T.C.
MANİSA CELAL BAYAR ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

YÜKSEK LİSANS TEZİ
YAZILIM MÜHENDİSLİĞİ ANABİLİM DALI
YAZILIM MÜHENDİSLİĞİ BİLİM DALI

ÇİFT YÖNLÜ ENKODER TRANSFORMATÖR TABANLI DUYGU ANALİZİ DERİN ÖĞRENME MODELİ GELİŞTİRİLMESİ

Cevhernur SÖYLEMEZ

Doç. Dr. Akın ÖZÇİFT

Doç. Dr. Deniz KILINÇ

MANİSA–2022

CEVHERNUR SÖYLEMEZ	**ÇİFT YÖNLÜ ENKODER TRANSFORMATÖR TABANLI DUYGU ANALİZİ DERİN ÖĞRENME MODELİ GELİŞTİRİLMESİ**	**2022**

TAAHHÜTNAME

Bu tezin Manisa Celal Bayar Üniversitesi Hasan Ferdi Turgutlu Teknoloji Fakültesi Yazılım Mühendisliği Bölümü'nde, akademik ve etik kurallara uygun olarak yazıldığını ve kullanılan tüm literatür bilgilerinin referans gösterilerek tezde yer aldığını beyan ederim.

CEVHERNUR SÖYLEMEZ

İÇİNDEKİLER

Sayfa

SİMGELER VE KISALTMALAR DİZİNİ

BERT	Bidirectional Encoder Representations from Transformers
BoW	Kelime Torbası (Bag-of-Words)
CBoW	Sürekli Kelime Torbası (Continous Bag-of-Words)
DVM	Destek Vektör Makinaları
DDİ	Doğal Dil İşleme
IDF	Ters Terim Frekansı
MCC	Matthews Korelasyon Katsayısı
MÖ	Makine Öğrenmesi
NB	Naive Bayes Algoritması
RNN	Yinelenen Sinir Ağları (Recurrent Neural Network)
SDM	Sinirsel Dil Modelleri
SG	Skip – Gram
TF	Terim Frekansı
YSA	Yapay Sinir Ağları
YZ	Yapay Zekâ

ŞEKİLLER DİZİNİ

Sayfa

TABLO DİZİNİ

Sayfa

TEŞEKKÜR

Tez çalışmamın her aşamasında bana destek olan, akademik kariyerime katkılarda bulunan, bilgi ve tecrübeleri ile yol gösteren, fikirleri ile yol açan danışman hocam Sayın Doç. Dr. Akın ÖZÇİFT'e, bilgi ve tecrübeleri ile çalışma hayatımın tüm zorlu aşamalarında maddi manevi her yönden katkıda bulunan, kendisini tanımaktan ve birlikte çalışmaktan onur duyduğum sevgili hocam Sayın Doç. Dr. Deniz KILINÇ'a, tez yazım sürecimde ve sonrasında verdiği desteklerle zorlu süreçlerimi kolaylaştıran, birlikte çalışmaktan mutluluk duyduğum hocam Sayın Dr. Öğr. Üyesi Okan ÖZTÜRKMENOĞLU'na ve her zaman yanımda olan, en zorlu zamanlarda moral ve güç veren çok sevgili iş arkadaşlarım İzmir Bakırçay Üniversitesi Mühendislik ve Mimarlık Fakültesi asistanlarına çok teşekkür ederim.

Hayatıma girdiği günden beri desteğini her zaman hissettiğim nişanlım Muhammed PEKTAŞ'a sabrı ve hoşgörüsü için teşekkür ederim.

Tabi ki öğrenim hayatım boyunca beni maddi ve manevi olarak destekleyen, her ihtiyacım olduğunda yanımda ve arkamda olan başta babam İbrahim Ethem SÖYLEMEZ ve annem Rukiye SÖYLEMEZ olmak üzere tüm aileme minnet ve teşekkürlerimi sunarım.

Cevhernur SÖYLEMEZ
Manisa, 2022

ÖZET

Yüksek Lisans Tezi

Manisa Celal Bayar Üniversitesi
Fen Bilimleri Enstitüsü
Yazılım Mühendisliği Anabilim Dalı

Danışman: Doç. Dr. Akın ÖZÇİFT

İkinci Danışman: Doç. Dr. Deniz KILINÇ

Bu tezde, Türkçe duygu analizi için yeni sinirsel dil modeli olan BERT modeli uygulanmıştır. Doğal dil işleme son zamanlarda ilerleme kaydetmiş olsa da bileşik anlamları temsil etmek zorlu bir iştir. Geleneksel derin öğrenme yöntemleri, cümlelerin sıradan bir doğrusal yapı, yani zincirler veya diziler olduğunu iddia eder. Bu tezde, kelimelerin morfolojik yapısı dikkate alınarak Türkçe duygu analizi problemi için daha başarılı sonuçlar elde etmek amacıyla önceden eğitilmiş sinir ağı kullanılarak uygulama geliştirilmiştir.

Etiketlenmemiş metinden yakın zamanda yapılan derin çift yönlü bir kendi kendine dikkat gösterimi olan BERT, ince ayarlı birçok Doğal Dil İşleme (DDİ) görevinde dikkate değer sonuçlar elde etti. Bu tezde, morfolojik olarak zengin bir dil olan Türkçe için BERT algoritması etkinliğini göstermek amaçlanmıştır. Morfolojik olarak zengin diller, verileri Makine Öğrenimi (MÖ) algoritmalarına uygun olacak şekilde modellemek için yoğun dil ön işleme adımları gerektirir. Özellikle, veri seyrekliği veya yüksek boyutlu problemlerin üstesinden gelmek için verimli bir veri modeli elde etmek için parçalama, kök bulma görevlerine ihtiyaç vardır. Bu bağlamda, literatürden Türkçe için yaygın DDİ araştırma problemlerinden duygu analizi problemi seçilmiştir. Sonrasında BERT'in deneysel performansı temel MÖ algoritmalarıyla karşılaştırıldı. Ağır ön işleme görevlerini ortadan kaldırırken, seçilen DDİ probleminde temel MÖ algoritmalarına kıyasla gelişmiş sonuçlar elde edilmiştir.

Anahtar Kelimeler: Makine öğrenmesi, doğal dil işleme, sinirsel dil modelleri, Türkçe duygu analizi

2022, 69 sayfa

ABSTRACT

M.Sc. Thesis

Cevhernur SÖYLEMEZ

Celal Bayar University
Faculty of Hasan Ferdi Turgutlu Technology
Department of Software Engineering

Supervisor: Assoc. Prof. Dr. Akın ÖZÇİFT

Co-Supervisor: Assoc. Prof. Dr. Deniz KILINÇ

In this thesis, the BERT model, which is a new neural language model for Turkish sentiment analysis, was applied. Although natural language processing has recently progressed, representing compound meanings is a challenge. Traditional deep learning methods claim that sentences are an ordinary linear structure, that is, chains or sequences. In this thesis, an application has been developed by using a previously trained neural network in order to obtain more successful results for the Turkish sentiment analysis problem, taking into account the morphological structure of the words.

BERT, a recent deep bidirectional demonstration of self-attention from unlabeled text, achieved remarkable results on many fine-tuned Natural Language Processing (NLP) tasks. This thesis, it is aimed to show the effectiveness of the BERT algorithm for Turkish, which is a morphologically rich language. Morphologically rich languages require intensive language preprocessing steps to model the data in a way that conforms to Machine Learning (ML) algorithms. In particular, fragmentation, and root finding tasks are needed to obtain an efficient data model to overcome data sparsity or high dimensional problems. In this context, the sentiment analysis problem, which is one of the common NLP research problems for Turkish, was chosen from the literature. The experimental performance of BERT was then compared with the basic ML algorithms. While eliminating heavy preprocessing tasks, improved results were obtained in the selected NLP problem compared to the basic ML algorithms.

Keywords: Machine learning, natural language processing, neural language models, Turkish sentiment analysis

2022, 69 pages

1. GİRİŞ

İnternet altyapısının genişlemesiyle beraber sosyal medya kullanımı oldukça artmıştır. Sosyal medya platformlarının gelişimi, insanların eylemlerini, inançlarını ve hayatlarını mecazi ve yaratıcı bir dil kullanarak ifade etmelerine olanak sağlamıştır. Aynı zamanda Web 2.0'ın hızlı büyümesi ve gelişmesi internet kullanıcısı verilerini arttırdığından, sosyal medyadan, film, otel, ürün incelemelerinden farklı veri kaynakları oluşturmak, insan davranışları ve düşünceleri hakkında stratejik bir anlayış sunar.

Artan veri üzerinde veriyi anlamlı hale getirmek için yapılan çalışmalar sonucunda Yapay Zekâ (YZ) kavramı ortaya çıkmıştır. YZ, insan düşünce sistemini matematiksel olarak modellemeyi ve insanlar gibi düşünebilen ve öğrenebilen akıllı sistemler geliştirmeyi amaçlayan bir araştırma konusudur. YZ'nin bir alt alanı olarak Doğal Dil İşleme (DDİ), bilgi çıkarma, anlam çıkarma ve duygu analizi gibi belirli amaçlar için dil işlemeyi mümkün kılar. Duygu Analizi (DA), belirli bir metin, belge veya cümledeki görüşleri sınıflandırma ve keşfetme ile ilgili yaygın bir araştırma konusudur. DA çalışmaları, insan davranışlarını etkileyen çok önemli bir faktör olarak insanların görüşlerinin anlaşılmasında akademiden ticari uygulamalara kadar önemli bir potansiyel etkiye sahiptir.

Bir bilgisayarda cümle yapısını öğrenmek ve anlamlarını anlamak zorlu bir iştir. Anlambilim açısından temel zorluk, daha uzun ifadelerin anlamını yakalamaktır. DDİ yöntemleri son yıllarda ilerleme kaydetmiş olsa da, metin anlamlarını temsil etmek DDİ'nin zorlu bir alanıdır. Mevcut makine öğrenimi algoritmalarının, yapılandırılmamış metin verileri uygulandığında genelleme sınırlamaları gibi belirli dezavantajları vardır. Örneğin, kelime torbaları gibi geleneksel metin temsil yöntemleri, kelimeleri tek bir vektör olarak temsil eder, bu da atomik temsiller nedeniyle kelimeler arasındaki anlamsal ilişkilerin kaybolmasına neden olur. Ayrıca, cümleler, bir dilin karmaşık, bileşik semantiğini temsil etmek için yeterince iyi olmayan, geniş bir kelime dağarcığındaki kelime sayılarına dayanan yüksek boyutlu seyrek kodlamalar olarak temsil edilebilir. Bu nedenle, bileşik temsil öğrenimindeki boşluğu doldurmak için çok daha fazla çalışma yapılmalıydı.

Şimdiye kadar, duygu analizi ve metin temsili öğrenme çalışmalarının çoğu İngilizceye odaklandı. İlk çalışma Pang ve diğerleri tarafından yapılmıştır [1]. Pang ve diğerleri bu çalışmada kelime torbası özelliklerinin kullanılmasını ve Destek Vektör Makinesi (DVM) ve Naive Bayes (NB) algoritmaları kullanılarak incelemelerin sınıflandırılmasını önerdi. Bu çalışmayı, temel bir çalışma olarak çoğu makine öğrenimi tabanlı yaklaşım izlemiştir ve n-gramlar, TF-IDF, konuşma etiketleme (POS) gibi özellikler üzerinden veri kümesini öğrenmek için başka birçok MÖ yöntemi geliştirilmiştir.

Dilbilimciler, doğal bir dildeki kelime sayısının sınırlı olmasına rağmen, doğal dili konuşanlar tarafından sonsuz sayıda cümle oluşturulabileceğini belirtirler. Dil bilgisi kuralları, dil oluşturmanın doğal bir yolu olarak kullanılır. Dilin bileşik anlamını (yani tümceler, cümleler ve cümleler için dağıtılmış temsiller) belirlemek, dil atomik, ayrık değerler ve dinamik bilgi akışlarını tutamayan sembolik veri yapıları olarak temsil edildiğinden çok zordur. Hem tek vektör olarak kodlamalar hem de kelime yerleştirmeleri, en az iki öğenin bağlantı anlamlarını modelleme konusunda sınırlı yeteneğe sahiptir ve yapısal veya sıralı olarak birleştiremedikleri için verilen cümlelerin anlamsal bilgilerini yakalayamazlar.

İnsan düşüncesi ve konuşması doğrusal olmayan bir şekilde gerçekleşir ve dil, insan zihninin bütünleşme ve yorumlama yeteneğinin bir sonucu olarak zihinde birbiriyle ilişkili süreçler tarafından özyinelemeli olarak üretilir. Bu yapısal bilgiyi temsil etmek, insan zihni ile dil arasında yinelemeli olarak işlenen doğal dilin bileşik semantiğini anlamak için çok önemlidir. Geleneksel yöntemler, cümlelerin sıradan bir lineer yapı, yani zincirler veya diziler olduğunu iddia eder. Bununla birlikte, verilerin bağlantılı yapısını tespit etmenin ve onu optimum bir yapı içinde temsil etmenin en uygun yolunu bulmak hala açık bir sorundur.

DÖ, gizli katmanları artırarak üst düzey ve soyut özelliklerin otomatik olarak öğrenilmesini sağlar ve DDİ görevinin optimizasyon prosedürü anlamında eğitimi kolaylaştıran verilerin daha iyi temsillerini sağlar. Derin katmanlar aracılığıyla, görev, geleneksel makine öğreniminin zaman öldürücü özellik mühendislik adımları olmadan yapılabilir. Daha iyi temsillerin kullanılması, yani yoğun kelime yerleştirme, aynı zamanda, kelime sırasını göz ardı eden bir vektör uzayında atomik

birimler gibi kelimelerin temsil edilmesinin bir sonucu olarak ortaya çıkan yüksek boyutlarla başa çıkmak için etkili bir yol olabilir. DÖ çağıyla birlikte, Sinirsel Dil Modelleri (SDM), büyük veri setlerinin ve iyileştirmelerin katkısıyla anlambilim ve duygu analizi görevlerinde çarpıcı başarılar elde etmeye başlamıştır. Tüm öğrenme modellerinin başarısının ardındaki sihir, verilerin bilgi kaybetmeden ne kadar iyi temsil edildiğine bağlıdır. Özyinelemeli Sinir Ağları, belirli bir cümledeki anlamsal semantiği yakalayabildikleri için duygu analizi için geliştirilmiştir [2, 3]. Yinelenen Sinir Ağlarının (RNN) standart versiyonu, yani Yinelenen Otomatik Kodlayıcılar (RAE), Socher ve diğerleri tarafından tanıtıldı [4]. Bu çalışmada dili, karakterlerden kelimelere ve kelimelerden cümlelere özyinelemeli bir şekilde akan sürekli bir anlam mimarisi olarak anlamak için otomatik kodlayıcılar kullanılmıştır.

Morfolojik olarak zengin diller (MZD) için dil modellerine odaklanan araştırmalar hala gelişmektedir ve geleneksel yaklaşımların çok ötesine geçmemiştir. Bununla birlikte, bu yaklaşımlar, kelimeleri bağımsız atomik birimler olarak temsil etmeleri nedeniyle, kelimeler arasında açık bilgi kaybına neden olan ve doğruluğu azaltan bir performans darboğazına sahiptir. Bu nedenle, dillerin doğrusal olmayan yapılarını öğrenmek için yeni dil temsil modellerinin geliştirilmesi gerekmektedir. Ayrıca Türkçe gibi eklemeli yapılı diller için daha az çalışma yapılmakta ve başarı oranları İngilizce çalışmalara göre daha düşüktür. İngilizce için geliştirilen araçlar Türkçe gibi MZD'de doğrudan kullanılamamaktadır.

Türkçe morfolojik olarak zengin bir dildir. Her kelime köküne çeşitli ekler eklenerek birçok farklı kelime türetilebilir. Dolayısıyla farklı kelimelerin sayısı artar, alan boyutu çok büyür ve işlem süresi çok fazla zaman alır. Ayrıca duygu analizi sırasında kullanılacak veriler ham veri olduğu için veri seti üzerinde ön hazırlık aşamasında ham verilerdeki birçok gürültü ve yazım hatası düzeltilmektedir. Ancak bu duygu ifade eden bazı ifadelerin kaybolmasına neden olabilir. Yaygın olarak kullanılan kelime torbası modelini kullanırken geniş alanlarla çalışmak gerekir.

Geleneksel DDİ süreçleri genel olarak parçalama (tokenizasyon), söz dizimi tabanlı kök bulma (stemming - köklendirme) veya anlam tabanlı kök bulma (lemmatizasyon - köklendirme) gibi temel ön işleme görevlerinden ve özellik mühendisliği görevinden oluşur. Bu görevler özellikle, yaygın olarak araştırılan bir

dil olan İngilizcenin aksine morfolojik karmaşıklıklarından dolayı MZD için (Türkçe, Arapça, Çekçe, Macarca, Fince, İbranice, Korece vb.) kolay değildir [5]. Bahsedilen ön işleme adımları, özellikle köklendirme, lemmatizasyon işlemleri bu dillerin eklemeli veya çekimsel morfolojileri nedeniyle karmaşık analiz görevlerine ihtiyaç duyar. Örneğin Türkçede aynı kökten ortalama 5 farklı kelimenin türetildiği açıkça görülmektedir [6] ve bu durum kök analizi için bir sorundur. Bahsedilen ön işleme adımları ve dil işleme görevleri Makine Öğrenmesi (MÖ) algoritmalarının performansını etkilemektedir [7, 8, 9]. Çift yönlü öğrenme anlayışıyla geliştirilen çift yönlü dil modellerinden olan Transformatörlerden Çift yönlü Kodlayıcı Temsilleri (BERT) [10], morfolojik olarak karmaşık kelimelerin bağlamsal anlamını [11, 12, 13] dil işleme görevleri işletilmeden kavramak için bir çözüm olarak görünmektedir.

Bu çalışma, geleneksel MÖ algoritmaları ile yeni önerilen BERT algoritmasını deneysel başarı olarak karşılaştırır. İki yaklaşımın performanslarını değerlendirmek için literatürden Türk film ve otel incelemelerinin duygu analizi [14] veri seti seçilmiştir. Bu çalışmanın sonuçları BERT sinirsel modelinin diğer gerçek dünya DDİ görevlerine uygulanabilirliği açısından özellikle Türkçe için MZD araştırmacılarına yardımcı olacaktır.

Tez metninin sonraki bölümleri şu şekilde özetlenebilir:

İkinci bölümde MÖ, Derin Öğrenme (DÖ) konuları temel alt başlıklarıyla birlikte açıklanmış ve bu algoritmaların performans değerlendirme ölçütlerinden bahsedilmiştir. DDİ süreçleri hakkında bilgi verilerek DA kavramı tanıtılmıştır. Son olarak MZD'den olan Türkçenin modelleme zorlukları anlatılmıştır.

