T.C.
YILDIZ TEKNİK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

bi-TEZAT: biLSTM YÖNTEMİYLE TÜRKÇE ŞİKAYET METİNLERİNDE ZAMAN İFADELERİNİN TESPİT EDİLMESİ

Ensar EMİRALİ

YÜKSEK LİSANS TEZİ
Bilgisayar Mühendisliği Anabilim Dalı
Bilgisayar Mühendisliği Programı

Danışman
Prof. Dr. M. Elif KARSLIGİL

Temmuz, 2022

T.C.

YILDIZ TEKNİK ÜNİVERSİTESİ

FEN BİLİMLERİ ENSTİTÜSÜ

bi-TEZAT: biLSTM YÖNTEMİYLE TÜRKÇE ŞİKAYET METİNLERİNDE ZAMAN İFADELERİNİN TESPİT EDİLMESİ

Ensar EMİRALİ tarafından hazırlanan tez çalışması 27.07.2022 tarihinde aşağıdaki jüri tarafından Yıldız Teknik Üniversitesi Fen Bilimleri Enstitüsü Bilgisayar Mühendisliği Anabilim Dalı, Bilgisayar Mühendisliği Programı **YÜKSEK LİSANS TEZİ** olarak kabul edilmiştir.

Prof. Dr. M. Elif KARSLIGİL
Yıldız Teknik Üniversitesi
Danışman

Jüri Üyeleri

Prof. Dr. M. Elif KARSLIGİL, Danışman
Yıldız Teknik Üniversitesi _____

Prof. Dr. Banu Diri, Üye
Yıldız Teknik Üniversitesi _____

Doç. Dr. Feza Buzluca, Üye
İstanbul Teknik Üniversitesi _____

Danışmanım Prof. Dr. M. Elif KARSLIGİL sorumluluğunda tarafımca hazırlanan bi-TEZAT: biLSTM Yöntemiyle Türkçe Şikayet Metinlerinde Zaman İfadelerinin Tespit Edilmesi başlıklı çalışmada veri toplama ve veri kullanımında gerekli yasal izinleri aldığımı, diğer kaynaklardan aldığım bilgileri ana metin ve referanslarda eksiksiz gösterdiğimi, araştırma verilerine ve sonuçlarına ilişkin çarpıtma ve/veya sahtecilik yapmadığımı, çalışmam süresince bilimsel araştırma ve etik ilkelerine uygun davrandığımı beyan ederim. Beyanımın aksinin ispatı halinde her türlü yasal sonucu kabul ederim.

Ensar EMİRALİ

İmza

Hayatımın güzel kadınlarına;

Anneme,

Eşime,

Ve kardeşlerime.

TEŞEKKÜR

Bu tez çalışmasında bana rehberlik eden değerli hocam Prof. Dr. M. Elif KARSLIGİL'e ve tüm bu süreçte güç veren sevgili eşim Nur Beyza'ya teşekkür ederim.

Ensar EMİRALİ

İÇİNDEKİLER

KISALTMA LİSTESİ ix

ŞEKİL LİSTESİ x

TABLO LİSTESİ xi

ÖZET xii

ABSTRACT xiv

1 GİRİŞ 1

1.1 Literatür Özeti 1

1.2 Tezin Amacı 6

1.3 Orijinal Katkı 7

2 VERİ KÜMESİNİN HAZIRLANMASI 9

2.1 Veri Kümesi 9

2.2 Etiketleme Standartları 10

2.2.1 TIMEX2 ve TIMEX3 11

2.2.2 Tarih İfadeleri Kuralları 12

2.2.3 Zaman İfadeleri Kuralları 14

2.2.4 Süre İfadeleri Kuralları 16

2.2.5 Tekrar İfadeleri Kuralları 17

2.2.6 İşaretlenen Karmaşık İfade Kuralları 18

2.2.7 İşaretlenmeyen İfade Kuralları 23

3 SİSTEM TASARIMI 26

3.1 Ön İşlemler 26

3.2 Model Girdileri 27

3.2.1 Kelime Vektörleri 27

3.2.2 Karakter Tabanlı Kelime Vektörleri 28

3.2.3 Ortografik Karakter Tabanlı Kelime Vektörleri 29

3.2.4 Morfolojik Analiz Sonucu Tabanlı Kelime Vektörleri 29

3.2.5 Yazınsal Öznitelik Vektörleri 30

3.2.6 BERT Kelime Öznitelikleri 31

3.3 Yöntemler 32
3.3.1 Karakter Tabanlı CNN Modeli 32
3.3.2 Öğe Tabanlı biLSTM Modeli 33
3.3.3 biLSTM ve CRF ile Zaman İfadelerinin Tespit Modeli 34
4 DENEYSEL SONUÇLARI 37
4.1 Kelime Vektörleri 38
4.2 İkili Vektörler 40
4.3 Morfolojik Analiz Sonucu Tabanlı Kelime Vektörleri 42
4.4 Yazınsal Öznitelik Vektörleri 43
4.5 Glove Kelime Vektörlerinin Diğer Vektörlerle Birlikte Kullanılması 45
4.6 BERT Kelime Özniteliklerinin Kullanılması 48
5 SONUÇ VE ÖNERİLER 51
KAYNAKÇA 53
TEZDEN ÜRETİLMİŞ YAYINLAR 60

KISALTMA LİSTESİ

ANN	Yapay Sinir Ağları *(Artificial Neural Network)*
BERT	Transformatörlerden Çift Yönlü Kodlayıcı Gösterimleri *(Bidirectional Encoder Representations from Transformers)*
biLSTM	Çift Yönlü Uzun Kısa Süreli Bellek *(Bidirectional Long Short-term Memory)*
CNN	Evrişimli Sinir Ağları *(Convolutional Neural Network)*
CRF	Şartlı Rastgele Alanlar *(Conditional Random Fields)*
GRU	Kapalı Tekrarlayan Birim *(Gated Recurrent Unit)*
LSTM	Uzun Kısa Süreli Bellek *(Long Short-term Memory)*
NER	Varlık İsmi Tanıma *(Named Entity Recognition)*
NLP	Doğal Dil İşleme *(Natural Language Processing)*
RNN	Yinelemeli Sinir Ağları *(Recurrent Neural Network)*
SGD	Olasılıksal Dereceli Azaltma *(Stochastic Gradient Descent)*
SVM	Destek Vektör Makinesi *(Support Vector Machine)*

ŞEKİL LİSTESİ

Şekil 1.1 TIMEX3 etiketleme şeması 2
Şekil 3.1 Metin uzunluklarının dağılımı 28
Şekil 3.2 Kelime uzunluklarının dağılımı 29
Şekil 3.3 Morfolojik analiz sonuç uzunluklarının dağılımı 30
Şekil 3.4 Karakter listeleri için CNN modeli 33
Şekil 3.5 Ortografik karakter listeleri için CNN modeli 33
Şekil 3.6 Morfolojik analiz sonuç listeleri için biLSTM modeli 34
Şekil 3.7 Zaman ifadelerini tespit için biLSTM + CRF modeli 35

