İmparatorluğu'nun başkenti İstanbul'u ele geçirmesiyle son buldu. İstanbul, daha önce de defalarca kuşatılmıştı; VII.-VIII. asırlarda Emeviler ve Abbasiler tarafından kuşatıldı ancak başarısız olundu. Osmanlılar da şehri daha önce kuşatmıştı, Matheos Kantakuzinos'un Bizans tahtına geçmesini sağlamışlar ve ödül olarak Çimpe Kalesi'ni alarak Rumeli'de ilk kez toprak kazanmışlardı. Rumeli'ye geçişle beraber bölgede sınırları genişleyen Osmanlılar ilk kez I. Bayezid komutasında 1395 yılında İstanbul'u kuşattı. Bazı kaynaklarda ise 1391 tarihli farklı bir kuşatmadan söz edilmektedir. I. Bayezid'in kuşatmasında mancınıklar kullanıldı, kuşatma üzerine Macar Krallığı günümüz Bulgaristan topraklarına taarruz etti ve İstanbul kuşatması sonlandırıldı. Ertesi yıl kuşatma tekrar başladı ve bu sefer deniz bağlantısını tümüyle koparmak için Anadolu Hisarı inşa edildi. **İçerik 2:** Bizans imparatorunun ateşkes talebi üzerine kuşatma kaldırıldı. Ankara Savaşı'yla beraber Osmanlı Devleti Fetret Devri'ne girdi. Bu dönemde Bayezid'in oğullarından Musa Çelebi tarafından 1412 yılında İstanbul tekrar kuşatıldı. Musa Çelebi, kargaşanın Bizans yüzünden olduğuna ve bazı rakip şehzadelerin Bizans tarafından desteklendiğine inanıyordu. Ancak rakip şehzadelerden I. Mehmed'in harekete geçmesi sebebiyle bu kuşatma da kaldırıldı. Dördüncü kuşatma ise II. Murad döneminde oldu; II. Murad elçiler göndererek Düzmece Mustafa'nın desteklenmemesini talep etti ancak karşılık bulamadı. İsyan ile uğraşan II. Murat, Şehzade Mustafa'ya yardım ettiğine inandığı için Bizans İmparatorluğu'nun üzerine yürüdü ve kuşatma başladı. Bizans İmparatoru VII. İoannis'in Karadeniz kıyılarındaki bazı toprakları ve haraç vermeyi teklif etmesiyle bu kuşatma da kaldırıldı. II. Mehmed tahta geçtiğinde etrafı bütünüyle sarılmış bir şehirle karşı karşıyaydı.

Çizelge 5.2. Veri setinde bulunan bozuk yapıdaki verilerin düzeltilmesi.

Durum	Veri
Eski	**İçerik:** \r\nBattani, matematikte trigonometride günümüzde kullanılan formüller üretmiştir: \r\nAyrıca sin x = a cos x eşitliğini buldu, formül: \r\nBattani, El-Mervezi'nin tanjant fikrini, tanjant ve kotanjant hesaplamaları amacıyla denklemler geliştirmek için konu hakkındaki matematiksel tablolarını derleyerek kullanmıştır. Bundan başka sekant ve kosekantın işteş fonksiyonalrını keşfetmiş ve O'nun gölgelerin tablosu olarak adlandırdığı, kosekantlar hakkındaki ilk matematiksel tabloyu, 1'den 90'a kadar her bir dereceyi içerecek şekilde hazırlamıştır.

Çizelge 5.2. (devam) Veri setinde bulunan bozuk yapıdaki verilerin düzeltilmesi.

Durum	Veri
Yeni	**İçerik:** Battani, matematikte trigonometride günümüzde kullanılan formüller üretmiştir: Ayrıca sin x = a cos x eşitliğini buldu, formül: Battani, El-Mervezi'nin tanjant fikrini, tanjant ve kotanjant hesaplamaları amacıyla denklemler geliştirmek için konu hakkındaki matematiksel tablolarını derleyerek kullanmıştır. Bundan başka sekant ve kosekantın işteş fonksiyonlarını keşfetmiş ve O'nun gölgelerin tablosu olarak adlandırdığı, kosekantlar hakkındaki ilk matematiksel tabloyu, 1'den 90'a kadar her bir dereceyi içerecek şekilde hazırlamıştır.

Çizelge 5.3. Bir den fazla cevap olacak şekilde güncelleme.

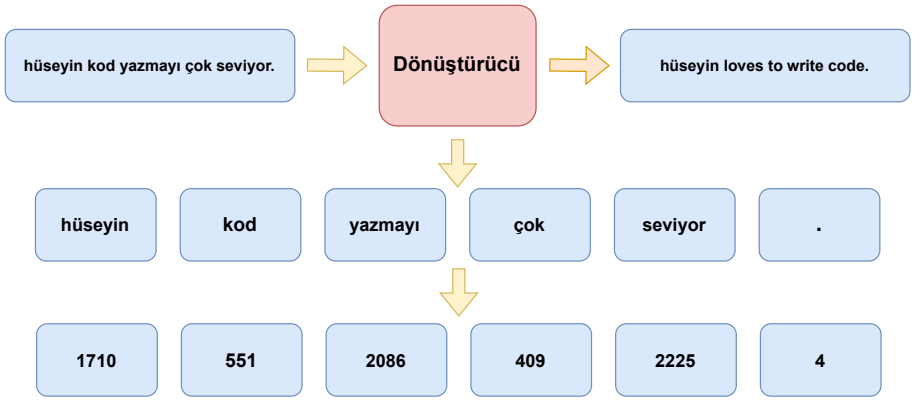
Durum	Veri
Eski	**İçerik:** Şeyhülislam olarak fetva yazımında büyük yetenek gösterdi. Şeyhülislamlığı ve müderrisliği dışında asıl ününü Hoca Tarihi olarak da anılan Tac üt-tevarih isimli yapıtıyla kazandı. Ayrıca padişah III. Murad'ın emri ile Molla Muslihittin Lari'nin iki eseri ile Abdülkadir Geylani'nin Menakıbını Türkçeye tercüme etti. **Soru:** Hoca Sadeddin Efendi asıl ününü neyle kazanmıştır? **Cevap:** Tac üt-tevarih isimli yapıtıyla
Yeni	**İçerik:** Şeyhülislam olarak fetva yazımında büyük yetenek gösterdi. Şeyhülislamlığı ve müderrisliği dışında asıl ününü Hoca Tarihi olarak da anılan Tac üt-tevarih isimli yapıtıyla kazandı. Ayrıca padişah III. Murad'ın emri ile Molla Muslihittin Lari'nin iki eseri ile Abdülkadir Geylani'nin Menakıbını Türkçeye tercüme etti. **Soru:** Hoca Sadeddin Efendi asıl ününü neyle kazanmıştır? **Cevap 1:** Tac üt-tevarih isimli yapıtıyla **Cevap 2:** Hoca Tarihi olarak da anılan Tac üt-tevarih isimli yapıtıyla **Cevap 3:** Tac üt-tevarih

Çizelge 5.4. Eklenen veri örneği.

Durum	Veri
Yeni	**İçerik:** Düzce Üniversitesi Çevre ve Sağlık Teknolojilerinde İhtisaslaşma Koordinatörlüğü bünyesinde faaliyet gösteren Merkez, çevre ve sağlık alanlarındaki ihtisaslaşma çalışmalarına odaklanmaktadır. Merkezin temel amacı, modern tıp uygulamalarını destekleyici nitelikteki geleneksel ve tamamlayıcı tıp yöntemlerini inceleyerek, bu yöntemlerin teşhis, tedavi ve korunma amaçlı kullanımının etkilerini belirlemek ve geliştirmektir. Bu çerçevede, Ar-Ge çalışmaları yoluyla, geleneksel ve tamamlayıcı tıp yöntemlerinin etki mekanizmalarını anlamak ve Sağlık Bakanlığı mevzuatına uygun ürünler üreterek döner sermaye işletmesi kapsamında bu ürünlerin satışını gerçekleştirmek hedeflenmektedir. Ayrıca, Merkez, sağlık alanında öğrenim gören öğrenci ve mezunlar başta olmak üzere, bölge insanına yönelik sertifikalı eğitim-öğretim faaliyetleri düzenlemekte ve geleneksel ve tamamlayıcı tıp uygulamaları için ürün geliştirme çalışmalarında bulunmaktadır. **Soru:** Merkezin hedefleri nelerdir? **Cevap 1:** Ar-Ge çalışmaları **Cevap 2:** Ar-Ge çalışmaları, ürün satışı **Cevap 3:** Ar-Ge çalışmaları, ürün satışı, sertifikalı eğitimler, ürün geliştirme.

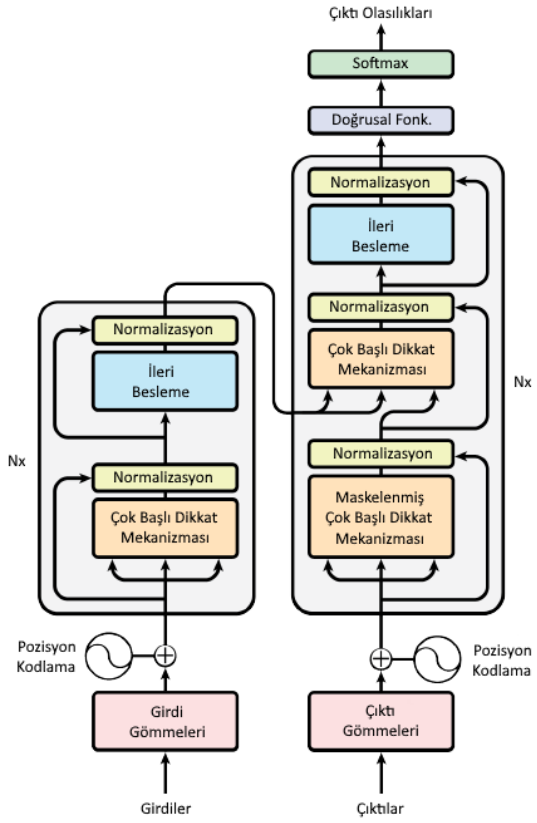
5.3. DÖNÜŞTÜRÜCÜ MİMARİSİ

Dönüştürücüler (Transformers), 2017 yılında Vaswani ve arkadaşları tarafından tanıtılan bir tür sinir ağı mimarisidir [31]. Doğal dil işleme görevlerinde RNN'lerin ve LSTM'lerin sınırlamalarının üstesinden gelmek için tasarlanmıştır. Şekil 5.7'deki örnekte bir çevirme işlemi için dönüştürücü girdi yapısı gösterilmiştir.



Şekil 5.7. Dönüştürücü girdi yapısı.

Dönüştürücüler, tekrarlayan bir yapıya dayanmadan bir cümledeki farklı kelimeler arasındaki ilişkileri yakalamalarını sağlayan Dikkat (Attention) fikrine dayanmaktadır. Dönüştürücü doğal dil işleme modeli, cümledeki tüm kelimeler arasındaki ilişkiyi dikkate alan bir "Dikkat" mekanizması getirmiştir. Cümledeki diğer unsurlardan hangilerinin bir kelimenin yorumlanmasında en kritik öneme sahip olduğunu gösteren farklı ağırlıklar oluşturur. Bu şekilde belirsiz unsurlar hızlı ve verimli bir şekilde çözülebilir. Kod çözücü sadece bağlam vektörüne güvenmek yerine kodlayıcının tüm geçmiş durumlarına erişebilir.Dönüştürücü mimarisi, modelin NPL eğitim uygulamaları için giderek daha fazla kullanılan GPU'larda bulunan güçlü paralel işleme rutinlerinden yararlanmasına da olanak tanır.Şekil 5.8'de Dönüştürücü'lerin genel mimarisi yer almaktadır. Sol tarafta yer alan kodlayıcı ve sağ tarafta yer alan ise kod çözücüdür.



Şekil 5.8. Dönüştürücü mimarisi ([31]).

5.3.1. Kodlayıcı ve Kod Çözücü

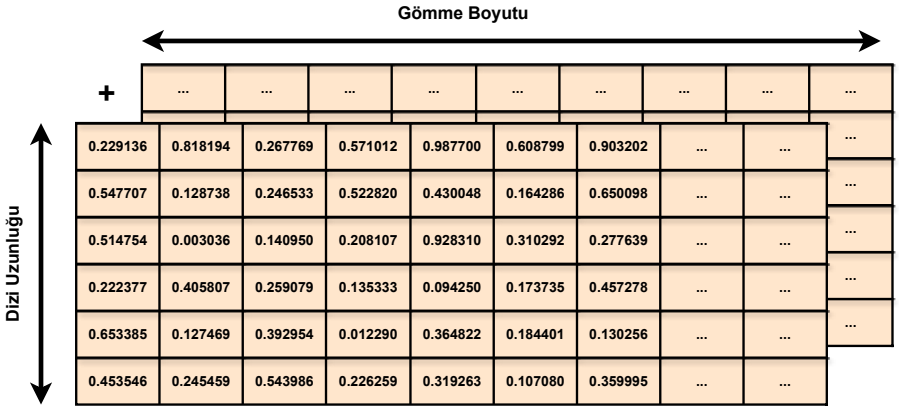
Kodlayıcı (Encoder), 6 özdeş katmandan oluşur. Her katmanın iki alt katmanı vardır: birincisi çok başlı kendi kendine dikkat mekanizması, ikincisi ise basit, pozisyona göre tam bağlı ileri beslemeli bir ağdır. İki alt katmanın her birinin etrafında bir artık bağlantı ve normalizasyon katmanı vardır [146]. Her bir alt katmanın çıktısı LayerNorm(x + Sublayer(x)) şeklindedir, burada Sublayer(x) alt katmanı tarafından uygulanan fonksiyondur. Modeldeki tüm alt katmanlar ve gömme katmanları 512 boyutlu çıktılar üretir.

Kod çözücü (Decoder) 6 özdeş katmandan oluşmaktadır. Her kodlayıcı katmanındaki iki alt katmana ek olarak kod çözücü, kodlayıcı yığınının çıktısı üzerine çoklu başı gerçekleştiren üçüncü bir alt katman ekler. Kodlayıcı gibi, alt katmanların her birinin etrafında artık bağlantılar ve ardından da normalizasyon katmanı yer alır [146]. Ek olarak öz dikkat alt katmanı da pozisyonların sonraki pozisyonlara katılmasını önlemek için değiştirilir. Bu maskelemeyle, çıktı gömülmelerinin bir konum kaydırılmış olması gerçeğiyle birleştiğinde i pozisyonu için tahminler yalnızca i'den küçük pozisyonlardaki bilinen çıktılara bağlı olabilir [146].

5.3.2. Pozisyon Kodlama

Kelimelerin konumu ve sırası herhangi bir dilin temel parçalarıdır. Dilbilgisini ve dolayısıyla bir cümlenin gerçek anlamını tanımlarlar. RNN ve LSTM gibi sıralı algoritmaların aksine, dönüştürücülerin bir cümledeki kelimelerin göreli konumlarını yakalamak için yerleşik bir mekanizması yoktur. Bu önemlidir, çünkü kelimeler arasındaki mesafe çok önemli bağlamsal bilgiler sağlar. Konumsal kodlamanın devreye girdiği yer burasıdır.

Konumsal kodlama, model mimarisinin bir parçası değildir. Aslında ön işlemenin bir parçasıdır. Konumsal kodlama vektörü, her kelime için gömme vektörü ile aynı boyutta olacak şekilde üretilir. Hesaplamadan sonra, konumsal kodlama vektörü gömme vektörüne eklenir. Gömme vektörüne "enjekte edilen" model, algoritmanın bu uzamsal bilgiyi öğrenmesini sağlar. Şekil 5.9'da örnek bir gömme vektörüne konum bilgisi eklenmesi gösterilmektedir.



Şekil 5.9. Gömme vektörüne konum bilgisi eklenmesi.

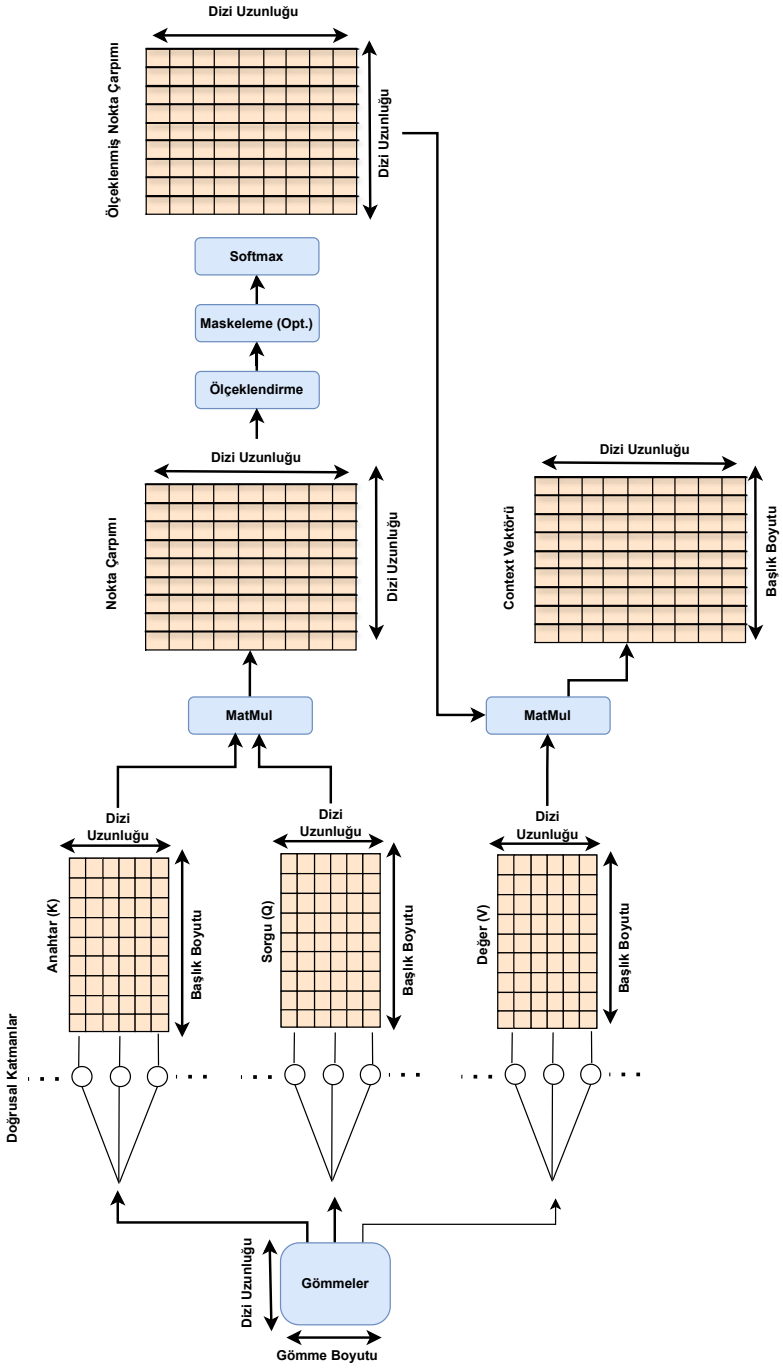
Dönüştürücü modeli bir dizideki tüm kelimeleri paralel olarak işleyebilir. Ancak dizideki konumu korumak için, kelimelerin konumu ve sırası modele açık hale getirilmelidir. Bu, konumsal kodlamalar aracılığıyla yapılır. Konumsal kodlamalar, girdi vektörü ile aynı boyutta vektörlerdir ve bir sinüs ve kosinüs fonksiyonu kullanılarak hesaplanır. Giriş vektörünün ve konumsal kodlamanın bilgilerini birleştirmek için bunlar basitçe toplanır. Pozisyon kodlama matrisi Denklem (5.1) ve Denklem (5.2) kullanılarak hesaplanır.

$$\mathrm{PE}_{(pos,2i)} = sin\left(\frac{pos}{10000^{\left(\frac{2i}{\mathrm{d}_{model}}\right)}}\right) \tag{5.1}$$

$$\mathrm{PE}_{(pos,2i+1)} = cos\left(\frac{pos}{10000^{\left(\frac{2i}{\mathrm{d}_{model}}\right)}}\right) \tag{5.2}$$

5.3.3. Dikkat Mekanizması

Bir dikkat işlevi, bir sorguyu ve bir dizi anahtar-değer çiftini, sorgunun, anahtarların, değerlerin ve çıktının tümünün vektör olduğu bir çıktıya eşlemek olarak tanımlanabilir. Çıktı, değerlerin ağırlıklı toplamı olarak hesaplanır; burada her bir değere atanan ağırlık, karşılık gelen anahtarla sorgunun bir uyumluluk işlevi tarafından [146]. Dikkat katmanının temel amacı kelimenin diğer kelimelerle, kendisi de dahil olmak üzere nasıl bir ilişkisi olduğunu belirlemektir. Şekil 5.10'daki gibi Dikkat kelime vektörleri arasında nokta çarpımı (dot product) yapacak ve diğer tüm kelimeler arasındaki kendisi de dahil olmak üzere ilişkiyi ortaya koyacaktır.

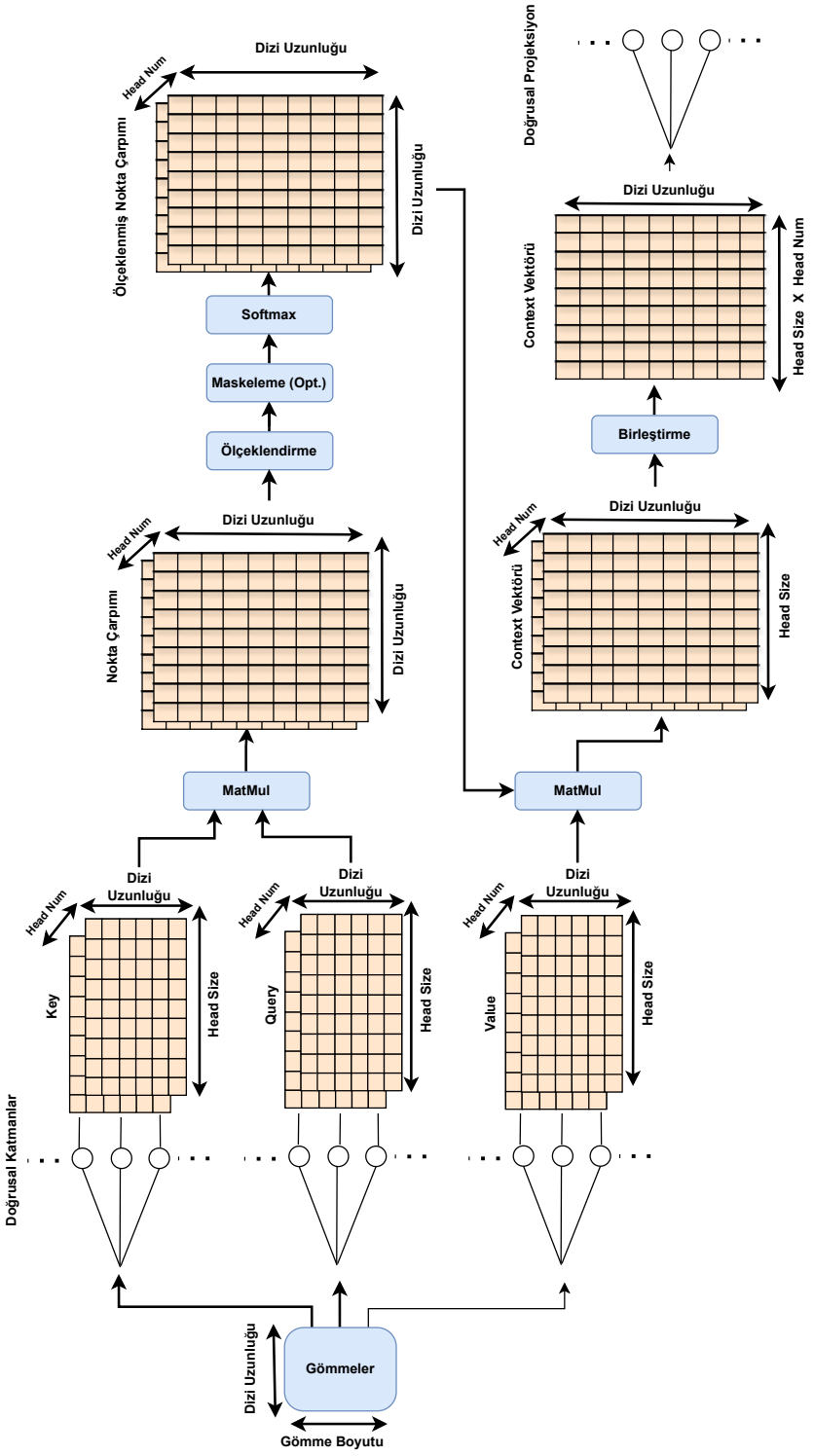


Şekil 5.10. Tek Başlı Dikkat yapısı.

Dikkat mekanizması, bir girdi sorgusuna ve öğelerin anahtarlarına dayalı olarak dinamik olarak hesaplanan ağırlıklara sahip dizi öğelerin ağırlıklı bir ortalamasını tanımlar. Amaç, birden fazla öğenin özellikleri üzerinden bir ortalama almaktır. Ancak, her bir öğeyi eşit olarak ağırlıklandırmak yerine, onları gerçek değerlerine bağlı olarak ağırlıklandırmayı hedefler.

5.3.4. Çok Başlı Dikkat

Dönüştürücü, Dikkat hesaplamalarını paralel olarak birden çok kez tekrarlar. Bunların her birine Dikkat Başlığı adı verilir. Şekil 5.11'de Çok Başlı Dikkat yapısının işleyişi gösterilmiştir.



Şekil 5.11. Çok Başlı Dikkat yapısı.