Üçüncü bölümde ilk olarak problemin tanımı yapılmış ve önerilen yöntemin mimarisi detaylı olarak anlatılmıştır. Sonrasında uygulamada kullanılan veri kümeleri ve veri kümelerinin elde edilişleri anlatılmıştır. Geleneksel yöntemleri içeren MÖ deneyleri sonuçları belirtilerek önerilen yöntemin deneyleri üzerinde durulmuştur. Son olarak iki yaklaşımın karşılaştırılması yapılarak sonuçların değerlendirilmesine yer verilmiştir.

Dördüncü bölümde duygu analizi problemi hakkında yapılan çalışmalara değinilmiştir. Ayrıca bu bölümde, bu çalışma ile literatüre yapılan katkı ele alınmış ve gelecek çalışmalar için öneriler sunulmuştur.

BERT'in geleneksel Türkçe analiz yöntemlerine göre etkinliğini göstermek için deneyler yapılan bu çalışma ile, BERT kullanımı ile Türkçe dil çözümlemesi için yeni bir yaklaşım sunulmuştur. Bu çalışma ayrıca yeni sinirsel dil modeli olan BERT'in Türkçedeki etkinliğini doğrulamak için yapılmış en büyük deneysel çalışmadır.

2. BİLİMSEL ALT YAPI

2.1. Makine Öğrenmesi

MÖ insanların öğrenme şeklini taklit etmek için verilerden yararlanan, algoritmaların bu yönde kullanımına odaklanan ve doğruluğunu kademeli olarak arttıran bir Yapay Zekâ (AI) ve bilgisayar bilimi dalıdır. Yapay Zekânın en büyük amacı olan makinelerin insanlar gibi anlaması ve öğrenmesi ancak MÖ ile gerçekleştirilebilir. MÖ algoritmaları tahminlerde bulunmak ya da kararlar almak için eğitim verileri olarak bilinen örnek verilere dayalı bir model oluşturur. MÖ ile birçok problem çözülebilmektedir. En yaygın problemler sınıflandırma, kümeleme, segmentasyon, anomali tespiti, filtreleme, yoğunluk kestirimi ve sentezlemedir. MÖ yöntemleri denetimli ve denetimsiz yöntemler olmak üzere iki kategoride incelenmektedir.

2.1.1. Denetimli ve Denetimsiz Öğrenme

Denetimsiz makine öğrenimi olarak da bilinen Denetimsiz Öğrenme, etiketlenmemiş veri kümelerini analiz etme ve kümeleme amacıyla algoritma geliştirilmesi yöntemidir. Bu tipte geliştirilen algoritmalar insanın dışardan müdahalesini gerektirmeden veri içerisindeki gizli kalıpları veya veri gruplanmalarını keşfeder. Verilerdeki benzerlik ve farklılıkları keşfetme yeteneği ile literatürde Keşifçi Veri Analizi olarak geçen çok değişkenli istatistiksel analiz ve veri görselleştirme yaklaşımları kullanılarak göz ile yakalanamayacak yapıların ortaya çıkarılması analizi için de ideal bir çözüm yaklaşımıdır. Denetimsiz öğrenmede bir başka yöntem olarak veri kümesi içerisindeki benzer örnekleri bulup bir araya getirerek veri kümesini gruplama/kümeleme örnek gösterilir. Çapraz satış stratejileri, müşteri segmentasyonu, görüntü ve örüntü tanıma gibi birçok gerçek dünya problemi denetimsiz öğrenme ile çözülebilmektedir. Denetimsiz makine öğrenmesinde kullanılan bazı algoritmalar arasında Yapay Sinir Ağları (YSA), k-ortalama kümeleme, olasılıklı kümeleme ve daha fazlası bulunur.

Denetimli makine öğrenimi olarak da bilinen Denetimli Öğrenme, verileri sınıflandırmak veya sonuçları doğru olarak tahmin etmek amacıyla algoritmaları

eğitirken etiketli veri kümelerinin kullanılması şeklinde tanımlanır. Denetimli öğrenme e-postaların gerekli veya gereksiz olarak sınıflandırılmasından, el yazısı ile yazılmış bir rakam görüntüsünden hangi rakam olduğunun tahmin edilmesine kadar birçok gerçek dünya probleminin büyük ölçekte çözülmesini sağlar. Literatürde regresyon olarak adlandırılan sayısal bir etiketin tahmin edilmesini içeren problemler ya da herhangi kategorik bir sınıf etiketinin tahmin edilmesini içeren sınıflandırma problemleri denetimli öğrenme ile çözülebilmektedir. Hem sınıflandırma hem de regresyon problemleri bir ya da daha fazla girdi değişkenine sahip olabilir. Bu girdi değişkenleri sayısal ya da kategorik olabilir. Denetimli makine öğrenmesinde kullanılan bazı algoritmalar arasında doğrusal/lojistik regresyon, karar ağaçları, Naive Bayes, Destek Vektör Makinaları (DVM), Rastgele Orman algoritmaları ve daha fazlası bulunur.

2.1.2. Naive Bayes Algoritması

Naive Bayes Algoritması (NB) adını Matematik bilim dalı içerisindeki olasılık kuramında incelenen önemli bir teorem olan Bayes Teoremi'nden almıştır. NB veri kümesindeki girdiler arasında bağımsızlık varsayımları ve Bayes Teoremi uygulanmasına dayanan olasılıksal bir sınıflandırma algoritmasıdır. Kolay uygulanabilirliğine ve sadeliğine rağmen NB şaşırtıcı derecede iyi sonuç verir ve genellikle daha karmaşık yapıda olan sınıflandırıcılardan daha iyi performans gösterdiği için sıklıkla kullanılır. En yaygın olarak tıbbi hastalık teşhisi, metin sınıflandırma gibi problemlerde kullanılır.

Bayes Teoremi bir olayın meydana gelmesinde birden fazla etken var ise, olayın hangi etkenin etkisi ile ortaya çıktığını gösteren teoremdir. Bayes teoremi, P(c), P(x) ve P(x|c)'den sonsal olasılığı, P(c|x)'i hesaplamanın bir yolunu sağlar. NB sınıflandırıcısı, bir tahmin edicinin (x) değerinin belirli bir sınıf (c) üzerindeki etkisinin diğer tahmin edicilerin değerlerinden bağımsız olduğunu varsayar [15].

$$P\ (c \mid x) = \frac{P(x \mid c\ )\ P(c)}{P(x)} \qquad \textbf{(2.1)}$$

$$P(c \mid x) = P(x_1 \mid c) \times P(x_2 \mid c) \times \cdots \times P(x_n \mid c) \times P(c) \quad \textbf{(2.2)}$$

- $P(c \mid x)$, x eğitim verisi verildiğinde c hipotezinin gerçekleşme olasılığı
- $P(c)$, c hipotezinin önsel gerçekleşme olasılığı
- $P(x \mid c)$, c olayı verildiğinde x eğitim setinin koşullu olasılığı
- $P(x)$, eğitim verisindeki herhangi bir örneğin x olma olasılığı

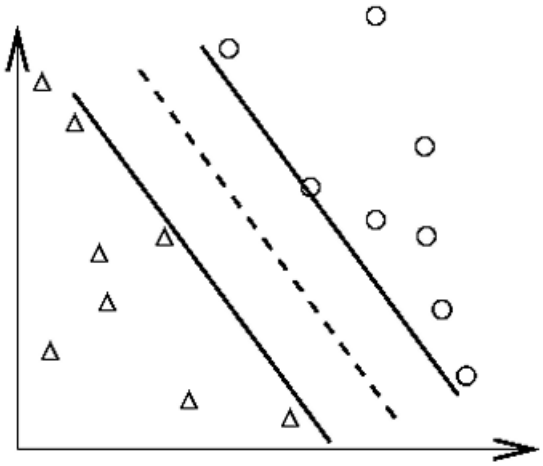
NB sınıflandırıcısı veri kümesinde yer almayan, daha önce görülmemiş, sınıflandırılmamış bir örneğin en olası sınıfını bulmak için tüm veri kümesinde olasılık teoremini işletir. NB sınıflandırma yaklaşımına göre veri kümesindeki örnekleri tanımlayan değişkenler/öznitelikler birbirleri ile verilen hipoteze göre şartlı bağımsızlardır. NB genellikle eğitim verisinde kategorik değişkenler bulunuyorsa güçlü ancak sayısal veriler bulunuyorsa daha zayıf bir performans sergilemektedir.

2.1.3. Destek Vektör Makineleri

Destek Vektör Makineleri (DVM) algoritması 1995 yılında Cortes ve Vapnik tarafından geliştirilmiştir. DVM, istatistiksel öğrenme teorisine dayanmakta olup denetimli öğrenme için en temel yaklaşımlardan biridir. Algoritma veri kümesi ile ilgili herhangi bir ön bilgi varsayımı yapmadan eğitim veri kümesi üzerinde öğrenme gerçekleştirerek, yeni veriyi tahmin etmeye çalışır. Eğitim veri kümesindeki girdi değişkeni ile onun çıktısı birbiri ile eşlenir. Böylelikle veri çiftleri oluşturulmuş olur. Bu veri çiftleri kullanılarak farklı sınıflardan diğer verileri ayıran karar fonksiyonları elde edilir. Eğitim verisinde olmayan, daha önce görülmemiş yeni veri bu karar fonksiyonuna göre sınıflandırılır.

DVM çalışma prensibi, verileri iki kategoriden oluşan hedef değişkendeki bu iki kategoriyi optimal şekilde ayıran karar sınırlarının (ayırıcı düzlem) belirlenmesine dayanır. DVM için en temel sınıflandırma problemi doğrusal olarak ayrılabilen iki kategorili bir hedef değişkeni olan veri kümesinin sınıflandırılmasıdır. Bu sınıflandırmayı yapabilmesi için DVM, iki kategori arasındaki ayrımı en iyi yapan ve

sınıflar arasındaki sınırı maksimum yapan en optimal ayırıcı düzlemi belirlemeye çalışır. En optimal ayırıcı düzlem, her sınıf için verilerin ayırıcı düzlemden uzaklığını maksimize eder. Bu ayırıcı düzlemlere en yakın eğitim verileri, iki sınıf arasındaki sınırı belirleyen destek vektörlerini oluşturur.



Şekil 2.1. Destek vektör makineleri ile ikili sınıflandırma temsili

DVM'in amacı öznitelikleri ile birlikte verilen test verisinin hedef değerlerini eğitim verilerine dayalı olarak tahmin eden bir model üretmektir. İkili sınıflandırma problemlerinde çalışma prensibi basitçe Şekil 2.1'de yer almaktadır. Şekilde görülen en optimal bulunmuş ayırıcı düzlemlerdir.

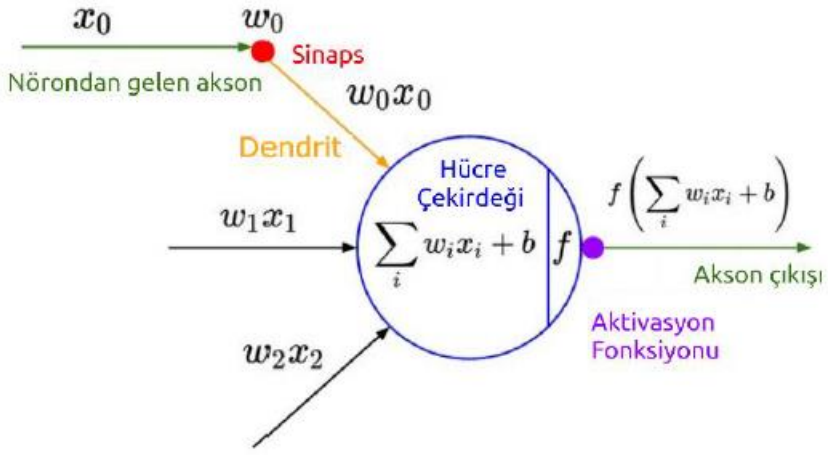
2.2. Derin Öğrenme

DÖ, Yapay Sinir Ağları (YSA) olarak adlandırılan insan beyninin yapısı ve işlevinden ilham alarak oluşturulmuş bir modele dayanan makine öğrenmesinin bir alt dalı olarak tanımlanır. YSA insan beyninden esinlenilerek insan sinir hücresi olan nöronların matematiksel olarak modellenmesi ile ortaya çıkmıştır. Konu ile ilgili ilk çalışmalar nöronların modellenerek bilgisayar sistemlerine uyarlanmasıyla başlamıştır. Bilgisayar sistemlerinin gelişmesi ile kullanımları oldukça yaygınlaşmıştır. DÖ oldukça geniş bir literatüre sahip olup son yılların en verimli araştırma alanı olarak görülmektedir.

2.2.1. Sinir Ağları

İnsan beyninden ilham alınarak oluşturulan YSA bir matematiksel modellemeye dayanır. Sinir hücresi olan nöronların çalışma biçiminin matematiksel olarak modellenmesi yapay sinir hücresini oluşturur. Yapay sinir hücreleri arasındaki insan beynindeki nöronlara benzer bir iletim sonucunda yapay sinir ağları oluşmuştur. Nöronların temel çalışma prensibi, kendilerine gelen sinyali aldıktan sonra, sinyal ile gerekiyorsa bir değişime yol açmak ve çıktıyı diğer nöronlara aktarmak şeklindedir. Yapay nöronlar da bu çalışma şekline göre sinir ağlarını oluşturmaktadır. Bir yapay nöron girdi olarak aldığı bilgiyi belirlenmiş bir ağırlık ile çarpar ve ek başka bir bilgi ile toplayarak çıktı olarak aktarır. Çıktı, daha önce belirlenmiş bir aktivasyon fonksiyonundan geçirilerek nöronun aktiflik durumu belirlenir. Aktivasyon fonksiyonundan geçtikten sonra elde edilen çıktı sinir ağının nihai sonucunu verebilir ya da bir başka nörona girdi olabilir.

Biyolojik nöronlar dendrit şeklinde ifade edilen sinyal toplayıcı uçlardan girdi sinyalini alır, hücre çekirdeğinde bu sinyal ile gerekli işlemleri yapar ve sonucu akson(gövde) boyunca taşıyarak sonuçta bir çıktı sinyali oluşturur. Biyolojik nöronlardan ilham alınarak oluşturulan bu işlemin matematiksel modellemesi Şekil 2.1'de gösterilmiştir.

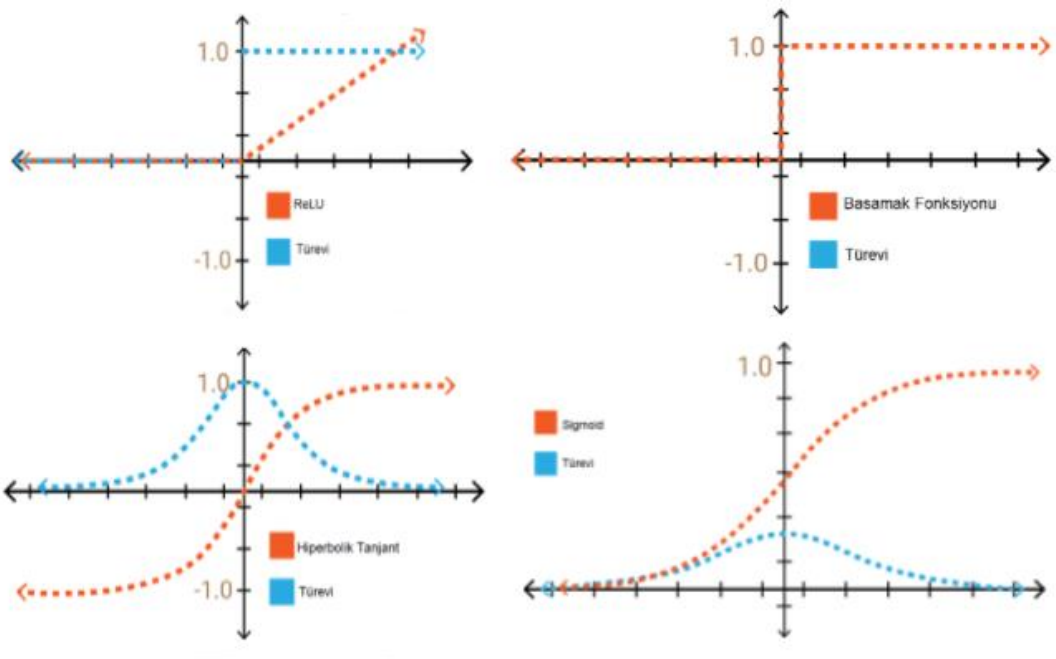


Şekil 2.2. Biyolojik nöron matematiksel modellemesi (University of Stanford, n.d.)

(x_0) girdi, (w_0) girdinin ağırlığı, (b) ek girdi, (f) aktivasyon fonksiyonu olmak üzere yapay nörona gelen tüm girdiler ağırlıklar ile çarpılıp ek girdiler ile toplanarak aktivasyon fonksiyonuna verilir. Akson çıkışına iletilen çıktı aktivasyon fonskiyonunun sonucudur. Yapay nöronda yapılan bu işlem Denklem 2.3 ile ifade edilebilir.

$$y = f\left(\sum(x_i * w_i + b)\right) \tag{2.3}$$

Buradaki aktivasyon fonksiyonu YSA'yı lineer problem çözen algoritmalardan ayıran etkendir. Doğrusal olmayan bir problemi modelleyebilmek için doğrusal olmayan aktivasyon fonksiyonlarına ihtiyaç vardır. Aktivasyon fonksiyonları seçilirken hesaplama kolaylığı baz alındığından türevi kolay alınabilen fonksiyonlar seçilmektedir. Aktivasyon fonksiyonlarının çıkışları genellikle [0, 1] aralığında ya da [-1, 1] aralığındadır. Şekil 2.2'de bazı aktivasyon fonksiyonları gösterilmiştir.



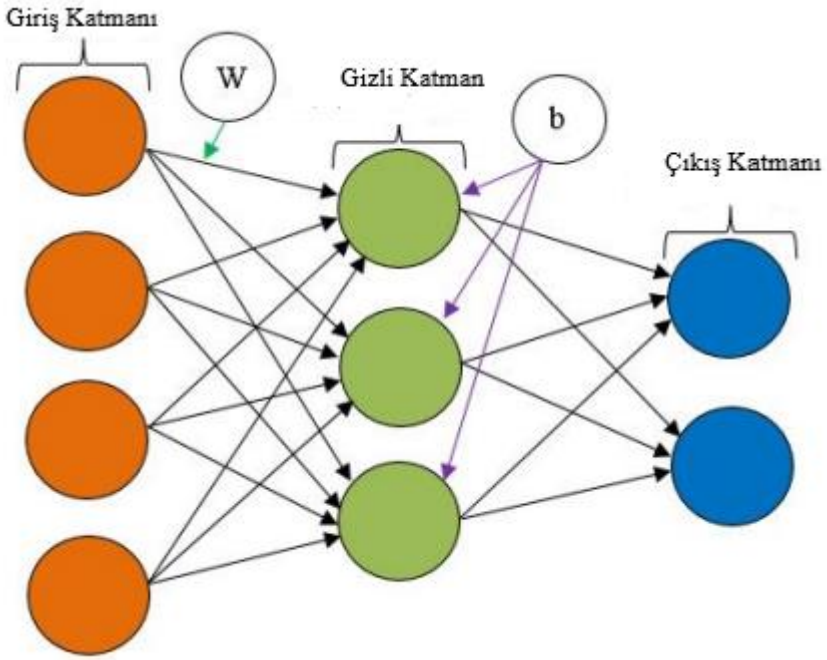
Şekil 2.3. Sıklıkla kullanılan aktivasyon fonksiyonları

ReLU aktivasyon fonksiyonu parçalı lineer bir fonksiyon olup $[0, +\infty]$ aralığında değer almaktadır. Kendisine verilen giriş değeri pozitif ise girişi doğrudan çıktısı olarak verecektir. Eğer kendisine verilen giriş pozitif değil ise sıfır çıkışını verecektir.