TABLO LİSTESİ

Tablo 2.1 Veri kümesinde etiket sayıları 10
Tablo 2.2 Standart farklılıklarına örnek ifadeler 12
Tablo 3.1 Örnek bir metin için yapılan düzeltmeler ve değişiklikler 27
Tablo 4.1 Karışıklık matrisi 37
Tablo 4.2 Kelime vektörlerinin başarı değerleri 38
Tablo 4.3 Kelime vektörlerinde bulunma oranları 39
Tablo 4.4 onurgu kelime vektörlerinin etiket bazlı karışıklık matrisi 40
Tablo 4.5 glove kelime vektörlerinin etiket bazlı karışıklık matrisi 40
Tablo 4.6 fasttext kelime vektörlerinin etiket bazlı karışıklık matrisi 40
Tablo 4.7 Kelime vektörleri ve diğer vektörleri ile başarı değerleri 41
Tablo 4.8 Morfolojik analiz girdisine göre başarı değerleri 42
Tablo 4.9 Yazınsal özniteliklerin model başarısına etkileri 44
Tablo 4.10 "glove" kelime vektörleri ile diğer vektörlerin birlikte başarı değerleri 45
Tablo 4.11 glove + morph + fea modelinin etiket bazlı karışıklık matrisi 47
Tablo 4.12 Örnek bir metin için kelimelerin etiketleri ve etiket tahminleri 48
Tablo 4.13 Örnek bir metin için kelimelerin etiketleri ve etiket tahminleri 48
Tablo 4.14 Örnek bir metin için kelimelerin etiketleri ve etiket tahminleri 48
Tablo 4.15 BERT kelime öznitelikleri ile başarı değerleri 49
Tablo 4.16 BERT son 4 katmanın modelinin etiket bazlı karışıklık matrisi 50

ÖZET

bi-TEZAT: biLSTM Yöntemiyle Türkçe Şikayet Metinlerinde Zaman İfadelerinin Tespit Edilmesi

Ensar EMİRALİ

Bilgisayar Mühendisliği Anabilim Dalı

Yüksek Lisans Tezi

Danışman: Prof. Dr. M. Elif KARSLIGİL

Metinlerde tarih, saat, süre ve tekrar belirten zaman ifadelerinin otomatik olarak tespit eden sistemler geliştirilmesi, Doğal Dil İşleme (NLP) araştırma alanının çalışma konularına girmektedir. Literatür incelendiğinde, ilk geliştirilen sistemlerin zaman ifadelerinin desen analizleri sonucu oluşturulan kural tabanlı sistemler olduğu gözlemlenmektedir. Daha sonrasında metinden ve kelimenin özelliklerinden çıkarılan öznitelikleri kullanarak makine öğrenmesi modelleri yardımıyla zaman ifadelerini tespit eden hibrit sistemler geliştirilmiştir. Araştırma alanındaki gelişim süreciyle birlikte, Yapay Sinir Ağları (ANN) tabanlı sistem geliştirilmesine yönelik çalışmalar son zamanlarda literatürde yer almıştır.

Türkçe metinlerde yapılan çalışmalar incelendiğinde, Varlık İsmi Tanıma (NER) kapsamında tarih ve saat ifadelerinin tespit edilmesine yönelik olarak yapıldığı gözlemlenmektedir. Bu kapsamda NER kapsamına giren kişi, lokasyon ve organizasyon ifadeleriyle birlikte tarih ve saat ifadeleri tespit eden kural tabanlı, hibrit ve ANN tabanlı modeller geliştirilmiştir. Zaman ifadelerinin tespit edilmesi konu özelinde ilk çalışma, ifadelerin desenlerinin analiz edilmesidir. Daha sonrasında HeidelTime isimli sistem için Türkçe diline yönelik kural tanımlaması çalışması bulunmaktadır.

Bu tezin amacı, İngilizce dilinde NER kapsamında başarılı olan modellerden esinlenerek Türkçe metinlerde zaman ifadelerini tespit edecek ANN tabanlı bir modelin geliştirilmesi ve gerçekleştirilmektir. Öncelikle veri toplama ve etiketleme çalışması yapılmıştır. Bu kapsamda daha sonraki çalışmalarda da faydalı olacağı düşünülerek bir etiketleme dokümanı hazırlanmıştır. Veri kümesinde yer alan metinler, ön işlemlerden geçirildikten sonra tokenize edilmiştir. Modelin girdisi olan kelime, karakter, ortografik karakter, morfolojik analiz sonucu tabanlı kelime vektörleri ile yazınsal öznitelik vektörleri oluşturulmuştur. Kelime vektörleri için daha önceden eğitilmiş vektörler kullanılmıştır, bu vektörler üzerinde ince ayar yapılmamıştır. Kelimelerin yazılışları üzerinden Evrişimli Sinir Ağları (CNN) yardımıyla karakter ve ortografik karakter tabanlı kelime vektörleri oluşturulmuştur. Morfolojik analiz sonuçları üzerinden Çift Yönlü Uzun Kısa Süreli Bellek (biLSTM) yardımıyla kelime vektörleri oluşturulmuştur. Her bir kelimenin 15 yazınsal özniteliği üzerinden vektörler oluşturulmuştur. Ayrıca, Transformatörlerden Çift Yönlü Kodlayıcı Gösterimleri (BERT) modelinden öznitelik çıkarma yöntemiyle kelime vektörleri elde edilmiştir. Tüm bu girdileri kullanarak metinlerdeki zaman ifadelerini tespit eden biLSTM modeli ve etiket sınıflandırıcı olacak CRF katmanı tasarlanarak Türkçe dilinde ilk ANN tabanlı sistem geliştirilmiş ve başarısı değerlendirilmiştir.

Anahtar Kelimeler: Zaman ifadeleri, biLSTM, CRF, BERT, kelime vektörleri, karakter tabanlı kelime vektörleri

ABSTRACT

bi-TEZAT: Extract Temporal Expressions on Turkish Complaint Texts via biLSTM

Ensar EMİRALİ

Department of Computer Engineering

Master of Science Thesis

Supervisor: Prof. Dr. M. Elif KARSLIGİL

Developing systems for automatic detection of date, time, duration and set expressions on texts is within the scope of natural language processing (NLP) research field. When the literature is reviewed, it is observed that the first developed systems are rule-based which ones are results of pattern analysis of time expressions. Later, hybrid systems that use extracted features from texts and word properties with machine learning models are developed. Along with the development process in the research area, development of Artificial Neural Networks (ANN) based systems have recently taken place in the literature.

When the studies on Turkish texts are reviewed, it is observed that the detection of date and time expressions is done in the scope of Named Entity Recognition (NER). In this context, rule-based, hybrid and ANN-based models that detect person, location and organization expressions as well as date and time expressions have been developed. The first study about determination of time expressions is to analyze time expression patterns. Afterwards, there is a rule definition study for HeidelTime on Turkish texts.

The aim of this thesis is to develop and implement an ANN-based model to detect time expressions on Turkish texts, inspired by the models that have been successful in scope of NER in English. First, data collection and labeling work to be used in the

thesis is done. In this context, a guidance to label documents has been prepared to be useful in future studies. The texts in the dataset are preprocessed and then tokenized. Word embeddings, character, orthogonal character, morphological analysis results based and spelling feature word vectors are created. For word embeddings, pre-trained word embeddings are used, and these vectors are not fine-tuned during the training. Character and orthogonal character-based word vectors are created via a Convolutional Neural Network (CNN) model. Morphological analysis results-based word vectors are created via a Bidirectional Long Short-Term Memory (biLSTM) model. Spelling vectors based on 15 different attributes of each word are created. Besides, word embeddings are obtained by using feature extraction methods from Bidirectional Encoder Representations from Transformers (BERT) model. Then, to detect time expressions on the texts by using these inputs, as first ANN based system on Turkish language, biLSTM model and CRF layer as a label classifier is designed and evaluated.

Keywords: Temporal expressions, biLSTM, BERT, CRF, word embeddings, character based word embeddings

1
GİRİŞ

1.1 Literatür Özeti

Doğal Dil İşleme (NLP) çalışma konularından birisi, metinlerde yer alan zaman ifadelerinin geliştirilen sistemler aracılığıyla otomatik olarak tespit ve normalize edilmesidir. Bu tezde, zaman bilgisi içeren ifadelerin tespit edilmesi kısmına odaklanılmıştır. Haberlerin zamansal özetlerinin çıkarılması, metinlerin otomatik özetlenmesi, sorulara cevap veren sistemler ve metnin zaman çizelgesinin çıkarılması gibi uygulamalar için zaman ifadelerinin otomatik tespiti önemli bir rol oynamaktadır. Literatürde, zaman ifadelerinin etiketleme standartlarının belirlenmesi, zaman ifadeleri için külliyatlar oluşturulması ve metinlerdeki zaman ifadelerinin tespit edilmesine yönelik çalışmalar yer almaktadır.