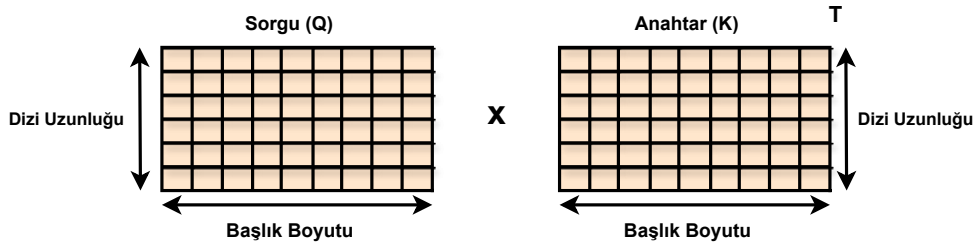
Sorgu (Q), dizide arananı yani neye dikkat edileceğini tanımlayan bir özellik vektörüdür. Anahtar (K), her girdi öğesi için, yine bir özellik vektörü olan bir anahtardır. Bu özellik vektörü kabaca elemanın ne sunduğunu veya ne zaman önemli olabileceğini ifade eder. Değer (V), her girdi elemanı için bir değer vektörü de vardır. Bu özellik vektörü, üzerinde ortalama alınacak vektördür.

Her kelime için bir K ve bir V vektörü vardır. Q, ağırlıkları belirlemek için nokta çarpımı ile tüm anahtarlarla karşılaştırılır. Son olarak, tüm kelimelerin değer vektörlerinin dikkat ağırlıkları kullanılarak ortalaması alınır.

Dikkat modülü sorgu, anahtar ve değer parametrelerini böler ve her bir bölümü bağımsız olarak ayrı bir Başlık'dan geçirir. Tüm bu benzer Dikkat hesaplamaları daha sonra nihai bir Dikkat puanı üretmek için birleştirilir. Buna Çok Başlı Dikkat (Multi-Head Attention) denir ve dönüştürücüye her kelime için çoklu ilişkileri ve nüansları kodlamak için daha fazla güç sağlar.

5.3.5. Ölçeklenmiş Nokta Çarpımı Dikkati

Ölçeklendirilmiş nokta çarpımı dikkati (Scaled Dot-Product Attention), çok başlı dikkatin ayrılmaz bir parçasıdır ve hem dönüştürücü kodlayıcısının hem de kod çözücüsünün önemli bir bileşenidir. Şekil 5.12'de ölçeklenmiş nokta çarpımı elde edilir.



Şekil 5.12. Sorgu ve Anaharın transpozunun çarpımı ile nokta çarpımının elde edilmesi.

Nokta çarpımı, iki vektör arasındaki benzerliğin bulunmasına yardım eder. Yönleri benzer olan iki vektör için nokta çarpımı büyük olurken, farklı yönleri işaret eden iki vektör için nokta çarpımı küçük olacaktır. Nokta çarpımı değeri daha sonra bir Softmax fonksiyonundan geçirilerek değerleri 1'e tamamlayan bir olasılık dağılımına ölçeklendirilir ve ayrıca yüksek değerlerin daha da yüksek ve düşük değerlerin daha da düşük olması için dağılımı keskinleştirir. Bu değerler dikkat matrisini oluşturur ve her bir sorgunun belirli bir

anahtarla ne kadar eşleştiğinin bir temsilidir. Genellikle sorgular ve anahtarlar aynı girdi belirteçleri dizisinden türetildiğinden, dikkat değerleri dizideki her bir belirtecin dizideki diğer belirteçlere ne kadar "dikkat ettiğini" gösterir. Dikkat değeri Denklem (5.3)'teki gibi Q, K ve V vektörlerine bağlı olarak hesaplanır [31].

$$Attention(Q,K,V) = Softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{5.3}$$

Şekil 5.13'teki örnek bir girdi metninin neden olduğu dikkati göstermektedir. Bu görünüm, dikkati güncellenen kelimeyle (solda) ilgilenilen kelimeyi (sağda) birleştiren çizgiler olarak görselleştirir. Renk yoğunluğu dikkat ağırlığını yansıtır; 1'e yakın ağırlıklar çok koyu çizgiler olarak gösterilirken, 0'a yakın ağırlıklar soluk çizgiler olarak görünür veya hiç görünmez. Daha koyu çizgiler daha güçlü dikkat ağırlıklarını göstermektedir.

Şekil 5.13. "hüseyin kod yazmayı çok seviyor" metni için Dikkat örneği.

5.3.6. İleri Beslemeli Ağlar

Öz dikkat katmanının çıktılarını bir sonraki kodlayıcı ya da kod çözücüye aktarmadan önce işler. Her ileri beslemeli ağ, aralarında bir ReLU fonksiyonu bulunan iki doğrusal katmandan oluşur. Basit bir ileri beslemeli sinir ağı, dikkat vektörlerini bir sonraki

kodlayıcı veya kod çözücü katman için kabul edilebilir bir forma dönüştürmek için her dikkat vektörüne uygulanır.

5.3.7. Dönüştürücü Modelinde Dikkat Uygulamaları

Kodlayıcı tarafında; giriş dizisi, giriş vektörü (embedding) ve pozisyon kodlamasına beslenir bu da giriş dizisindeki her kelime için her kelimenin anlamını ve konumunu yakalayan kodlanmış bir temsil üretir. İlk kodlayıcıdaki Öz Dikkat (Self-Attention) Sorgu, Anahtar ve Değer olmak üzere üç parametreye de beslenir ve bu da giriş dizisindeki her kelime için kodlanmış bir temsil üretir. Temsil artık her kelime için dikkat puanlarını da içerir. Bu, yığındaki tüm kodlayıcıdan geçerken, her bir Öz Dikkat modülü de kendi dikkat puanlarını her bir kelimenin temsiline ekler.

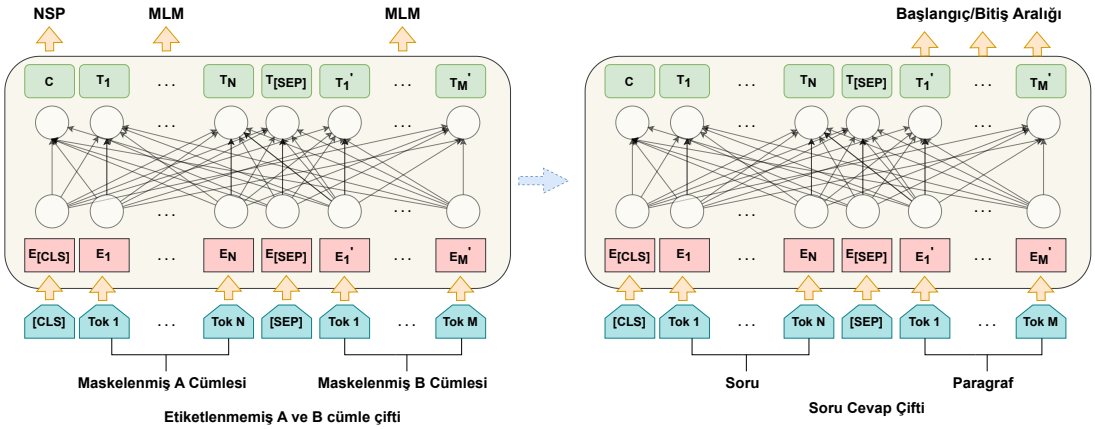
Kod çözücü yığınına gelindiğinde, hedef dizi, her bir kelimenin anlamını ve konumunu yakalayan hedef dizideki her bir kelime için kodlanmış bir temsil üreten Çıktı Gömme (Output Embedding) ve Pozisyon Kodlamasına (Position Encoding) beslenir. Bu, daha sonra hedef dizideki her kelime için kodlanmış bir temsil üreten ve her kelime için dikkat puanlarını da içeren ilk Kod Çözücüdeki Öz Dikkat'deki Sorgu, Anahtar ve Değer olmak üzere üç parametreye de beslenir. Norm katmanından geçtikten sonra ilk Kod Çözücüdeki Kodlayıcı-Kod Çözücü Dikkatindeki Sorgu parametresine beslenir.

Kodlayıcı-Kod Çözücü Dikkati hem hedef dizinin bir temsilini (Kod Çözücü Öz Dikkati'nden) hem de giriş dizisinin bir temsilini (Kodlayıcı yığınından) alır. Bu nedenle, her bir hedef dizi kelimesi için dikkat puanlarını içeren ve giriş dizisinden gelen dikkat puanlarının etkisini de yakalayan bir temsil üretir. Bu, yığındaki tüm kod çözücülerden geçerken, her bir Öz Dikkat ve her bir Kodlayıcı-Kod Çözücü Dikkati de kendi dikkat puanlarını her bir kelimenin temsiline ekler.

5.4. BERT

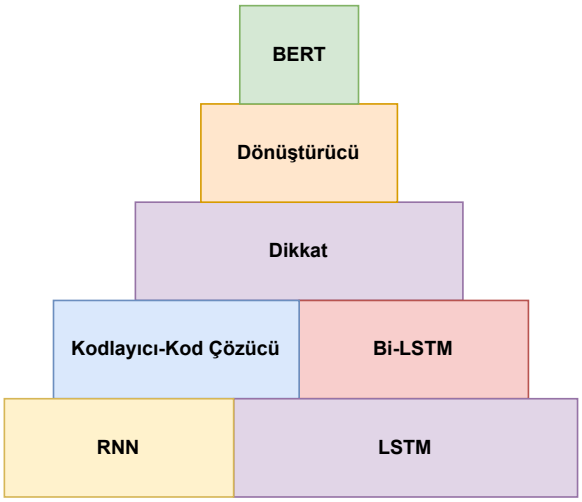
BERT (Bidirectional Encoder Representations from Transformers), önceden eğitilmiş derin çift yönlü temsilleri çıkarmak için tasarlanmış bir dil temsil modelidir [29]. Çift yönlü dönüştürücüleri kullanır [31], yani her kelime ağın her katmanında her iki taraftaki bağlamı dikkate alır. Önceden eğitilmiş BERT temsilleri, çeşitli görevlerde performans

elde etmek için kolay bir şekilde ince ayar yapılabilir [29], [147], [148]. BERT modelinin yapısı Şekil 5.14'te gösterilmiştir.



Şekil 5.14. BERT modeli ([29]).

BERT çeşitli NLP görevleri için çok sayıda etkileyici kıyaslama sonucu elde etmiştir [29]. BERT'in temel ilkelerinin yanı sıra BERT'in oluşturulmasına yönelik yapı taşlarını ve önceki teknik durumu içermektedir. Bert dağı Şekil 5.15'te gösterilmiştir.

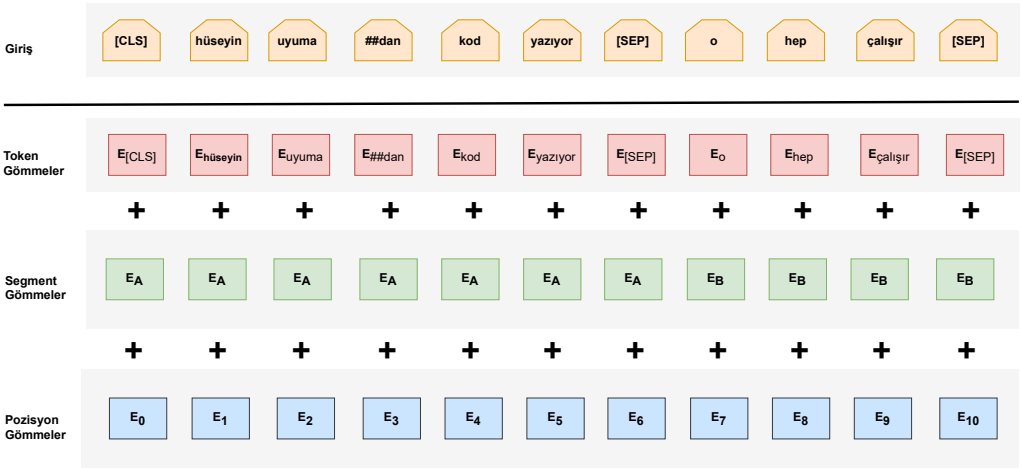


Şekil 5.15. BERT dağı olarak adlandırılan BERT'in inşasına yol açan bileşenler ve temel kavramlar.

BERT kullanılarak ince ayar tabanlı yaklaşımlar iyileştirilir. BERT, Maskeli Dil Modeli (Masked Language Model - MLM) eğitim öncesi hedefi kullanarak daha önce bahsedilen tek yönlü kısıtlamayı hafifletir. Maskelenmiş dil modeli, girdiden bazı simgeleri rastgele maskeler ve amaç, maskelenmiş sözcüğün orijinal sözcük kimliğini yalnızca içeriğine göre tahmin etmektir. Soldan sağa dil modeli ön eğitiminin aksine, MLM hedefi, temsilin

sol ve sağ bağlamı birleştirmesini sağlar, bu da derin bir çift yönlü Transformatörü önceden eğitmemizi sağlar. MLM' ye ek olarak, aynı zamanda metin çifti temsillerini birlikte önceden eğiten bir Sonraki Cümle Tahmini (Next Sentence Prediction - NSP) görevini de kullanır. BERT modelini eğitirken, MLM ve NSP, iki stratejinin birleşik kayıp fonksiyonunu en aza indirmek amacıyla birlikte eğitilir. BERT, çekirdek modele yalnızca küçük bir katman eklenerek çok çeşitli dil görevleri için kullanılabilir.

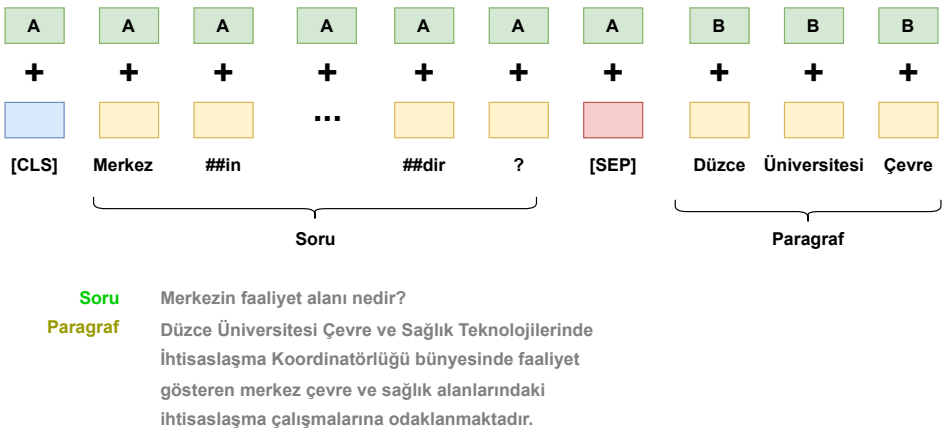
BERT büyük metin verileri üzerinde eğitildiği için, veri ön işleme işlemleri BERT kullanımı için çok önemlidir. BERT için veri ön işleme işlemler şu şekilde listelenebilir: Metin temizliği, metindeki gereksiz karakterlerin, boşlukların ve özel işaretlerin kaldırılmasını içerir. Ayrıca, büyük-küçük harf dönüşümü de yapılabilir [149].Paragraf ayrıştırma, metni paragraflara veya cümlelere ayrılabilir [150]. BERT, özellikle daha uzun metinler üzerinde çalışırken bu ayrıştırma işlemi ile daha iyi sonuçlar verir. Tokenizasyon, metnin BERT modeline uygun bir biçimde tokenize edilmelidir. Tokenizasyon, metni kelimelere veya alt birimlere ayırır [151]. Bu, her kelimenin veya alt birimin BERT modeline uygun bir şekilde temsil edilmesini sağlar. Özel jeton ekleme işlemi ise BERT özel jetonlar kullanır. Özellikle giriş ve çıkışa özel jeton eklenir. [CLS] jetonu girişi ve bu sınıflandırma belirtecini temsil eder. [SEP] jetonu ise cümle veya metinler arasındaki ayrımı gösterir ve bir sonraki cümle tahmin görevi veya soru cevaplama için önemlidir [152]. Segment göstergeleri, metinler birden fazla cümlenin birleştirilmesiyle oluşuyorsa eklenir. Bu, BERT'e metinlerin farklı bölümlerini ayırma konusunda yardımcı olur [153]. Jeton dizinleme, BERT modelinin kelime dağarcığına karşılık gelen indekslere dönüştürülür [154]. Maskeleme, metindeki bazı kelimelerin veya jetonların rastgele olarak gizlenmesini içerebilir. Bu, modelin kelimenin bağlamını anlamak için diğer kelimeleri kullanmayı öğrenmesine yardımcı olur [155]. Giriş oluşturma işleminde ise metni özel jetonlar, segment göstergeleri ve indekslerle birleştirilir ve BERT modeline beslemeye hazır hale getirilir [152]. Şekil 5.16'da örnek bir gösterime yer verilmiştir.



Şekil 5.16. BERT girdi gösterimi ([29]).

İki cümle arasındaki ilişkiyi anlamak için BERT eğitim süreci de bir sonraki cümle tahminini kullanır. Giriş olarak iki cümle alırken, BERT cümleleri özel bir [SEP] simgesiyle ayırır. Eğitim sırasında, BERT iki cümle beslenir ve ikinci cümlenin % 50'si ilk cümleden sonra gelir ve % 50'si rastgele örneklenmiş bir cümle olur. BERT'in daha sonra ikinci cümlenin rastgele olup olmadığını tahmin etmesi gerekir.

Şekil 5.17'de girdi olarak A ve B dizisi verildiğinde B'nin A'nın hemen devamı olup olmadığını tahmin eder [156]. BERT metodolojisinde, ilk olarak A'yı derlemden alır. Daha sonra B'yi ya A'nın sonlandığı yerden okur ya da B'yi derlemin farklı bir noktasından rastgele örnekler. İki diziyi ayırmak için özel bir belirteç [SEP] kullanılır. Ayrıca, girdi olarak ifade etmek için A ve B'ye özel bir belirteç [CLS] eklenir. [CLS]'nin ana fikri, B'nin derlemde A'yı takip edip etmediğini gerçekten bulmaktır [157]. Bunlar, BERT öğrendiği ve giriş katmanına beslemeden önce belirteç yerleştirmelerine eklediği iki yerleştirmedir.



Şekil 5.17. A ve B segmentleme işlemi.

Her dizinin ilk jetonu her zaman özel bir sınıflandırma simgesidir ([CLS]). Bu simgeye karşılık gelen son gizli durum, sınıflandırma görevleri için toplu sıra temsili olarak kullanılır. Cümle çiftleri tek bir dizide birlikte paketlenir. Cümleleri iki şekilde ayırt ederler. İlk olarak, onları özel bir jetonla ([SEP]) ayırırlar. İkinci olarak, her simgeye A cümlesine mi yoksa B cümlesine mi ait olduğunu belirten öğrenilmiş bir yerleştirme eklerler. Belirli bir jeton için, girdi temsili, karşılık gelen simge, segment ve konum yerleştirmelerinin toplanmasıyla oluşturulur.

Segmentasyon katıştırmaları farklı cümleleri ayırt etmek için kullanılır [158]. Konum katıştırmaları örnekteki belirteç konumlarını temsil eder. Her bir belirteç son olarak belirteç katıştırması, segmentasyon katıştırması ve konum katıştırmasının toplamı ile temsil edilir. Bu tokenizasyon ile BERT çok az sayıda alt kelime tutar. İngilizce 'de, eğitilmiş bir BERT modelinin sözcük dağarcığı boyutu yalnızca 30 522'dir. Nihai girdi gömme boyutu 768'dir [29]. BERT, Transformer mimarisinin kodlayıcılarını kullanır.

BERT-base modeli, 12 katmanlı, 768 boyutlu ve 110M parametreli bir sinir ağı yapısı bir mimaridir. Eğitim derlemleri 800 milyon ve 2,5 milyar kelime içeren BooksCorpus ve İngilizce Wikipedia derlemleridir [29]. BERTurk-base modeli her biri 768 adet gizli birime sahip 12 adet dönüştürücü bloğu içermektedir. OSCAR derlemi, Wikipedia metinleri ve Kemal Oflazer tarafından sağlanan özel bir derlem ile ön eğitilmiştir.

5.4.1. Kullanım Alanları

BERT, doğal dil işleme görevlerinde başarılı bir modeldir ve özetleme, sınıflandırma, soru cevaplama ve varlık ismi çıkarımı gibi birçok görevde üstün performans gösterir. BERT'in bağlamsal anlama kapasitesi, bu görevlerde yüksek doğruluk ve etkinlik sağlar. Her bir görevin kendine özgü gereksinimleri olsa da, BERT'in çift yönlü bağlam anlayışı, metin içindeki anlamlı ilişkileri doğru bir şekilde belirleyerek bu görevlerde başarılı olmasına katkıda bulunur [29]. Bu bölümde, BERT'in özetleme, sınıflandırma, soru cevaplama ve varlık ismi çıkarımı gibi görevlerde nasıl kullanıldığını anlatılmıştır.

5.4.1.1. *Özetleme*

Özetleme (Summarization), büyük bir metin parçasının ana fikirlerini kaybetmeden daha kısa bir versiyonunu oluşturmayı amaçlayan bir işlemdir. BERT, özellikle çekici özetleme (extractive summarization) yönteminde etkilidir. Çekici özetleme, metinden önemli cümleleri veya ifadeleri seçip çıkararak özet oluşturur. BERT'in çift yönlü bağlam anlayışı, metnin tüm kelimeleri arasındaki ilişkiyi dikkate alarak önemli kısımları belirlemesini sağlar. Bu sayede, BERT metin içinde anlamlı bağlamları çıkarır ve özeti oluşturur [159].

Örneğin, uzun bir makale düşünelim: "BERT, Google tarafından geliştirilen ve doğal dil işleme alanında devrim yaratan bir modeldir. BERT, dil modellerinin eğitiminde çift yönlü bir yaklaşım kullanır, bu da modelin metni hem ileri hem de geri yönde anlamasına olanak tanır. Bu yenilik, birçok NLP görevinde performans artışı sağlar." BERT, bu metni özetlerken, "BERT, Google tarafından geliştirilen ve dil modellerinde çift yönlü bir yaklaşım kullanan bir modeldir" gibi bir cümleyi çıkarabilir. Bu cümle, metnin ana fikrini korur ve kısa bir özet sunar.

5.4.1.2. *Sınıflandırma*

Metin sınıflandırma (Classification), metinlerin belirli kategorilere ayrılmasını içerir. BERT, bu görevde oldukça başarılıdır çünkü metnin bağlamını derinlemesine anlar ve bu bilgiyi kullanarak doğru sınıflandırmalar yapar [160]. BERT, genellikle önceden eğitilmiş bir dil modeli olarak kullanılır ve belirli bir sınıflandırma görevine uyacak şekilde ince ayar yapılır. Bu süreçte, BERT'in transformer mimarisi, her kelimenin diğer kelimelerle olan bağlamsal ilişkisini anlamasını sağlar, bu da sınıflandırma doğruluğunu artırır. Örneğin, bir duygu analizi görevinde, BERT metnin olumlu, olumsuz veya nötr olduğunu doğru bir şekilde sınıflandırabilir.

Örneğin, bir film yorumları veri seti üzerinde çalıştığımızı düşünelim. "Bu film harikaydı! Oyunculuk mükemmeldi ve hikaye çok sürükleyiciydi." şeklindeki bir yorumu BERT modeli, bağlamını analiz ederek olumlu bir yorum olarak sınıflandırabilir. Benzer şekilde, "Film çok sıkıcıydı ve oyunculuk berbattı." gibi bir yorumu da olumsuz olarak sınıflandıracaktır. BERT'in bağlamsal anlama yeteneği, bu gibi yorumların doğru bir şekilde kategorize edilmesini sağlar.

5.4.1.3. *Varlık İsmi Çıkarımı*

Varlık ismi çıkarımı (Named Entity Recognition - NER), metin içindeki özel isimleri (kişiler, yerler, organizasyonlar vb.) tanımlama işlemidir. BERT, metin içindeki kelimelerin bağlamını anlamada üstün olduğundan, NER görevlerinde de oldukça başarılıdır [161]. BERT, bir metni işlerken her kelimenin diğer kelimelerle olan bağlamsal ilişkilerini dikkate alır ve bu sayede özel isimleri doğru bir şekilde tanımlayabilir. Örneğin, "Hugging Face Inc. is a company based in New York City." cümlesinde, BERT "Hugging Face Inc." ve "New York City" ifadelerinin özel isimler olduğunu doğru bir şekilde belirleyebilir.

Örneğin, "Elon Musk founded SpaceX in 2002 in Hawthorne, California." cümlesinde, BERT modeli "Elon Musk", "SpaceX", "2002", "Hawthorne", ve "California" kelimelerini doğru bir şekilde tanımlayabilir. "Elon Musk" bir kişi adı olarak, "SpaceX" bir organizasyon adı olarak ve "Hawthorne" ile "California" yer adları olarak tanımlanır. Bu örnek, BERT'in metin içindeki farklı varlık türlerini doğru bir şekilde tanımlama yeteneğini gösterir.

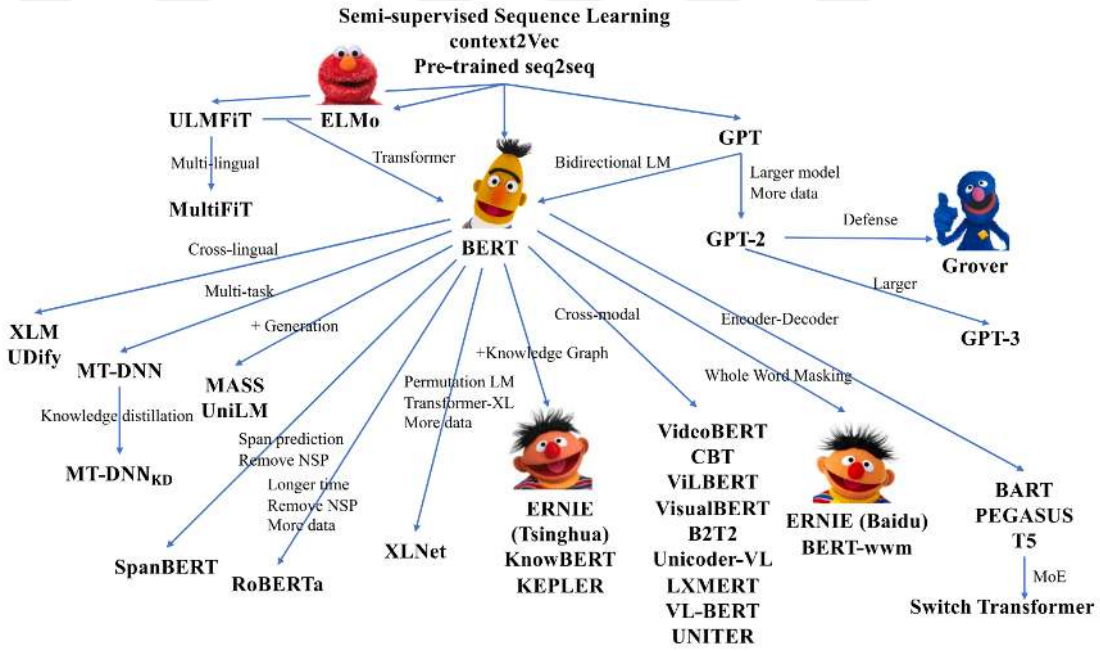
5.4.1.4. *Soru Cevaplama*

Soru cevaplama (Question Answering) sistemleri, kullanıcıların metinden spesifik sorulara cevaplar bulmasına yardımcı olur [162]. BERT, bu görevde de oldukça etkilidir. Soru-cevap görevinde, model bir soru ve ilgili bir metin parçası alır ve sorunun cevabının metnin hangi bölümünde olduğunu belirler. BERT'in çift yönlü bağlam anlayışı, sorunun ve metnin bağlamsal ilişkilerini doğru bir şekilde analiz etmesini sağlar. Bu sayede, BERT soruya uygun doğru cevabı metin içinde bulur.