Basamak fonksiyonu aktivasyon fonksiyonlarının en basit türlerinden biridir. Bu aktivasyon fonksiyonunda bir eşik değer göz önünde bulundurulur. Kendisine verilen giriş değeri bu belirlenen eşik değerinden yüksekse nöron aktifleştirilir. Genellikle ikili sınıflandırma problemleri çözümü için kullanılmaktadır.

Sigmoid fonksiyonu [0, 1] aralığında değer alır. Sıklıkla kullanılan bu aktivasyon fonksiyonu S şeklinde bir eğriye sahiptir. Hiperbolik tanjant fonksiyonu da sigmoid fonksiyonuna benzer olup yalnızca değer aralığı farklıdır. Hiperbolik tanjant fonksiyonu [-1, 1] aralığında değer alır.

Yapay nöronların birbirleri ile iletişimi sonucunda yapay sinir ağları meydana gelmektedir. YSA mimarisi basitçe Şekil 2.3'de yer almaktadır.



Şekil 2.4. Temel YSA mimarisi [16]

YSA'ya verilen girdiler, içerisinde aldığı ek girdiler ve aktivasyon fonksiyonunun sonucunda bir nöronun çıktısı hesaplanır. Bu çıktı bir başka nöronun girdisi olabilir. Yapay nöronlar bu şekilde bir sinir ağı oluştururlar. Oluşturulan bu sinir ağının kapasitesi arttıkça daha büyük bir hipotez uzayına sahip olur ve daha zorlu problemlerin çözümüne imkan sağlar [17]. Hipotez uzayı bir modelin ifade edeceği çözüm kümesidir [18].

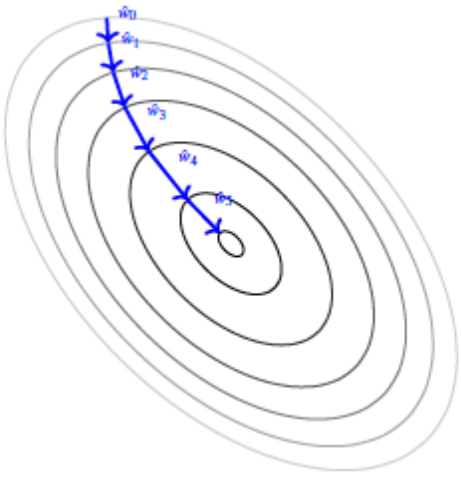
2.2.2. Eğitim: Geri Yayılım Algoritması

Bir öğrenme modelinin eğitimi, modelin öğrenilebilir parametrelerinin optimum değerlerini oluşturmayı amaçlar. Sinir ağı modelleri için eğitim geri yayılım algoritması kullanılarak yapılır. Algoritma sinir ağını etkili bir biçimde eğitmek amacıyla kullanılır. Bir YSA üzerinde her bir ileri gidiş sonrası, geri yayılım algoritması, modelin parametrelerini ayarlamak için geriye doğru bir geçiş gerçekleştirir.

Sinir ağında bir ileri gidişte, ilk girdiler ve ek girdiler aktivasyon fonksiyonu çıkışları ile tüm ağın çıktı sonucu elde edilir. Geri yayılım, bu ağın gerçek çıktı vektörü ile istenen çıktı vektörü arasındaki farkı en aza indirmek için ağdaki bağlantıların ağırlıklarını tekrar tekrar ayarlar. Kullanışlı yeni ağırlıklar oluşturma yeteneği, geri yayılımı, daha önceki daha basit şekilde uygulanan diğer yöntemlerden belirgin şekilde ayırmaktadır [19].

Geri yayılım algoritması kullanılan veri kümesi üzerinde en az hatayı vermek için modelde kullanılan parametreleri değiştirir. Bu parametreler Şekil 2.3'te gösterilen w ve b parametreleridir. Bu değiştirme işlemi aslında bir optimizasyon problemi olup farklı birçok teknikle gerçekleştirilmektedir. Ağın gerçek çıktı vektörü ile istenen çıktı vektörü arasındaki fark maliyet/kayıp fonksiyonu olarak adlandırılır. Geri yayılım algoritması bu fonksiyonu minimize etmeye çalışır. Geri yayılım algoritmalarına Gradyan İniş Algoritması bir örnek olarak verilebilir. Gradyan iniş YSA optimize etmede bilinir bir algoritmadır. Ayrıca bu algoritma genellikle kara kutu bir eniyileme yöntemi olarak kullanılmaktadır. Gradyan iniş, maliyet fonksiyonunu minimuma indirmeye çalışırken fonksiyonun gradyanlarını kullanan

bir yöntemdir. Gradyan iniş algoritmasının hipotez uzayında model parametrelerini optimize etme karakteristiği basitçe Şekil 2.5'te yer almaktadır.



Şekil 2.5. Gradyan iniş algoritması temsili optimizasyon karakteristiği

Gradyan iniş için uzayda rastgele bir nokta seçilir. Her iterasyonda gradyanlara bağlı olan ağırlıklar (*w* değerleri) minimuma indirilmeye çalışılarak güncellemeye gidilir. Bu azaltma işlemi sonsuza kadar sürmez ve belirli bir noktaya gelip sonlanmak durumundadır.

Gradyan inişi uzayda her zaman bu şekilde düzgün bir iniş sağlayamayabilir ve zikzak çizerek çok daha küçük adımlar atarak da inebilir. Bu tip farklı iniş yöntemleri de Stokastik Gradyan İniş, Mini – Yığın Gradyan İniş gibi yöntemleri ortaya çıkarmıştır. Gradyan inişi algoritmasının iyi bilinen DÖ kütüphanelerinde birçok farklı versiyonunun uygulamaları bulunmaktadır.

2.3. Algoritma Performansı Değerlendirme Ölçütleri

Herhangi bir MÖ probleminin çözümünde kullanılan algoritmaların performansı belirli metriklere göre belirlenir. Bu metrikler literatürde model başarı ölçütleri olarak yer almaktadır. En yaygın olarak kullanılan model başarı ölçütleri Doğruluk (Accuracy, ACC), Kesinlik (Precision), Duyarlılık (Recall) ve F1-Ölçütü (F1-measure)'dür [20].

Karmaşıklık matrisi, gerçek değerlerin bilindiği bir test verisinde sınıflandırma modelinin performansını göstermek için yaygınlıkla kullanılan bir tanımlama tablosudur.

Tablo 2.1. Karmaşıklık Matrisi

Karmaşıklık Matrisi	Gerçek Pozitif (1)	Gerçek Negatif (0)
Tahmin Pozitif (1)	TP	FP
Tahmin Negatif (0)	FN	TN

Tablo 2.1'de yapısı gösterilen bu tablodaki değerler sırasıyla, gerçek ve tahmin değerinin pozitif olduğu TP (true-positive), gerçek ve tahmin değerinin negatif olduğu TN (true-negative), gerçek değerinin negatif tahmin değerinin pozitif olduğu FP (false-positive), gerçek değerinin pozitif tahmin değerinin negatif olduğu FN (false-negative)'dir. Model başarı ölçütleri bu karmaşıklık matrisindeki değerlerden hesaplanan bazı oranlardan oluşurlar.

$$Doğruluk\ (Acc) = \frac{TP + TN}{TP + TN + FP + FN} \quad \textbf{(2.4)}$$

$$Duyarlılık\ (R) = \frac{TP}{TP + FN} \quad \textbf{(2.5)}$$

$$Kesinlik\ (P) = \frac{TP}{TP + FP} \quad \textbf{(2.6)}$$

$$F1 - Ölçütü\ = \frac{2 \times P \times R}{(P + R)} \quad \textbf{(2.7)}$$

MÖ algoritmalarının performansını ve istatistiksel doğrulamasını değerlendirmek için kullanılan bir başka model başarı ölçütü olan Matthews Korelasyon Katsayısı (MCC) da karmaşıklık matrisindeki değerlerden oluşmaktadır. Denklem 2.8'de ifade edildiği gibi hesaplanan MCC, karmaşıklık matrisinin dört sonucunun (TP, TN, FP, FN) tümü ile tahmin ettiği için kapsamlı bir ölçümdür.

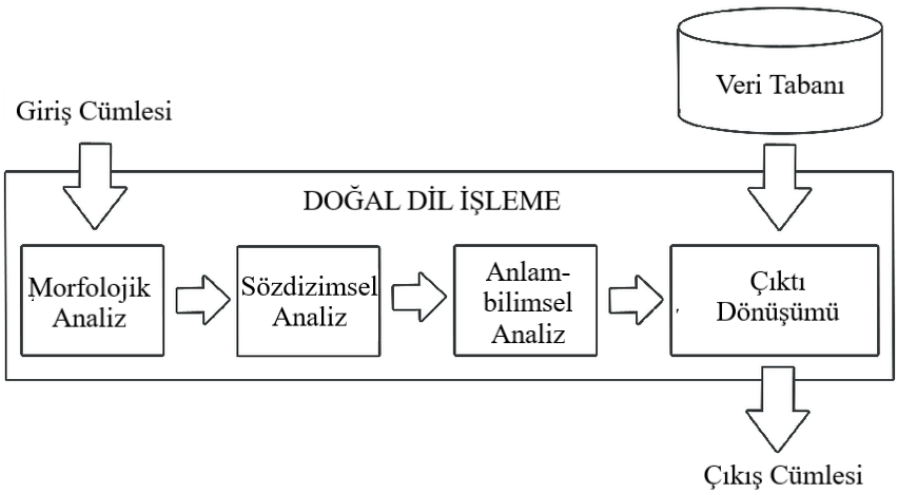
$$MCC = \frac{(TP \times TN) - (FP \times FN)}{(TP \times FP) \times (TP \times FN) \times (TN \times FP) \times (TP \times FN)} \quad \textbf{(2.8)}$$

2.4. Doğal Dil İşleme

Doğal Dil İşleme (DDİ) doğal dillerin işlenmesi ve anlaşılmasında kullanılacak olan bilgisayar sistemlerini tasarlayan ve uygulayan bir bilim ve mühendislik alanıdır. DDİ insanların kendi aralarında anlaşmak için kullandıkları dili, insan-bilgisayar etkileşimini en üst düzeye çıkarabilmek veya farklı doğal dilleri kullanan insanlar arasındaki iletişimi kolaylaştırarak güçlendirmek üzere çözümler üretir. DDİ, bilgisayar mühendisliği, yazılım mühendisliği, hesaplamalı dilbilim vb. birçok disiplinde kullanılabilir. İnsan ifadeleri ile yapay zekâ arasındaki boşluğu doldurmayı sağlar. Bilişim teknolojilerinde gerçekleşen hızlı gelişim literatürde doğal dilleri ele alan çalışmalara da ivme kazandırmıştır. DDİ problemlerinin çözümünde MÖ, DÖ, istatistiksel analiz ve kural tabanlı yaklaşımlar kullanılmaktadır. Yazım yanlışlarının düzeltilmesi, otomatik çeviri sistemlerinin oluşturulması, metinlerin sınıflandırılması, spam e-postalarının tespit edilmesi, metin özetleme, konuşma etiketlenmesi, metinden duygu analizi uygulamaları gibi birçok farklı problem üzerinde DDİ ile çalışılmaktadır.

2.4.1. Doğal Dil İşleme Aşamaları

DDİ uygulanması morfolojik, sözdizimsel, anlambilimsel ve pragmatik analiz adımlarından oluşmaktadır. DDİ süreci basitçe Şekil 2.6'da yer almaktadır. Bilgisayar sistemleri; kelime kökünü morfolojik analiz ile, kelimelerin dizilmesini sözdizimsel analiz ile, cümle bütününün anlamını anlambilimsel analiz ile çıkarır. Bu aşamalar sonrasında yapılan en son analiz ise metnin bütününün analizi olan pragmatik analizdir.



Şekil 2.6. DDİ süreci

2.3.1.1. Morfolojik Analiz

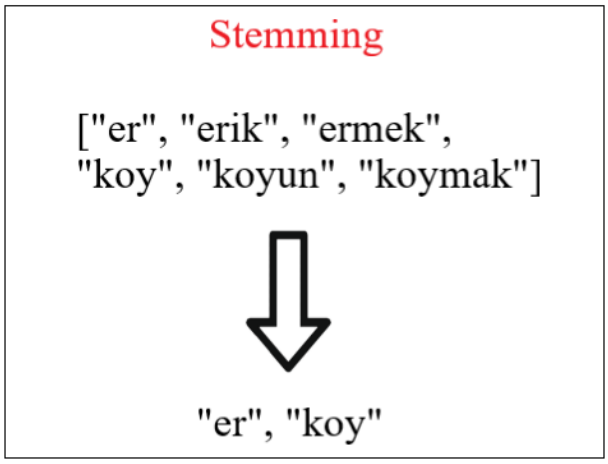
Morfolojik analiz, DDİ'nin ana süreçlerinin başlangıcı kabul edilmektedir. Bu aşama en küçük anlam birimleri olan biçimbirimlerden oluşan sözcüklerin bileşen doğasıyla ilgilenir. Bir kelime üç ayrı biçimbirim (ön ek, kök, son ek) halinde analiz edilebilir. Morfolojik analiz aşamasında ilk olarak cümle içerisindeki boşluklar baz alınarak cümle kelimelere ayrılmaktadır. İkinci adım olarak bulunan her kelime de üç ayrı biçimbirime göre tekrar bir ayrışmaya tabi tutulmaktadır. Bu sayede morfolojik analiz sonucunda her bir kelime için bu kelimenin kullanılabildiği tüm durumlar bulunarak bir kelime için birden çok sonuç elde edilebilmektedir. Morfolojik analiz aynı zamanda kelimenin bulunan birden fazla kullanımları arasında en yaygın olanını da tespit edebilmekte ve çıktı olarak bu seçeneği en üstte göstermektedir.

Parçalama (tokenizasyon), metnin parçalara ayrılması işlemi bu aşamada gerçekleştirilen bir işlemdir. Bu işlem bir cümleyi daha küçük anlamlı birimlere ayırmayı amaçlar. Parça (token) anlamlı küçük birimlere verilen isim olup, semboller, kelimeler ve deyimler de Token olarak ifade edilebilir. Parçalama işlemi farklı ayrıştırmalar ile gerçekleştirilmektedir. Bu ayrıştırmalara cümle içerisindeki kelimeleri boşluklara göre ayıran ve noktalama işaretlerine göre ayıran, paragrafı cümlelere ayıran vb. örnek olarak gösterilebilir. Cümleyi noktalama işaretlerine ve boşluklara göre ayıran bir ayırıcı ile uygulanan parçalama işlemi basitçe Şekil 2.7'de yer almaktadır.

Şekil 2.7. Örnek bir tokenizasyon işlemi

Söz dizimi tabanlı kök bulma işlemi (Stemming) ve anlam tabanlı kök bulma işlemi (Lemmatizasyon) da morfolojik analiz aşamasında gerçekleştirilen işlemlerdir. Hem stemming hem de lemmatizasyon, kelimelerin çekim veya türetme biçimlerini ortak bir temel biçime indirgeme amacına sahiptir.

Şekil 2.8'de Türkçe kelimeler için örnek bir stemming işlemi yer almaktadır. Tablo 2.2'de bu kelimelerin anlamları verilmiştir.

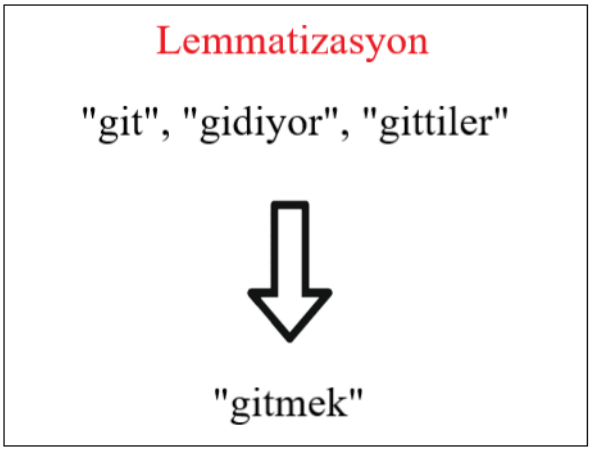


Şekil 2.8. Örnek bir stemming işlemi

Tablo 2.2. Örnek Türkçe kelimelerin sözlük anlamları

Kelime	**Türkçe anlamı**
er	Ordudaki rütbesiz olan asker
erik	Mayhoş veya tatlı olan mevyesi yenen bir ağaç türü
ermek	Bir şeye ulaşmak, erişmek
koy	Küçük boyutlu körfez
koyun	Küçükbaş bir hayvan türü
koymak	Belirli bir şeyi yerine bırakmak, yerleştirmek

Tablo 2.2'de belirtilen kelimelerin anlamlarına bakıldığında her örnek kelimenin farklı anlama geldiği görülmektedir. Stemming işlemi köklendirme yaparken "er", "erik", "ermek" kelimelerinin hepsi için kökü 'er' olarak belirlemektedir. Ancak "erik" kelimesinin kendisi çekim veya yapım eki almadığından yalın haldedir ve tek başına bir kök (root) oluşturmaktadır. Köklendirme kelimeler arasında ortak bir payda bulmaya çalışırken, fazla kırpma ya da az kırpma gibi bazı dezavantajlara sahiptir. Lemmatizasyon kelimenin morfolojik olarak köküne inilmesi işlemidir. Kelimenin kökü tespit edilirken morfolojik bir analiz yapılır. Kelimenin hiç ek almamış ya da çekimlenmemiş yalın haline 'Lemma' ismi verilir. Lemmatizasyon algoritmaları çalıştırılabilmek için bir sözlüğe ihtiyaç duyar. Kelimenin yalın halinin daha önceden belirlenmiş bu sözlükteki karşılığını bulur ve çıktı olarak verir. Örnek bir lemmatizasyon Şekil 2.9'da gösterilmiştir.



Şekil 2.9. Örnek bir lemmatizasyon işlemi

2.3.1.2. Sözdizimsel Analiz

Sözdizimsel analiz aşaması cümlenin gramer yapısını ortaya çıkarmak için cümledeki kelimeleri analiz etmeye odaklanır. Bu analiz hem dilbilgisi hem de bir ayrıştırıcı gerektirmektedir. Sözdizimsel analiz aşamasının çıktısı, kelimeler arasındaki yapısal bağımlılık ilişkilerini ortaya çıkaran cümlenin bir temsilidir. Söz dizimi çoğu dilde anlam taşır çünkü cümle düzeni ve cümle içerisindeki kelimelerin birbirlerine yapısal bağımlılığı cümle anlamını direkt olarak etkilemektedir. Bir kelimenin cümle içerisindeki konumuna göre cümle anlamı tamamen değişebilmektedir. Bu değişim iki cümle ile örneklendirilebilir. "Köpek kediyi kovaladı." ve "Kedi köpeği kovaladı." cümleleri anlamca birbirlerinin tersidir.