Zaman ifadeleri, metindeki olay ve durumlarla ilgili "Ne zaman?", "Ne kadar süre?" ve "Ne sıklıkla?" sorulara cevap olacak ifadelerdir. Bunlara cevap verebilmek için zaman ifadelerinin otomatik tespit edilmesine ihtiyaç duyulur. Zaman ifadeleri, tarih, saat, süre ve tekrar olarak 4 farklı tip altında belirlenmektedir.

Metinlerdeki zaman ifadelerini etiketleme standartlarının belirlenmesine yönelik olarak Setzer ve Gaizauskas, zaman ifadelerini değerleri ve olaylarla ilişkilerini etiketlemeye yönelik bağımsız olarak çalışma yapmışlardır [1]. TIDES araştırma programı kapsamında zaman ifadelerini değer, niteleyici ve tekrar özellikleriyle etiketleme için TIMEX2 etiketleme standartları yayınlanmıştır. Bu standartlar, 2005 yılında yenilenerek tekrar yayınlanmıştır [2]. AQUIANT programı kapsamında olay ve zaman ifadelerini ve aralarındaki ilişkileri işaretlemek için TimeML dili geliştirilmiştir [3]. TimeML dili kapsamında zaman ifadelerine yönelik olarak TIMEX2 esas alınarak TIMEX3 etiketleme standartları belirlenmiş ve etiketler, tarih, saat, süre ve tekrar olarak 4 gruba ayrılmıştır. Buna yönelik olarak bir rehber yayınlanmıştır [4]. Rehbere göre, ifadelerin tipi ve değeri tespit edilmekte, tipine bağlı olarak ekstra özellikleri belirtilmektedir. Rehberde yer alan etiketleme şeması Şekil 1.1'de paylaşılmıştır.

```
attributes ::= tid type [functionInDocument][beginPoint][endPoint][quant][freq]
[temporalFunction] (value | valueFromFunction) [mod][anchorTimeID]
tid ::= ID
{tid ::= TimeID
TimeID ::= t<integer>}
type ::= 'DATE' | 'TIME' | 'DURATION' | 'SET'
beginPoint ::= IDREF
{beginPoint ::= TimeID}
endPoint ::= IDREF
{endPoint ::= TimeID}
quant ::= CDATA
freq ::= CDATA
functionInDocument ::= 'CREATION_TIME' | 'EXPIRATION_TIME' |
 'MODIFICATION_TIME' | 'PUBLICATION_TIME' |
 'RELEASE_TIME'| 'RECEPTION_TIME' | 'NONE'
 {default, if absent, is 'NONE'}
temporalFunction ::= 'true' | 'false' {default, if absent, is 'false'}

{temporalFunction ::= boolean}
value ::= CDATA
{value ::= duration | dateTime | time | date | gYearMonth |
 gYear | gMonthDay | gDay | gMonth}
valueFromFunction ::= IDREF
{valueFromFunction ::= TemporalFunctionID
TemporalFunctionID ::= tf<integer>}
mod ::= 'BEFORE' | 'AFTER' | 'ON_OR_BEFORE' | 'ON_OR_AFTER' |
 'LESS_THAN' | 'MORE_THAN' | 'EQUAL_OR_LESS' |
 'EQUAL_OR_MORE' | 'START' | 'MID' | 'END' | 'APPROX'
anchorTimeID ::= IDREF
{anchorTimeID ::= TimeID}
```

Şekil 1.1 TIMEX3 etiketleme şeması

Metinlerde zaman ifadelerinin otomatik tespit edilmesine yönelik günümüze kadar geliştirilen sistemler üç kategoriye ayrılabilir:

- Kural tabanlı
- Hibrit
- Yapay Sinir Ağları (ANN) tabanlı

Yapılan literatür araştırmasında, İngilizce dili dışında diğer dillerde geliştirilen sistemlerde İngilizce dili için yapılan çalışmaların esas alındığı gözlemlenmiştir. İngilizce dilinde zaman ifadelerini tespit eden kayda değer ilk çalışma, TARSQI projesinin bir parçası ve gözleme dayalı kurallardan oluşan GUTime isimli kural tabanlı sistem olmuştur [5]. GUTime, TIMEX2 standartlarında etiketleme amacıyla geliştirilmiş TempEx isimli etiketleme yönteminin TIMEX3 standartlarına göre etiketlenmiş metinler için geliştirilmiş versiyonudur. SemEval-2010 Task-13: TempEval-2 yarışmasında, HeidelTime, 86.0 F1 skoru ile en başarılı sistem olmuştur [6]. HeidelTime, metinlerdeki zaman ifadelerinin analizi sonucu oluşturulan kurallar ile çalışan bir sistemdir [7]. SemEval-2013 Task-1: TempEval-3 yarışmasında zaman ifadelerinin tespitinde en başarılı sistem, tespitinde 90.32 F1

skoru ile SUTime ve NavyTime olmuştur [8]. SUTime, kural tabanlı geliştirilmiş bir sistemdir [9]. NavyTime, SUTime'in uyarlaması olarak geliştirilmiş bir sistemdir [10]. Yarışmada, bu iki sistemi 90.30 F1 skoru ile HeidelTime takip etmiştir. HeidelTime, tarihi metinlerdeki günleri de tespit edecek şekilde geliştirilmiştir [11]. Çeşitli hastanelerden toplanan klinik notları veri kümesiyle düzenlenen i2b2 yarışmasında, zaman ifadelerinin tespitinde en başarılı 10 sistemin kural tabanlı veya Şartlı Rastgele Alanlar (CRF) ve Destek Vektör Makinesi (SVM) metotlarını hibrit yapıda olduğu gözlemlenmiştir [12].

Hibrit yapıdaki sistemlerde kayda değer ilk çalışma, Hacıoğlu'nun İngilizce ve Çince dilleri için metinden çıkardığı öznitelikleri kullanarak SVM ile sınıflandırma yapan çalışması olmuştur [13]. UzZaman ve Allen, kural bazlı özniteliklerin belirlendiği ve CRF kullanarak sınıflandırma yapan TRIPS/TRIOS adını verdikleri sistemi geliştirmişlerdir [14]. Kayda değer hibrit modellerden birisi de Filannino'nun geliştirdiği ve sınıflandırıcı olarak CRF kullanan ManTIME isimli sistemdir [15]. ManTIME, zaman ifadelerin tespitinde 89.66 F1 skoru ile SemEval-2013 Task-1: TempEval-3 yarışmasının en başarılı ilk üç sisteminden biri olmayı başarmıştır [8]. Zhong ve Cambria, zaman, değiştirici ve rakam olarak 3 öğeden oluşan yeni bir ön etiketleme şeması geliştirmişlerdir [16]. Geliştirdikleri şemayı ve metinden çıkardıkları öznitelikler ile CRF tabanlı TOMN ismini verdikleri sistem, 94.51 F1 skoru elde etmiştir [16].