Örneğin, "BERT, Google tarafından geliştirilen bir doğal dil işleme modelidir. Dil modellerinin eğitiminde çift yönlü bir yaklaşım kullanır ve metinlerin bağlamını hem ileri hem de geri yönde anlar." metni verildiğinde, "BERT kim tarafından geliştirilmiştir?" sorusuna BERT modeli, "Google tarafından" cevabını verebilir. Bu, modelin metnin belirli bölümlerini tanımlama ve anlamlandırma yeteneğini gösterir.

5.4.2. Eğitim Bilgileri

Dil modeli ön eğitiminin birçok doğal dil işleme görevini geliştirmek için etkili olduğu gösterilmiştir. Bunlar, cümleleri bütünsel olarak analiz ederek cümle arasındaki ilişkileri tahmin etmeyi amaçlayan doğal dil çıkarımı ve başka kelimelerle ifade etme gibi cümle düzeyinde görevlerin yanı sıra modellerin iyi üretmesi gereken, Adlandırılmış Varlık Tanıma ve Soru Cevaplama gibi simge düzeyinde görevleri içerir. Bir çok önceden eğitilmiş dil modeli ailesi Şekil 5.18'de gösterilmiştir. BERT'in önceden eğitilmiş modelleri iki boyutta mevcuttur: Bert-base: 12 kodlayıcı, 768 gizli boyut (hidden size), 12 öz dikkat başı, 110M parametre. Bert-large: 24 layers, 1024 gizli boyut (hidden size), 16 öz dikkat başı, 340M parametre.



Şekil 5.18. Önceden eğitilmiş dil modeli ailesi ([163]).

Bert-base, 4 Cloud TPU ve Bert-large, 16 Cloud TPU üzerinde eğitilmesi 4 gün sürmüştür. Ön eğitim için 256 dizilik bir yığın (batch) boyutu kullanılmıştır. Her dizi 512 belirteç içerir. Bu da yığın başına 128K belirteç anlamına gelir. BERT'in ön eğitimi derlemesi için 3,3 milyar kelime vardır: BooksCorpus'tan 800M ve Wikipedia'dan 2500M kelime alınmıştır. Bu da bir milyon eğitim adımı için 40 eğitim tur sayısı (epoch) ile sonuçlanmıştır. Tüm katmanlarda 0,1 nokta çarpımı (dropout) ve ReLU aktivasyonu kullanılmıştır. İnce ayar için 16 veya 32'lik yığın boyutları önerilmiştir. Öğrenme oranı (Learning rate) da ön eğitim için kullanılandan farklıdır ve göreve özeldir.

5.4.3. BERT Türevleri

Yıllar içinde BERT modelinin bir çok türevi tanıtılmıştır. Bu bölümde BERT türevlerinden olan RoBERTa, DistilBERT, ALBERT ve BERTurk ile ilgili bilgiler verilmiştir. En çok kullanılan BERT türevlerinin karşılaştırması ise Çizelge 5.5'te yer almaktadır.

Çizelge 5.5. En çok kullanılan BERT türevlerinin karşılaştırması ([164]).

Karşılaştırma	BERT	RoBERTa	DistilBERT	ALBERT
Çıkış Tarihi	11 Ekim 2018	26 Temmuz 2019	2 Ekim 2019	26 Eylül 2019
Parametreler	Base: 110M Large: 340M	Base: 125 Large: 355	Base: 66	Base: 12M Large: 18M
Katmanlar / Gizli Boyutlar / Öz Dikkat Başlıkları	Base: 12/768/12 Large: 24/1024/16	Base: 12/768/12 Large: 24/1024/16	Base: 12/768/12	Base: 12/768/12 Large: 24/1024/16
Eğitim Süresi	Base: 8xV100x12Gün Large: 280xV100x1Gün	1024xV100x1Gün (BERT'ten 4-5 kat daha fazla)	Base: 8xV100x3,5Gün	[Verilmedi] Large: 1,7 kat hızlı
Performans	Ekim 2018'de SOTA'yı geride bıraktı	GLUE'da 88,5	BERT-base'in GLUE üzerindeki performansının %97'si	GLUE' 89,4
Ön Eğitim Verileri	BooksCorpus + İngilizce Wikipedia = 16GB	BERT + CCNews + OpenWebText + Hikayeler = 160GB	BooksCorpus + İngilizce Wikipedia = 16GB	BooksCorpus + İngilizce Wikipedia = 16GB
Metod	Çift Yönlü Transformatör, MLM ve NSP	Dinamik Maskeleme NSP'siz BERT	BERT Distilasyonu	Azaltılmış parametreler ve SOP ile BERT (NSP değil)

5.4.3.1. *RoBERTa*

BERT'in yeteri kadar eğitilmediğini düşünerek, BERT'in ön-eğitim safhası üzerinde iyileştirmelerde bulunmuşlardır [71]. Mimari olarak standart Transformer'ları kullanan

RoBERTa (Robustly-optimized BERT approach), eğitim safhası için aşağıdaki dört değişikliği uygulamıştır: Eğitim veri setini artırmıştır. RoBERTa modeli, BERT'in eğitim setinden 10 kat daha geniş bir İngilizce-dil derlemi ile eğitilmiştir. BERT'in aksine, maskeleme işlemini, veri ön-işleme esnasında bir kez hesaplamak yerine; dizinin modele her geçişinde maskenin hesaplandığı Dinamik Maskeleme Deseni (Dynamic Masking Pattern) yöntemini önermiştir.NSP kayıp fonksiyonunu eğitim işleminden kaldırarak, dört farklı NLP görevi üzerinde test etmiş ve daha başarılı sonuçlar verdiğini göstermiştir. Eğitim işlemini BERT'e göre daha geniş bir yığın boyutu ve daha az adımla gerçekleştirmiş ve böyle bir eğitim şeklinin daha başarılı sonuç verdiğini göstermiştir.

5.4.3.2. *DistilBERT*

BERT bilindiği üzere, zaman ve hesaplama açısından maliyetli bir modeldir. BERT'ten türetilen modeller ise genellikle daha ağır ve geniş modeller önererek başarıyı artırma yoluna gitmektedir. Sanh ve arkadaşları [165], BERT'i sadeleştirme yoluna giderek de başarının artırılabileceğini yaptıkları çalışmada göstermişlerdir. DistiBERT (a Distilled version of BERT) ile, BERT modelinin boyutu %40 azaltılmasına karşılık, modelin dili anlama kapasitesi BERT kadar yüksek bir başarı gösterebilmiştir. Böylece, daha hızlı bir model ile neredeyse BERT ile aynı performans elde edilmiştir. Mimarinin boyutunun sadeleştirilmesi, bazı katmanların, belirteç tipi gömülmelerin, NSP görevinin ve biriktirme (pooling) işlemelerinin kaldırılması ile gerçekleştirilmiştir. Ayrıca, RoBERT gibi daha geniş yığın boyutu ve daha az adımla eğitim stratejisini uygulamışlardır.

Eğitim kaybı için; arıtma kaybı (distillation loss), MLM kaybı ve kosinüs gömülme kaybı (cosine embedding loss) fonksiyonu kullanılmıştır. Bu kayıp fonksiyonları, modelin düzgün bir şekilde öğrenilmesini ve verimli bir bilgi aktarımı gerçekleşmesini sağlamaktadır.

5.4.3.3. *ALBERT*

ALBERT (A Little BERT), RoBERTa'nın amaçladığı gibi daha başarılı ve DistilBERT'in amaçladığı gibi daha hızlı olma özelliğini tek bir modelde birleştirmek için Lan ve arkadaşları [166] tarafından önerilmiştir. Bu işlemi gerçekleştirmek için, 110 milyon parametre içeren BERT'in parametre açısından verimsiz olduğu öne sürülmüş ve parametre

sayısı düşürülerek yeni bir model önerilmiştir. ALBERT, 12 milyon parametre, 768 gizli katman ve 128 gömülme katmanından oluşmaktadır. Bu model temel olarak aşağıdaki iki yöntem ile gerçeklenmiştir: Çarpanlarına ayrılmış gömülme parametrelendirmesi (Factorized Embedding Parameterization) ile gömülme katmanı 728 katmandan 128 katmana düşürülmüş ve parametrelerin daha verimli ve az sayıda olmaları sağlanmıştır. İlk kodlayıcının parametresi öğrenilerek, katmanlar arası parametre paylaşımı (Cross-Layer Parameter Sharing) yapılmıştır.

BERT'in kullandığı veri seti üzerinde, BERT ile aynı mimari kullanılmasına rağmen, ALBERT modeli ile daha düşük hafıza tüketimi ve daha hızlı eğitim başarısı elde edilmiştir. Ayrıca, parametre paylaşımı yapılarak parametre sayısının azaltılmasının doğurduğu başarı azalmasının, kazanılan verimliliğe göre kabul edilebilir seviyede olduğu ilgili çalışmadaki sonuçlar ile gösterilmiştir.

5.4.3.4. *BERTurk*

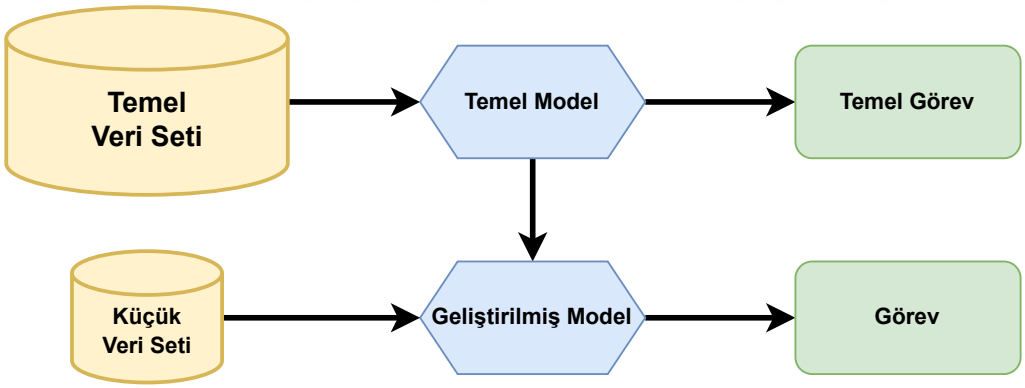
BERT modelinin açık kaynaklı olması üzerine, ana dili olan İngilizce dışında diğer diller üzerine de BERT modeli kurgulanabilmektedir. Türkçe diline özel olarak kurgulanan BERTurk modeli Stefan Schweter tarafından Türkçe OSCAR ve OPUS korpuslarını, Türkçe Wikipedia metinlerini ve Kemal Oflazer'in paylaştığı Türkçe haber metinlerini kaynak olarak kullanarak oluşturulmuştur. Bu modelin ön eğitiminde kullanılan kaynak verilerin boyutu 35GB ve toplamda 299 245 100 adet cümleden oluşmaktadır [167].

Bir dil modeli olarak BERTurk, metin sınıflandırma, isim varlık tanıma, duygu analizi ve soru cevaplama gibi çeşitli dil işleme görevlerinde kullanılabilir. Bu, onu Türkçe metinlerle çalışan dil işleme uygulamaları için değerli bir araç haline getirir. Örneğin, bir müşteri hizmetleri robotu, BERTurk kullanarak müşterinin taleplerini daha doğru bir şekilde anlayabilir ve yanıtlayabilir.

BERTurk'ün diğer bir önemli özelliği, Google'ın BERT modelinin iki yönlü doğasını korumasıdır. Bu, her bir kelimenin anlamını belirlerken, kelimenin hem önceki hem de sonraki kelimelerini aynı anda analiz eder. Bu özellik, metinlerin anlamının daha doğru bir şekilde yakalanmasını sağlar, çünkü kelimelerin anlamı genellikle onların bağlamına bağlıdır.

5.4.4. Transfer Öğrenme

Derin öğrenme, geleneksel makine öğrenmesi metotlarına göre yüksek sayıda eğitim verisine ihtiyaç duymaktadır. Transfer öğrenme özellikle büyük veri kümeleri ve yüksek hesaplama gücü gerektiren derin öğrenme modelleri için oldukça faydalıdır. Bir modelin, Şekil 5.19'da gösterildiği üzere önceden eğitilmiş olduğu bir görevin bilgisini, başka bir göreve uygulamasını ifade eder [168]. Örneğin Google tarafından geliştirilen dil sunum modeli BERT, aralıksız 4 gün boyunca 16 Bulut TPU (tensör işleme birimi)'sunda eğitilmiştir. Ayrıca MIT çalışmasına göre 200 milyon veya daha üstü parametreli bir derin sinir ağının Bulut TPU'daki eğitimi, çok üzün sürede normal bir sistemin harcadığı büyük oranda güç tüketime eşdeğerdir [169]. Her derin öğrenme ağını eğitirken ihtiyaç duyulan bolca veriyi bulamadığımız için bir makine öğrenmesi tekniği olarak transfer öğrenme geliştirilmiştir [170]. Bu yaklaşım, özellikle büyük veri setleri ve karmaşık modeller söz konusu olduğunda oldukça etkilidir. Geçmişte karşılaşılan probleme karşı üretilen çözümün bilgisi, daha sonra karşılaşılan farklı ama benzer problemlere de uygulanabilir. Örneğin daktilo yazmayı öğrenmiş birisi, ileride bilgisayar kullanmaya çalıştığında fazla zorlanmayacaktır.



Şekil 5.19. Transfer öğrenme diyagramı.

Transfer öğrenme araştırmalarının ana motivasyonu; insanların önceden öğrendiği bilgiyi, yeni problemleri zekice daha hızlı ve iyi çözmesinde kullanmasıdır. Teoride sıkça kullanılan iki kavram; kaynak ve hedeftir. Yüksek miktarlı veride çalışmış derin öğrenme modelini temsil eden kaynak, düşük miktarlı veride çalışacak modeli temsil eden ise hedeftir. Transfer öğrenme işlemi basitçe, kaynak çıktısı ağırlıklarının hedef girdisine transferidir [171]–[174]. Transfer öğrenme işlemi: hedef verisi küçük, kaynak verisi çok büyükse ve hedef görev kaynak göreve benzer ise anlam kazanmaktadır.

Transfer öğrenmenin faydaları: (1) Model, dilin genel özelliklerini önceden öğrendiği için, daha az veriyle daha hızlı ve verimli bir şekilde hedef göreve adapte olabilir. (2) Model, geniş veri kümelerinde öğrendiği bilgileri kullanarak, spesifik görevlerde yüksek performans sergiler. (3) Farklı görevler için aynı önceden eğitilmiş model kullanılabilir ve her görev için yeniden ince ayar yapılabilir.

5.4.5. Maskelenmiş Dil Modeli

Kelime dizilerini BERT'e beslemeden önce, her dizideki kelimelerin % 15'i bir [MASK] belirteci ile değiştirilir. Model daha sonra, dizideki diğer maskelenmemiş kelimeler tarafından sağlanan bağlama dayalı olarak maskelenmiş kelimelerin orijinal değerini tahmin etmeye çalışır. Kelime tahmini şunları gerektirir: (1) Kodlayıcı(Encoder) çıktısının üstüne bir sınıflandırma katmanı ekleniyor. (2) Çıktı vektörlerini gömme(Embedding) matrisiyle çarparak onları kelime boyutuna dönüştür. (3) Softmax ile kelime haznesindeki her bir kelimenin olasılığının hesaplanmasıdır. BERT loss fonksiyonu yalnızca maskelenmiş değerlerin tahminini dikkate alır ve maskelenmemiş kelimelerin tahminini göz ardı eder.

BERT tabanlı model, bir cümleyi girdi olarak alır ve cümledeki kelimelerin %15'ini maskeler ve maskelenmiş kelimelerle cümleyi modelden geçirerek, sorulan kelimeleri ve kelimelerin arkasındaki bağlamı tahmin eder. Ayrıca bu modelin faydalarından biri de tahmini daha kesin hale getirmek için cümlelerin çift yönlü temsilini öğrenmesidir. Bu model ayrıca iki maskelenmiş cümleyi kullanarak kelimeleri tahmin edebilir. İki maskelenmiş kelimeyi birleştirir ve tahmin etmeye çalışır. Böylece iki cümle birbiriyle ilişkiliyse daha kesin tahmin yapabilir.

Temel işleyiş aşağıdaki maddelerle özetlenebilir: Girdideki kelimelerin %15'ini rastgele maskeleyerek bir [MASK] belirteci ile değiştirmek. Tüm diziyi BERT dikkat tabanlı kodlayıcıdan geçirmek. Ardından dizideki maskelenmemiş diğer kelimeler tarafından sağlanan bağlama dayanarak yalnızca maskelenmiş kelimeleri tahmin etmek.

Eğitim sırasında BERT kayıp fonksiyonu, yalnızca maskelenmiş belirteçlerin tahminini dikkate alır ve maskelenmemiş olanların tahminini göz ardı eder. Bu, soldan sağa veya sağdan sola modellere göre çok daha yavaş yakınsayan bir modelle sonuçlanır.

5.4.6. Sonraki Cümle Tahmini

BERT iki cümle arasındaki ilişkiyi anlamak için eğitim sürecinde bir sonraki cümle tahminini de kullanır. Bu tür bir anlayışa sahip önceden eğitilmiş bir model soru cevaplama gibi görevler için önemlidir. Eğitim sırasında model girdi olarak cümle çiftleri alır ve ikinci cümlenin orijinal metinde de bir sonraki cümle olup olmadığını tahmin etmeyi öğrenir. Daha önce bahsedildiği gibi BERT, cümleleri özel bir [SEP] belirteci ile ayırır. Eğitim sırasında model her seferinde iki girdi cümlesi ile beslenir. %50 oranında ikinci cümle birincisinden sonra gelir. Zamanın %50'si tüm külliyattan rastgele bir cümledir. Daha sonra BERT, rastgele cümlenin ilk cümleyle bağlantısının kesileceği varsayımıyla, ikinci cümlenin rastgele olup olmadığını tahmin etmesi gerekmektedir

5.4.7. İnce Ayar

Bert modelini özelleştirmek için ince ayar (Fine-Tunning) işlemi, genellikle belirli bir görev için daha iyi performans elde etmek amacıyla gerçekleştirilir [29]. Bu süreç, temel Bert modelini belirli bir görevle ilişkilendirmek için veri hazırlığı, eğitim verisi ve etiketlerin belirlenmesi, eğitim süreci ve hiperparametrelerin ayarlanması gibi adımları içerir [31], [71], [146], [175]. Eğitim sürecinde, modelin performansını değerlendirmek ve sonuçları analiz etmek de önemlidir [27]. Bert ince ayar, doğal dil işleme alanında çeşitli uygulamalara sahip olmuştur, bu da duygu analizi, metin sınıflandırma, soru cevaplama sistemleri gibi birçok alanda kullanılabilir [176].

BERT'te ince ayar yapmak, önceden eğitilmiş BERT modelinin parametrelerini az miktarda etiketli veri ile güncelleyerek belirli bir göreve veya etki alanına uyarlama işlemidir [29], [176]. Örneğin, BERT soru cevaplama için kullanılmak istenirse, metinlerden oluşan bir veri kümesi ve bunlara karşılık gelen cevap kümesi ile ince ayar yapmanız gerekir.

Özellik tabanlı yaklaşımlarda olduğu gibi, ilki bu yönde sadece etiketlenmemiş metinden önceden eğitilmiş kelime gömme parametreleri üzerinde çalışır. Daha yakın zamanlarda, bağlamsal simge gösterimleri üreten cümle veya belge kodlayıcılar, etiketlenmemiş metinden önceden eğitilmiş ve denetimli bir aşağı akış görevi için ince ayar yapılmıştır. Bu yaklaşımların avantajı, birkaç parametrenin sıfırdan öğrenilmesinin gerekmesidir. Soldan

sağa dil modelleme ve otomatik kodlayıcı hedefleri, bu tür modellerin ön eğitimi için kullanılmıştır [29], [177].

BERT'te ince ayar yapmak, çeşitli faydalar sağlayabilir. İlk olarak, BERT'in çeşitli metin derlemleri üzerinde büyük ölçekli ön eğitimden öğrendiği zengin anlamsal ve sözdizimsel bilgiden yararlanabilir. İkinci olarak, BERT girdi metninden ilgili özellikleri otomatik olarak çıkarabildiğinden, model geliştirme ve dağıtım sürecini basitleştirir ve daha hızlı denemeye olanak sağlar. Üçüncüsü, farklı görevler ve alanlar arasında transfer öğrenimini mümkün kılar. Bu, modelin yeniden kullanımını ve uyarlanmasını kolaylaştırır ve sağlamlığını ve çok yönlülüğünü arttırır [29], [71].

5.4.8. Hiperparametre Optimizasyonu

Derin ağlarda eğitim algoritmasının ve ağ mimarisinin değiştirilebilir parametrelerine Hiperparametre adı verilir. Bu parametreler eğitim başlamadan önce seçilir. Ancak seçilen parametreler eğitim başarısını doğrudan etkiler. İşte Hiperparametre optimizasyonu eğitim başarısını maksimize hatayı veya kaybı minimize edecek şekilde parametrelerin seçilmesi sürecidir. Bu işlem için model eğitiminde ve testinde THQuADv2.0 eğitim ve test seti kullanılmıştır. Dönüştürücü modellerinin hiperparametre optimizasyonunda en yaygın kullanılan hiperparametreler şunlardır: Epok sayısı (epoch), maksimum uzunluk (maximum length), öğrenme oranı (learning rate), küme büyüklüğü (batch size), seyreltme oranı (dropout rate) ve ağırlık azaltılmasıdır (weight decay).

Epok sayısı, tüm veri setinin model tarafından kaç kez tamamlandığıdır. Az bir epok sayısı, modelin veriyi öğrenme fırsatlarını kısıtlayabilir ve eğitim süresi boyunca yetersiz öğrenme oluşabilir. Fazla bir epok sayısı ise aşırı öğrenmeye yol açabilir. Bu nedenle, epok sayısı da optimizasyon yapılırken göz önünde bulundurulur.

Küme boyutu, toplam örnek sayısı bir adımda modele verilen veri örneklerinin miktarını belirler. Daha büyük küme boyutları daha hızlı eğitim sağlayabilir, ancak daha fazla bellek kullanımına ve eğitim süresinin uzamasına neden olabilir. Daha küçük küme boyutları ise daha az bellek kullanır, ancak eğitim süresi artar. Bu nedenle, daha etkin bir model eğitimi için küme boyutunun optimizasyonu önemlidir.

Seyreltme, ağın eğitim örneklerini aşırı eğitimini (overfitting) önlemek için kullanılan bir düzenleme tekniğidir. Seyreltme oranı, ağdaki bağlantıların belli bir olasılıkla geçici olarak kapatılacağını belirler. Dropout oranı düşükse, ağ aşırı eğitime ve dolayısıyla ezberlemeye eğilimli olabilir. Dropout oranı yüksekse, modelin öğrenme kapasitesi düşebilir. Bu nedenle, dropout oranı da optimize edilmesi model performansı açısından önemlidir.

Öğrenme oranı, modelin ne kadar hızlı öğrenme yapacağını belirler. Yüksek bir öğrenme hızı, modelin daha hızlı öğrenmesine yol açabilir, ancak eğitim sürecini istikrarsızlaştırabilir. Düşük bir öğrenme hızı ise modelin yavaş öğrenmesine neden olabilir. Bu nedenle, uygun bir öğrenme hızı belirlenerek hiperparametre optimizasyonu gerçekleştirilir.

Ağırlık azaltılması, modelin öğrenme sürecinde ağırlıkların büyüklüğünü sınırlandırarak aşırı öğrenmeyi (overfitting) önlemeye yönelik bir düzenleme tekniğidir. Genellikle, modelin kayıp fonksiyonuna bir ceza terimi eklenerek uygulanır.

Maksimum metin uzunluğu, dönüştürücü modellerde metinlerin belirli bir maksimum uzunlukta olması gerekmektedir. Bu uzunluk, jeton sayısına karşılık gelir. Metinlerin uzunluğu belirlenen sınıra gelirse, metinler parçalara ayrılır veya kısaltılır. Maksimum metin uzunluğu, hem modelin bellek kullanımını hem de metinlerin kesilerek bilgi kaybı oluşmasını etkiler.

BERT modelinde kullanılan hiperparametre değerleri Çizelge 5.6'da verilmiştir. Bu hiperparametrelerin bir kısmının en iyi değerleri, eğitilmiş modelin en iyi performansı elde edilirken denemeler yapılarak bulunmuştur. Bazı parametreler için en iyi değerler, göreve özel çalışmalarda deneme yanılma veya hiperparametre optimizasyonu ile bulunur. Devlin ve arkadaşları [29], ince ayar yapılırken kullandıkları hiperparametreler kalın olarak işaretlenmiştir.

Çizelge 5.6. BERT Modeli için başarıyla ilişkilendirilen hiperparametre değerleri.