POS (Part of Speech) etiketleme sözdizimsel analiz aşamasında gerçekleştirilen işlemlerden biridir. Kelime ya da kelime gruplarının cümle içindeki işlevlerine göre kategorilere ayrılması işlemine POS etiketleme denir (Straková, Straka, & Hajič, 2014). Bu etiketleme sonucunda isim, fiil, bağlaç vb. kategorilere ayrılır. Tablo 2.3'de bir POS etiketleme örneği gösterilmektedir. Bu örnekte cümle yapısı önce kelimelere dönüştürülür sonra morfolojik köklerine indirgenir. Son adımda da her kelime cümledeki görevlerine göre etiketlenir.

Tablo 2.3. Temel DDİ yöntemlerini gösteren bir örnek

Aşamalar	**Sonuç**
Cümle	Bütün cümleler öğelerine ayrılır.
Token	Bütün, cümleler, öğelerine, ayrılır
Stem/Lemma	Bütün, cümle, öğe, ayrılmak
POS etiketleri ile	Bütün (sıfat), cümle (isim), öğe (isim), ayrılmak (fiil)

2.3.1.3. Anlambilimsel Analiz

Anlambilimsel analiz, cümledeki kelime düzeyindeki anlamlar arasındaki etkileşimlere odaklanarak bir cümlenin tek bir bütün olarak olası anlamlarını belirler. Bu analiz aşaması birden çok anlama sahip sözcüklerin anlam belirsizliğini gidermeyi de içermektedir. Anlam belirsizliğini giderme, çok anlamlı sözcüklerin tek

bir anlamının seçilmesine ve cümlenin anlamsal temsiline dahil edilmesini sağlayan işlemdir. Kelimenin cümle içerisindeki anlam değerini ortaya çıkarmaya çalışırken yan ya da mecaz anlamlar ile karşılaşılabilir. Örneğin bir cümle içerisindeki "yüz" kelimesi Türkçede sayıca çokluğu belirtmek için kullanılmış olabilir ya da başın ön kısmını anlamına gelen çehre şeklinde ifade edilmiş olabilir.

Anlambilimsel analiz aşamasında insanların yapacağı gibi yorumlanmaya çalışılmaktadır. Ancak bu analiz, "tuzlu baklava" gibi yanlış anlam olduğunu yalnızca insanların anlayabileceği cümleleri göz ardı etmektedir. Bu aşamada ayrıca varlık ismi tanıma işlemi de gerçekleştirilen işlemlerden biridir. Bu işlem kişi, kurum, etkinlik gibi daha önceden tanımlanmış kategorilere giren bazı özel isimleri metin dokümanları içerisinden ayıklamayı sağlar.

2.4.2. Metin Temsil Yöntemleri

Bu tezde uygulanan metin temsil yöntemleri terim ağırlıklandırma formüllerine dayalı geleneksel metin gösterim yöntemleri ve sinir ağı mimarisine dayalı kelime gömme yöntemleri olmak üzere iki ana başlık altında toplanabilir.

2.4.2.1. Kelime Ağırlıklandırma Teknikleri

Metin madenciliği ve metinden bilgi çıkarımı görevlerinde, metinler vektör uzayında sayısal olarak ifade edilir. Metin üzerinde bir analiz yapılmadan önce metin, metni temsil eden sayısal vektörlere dönüştürülmelidir. Vektör uzayı modelinde her veri uzayda bir vektör olarak ifade edilir. Bu sayısal gösterim için önce metinde yer alan terimler ayıklanır ve bu terimlere sayısal değerler atanır. Bu atama işleminde verilerin özelliklerinin değerleri baz alınarak vektör koordinatları tespit edilir. Literatürde daha çok TF ve TF-IDF terim ağırlıklandırma yöntemleri kullanılmaktadır. Bu metin temsil yöntemlerinin yaklaşımına 'Kelime Torbası' yaklaşımı denir.

Kelime Torbası (Bag of Words – BoW) kelimenin sayılarla ifade edilmesini sağlayan bir DDİ tekniğidir. BoW yaklaşımıyla oluşturulmuş bir vektör uzayı modelinde, kelimeler, sırası ya da yapısı ne olursa olsun metin içindeki geçişleri göz

önünde bulundurularak ‘torba’ adı verilen bir tabloda tutulur. Basit bir örnek belge üzerinden BoW hesaplaması yapılabilir. Belge aşağıdaki gibi iki cümleden oluşsun:

- Cümle 1: Cevher kitap okumayı sever. Muhammed de kitap okumayı sever.
- Cümle 2: Cevher ayrıca blog yazısı okumayı sever.

Bu cümleler için BoW vektörleri, kelime sayısı ve cümle içerisindeki geçişlerinden oluşur. Vektör oluşturulurken kelimelerin sırası göz ardı edilir. Bu örnek belge için BoW vektörü gösterimi aşağıdaki gibi olacaktır.

- BoW: “Cevher”, “kitap”, “okumayı”, “sever”, “Muhammed”, “de”, “ayrıca”, “blog”, “yazısı”

Her cümle BoW ile vektörleştirildikten sonra bir matris yapısı ile temsil edilebilir.

- Cümle 1: [1, 2, 2, 2, 1, 1, 0, 0, 0]
- Cümle 2: [1, 0, 1, 1, 0, 1, 1, 1, 1]

TF-IDF vektör uzay modelleri arasında popüler olan bir yöntemdir [21]. Bu yöntemde hem belgede geçme sıklığı olan Terim Frekansı (TF) hem de tüm belgelerde bulunma sıklığı olan Ters Belge Frekansı (IDF) dikkate alınmaktadır. Bu ağırlıklandırma yönteminde her kelimenin önem değeri temsil edilmektedir. TF $(tf_{i,j})$, j belgesinde i kelimesinin kaç kez geçtiğini kontrol eder. Denklem 2.9’daki gibi hesaplanmaktadır. Terim (kelime), j belgesinde ne kadar çok geçiyorsa önem derecesi o kadar yüksektir.

$$t f_{i,j} = \frac{f_{i,j}}{n_j} \tag{2.9}$$

Denklem 2.9’da görülen eşitlikteki $f_{i,j}$, bir j belgesindeki i teriminin sıklığını belirtir. Ayrıca n_j, j belgesindeki toplam kelime sayısını ifade etmektedir. Denklem 2.10’da görülen IDF (idf_i) belgedeki i teriminin tüm belge boyunca kaç kez

geçtiğini kontrol eder. Terim tüm belgede ne kadar az geçiyorsa önem derecesi o kadar yüksektir.

$$idf_i = 1 + \log\frac{n}{c_i} \quad \textbf{(2.10)}$$

Denklem 2.10'da görülen eşitlikteki n belirli bir tüm belge içindeki toplam alt belge sayısını gösterir. Ayrıca c_i , i terimini içeren belge sayısını ifade etmektedir. TF-IDF yöntemi sonuç olarak Denklem 2.11'deki gibi TF ve IDF değerlerinin çarpımıyla elde edilir. $w_{i,j}$ TF-IDF sonucunu göstermektedir.

$$w_{i,j} = t\,f_{i,j} \times idf_i \quad \textbf{(2.11)}$$

2.4.2.2. Sinirsel Dil Modelleri

Sinirsel Dil Modelleri (SDM) temel olarak kelimelerden cümlelerin yüksek kaliteli temsillerini öğrenmek için kurgulanır. Gizli katmanlarla tasarlanmış ileri beslemeli YSA yapıları olan SDM'ler, sözdizimsel ve anlamsal benzerliklere dayalı kelime vektörlerini hesaplamak için tasarlanmıştır. YSA gibi kategorik değişkenlerle doğrudan çalışamayan yapılar için belgeleri sayısal olarak temsil etmenin bir başka yolu da vektörleri kullanmaktır. Belgeler kelime torbası kullanılarak temsil edildiğinde, belgelerdeki kelimelerin sırası kaybolur. Ayrıca belgeleri temsil eden vektörler yüksek boyutlu ve nadirdir. Bu tip sorunları ortadan kaldırma amacıyla kelime gömme yöntemleri önerilmiştir. Gömme yöntemlerinin temel amacı; kelimeler arası anlamsal ilişkileri kaybetmeden kelimeleri düşük boyutlu vektörlerle temsil etmektir.

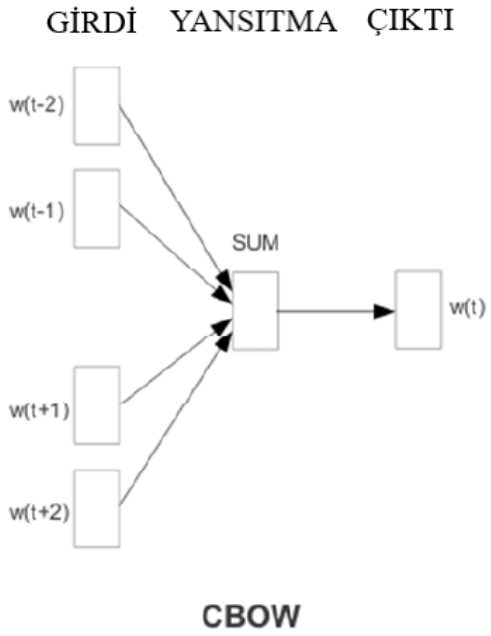
Kelimeler arasında kavramsal ilişki kurmak için kelimelerin ne kadar sık bir arada geçtiği verisini içeren kelime temsillerinin öğreniminin, literatürde metin sınıflandırma başarısını büyük ölçüde etkilemekte olduğu görülmektedir [22]. Çok büyük boyutlu ve kelimenin anlamını kaybeden vektörleştirme teknikleri metin sınıflandırma görevinde bir açık oluşturmaktaydı. Literatürdeki bu açığı önerilen Word2vec [23], Doc2vec [24], Fasttext [25] ve Glove [26] kelime gömme teknikleri kapattı.

Word2Vec modeli Mikolov ve arkadaşları tarafından önerilmiş olan, giriş katmanı, özelleştirilmiş bir gizli katman ve çıkış katmanı olmak üzere üç katmandan oluşan SDM ve bir kelime gömme tekniğidir [23]. Ağın eğitimi sırasında geri yayılım algoritmalarından stokastik gradyan inişi geri yayılım tekniği kullanılır. Bu yöntem sayesinde kelimeler vektörlere dönüştürülerek aralarındaki uzaklık hesaplanır. Word2Vec modeli bu uzaklık değerlerini kullanarak anlamca yakın olan kelimeleri vektör uzayında birbirine yakın olacak şekilde konumlandırır. Bu sayede kelimelerin uzayda dağılımı otomatik olarak sağlanmaktadır. Modelin eğitimi sonucunda hangi kelimelerin birlikte kullanıldığı veya hangi kelimelerin anlamsal olarak birbirlerine yakın olduğu istatistiksel olarak hesaplanarak elde edilebilmektedir.

Ağı eğitmek amacıyla kullanılan hata payını minimize etmeye çalışan geri yayılım algoritmasının türü, kullanılan aktivasyon fonksiyonu ve bazı parametreler Word2Vec modelinin kullanım amacına göre her uygulamada farklılık göstermektedir.

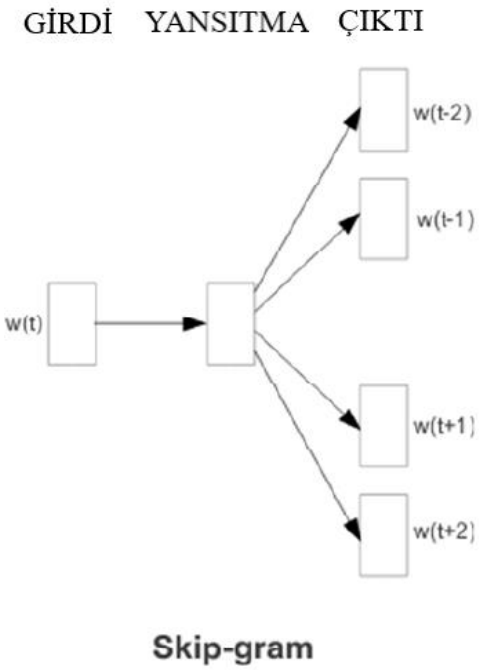
Word2vec, kelimeler arasında anlamsal bağlamda ilişki kurabilmek için Continous Bag-of-Words (CBoW) ve Skip – Gram (SG) mimarileri olmak üzere iki farklı öğrenme mimarisi kullanır. Her iki mimari de eğitim aşaması için, yazılı metinlerin bir araya getirilmesinden oluşan derlemleri kullanan ve derlemdeki her bir kelime için kelime vektörleri üreten denetimsiz öğrenme modelleridir.

CBoW mimarisi Sürekli Kelime Torbası şeklinde literatürde yer almaktadır. CBoW'da bir kelimenin belirli bir sınır içerisindeki komşu kelimelerine bakılmakta ve ilgili kelime komşu kelimelerden yola çıkılarak tahmin edilmektedir. SG'de ise CBoW mimarisinin tam tersi şekilde bu sefer hedef kelimeye bakılarak komşu kelimeler tahmin edilmektedir [27]. Şekil 2.10'da Word2Vec CBoW mimarisi gösterilmektedir.



Şekil 2.10. Word2Vec CBoW mimarisi [23]

Çevredeki kelimelerden $(w(t-2)\,,\ w(t-1),\ w(t+1),\ w(t+2))$ kelime $(w(t))$ tahmin edilmeye çalışılmaktadır. SG mimarisinde de bu işlemin tam tersi gerçekleşmektedir. Mimariye giriş olarak verilen kelimeden $(w(t))$ çevredeki kelimeler $(w(t-2)\,,\ w(t-1),\ w(t+1),\ w(t+2))$ tahmin edilmeye çalışılmaktadır. Şekil 2.11'de Word2Vec SG mimarisi gösterilmektedir.



Şekil 2.11. Word2Vec SG mimarisi [23]

Word2Vec metin sınıflandırması konusunda son yıllarda oldukça tercih edilen bir yöntem olmuştur. Birçok çalışmada [28 – 35] sınıflandırma işleminden önce Word2Vec YSA modeli kullanılmış ve veri seti zenginleştirilerek metin verileri vektörel formata dönüştürülmüştür.

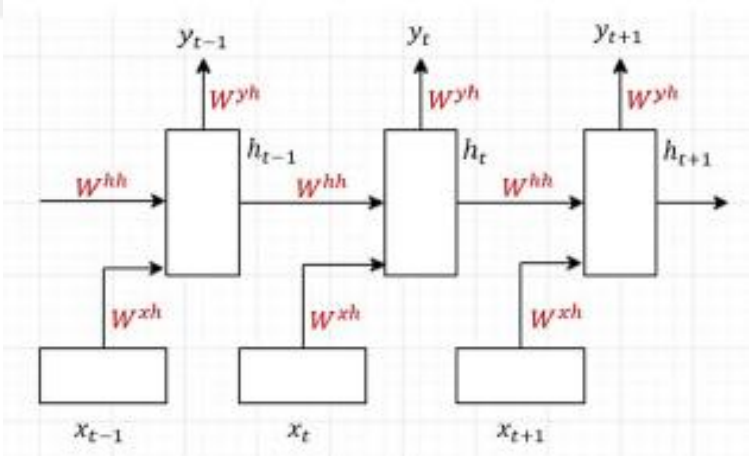
Önceki kelime gömme modellerinin dezavantajı kelimeleri vektörleştirirken kelime morfolojisini işleyememesidir. Kelimelerin içerisindeki yapı, heceler ve harflerin bir arada bulunup bulunmadığı gibi alt kelime bilgileri bu modellerde yok sayılır. Bu bilgilerin yok sayılması Türkçe gibi morfolojik olarak zengin dillerde problem teşkil etmektedir. Örneğin Türkçede “gözlük” ve “gözlükçü” gibi aynı kökten gelen kelimelerin benzer kelime vektörleri olması beklenirken önceki modeller ile bu sağlanamamaktadır. Bu soruna çözüm bulmak için Bojanowski ve arkadaşları karakter düzeyindeki özellikleri ve SG mimarisini birleştirerek temsili bir hesaplama olan FastText modelini önerdi. Bu model kelimeleri karakterlerin toplamı olarak temsil edebilir, bu nedenle daha iyi vektör temsillerinin oluşturulabilmesi özellikle morfolojik olarak zengin diller için kolaylaşır. Ayrıca bu model hızlı ve sağlam bir metin sınıflandırması sağlar [36].

2.4.3. Yinelenen Dil Modelleri

Cümlelerin vektörel temsili, SDM'lere girdi olarak verilen kelime gömmeleri kullanılarak hesaplanabilir. Bu bölümde değişken uzunluklu, zamansal veya sıralı girdilerden öğrenmek için kullanılan zincir yapılı modeller anlatılacaktır. Bu tezde DA görevine odaklanıldığı için ilk olarak Derin Yinelenen Sinir Ağı (RNN) modeli açıklanmıştır. Sonrasında bir RNN mimarisi kullanılarak özelleştirilmiş olan Uzun Kısa Vadeli Bellek Ağları (LSTM) modeli açıklanmıştır.

2.4.3.1. Derin Yinelenen Sinir Ağları

RNN [37], geçmiş bilgilerini depolayan ve bu bilgileri geleceği tahmin etmek için kullanan sıralı verileri işlemek için geliştirilmiş bir YSA mimarisidir. Bu mimari Şekil 2.12'de görülebileceği gibi ardışık bloklarla temsil edilir. Giriş verilerindeki tüm elemanlar için aynı hesaplamalar yapılır ve bu hesaplama sonuçları bir sonraki bloğun girişidir. RNN dil modelleme, makine çevirisi, konuşma tanıma, DA gibi farklı DDİ problemlerinde kullanılmaktadır.