Klinik notlarındaki zaman ifadelerinin otomatik tespitine yönelik yapılan SemEval-2015 Task-6 yarışmasında BluLab isimli sistem, zaman ifadelerinin tespit edilmesinde en başarılı sistem olmuştur [17]. BluLab, kural bazlı çıkarılan öznitelikler üzerinde SVM kullanan bir sistemdir [18]. SemEval-2017 Task-12 yarışmasının en başarılısı ise, GUIR isimli sistem olmuştur [19]. GUIR, sözcüksel ve kural bazlı çıkarılan öznitelikler üzerinde CRF ve karar ağaçları kullanmaktadır [20]. Ding, otomatik olarak zaman ifadeleri desenleri oluşturan ve Extended Budgeted Maximum Coverage modeli ile seçim yapan Ptime isimli sistem geliştirmiştir [21].

Literatürde yapılan çalışmalar incelendiğinde son zamanlarda ANN tabanlı zaman ifadelerini tespit eden modellerin geliştirilmeye başlandığı gözlemlenmektedir. Laparra ve Xu, Kapalı Tekrarlayan Birim (GRU) ve sınıflandırıcı olarak Sigmoid ve Softmax kullanarak metinde yer alan kelimeleri karakter tabanlı ele alarak zaman

ifadelerini tespit eden 7 farklı model geliştirmişlerdir [22]. Chen, metinden girdi olarak sadece metindeki kelimelerin karşılığı önceden eğitilmiş vektörleri kullanarak Transformatörlerden Çift Yönlü Kodlayıcı Gösterimleri (BERT) modelinde ince ayar yapmış ve hibrit modeller kadar başarılı bir sistem geliştirmiştir [57, 23]. Almasian ve Aumiller, Chen'in çalışmasını takip ederek BERT ve sınıflandırıcı olarak CRF kullanarak 3 farklı model geliştirmişlerdir [24].

İngilizce haricindeki diller için zaman ifadelerini tespit etmesine yönelik olarak HeidelTime üzerinde, dil özelinde kurallar tanımlanarak ve birçok dil için geçerli olacak kuralları belirleyerek çalışmalar yapılmıştır [25] [26]. Ayrıca Lehçe, Bask dili ve İspanyolca için olduğu gibi dil özelinde zaman ifadelerini tespit eden sistemler geliştirilmiştir [27] [28] [29].

Literatürde Türkçe için yapılan çalışmaları incelediğimizde, genellikle Varlık İsmi Tanıma (NER) kapsamında geliştirdikleri sistemlerin tespit edecekleri ifadelere tarih ve zaman gibi zaman ifadelerini dahil ettikleri gözlemlenmiştir. Zaman ifadelerinin tespit edilmesiyle ilgili gözlemlediğimiz ilk çalışma, olayların sıralanmasına yönelik geliştirilen sistemde, kelime köklerinden oluşturulan bir listedeki varlığına göre zaman ifadelerinin etiketlenmesidir [30]. Tatar, NER kapsamında kural tabanlı geliştirdiği sistemde 4 farklı kategorideki öznitelikleri kullanarak metindeki kişi, yer ve organizasyon ifadelerinin yanında tarih ve saat ifadelerini tespit etmektedir [31]. Kazkılınç, yine NER kapsamında CRF tabanlı geliştirdiği sistemde 4 farklı kategorideki öznitelikleri kullanarak metinlerdeki tarih ifadelerini tespit etmektedir [32]. Küçük, Türkçe twitlerde NER kapsamında kural bazlı geliştirdiği sistem ile tarih ve saat ifadelerini tespit etmektedir [33]. Küçük ve Küçük, Türkçe metinleri analiz ederek zaman ifadelerinin tespiti için kural tanımlama çalışması yapmışlar fakat bir sisteme dönüştürmemişlerdir [34]. Şeker, NER kapsamında CRF tabanlı geliştirdiği sistem ile tarih ve saat ifadelerini tespit etmektedir [35]. Akkaya, Çift Yönlü Uzun Kısa Süreli Bellek (biLSTM) ve CRF tabanlı modelinde oluşturduğu ortografik karakter ve karakter tabanlı kelime vektörleri ile çeşitli kelime vektörlerini kullanarak NER kapsamında kişi, lokasyon, organizasyon, para, yüzde ifadelerinin yanında tarih ve saat ifadelerini tespit etmektedir [36]. Çamcı, zaman ifadelerini tespit etmeye yönelik kural tabanlı bir sistem olan

HeidelTime'ın Türkçe için ifadelerini tespit kurallarını oluşturma çalışması yapmıştır [37].

Geliştirilen sistemlerin başarısını ölçmeye yönelik olarak kesinlik, hassasiyet ve F1 skorları kullanılmaktadır [38]. Değerlendirme aşamasında kesin eşleşme ve esnek eşleşme altında skorlar belirlenmektedir [38]. Kesin eşleşme, zaman ifadesinin tam sınırlarıyla tespit edilmesidir. Esnek eşleşme, zaman ifadesinin sınırlarıyla kesişecek şekilde tespit edilmesidir, kesişme dikkate alınarak hatalar göz ardı edilir. Örneğin, "25 Şubat sabahı" ifadesinin kesin eşleşme için "25 Şubat sabahı" olarak tespit edilmesi gerekir, fakat esnek eşleşme için "25 Şubat" ifadesi yeterlidir. Reimers ve Gurevych, yaptıkları çalışmada, başarı değeri olarak tek rakam raporlamanın yetersiz olduğunu gözlemlemiş ve model koşularının başarı ortalaması ile standart sapma değerlerini raporlamayı önermişlerdir [39].

Bu tezde geliştirilen modelde, İngilizce dili için NER kapsamında başarılı olan modeller örnek alınmıştır. Bu çalışmaların ilki, Huang'ın biLSTM ve sınıflandırıcı olarak CRF kullanarak geliştirdiği modeldir [40]. Huang'ın geliştirdiği modelin girdisi, kelime vektörleri ve yazımına göre kelimelerin öznitelikleri olmuştur. Kelime vektörleri olarak rastgele vektörler ve Senna kullanmayı denemiş ve daha önceden eğitilmiş Senna'nın daha iyi sonuç verdiğini gözlemlemiştir. Lample, Huang'ın modeline benzer bir model geliştirmiştir [41]. Huang'ın sisteminden ayrılan en belirgin özelliği, biLSTM kullanarak oluşturduğu karakter tabanlı vektörlerini modelinde girdi olarak ve biLSTM modelinin hem girdi hem çıktı tarafında Dropout kullanmasıdır. Kelime vektörleri olarak word2vec'in bir varyasyonunu kullanmıştır. Ma ve Hovy, Evrişimli Sinir Ağları (CNN) kullanarak karakter tabanlı vektörler oluşturmuş ve biLSTM ve CRF kullanarak geliştirdiği modelde girdi olarak kullanmışlardır [42]. Geliştirdikleri sistemin diğer en belirgin özelliği, kullandıkları modellerin ilk değerlerini Glorot ve He'nin çalışmalarını dikkate alarak belirlemişlerdir [43] [44]. Kelime vektörleri olarak GloVe'un en başarılı olduğunu gözlemlemişlerdir.

Devlin'in yapmış olduğu çalışmada, önceden eğitilmiş modelin çıktısı üzerinden çıkarılacak öznitelikleri kullanmanın avantajlı olduğu ifade edilmiştir [57]. NER kapsamında yaptıkları deneylerde, öznitelikleri çıkarma yöntemiyle elde edilen başarı değerleri, ince ayar yaparak elde edilen başarı değerlerine yakın çıkmıştır.

Peters ve Ruder, ELMo ve BERT modelleri için ince ayar ve öznitelik çıkarma yöntemlerini 7 farklı taskta deneyip başarılarını kıyaslamışlardır [58]. Çalışma sonucunda, çoğu durumda her iki yöntemin karşılaştırılabilir performansa sahip olduklarını belirlemişlerdir. Souza, Portekiz dilinde NER kapsamında yaptığı deneylerde ise öznitelik çıkarma yaklaşımının, ince ayar yaklaşımına göre önemli ölçüde kötü performans gösterdiğini gözlemlemiştir [59].