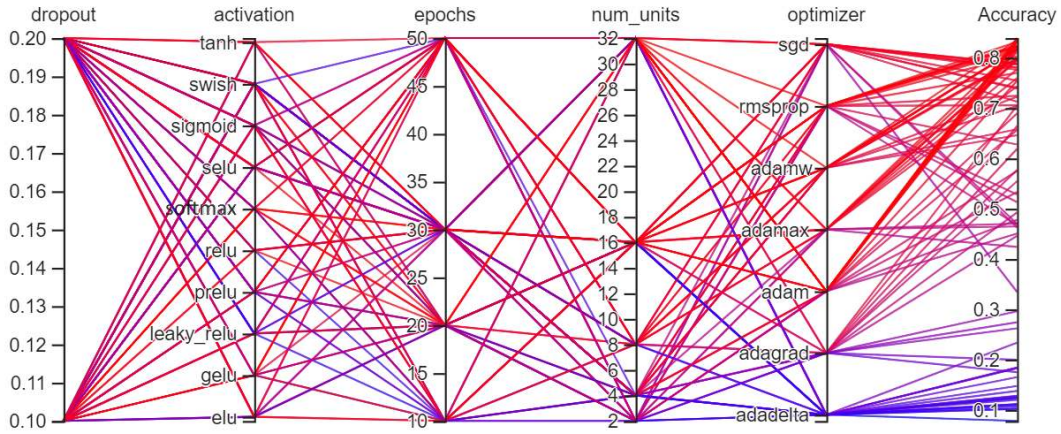
Hiperparametre	Değerler
Katman sayısı (L)	**12**, **24**, 36
Gizli katman boyutu	**768**, **1024**
Dikkat başlığı sayısı	2, 3, **12**, 16, **24**
Yığın boyutu	**16**, **32**, 64, 128
Epok sayısı	**2** , **3**, **4**, 5, 10, 15, 20
Öğrenme oranı	1e-5, **2e-5**, **3e-5**, **5e-5**, 4e-05, 5e-6
Öğrenme Algoritması	**Adam**, AdamW, SGD, RMSprop, Adamax, Adagrad, Adadelta
Seyreltme oranı	**0.1**, **0.2**, 0.3, 0.4, 0.5
Aktivasyon fonksiyonu	**ReLU**, **Sigmoid**, Tanh, Leaky ReLU, GELU, Swish, PReLU, ELU, SELU

BERT-base modeli üzerinde Grid Search algoritmasını kullanarak hiperparametre optimizasyonu yapan çeşitli makaleler ve tezler incelenmiştir. Bu çalışmalardan elde edilen bilgiler doğrultusunda Çizelge 5.7 oluşturulmuştur. Oluşturulan bu değerlere göre deneyler yapılmıştır.

Çizelge 5.7. Hiperparametre testi için kullanılan değerler.

Parametreler	Değerler
Epok Sayısı	**10**, **20**, **30**, **50**
Küme Boyutu	2, **4**, **8**, 16, 32
Seyreltme Oranı	**0.1**, 0.2
Öğrenme Oranı	1e-05, 2e-05, **3e-05**, 4e-05, 5e-05
Optimizasyon Yöntemi	adam, **adamw**, sgd, rmsprop, adagrad, adadelta, adamax
Aktivasyon Fonksiyonu	**relu** , tanh, selu, elu, leaky_relu, **sigmoid**, prelu, swish, gelu

Parametrelerin en optimumunu bulmak için Grid Search'ü grafikleştirerek sunan Tensorflow'un sunduğu Tensoarboard kullanılmıştır [178], [179]. Şekil 5.20'deki değerler elde edilmiştir.



Şekil 5.20. Grid Search hiperparametre optimizasyonu.

Bu tez çalışmasında kullanılan Çizelge 5.7'de verilen hiperparametreler değerkeri kullanılmıştır. Şekil 5.20'deki en optimum değerler Çizelge 5.7'de kalın olarak ile işaretlenmiştir. Bu bilgiler ışığında Optimizasyon algoritması olarak Adamw ve aktivasyon fonksiyonu olarak ReLu sabit tutulmuş ve diğer parametreler için yapılan denemeler Çizelge 5.9, Çizelge 5.10, Çizelge 5.11 ve Çizelge 5.12'de verilmiştir.

5.5. PERFORMANS METRİKLERİ

Oluşturulan soru cevap sistemi başlıca iki farklı değerlendirme metriği ile hesaplanmıştır. Bunlar; (1) EM, Bu metrik işaretleme yapılarak ulaşılmaya çalışılan cevap metni ile oluşturulan sistem çıktısı ile elde edilen cevap metni ile kesin eşleşen oranını ölçmektedir [36]. (2) F1 skoru, Bu metrikte ise ölçülen soru başına elde edilen cevap ve işaretlenen cevap arasındaki ortalama örtüşmedir. Bir soru üzerinden elde edilen maksimum F1 skorları, bütün sorular üzerinden ortalama sonuçları ile nihai F1 skoru hesaplanmış olur [36].

Bu ölçümlerin hesaplanması için bir test kümesi ve bir cevap tahmin kümesi gerekir. Bu iki kümenin örneklem büyüklüğü N dir. Test kümesinin i. örneği soru q_i bağlam t_i ve cevap a_i içerir. Her bir i inci örneği için bir test modeli, cevap tahmin kümesindeki w_i tahmin cevabını döndürür.

$a_i = a_{i1}a_{i2}...a_{im}$ ve $w_i = w_{i1}w_{i2}...w_{in}$ varsayıldığında, $i \in [1,N]$ burada a_{ij} ve w_{ik} sırasıyla a_i cevabındaki ve w_i tahmin cevabındaki kelimelerdir, daha sonra w_i tahmin yanıtının tam

eşleşme EM_i, kesinlik P_i, geri çağırma R_i ve F1-skoru F_{1i} sırasıyla Denklem (5.4), (5.5), (5.6) ve (5.7) ile hesaplanır.

$$\mathrm{EM}_i = \begin{cases} 0 \text{ if } \mathrm{a}_i \neq \mathrm{w}_i \\ 1 \text{ if } \mathrm{a}_i = \mathrm{w}_i \end{cases} \tag{5.4}$$

$$\mathrm{P}_i = \frac{\left|\{\mathrm{a}_{ij}\} \cap \{\mathrm{w}_{ik}\}\right|}{|\{\mathrm{w}_{ik}\}|} \tag{5.5}$$

$$\mathrm{R}_i = \frac{\left|\{\mathrm{a}_{ij}\} \cap \{\mathrm{w}_{ik}\}\right|}{|\{\mathrm{a}_{ik}\}|} \tag{5.6}$$

$$\mathrm{F}_{1i} = \frac{2 \times \mathrm{Precision}_i \times \mathrm{Recall}_i}{\mathrm{Precision}_i + \mathrm{Recall}_i} \tag{5.7}$$

Ardından, modelin F1 ve EM puanları sırasıyla Denklem (5.8) ve (5.9) ile hesaplanır. Algoritma 1 ve Algoritma 2'te de bu metriklerin sözde kod olarak kullanımı gösterilmiştir.

$$F1 = \frac{1}{N} \sum_{N} \mathrm{F}_{1i} \tag{5.8}$$

$$EM = \frac{1}{N} \sum_{N} \mathrm{EM}_i \tag{5.9}$$

Algoritma 1 F1 skoru hesaplama.

function F1SKORUHESAPLA(gercekCevap, tahmin)

$tamEslesmeSayisi \leftarrow \sum_{(g,t)\in \text{zip}(gercekCevap,tahmin)} 1 \text{ if } g == t$ ▷ Tam eşleşme sayısı hesapla

$yanlisPozitifSayisi \leftarrow \text{len}(\text{set}(tahmin) - \text{set}(gercekCevap))$ ▷ Yanlış pozitif sayısı hesapla

$yanlisPozitifSayisi \leftarrow \text{len}(\text{set}(gercekCevap) - \text{set}(tahmin))$ ▷ Yanlış negatif sayısı hesapla

if $(tamEslesmeSayisi + yanlisPozitifSayisi == 0)$
and $(tamEslesmeSayisi + yanlisPozitifSayisi == 0)$ **then**

return 0 ▷ Sıfıra bölme hatası önleme

end if

$hassasiyet \leftarrow \frac{tamEslesmeSayisi}{tamEslesmeSayisi + yanlisPozitifSayisi}$ ▷ Hassasiyet hesapla

$geriCagirma \leftarrow \frac{tamEslesmeSayisi}{tamEslesmeSayisi + yanlisPozitifSayisi}$ ▷ Geri çağırma hesapla

if $(hassasiyet + geriCagirma == 0)$ **then**

return 0 ▷ Sıfıra bölme hatası önleme

end if

$f1Skoru \leftarrow \frac{2 \cdot (hassasiyet \cdot geriCagirma)}{hassasiyet + geriCagirma}$ ▷ F1 skoru hesapla

return $f1Skoru$

end function

Algoritma 2 Tam eşleşme metriği hesaplama.

function EXACTMATCHHESAPLA(gerçekCevap, tahmin)

$normalGercekCevap \leftarrow$ KucukHarfeDonustur$(gercekCevap)$

$normalTahmin \leftarrow$ KucukHarfeDonustur$(tahmin)$

$temizlenmisGercekCevap \leftarrow$ GereksizKarakterTemizle$(normalGercekCevap)$

$temizlenmisTahmin \leftarrow$ GereksizKarakterTemizle$(normalTahmin)$

if $temizlenmisGercekCevap == temizlenmisTahmin$ **then**

$emSkoru \leftarrow 1$ ▷ Tam eşleşme durumu

else

$emSkoru \leftarrow 0$ ▷ Tam eşleşme yok

end if

return $emSkoru$

end function

function KUCUKHARFEDONUSTUR(metin)

return kucukHarfDonusturucu(metin)

end function

function GEREKSIZKARAKTERTEMIZLE(metin)

$temizMetin \leftarrow$ gereksizKarakterTemizleme$(metin)$

return $temizMetin$

end function

5.6. DOĞRULAMA METRİKLERİ

Eğitim kaybı (train loss), doğrulama kaybı (validation lost), eğitim doğruluğu (train accuracy) ve doğrulama doğruluğu (validation accuracy) derin öğrenme modellerinin eğitimi sırasında önemli doğrulama metriklerindendir. Eğitim kaybı, modelin eğitim veri kümesine ne kadar iyi uyum sağladığını gösterirken, doğrulama kaybı modelin genelleme yeteneğini ölçer. Eğitim doğruluğu, modelin eğitim veri seti üzerindeki doğru tahminlerinin oranıdır. Doğrulama doğruluğu ise modelin doğrulama veri seti üzerindeki doğru tahminlerinin oranıdır.

Eğitim Kaybı: Modelin eğitim verisi üzerindeki performansını ölçer. Bu metrik, modelin tahminleri ile gerçek değerler arasındaki hatanın bir göstergesidir. Genellikle optimizasyon algoritmasının ilerleyişini izlemek ve modelin ne kadar iyi öğrendiğini değerlendirmek için kullanılır [89].

Doğrulama kaybı, modelin doğrulama veri kümesi üzerindeki performansını ölçmek için kullanılan bir metriktir [112]. Modelin daha önce görmediği doğrulama verisi üzerindeki performansını ölçer. Doğrulama kaybı artmaya başlaması, modelin overfitting yapmaya başladığını gösterebilir.

Eğitim doğruluğu, modelin eğitim veri seti üzerindeki doğru tahmin ettiği örneklerin toplam eğitim veri setine oranıdır [88]. Bu oran, modelin eğitim sırasında öğrendiği bilgileri ne kadar iyi uygulayabildiğini gösterir.

Doğrulama doğruluğu, modelin doğrulama veri kümesi üzerinde doğruluğunu ölçmek için kullanılan bir metriktir [89].

Bu dört metrik, modelin eğitim süreci boyunca ve sonrasında performansını değerlendirmek için kritik öneme sahiptir. Eğitim ve doğrulama kayıpları modelin hatalarını, eğitim ve doğrulama doğrulukları ise modelin başarı oranını gösterir.

5.7. ÖNERİLEN MODEL VE TESTLER

Kullanıcıların sorularına etkin bir şekilde yanıt vermek amacıyla İngilizce ve Türkçe dillerinde iki farklı dil modeli geliştirilmesi amaçlanmıştır. Bu hedefe ulaşmak

için gerçekleştirilen eylemler hiperparametrelerin belirlenmesi ve bunların modellere uygulanması şeklinde belirlenmiştir. Tensorflowun hiperparametre optimizasyonaracından çıkan sonuçlara göre ve makalelerde uygulanan diğer optimal değerler kullanılarak eğitimler gerçekleştirilmiştir.

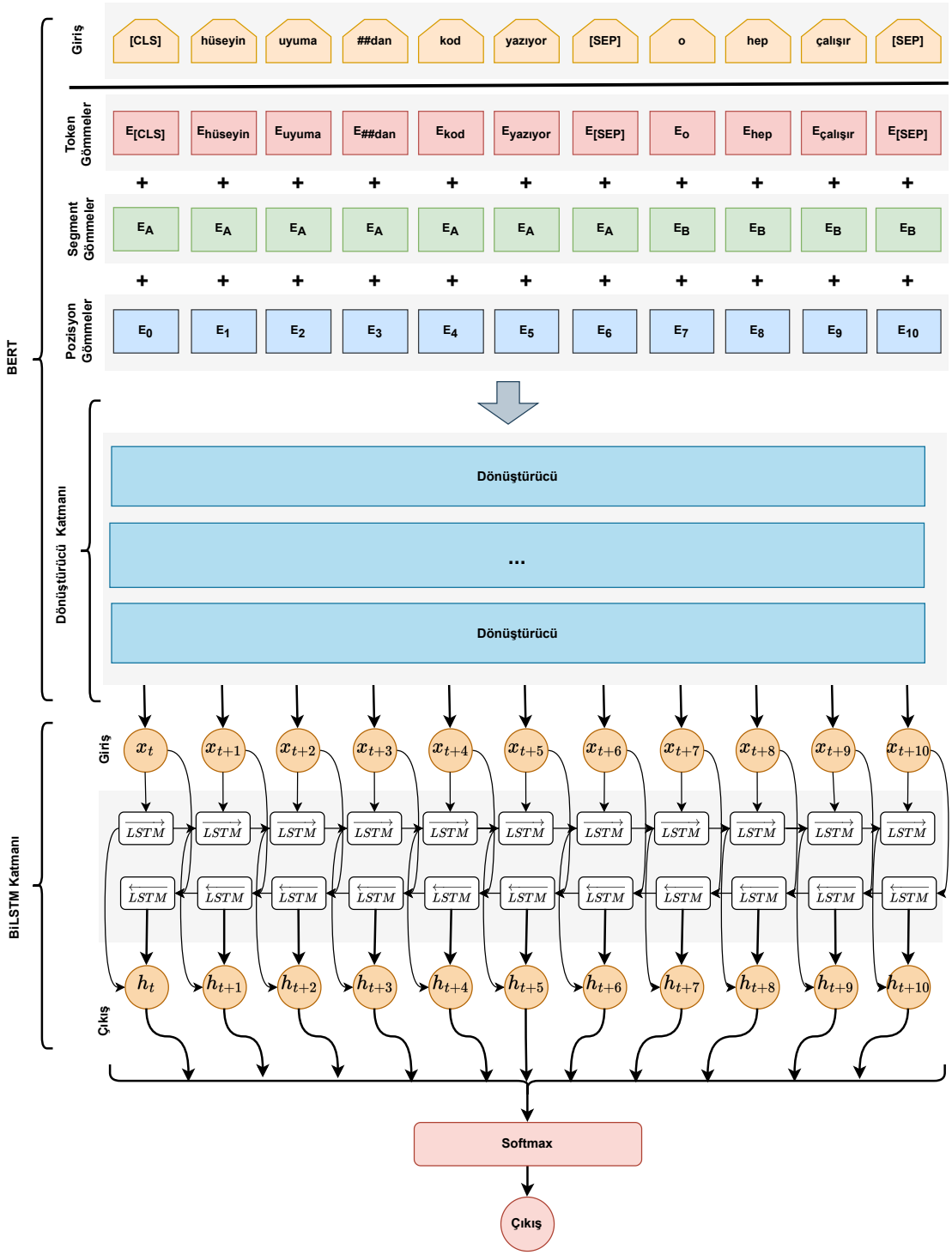
İngilizce modelde öğrenme oranı, boyut ve döngü sayısı gibi çeşitli parametreler kullanılarak testler gerçekleştirilmiştir. Model: epok sayısı 10, 20, 30, ve 50; öğrenme oranları 1e-05, 2e-05, 3e-05, 4e-05 ve 5e-05; ve küme boyutları 2, 4, 8, 16, ve 32 hiperparametre değerleri kullanılarak eğitilmiştir [29].

Türkçe model de ise veri kümesi üzerinde iyileştirme çalışmaları yapılmıştır. Türkçe yazım kuralları uygulannıştır, hatalı veriler çıkarılmıştır ve standart uygulamalara göre düzenlenmiştir. Veri kümesindeki uzun içerikler daha kolay analiz edilebilmesi için segmentlere ayrılmıştır. Düzce Üniversitesi'nin birimleri ile ilgili sorgulamalar veri setine dahil edilerek modelin genelleme yeteneği artırılmıştır. İngilizce modelde başarılı olan hiperparametreler ince ayar tekniği ile uygulanmıştır. Model: epok sayısı 10, 20, 30, ve 50; öğrenme oranları 1e-05, 2e-05, 3e-05, 4e-05 ve 5e-05; ve küme boyutları 2, 4, 8, 16, ve 32 hiperparametre değerleri kullanılarak eğitilmiştir [29].

Üçüncü model de ise Şekil 5.21'deki Türkçe modele katman olarak BiLSTM eklenmiştir. Model: epok sayısı 10, 20, 30, ve 50; öğrenme oranları 1e-05, 2e-05, 3e-05, 4e-05 ve 5e-05; ve küme boyutları 2, 4, 8, 16, ve 32 hiperparametre değerleri kullanılarak eğitilmiştir. Son olarak ise oluşturulan modellerin kullanılabileceği bir Web uygulaması geliştirilmiştir. Yapılan çalışmada kullanılan sistemin özellikleri Çizelge 5.8'de gösterilmiştir.

Çizelge 5.8. Donamım özellikleri.

Donanım	Özellik
İşletim Sistemi	Windows 11 (64-bit)
İşlemci	Intel i9 9900K 3.6 GHz 8 Çekirdek
Ekran Kartı	Nvidia GTX 3090 24GB
RAM	32GB DDR4 3600MHz
Disk	1 TB M2 SSD, 7.000 MB/sn Okuma, 5.100 MB/sn Yazma



Şekil 5.21. Önerilen BERT&BiLSTM model mimarisi.

Çeşitli parametrelere göre veri setleri test edilmiştir. Test edilen modeller İngilizce (SQuADv1.1) ve Türkçe (THQuADv2.0) olarak sunulmuştur. Yapılan farklı testlerin değerleri ve deneylerin sonuçları Çizelge 5.9, Çizelge 5.10, Çizelge 5.11 ve Çizelge 5.12'de verilmiştir.

Çizelge 5.9. BERT-Base ile SQuADv1.1 veri setinin eğitim parametreleri ve sonuçları.

Test No	Epok	Öğrenme Oranı	Küme Boyutu	Doğrulama Küme Boyutu	Test Küme Boyutu	SQuADv1.1 F1 Skoru (%)	SQuADv1.1 EM (%)
1	10	2e-05	32	32	32	63,53	56,01
2	10	2e-05	32	32	32	62,09	55,44
3	10	3e-05	32	16	32	60,81	56,48
4	10	3e-05	32	32	32	61,99	57,32
5	10	1e-05	32	32	32	63,38	56,31
6	10	4e-05	32	32	32	65,98	60,99
7	20	3e-05	2	2	2	81,16	76,71
8	20	1e-05	4	4	4	76,08	68,14
9	20	4e-05	4	4	4	83,41	73,54
10	20	3e-05	4	4	4	**88,13**	**80,74**
11	20	3e-05	8	4	2	72,72	61,10
12	20	3e-05	8	8	8	73,47	65,96
13	20	3e-05	8	8	8	75,43	70,49
14	20	3e-05	32	32	32	62,75	57,41
15	20	1e-05	32	32	32	67,39	61,75
16	30	4e-05	2	2	2	80,90	73,67
17	30	4e-05	2	2	2	81,40	76,87
18	30	4e-05	2	2	2	81,81	74,86
19	30	3e-05	2	2	2	83,13	72,74
20	30	5e-05	4	4	4	70,43	64,68
21	30	2e-05	4	4	4	83,21	75,81
22	50	5e-05	4	4	4	78,75	64,17
23	50	4e-05	4	4	4	81,81	72,97
24	50	3e-05	4	4	4	84,12	76,90
25	50	2e-05	8	8	8	70,13	65,74

Çizelge 5.9. (devam) BERT-Base ile SQuADv1.1 veri setinin eğitim parametreleri ve sonuçları.

Test No	Epok	Öğrenme Oranı	Küme Boyutu	Doğrulama Küme Boyutu	Test Küme Boyutu	SQuADv1.1 F1 Skoru (%)	SQuADv1.1 EM (%)
26	50	5e-05	8	16	8	70,88	67,50
27	50	4e-05	8	8	8	71,46	66,93

Çizelge 5.10. BERTurk-Base ile THQuADv1.0 veri setinin eğitim parametreleri ve sonuçları.

Test No	Epok	Öğrenme Oranı	Küme Boyutu	Doğrulama Küme Boyutu	Test Küme Boyutu	THQuADv1.0 F1 Skoru (%)	THQuADv1.0 EM (%)
1	10	2e-05	32	32	32	45,00	40,18
2	10	2e-05	32	32	32	46,59	40,24
3	10	3e-05	32	16	32	47,49	41,22
4	10	3e-05	32	32	32	46,86	40,27
5	10	1e-05	32	32	32	44,01	60,54
6	10	4e-05	32	32	32	44,72	41,77
7	20	3e-05	2	2	2	74,25	65,31
8	20	1e-05	4	4	4	70,33	60,94
9	20	4e-05	4	4	4	73,23	67,92
10	20	3e-05	4	4	4	69,73	61,96
11	20	3e-05	8	4	2	58,03	47,00
12	20	3e-05	8	8	8	63,03	59,34
13	20	3e-05	8	8	8	66,67	61,08
14	20	3e-05	32	32	32	47,35	41,24
15	20	1e-05	32	32	32	45,35	40,32
16	30	4e-05	2	2	2	69,62	62,14
17	30	4e-05	2	2	2	73,62	69,25
18	30	4e-05	2	2	2	72,74	63,40

Çizelge 5.10. (devam) BERTurk-Base ile THQuADv1.0 veri setinin eğitim parametreleri ve sonuçları.

Test No	Epok	Öğrenme Oranı	Küme Boyutu	Doğrulama Küme Boyutu	Test Küme Boyutu	THQuADv1.0 F1 Skoru (%)	THQuADv1.0 EM (%)
19	30	3e-05	2	2	2	76,71	64,29
20	30	5e-05	4	4	4	59,28	55,43
21	30	2e-05	4	4	4	70,90	63,18
22	50	5e-05	4	4	4	67,40	57,68
23	50	4e-05	4	4	4	66,71	62,23
24	50	3e-05	4	4	4	**79,71**	**70,10**
25	50	2e-05	8	8	8	65,10	60,90
26	50	5e-05	8	16	8	65,33	59,12
27	50	4e-05	8	8	8	63,26	54,97

Çizelge 5.11. BERTurk-Base ile THQuADv2.0 veri setinin eğitim parametreleri ve sonuçları.

Test No	Epok	Öğrenme Oranı	Küme Boyutu	Doğrulama Küme Boyutu	Test Küme Boyutu	THQuADv2.0 F1 Skoru (%)	THQuADv2.0 EM (%)
1	10	2e-05	32	32	32	59,14	49,24
2	10	2e-05	32	32	32	59,17	50,93
3	10	3e-05	32	16	32	58,51	51,07
4	10	3e-05	32	32	32	56,44	49,04
5	10	1e-05	32	32	32	57,47	51,71
6	10	4e-05	32	32	32	60,99	52,86
7	20	3e-05	2	2	2	78,91	73,52
8	20	1e-05	4	4	4	75,21	72,54
9	20	4e-05	4	4	4	82,50	73,17
10	20	3e-05	4	4	4	80,75	71,35
11	20	3e-05	8	4	2	71,73	60,47

Çizelge 5.11. (devam) BERTurk-Base ile THQuADv2.0 veri setinin eğitim parametreleri ve sonuçları.

Test No	Epok	Öğrenme Oranı	Küme Boyutu	Doğrulama Küme Boyutu	Test Küme Boyutu	THQuADv2.0 F1 Skoru (%)	THQuADv2.0 EM (%)
12	20	3e-05	8	8	8	70,59	65,45
13	20	3e-05	8	8	8	71,16	64,18
14	20	3e-05	32	32	32	59,59	50,46
15	20	1e-05	32	32	32	56,78	50,62
16	30	4e-05	2	2	2	76,62	64,05
17	30	4e-05	2	2	2	76,05	72,8
18	30	4e-05	2	2	2	76,25	70,02
19	30	3e-05	2	2	2	78,06	70,96
20	30	5e-05	4	4	4	67,77	65,29
21	30	2e-05	4	4	4	79,93	71,2
22	50	5e-05	4	4	4	76,13	68,86
23	50	4e-05	4	4	4	79,46	71,44
24	50	3e-05	4	4	4	**87,10**	**76,90**
25	50	2e-05	8	8	8	76,83	66,67
26	50	5e-05	8	16	8	68,34	63,33
27	50	4e-05	8	8	8	78,10	68,88

Çizelge 5.12. BERTurk-Base & BiLSTM ile THQuADv2.0 veri setinin eğitim parametreleri ve sonuçları.

Test No	Epok	Öğrenme Oranı	Küme Boyutu	Doğrulama Küme Boyutu	Test Küme Boyutu	THQuADv2.0 F1 Skoru (%)	THQuADv2.0 EM (%)
1	10	2e-05	32	32	32	60,32	50,22
2	10	2e-05	32	32	32	60,35	51,94
3	10	3e-05	32	16	32	59,68	52,09
4	10	3e-05	32	32	32	57,56	50,02

Çizelge 5.12. (devam) BERTurk-Base & BiLSTM ile THQuADv2.0 veri setinin eğitim parametreleri ve sonuçları.