Şekil 2.12. Temel RNN mimarisi

RNN'ler, cümleleri, paragrafları ve belge temsillerini sırayla oluşturmak için uygulanmıştır. Zaman bağımlılıkları arasındaki sıralama adımlarına bilgi aktarabilir ve sınırsız uzunluktaki verileri modelleyebilirler. RNN'lerin standart ve en basit versiyonu, bir giriş katmanı, bir gizli katman ve bir çıkış katmanına sahip olan ağ

Elman Ağı olarak adlandırılır [37]. Derin bir RNN'in tekrarlayan sinir ağları, sıralı bir zamansal veriden öğrenebilen dinamik katmanlardan oluşur.

$$x^{(i)} = (x_1, \quad x_2, \dots, \quad x_{t-1}, \quad x_t) \tag{2.12}$$

Denklem 2.12 veri kümesindeki her bir gözlem için oluşturulmuş n boyutlu kelime gömmeleri veya kelime vektörleri olsun. Derin bir RNN modeli aşağıdaki Denklem 2.13'te verilen yineleme formülü kullanılarak oluşturulmuştur. Ayrıca Denklem 2.14'te yaygın olarak kullanılan iki doğrusal olmayan aktivasyon fonksiyonları olan tanh() (tanjant hiperbolik) fonksiyonu ve sigmoid kullanılarak her t adımı gösterilmiştir.

$$h_t = \tanh(W_{hh}h_{t-1} + W_{xh}x_t + b_h) \tag{2.13}$$

$$y_t = sigmoid\big(W_{hy}h_t + b_y\big) \tag{2.14}$$

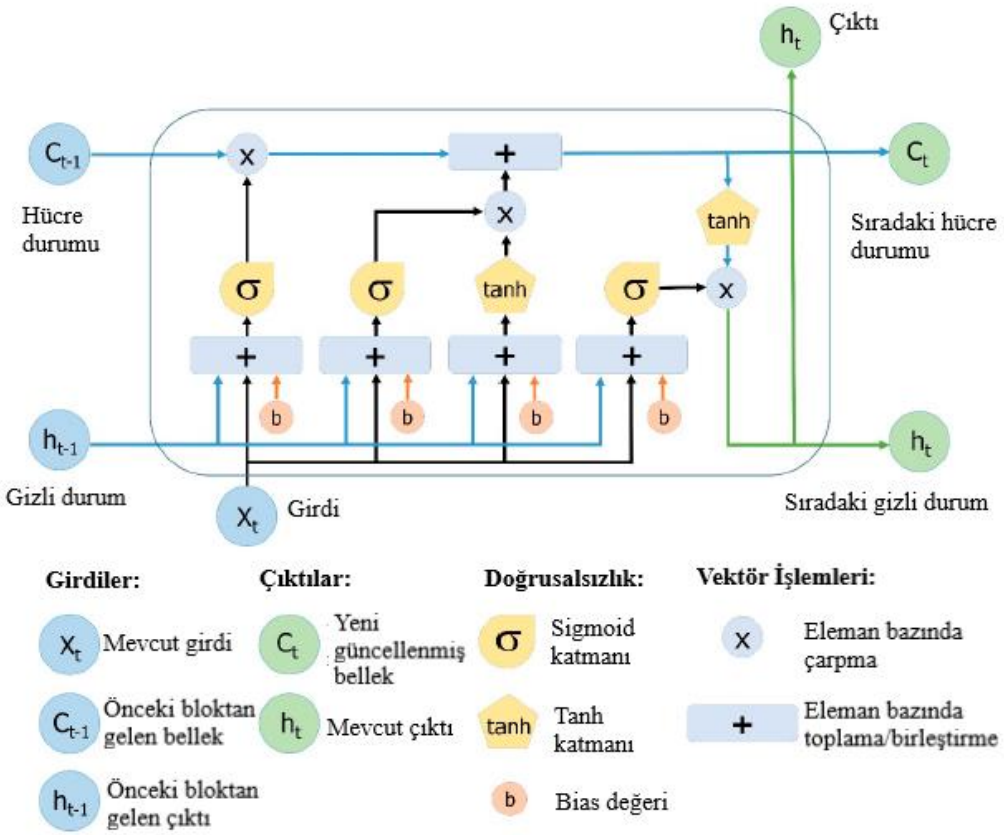
Denklem 2.13'te yer alan h_t , t zaman adımındaki gizli bir durumdur ve x_t , t adımındaki girdi vektörünü temsil eder. h_t , önceki h_{t-1} zaman adımından gelen bilgiyi iletir ve onu verilen $x^{(i)}$ gözlemini sınıflandırmak için kullanır. Buradaki W_{hh} önceki durum h_{t-1}'i koşullandırmak için kullanılan ağırlık matrisidir, W_{xh} giriş x_t'yi koşullandırmak için kullanılan ağırlık matrisidir, son olarak y_t , t zamanında kelime dağarcığı üzerindeki olasılık dağılımının çıktısıdır ve bu çıktı Denklem 2.14 ile hesaplanır. DA görevinde sigmoid fonksiyonu $x^{(i)}$'nin duygu sınıfını, yani veri kümesindeki her bir gözlemin hangi sınıfa ait olduğunu tahmin etmek için kullanılır. Öğrenme işlemi W_{xh} ve W_{hh}'yi güncelleyerek gerçekleştirilir. Kayıp fonksiyonunu minimize etmek amacıyla sistem parametrelerinin en optimal değerlerini öğrenmek için stokastik gradyan iniş gibi geri yayılım teknikleri kullanılabilir.

RNN mimarisinde h_t gizli durum, sinir ağının hafızasıdır. Denklem 2.14'ün öne sürdüğü gibi tahmin bu gizli duruma göre yapılır. Ayrıca bu mimaride W ağırlık değerleri ortak olması her blok için parametre sayısını ve hesaplama süresini azaltır.

Ancak RNN geçmiş bilgiyi kullanarak hesap yapabilmesine rağmen pratikte yalnızca birkaç adım geriye gidebilir. Son katmanda hatanın geri yayılması sırasında geçmiş bilginin sadece küçük bir kısmı iletilebilir. Bu durum kaybolan gradyan problemi olarak ifade edilir. Anlatılan RNN yapısı kaybolan gradyan probleminin üstesinden gelemez. Bu problemi çözebilmek için çift yönlü RNN, derin çift yönlü RNN ve LSTM gibi farklı RNN mimarileri geliştirilmiştir.

2.4.3.2. Uzun Kısa Vadeli Bellek Ağları

LSTM, Hochreiter ve Schmidhuber tarafından rastgele boyutlu sıralı veriler için uzun vadeli zaman bağımlılık probleminin üstesinden gelmek amacıyla geliştirilmiş özel bir RNN mimarisidir [38]. Standart RNN mimarisinden daha karmaşık bir mimariye sahip olan LSTM, RNN'i temel aldığı için yinelenen bloklar olarak ifade edilir. Standart RNN mimarisi yalnızca gizli duruma sahipken, LSTM'in gizli ve hücre durumu olmak üzere iki durumu vardır. LSTM mimarisinde hücre durumuna bilginin aktarılmasını sağlayan üç kapı vardır. LSTM'deki kaybolan gradyan problemine çözüm sağlayan hücre durumu ve unutma kapısı yapılarıdır. Her LSTM bloğu iki durum ve dört katmandan oluşur. Bir LSTM bloğu ve içerdiği katmanlar Şekil 2.13'te daha iyi anlaşılabilir.



Şekil 2.13. LSTM sinir ağının yapısı. (Yan [39]'den çoğaltılmıştır.)

Bir LSTM ağı oluştururken ilk adım gerekli olmayan bilgileri ve bu adımda hücrenin çıktısı olarak alınacak olan bilgileri tanımlamaktır. Bu tanımlama ve elimine etme işlemine, $t-1$ zamanında en son LSTM bloğunun h_{t-1} çıkışını ve t zamanında mevcut giriş anlamına gelen x_t'yi alan sigmoid fonksiyonu tarafından karar verilir. Sigmoid fonksiyonu önceki çıktıdan hangi parçanın hariç tutulacağını belirler. Bu kapıya unutma kapısı denir. Denklem 2.15'te hesaplanma formülü verilen f_t, hücre durumundaki her sayıya (C_{t-1}) karşılık gelen 0 ile 1 arasında değişen değerlere sahip bir vektördür.

$$f_t = \sigma\left(W_f[h_{t-1}, X_t] + b_f\right) \tag{2.15}$$

Denklem 2.15'teki formülde σ sigmoid fonksiyonudur ve W_f, b_f, unutma kapısının sırasıyla ağırlık matrisi ve sapmasını ifade etmektedir. Ağda sonraki adım, hücre durumunu güncellemenin yanı sıra hücre durumundaki yeni girişten (X_t) gelen

bilgilerin kararlaştırılması ve saklanmasıdır. Bu adım sigmoid ve tanh olmak üzere iki fonksiyonu içerir. Sigmoid fonksiyonu $[0, 1]$ yeni bilginin güncellenmesine ya da yok sayılmasına karar verir. Tanh fonksiyonu $[-1, 1]$ bu bilginin önem derecesine karar vererek ona bir ağırlık değer atar. Denklem 2.18'de görüldüğü gibi yeni hücre durumunu güncellemek için iki değer çarpılır. Bu yeni bellek değeri eski değer C_{t-1}'e eklenir ve C_t sonucu elde edilir.

$$i_t = \sigma(W_i[h_{t-1}, X_t] + b_i) \quad \textbf{(2.16)}$$

$$N_t = \tanh(W_n[h_{t-1}, X_t] + b_n) \quad \textbf{(2.17)}$$

$$C_t = C_{t-1} f_t + N_t i_t \quad \textbf{(2.18)}$$

Son adımda çıkış değerleri h_t, çıkış hücresi durumunu ifade eden ve Denklem 2.9'da hesaplanması yer alan O_t temel alınarak hesaplanır. Sigmoid katmanının O_t çıktısı, hücre durumunun tanh katmanına verilmesinden elde edilen $[-1, 1]$ aralığındaki bir değerle çarpılarak h_t çıkış değeri Denklem 2.21'deki şekilde hesaplanır.

$$O_t = \sigma(W_o[h_{t-1}, X_t] + b_o) \quad \textbf{(2.19)}$$

$$O_t = \sigma(W_o[h_{t-1}, X_t] + b_o) \quad \textbf{(2.20)}$$

$$h_t = O_t \tanh(C_t) \quad \textbf{(2.21)}$$

Denklem 2.10'daki formülde σ sigmoid fonksiyonudur ve W_o, b_o, çıkış kapısının sırasıyla ağırlık matrisi ve sapmasını ifade etmektedir.

2.4.4. Çift Yönlü Dil Modelleri

Bölüm 2.4.2.2'de anlatılan kelime gömme teknikleri Word2Vec ve FastText modelleri anlamsal veya sözdizimsel bilgileri kavramak için daha uygun olsalar da

bu modeller bağlamdan bağımsızdır ve her kelime için yalnızca bir gömme vektörü üretirler. Ayrıca bu dil modelleri yalnızca sol veya sağ bağlamı kullanırlar. Bölüm 2.4.3'te anlatılan RNN mimarileri, sonsuz sol bağlam (hedef kelimenin solundaki kelimeler) ya da sonsuz sağ bağlam (hedef kelimenin sağındaki kelimeler) sağlayabilir. Ancak DDİ problemlerini çözerken istenilen hem sol hem de sağ bağlamları kullanarak kelimenin cümle içerisindeki anlamını daha sağlam şekilde ortaya çıkarmaktır. Bu düşünceden yola çıkılarak, çift yönlü öğrenme ile dil anlayışı geliştirildiğinden, yeni SDM'ler, Dil Modellerinden Gömmeler (ELMo) [40] ve BERT [10] geliştirilmiştir.

Yeni SDM'lerden ELMo kısaca, bir cümledeki kelimelerin her iki yönlü pozisyonunu da hesaba katan çift yönlü bir RNN kullanır. Benzer şekilde bağlamsallaştırılmış başka bir dil öğrenme modeli olan BERT, sıralı tekrarlamadan ziyade paralel dikkat katmanlarına sahip çok katmanlı çift yönlü dönüştürücü-kodlayıcı faydası sağlar. Ayrıca bu tez çalışmasında kullanılan yöntem olan BERT Deneysel Kısım'da detaylı olarak anlatılmıştır. Genel olarak, çift yönlü modeller kelimeleri bir cümledeki diğer kelimelere göre işler ve bu nedenle zengin bağlamsal bilgilerin kullanımıyla amaçlanan anlamaya daha yakındır.

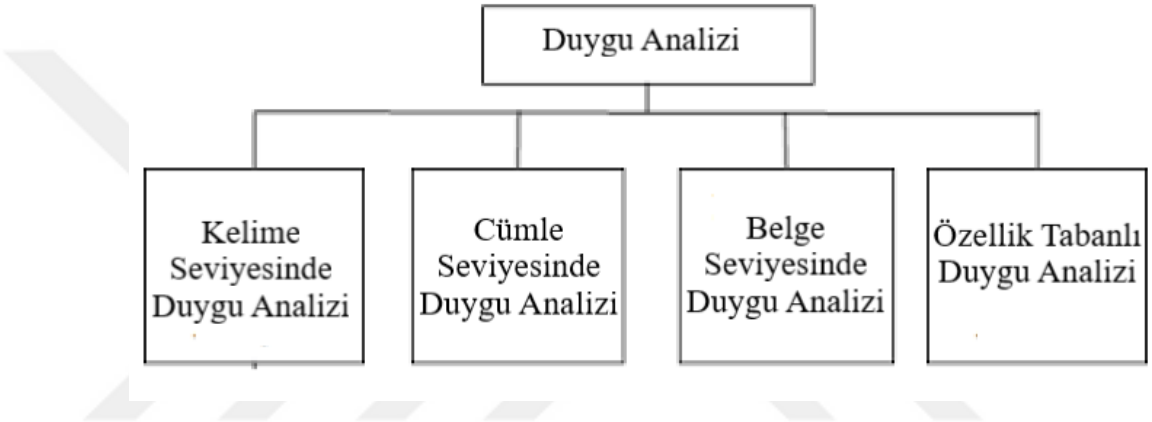
2.5. Duygu Analizi

DA belirli alanlar, konular veya öğeler hakkında genellikle olumlu, nötr veya olumsuz bir düşünce olarak sınıflandırılan belirli bir metnin kutupluluğunu tespit etmeye dayanır [41]. DA kavramı ilk olarak Nasukawa ve Jeonghee tarafından kullanılmıştır [42]. DA dilbilim, psikoloji ve bilgisayar bilimleriyle ilgili disiplinler arası bir araştırma konusudur. Aynı zamanda literatürde fikir madenciliği olarak da ifade edilmektedir. Son yıllarda gelişen sosyal medya kaynaklarının etkileri ve sektördeki yaygın kullanımı nedeniyle bu konu araştırmacıların ilgi odağı haline gelmiştir.

Bir fikrin duyguları duygusal ya da rasyonel olabilir. Örneğin, "Bu kitap en çok okunanlar listesinde birinci sırada." ve "Bu cep telefonunun pil ömrü uzun değil." cümleleri sırasıyla kitap ve bilgisayar hedefleri hakkında rasyonel bir duygu içerir. "Tezimi yazarken aşırı mutluyum." ve "Şimdiye kadar yediğim en iyi yemek

bu.” cümleleri sırasıyla tez ve yemek hedefleri hakkında duygusal duygu içerir. DA bu duygu ve duyguların etkilerinin metinde nasıl ifade edildiğini keşfetmek amacıyla doğal dili işler. Metinler ham formda oldukları için DA tarafından doğrudan analiz edilemezler. Duyguları ortaya çıkarmak için ham verinin üzerinde bazı ön işlemler yapmak gerekmektedir. Bu işlemler ham verinin toplanmasıyla başlar. DA, MÖ tabanlı yöntemlerle gerçekleştirilmektedir.

Literatürde DA görevinin farklı seviyeleri vardır. Şekil 2.14’te bu seviyeler gösterilmektedir.



Şekil 2.14. DA görevi seviyeleri

Tek bir hedefe dayalı olarak ifade edilen incelemenin genel kutupluluğu dikkate alınırsa, A incelemesinde gösterilen belge seviyesinde DA’dir. Duyguyu tespit etmek yalnızca bir cümleyi dikkate alarak yapıldığında, bu, cümle seviyesinde DA’dır. Cümle seviyesi DA, bir belgedeki cümleleri tek tek ayırır. Ancak her cümlenin fikir sahibinin olduğu varsayılamaz. Genellikle ilk adım olarak öznellik sınıflandırması olarak adlandırılan işlemle cümleler fikir sahibi olan ya da olmayan olarak iki kategoriye ayrılır. Daha sonra ortaya çıkan fikir sahibi olan cümleler olumlu, olumsuz görüş bildiren ifadeler olarak sınıflandırılır. Özellikle belirli bir alana özgü uygulamalar için kullanılabilen, cümlenin her bir kelimesinin duygu yönelimini inceleyen DA ise kelime düzeyinde DA’dır. Son olarak görüş seviyesinde DA özellik bazında fikir madenciliği ve özetlemenin yapıldığı seviyedir. Bu DA uygulama türü, istenen özellik hakkındaki duygulara ihtiyaç olduğunda kullanılır, ürünün belirli bir özelliğine odaklanılarak gerçekleştirilir. Örneğin bir otel alanına

yönelik bir DA uygulaması için bu özellik, otelin odaları, hizmet kalitesi, temizliği, konumu veya yemekleri olabilir. Görüş düzeyinde DA görevi, görünüm çıkarımı, varlık çıkarımı, görünüm duyarlılığı sınıflandırması şeklinde çeşitli alt görevlerden oluşur. Örneğin, "iPhone'un ses kalitesi harika, ancak pili berbat." Cümlesinden varlık çıkarımı alt görevi, varlık olarak "iPhone" belirlemelidir. Görüş düzeyinde DA "ses kalitesi" ve "pil" konularının görüş hakkında ayrı iki husus olduğunu tanımlamalıdır.

DA görevinin doğruluk düzeyi DA kaynaklarının ve DDİ araçlarının mevcudiyeti ile ilgilidir. Örneğin bir alana özgü incelemelerin kutupluluğunu anlamak, alana özgü kutupluluk sözlüklerini ve gelişmiş özellik çıkarma kuralları gibi önemli araçları kullanmayı gerektirir [43]. Kutupluluk sözlükleri [44, 45, 46] ve duygu ağaç bankaları [47] gibi çoğu DA yöntemi, DDİ kaynağı ve ayrıştırıcılar ilk olarak İngilizce için uygulanmıştır [48].

Sözlükler ve derlemler gibi birçok kaynak, anlambilim ve DA çalışmalarında kullanılmak üzere geliştirilmiştir. Derlem, araştırmacıların elektronik veri tabanında kayıtlı yazılı ve sözlü metinleri toplamasına ve araştırmacıların dilin farklı yönlerini betimleyerek belirli veya genel amaçlı araştırmalar yapmasına olanak sağlayan metinler bütünüdür. WordNet, "Princeton Üniversitesi Bilişsel Bilimler Laboratuvarı" tarafından oluşturulmuş İngilizce bir veri tabanıdır ve halen güncellenmektedir [49]. WordNet'in Türkçe versiyonu da geliştirilme aşamasındadır [50]. SentiWordNet ise WordNet duygu sınıflandırma sözlüğü ile bağlantılı olarak çalışan her bir kelimeye pozitif (Pos), negatif (Neg) ve nötr (Obj) olmak üzere en fazla 1 olmak üzere sayısal duygu puanları atar [51]. Bu tür veritabanları sınırlı sayıda kelime içerdiğinden, kullanılan modellerin hesaplama performansını azaltmak için sürekli güncellemeye ihtiyaç duyarlar. Ancak Türkçe bir eklemeli dil olduğu için her kelime köküne çeşitli ekler eklenerek birçok farklı kelime türetilebilir ve bunun sonucunda farklı kelime sayısı artar, boşluk boyutu çok büyür ve işlem süresi çok uzun sürer.

DA kavramını ve görevin uygulanmasındaki zorlukları daha iyi gözlemleyebilmek için bir belge inceleyelim.