Geliştirilen modelin hiper parametrelerini belirlemek için İngilizce dilinde ANN tabanlı NER modellerinin hiper parametrelerine yönelik çalışmalar dikkate alınmıştır. Reimers ve Gurevych, yaptıkları çalışmada, karakter tabanlı CNN ve biLSTM modellerinin başarı skorları arasında önemli bir fark olmadığını, CNN modelinin daha hızlı olduğunu tespit etmişlerdir [45]. Ayrıca küçük veri kümeleri için 1 ile 16, büyük veri kümeleri için 8-32 arasında mini-batch boyutunun iyi bir seçim olduğunu gözlemlemişlerdir. Optimize edici olarak en başarılısı Nadam olmuştur. Yang'ın, yaptığı çalışmada Olasılıksal Dereceli Azaltma (SGD), en başarılı optimize edici olmuştur [46]. Sınıflandırıcı olarak Softmax yerine CRF kullanmanın model başarısını artırdığını gözlemlemişlerdir. Kelime vektörünü rastgele değerlerle oluşturmak yerine eğitilmiş vektörleri kullandıklarında daha yüksek başarı elde etmişlerdir. Aghaebrahimian ve Cieliebak'ın yaptıkları çalışmada eğitilmiş kelime vektörlerinin ince ayar yapılmadan kullanıldığı model, ince ayar yapılarak kullanıldığı modelden daha başarılı olmuştur [47].

1.2 Tezin Amacı

Günümüzde, büyük boyutlara ulaşan veri içerisinden otomatik anlamlı bilgi çıkaran sistemlerin önemi artmaktadır. Zaman ifadelerini tespit eden sistemler, örneğin doktorların klinik notlarından otomatik zaman ifadelerini tespit ettikten sonra oluşturulacak zaman çizelgesiyle uzun notları okurken gözden bir şey kaçırmamalarına yardımcı olacağı veya şirketlerin çok miktardaki şikayet verisi üzerinden zaman ifadelerini çıkararak şikayetin seyrini oluşturabilecekleri gibi birçok durumda çözümlere yardımcı olmaktadır.

Bu tezin amacı, farklı tabanlarda oluşturulan kelime vektörlerine metinden çıkarılacak öznitelikleri ekleyerek İngilizce dili için NER kapsamında geliştirilen modellerden esinlenerek Türkçe metinlerdeki tarih, saat, süre ve tekrar belirten zaman ifadelerini tespit eden ANN tabanlı model geliştirilmesidir. Çalışma

kapsamında CNN modeli kullanarak karakter ve ortografik karakter tabanlı kelime vektörleri, biLSTM modeli kullanarak morfolojik analiz sonucu tabanlı kelime vektörleri oluşturulacaktır. Ayrıca BERT modelinin çıktı katmanları üzerinden kelime vektörleri oluşturulacaktır. Sonuç kısmında bu vektörlerin farklı kombinasyonlarda birlikte kullanılmalarının model başarısına katkıları değerlendirilecektir. Ayrıca çalışma kapsamında 5 farklı kelime vektörünün model başarısı üzerindeki etkisi analiz edilecektir.

Tezin sonunda geliştirilen modelin akademik ve ticari çalışmalarda rahatlıkla kullanılabilmesi hedeflenmiştir. Bu sebeple modele girdi oluşturacak kaynak seçimindeki tek kriter, kaynağın herhangi bir beklenti olmadan topluma açık olması olmuştur.

1.3 Orijinal Katkı

Bu tez kapsamında Türkçe metinlerde zaman ifadelerinin tespiti konusunda literatüre yapılacak katkılar, aşağıda özetlenerek listelenmiştir:

- Gelecekte yapılacak zaman ifadelerinin etiketlenmesi çalışmalarında yardımcı olması için TIMEX3 standartlarının Türkçe metinler için uyarlama çalışması yapılmıştır.
- Veri kümesi olarak kirli nitelendirilen şikayet metinleri kullanılmış ve bu metinlerdeki zaman ifadeleri etiketlenmiştir.
- Türkçe metinlerde zaman ifadelerinin tespiti özelinde ANN tabanlı bir model geliştirilmiş ve başarısı değerlendirilmiştir.
- Türkçe metinlerdeki zaman ifadeleri için farklı yöntemlerle oluşturulmuş kelime vektörlerinin başarıları kıyaslanmıştır.
- Türkçe metinlerdeki zaman ifadeleri için farklı tabanlarda kelime vektörlerinin farklı kombinasyonlarda birlikte kullanılmasıyla elde edilen model başarıları kıyaslanmıştır.
- Zemberek-nlp kullanarak Türkçe metinlerinde morfolojik analiz sonucu tabanlı kelime vektörlerinin model başarısının paragraf ve cümle tabanlı kıyaslanması yapılmıştır.

- Daha önceden eğitilmiş BERT modeli çıktıları üzerinden çeşitli kelime öznitelikleri çıkarma yöntemiyle elde edilen kelime vektörleri kullanılarak model başarısı değerlendirilmiştir.

2
VERİ KÜMESİNİN HAZIRLANMASI

Yapılan literatür çalışması sonucunda, bu çalışma için ihtiyaç duyulduğu gibi tarih, saat, süre ve tekrar belirten zaman ifadelerinin TIMEX3 standartlarında etiketlendiği bir Türkçe veri kümesinin olmadığı gözlemlenmiştir. Bu yüzden, çalışma için açık kaynaklardan şikayet metinleri toplanıp veri kümesi oluşturulmuştur. Oluşturulan veri kümesi için etiketleme işlemine başlandığında bir doküman eksikliği hissedilmiştir. Bu eksikliği gidermek amacıyla TIMEX3 standartlarının Türkçe metinler için uyarlama çalışması yapılmıştır. Bu kısımda veri kümesini oluşturma süreci ve uyarlanan standart kuralları anlatılmaktadır.

2.1 Veri Kümesi

Yapılan değerlendirmede, şikayet metinlerinde zaman ifadelerine daha sık karşılaşıldığı gözlemlenmiştir. Bu amaçla internet ortamında herkese açık olan siteler üzerinden 23.127 Türkçe şikayet metni toplanmıştır. Hazırlanan TIMEX3 standartlarında etiketleme dokümanı rehberliğinde toplanan metinlerden rastgele 500 tanesi üzerinde etiketleme çalışması yapılmıştır. Etiketleme süresince açık kaynak yazılım olan GATE[1] uygulaması kullanılmıştır. Etiketleme işlemi tamamlandıktan sonra, seçilen 500 metin içerisinde aynı konuda aynı uzunlukta çok kez tekrar eden metin olduğu görülmüş ve modelde önyargı oluşturma ihtimaline karşı seçilen metinlerden bazıları silinmiştir. Tez çalışması aşamasında, geriye kalan etiketlenmiş 482 şikayet metni kullanılmıştır. Bu metinlerde tokenizasyon işlem sonucunda toplamda 35.384 token yer almaktadır. Veri kümesindeki metinlerin 385 tanesi eğitim sırasında kullanmak için ayrılmış, geri kalan 97 tanesi ise modelin başarısını ölçerken kullanılmıştır. Veri kümesinde yer alan etiket sayıları, token bazlı olarak Tablo 2.1'de paylaşılmıştır. "<unk>" etiketi, zaman ifadesi olmayan token için kullanılmıştır.