Test No	Epok	Öğrenme Oranı	Küme Boyutu	Doğrulama Küme Boyutu	Test Küme Boyutu	THQuADv2.0 F1 Skoru (%)	THQuADv2.0 EM (%)
5	10	1e-05	32	32	32	58,61	52,74
6	10	4e-05	32	32	32	62,20	53,91
7	20	3e-05	2	2	2	80,48	74,99
8	20	1e-05	4	4	4	76,71	73,99
9	20	4e-05	4	4	4	84,15	74,63
10	20	3e-05	4	4	4	**88,84**	**78,43**
11	20	3e-05	8	4	2	73,16	61,67
12	20	3e-05	8	8	8	72,00	66,75
13	20	3e-05	8	8	8	72,58	65,46
14	20	3e-05	32	32	32	60,78	51,46
15	20	1e-05	32	32	32	57,91	51,63
16	30	4e-05	2	2	2	78,15	65,33
17	30	4e-05	2	2	2	77,57	74,25
18	30	4e-05	2	2	2	77,77	71,42
19	30	3e-05	2	2	2	79,62	72,37
20	30	5e-05	4	4	4	69,12	66,59
21	30	2e-05	4	4	4	81,52	72,62
22	50	5e-05	4	4	4	77,65	70,23
23	50	4e-05	4	4	4	81,04	72,86
24	50	3e-05	4	4	4	82,36	72,77
25	50	2e-05	8	8	8	78,36	68,00
26	50	5e-05	8	16	8	69,70	64,59
27	50	4e-05	8	8	8	79,66	70,25

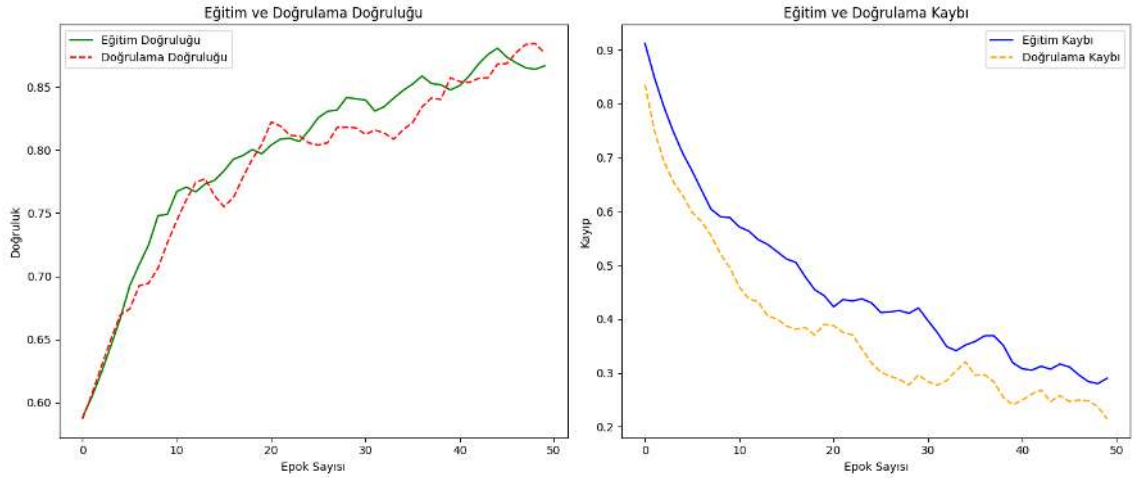
Çizelge 5.13. Sonuçların birbirleriyle karşılaştırması.

Test No	BERT-Base		BERTurk-Base		BERTurk-Base		BERTurk-Base & BiLSTM	
	SQuADv1.1		THQuADv1.0		THQuADv2.0			
	F1 Skoru (%)	EM (%)	F1 Skoru (%)	EM (%)	F1 Skoru (%)	EM (%)	F1 Skoru (%)	EM (%)
1	63,53	56,01	45,00	40,18	59,14	49,24	60,32	50,22
2	62,09	55,44	46,59	40,24	59,17	50,93	60,35	51,94
3	60,81	56,48	47,49	41,22	58,51	51,07	59,68	52,09
4	61,99	57,32	46,86	40,27	56,44	49,04	57,56	50,02
5	63,38	56,31	44,01	60,54	57,47	51,71	58,61	52,74
6	65,98	60,99	44,72	41,77	60,99	52,86	62,20	53,91
7	81,16	76,71	74,25	65,31	78,91	73,52	80,48	74,99
8	76,08	68,14	70,33	60,94	75,21	72,54	76,71	73,99
9	83,41	73,54	73,23	67,92	82,5	73,17	84,15	74,63
10	**88,13**	**80,74**	69,73	61,96	80,75	71,35	**88,84**	**78,43**
11	72,72	61,10	58,03	47,00	71,73	60,47	73,16	61,67
12	73,47	65,96	63,03	59,34	70,59	65,45	72,00	66,75
13	75,43	70,49	66,67	61,08	71,16	64,18	72,58	65,46
14	62,75	57,41	47,35	41,24	59,59	50,46	60,78	51,46
15	67,39	61,75	45,35	40,32	56,78	50,62	57,91	51,63
16	80,90	73,67	69,62	62,14	76,62	64,05	78,15	65,33
17	81,40	76,87	73,62	69,25	76,05	72,8	77,57	74,25
18	81,81	74,86	72,74	63,40	76,25	70,02	77,77	71,42
19	83,13	72,74	76,71	64,29	78,06	70,96	79,62	72,37
20	70,43	64,68	59,28	55,43	67,77	65,29	69,12	66,59

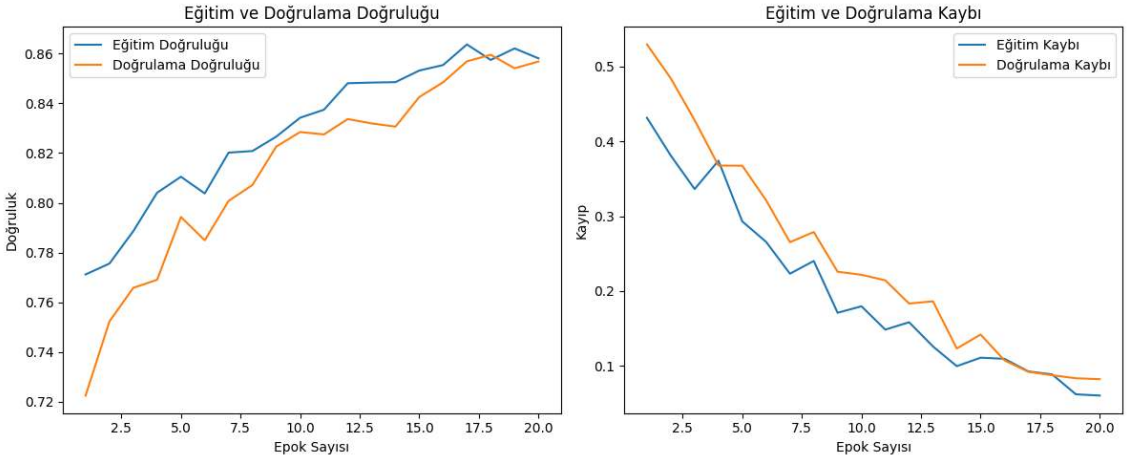
Çizelge 5.13. (devam) Sonuçların birbirleriyle karşılaştırması.

Test No	BERT-Base		BERTurk-Base		BERTurk-Base		BERTurk-Base & BiLSTM	
	SQuADv1.1		THQuADv1.0		THQuADv2.0			
	F1 Skoru (%)	EM (%)	F1 Skoru (%)	EM (%)	F1 Skoru (%)	EM (%)	F1 Skoru (%)	EM (%)
21	83,21	75,81	70,90	63,18	79,93	71,2	81,52	72,62
22	78,75	64,17	67,40	57,68	76,13	68,86	77,65	70,23
23	81,81	72,97	66,71	62,23	79,46	71,44	81,04	72,86
24	84,12	76,90	**79,71**	**70,10**	**87,10**	**76,90**	82,36	72,77
25	70,13	65,74	65,10	60,90	76,83	66,67	78,36	68,00
26	70,88	67,50	65,33	59,12	68,34	63,33	69,70	64,59
27	71,46	66,93	63,26	54,97	78,10	68,88	79,66	70,25

Çizelge 5.13'te sonuçların karşılaştırması yer almaktadır. Şekil 5.22'de THQuADv2.0 veri setinin Çizelge 5.11'deki 24 numaralı test sonuçlarının doğrulama metriklerinin grafiği gösterilmiştir. Şekil 5.23'te THQuADv2.0 veri setinin Çizelge 5.12'deki 10 numaralı test sonuçlarının doğrulama metriklerinin grafiği gösterilmiştir. Çizelge 5.14'te elde edilen başarılı sonuçların literatürdeki veriler ile karşılaştırması yer almaktadır.



Şekil 5.22. Çizelge 5.11'deki Test No 24'ün eğitim kaybı, doğrulama kaybı, eğitim doğruluğu ve doğrulama doğruluğu grafiği.



Şekil 5.23. Çizelge 5.12'deki Test No 10'un eğitim kaybı, doğrulama kaybı, eğitim doğruluğu ve doğrulama doğruluğu grafiği.

Çizelge 5.14. Sonuçların literatür ile karşılaştırılması.

No	Dil	Yıl	Yazar	EM	F1 Skor	Veri Seti	Model / Yapı
1	Eng	2021	Yang [180]	%68,5	%77,7	SQuADv1.1	Dikkat akışı kodlayıcı kod çözücü
2	Eng	2021	Japa ve Rekabdar [181]	-	%56,4	Web Sorularından oluşan bir veri kümesi	CNN ile BERT
3	Eng	2021	Lan vd. [182]	-	%82,48	Karmaşık Çince Soru ve Cevap Derlemi	LSTM ile BERT
4	Eng	2021	Sharath ve Banafsheh [183]	-	%55,4	Web Sorularından oluşan bir veri kümesi, Google Öneri API'si ve Amazon Mechanical Turk'ten gelen yanıt.	BERT, CNN ve Dikkat
5	Eng	2021	Wang ve Lu [184]	%68,7	%77,9	SQuADv1.1	Gömme, Dönüştürme ve Dikkat katmanları

Çizelge 5.14. (devam) Sonuçların literatür ile karşılaştırılması.

No	Dil	Yıl	Yazar	EM	F1 Skor	Veri Seti	Model / Yapı
6	Eng	2022	Yin [185]	%69,54	%73,21	Çin bilgi işlem görevi ve web tarayıcısı tarafından taranan finansal veri kümesi	BiLSTM ve CRF ile BERT
7	Eng	2022	Zheng vd. [186]	%46,56	%58,90	SQuAD ve NewsQA	BERT-Base
8	Eng	2021	Chen vd. [187]	%75,4	%78,8	SQuADv1.1 ve SQuADv2.0	BERT ve ALBERT-base
9	Eng	2020	Lewis [188]	%72,9	%84,8	SQuADv1.1 ve SQuADv2.0	BERT-large
10	Eng	2023	**BERT Modeli (İngilizce)**	%80,74	%88,31	SQuADv1.1	BERT-Base
11	TR	2021	Çetiner vd. [189]	%74,28	%84,82	UYAP Vergi Usul Kanunu kullanılarak 355 soru ve cevap	BERTurk-Base
12	TR	2021	Soygazi vd. [190]	%62,63	%81,19	THQuADv1.0	BERT
13	TR	2023	**BERTDuQuA [191]**	76,90	87,10	THQuADv2.0	BERTurk-Base
14	TR	2024	**BERTBiDuQuA**	78,43	88,84	THQuADv2.0	BERTurk-Base & BiLSTM

5.8. WEB UYGULAMASI

Bu bölümde modelinin son kullanıcının hizmetine sunulması ve nihayetinde Türkçe soru cevaplama için bir web uygulaması oluşturulması ile ilgili bilgiler sunulmuştur. Söz konusu sistemin oluşturulması için önce kullanılacak makine öğrenme modelinin bir web servisi hizmetiyle erişilebilir hale getirilmesi ve devamında da bu web servisi arka planda kullanacak bir web arayüzünün geliştirilmesi gerekmektedir. Aşağıdaki bölümlerde bu aşamalar ayrıntılı olarak aktarılmıştır. Bu bölümde oluşturulan web servisi ve web arayüzü ile ilgili kodlamalar GitHub sitesinde paylaşılmıştır.

5.8.1. Web Servisi

Türkçe soru cevaplama için modelin bir sunucuda çalıştırılması ve web servis hizmetiyle gelen talepleri cevaplaması kararlaştırılmıştır. Bunun için Python programlama dilinde yazılmış olan Flask web çerçevesi (web framework) kullanılmıştır. Flask ile oluşturulan web serviste sunucuya erişim yetkisi olan kişilerin gönderdiği sorguların modele sorgulatılması ve devamında modelin tahmin sonuçlarının da geri döndürülmesini sağlayacak kodlamalar gerçekleştirilmiştir. Bir sonraki adımda bu verilerin bir web arayüzüyle gerçekleştirilmesine yönelik bilgiler verilmiştir.

5.8.2. Web Arayüzü

Bu tez kapsamında oluşturulan Türkçe soru cevaplama modelinin bir web arayüzü ile son kullanıcının kullanımına sunulmasına karar verilmiştir. Bunun için React web kütüphanesiyle arayüz kodlaması gerçekleştirilmiştir. Elde edilen sonuçlardan sonra web ortamında içeriğe uygun soruya cevap verebilecek bir arayüz uygulaması tasarlanmıştır. Şekil 5.24 ve Şekil 5.25'te Türkçe ve Şekil 5.26 ve Şekil 5.27'de İngilizce modellerimiz test edilmiştir. Kullanıcıların metin ve soru girişi yapması, sisteminde web servis aracılığıyla modele sorgulatılması ve modelin verdiği sonucun kullanıcıya sunulması adımları kolaylaştırılmıştır. Sorulan soruya sistem başarılı şekilde cevap verilmiştir.

Şekil 5.24. Web sitesi Türkçe soru cevap örneği - 1.

Şekil 5.25. Web sitesi Türkçe soru cevap örneği - 2.

Şekil 5.26. Web sitesi İngilizce soru cevap örneği - 1.

Şekil 5.27. Web sitesi İngilizce soru cevap örneği - 2.

6. SONUÇLAR VE TARTIŞMA

Yapılan bu çalışma, içeriği belli bir konu üzerinde sorulan sorulara cevap verebilen bir soru cevap sistemi oluşturma amacı taşımaktadır. İlk olarak hiperparametreleri belirlemek için İngilizce BERT-base modeliyle SQuADv1.1 veri seti üzerinde çalışma yapılmıştır. Bu işlemler doğrultusunda oluşturulması hedeflenen model için hiperparametreler belirlenmiştir. Oluşturulan model literatürde yapılan çalışmalara kıyasla %88,13 F1 skoru ve %80,74 EM oranıyla başarı elde edilmiştir. Ardından Türkçe soru cevap veri seti olan THQuADv1.1 üzerinde iyileştirme çalışmaları yapılmıştır. Yapılan iyileştirmeyle hatalı olan veriler temizlenmiştir. Düzce Üniversitesi yönetmelikleri ve birimler ile ilgili soru cevap verileri ekleme yapılarak modelin veri kümesi arttırılmıştır. Eklenen soru cevaplar ile modelin genelleme yeteneğini arttırılmıştır. Soru cevap sistemi eğitimini iyileştirmek amacıyla BERTurk-base modeli kullanılmış ve hiperparametre ayarına bağlı bir yaklaşım benimsenmiştir. Hiperparametre olarak İngilizce modelde başarılı sonuç veren parametreler kullanılmıştır. Yapılan eğitim sonucunda THQuADv2.0 için BERTDuQuA modelimiz oluşturulmuştur. BERTDuQuA Türkçe dil modelleri arasında yapılan çalışmalara kıyasla %76,90 EM ve %87,10 F1 skoru ile yüksek başarım elde edilmiştir. Son olarak BERT modeline BiLSTM katmanı da eklenerek yeni bir model yapısı oluşturulmuştur. Bu model de hiperparametre olarak Türkçe modelde başarılı sonuç veren parametreler kullanılmıştır. THQuADv2.0 veri seti ile yapılan eğitimler sonucunda BERTBiDuQuA modeli elde edilmiştir. BERTBiDuQuA Türkçe dil modelleri arasında yapılan çalışmalara kıyasla %78,43 EM ve %88,84 F1 skoru ile yüksek başarım elde edilmiştir. Oluşturulan bu model bir web uygulamasına entegre edilerek sorulara başarılı bir şekilde cevap verebildiği de görülmüştür.

Başarılı bir şekilde sorulan sorulara cevap verebildiği görülse de başarının daha da artması eğitim sürecinde kullanılan veri setlerinin zenginliği ve çeşitliliğine bağlıdır. Bu veri setleri, modelin genel performansını belirleyen temel unsurlardır. Çünkü eğitim süreci boyunca model, farklı dil yapılarını ve bağlamları anlamayı bu veriler aracılığıyla öğrenir. Ayrıca, sürekli iyileştirme ve ince ayarlar ile modelin doğruluğu artırılır ve kullanıcıların

taleplerine daha uygun yanıtlar üretilmesi sağlanır. Ancak, soru cevap uygulamalarında güvenilirliği ve sürekli olarak doğru cevaplar üretebilmesi için kullanıcı geri bildirimlerine ve sistem güncellemelerine de önem verilmesi gerekmektedir.

Gelecekteki çalışmalarda veri setinin genişletilmesi, soru cevaplama ikililerinin artırılması, mimarinin güçlendirilmesi başarı oranlarını arttırabilir. Bu bağlamda, mevcut soru cevaplama sistemlerinin performansını sürekli olarak iyileştirmek ve sorunlarına daha etkili çözümler üretmek için yeni yaklaşımların tanımlanması üzerine odaklanılmalıdır. Genel veri setleri yerine belirli bir alana özgü veri setine dayalı modellerin kullanılır ise o alana özgü gelebilecek sorulara daha kesin ve tutarlı cevaplar elde edilebilir. Roberta, Distilbert, Bart, Albert gibi farklı Transformers yapıları kullanılabilir. Farklı mimariler denenebilir. Büyük dil modellerinde performans karşılaştırması yapılabilir. Ayrıca, soru cevap sistemlerinin farklı endüstriyel uygulamalarda nasıl kullanılabileceği, etik ve güvenlik konuları, kullanıcı deneyimi ve insanlarla etkileşim gibi konular, bu alandaki gelecek araştırma ve geliştirmelerin odak noktaları olabilir.

7. KAYNAKLAR

[1] D. Khurana, A. Koli, K. Khatter, ve S. Singh, "Natural language processing: State of the art, current trends and challenges," *Multimedia tools and applications*, c. 82, sayı 3, ss. 3713–3744, 2023.

[2] E. Popoff, M. Besada, J. Jansen, S. Cope, ve S. Kanters, "Aligning text mining and machine learning algorithms with best practices for study selection in systematic literature reviews," *Systematic reviews*, c. 9, ss. 1–12, 2020.

[3] A. Tyagi, "A review study of natural language processing techniques for text mining," *International Journal of Engineering Research & Technology*, c. 10, sayı 9, ss. 586–589, 2021.

[4] T. Shao, X. Kui, P. Zhang, ve H. Chen, "Collaborative learning for answer selection in question answering," *IEEE Access*, c. 7, ss. 7337–7347, 2018.

[5] A. Mishra, A. Sahay, M. anjusha Pandey, ve S. S. Routaray, "News text analysis using text summarization and sentiment analysis based on nlp," in *2023 3rd International Conference on Smart Data Intelligence (ICSMDI)*, IEEE, 2023, ss. 28–31.

[6] Z. A. Guven ve M. O. Unalir, "Improving the bert model with proposed named entity recognition method for question answering," in *2021 6th International Conference on Computer Science and Engineering (UBMK)*, IEEE, 2021, ss. 204–208.

[7] D. Mollá ve J. L. Vicedo, "Question answering in restricted domains: An overview," *Computational Linguistics*, c. 33, sayı 1, ss. 41–61, 2007.

[8] A. Singhal ve ark. "Introducing the knowledge graph: Things, not strings," *Official google blog*, c. 5, sayı 16, p. 3, 2012.

[9] F. M. Suchanek, G. Kasneci, ve G. Weikum, "Yago: A core of semantic knowledge," in *Proceedings of the 16th international conference on World Wide Web*, 2007, ss. 697–706.

[10] J. Lehmann, R. Isele, M. Jakob, ve ark. "Dbpedia–a large-scale, multilingual knowledge base extracted from wikipedia," *Semantic Web*, c. 6, sayı 2, ss. 167–195, 2015.

[11] K. Bollacker, C. Evans, P. Paritosh, T. Sturge, ve J. Taylor, "Freebase: A collaboratively created graph database for structuring human knowledge," in *Proceedings of the 2008 ACM SIGMOD international conference on Management of data*, 2008, ss. 1247–1250.

[12] G. Klyne, “Resource description framework (rdf): Concepts and abstract syntax,” *http://www.w3.org/TR/rdf-concepts/*, 2004.

[13] A. Seaborne, G. Manjunath, C. Bizer, ve ark. “Sparql/update: A language for updating rdf graphs,” *W3c member submission*, c. 15, 2008.

[14] F. Bry, P.-L. Pătrânjan, ve S. Schaffert, “Xcerpt and xchange–logic programming languages for querying and evolution on the web,” in *International Conference on Logic Programming*, Springer, 2004, ss. 450–451.

[15] B. Chardin, E. Coquery, M. Pailloux, ve J.-M. Petit, “Rql: A sql-like query language for discovering meaningful rules,” in *2014 IEEE International Conference on Data Mining Workshop*, IEEE, 2014, ss. 1203–1206.

[16] C. Unger, A. Freitas, ve P. Cimiano, “An introduction to question answering over linked data,” *Reasoning Web. Reasoning on the Web in the Big Data Era: 10th International Summer School 2014, Athens, Greece, September 8-13, 2014. Proceedings*, c. 10, ss. 100–140, 2014.

[17] L. S. Zettlemoyer ve M. Collins, “Learning context-dependent mappings from sentences to logical form,” in *Proceedings of the Association for Computational Linguistics and International Joint Conference on Natural Language Processing*, Association for Computational Linguistics, 2009, ss. 976–984.

[18] W. Cui, Y. Xiao, H. Wang, Y. Song, S.-w. Hwang, ve W. Wang, “Kbqa: Learning question answering over qa corpora and knowledge bases,” *arXiv preprint arXiv:1903.02419*, 2019.

[19] A. Bordes, N. Usunier, S. Chopra, ve J. Weston, “Large-scale simple question answering with memory networks,” *arXiv preprint arXiv:1506.02075*, 2015.

[20] A. Bordes, S. Chopra, ve J. Weston, “Question answering with subgraph embeddings,” *arXiv preprint arXiv:1406.3676*, 2014.

[21] Y. Hao, Y. Zhang, K. Liu, ve ark. “An end-to-end model for question answering over knowledge base with cross-attention combining global knowledge,” in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, c. 1, 2017, ss. 221–231.

[22] X. Yao ve B. Van Durme, “Information extraction over structured data: Question answering with freebase,” in *Proceedings of the 52nd annual meeting of the association for computational linguistics (volume 1: long papers)*, 2014, ss. 956–966.

[23] I. A. Sag, “Linguistic theory and natural language processing,” in *Natural Language and Speech: Symposium Proceedings Brussels, November 26/27, 1991*, Springer, 1991, ss. 69–83.

[24] M. Nadin, *Language as a cognitive process, volume i: Syntax: T. winograd,(addison-wesley, reading, ma, 1983); 640 pages*, 1985.

[25] A. Bordes, J. Weston, ve N. Usunier, "Open question answering with weakly supervised embedding models," in *Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2014, Nancy, France, September 15-19, 2014. Proceedings, Part I 14*, Springer, 2014, ss. 165–180.

[26] H. Sak, A. Senior, ve F. Beaufays, "Long short-term memory based recurrent neural network architectures for large vocabulary speech recognition," *arXiv preprint arXiv:1402.1128*, 2014.

[27] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever, ve ark. "Language models are unsupervised multitask learners," *OpenAI blog*, c. 1, sayı 8, p. 9, 2019.

[28] M. E. Peters, M. Neumann, M. Iyyer, ve ark. *Deep contextualized word representations*, 2018. arXiv: 1802.05365 [cs.CL].

[29] J. Devlin, M.-W. Chang, K. Lee, ve K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[30] T. Brown, B. Mann, N. Ryder, ve ark. "Language models are few-shot learners," *Advances in neural information processing systems*, c. 33, ss. 1877–1901, 2020.

[31] A. Vaswani, N. Shazeer, N. Parmar, ve ark. "Attention is all you need," *Advances in Neural Information Processing Systems*, c. 30, 2017.

[32] L. Dong, J. Mallinson, S. Reddy, ve M. Lapata, "Learning to paraphrase for question answering," *arXiv preprint arXiv:1708.06022*, 2017.

[33] Y. Chen ve F. Zulkernine, "Bird-qa: A bert-based information retrieval approach to domain specific question answering," in *2021 IEEE International Conference On Big Data (Big Data)*, IEEE, 2021, ss. 3503–3510.

[34] T.-T. Tieu, C.-N. Chau, T.-S. Nguyen, L.-M. Nguyen, ve ark. "Apply bert-based models and domain knowledge for automated legal question answering tasks at alqac 2021," in *2021 13th International Conference on Knowledge and Systems Engineering (KSE)*, IEEE, 2021, ss. 1–6.

[35] D. Gupta, S. Kumari, A. Ekbal, ve P. Bhattacharyya, "Mmqa: A multi-domain multi-lingual question-answering framework for english and hindi," in *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, 2018.

[36] Z.-J. Liu, X.-L. Wang, Q.-C. Chen, Y.-Y. Zhang, ve Y. Xiang, "A chinese question answering system based on web search," in *2014 International Conference on Machine Learning and Cybernetics*, IEEE, c. 2, 2014, ss. 816–820.

[37] Y. Uğurlu, M. Karabulut, ve İ. Mayda, "A smart virtual assistant answering questions about covid-19," in *2020 4th International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT)*, IEEE, 2020, ss. 1–6.

[38] A. M. N. Allam ve M. H. Haggag, "The question answering systems: A survey," *International Journal of Research and Reviews in Information Sciences (IJRRIS)*, c. 2, sayı 3, 2012.