İnceleme A: *Kullanıcı Cevher'in arkadaşı* ***Tarih:*** *Nisan 15, 2022*

"(1) YARGI dizisinin hayranıyım. (2) Karakterlerin her biri birbirlerinden bağımsız olarak çok iyi kurgulanmış ve birlikte çok uyumlular. (3) Dizide bazı anlar dramatik olabiliyorken çoğunda ders verici ve keyifli vakit geçirten sahneler yer alıyor. (4) Ayrıca oyuncular karakterlerini benimsemiş şekilde karşıya yansıtıyorlar. (5) Bu korkunç bir şey! (6) Hikâye daha önce kurgulanmamış, benzersiz ve harika."

Bu inceleme, bir fikrin hedefi (g) veya varlığı (g) olarak adlandırılan bir dizi hakkındadır. Duygular bir hedef veya varlık hakkında oluşurlar. Olumlu, olumsuz, nötr olarak ya da bir derecelendirme puanı ile ifade edilebilirler. X kullanıcısı, t zamanında ilgili bir hedef hakkında görüş bildiren bir fikir sahibidir (h). DA esas olarak kutupsal görüşlere odaklanır. Dolayısıyla daha resmi ve geniş şekilde bir görüş dörtlü (g, s, h, t) olarak tanımlanır.

DA'nın bazı zorlu noktaları vardır. Bir Türk TV dizisi olan YARGI dizisini konu alan 'Cevher'in Arkadaşı' kullanıcısı tarafından görüş olarak bildirilen A incelemesi analiz edilirse, belgenin genel duygu yönelimi olumlu olsa da belge içerisindeki cümleler farklı kutupluluklara sahip olabilir. Cümlelerin bazıları duyguyu açık şekilde ifade ederken bazıları dolaylı olarak aktarmıştır. Örneğin 2 ve 6 numaralı cümlelerde geçen "çok iyi, çok uyumlu, benzersiz, harika" gibi ifadeler açık/doğrudan görüş ifadeleridir. 3 numaralı cümlede ise dolaylı olarak duygu aktarılmaktadır.

Bir cümlenin kutupluluğunu saptamak, olumlu duygu sözcükleri olarak "iyi, harika ve şaşırtıcı" gibi kutupsal sözcükleri ya da olumsuz sözcükler için "kötü, berbat, sevmediğim" gibi kutupsal sözcükleri kullanmaktır. Yoğunlaştırıcılar genellikle yoğunluğu ifade etmek için "çok, çok, aşırı, korkunç, gerçekten, çok, çok, korkunç" gibi kullanılır. Örneğin A incelemesindeki 5 numaralı cümlede "korkunç" kelimesi olumlu bir duygunun güçlülüğünü ifade etmek için kullanılmış bir yoğunlaştırıcıdır. Ancak bazı cümlelerde "korkunç" kelimesi korkulacak olan kötü bir olayı da ifade edebilmektedir.

2.6. Morfolojik Karmaşıklığına Göre Türk Dili Modellemenin Zorlukları

Bu bölüm DA için morfolojik olarak zengin bir dil olan Türkçe'nin zorlu dilbilimsel sorunlarını ele almaktadır. Ethnologue [52]'e göre Türkçe, 88 milyondan fazla konuşmacıyla en yaygın olarak konuşulan Türk dili olarak, dünyada en çok konuşulan 17. dildir [53]. Dünya Dil Yapıları Atlası [54], yüksek sentez düzeyine sahip olduğu, yani kelime başına ortalama sekiz veya dokuz kategoriye sahip olduğu için Türkçeyi Avrasya dilleri arasında aykırı bir değer olarak tanımlamaktadır [55].

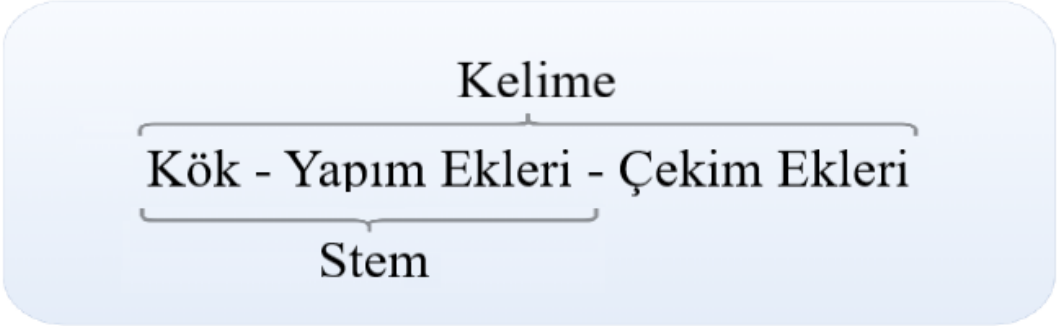
MZD, özellikle Türkçe, hesaplamalı bir DDİ modeli oluştururken kendine has dil zorluklarına sahiptir [56, 57]. Analitik bir dil olan Türkçeye benzer MZD İngilizcenin aksine sondan eklemeli karmaşıklığa sahiptir. Dilin en basit üyesi olan kelime, temelde bir kök ve bağlamı oluşturan zaman, hal (durum) ve şahıs (tekil veya çoğul) gibi çekim eklerinden oluşur. İngilizce benzeri analitik diller bu bağlamı oluşturmak için ayrı edatlardan yararlanır ve doğal olarak analizleri Türkçe gibi MZD'e göre daha kolaydır. Daha açık bir ifadeyle, tek bir Türkçe sözcük bile biçimbirimlerle (kökler, önekler ve ekler) oluşturulmaktadır ve bu sözcük birçok karşılık gelen çekim biçimine sahip olacaktır [58]. Bu Tablo 2.4'te verildiği gibi aynı kökün birçok yapım biçimini üretebileceği anlamına gelir. Örneğin Türkçedeki "*Muvaffakiyetsizleştiricileştiriveremeyebileceklerimizdenmişsinizcesine*" kelimesi en uzun kelime olup 70 harften oluşmaktadır [59].

Tablo 2.4. "yap" kelimesi için Türkçe türetmelerin İngilizcede karşılıkları

Kök: yap	**Kök: do**
yapıyorum	I am doing
yapıyorsun	You are doing
yapıyor	She/He/It is doing
yapıyoruz	We are doing
yapıyorsunuz	You are doing
yapıyorlar	They are doing
yaptık	We done
yaptıkça	As long as (somebody) does
yapmalıyız	We must do
yapmadan	Without doing
yapman	Your doing

yaparken	While (somebody) is doing
yapınca	When (somebody) does

Ayrıntılı bir çalışmada [60] Türkçenin bu muazzam kelime üretme yeteneği ve "dil işlemedeki zorlukları" gösterilmiştir. Araştırmaya göre, türetme biçimbirimlerinin kullanılmasıyla; bir İsimden [masa (table)] ve bir Fiilden [oku (read)] yaklaşık 1,5 milyon farklı kelime türetilmesi mümkündür [58]. Türkçe kelimenin morfolojik yapısı Şekil 2.15'te gösterilmektedir [61].



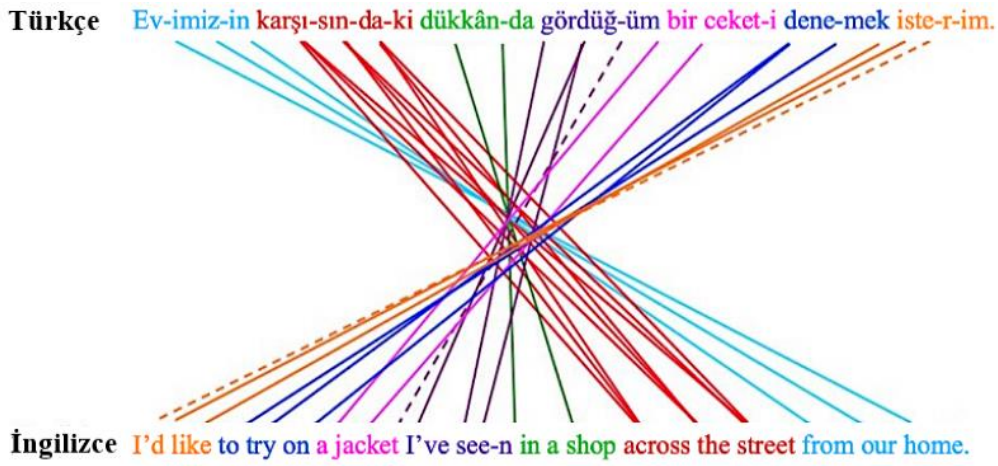
Şekil 2.15. Türkçede bir kelimenin morfolojik yapısı

Türk dilinin morfolojik üretkenliği için bazı örnekler Tablo 2.5'te verilmiştir [62]. Tablo 2.5'ten de anlaşılacağı gibi, bir kelimeye eklenebilecek eklerin sayısı ve bunların kombinasyonları, olası türetmelerden fiili ve kökü elde etmekte ciddi bir dil analizi problemi yaratmaktadır. Ayrıntılı bir çalışmada [43] Türkçe kelimelerin ek yapısına dikkat çekilerek DA görevinde Türkçenin zorlandığı noktalara değinilmiştir.

Tablo 2.5. Türkçedeki "göz" kelimesinden türetilen kelimeler

Kelime	**Kök**	**Stem**	**Çekim Ekleri**
göz/ler-im	göz	göz	ler-im
göz/ün-de	göz	göz	ün-de
göz/cü	göz	gözcü	-
göz/lük	göz	gözlük	-
göz/lük-çü	göz	gözlükçü	-
göz/lük-çü-y-dü	göz	gözlükçü	-y-dü

Türk dili modellemenin bir diğer zorluğu, cümlelerin serbest bir kurgulama yapısına sahip olmasıdır. Bu nedenle aynı sözcüklerin cümle içerisindeki farklı sıralanışları farklı anlamlara neden olabilmektedir. Türkçenin ve İngilizcenin cümle kurgusu ve morfolojik yapıları oldukça farklıdır. İngilizcedeki genel cümle yapısı (özne-nesne-fiil) Türkçede görülmez. Bu farklılık Şekil 2.16'da yer almaktadır.



Şekil 2.16. Türkçe ve İngilizce cümle kurgusu ve morfolojik farklılıkları [63]

Bahsedilen Türk dili modelleme zorlukları, Türkçe DDİ çalışmalarını hedef probleme bir çözüm modellemeden önce dil işleme görevleriyle ilgilenmeye yöneltmektedir. Genel olarak kelimelerin çoğu birçok biçimbirimden oluşması, hesaplamalı modelleme açısından veri seyrekliği ve boyutsallık laneti [64],[65] problemlerine yol açmaktadır. Bu karmaşıklık Türkçe DDİ'de, örneğin İngilizce için geliştirilmiş en son modellerin ve algoritmaların kullanımını kısıtlamaktadır. Türkçedeki veri seyrekliğinin üstesinden gelmek için morfolojik analiz sürecinin iyi işletilmesi ve köklendirme, lemmatizasyon gibi yoğun ön işleme görevleri tamamlanmalıdır [66]. Daha önce de bahsedildiği gibi hem köklendirme hem de lemmatization, kelimelerin çekim veya yapım eklerini ortak bir temel biçime indirgeme amacına sahiptir. Köklendirme veya lemmatizasyon işlemleri birbirinin yerine geçemez, hangisinin uygulanacağının seçilmesi karşılaşılan dil sorununa göre değişmektedir.

Türkçe metinlerde DA görevinin, Türkçedeki bahsedilen dilbilimsel güçlüklere ek olarak hemen hemen her dilde karşılaşılan sözdizimsel ve

anlambilimsel zorlukları da vardır. Ayrıca belirli bir alana özgü DA görevinde kutupluluğun anlaşılması, alana özgü [43] kutupluluk sözlüklerinin, gelişmiş özellik çıkarma kurallarının ve açıklama stratejilerinin kullanılmasını gerektirir. Bu işlemler genellikle insanlar tarafından manuel olarak zaman alıcı şekilde gerçekleştirilir. Ayrıca insanlar kelimeleri işaretlerken kelimeye özel ön yargıya sahip olabilir ve bu sebeple sınıflandırmaya öznellik ekleyebilirler.

Bu tez çalışmasında bahsedilen zorlukların üstesinden gelmek ve bu zorlukların çözümü için gerekli olan yoğun işlemlerden kurtulmak amacıyla Türkçe DDİ ile DA probleminde BERT algoritmasının kullanılması önerilmiştir. Bölüm 3'te BERT algoritmasının mimarisi, Türkçe metinlerde uygulanması ve performansı anlatılmaktadır.

3. DENEYSEL ÇALIŞMALAR VE SONUÇLAR

Bölüm 2'de Türkçe metinlerde DDİ ile DA görevini gerçekleştirmek için gerekli olan alt yapı oluşturulmak için MÖ algoritmaları ve bu algoritmaların başarı ölçütleri, DÖ kavramı ve eğitimi, DDİ kavramı ve aşamaları, dil modelleri, DA kavramı ve Türk dili modellemenin zorlukları anlatılmıştır. Bu çalışmada bahsedilen MÖ algoritmaları ve literatürden elde edilen sonuçları, elde edilen BERT modelinin sonuçları ile karşılaştırılırken temel olarak kullanılacaktır. Bu bölümde ilk olarak; problem tanımlanarak DA görevini gerçekleştirmek için deneylerde kullanılan BERT algoritması mimarisi açıklanmaktadır. Sonrasında çalışmada kullanılan veriler ile yapılan deneyler gösterilmektedir.

3.1. Problem Tanımı

Temel sınıflandırma görevi bir metnin önceden tanımlanmış kategorilere otomatik olarak atanmasıdır. Formüle edilirse, kategorileri olan belirli bir metin kümesi için bir $Y = f(T, \theta) + \in$ modeli oluşturulur. Bu modelde T bir metin temsil şeması iken, θ, sınıflandırıcı model f 'nin eğitimi sırasında tahmin edilmesi gereken bilinmeyen parametreler kümesini ifade eder. Son olarak $\in$, sınıflandırma hatasını ifade etmektedir.

Sınıflandırıcı modelini ifade eden f fonksiyonu, $Y = h(T)$ temsili ile bilinmeyen bir h fonksiyonunun basit bir yaklaşımı olduğundan, sınıflandırma görevinin verimliliği daha küçük hata terimi değerleri için daha iyi hale gelir. Y'nin değeri, sınıflandırma sonucuna karşılık gelen kategorileri gösteren bir dizi ayrık tamsayı üzerinde değişir. Örneğin Y, bir duygu sınıflandırma görevi için sırasıyla pozitif duygu, nötr ve negatif duyguyu ifade eden $+1, 0, -1$ tamsayı değerlerini alabilir [67].

3.2. BERT Algoritması Mimarisi

Devlin ve arkadaşları tarafından 2018'de yapılan bir çalışma neticesinde önerilen BERT, çok katmanlı transformatörler aracılığıyla iki yönlü bağlamları kodlamak için büyük veri kümeleriyle önceden eğitilmiş, çok uygun ve etkili olabilen bağlam temsilleri üreten bir çift yönlü bir SDM'dir. Bu önceden eğitilmiş

model, daha sonra ince ayar [10, 68] ile çeşitli dil görevlerini çözmek için etkili şekilde kullanılmaktadır. BERT, cümledeki kelimelerin bağlamsal anlamlarını öğrenebilir. Örneğin, "Amacım sadece objektif bir konuşma yapmak değil." Cümlesindeki "objektif" kelimesi farklı bir kelime temsiline sahip olmalıdır. Aynı şekilde "Elmayı seviyorum, özellikle Apple mağazasından aldığımda." cümlesi farklı bir kelime temsillerine sahip olmalıdır.

Son zamanlarda SDM yaklaşımlarından çoğu, öğrenilen bilgiyi diğer görevlerden ödünç alma kavramını kullanmaktadır. Bu bağlamda, modern DDİ sistemleri, sıfırdan öğrenen gömmeler yerine BERT gibi daha önceden eğitilmiş dil modellerini tercih etmektedir [10]. DDİ görevleri için önceden eğitilmiş dil modellerini kullanırken iki yaklaşım vardır: özellik tabanlı (feature-based) ve ince ayar (fine-tuning). Yeni SDM'lerden ELMo, DDİ problemi için ek özellikler olarak önceden eğitilmiş temsilleri içeren ilk yaklaşımı kullanır. BERT'in Transformatör kodlayıcı stratejisi bir kelimenin içeriğini ya soldaki ya da sağdaki (çift yönlü olarak) tüm komşularla ilişkili olarak elde etmesini sağlar. Bu noktada çift yönlü dil temsiline sahip olan BERT, ince ayar şeması ile DDİ görevlerinde verimliliği arttırmaktadır.

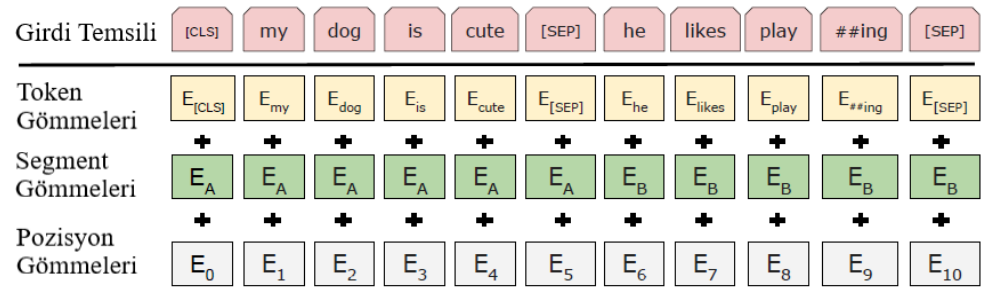
BERT mimarisi, sırasıyla 110M ve 340M ağır parametre ayarlarına sahip temel (BERT-base) ve büyük (BERT-large) olmak üzere başlıca iki versiyona sahiptir. BERT-base mimarisi, 12 adet Transformatör bloğu ve 768 gizli katman bulundururken, BERT-large mimarisi, 24 adet Transformatör bloğu ve 1024 gizli katman boyutuna sahiptir. BERT-large daha fazla GPU bellek desteği gerektirdiğinden bu çalışmada BERT-base mimarisi kullanılmıştır.

3.2.1. BERT Modelinde Denetimsiz Eğitim Öncesi Görevleri

BERT modeli iki denetimsiz görev altında uçtan uca eğitilir: MDM (Close-Task [69] olarak da bilinir) ve Sonraki Cümle Tahmini (NSP) [70]. Maskeli Dil Modeli (MDM) ile tek yönlü temsil üretme problemi çözülmüştür. Bir metin sınıflandırma görevi için dizideki ilk kelime, [CLS] ile gösterilen benzersiz bir simgedir. DDİ görevinde, soru-cevap durumunda olduğu gibi cümle çiftleri varsa, bu cümle çiftleri başka bir özel token ile [SEP] ayrılabilir. MDM, girdi token'larının

belli bir yüzdesini rastgele maskeler ve ardından bağlamına bağlı olarak maskelenen sözcüklerin tahmini ile devam eder [10].