[1] https://gate.ac.uk/

Tablo 2.1 Veri kümesinde etiket sayıları

Etiket	Eğitim kümesinde token sayısı	Başarı ölçüm kümesinde token sayısı	TOPLAM
<unk>	26.506	6.708	33.214
TARİH	906	206	1.112
SAAT	107	30	137
SÜRE	649	150	799
TEKRAR	78	44	122

Etikelenen değerler üzerinde yapılan analizde, bu değerlerin %71'inin sadece harflerden oluştuğu %21'inin ise sadece rakamlardan oluştuğu gözlemlenmiştir. Ayrıca, değerlerin %5'i noktalama işareti içermektedir. Etiketlenen değerlerin %89'unun öncesinde ve/veya sonrasında etiketlenmiş bir değer bulunduğu gözlemlenmiştir.

2.2 Etiketleme Standartları

2000 yılında TIDES (Translingual Information Detection, Extraction, and Summarization) araştırma programı kapsamında TIMEX2 etiketleme standartları geliştirilmiştir. Bu standartlar 2005 yılında yenilenerek tekrar yayınlanmıştır [2].

2005 yılında Pustejovsky ve diğ., AQUAINT programı kapsamında olay ve zaman ifadelerini işaretlemek için TimeML dilini geliştirmişlerdir. Bu kapsamda yer alan TIMEX3 etiketi ile zaman ifadeleri, tarih, zaman, süre ve tekrar olarak 4 grupta nasıl işaretleneceğine yönelik bir rehber yayınlanmıştır [4].

Bu tez çalışmada, TIMEX3 standartlarında etiketleme rehberinde yer alan etiketleme kurallarının Türkçe metinler için uyarlaması yapılmıştır. Türkçe zaman ifadeleri için yapılacak işaretleme çalışmaları başka diller üzerinde çalışan insanlar için de anlaşılır olabilmesi için öznitelikler ve kısaltmaların İngilizce karşılıkları aynen kullanılmış, Türkçe'ye çevrilmeye çalışılmamıştır. Verilen örneklerde referans tarihi olarak 2021-04-14 (2021-W15) alınmıştır. İfadelerin normalleştirilmemiş değerleri belirtilmiştir.

Standartların Türkçe'ye uyarlanışında TIMEX3 etiketlemede kullanılan özniteliklerden sadece aşağıdaki üçü kullanılacaktır:

- *tid:* Etiketin tüm külliyat içerisinde tekil numarası
- *type:* Etiketin tipi; "DATE" | "TIME" | "DURATION" | "SET" (tarih|zaman|süre|tekrar)
- *value:* Etiketin değeri; metnin içeriğine göre referans tarihi belirlenir ve ifadeye göre değer hesaplanır

2.2.1 TIMEX2 ve TIMEX3

TIMEX3 standartları, birçok açıdan TIMEX2 standartları ile benzer olsa da aralarında bazı önemli farklar bulunmaktadır; durumlara örnek olarak metinlerde geçen ifadelere Table 2.2'de yer verilmiştir:

- TIMEX3'te, TIMEX2'de olduğu gibi VAL ve MOD öznitelikleri aynen kullanılmış, ancak SET, PERIODICITY ve GRANULARITY öznitelikleri kullanılmamıştır. Bunların yerine ve fazladan etiketin tekil numarası olan TID, etiket tipini belirten TYPE ile değerin kesin zaman bilgisinin tespiti için fonksiyona ihtiyacı olup olmadığını belirten TEMPORALFUNCTION öznitelikleri vardır.
- Bağlaç ve edat ile bağlanan zaman ifadeleri, TIMEX3'te yer alan SIGNAL etiketinden dolayı TIMEX2'ye göre daha küçük parçalara ayrılarak etiketlenmiştir: Örnek 1.
- Olaya bağlı zaman ifadeleri, TIMEX2'de olay ifadesiyle birlikte etiketlenirken TIMEX3'te olay kısmı EVENT olarak etiketlenmiştir: Örnek 2.
- Kendinden sonra gelen ifadeyi değiştiren ifadeler, TIMEX2'de iç içe olarak etiketlenirken TIMEX3'te tek olarak etiketlenmiştir: Örnek 3.

Tablo 2.2 Standart farklılıklarına örnek ifadeler

Örnek No	İfade	TIMEX2 Etiketleme	TIMEX3 Etiketleme
1	two weeks from next Tuesday	<TIMEX2 VAL="1999-08-03"> two weeks from <TIMEX2 VAL="1999-07-20"> next Tuesday </TIMEX2>	<TIMEX3 tid="t1" type="DURATION" value="P2W"> two weeks </TIMEX3> <SIGNAL sid="s1"> from </SIGNAL> <TIMEX3 tid="t2" type="DATE" value="2002-07-02"> next Tuesday </TIMEX3>
2	five days after he came back	<TIMEX2 VAL="XXXX-XX-XX"> five days after he came back </TIMEX2>	<TIMEX3 tid="t1" type="DURATION" value="P5D"> five days </TIMEX3> after he <EVENT eid="e1" class="OCCURRENCE"> came </EVENT> back
3	this year's summer	<TIMEX2 VAL="1999-SU"> <TIMEX2 VAL="1999"> this year </TIMEX2>'s summer </TIMEX2>	<TIMEX3 tid="t1" type="DATE" value="1999-SU"> this year's summer </TIMEX3>

2.2.2 Tarih İfadeleri Kuralları

- Kesin tarih belirten ifadeler YYYY-MM-DD formatında işaretlenir:

<TIMEX3 tid="t1" type="DATE" value="2021-01-10"> 10.01.2021 tarihinde </TIMEX3> kredi kartımdan alınan üyelik aidatının tarafıma iadesini istiyorum.

Yurtiçi Kargo'nun <TIMEX3 tid="t2" type="DATE" value="2021-04-13"> dün </TIMEX3> getirdiği kargoda kırık ürün vardı.

Daha sonra <TIMEX3 tid="t3" type="DATE" value="2021-04-12"> 12 Nisan 2021 tarihinde </TIMEX3> 3 defa aramama rağmen beni çok beklettiler.

- Tarih ifadeleri, formatın en solundan başlanılarak işaretlenir. Formatın solundan itibaren belirtilmeyen kısımlar X ile doldurulur; formatın sağında eksik olan kısımlar ise doldurulmaz:

<TIMEX3 tid=”t1” type=”DATE” value="2020"> Geçen sene </TIMEX3> internet üzerinden Corona kapsamında kredi çektim.

<TIMEX3 tid=”t2” type=”DATE” value="XXXX-04-01"> 1 Nisan tarihinde </TIMEX3> eşimle birlikte kullandığımız Aile 6 paketi taahhüdü bitmek üzere denildi.

Amerikan gençliği arasında yeşeren bu yeni akım, <TIMEX3 tid=”t3” type=”DATE” value=“XX60”> 60'ları </TIMEX> domine etti.

Siparişimi <TIMEX3 tid=”t4” type=”DATE” value="2020-06"> 2020 haziran ayında </TIMEX> daha sonra birkaç gün içinde iptal ettim.

- Yıl ifadesi 4 rakamdan (YYYY) oluşur; ilk rakam milenyum, ikincisi yüzyıl, üçüncüsü onyıl, dördüncüsü ise yıl bilgisini verir. Yıl belirten ifadeler, 4ten küçük sayıda rakamın yan yana gelmesinden oluşabilir. 1000 yılı öncesi yıllarda 4 rakama tamamlamak için önüne bir sıfır eklenir:

İstanbul’da <TIMEX3 tid=”t1” type=”DATE” value="197"> 1970'lerde </TIMEX3> en çok konuşulan konularından biriydi.

Çinli bir kadın savaşçı hakkında Mu Lan'ın hikayesi <TIMEX3 tid=”t2” type=”DATE” value="0500"> MS 500'de </TIMEX3> yazılmıştır.

Romanın ana karakteri, <TIMEX3 tid=”t3” type=”DATE” value="17"> 17. yy.'da </TIMEX3> istanbul'da yaşayan bir yeniçeridir.