[39] C.-N. Chau, T.-S. Nguyen, ve L.-M. Nguyen, "Vnlawbert: A vietnamese legal answer selection approach using bert language model," in *2020 7th NAFOSTED Conference on Information and Computer Science (NICS)*, IEEE, 2020, ss. 298–301.

[40] T. Larson, J. H. Gong, ve J. Daniel, "Providing a simple question answering system by mapping questions to questions.," Technical report, Department of Computer Science, Stanford University, Tech. Rep., 2006.

[41] P. Biswas, A. Sharan, ve R. Kumar, "Question classification using syntactic and rule based approach," in *2014 international conference on advances in computing, communications and informatics (ICACCI)*, IEEE, 2014, ss. 1033–1038.

[42] P. Ranjan ve R. C. Balabantaray, "Question answering system for factoid based question," in *2016 2nd International Conference on Contemporary Computing and Informatics (IC3I)*, IEEE, 2016, ss. 221–224.

[43] M.-Y. Day ve Y.-L. Kuo, "A study of deep learning for factoid question answering system," in *2020 IEEE 21st International Conference on Information Reuse and Integration for Data Science (IRI)*, IEEE, 2020, ss. 419–424.

[44] T. Dodiya ve S. Jain, "Question classification for medical domain question answering system," in *2016 IEEE International WIE Conference on Electrical and Computer Engineering (WIECON-ECE)*, IEEE, 2016, ss. 204–207.

[45] D. Luo, J. Su, ve S. Yu, "A bert-based approach with relation-aware attention for knowledge base question answering," in *2020 International Joint Conference on Neural Networks (IJCNN)*, IEEE, 2020, ss. 1–8.

[46] D. Singh, K. R. Suraksha, ve S. J. Nirmala, "Question answering chatbot using deep learning with nlp," in *2021 IEEE International Conference on Electronics, Computing and Communication Technologies (CONECCT)*, IEEE, 2021, ss. 1–6.

[47] N. Kanodia, K. Ahmed, ve Y. Miao, "Question answering model based conversational chatbot using bert model and google dialogflow," in *2021 31st International Telecommunication Networks and Applications Conference (ITNAC)*, IEEE, 2021, ss. 19–22.

[48] Z. Zhang, Y. Wu, J. Zhou, S. Duan, H. Zhao, ve R. Wang, "Sg-net: Syntax-guided machine reading comprehension," in *Proceedings of the AAAI conference on artificial intelligence*, c. 34, 2020, ss. 9636–9643.

[49] Z. Zhang, Y. Wu, H. Zhao, ve ark. "Semantics-aware bert for language understanding," in *Proceedings of the AAAI Conference on Artificial Intelligence*, c. 34, 2020, ss. 9628–9635.

[50] M. Hu, Y. Peng, Z. Huang, X. Qiu, F. Wei, ve M. Zhou, "Reinforced mnemonic reader for machine reading comprehension," *arXiv preprint arXiv:1705.02798*, 2017.

[51] R. Liu, W. Wei, W. Mao, ve M. Chikina, "Phase conductor on multi-layered attentions for machine comprehension," *arXiv preprint arXiv:1710.10504*, 2017.

[52] B. Pan, H. Li, Z. Zhao, B. Cao, D. Cai, ve X. He, "Memen: Multi-layer embedding with memory networks for machine comprehension," *arXiv preprint arXiv:1707.09098*, 2017.

[53] C. Xiong, V. Zhong, ve R. Socher, "Dcn+: Mixed objective and deep residual coattention for question answering," *arXiv preprint arXiv:1711.00106*, 2017.

[54] H.-Y. Huang, C. Zhu, Y. Shen, ve W. Chen, "Fusionnet: Fusing via fully-aware attention with application to machine comprehension," *arXiv preprint arXiv:1711.07341*, 2017.

[55] X. Liu, Y. Shen, K. Duh, ve J. Gao, "Stochastic answer networks for machine reading comprehension," *arXiv preprint arXiv:1712.03556*, 2017.

[56] S. Salant ve J. Berant, "Contextualized word representations for reading comprehension," *arXiv preprint arXiv:1712.03609*, 2017.

[57] M. Hu, F. Wei, Y. Peng, Z. Huang, N. Yang, ve D. Li, "Read+ verify: Machine reading comprehension with unanswerable questions," in *Proceedings of the AAAI Conference on Artificial Intelligence*, c. 33, 2019, ss. 6529–6537.

[58] C. Wang ve H. Jiang, "Explicit utilization of general knowledge in machine reading comprehension," *arXiv preprint arXiv:1809.03449*, 2018.

[59] F. Sun, L. Li, X. Qiu, ve Y. Liu, "U-net: Machine reading comprehension with unanswerable questions," *arXiv preprint arXiv:1810.06638*, 2018.

[60] O. Ram, Y. Kirstain, J. Berant, A. Globerson, ve O. Levy, "Few-shot question answering by pretraining span selection," *arXiv preprint arXiv:2101.00438*, 2021.

[61] R. Arora, P. Singh, H. Goyal, S. Singhal, ve S. Vijayvargiya, "Comparative question answering system based on natural language processing and machine learning," in *2021 International Conference on Artificial Intelligence and Smart Systems (ICAIS)*, IEEE, 2021, ss. 373–378.

[62] S. Chakraborty, E. Bisong, S. Bhatt, T. Wagner, R. Elliott, ve F. Mosconi, "Biomedbert: A pre-trained biomedical language model for qa and ir," in *Proceedings of the 28th international conference on computational linguistics*, 2020, ss. 669–679.

[63] W. Yoon, J. Lee, D. Kim, M. Jeong, ve J. Kang, "Pre-trained language model for biomedical question answering," in *Joint European conference on machine learning and knowledge discovery in databases*, Springer, 2019, ss. 727–740.

[64] D. Su, Y. Xu, G. I. Winata, ve ark. “Generalizing question answering system with pre-trained language model fine-tuning,” in *Proceedings of the 2nd workshop on machine reading for question answering*, 2019, ss. 203–211.

[65] I. Beltagy, K. Lo, ve A. Cohan, “Scibert: A pretrained language model for scientific text,” *arXiv preprint arXiv:1903.10676*, 2019.

[66] X. Zhou, B. Hu, Q. Chen, ve X. Wang, “Recurrent convolutional neural network for answer selection in community question answering,” *Neurocomputing*, c. 274, ss. 8–18, 2018.

[67] J. Martinez-Gil, B. Freudenthaler, ve A. M. Tjoa, “Multiple choice question answering in the legal domain using reinforced co-occurrence,” in *Database and Expert Systems Applications: 30th International Conference, DEXA 2019, Linz, Austria, August 26–29, 2019, Proceedings, Part I 30*, Springer, 2019, ss. 138–148.

[68] M. Esposito, E. Damiano, A. Minutolo, G. De Pietro, ve H. Fujita, “Hybrid query expansion using lexical resources and word embeddings for sentence retrieval in question answering,” *Information Sciences*, c. 514, ss. 88–105, 2020.

[69] Y.-T. Yeh ve Y.-N. Chen, “Qainfomax: Learning robust question answering system by mutual information maximization,” *arXiv preprint arXiv:1909.00215*, 2019.

[70] C. P. Carrino, M. R. Costa-Jussà, ve J. A. Fonollosa, “Automatic spanish translation of the squad dataset for multilingual question answering,” *arXiv preprint arXiv:1912.05200*, 2019.

[71] Y. Liu, M. Ott, N. Goyal, ve ark. “Roberta: A robustly optimized bert pretraining approach,” *arXiv preprint arXiv:1907.11692*, 2019.

[72] N. Abadani, J. Mozafari, A. Fatemi, M. A. Nematbakhsh, ve A. Kazemi, “Parsquad: Machine translated squad dataset for persian question answering,” in *2021 7th International Conference on Web Research (ICWR)*, IEEE, 2021, ss. 163–168.

[73] J. Chen, Z. Wei, J. Wang, ve ark. “Supplementing domain knowledge to bert with semi-structured information of documents,” *Expert Systems with Applications*, c. 235, sayı 121054, 2024.

[74] A. Saedi, A. Fatemi, ve M. A. Nematbakhsh, “Representation-centric approach for classification of consumer health questions,” *Expert Systems with Applications*, c. 229, sayı 120436, 2023.

[75] A. Gupta, A. Chadha, ve V. Tewari, “A natural language processing model on bert and yake technique for keyword extraction on sustainability reports,” *IEEE Access*, ss. 7942–7951, 2024.

[76] S. Kotstein ve C. Decker, “Restberta: A transformer-based question answering approach for semantic search in web api documentation,” *Cluster Computing*, ss. 1–27, 2024.

[77] C. Liu, X. Ji, Y. Dong, M. He, M. Yang, ve Y. Wang, "Chinese mineral question and answering system based on knowledge graph," *Expert Systems with Applications*, c. 231, sayı 120841, 2023.

[78] A. Kazemi, Z. Zojaji, M. Malverdi, ve ark. "Farsnewsqa: A deep learning-based question answering system for the persian news articles," *Information Retrieval Journal*, c. 26, sayı 1, p. 3, 2023.

[79] Z. Wang, X. Xu, X. Li, H. Li, L. Zhu, ve X. Wei, "A more robust model to answer noisy questions in kbqa," *IEEE Access*, c. 11, ss. 22 756–22 766, 2023.

[80] K. Tohma, H. I. Okur, Y. Kutlu, ve A. Sertbas, "Sentiment analysis in turkish question answering systems: An application of human-robot interaction," *IEEE Access*, 2023.

[81] J. Wang, Q. Cheng, W. Lu, Y. Dou, ve P. Li, "A term function–aware keyword citation network method for science mapping analysis," *Information Processing & Management*, c. 60, sayı 4, p. 103 405, 2023.

[82] J.-H. Kim, S.-W. Park, J.-Y. Kim, J. Park, S.-H. Jung, ve C.-B. Sim, "Roberta-coa: Roberta-based effective finetuning method using co-attention," *IEEE Access*, c. 11, ss. 120 292–120 303, 2023.

[83] E. Adalı, "Doğal dil işleme," *Türkiye Bilişim Vakfı Bilgisayar Bilimleri ve Mühendisliği Dergisi*, c. 5, sayı 2, 2012, [Online]. Erişim Adresi: https://dergipark.org.tr/tr/download/article-file/207209, Erişim: 15 Haziran 2023.

[84] S. Russel, P. Norvig, ve ark. *Artificial intelligence: A modern approach (vol. 256)*, 2013.

[85] S. Bird, "Nltk: The natural language toolkit," in *Proceedings of the COLING/ACL 2006 Interactive Presentation Sessions*, 2006, ss. 69–72.

[86] S. E. Şeker, "Doğal dil i şleme (natural language processing)," *YBS Ansiklopedi*, c. 2, sayı 4, ss. 14–31, 2015.

[87] Y. LeCun, Y. Bengio, ve G. Hinton, "Deep learning," *nature*, c. 521, sayı 7553, ss. 436–444, 2015.

[88] I. Goodfellow, Y. Bengio, ve A. Courville, *Deep learning*. MIT press, 2016.

[89] A. Krizhevsky, I. Sutskever, ve G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Advances in neural information processing systems*, c. 25, 2012.

[90] C. Szegedy, W. Liu, Y. Jia, ve ark. "Going deeper with convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, ss. 1–9.

[91] K. Simonyan ve A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[92] A. Onan, "Mining opinions from instructor evaluation reviews: A deep learning approach," *Computer Applications in Engineering Education*, c. 28, sayı 1, ss. 117–138, 2020.

[93] M. C. Castro, S. Kim, L. Barberia, ve ark. "Spatiotemporal pattern of covid-19 spread in brazil," *Science*, c. 372, sayı 6544, ss. 821–826, 2021.

[94] S. Hochreiter ve J. Schmidhuber, "Long short-term memory," *Neural Computation*, c. 9, sayı 8, ss. 1735–1780, 1997.

[95] Y. Bengio, P. Simard, ve P. Frasconi, "Learning long-term dependencies with gradient descent is difficult," *IEEE transactions on neural networks*, c. 5, sayı 2, ss. 157–166, 1994.

[96] F. A. Gers, J. Schmidhuber, ve F. Cummins, "Learning to forget: Continual prediction with lstm," *Neural Computation*, c. 12, sayı 10, ss. 2451–2471, 2000.

[97] C. Li ve X. Liu, "Research on the estimate of gas hydrate saturation based on lstm recurrent neural network," *Energies*, c. 13, sayı 6536, 2020.

[98] J. Deng, L. Cheng, ve Z. Wang, "Attention-based bilstm fused cnn with gating mechanism model for chinese long text classification," *Computer Speech & Language*, c. 68, p. 101 182, 2021.

[99] S. Liu, K. Lee, ve I. Lee, "Document-level multi-topic sentiment classification of email data with bilstm and data augmentation," *Knowledge-Based Systems*, c. 197, p. 105 918, 2020.

[100] M. Pi, N. Jin, D. Chen, ve B. Lou, "Short-term solar irradiance prediction based on multichannel lstm neural networks using edge-based iot system," *Wireless Communications and Mobile Computing*, c. 2022, p. 2 372 748, 2022.

[101] S. Sharma, S. Sharma, ve A. Athaiya, "Activation functions in neural networks," *Towards Data Sci*, c. 6, sayı 12, ss. 310–316, 2017.

[102] P. Ramachandran, B. Zoph, ve Q. V. Le, "Searching for activation functions," *arXiv preprint arXiv:1710.05941*, 2017.

[103] S. Pattanayak, *Pro deep learning with TensorFlow 2.0: A mathematical approach to advanced artificial intelligence in Python*. Springer, 2023.

[104] A. L. Maas, A. Y. Hannun, A. Y. Ng, ve ark. "Rectifier nonlinearities improve neural network acoustic models," in *Proc. icml*, Atlanta, GA, c. 30, 2013, p. 3.

[105] D.-A. Clevert, T. Unterthiner, ve S. Hochreiter, *Fast and accurate deep network learning by exponential linear units (elus)*, https://arxiv.org/abs/1511.07289, 2016.

[106] P. Ramachandran, B. Zoph, ve Q. V. Le, "Searching for activation functions," *CoRR*, c. abs/1710.05941, 2017, http://arxiv.org/abs/1710.05941.

[107] D. Hendrycks ve K. Gimpel, "Bridging nonlinearities and stochastic regularizers with gaussian error linear units," *CoRR*, c. abs/1606.08415, 2016, http://arxiv.org/abs/1606.08415.

[108] G. Klambauer, T. Unterthiner, A. Mayr, ve S. Hochreiter, "Self-normalizing neural networks," *CoRR*, c. abs/1706.02515, 2017, http://arxiv.org/abs/1706.02515.

[109] K. He, X. Zhang, S. Ren, ve J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification," *CoRR*, c. abs/1502.01852, 2015, http://arxiv.org/abs/1502.01852.

[110] P. Erdoğmuş, "Parçacik sürü opti mi zasyonu i le doğrusal olmayan denklemleri n kökleri ni n bulunmasi ve geneti k algori tma i le mukayesesi," *İleri Teknoloji Bilimleri Dergisi*, c. 5, sayı 1, 2016.

[111] D. P. Kingma ve J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[112] J. Duchi, E. Hazan, ve Y. Singer, "Adaptive subgradient methods for online learning and stochastic optimization.," *Journal of machine learning research*, c. 12, sayı 7, 2011.

[113] M. D. Zeiler, "Adadelta: An adaptive learning rate method," *arXiv preprint arXiv:1212.5701*, 2012.

[114] L. Bottou, "Large-scale machine learning with stochastic gradient descent," in *Proceedings of COMPSTAT'2010: 19th International Conference on Computational Statistics, Paris, France, August 22-27, 2010 Keynote, Invited and Contributed Papers*, Springer, 2010, ss. 177–186.

[115] M. Riedmiller ve H. Braun, "A direct adaptive method for faster backpropagation learning: The rprop algorithm," in *IEEE international conference on neural networks*, IEEE, 1993, ss. 586–591.

[116] T. Tieleman, "Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude," *COURSERA: Neural networks for machine learning*, c. 4, sayı 2, p. 26, 2012.

[117] I. Loshchilov, F. Hutter, ve ark. "Fixing weight decay regularization in adam," *arXiv preprint arXiv:1711.05101*, c. 5, 2017.

[118] S. Ruder, "An overview of gradient descent optimization algorithms," *arXiv preprint arXiv:1609.04747*, 2016.

[119] N. Shazeer ve M. Stern, "Adafactor: Adaptive learning rates with sublinear memory cost," in *International Conference on Machine Learning*, PMLR, 2018, ss. 4596–4604.

[120] D. R. Harper, A. Nandy, N. Arunachalam, C. Duan, J. P. Janet, ve H. J. Kulik, "Representations and strategies for transferable machine learning improve model

performance in chemical discovery," *The Journal of Chemical Physics*, c. 156, sayı 7, 2022.

[121] P. Liu, W. Yuan, J. Fu, Z. Jiang, H. Hayashi, ve G. Neubig, "Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing," *ACM Computing Surveys*, c. 55, sayı 9, ss. 1–35, 2023.

[122] J. Bybee, *Frequency of use and the organization of language*. Oxford University Press, 2006.

[123] M. Tezgider, B. Yıldız, ve G. Aydın, "Improving word representation by tuning word2vec parameters with deep learning model," in *2018 International Conference on Artificial Intelligence and Data Processing (IDAP)*, IEEE, 2018, ss. 1–7.

[124] C. D. Manning, *Introduction to information retrieval*. Syngress Publishing, 2008.

[125] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, ve J. Dean, "Distributed representations of words and phrases and their compositionality," c. 26, C. Burges, L. Bottou, M. Welling, Z. Ghahramani, ve K. Weinberger, Eds., ss. 3111–3119, 2013.

[126] M. S. Başarslan ve F. Kayaalp, "Sentiment analysis on social media reviews datasets with deep learning approach," *Sakarya University Journal of Computer and Information Sciences*, c. 4, sayı 1, ss. 35–49, 2021.

[127] H. P. Luhn, "A statistical approach to mechanized encoding and searching of literary information," *IBM Journal of research and development*, c. 1, sayı 4, ss. 309–317, 1957.

[128] K. Sparck Jones, "A statistical interpretation of term specificity and its application in retrieval," *Journal of documentation*, c. 28, sayı 1, ss. 11–21, 1972.

[129] R. Sjögren, K. Stridh, T. Skotare, ve J. Trygg, "Multivariate patent analysis—using chemometrics to analyze collections of chemical and pharmaceutical patents," *Journal of Chemometrics*, c. 34, sayı 1, e3041, 2020.

[130] G. Mesnil, T. Mikolov, M. Ranzato, ve Y. Bengio, "Ensemble of generative and discriminative techniques for sentiment analysis of movie reviews," *arXiv preprint arXiv:1412.5335*, 2014.

[131] M. S. Başarslan ve F. Kayaalp, "Sentiment analysis with ensemble and machine learning methods in multi-domain datasets," *Turkish Journal of Engineering*, c. 7, sayı 2, ss. 141–148, 2023.

[132] D. W. Otter, J. R. Medina, ve J. K. Kalita, "A survey of the usages of deep learning for natural language processing," *IEEE transactions on neural networks and learning systems*, c. 32, sayı 2, ss. 604–624, 2020.

[133] J. Thanaki, *Python natural language processing*. Packt Publishing Ltd, 2017.

[134] F. Pérez ve B. E. Granger, "Ipython: A system for interactive scientific computing," *Computing in Science & Engineering*, c. 9, sayı 3, ss. 21–29, 2007.

[135] T. Potluri, Y. Shi, H. Shahriar, ve ark. "Secure software development in google colab," in *2023 IEEE World AI IoT Congress (AIIoT)*, IEEE, 2023, ss. 398–402.

[136] A. Paszke, S. Gross, S. Chintala, ve ark. *Automatic differentiation in pytorch*, [Online]. Erişim Adresi: https://openreview.net/pdf?id=BJJsrmfCZ, Erişim: 29 Haziran 2024, 2017.

[137] P. L. Foalem, F. Khomh, ve H. Li, "Studying logging practice in machine learning-based applications," *Information and Software Technology*, c. 170, p. 107 450, 2024.

[138] V. Moscato, M. Postiglione, C. Sansone, ve G. Sperli, "Taughtnet: Learning multi-task biomedical named entity recognition from single-task teachers," *IEEE Journal of Biomedical and Health Informatics*, 2023.

[139] Wikipedia contributors, *Pycharm*, [Online]. Erişim Adresi: https://en.wikipedia.org/wiki/PyCharm, Erişim: 29 Haziran 2024, 2024.

[140] JetBrains, *Pycharm features*, [Online]. Erişim Adresi: https://www.jetbrains.com/pycharm/features/, Erişim: 29 Haziran 2024.

[141] Wikipedia contributors, *React (javascript library)*, [Online]. Erişim Adresi: https://en.wikipedia.org/wiki/React_(JavaScript_library), Erişim: 29 Haziran 2024, 2024.

[142] React Contributors, *React - a javascript library for building user interfaces*, [Online]. Erişim Adresi: https://reactjs.org/, Erişim: 29 Haziran 2024.

[143] Wikipedia contributors, *Flask (web framework)*, [Online]. Erişim Adresi: https://en.wikipedia.org/wiki/Flask_(web_framework), Erişim: 29 Haziran 2024, 2024.

[144] Flask Contributors, *Flask - official documentation*, [Online]. Erişim Adresi: https://flask.palletsprojects.com/, Erişim: 29 Haziran 2024.

[145] OpenAI, *Chatgpt*, [Online]. Erişim Adresi: https://www.openai.com/chatgpt, Erişim: 26 Haziran 2024, 2024.

[146] T. Wolf, L. Debut, V. Sanh, ve ark. "Huggingface's transformers: State-of-the-art natural language processing," *arXiv preprint arXiv:1910.03771*, 2019.

[147] L. Gong, D. He, Z. Li, T. Qin, L. Wang, ve T. Liu, "Efficient training of bert by progressively stacking," in *International conference on machine learning*, PMLR, 2019, ss. 2337–2346.

[148] Z. Li, X. Ding, ve T. Liu, "Story ending prediction by transferable bert," *arXiv preprint arXiv:1905.07504*, 2019.

[149] M. A. Akber, T. Ferdousi, R. Ahmed, R. Asfara, ve R. Rab, "Personality prediction based on contextual feature embedding sbert," in *2023 IEEE Region 10 Symposium (TENSYMP)*, IEEE, 2023, ss. 1–5.

[150] X. Li, H. Shu, Y. Zhai, ve Z. Lin, "A method for resume information extraction using bert-bilstm-crf," in *2021 IEEE 21st International Conference on Communication Technology (ICCT)*, IEEE, 2021, ss. 1437–1442.

[151] X. Lu, W. Liu, S. Jiang, ve C. Liu, "Multilingual bert cross-lingual transferability with pre-trained representations on tangut: A survey," in *2023 5th International Conference on Natural Language Processing (ICNLP)*, IEEE, 2023, ss. 229–234.

[152] Y. Zhao, R. Cao, J. Bai, W. Ma, ve H. Shinnou, "Determining the logical relation between two sentences by using the masked language model of bert," in *2020 International Conference on Technologies and Applications of Artificial Intelligence (TAAI)*, IEEE, 2020, ss. 228–231.

[153] Y. Nie, J. Zhao, W.-Q. Zhang, ve J. Bai, "Bert-lid: Leveraging bert to improve spoken language identification," in *2022 13th International Symposium on Chinese Spoken Language Processing (ISCSLP)*, IEEE, 2022, ss. 384–388.

[154] Y. Wang, X. Xin, ve P. Guo, "Relation extraction via attention-based cnns using token-level representations," in *2019 15th International Conference on Computational Intelligence and Security (CIS)*, IEEE, 2019, ss. 113–117.

[155] Q. T. Nguyen, T. L. Nguyen, N. H. Luong, ve Q. H. Ngo, "Fine-tuning bert for sentiment analysis of vietnamese reviews," in *2020 7th NAFOSTED conference on information and computer science (NICS)*, IEEE, 2020, ss. 302–307.

[156] J. Shan, Y. Nishihara, ve Y. Han, "Identifying reply-to relation in textual group chat using unlabeled dialogue scripts and next sentence prediction," in *2022 International Conference on Technologies and Applications of Artificial Intelligence (TAAI)*, IEEE, 2022, ss. 89–94.

[157] X. Yang ve Y. Xiao, "Named entity recognition based on bert-mbigru-crf and multi-head self-attention mechanism," in *2022 4th International Conference on Natural Language Processing (ICNLP)*, IEEE, 2022, ss. 178–183.

[158] A. Espinal, Y. Haralambous, D. Bedart, ve J. Puentes, "A format-sensitive bert-based approach to resume segmentation," in *2023 33rd Conference of Open Innovations Association (FRUCT)*, IEEE, 2023, ss. 30–37.

[159] Y. Liu ve M. Lapata, "Text summarization with pretrained encoders," *arXiv preprint arXiv:1908.08345*, 2019.

[160] C. Sun, X. Qiu, Y. Xu, ve X. Huang, "How to fine-tune bert for text classification?" In *Chinese computational linguistics: 18th China national conference, CCL 2019, Kunming, China, October 18–20, 2019, proceedings 18*, Springer, 2019, ss. 194–206.

[161] J. Y. Lee ve F. Dernoncourt, "Sequential short-text classification with recurrent and convolutional neural networks," *arXiv preprint arXiv:1603.03827*, 2016.

[162] S. Wang ve J. Jiang, "Machine comprehension using match-lstm and answer pointer," *arXiv preprint arXiv:1608.07905*, 2016.

[163] X. Han, Z. Zhang, N. Ding, ve ark. "Pre-trained models: Past, present and future," *AI Open*, c. 2, ss. 225–250, 2021.

[164] B. Kumar, *Bert variants and their differences*, [Online]. Erişim Adresi: https://360digitmg.com/blog/bert-variants-and-their-differences/, Erişim: 30 Haziran 2024, 2023.

[165] V. Sanh, L. Debut, J. Chaumond, ve T. Wolf, "Distilbert, a distilled version of bert: Smaller, faster, cheaper and lighter," *arXiv preprint arXiv:1910.01108*, 2019.

[166] Z. Lan, M. Chen, S. Goodman, K. Gimpel, P. Sharma, ve R. Soricut, "Albert: A lite bert for self-supervised learning of language representations," *arXiv preprint arXiv:1909.11942*, 2019.