BERT daha önce belirtildiği gibi SEP, CLS ve MASK adlarında ayrı token dizileri ve bir dizi özel token olarak temsil edilen kelime dağarcığı üzerinde çalışır. BERT mimarisi girdi metnini sıralı token'lar olarak temsil eder. Girdi temsili, token, segmentasyon ve pozisyon gömmelerinden oluşur [10]. Giriş gömmeleri, token özelinde öğrenilmiş gömmeler ve pozisyon segmentasyona karşılık gelen kodlamaların toplamı ile elde edilir. Konum, dizideki token'ın indeksi olarak tanımlanırken, segment belirli bir token'ın cümlesinin indeksidir [70]. BERT giriş temsili Şekil 3.1'de gösterilmiştir.

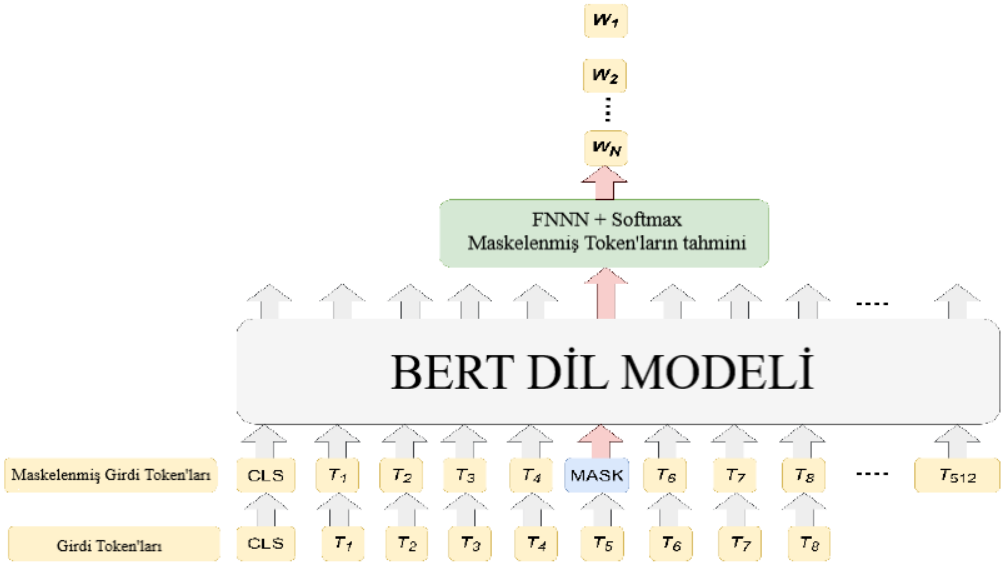


Şekil 3.1. BERT girdi vektörü temsili.

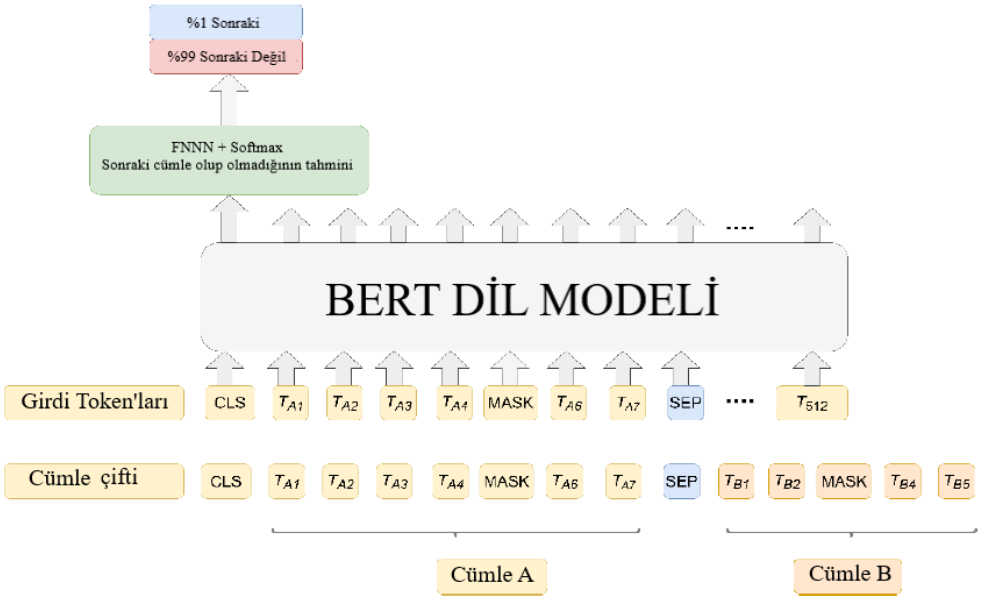
MDM görevi, girdi token'larını maskelenmiş ve gözlemlenmiş olmak üzere iki ayrık kümeye bölerek gerçekleştirilir. Girdi token'larının %15'i seçilir ve buna maskelenmiş token'lar denilebilir. Maskelenmiş token'ların %80'i MASK token'ı ile değiştirilir, %10'u rastgele bir sözcük ile değiştirilir ve kalan %10'luk kısmı değişime uğramaz. Bu maskeleme sürecinin ardından BERT modeli, gözlenen kümeye dayalı olarak maskelenmiş token'ları yeniden yapılandırmak için eğitilir [70].

İkinci denetimsiz görev olan NSP, soru cevaplama ve iki cümle arasındaki ilişkiyi anlamayı gerektiren DDİ problemleri için özellikle önemlidir. BERT, NSP görevi için cümle çiftlerini kullanır. Girdinin ilk cümlesi olduğu gibi bırakılırken, ikinci cümlesinin %50'si rastgele token'lar ile değiştirilir. Bu değişim sürecinin

ardından BERT modeli, gözlenen kümeye dayalı olarak ikinci cümlenin ilk cümlenin devamı olup olmadığının tahmini yapılmaya çalışılarak eğitilir. Bahsedilen iki görev sırasıyla Şekil 3.2 ve Şekil 3.3'te verilmiştir.



Şekil 3.2. BERT denetimsiz MDM görevi [61]



Şekil 3.3. BERT denetimsiz NSP görevi [61]

3.2.2. Özel Problemler için BERT'i İnce Ayarlama

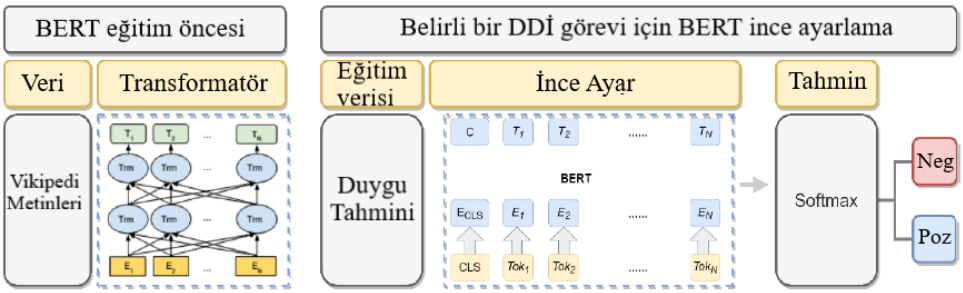
Önceden eğitilmiş BERT modeli ana mimarisinde önemli değişiklikler yapılmadan sadece ek bir çıktı katmanı ile çeşitli DDİ görevleri için ince ayar (fine-tuned) yapılabilir [10]. Bu yaklaşım, BERT'in DA gibi çeşitli önceden eğitilmiş modellerden denetimli öğrenme görevleri (down-stream [71]) için en gelişmiş sonuçları elde etmesini sağlar.

Temel olarak ince ayar; BERT'den gelen ilk token'ın [CLS] son gizli durumunun softmax değerlendirmesi için tam bağlantılı ek bir katmana tanıtılmasıdır [72]. Formül halinde gösterilirse: h ilk token'ın [CLS] son gizli durumunu gösterirken, c etiketinin tahmini Denklem 3.1 ile hesaplanır.

$$p(c|h) = softmax(Wh) \quad \textbf{(3.1)}$$

W, DDİ görevine özgü ağırlık matrisidir. BERT parametrelerinin tümü doğru etiketin olasılığını en üst düzeye çıkarmak için W ile karşılıklı olarak ince ayarlanmıştır [73].

Transformatördeki literatürde kendi kendine dikkat (self-attention) mekanizması olarak geçen mekanizma, cümledeki herhangi bir kelimenin diğer kelimelerle olan ilişkisini ortaya çıkararak, BERT'in uygun girdi ve çıktıları değiştirerek birçok DDİ görevini modellemesine olanak tanır. BERT mimarisinde basitçe, göreve özel giriş ve çıkışlar BERT'e sokulur ve tüm parametreler uçtan uca bir ilkeye göre ince ayar işlemine tabi tutulur. Bir metin sınıflandırma görevi için ince ayarlı BERT modelinin üzerine softmax çıktılı tek katmanlı ileri beslemeli bir sinir ağı yerleştirilir. Örnek bir ikili duygu sınıflandırma şeması Şekil 3.4'te gösterilmiştir.



Şekil 3.4. Örnek bir ikili duygu sınıflandırma için BERT ince ayarlama görevi

3.3. Çalışmada Kullanılan Veriler

Çoğu çalışmada duygu etiketli veriler genellikle film ve otel incelemelerinden alınır [15, 74-77]. Türkçe için ücretsiz olarak erişilebilen etiketli verilerin yetersizliği nedeniyle bu çalışma büyük bir hotel ve film incelemelerinden duygu etiketli bir veri üzerinde yapılmıştır. Bu veri seti bir Türk duygu araştırması için yapılan çalışmada toplanmış ve araştırmalara destek amacıyla ücretsiz olarak sunulmuştur [14]. Alexa5 sitesinde yüksek oranlara ulaşan en popüler iki film ve otel tavsiye sitesi seçilerek oluşturulmuş olan bu veriyi oluşturmak amacıyla, film incelemeleri için beyazperde.com'u, otel yorumları için ise otelpuan.com'u siteleri seçilmiştir.

HTML ayrıştırıcısı olan HTML Agility Pack6 kullanılarak 5660 filmin 220.000 yorumu çıkarılmıştır. 1 veya 2 yıldızla derecelendirilen toplam olumsuz yorum 26700 olduğundan, 4 veya 5 yıldızla derecelendirilen 130210 yorumun da 26700'ü rastgele seçilerek veri kümesi oluşturulmuştur. Toplamda ortalama 33 kelime uzunluğunda 53400 film incelemesi bulunan bir veri kümesi oluşmuştur. Otel incelemelerinin 0 ile 100 arasındaki sayılar ile derecelendirilmesi farkıyla otel incelemelerinde de film incelemelerine benzer bir yaklaşım kullanılmıştır. 550 otelden elde edilen 18478 yorumdan dengeli şekilde bir olumlu ve olumsuz yorum kümesi seçilmiştir. 0 ile 40 arasında derecelendirilen toplam olumsuz yorum 5802 olduğundan, 80 ile 100 arasında derecelendirilen 6499 yorumun da 5800'ü rastgele seçilerek veri kümesi oluşturulmuştur. Toplamda ortalama 74 kelime uzunluğunda 11600 hotel incelemesi bulunan bir veri kümesi oluşmuştur.

Türkçe için DDİ aracı olan Zemberek [78] kullanılarak normalizasyon ve lemmatizasyon işlemleri uygulandıktan sonra her iki veri seti de vektör uzay modelinde temsil edilmektedir.

Tablo 3.1. Kullanılan veri kümelerinin istatistiksel özellikleri özeti

Veri seti	Kategori Sayısı	Örnek Sayısı	Tip	Maksimum Uzunluk
Film	2	Her kategori için 5800	Duygu	124
Hotel	2	Her kategori için 26700	Duygu	166

3.4. Makine Öğrenmesi Deneyleri

Türkçe için BERT'in etkinliğini deneysel olarak gösteren ve çeşitli dil problemlerini çözmek için BERT SDM'sini uygulayan ilk çalışma olan çalışmamız [61], BERT etkinliğini temel MÖ deneyleri sonuçları ile karşılaştırarak göstermiştir. Karşılaştırma için 2019 yılında, bu çalışmada da kullanılan film ve hotel veri seti kullanılarak önemli ön işleme adımlarının sonrasında DVM ve NB algoritmalarının denendiği çalışma [79] baz alınmıştır.

3.4.1. Temel Makine Öğrenmesi Algoritmaları Deneyleri

Bahsedilen çalışmada [79] temel MÖ algoritmaları uygulanmadan önce tokenizasyon, stemming ve lemmatizasyon, özellik çıkarımı gibi birçok önemli ön işleme adımları uygulanmıştır. Daha iyi anlaşılması için Şekil 3.5'te bahsedilen çalışmadan alınan örnek bir lemmatizasyon işlemi gösterilmiştir. Ön işleme adımlarından sonra elde edilen veri kümesi TF-IDF gösterimi ile vektör uzayında temsil edilmiştir.

Lemmatizasyon öncesi	Kesinlikle izlenip desteklenmesi gereken bir müthiş bir film konu olarak orjinal bir film olduğunu da söylemeliyim (16 tokens)
Lemmatizasyon Sonrası	kesin izlemek desteklemek gerek müthiş film konu olmak orjinal film olmak söylemek (12 tokens)

Şekil 3.5. Çalışmada geçen örnek bir lemmatizasyon işlemi [79]

Bu tez çalışması için karşılaştırma amacıyla kullanılacak olan temel MÖ algoritmalarından NB ve DVM'nin kullanıldığı çalışmada elde edilen sonuçlar Tablo 3.2'de gösterilmiştir.

Tablo 3.2. Baz alınan makine öğrenmesi deneylerinin sonuçları

Veri Seti	Temel MÖ Algoritması	Kesinlik	Duyarlılık	F-ölçütü	Doğruluk ACC
Film	NB	0,891	0,889	0,899	%88,68
	DVM	0,863	0,863	0,863	%86,31
Hotel	NB	0,909	0,9	0,899	%89,98
	DVM	0,92	0,92	0,92	%92,26

3.4.2. BERT Algoritması Deneyleri

Bölüm 3.5.1'de belirtilen MÖ algoritmaları için yoğun ön işleme adımlarının gerektiği bilinmektedir. MZD ile ilgili yapılan çalışmaların çoğunun yeterli bir MÖ modeli elde etmek için MÖ performansını etkileyecek olan ön işleme adımlarından sıklıkla yararlandığı görülmektedir. Ancak BERT algoritması için bu yoğun ön işleme görevleri gerekli değildir.

BERT algoritması tasarımcıları makalelerinde [10] yığın büyüklüğü (batch-size), öğrenme oranı (learning rate) ve dönem sayısı (epoch) dışındaki tüm parametreleri sabit tutmayı önerir. Sayılan parametreler için ise önerilen değerler aşağıdaki gibidir:

- Batch-size:16, 32
- Learning rate: 5e-5, 3e-5, 2e-5
- Epoch size: 2, 3, 4

BERT algoritması deneyinin doğruluk (acc), f1 ölçütü, kesinlik ve duyarlılık değerlendirme ölçütlerine göre sonuçları Tablo 3.3'te yer almaktadır.

Tablo 3.3. BERT algoritması deney sonuçları

DA Görevi	ACC	F1-Ölçütü	Kesinlik (P)	Duyarlılık (R)
Hotel veri seti	0,9901	0,9901	0,9901	0,9901
Film veri seti	0,9728	0,9742	0,9722	0,9761

Tüm deneylerde sırasıyla yığın büyüklüğü, öğrenme oranı ve dönem sayısını 16, 2e-5 ve 4 olarak seçilmiş ve bu değerler sabit tutulmuştur. Transformatör bloklarının sayısı 12, gizli katman 768 ve kendi kendine dikkat başlığı (self-attention head) 12 olan önceden eğitilmiş BERT modeli kullanılmıştır. Deneyler yapılırken 128 GB RAM'e sahip Kişisel Bilgisayar (PC), Intel Core i9 9900X CPU ve 11 GB belleğe sahip GeForce RTX 2080 Ti GPU'dan faydalanılmıştır.

3.5. Sonuçların Değerlendirilmesi

Çalışma yapılırken baz alınan temel MÖ algoritmalarının performansları ve BERT deneylerinin sonuçları Tablo 3.3'te sunulmaktadır. İlgili makaleden alınan deney sonuçları kesinlik, duyarlılık, f1 ölçütü ve doğruluk değerlendirme ölçütlerine göre detaylı olarak Tablo 3.2'de yer almaktadır.

BERT etkinliğinin daha iyi anlaşılması için Tablo 3.4'te literatürde kullanılan veri seti ile yapılmış olan çalışmalardan en iyi sonucu vermiş temel MÖ algoritmasının acc değeri ile BERT algoritmasının acc değeri ile karşılaştırılarak performans artışı ifade edilmiştir.

Tablo 3.4. DVM ile BERT algoritmaları sonuçlarının karşılaştırılması

Değerlendirme	Temel MÖ Algoritması	Temel MÖ ACC	BERT ACC	Performans Karşılaştırma
Hotel veri seti	DVM	%92,26	%99,01	+6,75
Film veri seti	NB	%88,68	%97,28	+8,60

Tablo 3.4'te en iyi temel MÖ performansı ile karşılaştırmalı olarak incelenmesi, BERT'in gelişmiş tahmin performansı ürettiğini göstermektedir. Tablo 3.4'te belirtildiği gibi performans artışı ortalama 7,675'tir. Türkçe dilinde yapılmış bu deneyin gösterdiği performans artışı olası bir araştırma yönü olması açısından önemlidir.

BERT modelinin deneysel sonuçlarının son bir değerlendirmesi, MÖ görevlerinde önemli bir başarı kriteri olan algoritma çalışma süresidir. Belirtilen donanımda görevi tamamlamak için geçirilen eğitim süresi 1 saat 46 dakikadır.

4. TARTIŞMA VE GELECEK ÇALIŞMALAR

Kullanıcı yorumlarından duygu tespiti hizmetlerin kalitesini arttırmak için geniş çapta çalışılan bir araştırma alanıdır. Bu alanda MZD ile ilgili çalışmaların yetersizliği dikkat çekmektedir. MZD, elde edilen MÖ modelinin performansını etkileyen çeşitli ön işleme teknikleri kullanılarak analiz edilir. Bu adımların kombinasyonları veya varyasyonları, elde edilen sonuçları tahmin verimliliği açısından değiştirir. Bahsedilen ön işleme görevleri, yoğun yazılım tabanlı adımlara ihtiyaç duyar ve kesin olarak iyi tanımlanmış bir ardışık düzende uygulanmazlar. Öte yandan, SDM'lerden BERT çeşitli dil görevlerini çözmek için nispeten basit bir uygulamaya sahiptir. Ayrıca BERT, DDİ problemlerinde dikkate değer sonuçlar üretebilir.

Bu çalışmada Türk dili alanından duygu analizi problemini çözmek ve film ve otel incelemelerinden oluşan veri setlerini analiz etmek için BERT algoritması kullanılmıştır. Elde edilen sonuçlar literatürdeki temel ML algoritmalarının en iyi tahminleriyle karşılaştırılmıştır. BERT'in tahmin performansını artırdığı deneysel olarak gösterilmiştir. Ayrıca BERT tahminlerindeki performans artışının temel algoritmalara göre dikkat çekici olduğu gözlemlenmiştir. Performans sayısal olarak %8,60'a kadar yükselmiştir.