- Mali yıllar, FY kısaltmasıyla işaretlenir:

Yardımın <TIMEX3 tid="t1" type="DATE" value="FY1998"> 1998 mali yılında </TIMEX3> yaklaşık 37 milyon dolar olması bekleniyordu.

- Çeyrek yıllar Q, yarı yıllar H kısaltmasıyla işaretlenir. Eğer açık şekilde herhangi bir yılın bir parçası olarak belirtiliyorlarsa, mali yıl ile birlikte de kullanılabilir:

Dow, <TIMEX3 tid="t1" type="DATE" value="2020-Q4"> 4. çeyrekte </TIMEX3> yüzde 17 yükseldi.

Masraflar <TIMEX3 tid="t2" type="DATE" value="FY1998-H1"> 1998 mali ilk yarı yılında </TIMEX3> arttı.

- Mevsim belirten ifadeler, İngilizce kuraldaki kısaltmalarla ilkbahar (SP), yaz (SU), sonbahar (FA) ve kış (WI) şeklinde işaretlenir:

<TIMEX3 tid="t1" type="DATE" value="2021-SU"> Bu yaz </TIMEX3> sıcaklar rekor kırdı.

Üniversite tarafından <TIMEX3 tid="t2" type="DATE" value="2021-FA"> bu yılın güz dönemi </TIMEX3> akademik tavimi açıklandı.

- Hafta olarak zaman belirten ifadeler, YYYY-Wnn-DD formatıyla işaretlenir:

Bugün aradığımda ise <TIMEX3 tid="t1" type="DATE" value="2021-W15"> bu hafta </TIMEX3> yoğunduk diye bir tepkiyle karşılaştım.

Kurulum için <TIMEX3 tid="t2" type="DATE" value="2021-W16-1"> haftaya pazartesi </TIMEX3> geleceklerini söylediler.

- Hafta içi ifadeleri, hafta formatında işaretlenir:

<TIMEX3 tid="t1" type="DATE" value="2021-W16 "> Önümüzdeki hafta içinde </TIMEX3> Bursa'ya gitmem gerekecek.

- Haftanın günleri, hafta tarih formatının sağına Pazartesi 1.gün olmak haftanın gün sırası eklenerek işaretlenir:

<TIMEX3 tid="t1" type="DATE" value="XXXX-WXX-1 "> Pazartesi </TIMEX3> başlattığım nakil işlemi için 7 iş günü verildi.

2.2.3 Zaman İfadeleri Kuralları

- Gün içerisindeki zamanı belirten ifadeler THH:MI:SS formatıyla işaretlenir:

Ne zaman geleceğini sorduğumda <TIMEX3 tid ="t1" type="TIME" value="T15:30"> 15.30 </TIMEX3> gibi burada olacağını söyledi.

Gelen bilgilendirme mesajı <TIMEX3 tid ="t2" type="TIME" value="T03:07"> gece 03.07 saatinde </TIMEX3> gelmiştir.

- Zamanın tarihi belli ise tarih bilgisi, zaman formatının soluna tarih formatı ile ilave edilir:

Evde internetim <TIMEX3 tid =”t1” type=”TIME” value="2021-04-14T09:00"> bugün sabah 9’dan </TIMEX3> beri yok.

Bir kere uyanınca <TIMEX3 tid =”t2” type=”TIME” value="T05:00"> gece 5’ten </TIMEX3> sonra bir daha uyuyamadım.

- Zaman dilimi ifadeleri işaretlenir. Greenwich saati, zaman gösterimi sonunda Z ile, diğer zaman dilimleri +- ile gösterilir:

<TIMEX3 tid =”t1” type=”TIME” value="1996-04-11T11:13Z"> 11 Nisan 1996 11:13 GMT </TIMEX3>.

<TIMEX3 tid =”t2” type=”TIME” value="1994-01-21T08:29-05"> 21 Ocak 1994 08:29 Doğu Zaman Dilimi </TIMEX3>.

- Günün vaktini belirten ifadeler, İngilizce kuraldaki kısaltmalarla sabah (MO), öğlen (MI), öğleden sonra (AF), akşam (EV), gece (NI) ve çalışma saatleri (DT) şeklinde işaretlenir:

<TIMEX3 tid =”t1” type=”TIME” value="TEV"> Akşam </TIMEX3> tekrar aradığımda, kuryenin 10 Nisan'da dağıtıma çıktığı söylendi.

- Ayrıca Türkçe’de yer alan ve günün vaktini belirten ifadeler, gündüz saatleri (MO), akşamüstü (EV) ve gün boyu (DT) ile işaretlenir:

Havayollarından rica minnet uçuş saatini değiştirdik, <TIMEX3 tid =”t1” type=”TIME” value="TEV"> akşamüstüne </TIMEX3> aldırdık.

Bu dizinin <TIMEX3 tid =”t2” type=”TIME” value="TMO"> gündüz saatlerine </TIMEX3> alınmasını önemle rica ettim.

- “gün içinde”, “gün içerisinde” ifadeleri, tarih bilgisi sonrasına (MO) kısaltması eklenerek zaman olarak işaretlenir:

Banka ve şirket ile <TIMEX3 tid =”t1” type=”TIME” value="XXXX-XX-XXTMO"> aynı gün içerisinde </TIMEX3> iletişime geçtim ve kredi kartımı kullanıma kapattım.

- Gece yarısı ifadesi, bir gün öncesinin T24:00 zamanı olarak işaretlenir:

<TIMEX3 tid =”t1” type=”TIME” value=“2021-01-10T24:00”> Gece yarısı </TIMEX3> boyunca hep ayakta haber bekledim.

- Saat belirtilmeden “geçen gece” ifadesi, bir önceki güne işaretlenir:

Senin bahsettiğin maç <TIMEX3 tid =”t1” type=”TIME” value="2021-04-13TNI"> geçen gece </TIMEX3> oynandı.

2.2.4 Süre İfadeleri Kuralları

- Süre belirten ifadeler, P kısaltmasının yanına uzunluğu ve İngilizce kuraldaki kısaltmalarla saniye (S), dakika (M), saat (H), gün (D), hafta (W), ay (M) ve yıl (Y) şeklinde birimi ilave edilerek işaretlenir:

<TIMEX3 tid=”t1” type=”DURATION” value="P3M"> 3 aydır </TIMEX3> mail adresime ödeme bilgisi gelecek

Nakil işlemi başlattım, neredeyse <TIMEX3 tid=”t2” type=”DURATION” value="P1W"> bir hafta </TIMEX3> olacak hala bir sonuç yok.

Zaten <TIMEX3 tid=”t3” type=”DURATION” value="P3D"> 3 günlük </TIMEX3> farkı <TIMEX3 tid=”t4” type=”DURATION” value="P4D"> 4 gün </TIMEX3> olarak Mayıs ayına yansıttınız.

- “tüm gün”, “bütün gün” gibi ifadeler, 1 günden daha kısa bir süreyi ifade ettiği düşünülerek (DT) kısaltmasıyla işaretlenir:

Mesaj atılıp evde olmadığım söylendi fakat <TIMEX3 tid=”t1” type=”DURATION” value="P1DT"> tüm gün </TIMEX3> evdeydim.

Paketiniz dağıtıma çıkarıldı diye mesaj geldi ama <TIMEX3 tid=”t2” type=”DURATION” value="P1DT"> bütün gün </TIMEX3> ne arayan ne soran var.

- “iş günü” ifadesi, gün gibi işaretlenir:

Bir açıklama gelmedi ve TIMEX3 tid=”t1” type=”DURATION” value="P10D"> 10 iş gününü </TIMEX3> geçeli çok oldu.