[167] S. Schweter, *Berturk-bert models for turkish*, [Online]. Erişim Adresi: https://zenodo.org/record/3770924, Erişim: 25 Haziran 2023, Apr. 2020.

[168] K. Hao, "Training a single ai model can emit as much carbon as five cars in their lifetimes," *MIT technology Review*, c. 75, p. 103, 2019.

[169] E. S. Olivas, J. D. M. Guerrero, M. Martinez-Sober, J. R. Magdalena-Benedito, L. Serrano, ve ark. *Handbook of research on machine learning applications and trends: Algorithms, methods, and techniques: Algorithms, methods, and techniques*. IGI global, 2009.

[170] Y. Gao, Y. Ruan, C. Fang, ve S. Yin, "Deep learning and transfer learning models of energy consumption forecasting for a building with poor information data," *Energy and Buildings*, c. 223, p. 110 156, 2020.

[171] C. Tan, F. Sun, T. Kong, W. Zhang, C. Yang, ve C. Liu, "A survey on deep transfer learning," in *Artificial Neural Networks and Machine Learning–ICANN 2018: 27th International Conference on Artificial Neural Networks, Rhodes, Greece, October 4-7, 2018, Proceedings, Part III 27*, Springer, 2018, ss. 270–279.

[172] B. Koçer, "Transfer öğrenmede yeni yaklaşımlar," Doktora Tezi, Selçuk Üniversitesi Fen Bilimleri Enstitüsü, 2012.

[173] F. Chollet, *Transfer learning & fine-tuning*, [Online]. Erişim Adresi: https://keras.io/guides/transfer_learning/, Erişim: 03 Temmuz 2023, 2020.

[174] F. İ. Eyiokur, "Deep convolutional neural network based unconstrained ear recognition," Yüksek Lisans Tezi, İstanbul Teknik Üniversitesi Fen Bilimleri Enstitüsü, 2018.

[175] Z. Dai, Z. Yang, Y. Yang, J. Carbonell, Q. V. Le, ve R. Salakhutdinov, "Transformer-xl: Attentive language models beyond a fixed-length context," *arXiv preprint arXiv:1901.02860*, 2019.

[176] Z. Yang, Z. Dai, Y. Yang, J. Carbonell, R. R. Salakhutdinov, ve Q. V. Le, "Xlnet: Generalized autoregressive pretraining for language understanding," *Advances in neural information processing systems*, c. 32, 2019.

[177] C. Raffel, N. Shazeer, A. Roberts, ve ark. "Exploring the limits of transfer learning with a unified text-to-text transformer," *arXiv preprint arXiv:1910.10683*, 2019.

[178] A. Kamsetty. "Hyperparameter optimization for transformers: A guide." [Online]. Erişim Adresi: https://medium.com/distributed-computing-with-ray/hyperparameter-optimization-for-transformers-a-guide-c4e32c6c989b/, Erişim: 25 Haziran 2024. (2020).

[179] R. Khandelwal, *Tensorboard hyperparameter optimization*, [Online]. Erişim Adresi: https://towardsdatascience.com/tensorboard-hyperparameter-optimization-a51ef7af71f5, Erişim: 26 Mayıs 2024, 2020.

[180] Y. Yang, "BiEAF: An Bidirectional Enhanced Attention Flow Model for Question Answering Task," in *2021 2nd International Conference on Information Science and Education (ICISE-IE)*, doi: 10.1109/ICISE-IE53922.2021.00086, Chongqing, China, 2021, ss. 344–348.

[181] S. S. Japa ve B. Rekabdar, "Memory Efficient Knowledge Base Question Answering with Chatbot Framework," in *2021 IEEE Seventh International Conference on Multimedia Big Data (BigMM)*, doi: 10.1109/BigMM52142.2021.00013, Taichung, Taiwan, 2021, ss. 33–39.

[182] J. Lan, W. Liu, Y. Hu, ve J. Zhang, "Semantic Parsing and Text Generation of Complex Questions Answering Based on Deep Learning and Knowledge Graph," in *2021 4th International Conference on Robotics, Control and Automation Engineering (RCAE)*, doi: 10.1109/RCAE53607.2021.9638851, Wuhan, China, 2021, ss. 201–207.

[183] J. S. Sharath ve R. Banafsheh, "Conversational Question Answering Over Knowledge Base using Chat-Bot Framework," in *2021 IEEE 15th International Conference on Semantic Computing (ICSC)*, doi: 10.1109/ICSC50631.2021.00020, Laguna Hills, CA, USA, 2021, ss. 84–85.

[184] H. Wang ve X. Lu, "Question Answering System with Enhancing Sentence Embedding," in *2022 11th International Conference of Information and Communication Technology (ICTech)*, doi: 10.1109/ICTech55460.2022.00109, Wuhan, China, 2022, ss. 521–524.

[185] J. Yin, "Research on Question Answering System Based on BERT Model," in *2022 3rd International Conference on Computer Vision, Image and Deep Learning & International Conference on Computer Engineering and Applications (CVIDL &*

ICCEA), doi: 10.1109/CVIDLICCEA56201.2022.9824408, Changchun, China, 2022, ss. 68–71.

[186] C. Zheng, Z. Wang, ve J. He, "BERT-Based Mixed Question Answering Matching Model," in *2022 11th International Conference of Information and Communication Technology (ICTech)*, doi: 10.1109/ICTech55460.2022.00077, Wuhan, China, 2022, ss. 355–358.

[187] Y. Chen ve F. Zulkernine, "BIRD-QA: A BERT-based Information Retrieval Approach to Domain Specific Question Answering," in *2021 IEEE International Conference on Big Data (Big Data)*, doi: 10.1109/BigData52589.2021.9671523, Orlando, FL, USA, 2021, ss. 3503–3510.

[188] P. Lewis ve ark. "MLQA: Evaluating cross-lingual extractive question answering," *arXiv preprint arXiv:1910.07475*, 2019, doi: 10.48550/arXiv.1910.07475.

[189] M. Çetiner, A. Yıldırım, C. Öksüz, ve B. Onay, "Mevzuat Verisetinde Soru Cevaplama Uygulamasi Question Answering Application on Legalisation Dataset," in *2021 6th International Conference on Computer Science and Engineering (UBMK)*, doi: 10.1109/UBMK52708.2021.9558981, Ankara, Turkey, 2021, ss. 603–607.

[190] F. Soygazi, O. Çiftçi, U. Kök, ve S. Cengiz, "THQuAD: Turkish Historic Question Answering Dataset for Reading Comprehension," in *2021 6th International Conference on Computer Science and Engineering (UBMK)*, doi: 10.1109/UBMK52708.2021.9559013, Ankara, Turkey, 2021, ss. 215–220.

[191] H. A. Ardaç ve P. Erdoğmuş, "Question answering system with text mining and deep networks," *Evolving Systems*, ss. 1–13, 2024.

8. EKLER

8.1. EK 1. Düzce Üniversitesi Soru-Cevap Seti Örnekleri

Çizelge 8.1. Geleneksel ve tamamlayıcı tıp (GTT) uygulama ve araştırma merkezi yönetmeliği.

Durum	Veri
İçerik 1	Düzce Üniversitesi Çevre ve Sağlık Teknolojilerinde İhtisaslaşma Koordinatörlüğü bünyesinde faaliyet gösteren Merkez, çevre ve sağlık alanlarındaki ihtisaslaşma çalışmalarına odaklanmaktadır. Merkezin temel amacı, modern tıp uygulamalarını destekleyici nitelikteki geleneksel ve tamamlayıcı tıp yöntemlerini inceleyerek, bu yöntemlerin teşhis, tedavi ve korunma amaçlı kullanımının etkilerini belirlemek ve geliştirmektir. Bu çerçevede, Ar-Ge çalışmaları yoluyla, geleneksel ve tamamlayıcı tıp yöntemlerinin etki mekanizmalarını anlamak ve Sağlık Bakanlığı mevzuatına uygun ürünler üreterek döner sermaye işletmesi kapsamında bu ürünlerin satışını gerçekleştirmek hedeflenmektedir. Ayrıca, Merkez, sağlık alanında öğrenim gören öğrenci ve mezunlar başta olmak üzere, bölge insanına yönelik sertifikalı eğitim-öğretim faaliyetleri düzenlemekte ve geleneksel ve tamamlayıcı tıp uygulamaları için ürün geliştirme çalışmalarında bulunmaktadır.
Soru 1	Merkezin hedefleri nelerdir?
Cevap 1	1: Ar-Ge çalışmaları 2: Ar-Ge çalışmaları, ürün satışı 3: Ar-Ge çalışmaları, ürün satışı, sertifikalı eğitimler, ürün geliştirme.
İçerik 2	Merkezin faaliyet alanları, Sağlık Bakanlığı mevzuatında izin verilen geleneksel ve tamamlayıcı tıp yöntemlerini araştırmak, uygulamak ve ilgili eğitimleri vermekten tutun, bu yöntemlere dair bilimsel çalışmalar yapmak için laboratuvarlar kurmaya, ulusal ve uluslararası düzeyde bilimsel araştırma projeleri hazırlamaktan, geleneksel ve tamamlayıcı tıp uygulamalarını bölgeye yaymak gibi geniş bir yelpazeyi kapsamaktadır. Merkezin yönetim organları, Müdür, Yönetim Kurulu ve Danışma Kurulu olarak belirlenmiştir. Müdür, Üniversitenin kadrolu öğretim üyeleri arasından Rektör tarafından üç yıl süreyle görevlendirilir ve Merkezin faaliyetlerini temsil eder. Yönetim Kurulu, Müdür ile birlikte çalışarak Merkezin faaliyetlerini gözden geçirir ve kararlar alır. Danışma Kurulu ise Müdür ve müdür yardımcılarıyla birlikte faaliyetleri değerlendirir, önerilerde bulunur ve uzman kişilerden oluşan bir heyet olarak Merkeze destek sağlar. Bu organlar arasındaki etkileşim, Merkezin çevre ve sağlık alanındaki ihtisaslaşma çalışmalarının etkin bir şekilde yürütülmesine ve geliştirilmesine olanak tanır. Her bir organ, belirlenen hedeflere yönelik olarak işbirliği içinde çalışarak Merkezin amacına katkıda bulunmayı hedefler.
Soru 1	Merkezin faaliyet alanları nelerdir?
Cevap 1	Geleneksel ve tamamlayıcı tıp yöntemleri
Soru 2	Merkezin yönetim organları nelerdir?
Cevap 2	Müdür, Yönetim Kurulu, Danışma Kurulu
Soru 3	Müdür nasıl belirlenir ve görev süresi nedir?

Çizelge 8.1. (devam) Geleneksel ve tamamlayıcı tıp (GTT) uygulama ve araştırma merkezi yönetmeliği.

Durum	Veri
Cevap 3	Rektör tarafından üç yıl süreyle görevlendirilir
Soru 4	Yönetim Kurulu'nun görevi nedir?
Cevap 4	Faaliyetleri gözden geçirir ve kararlar alır
Soru 5	Danışma Kurulu ne işe yarar?
Cevap 5	1: Faaliyetleri değerlendirir 2: Faaliyetleri değerlendirir, önerilerde bulunur 3: Faaliyetleri değerlendirir, önerilerde bulunur, Merkeze destek sağlar

Çizelge 8.2. Radyo-Televizyon uygulama ve araştırma merkezi yönetmeliği.

Durum	Veri
İçerik 1	Bu Yönetmeliğin amacı, Düzce Üniversitesi Radyo-Televizyon Uygulama ve Araştırma Merkezinin faaliyet alanları, yönetim organları ve çalışma şeklini düzenlemektir. Kapsamı, merkezin amaçlarına, faaliyet alanlarına, yönetim organlarına ve görevlerine ilişkin hükümleri içerir. Yönetmelik, 4/11/1981 tarihli ve 2547 sayılı Yükseköğretim Kanununun 7 nci maddesinin birinci fıkrasının (d) bendinin (2) numaralı alt bendi ile 14 üncü maddesine dayanılarak hazırlanmıştır. Tanımlar kısmında, Merkez (DÜTRAM), Müdür, Rektör, Üniversite ve Yönetim Kurulu terimleri açıklanmıştır. Merkezin amaçları arasında, iletişim teknolojisi konusunda bilimsel araştırma ve eğitim çalışmalarında bulunmak, öğrencilere uygulama imkânları sağlamak, çeşitli projeler üretmek ve yerel, ulusal ve uluslararası düzeyde etkinlikler düzenlemek bulunmaktadır. Ayrıca, stüdyolarda hazırlanan programların medya kuruluşlarında yayımlanmasını sağlamak da amaçlar arasındadır. Merkezin faaliyet alanları, radyo, televizyon ve kitle iletişim araçlarının iletişim teknolojisi konusunda araştırma, uygulama, eğitim ve iş birliği projelerini içerir. Merkez, aynı zamanda internet üzerinden radyo yayınları yapmak ve programları personel ve öğrencilerin yararına sunmak gibi faaliyetlerde bulunur.
Soru 1	Yönetmeliğin temel amacı nedir?
Cevap 1	Merkezin faaliyetlerini, yönetimini ve çalışma şeklini düzenlemektir.
Soru 2	Hangi kanuna dayanarak hazırlanmıştır?
Cevap 2	2547 sayılı Yükseköğretim Kanunu
Soru 3	Tanımlar kısmında hangi terimler açıklanmıştır?
Cevap 3	Merkez (DÜTRAM), Müdür, Rektör, Üniversite ve Yönetim Kurulu terimleri açıklanmıştır
Soru 4	Merkezin amaçları neleri içerir?
Cevap 4	1: Bilimsel araştırma 2: Bilimsel araştırma, eğitim çalışmaları 3: Bilimsel araştırma, eğitim çalışmaları, projeler üretme
Soru 5	Stüdyolarda hazırlanan programların hedefi nedir?

Çizelge 8.2. (devam) Radyo-Televizyon uygulama ve araştırma merkezi yönetmeliği.

Durum	Veri
Cevap 5	Medya kuruluşlarında yayımlanmasını sağlamaktır
Soru 6	Merkezin faaliyet alanları hangi konuları içerir?
Cevap 6	Radyo, televizyon, iletişim teknolojisi konularında araştırma, uygulama, eğitim ve iş birliği projelerini içerir
Soru 7	Merkez hangi tarih ve kanuna dayanarak oluşturulmuştur?
Cevap 7	4/11/1981 tarihinde, 2547 sayılı Yükseköğretim Kanunu'nun belirli maddelerine dayanılarak oluşturulmuştur.
Soru 8	Danışma Kurulu kimlerden oluşur?
Cevap 8	Yönetim Kurulu tarafından önerilen ve Rektör tarafından üç yıl süre ile görevlendirilen üyelerden oluşur.
Soru 9	Yönetmeliğin amacı dışında hangi faaliyetleri içerir?
Cevap 9	Internet üzerinden radyo yayınları yapmak ve programları personel ve öğrencilere sunmak gibi faaliyetleri içerir.
Soru 10	Yılda en az kaç kere Danışma Kurulu toplanır?
Cevap 10	Yılda en az bir kere toplanarak görüş ve önerilerini bildirir.
İçerik 2	Merkezin yönetim organları Müdür ve Yönetim Kurulundan oluşur. Müdür, Üniversite öğretim elemanları arasından Rektör tarafından üç yıl için görevlendirilir. Müdürün görevleri arasında, merkezi temsil etmek, çalışmaları düzenli olarak yürütmek, Yönetim Kurulunu toplantıya çağırmak ve faaliyet alanı ile ilgili iş birliği çalışmalarını yapmak bulunmaktadır. Yönetim Kurulu, Müdür ve Rektör tarafından görevlendirilen altı öğretim elemanından oluşur. Yönetim Kurulunun görevleri arasında, merkezin çalışma programını hazırlamak, plan ve programları oluşturmak, yıllık faaliyet raporunu hazırlamak ve teknik koşulları oluşturarak radyo ve televizyon yayınlarını yürütmek bulunmaktadır. Merkezin personel ihtiyacı, 2547 sayılı Kanun un 13 üncü maddesi uyarınca Rektör tarafından görevlendirilen personel tarafından karşılanır. Merkez tarafından desteklenen projeler kapsamında alınan her türlü alet, ekipman ve demirbaşlar, proje bitiminde Merkezin kullanımına tahsis edilir. Merkezin harcama yetkilisi Rektördür. Rektör, bu yetkisini kısmen veya tamamen Müdüre devredebilir. Bu Yönetmelikte hüküm bulunmayan hallerde, ilgili diğer mevzuat hükümleri ile Senato, Üniversite Yönetim Kurulu ve Yönetim Kurulu kararları uygulanır. Bu Yönetmelik yayımı tarihinde yürürlüğe girer. Bu Yönetmelik hükümlerini Düzce Üniversitesi Rektörü yürütür.
Soru 1	Merkezin yönetim organları nelerden oluşur?
Cevap 1	Müdür ve Yönetim Kurulundan
Soru 2	Müdür kimler arasından seçilir?
Cevap 2	Üniversite öğretim elemanları arasından
Soru 3	Müdür ne kadar süreyle görevlendirilir?
Cevap 3	Üç yıl

Çizelge 8.2. (devam) Radyo-Televizyon uygulama ve araştırma merkezi yönetmeliği.

Durum	Veri
Soru 4	Müdürün görevleri neleri içerir?
Cevap 4	1: Merkezi temsil etmek 2: Merkezi temsil etmek, çalışmaları düzenli olarak yürütmek 3: Merkezi temsil etmek, çalışmaları düzenli olarak yürütmek, Yönetim Kurulunu toplantıya çağırmak
Soru 5	Yönetim Kurulu kimlerden oluşur?
Cevap 5	altı öğretim elemanından oluşur
Soru 6	Yönetim Kurulunun görevleri neleri içerir?
Cevap 6	1: Çalışma programını hazırlamak 2: Çalışma programını hazırlamak, plan ve programları oluşturmak 3: Çalışma programını hazırlamak, plan ve programları oluşturmak, yıllık faaliyet raporunu hazırlamak
Soru 7	Merkezin personel ihtiyacını karşılayan kimlerdir?
Cevap 7	Rektör tarafından görevlendirilen personel
Soru 8	Merkezin desteklediği projeler kapsamında alınan ekipman ne zaman kullanıma tahsis edilir?
Cevap 8	Proje bitiminde Merkezin kullanımına tahsis edilir
Soru 9	Merkezin harcama yetkilisi kimdir?
Cevap 9	Rektör

Çizelge 8.3. Lisans Eğitim–Öğretim ve Sınav Yönetmeliği (Birinci Kısım)

Durum	**Veri**
İçerik 1	Üniversitenin birimlerine ait öğrenci kontenjanları ile dikey geçiş kontenjanları, her eğitim-öğretim yılında ilgili kurulun önerisi ve Senato kararı alınmak suretiyle Yükseköğretim Kurulu tarafından belirlenir. Üniversitenin yatay geçiş kontenjanları; 24/4/2010 tarihli ve 27561 sayılı Resmî Gazete'de yayımlanan Yükseköğretim Kurumlarında Önlisans ve Lisans Düzeyindeki Programlar Arasında Geçiş, Çift Anadal, Yan Dal ile Kurumlar Arası Kredi Transferi Yapılması Esaslarına İlişkin Yönetmelik hükümlerine göre belirlenir. Öğrencilerin kayıt işlemleri; Ölçme, Seçme ve Yerleştirme Merkezi (ÖSYM) tarafından belirlenen esaslara göre yapılır. Üniversiteye ilk kayıt için her yıl; kayıt tarihi, kayıt yerleri ve istenilen belgeler, birimlere göre Rektörlük tarafından belirlenir ve ilan edilir. Kayıt işlemleri Öğrenci İşleri Daire Başkanlığı tarafından yürütülür. Eksik belge ve posta yoluyla kesin kayıt yapılmaz. Belgelerde tahrifat yaptığı tespit edilenlerin kesin kaydı yapılmaz, kaydı yapılmış olanların kayıtları silinir.
Soru 1	Kayıt işlemleri kim tarafından yürütülür?
Cevap 1	Öğrenci İşleri Daire Başkanlığı.

Çizelge 8.3. (devam) Lisans Eğitim–Öğretim ve Sınav Yönetmeliği (Birinci Kısım)

Durum	**Veri**
İçerik 2	Öğrenciler, akademik takvimde belirtilen süre içerisinde öğrenim katkı paylarının yarıyıl taksitini yatırarak ve alacakları dersleri seçerek internet üzerinden kayıt yenileme işlemlerini yapar. Ders kaydı danışmanın onayından sonra kesinleşir. Akademik takvimde belirtilen süre içerisinde kayıt yenilemeyen öğrenciler haklı ve geçerli nedenlerle dönemin ilk haftası içerisinde kaydını yenilemek zorundadır. Bu süre içerisinde kayıt yenilemeyen öğrencilerin mazeret belgeleri ve başvuruları ilgili yönetim kurulunca karara bağlanır. Geç kayıt sebebiyle ilgili yönetim kurulu kararından önce geçen süre devamsızlıktan sayılır. Belirlenen süreler içinde kayıt yenileme işlemini ve ders kaydını yaptırmayan öğrenciler o yarıyılda derslere devam edemez, sınavlara alınmaz ve öğrencilik haklarından yararlanamazlar. Öğrencinin kayıt yenilememesi sebebiyle geçen süreler, azami öğrenim süresi hesabına katılır. 2547 sayılı Kanunun 46 ncı maddesine göre hesaplanan öğrenim katkı paylarının yarıyıl taksitini akademik takvimde belirtilen kayıt yenileme ve ders kaydı süresi içinde yatırmayan öğrencilerin kayıtları yapılmaz ve yenilenmez.
Soru 1	Geç kayıt sebebiyle geçen süre ne olarak sayılır?
Cevap 1	Devamsızlıktan sayılır
İçerik 3	Öğrenciler, ilk iki yarıyıl dışında, yarıyılın başlangıcından itibaren ilk üç gün içerisinde danışmanın onayını alarak kayıt sırasında aldığı derslerden bazılarını bırakabilir ve başka derslere kayıt yaptırabilir. Yeni eklenen dersler için bu üç günlük süre devamsızlıktan sayılır. Lisans programlarında öğretim dili Türkçedir. Ancak bazı programlarda tamamen veya en az %30 yabancı dilde de öğretim yapılabilir. Ayrıca, Senato kararı ve Yükseköğretim Kurulunun onayı ile yabancı dil hazırlık sınıfı konulan bölümlerde bazı dersler yabancı dilde yapılır. Eğitim-öğretim yılı güz ve bahar yarıyıllarından oluşur. Yeterli talep olduğu takdirde bazı dersler yaz öğretiminde de açılabilir. Güz ve bahar yarıyıllarından her birinin normal eğitim süresi ara sınavlar da dâhil en az on dört haftadır. Yarıyıl sonu sınavları bu sürelere dâhil değildir. Senato gerekli gördüğü hallerde yarıyıl sürelerini uzatabilir veya kısaltabilir. Yaz aylarında yaz öğretimi yürütülebilir. Yaz öğretimi Senato tarafından kabul edilen usul ve esaslara göre yürütülür. Her eğitim-öğretim yılının akademik takvimi, Senato tarafından belirlenir ve ilan edilir. Dersler ve sınavlar, resmi tatil günlerinde yapılmaz. Ancak, gerekli hallerde bölüm başkanlığının önerisi, ilgili yönetim kurulunun kararı ile Rektörlüğe bilgi verilmek suretiyle bazı dersler ve sınavları Cumartesi ve Pazar günleri de yapılabilir.
Soru 1	Eğitim-öğretim yılı hangi yarıyıllarından oluşur?
Cevap 1	Güz ve bahar yarıyıllarından

Çizelge 8.3. (devam) Lisans Eğitim–Öğretim ve Sınav Yönetmeliği (Birinci Kısım)

Durum	**Veri**
İçerik 5	Ön koşullu dersler ve ön koşulları, bölüm kurulu tarafından önerilir, ilgili kurul ve Senato kararıyla kesinleşir. Öğrenciler, ön koşulunu sağlamamış oldukları derslere kayıt yaptıramaz. Eğitim-öğretim çalışmalarının değerlendirilmesi kredi esasına göre yapılır. Ders kredilerinin hesaplanmasında; öğrencinin kazanacağı bilgi, beceri ve yetkinliklere o dersin katkısını ifade eden öğrenim kazanımları ile açıkça belirlenmiş teorik veya uygulamalı derslerle birlikte öğrenciler için öngörülen diğer faaliyetler için gerekli çalışma saatlerini göz önünde bulunduran öğrenci iş yükü esas alınır. Bir öğrencinin her yarıyılda alacağı normal ders yükü, kayıtlı olduğu bölümün öğretim programında belirlenen yüktür. AGNO'su 2.25 veya daha yüksek olan öğrenciler, ilgili yarıyıla ait ders kredisi toplamının 1/3 fazlasını aşmamak kaydıyla içinde bulundukları yarıyıl ve üst yarıyıllara ait derslere danışman onayı ile kayıt yaptırabilir. İlk iki yarıyıldan sonraki ders kayıtlarında öğrenciler öncelikle daha önceki dönemlerde başarısız oldukları veya hiç almadıkları dersleri almak zorundadır. Öğrencinin kayıt olduğu derslerin tümü alt yarıyıllara ait başarısız olduğu dersler ise, bu derslerin toplam kredisi yarıyıl ders yükünden fazla olabilir. Ancak alt yarıyıllardan almadığı veya başarısız olduğu derslerin olması halinde, en alttaki yarıyıl derslerden başlamak şartıyla kayıt yaptıracağı derslerin toplam kredisi bulunduğu yarıyıl ders yükünün 1/3 fazlasını aşamaz. Öğrenciler daha önce aldıkları dersleri, AGNO'larını yükseltmek amacıyla tekrar alabilirler. Tekrar edilen derslerde en son alınan not geçerlidir.
Soru 1	Öğrenciler tekrar edilen derslerde ne yapabilir?
Cevap 1	En son alınan not geçerlidir.
İçerik 6	Öğrenciler ilk defa aldıkları derslerin teorik bölümüne %70 oranında, uygulama bölümüne %80 oranında devam etmek zorundadır. Daha önce alınıp devam koşulu yerine getirilen bir dersin tekrarında devam zorunluluğunun aranıp aranmamasına; birimin fiziki kapasitesi, öğretim elemanı ve öğrenci sayısı da göz önüne alınarak, ders sorumlusunun uygun görüşü, bölüm başkanlığının önerisi ile ilgili yönetim kurulu tarafından karar verilir. Devam zorunluluğunun aranmaması halinde de öğrenci o derse kayıt olmak, varsa ödevleri yapmak, atölye ve uygulamalı derslerde uygulamalara ve ara sınavlara katılmak zorundadır. Öğrencilerin; bilimsel, sosyal, kültürel ve sportif etkinlikler ve karşılaşmalar sebebiyle bölüm başkanlığının önerisi ve ilgili yönetim kurulunun onayı ile izinli oldukları süreler devamsızlıktan sayılmaz. Ders sorumluları öğrencilerin derse devamlarını izlemekle yükümlü ve bu konuda bölüm başkanlığına karşı sorumludur. Eğitim-öğretim programlarının özelliklerine göre zorunlu görülecek meslek stajları ve endüstriye dayalı eğitime ilişkin esaslar, Yükseköğretim Kurulu tarafından alınan kararlar çerçevesinde Senato tarafından düzenlenir. Staj zorunluluğu olan programlarda stajlar, ilgili kurulca belirlenen staj programı esaslarına göre yapılır. Bazı programlarda ise isteğe bağlı staj uygulaması Senato tarafından düzenlenir. Staj yerleri daha önceden belirlenmiş ve birimleri tarafından onaylanmış öğrencilerin, staj yerlerini birimlerinin haberi olmadan değiştirmeleri halinde yapmış oldukları staj geçersiz sayılır. Bitirme çalışması ve proje çalışması alma, çalışmanın süresi, çalışmanın teslimi ve değerlendirilmesi gibi konulara ilişkin usul ve esaslar ilgili bölümce belirlenir ve ilgili kurulun onayı alınarak uygulamaya konulur.
Soru 1	Staj zorunluluğu olan programlarda stajlar nasıl yapılır?
Cevap 1	İlgili kurulca belirlenen staj programı esaslarına göre yapılır.