Türkçe NLP alanından çeşitli veri kümelerinin ampirik değerlendirmesi, morfolojik olarak zengin dillerin BERT algoritmasından ön işleme dil adımlarının ortadan kaldırılması ve tahminin güçlenmesi olmak üzere iki önemli noktada yararlanabileceğini göstermektedir. Bir ince ayar görevi için bile gerekli eğitim süresi daha fazla olsa da geçen süreyi BERT'in hafif versiyonları veya önceden eğitilmiş diğer bazı modellerin kullanımı ile azaltmak mümkün olabilir.

Bu çalışmada sadece Türkçede uygulanmış bir dil problemi analiz edilmiş olsa da, BERT dil modelinin Arapça, Çekçe, Macarca ve Fince gibi morfolojik olarak zengin ve kaynakları az olan dillere de uygun olacağı öngörülmektedir. Bu sebeple gelecekte, tahmin performanslarını yüksek tutarken eğitim sürelerini azaltmak için BERT benzeri mimarilerin daha hafif sürümlerini kullanılabilir.

KAYNAKLAR

[1] Pang, B., Lee, L., Vaithyanathan, Thumbs up? Sentiment Classification using Machine Learning Techniques. EMNLP, 6-7 Temmuz, 2002, Philadelphia, ABD (Bildiriler Kitabı 79- 86.)

[2] Socher R., Perelygin A., Wu J. Y., Chuang J., Manning C. D., Ng A. Y., Potts C. Recursive deep models for semantic compositionality over a sentiment treebank, EMNLP, 18- 21 Ekim, 2013, Seattle, ABD (Bildiriler Kitabı 1631- 1642.)

[3] Baly R., Hajj H., Habash N., Shaban K. B., El-Hajj W. A sentiment treebank and morphologically enriched recursive deep models for effective sentiment analysis in arabic. Transactions on Asian and Low-Resource Language Information Processing. 2017, 16(4), 23.

[4] Socher R., Pennington J., Huang E. H., Ng A. Y., Manning C. D. Semisupervised recursive autoencoders for predicting sentiment distributions. EMNLP, 27- 31 Temmuz, 2011, Scotland, İngiltere (Bildiriler Kitabı 151–161.)

[5] Alhaj Y. A., Xiang J., Zhao D., Al-Qaness M. A. A., Abd Elaziz M, Dahou A. A Study of the Effects of Stemming Strategies on Arabic Document Classification. IEEE Access, 2019,7, 32664- 32671.

[6] Demir, H., Özgür A. Improving Named Entity Recognition for Morphologically Rich Languages Using Word Embeddings. IEEE 13th International Conference on Machine Learning and Applications, 3- 4 Aralık, 2014, Michigan, ABD (Bildiriler Kitabı 117–22.)

[7] Uysal, A. K., Gunal S. The Impact of Preprocessing on Text Classification. Information Processing & Management. 2014, 50(1), 104–112.

[8] Mulki, H., Haddad H., Ali C. B., Babaoğlu İ. Preprocessing Impact on Turkish Sentiment Analysis. 26th IEEE Signal Processing and Communications Applications Conference (SIU), 2- 5 Mayıs, 2018, İzmir, Türkiye (Bildiriler Kitabı 1–4.)

[9] Ebert S., Müller T., Schütze H. LAMB: A Good Shepherd of Morphologically Rich Languages. EMNLP, 1- 5 Kasım, 2016, Texas, ABD (Bildiriler Kitabı 742-752.)

[10] Devlin, J., Chang, M. W., Lee, K., Toutanova, K. BERT: Pre-Training of Deep Bidirectional Transformers for Language Understanding, 2018, arXiv preprint arXiv:1810.04805.

[11] Romanov V., Khusainova A. Evaluation of Morphological Embeddings for English and Russian Languages. Proceedings of the 3rd Workshop on Evaluating Vector Space Representations for NLP, Haziran, 2019, Minneapolis, ABD (Bildiriler Kitabı 77–81.)

[12] Belinkov Y., Durrani N., Dalvi F., Sajjad H., Glass J. On the Linguistic Representational Power of Neural Machine Translation Models. 2019, arXiv preprint arXiv:1911.00317.

[13] Jawahar G., Sagot B., Seddah D. What Does BERT Learn about the Structure of Language. 57th Annual Meeting of the Association for Computational Linguistics, 28 Temmuz-2 Ağustos, 2019, Florence, İtalya (Bildiriler Kitabı 3651–3657.)

[14] Ucan A., Naderalvojoud B., Akcapinar Sezer E., Sever H. SentiWordNet for New Language: Automatic Translation Approach. 12th International Conference on Signal-Image Technology Internet-Based Systems (SITIS), 28 Kasım- 1 Aralık, 2016, Naples, İtalya (Bildiriler Kitabı 308–315.)

[15] Vembandasamy K., Sasipriya, *R.*, Deepa, E. Heart Diseases Detection Using Naive Bayes Algorithm. International Journal of Innovative Science, Engineering & Technology. 2015, 2(9), 441-444.

[16] Rahman M. A., Muniyandi R. C., Islam K. T., Rahman M. M. Ovarian Cancer Classification Accuracy Analysis Using 15-Neuron Artificial Neural Networks Model. IEEE Student Conference on Research and Development. 15-17 Ekim, 2019, Perak, Malezya.

[17] Pektaş, Muhammed. Çekişmeli Üretici Ağlar ile Gerçekçi Saç Sentezi. Ege Üniversitesi, Fen Bilimleri Enstitüsü, İzmir, 2022, 78. (Yüksek Lisans Tezi)

[18] Goodfellow, I., Bengio, Y., Courville, A., Bengio, Y. Deep learning. MIT press. Cambridge, ABD, 2016, 108-109s.

[19] Rumelhart, D., Hinton, G., Williams, R. Learning representations by back-propagating errors. Nature. 1986, **323,** 533–536.

[20] Mirza, Amaad., Asghar, S., Manzoor, A., Noor, M. A. A Classification Model For Class Imbalance Dataset Using Genetic Programming. IEEE Access. 2019, 7(1), 71013–71037.

[21] Yıldırım, M., Okay, F.Y., Özdemir, S. Sentiment Analysis for Turkish Unstructured Data by Machine Translation. IEEE International Conference on Big Data. 10-13 Aralık, 2020, Georgia, ABD.

[22] Koç, A. Eğitim Kümesi Seçiminin Kelime Temsillerine Etkisi ve Türkçe için Benzerlik Test Kümesi. 24. IEEE Sinyal İşleme ve İletişim Uygulamaları Kurultayı, 16-19 Mayıs, 2016, Zonguldak.

[23] Mikolov, T., Chen, K., Corrado, G., Dean, J. Efficient estimation of word representations in vector space. 2013, arXiv preprint arXiv:1301.3781.

[24] Le, Q., Mikolov, T. Distributed representations of sentences and documents. In International conference on machine learning. 2014, 1188-1196.

[25] Joulin, A., Grave, E., Bojanowski, P., Douze, M., Jégou, H., Mikolov, T. Fasttext. zip: Compressing text classification models. 2016, arXiv preprint arXiv:1612.03651.

[26] Pennington, J., Socher, R., Manning, C. GloVe: Global Vectors for Word Representation. EMNLP, 25-29 Ekim, 2014, Doha, Qatar (Bildiriler Kitabı 1532–1543).

[27] Şahin, G. Word2Vec ve SVM Tabanlı Türkçe Doküman Sınıflandırma. 25. IEEE Sinyal İşleme ve İletişim Uygulamaları Kurultayı, 15-18 Mayıs, 2017, Antalya.

[28] Lilleberg J., Zhu, Y., Zhang Y. Support vector machines and Word2vec for text classification with semantic features. IEEE 14th International Conference on Cognitive Informatics & Cognitive Computing, 6-8 Haziran, 2015, Beijing, China (Bildiriler Kitabı 136–140).

[29] Seyfioğlu, M., Demirezen, M. A Hierarchical Approach for Sentiment Analysis and Categorization of Turkish Written Customer Relationship Management Data. Federated Conference on Computer Science and Information Systems, 3-6 Eylül, 2017, Prag, Çek Cumhuriyeti (Bildiriler Kitabı 361–365).

[30] Ge L. Improving Text Classification with Word Embedding**.** San José State Üniversitesi, Washington, 2017, 51. (Yüksek Lisans Tezi)

[31] Çoban Ö., Karabey I. Kelime ve Doküman Vektörleri ile Müzik Türü Sınıflandırması. 25. Sinyal İşleme ve İletişim Uygulamaları Kurultayı, 15-18 Mayıs, 2017, Antalya, Türkiye.

[32] Karasoy, O., Balli, S. Classification Turkish SMS with deep learning tool Word2Vec, International Conference on Computer Science and Engineering (UBMK), 5-8 Ekim, 2017, Antalya, Türkiye.

[33] Ayata D., Saraclar M., Özgür, A. Makine Öğrenmesi ve Kelime Vektör Temsili ile Türkçe Tweet Sentiment Analizi. 25. Sinyal İşleme ve İletişim Uygulamaları Kurultayı, 15-18 Mayıs, 2017, Antalya, Türkiye.

[34] Güngör O., Yıldız E. Türkçe Sözcük Temsillerinde Dilbilimsel Özellikler, 25. Sinyal İşleme ve İletişim Uygulamaları Kurultayı, 15-18 Mayıs, 2017, Antalya, Türkiye.

[35] Bilgin, M. Kelime Vektörü Yöntemlerinin Model Oluşturma Sürelerinin Karşılaştırılması. Bilişim Teknolojileri Dergisi. 2019, 12(2), 141-146.

[36] Bojanowski, P., Grave, E., Joulin, A., Mikolov, T. Enriching word vectors with subword information. Transactions of the association for computational linguistics. 2017, 5, 135-146.

[37] Elman J. L. Finding structure in time. Cognitive Science. 1990, 14(2), 179-211.

[38] Hochreiter, S., Schmidhuber, J. Long Short-Term Memory. Neural Computation. 1997, 9(8), 1735-1780.

[39] Yan, S. Understanding LSTM and Its Diagrams. Çevrimiçi kullanılabilir: https://medium.com/mlreview/understanding-lstm-and-its-diagrams-37e2f46f1714 (18 Nisan 2022 tarihinde erişildi).

[40] Peters, M. E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., Zettlemoyer L. Deep Contextualized Word Representations. 2018, arXiv preprint arXiv:1612.03651.

[41] Pang, B., Lee L. Opinion mining and sentiment analysis. Foundation Trends Information Retrieval. 2008, 2(1-2), 1-135.

[42] Nasukawa T., Yi, J. Sentiment analysis: Capturing favorability using natural language processing. K-CAP '03: Proceedings of the 2nd international conference on Knowledge capture, 23-25 Ekim, 2003, New York, ABD (Bildiriler Kitabı 70–77).

[43] Dehkharghani, R., Yanikoglu, B., Saygin, Y., Oflazer, K. Sentiment analysis in turkish at different granularity levels. Natural Language Engineering. 2017, 23(4), 535–559.

[44] Cambria, E., Havasi, C., Hussain, A. SenticNet 2: A semantic and affective resource for opinion mining and sentiment analysis. 25th International Florida Artificial Intelligence Research Society Conference, FLAIRS 12. 23-25 Mayıs, 2012, Florida, ABD (Bildiriler Kitabı 202–207).

[45] Cambria, E., Olsher, D., Rajagopal, D. SenticNet 3: A common and commonsense knowledge base for cognition-driven sentiment analysis. National Conference on Artificial Intelligence. American Association for Artificial Intelligence (AAAI) Press. Kanada, ABD, 2014, 1515–1521s.

[46] Hatzivassiloglou, V., McKeown, K. R. Predicting the semantic orientation of adjectives. 35th Annual Meeting of the Association for Computational Linguistics and 8th Conference of the European Chapter of the Association for Computational Linguistics, Temmuz, 1997, Madrid, Spain (Bildiriler kitabı 174–181).

[47] Socher, R., Perelygin, A., Wu, J. Y., Chuang, J., Manning, C. D., Ng, A. Y., Potts, C. Recursive deep models for semantic compositionality over a sentiment treebank. EMNLP, 18-21 Ekim, 2013, Seattle, ABD (Bildiriler Kitabı 1631–1642).

[48] Oflazer, K. Turkish and its challenges for language processing. Language Resources and Evaluation, 26-31 Mayıs, 2014, Reykjavik, İzlanda.

[49] Miller, G. A., Beckwith, R., Fellbaum, C., Gross, D., Miller, K. J. Introduction to wordnet: An on-line lexical database. International Journal of Lexicography. 1990, 3(4), 235–244.

[50] Bilgin, O., Çetinoğlu, Ö., Oflazer, K. Building a Wordnet for Turkish. Romanian Journal of Information Science and Technology. 2004, 7(1-2), 163-172.

[51] Baccianella, S., Esuli, A., Sebastiani, F. SentiWordNet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining. in Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10), Mayıs, 2010, Valletta, Malta.

[52] Ethnologue. En Çok Konuşulan 200 Dil Hangileridir? Çevrimiçi erişilebilir: https://www.ethnologue.com/guides/ethnologue200 (19.04.2022 tarihinde erişildi).

[53] Turkic languages Wikipedia. Çevrimiçi erişilebilir: https://en.wikipedia.org/wiki/ Turkic_languages (19/04/2022 tarihinde erişildi).

[54] Haspelmath, M., Dryer, M. S., Gil, D., Comrie, B. The World Atlas of Language Structures. Oxford University Press, 2005, 10-13.

[55] Bickel, B., Nichols, J. Inflectional synthesis of the verb. The World Atlas of Language Structures. . Oxford University Press, 2005, 94-97.

[56] Sak, H., Tunga G., Saraçlar, M. Resources for Turkish Morphological Processing. Language Resources and Evaluation. 2011, 45(2), 249–261.

[57] Vylomova, E., Trevor C., He X., Haffari G. Word Representation Models for Morphologically Rich Languages in Neural Machine Translation. First Workshop on Subword and Character Level Models in NLP, Eylül 2017, Copenhagen, Danimarka (Bildiriler Kitabı 103-108).

[58] Hans, K., Milton, R. S. Improving the Performance of Neural Machine Translation Involving Morphologically Rich Languages. 2017, arXiv preprint ArXiv:1612.02482.

[59] En uzun Türkçe Kelime Vikipedi. Çevrimiçi erişilebilir: https://tr.wikipedia.org/wiki/En_uzun_T%C3%BCrk%C3%A7e_kelime#:~:text=Ancak%20ba%C5%9F%C4%B1nda%2015%20harften%20olu%C5%9Fan,%2C%20ademimerkeziyet%C3%A7ilik%2C%20egzistansiyalizm%20veya%20elektroensefalografi. (19/04/2022 tarihinde erişildi).

[60] Oflazer, K. Turkish and Its Challenges for Language Processing. Language Resources and Evaluation. 2014, 48(4), 639–53.

[61] Özçift, A., Akarsu, K., Yumuk, F., Söylemez, C. Advancing natural language processing (NLP) applications of morphologically rich languages with bidirectional encoder representations from transformers (BERT): an empirical case study for Turkish. Automatika. 2021, 62(2), 226-238.

[62] Kışla, T., Karaoglan B. A Hybrid Statistical Approach to Stemming in Turkish: An Agglutinative Language. Computer Science Anadolu University Journal of Science and Technology. 2016, 17(2), 401-412.

[63] Zeybek, Sultan. Recursive deep learning for Turkish sentiment analysis. Yıldız Teknik Üniversitesi, Fen Bilimleri Enstitüsü, İstanbul, 2021, 156. (Doktora Tezi)

[64] Abudukelimu, H., Liu Y., Chen X., Sun, M., Abulizi, A. Learning Distributed Representations of Uyghur Words and Morphemes. China National Conference CCL, 13-14 Kasım, 2015, Guangzhou, Çin (Bildiriler Kitabı 202-211).

[65] Wolk, K. Machine Learning in Translation Corpora Processing. CRC Press, 2019.

[66] Nuzumlalı, M. Y., Özgür, A. Analyzing Stemming Approaches for Turkish Multi-Document Summarization. EMNLP, 26-28 Ekim, 2014, Doha, Katar (Bildiriler Kitabı 702–706).

[67] Kobayashi, V. B., Mol, S. T., Berkers, H. A., Kismihók, G., Hartog, D. N. D. Text Classification for Organizational Researchers: A Tutorial. Organizational Research Methods. 2018, 21(3), 766–799.

[68] Gao, Z., Feng, A., Song, X., Wu, Xi. Target-Dependent Sentiment Classification With BERT. IEEE Access, 2019, 7, 154290–154299.

[69] Taylor, W. L. 'Cloze Procedure': A New Tool for Measuring Readability. Journalism Quarterly. 1953, 30(4), 1953, 415–433.

[70] Lu, J., Batra, D., Parikh, D., Lee, S. ViLBERT: Pretraining Task-Agnostic Visiolinguistic Representations for Vision-and-Language Tasks. 2019, arXiv preprint arXiv:1908.02265.

[71] McCann, B., Bradbury, J., Xiong, C., Socher, R. Learned in Translation: Contextualized Word Vectors. 2018, arXiv preprint arXiv:1708.00107.

[72] Ma, G. Tweets Classification with BERT in the Field of Disaster Management. Semantic Scholar. Accessed May 4, 2020.

[73] Sun, C., Qiu, X., Xu, Y., Huang, X. How to Fine-Tune BERT for Text Classification?. ArXiv [Cs], 2020.2020, arXiv preprint arXiv:1905.05583.

[74] Dehkharghani, R., Yanikoglu, B., Tapucu, D., Saygin, Y. Adaptation and use of subjectivity lexicons for domain dependent sentiment classification. 12th IEEE International Conference on Data Mining Workshops (ICDMW), 10 Aralık, 2012, Brussels, Belçika (Bildiriler Kitabı 669–673).

[75] Turney, P. D. Thumbs up or thumbs down?: semantic orientation applied to unsupervised classification of reviews. 40th annual meeting on association for computational linguistics, Temmuz, 2002, Pensilvanya, ABD (Bildiriler Kitabı 417–424).

[76] Vural, A. G., Cambazoglu, B. B., Senkul, P., Tokgoz, Z. O. A framework for sentiment analysis in turkish: Application to polarity detection of movie reviews in turkish. Computer and Information Sciences (ISCIS), 28-30 Ekim, 2012, (Bildiriler Kitabı 437– 445). Çevrimiçi erişilebilir: https://hdl.handle.net/11511/88210. (19/04/2022 tarihinde erişildi).

[77] Waltinger, U. Germanpolarityclues: A lexical resource for german sentiment analysis. LREC, 17-23 Mayıs, 2010, Valletta, Malta.

[78] Akın, A. A., Akın, M, D. Zemberek, an open source nlp framework for turkic languages. Structure. 2007, 10, 1–5.

[79] Erşahin, B., Aktaş, Ö., Kılınç, D., Erşahin. M. A hybrid sentiment analysis method for Turkish. Turkish Journal of Electrical Engineering and Computer Science. 2019, 27(3), 1780–1793.