Çizelge 8.3. (devam) Lisans Eğitim–Öğretim ve Sınav Yönetmeliği (Birinci Kısım)

Durum	**Veri**
İçerik 7	Yatay ve dikey geçişle gelen öğrenciler hariç, ilk defa kayıt yaptıran öğrenciler, yarıyıl başlangıcından itibaren en geç ilk iki hafta içinde, ek kontenjan ile gelen öğrenciler ise kayıt oldukları gün itibarıyla bir hafta içinde, daha önce başka bir yükseköğretim kurumunda aldıkları ve başarılı oldukları derslerden muaf tutulmak için, derslerin alındığı tarih üzerinden beş yıldan fazla süre geçmemek koşuluyla bölüm başkanlığına başvurabilir. Muafiyet istekleri, bölüm başkanlıklarınca oluşturulacak intibak komisyonunda değerlendirilip, ilgili yönetim kurulunca görüşülerek karara bağlanır ve öğrenciye bildirilir. Öğrencinin muafiyet aldığı derslerin notları, Üniversitenin not sistemine dönüştürülerek AGNO hesabına katılır. Muafiyet istekleri uygun görülen öğrenciler, muaf tutuldukları dersleri not yükseltmek amacıyla tekrar alabilirler. Yatay ve dikey geçişle gelen öğrenciler hariç, ilk defa kayıt yaptıran öğrenciler, dönem kredilerini geçmemek koşuluyla muaf oldukları derslerin yerine daha önce hiç almadıkları veya bir üst yıla ait olan dersleri alabilirler.
Soru 1	Muafiyet başvurusu için süre nedir?
Cevap 1	İlk iki hafta içinde.
İçerik 8	Yabancı dil hazırlık sınıfı bulunan birimlere kayıt olan öğrenciler, ilk yıl hazırlık sınıfında öğrenim görür. Bu öğrencilerden isteyenler, yabancı dil muafiyet sınavına girebilirler. Muafiyet sınavında başarılı olanlar hazırlık sınıfından muaf tutulur. Yabancı dil hazırlık sınıfı bulunmayan birimlerin öğrencileri, eğitim-öğretim yılı başında açılacak yabancı dil muafiyet sınavında başarı gösterirlerse, yabancı dil derslerinden muaf olurlar. Bu sınavın yapılışı, değerlendirilmesi ve alınan notun sağlayacağı muafiyetler, Senato tarafından belirlenen usul ve esaslara göre yürütülür. Üniversitenin yabancı dil hazırlık sınıfı okutulan birimine başka yükseköğretim kurumundan yatay veya dikey geçişle gelen öğrenciler, geldikleri birimde okuyup başarılı oldukları hazırlık sınıfından muaf tutulurlar. Hazırlık sınıfı okumadan yatay veya dikey geçişle gelen öğrencilerin, yabancı dil yeterlik sınavında yeterlik gösterememeleri halinde Üniversitede hazırlık sınıfı okumaları gerekir. Yabancı dil hazırlık sınıfında başarılı olan öğrencilere bu durumlarını ve başarı notlarını içeren bir belge verilir. Yabancı dil hazırlık sınıfında başarılı olamayan öğrenciler, isterlerse ikinci yıl bu sınıfı tekrar edebilir veya esas kayıtlı oldukları programa devam edebilirler. Yabancı dil hazırlık sınıfı okutulan birimlerde hazırlık sınıfının süresi, bu Yönetmeliğin 26. maddesinde yer alan sürelere dahil değildir. Hazırlık sınıfına ait süre ve başarı esasları Senato tarafından belirlenir.
Soru 1	Yabancı dil muafiyet sınavı kimler için geçerlidir?
Cevap 1	Hazırlık sınıfındakiler.

Çizelge 8.4. Lisans Eğitim–Öğretim ve Sınav Yönetmeliği (İkinci Kısım)

Durum	**Veri**
İçerik 9	Herhangi bir programa kayıtlı olan öğrencilerin istekleri halinde, kendi lisans programlarına ek olarak başka bir programdan ders almalarına yan dal programı denir. Yan dal programı ayrı bir lisans programı değildir. Yan dal programı uygulama usul ve esasları Senato tarafından belirlenir. Herhangi bir lisans programına kayıtlı öğrenciler; 2547 sayılı Kanunla belirlenen azami öğrenim süresini kullanmak üzere, kayıtlı oldukları lisans programından başka bir programa ikinci dal olarak devam edip, başarmaları halinde ikinci bir lisans diploması alabilir. Çift ana dal programı esasları Senato tarafından belirlenir.
Soru 1	Yan dal programının uygulama usul ve esaslarını kim belirler?
Cevap 1	Senato tarafından belirlenir.
İçerik 10	Sınavlar sekiz farklı türde gerçekleşir. Bunlar ara sınavlar, kısa sınavlar, mazeret sınavları, yarıyıl sonu sınavları, ek sınavlar, yaz okulu sonu sınavları, tek/çift ders sınavları ve muafiyet sınavlarıdır. Sınavlar yazılı, sözlü ve/veya uygulamalı olabilir, sözlü sınavlar için komisyon oluşturulabilir. Hafta sonu tatilleri dışında sınav yapılabilir ve öğrenciler sınav programlarına uygun olarak sınavlara girmek zorundadır. Sağlık, doğal afet ve benzeri mazeretlerle sınavlara gelemeyen öğrencilerin mazeretleri kabul edilebilmesi için belgelerini ilgili birime teslim etmeleri gerekmektedir. Her öğretim elemanı, bir yarıyıl boyunca işleyeceği konuların haftalık planını, ara sınav, yarıyıl sonu sınavı notu ve diğer çalışmaların başarı puanına katılım oranlarını içeren ders planını öğrencilere ve bölüm başkanlığına her yarıyılın ilk iki haftası içinde yazılı olarak vermelidir. Her ders için her yarıyılda en az bir ara sınav yapılır ve ara sınav notları öğrencilere en geç yarıyıl derslerinin sona ermesinden bir hafta önce duyurulur.
Soru 1	Sınavlara gelmeyen öğrencilerin ilişkisi ne olur?
Cevap 1	İlişikleri kesilir.
İçerik 11	Yarıyıl sonu sınavları programı, yarıyılda derslerin sona ermesinden onbeş gün önce bölüm başkanlığı tarafından hazırlanarak öğrencilere ilan edilir. Ara sınav ve yarıyıl sonu sınavları, bölüm başkanlığı tarafından belirlenen gün, saat ve yerde, ders sorumlularının ve bölüm başkanlığınca görevlendirilen gözcülerin denetimi ve gözetiminde yapılır. Ara sınav ve yarıyıl sonu sınav sonuçları, öğrencilere ilan edilir ve internet ortamındaki listelere kaydedilir. Yarıyıl sonu sınavları, Senato tarafından belirlenen akademik takvimdeki süreler içinde yapılır.
Soru 1	Sınavlar kimin denetimi ve gözetiminde yapılır?
Cevap 1	Ders sorumluları ve gözcülerin.
İçerik 12	Son yarıyıl öğrencilerine, mezun olabilecekleri dersleri başarmaları durumunda 'tek/çift ders sınav hakkı' verilir. Bu haktan yararlanmak isteyen öğrencilerin başvuruları, yarıyıl sonu sınavları sonuçlarının ilanından itibaren bir hafta içinde bölüm başkanlığına dilekçe ile yapılmalıdır. Tek/çift ders sınavından alınan puanın 100 üzerinden 50 puan ve daha yukarısı olması durumunda öğrenci başarılı sayılır ve sınav notu harf notuna çevrilir. Sınavlar; ara sınavlar, kısa sınavlar, mazeret sınavları, yarıyıl sonu sınavları, ek sınavlar, yaz okulu sonu sınavları, tek/çift ders sınavları, muafiyet sınavları olmak üzere sekiz çeşittir. Yazılı, sözlü ve/veya uygulamalı olarak yapılabilir. İlgili yönetim kurulu kararıyla sadece sözlü sınavlar için komisyon oluşturulabilir. Sınavlar, dini ve milli bayramlar dışında hafta sonu tatillerinde de yapılabilecektir. Öğrenciler, sınav programlarında belirtilen gün, saat ve yerde sınava girmek ve öğrenci kimlik belgeleri ile diğer kimlik belgelerini yanlarında bulundurmak zorundadır. Usulüne uygun kayıt olmadıkları ve belirlenen şartları sağlamadıkları dersin sınavına girmeleri halinde aldıkları not iptal edilir.

Çizelge 8.4. (devam) Lisans Eğitim–Öğretim ve Sınav Yönetmeliği (İkinci Kısım)

Durum	**Veri**
Soru 1	Öğrenciler sınavlarda yanlarında ne bulundurmalıdır?
Cevap 1	Kimlik belgeleri.

Çizelge 8.5. Lisans Eğitim–Öğretim ve Sınav Yönetmeliği (Üçüncü Kısım)

Durum	**Veri**
İçerik 13	Tek/Çift ders sınav hakkı konusundaki düzenlemeler şu şekildedir: Kaydoldukları dersleri başarmaları halinde mezun olabilecek öğrencilere son yarıyıl öğrencileri denir. Bu öğrencilerden, yarıyıl sonu sınavları sonunda başarısız oldukları tek/çift dersi kalarak bu derslerin devam ve diğer koşullarını sağlayanlara tek/çift ders sınav hakkı verilir. Hiç alınmamış dersler için tek/çift ders sınav hakkı verilmez. Tek/çift ders sınav hakkı kullanmak durumunda olan öğrencilerin, yarıyıl sonu sınavları sonuçlarının ilanından itibaren bir hafta içerisinde dilekçe ile bölüm başkanlığına başvurmaları gerekir. Tek/çift ders sınavından alınan puanın 100 üzerinden 50 puan ve daha yukarısı olması halinde öğrenci başarılı sayılır. Ara sınav notu ve diğer etkinliklerin katkısı olmadan tek/çift ders sınavının puanı, harf notuna çevrilir. Başarı puanı ve harf notu belirleme konusundaki düzenlemeler şu şekildedir: Öğrencinin; yarıyıl içindeki çalışmaları, ara sınav ve yarıyıl sonu sınav notları, derse devamı, öğretim elemanının yarıyıl başında bölüm başkanlığına teslim ettiği ders planında gösterilen şekilde değerlendirmeye alınarak dersin başarı puanı ve harf notu takdir edilir. Başarı puanlarına karşılık gelen harf notları ve katsayıları belirli bir sistem dahilinde belirlenir.
Soru 1	Başarılı sayılmak için kaç puan alınmalıdır?
Cevap 1	50 puan ve daha yukarısı.
İçerik 14	Ağırlıklı genel not ortalaması hesaplama konusundaki düzenlemeler şu şekildedir: Ağırlıklı genel not ortalaması (AGNO); derslerin harf notu karşılığı katsayıları ile derslerin kredi değerleri çarpılarak elde edilen toplam sayının krediler toplamına bölünmesiyle hesaplanır. DVZ notu alınmış derslerin kredileri de kredi toplamına dahil edilir. AGNO, iki haneli ondalık sayı olarak belirlenir. Onur ve üstün onur öğrencileri belirleme konusundaki düzenlemeler şu şekildedir: Dördüncü yılın sonunda, öğretim programlarındaki tüm derslerini almış ve başarmış öğrencilerden AGNO'su 2.50 olan öğrenciler başarılı sayılır. AGNO 3.25-3.49 olan öğrenciler onur listesinde, 3.50-4.00 olan öğrenciler üstün onur listesinde yer alır. Bu listeler bölümlerin ilan panolarında en az bir ay süreyle asılır. Notlarda maddi hata ve sınav sonuçlarına itiraz konusundaki düzenlemeler şu şekildedir: Not düzeltme işlemleri, öğrencinin sınav sonuçlarının duyurulmasından itibaren belirli bir süre içinde yapılabilir. Öğrenci, sınav kağıdının yeniden incelenmesini istemek için belirli bir süre içinde başvuruda bulunabilir. Ayrıca, not ortalamalarına ilişkin herhangi bir maddi hatanın belirlenmesi halinde, ilgili öğretim elemanı düzeltme talebinde bulunabilir.
Soru 1	Onur ve üstün onur listeleri ne kadar süre asılır?
Cevap 1	En az bir ay.

Çizelge 8.5. (devam) Lisans Eğitim–Öğretim ve Sınav Yönetmeliği (Üçüncü Kısım)

Durum	**Veri**
İçerik 15	Başarı denetimi konusundaki düzenlemeler şu şekildedir: Sekizinci yarıyıl sonunda bütün derslerden başarılı olsalar dahi AGNO'su 2.50'nin altında olan öğrenciler mezun olamaz, bu durumdaki öğrenciler AGNO'larını 2.50'nin üzerine çıkartıncaya kadar derslere yeniden kayıt yaptırmak, derslerin ara ve dönem sonu sınavlarına girmek durumundadır. Meslek yüksekokullarına intibak konusundaki düzenlemeler şu şekildedir: Lisans programını tamamlamayan veya tamamlayamayanlar, ilgili yükseköğretim kurumları ile ilişiklerinin kesildiği tarihten itibaren altı ay içinde müracaat etmek şartıyla meslek yüksekokullarının uygun programlarına intibak ettirilebilirler. Sınav evrakının saklanması konusundaki düzenleme şu şekildedir: Sınavlara itiraz süresi tamamlandıktan sonra sınav evrakları, öğretim elemanları tarafından ilgili bölüm başkanlığına teslim edilir. Sınav evrakları, bölüm başkanlığında bir yıl saklanır ve dört yıl daha saklanmak üzere ilgili birimin arşivine kaldırılır.
Soru 1	Meslek yüksekokullarına intibak süresi nedir?
Cevap 1	Altı ay.
İçerik 16	Mezun olmaya hak kazanmak için öğrencilerin; öğretim programlarındaki derslere devam edip tümünden başarılı olmaları, AGNO'larının en az 2.50 düzeyinde olması ve diğer bütün çalışmalarını tamamlamış olmaları gerekir. Mezun olmaya hak kazanan öğrenciye; diploma ile başardığı derslerin isimleri, harf notları ve diğer faaliyetlerin başarılmasıyla ilgili bilgilerin yer aldığı bir not döküm belgesi verilir. Bir programın mezunları arasından ilk üçe girenlerin belirlenmesinde, AGNO virgülden sonra ayırt edici basamağa kadar yürütülür. Mezun olan öğrenciler arasında dereceye girenlere, mezuniyet töreninde başarı belgeleri verilir. Bölümlerde dereceye giren öğrencilere de bölüm başkanları tarafından başarı belgeleri verilir. Fakülte ve yüksekokullardan bu Yönetmelik hükümlerine göre öğrenimlerini tamamlayarak mezun olan öğrencilere, fakülte, yüksekokul ve bölüm adı ile varsa yan dal adı belirtilmek suretiyle hazırlanan lisans diploması verilir. Fakülte diplomasında dekan ve Rektörün, yüksekokul diplomasında müdür ve Rektörün imzalarının olması gerekir. Ayrıca, diplomanın arka yüzüne, öğrencinin mezuniyetinin onaylandığına ilişkin ibare yazılır ve Öğrenci İşleri Daire Başkanı ve ilgili bölüm başkanı tarafından imzalanır.
Soru 1	Mezuniyet töreninde ne verilir?
Cevap 1	Başarı belgeleri.
İçerik 17	Diplomanın şekli, ölçüleri ve üzerinde yer alacak diğer bilgiler Senato tarafından belirlenir. Diploma hazırlanıncaya kadar öğrenciye bir defaya mahsus olmak üzere bölüm başkanı ve Öğrenci İşleri Daire Başkanının imzalarını taşıyan geçici mezuniyet belgesi verilir. Bu belge, diploma öğrenciye verilirken Üniversite tarafından geri alınır ve bir yenisi verilmez. Diploma bir defa için verilir. Diplomanın kaybedilmesi durumunda, belirlenen harç yatırılarak dilekçe ile başvurulması üzerine ikinci nüshası hazırlanır. Bu şekilde verilen diploma üzerinde ikinci nüsha ibaresi konulur. Diploma ile birlikte her öğrenciye tamamlayıcı bir belge olan diploma eki verilir. Diploma ekinde; alınan derece ile ilgili bilgiler, derecenin düzeyi, içeriği ve kullanım alanları, eğitim sistemi ve Üniversitenin eğitim ve değerlendirme sistemi gibi bilgilere yer verilir. Diploma ekinin dili İngilizce'dir. Öğrencilerin disiplin iş ve işlemleri; 2547 sayılı Kanunun 54 üncü maddesi ve Yükseköğretim Kurumları Öğrenci Disiplin Yönetmeliği hükümlerine göre yürütülür. Okuldan uzaklaştırma cezası alan öğrencilerin bu süreleri öğrenim süresinden sayılır ve bu süreler için katkı payı ödemeleri gerekir. Okuldan uzaklaştırma cezası alan öğrenciler, hiçbir şekilde Üniversite binalarına ve tesislerine giremez, bilimsel, sosyal, kültürel ve sportif etkinliklerde takımlarda yer alamaz.
Soru 1	Diplomanın şekli ve ölçüleri kim tarafından belirlenir?

Çizelge 8.5. (devam) Lisans Eğitim–Öğretim ve Sınav Yönetmeliği (Üçüncü Kısım)

Durum	**Veri**
Cevap 1	Senato.
İçerik 18	Aşağıda sayılan haklı ve geçerli mazeretler nedeniyle ilgili yönetim kurulu tarafından, öğrencinin kayıt dondurmasına ve öğrenimine ara verilmesine karar verilir: a) Üniversite hastanesinden veya diğer sağlık kuruluşlarından alınan sağlık raporlarıyla belgelenmiş bulunan sağlıkla ilgili mazeretinin olması, b) Mahallin en büyük mülki amirince verilecek bir belge ile belgelendirilmiş olmak koşuluyla doğal afetler nedeniyle öğrencinin öğrenimine ara vermek zorunda kalması, c) Anne, baba, kardeş, eş veya çocuğunun ölümü ya da bunlardan birinin ağır hastalığı halinde bakacak başka bir kimsenin bulunmaması sebebiyle öğrencinin öğrenimine ara vermek zorunda kaldığını belgelemesi, ç) Belgelendirilmek kaydıyla ekonomik nedenlerle öğrencinin öğrenimine devam edememesi, d) Tutukluluk hali, e) Tecil hakkını kaybetmesi veya tecilin kaldırılması suretiyle askere alınması, f) İlgili yönetim kurulunca kabul edilen diğer mazeretlerin ortaya çıkması. Kayıt dondurmak için kayıt dondurma başvuruları ilgili yarıyılda derslerin başlama tarihinden itibaren en geç 15 gün içerisinde yapılır. Belgelemek koşulu ile ani hastalık ve beklenmedik durumlar dışında bu süreler bittikten sonra yapılan başvurular kabul edilmez. Kaydı dondurulan öğrenci derslere devam edemez ve sınavlara giremez. Kayıt dondurma ile geçen süreler, öğrenim süresinden sayılmaz.
Soru 1	Kayıt dondurma ile geçen süreler ne olarak sayılmaz?
Cevap 1	Öğrenim süresinden sayılmaz.
İçerik 19	Aşağıda sayılan hallerde öğrencinin Yönetim Kurulu kararıyla Üniversite ile ilişiği kesilir: a) Yükseköğretim Kurumları Öğrenci Disiplin Yönetmeliğine göre Üniversiteden çıkarma cezası almış olması, b) Kendi isteğiyle Üniversiteden ayrılma talebinde bulunması. c) İlgili yönetim kurulu kararı ve Yükseköğretim Kurulunun onayı alınarak dört yıl üst üste katkı payı veya öğrenim ücretinin ödenmemesi ile kayıt yenilenmemesi nedeniyle öğrencilerin ilişikleri kesilebilir. ç) Azami süresi sonunda kayıtlı oldukları programdan mezun olamayan öğrencilerin; hiç almadıkları ve alıp da devam koşulunu sağlamadıkları derslerin toplam sayısı beşten fazla olanların kayıtları silinir. d) Derslere devam koşulunu yerine getirdikleri halde, kullandıkları ek süreler sonunda yıl içi ve yıl sonu sınav yükümlülüklerini 26 ncı maddede belirtilen hükümlere uygun olarak yerine getirmeyenlerin kayıtları silinir. Kaydı silinen ya da kaydını kendi isteğiyle sildiren öğrenciye yatırdığı öğrenim katkı payı iade edilmez.
Soru 1	Kaydı silinen veya kendi isteğiyle sildiren öğrenciye ne iade edilmez?
Cevap 1	Öğrenim katkı payı iade edilmez.
İçerik 20	Öğrenciye her türlü tebligat, Öğrenci İşleri Daire Başkanlığına bildirmiş olduğu adrese, ilgili mevzuat hükümlerine göre yapılır. Öğrenci adres değişikliklerini Öğrenci İşleri Daire Başkanlığına bir dilekçe ile bildirmek zorundadır. Yeni adres bildirimi yapılmaması nedeniyle eski adrese yapılan tebligat geçerli sayılır. Yanlış veya eksik adres bildiren ya da adres değişikliğini bildirmeyen öğrencilerin mevcut adreslerine tebligat yapılmış sayılır. Bu Yönetmelikte hüküm bulunmayan hallerde, ilgili diğer mevzuat hükümleri ile Yükseköğretim Kurulu, Senato ve Yönetim Kurulu kararları uygulanır. 10/7/2008 tarihli ve 26932 sayılı Resmî Gazete'de yayımlanan Düzce Üniversitesi Lisans Eğitim-Öğretim ve Sınav Yönetmeliği yürürlükten kaldırılmıştır.
Soru 1	Yanlış adres bildiren öğrencilerin durumu nasıl değerlendirilir?
Cevap 1	Mevcut adrese tebligat yapılmış sayılır.

Çizelge 8.5. (devam) Lisans Eğitim–Öğretim ve Sınav Yönetmeliği (Üçüncü Kısım)

Durum	**Veri**
İçerik 21	2011-2012 eğitim öğretim yılı güz yarıyılında ilk defa kayıt yaptıracak öğrenciler dışında kalan öğrenciler, 2011-2012 eğitim öğretim akademik yılı başlangıcından itibaren bir ay içinde başvurmaları halinde bu Yönetmelik hükümlerine tabii olurlar. 2006-2007 eğitim-öğretim yılından önce kayıt olan öğrenciler, istekleri halinde diplomalarını Abant İzzet Baysal Üniversitesinden almak üzere başvurabilirler. AKTS kredisi, not değerlendirme sistemi ve derslerin kredi değerine ilişkin hükümler, 2013-2014 eğitim-öğretim yılından önce kayıtlı olan öğrencilere de uygulanır. 2547 sayılı Yükseköğretim Kanununun geçici 67 nci maddesinde belirtildiği üzere, bu maddenin yürürlüğe girdiği tarihte kayıtlı olan öğrenciler bakımından azami sürelerin hesaplanmasında, daha önceki öğrenim süreleri dikkate alınmaz. Bu Yönetmelik yayımı tarihinde yürürlüğe girer. Bu Yönetmelik hükümlerini Düzce Üniversitesi Rektörü yürütür.
Soru 1	2006-2007 öncesi kayıtlı öğrenciler diplomalarını nereden alabilir?
Cevap 1	Abant İzzet Baysal Üniversitesinden

ÖZGEÇMİŞ

KİŞİSEL BİLGİLER

Adı Soyadı : Hüseyin Avni ARDAÇ

Yabancı Dili : İngilizce

ÖĞRENİM DURUMU

Derece	**Alan**	**Okul/Üniversite**	**Mezuniyet Yılı**
Doktora	Bilgisayar Müh.	Düzce Üniversitesi	2024
Y. Lisans	Bilgisayar Müh.	Düzce Üniversitesi	2018
Lisans	Bilgisayar Müh.	Düzce Üniversitesi	2015
Lise		Türk Telekom Şehit Murat Naiboğlu Anadolu Lisesi	2010

TEZDEN YAYINLAR

H. A. Ardaç and P. Erdoğmuş, "Question answering system with text mining and deep networks," *Evolving Systems,* pp. 1-13, 2024.

DİĞER YAYINLAR

H. A. Ardaç and P. Erdoğmuş, "Image Denoising with Modified Grey Wolf Optimizer," *Düzce Üniversitesi Bilim ve Teknoloji Dergisi,* vol. 6, no. 4, pp. 962-982, 2018.