- “yaş” kelimesini içeren ve yaş belirten ifadeler, (Y) kısaltmasıyla süre olarak işaretlenir:

Hatanız yüzünden TIMEX3 tid=”t1” type=”DURATION” value="P8Y"> 8 yaşındaki </TIMEX3> çocuk TIMEX3 tid=”t2” type=”DURATION” value="P1W"> 1 haftadır </TIMEX3> kitabının nerede olduğunu soruyor.

- Belli bir sürenin kaçıncı biriminde olduğunu belirten ifadeler süre olarak işaretlenir:

Tarife detaylarına girdiğimde ise tarifenizin <TIMEX3 tid="t1" type="DURATION" value="P6M"> 6. ayındasınız </TIMEX3> yazıyor.

İletişime geçilmediği için değişikliği <TIMEX3 tid="t2" type="DURATION" value="P3D"> 3. gün </TIMEX3> fark ettim.

<TIMEX3 tid="t3" type="DURATION" value="P1M"> İlk ayın </TIMEX3> faturasına kurulum ücreti diye bir kalem yansıtılmış.

- Yuvarlama amaçlı birden fazla süre belirten ifadeler ayrı ayrı olarak aynı birimle işaretlenir:

Yaklaşık <TIMEX3 tid="t1" type="DURATION" value="P3D"> 3 </TIMEX3> - <TIMEX3 tid="t2" type="DURATION" value="P4D"> 4 gündür </TIMEX3> almamız gereken hizmeti alamıyoruz.

- Belirsiz çoğul süre ifadelerinde, uzunluğunu belirtmek için X kullanılır:

<TIMEX3 tid="t1" type="DURATION" value="PXM"> Birkaç ay </TIMEX3> önce ekipler altyapı için projelendirme çalışması adına geldiler.

Bunun dışında <TIMEX3 tid="t2" type="DURATION" value="PXD"> günlerdir </TIMEX3> bir çok müşteri hizmetleri numarasını aramama rağmen yetkili kimseye ulaşamadım.

Mağazanızdan <TIMEX3 tid="t3" type="DURATION" value="PXY"> yıllardır </TIMEX3> alışveriş yaparım.

2.2.5 Tekrar İfadeleri Kuralları

- Tekrar ifadeleri, süre ifadelerinde olduğu gibi, tekrar periyot uzunluğu ve yanında İngilizce kuraldaki kısaltmalarla saniye (S), dakika (M), saat (H), gün (D), hafta (W), ay (M) ve yıl (Y) şeklinde birimi ilave edilerek işaretlenir:

Kargo bizim buraya <TIMEX3 tid="t1" type="SET" value="P1W"> haftada 1 kez </TIMEX3> geliyormuş.

Kredi kartımı bankanız bana verirken kesinlikle <TIMEX3 tid="t2" type="SET" value="P1Y"> yıllık </TIMEX3> abonelik ücreti olmayacağını ısrarla söylemiştir.

Bilgi mesaj uyarısı <TIMEX3 tid="t3" type="SET" value="P1M"> ayda birkaç kez </TIMEX3> ekranıma geliyor.

Paris'in tamamında hız limiti <TIMEX3 tid="t4" type="SET" value="P1H"> saatte </TIMEX3> 30 kilometreyle sınırlandırıldı.

Çalıştığım yere neredeyse <TIMEX3 tid="t5" type="SET" value="P1D"> her gün </TIMEX3> kargo teslim ediliyor.

TIMEX3 tid="t6" type="SET" value="P1D"> Günde </TIMEX3> 1500 kişiye kargo dağıtıyorum.

- Tekrar ifadelerinde tarihi belirtiliyorsa tarih formatında işaretlenir:

<TIMEX3 tid="t1" type="SET" value="XXXX-04-23"> Her 23 Nisan'da </TIMEX3> olduğu gibi, yine bayram coşkuyla kutlandı.

Patronun isteğiyle bundan sonra <TIMEX3 tid="t2" type="SET" value="XXXX-WXX-5"> her Cuma </TIMEX3> durum raporu çıkılacak.

<TIMEX3 tid="t3" type="SET" value="XXXX-WXX-WE"> Hafta sonları </TIMEX3> yasak olduğu için beni kuryenin evde bulamama ihtimali yok.

- Zamanı net olmayan tekrar ifadeleri de işaretlenir:

<TIMEX3 tid="t1" type="SET" value="XXXX-03"> Mart ayı </TIMEX3> genellikle yağışlıdır.

Arkadaşları neredeyse <TIMEX3 tid="t2" type="SET" value="XXXX-WXX"> her hafta </TIMEX3> görüştüklerini söylüyor.

2.2.6 İşaretlenen Karmaşık İfade Kuralları

- Geçmiş (PAST_REF), şimdi (PRESENT_REF) veya geleceğe (FUTURE_REF) atıf yapan zaman ifadeleri, doküman referans tarihiyle uyumlu şekilde belirteçler kullanılarak tarih olarak işaretlenir:

<TIMEX3 tid="t1" type="DATE" value="PAST_REF"> Geçenlerde </TIMEX3> aynı şekilde dağıtımda E blok yerine C bloğa gittiğini tespit etmiştim.

Altyapı dilekçesini <TIMEX3 tid="t2" type="DATE" value="PAST_REF"> daha önceden </TIMEX3> il müdürlüğüne vermiştik.

Eşim <TIMEX3 tid="t3" type="DATE" value="PAST_REF"> az önce </TIMEX3> sizi aradı.

<TIMEX3 tid="t4" type="DATE" value="PAST_REF"> Zamanında </TIMEX3> internet siteleri için digital koruma almıştım

Sipariş geçmişime bakarsanız <TIMEX3 tid="t5" type="DATE" value="PAST_REF"> ilk zamanlarınızdan </TIMEX3> beri alışveriş yaptığımı göreceksiniz.

Karar uyarınca <TIMEX3 tid="t6" type="DATE" value="PAST_REF"> geriye </TIMEX3> dönük kesilen kart aidatlarının iadesinin olacağına dair kesin hükmü bulunmaktadır.

Bu tarife yeterli olmazsa <TIMEX3 tid="t7" type="DATE" value="PAST_REF"> eski </TIMEX3> tarifeme geçebilecek miyim diye sordum.

İlk önce yanlış adrese teslim ettiler, <TIMEX3 tid="t1" type="DATE" value="PRESENT_REF"> şimdi </TIMEX3> de 3 gündür oyalıyorlar.

<TIMEX3 tid="t2" type="DATE" value="PRESENT_REF"> Şu an </TIMEX3> işleminizi gerçekleştiremiyoruz.

<TIMEX3 tid="t3" type="DATE" value="PRESENT_REF"> Bu zamana </TIMEX3> kadar tüm kargoları ben aldım.

Pandeminin baş gösterdiği <TIMEX3 tid="t4" type="DATE" value="PRESENT_REF"> bu günlerde </TIMEX3> zor da olsa kıt kanaat müşterilerimize hizmet vermeye çalışıyoruz.

<TIMEX3 tid="t5" type="DATE" value="FUTURE_REF"> İleriki bir tarihte </TIMEX3> teslimat için bekletiliyor.

- "güncel" kelimesi içeren zaman ifadeleri, değeri (PRESENT_REF) olarak işaretlenir:

Yeni adresime dair TIMEX3 tid="t1" type="DATE" value="PRESENT_REF"> güncel tarihli </TIMEX3> ıslak imza kaşeli ikamet belgesinin sunulması gerektiği belirtilmiştir.

- Hiyerarşi ilişkisine sahip ifadeler aralarına işaretleme olsa da tek etiketleme ile işaretlenir:

<TIMEX3 tid="t1" type="TIME" value="2021-04-13T12:50"> 13 Nisan 2021 saat 13.15'de </TIMEX3> müşteri hizmetleri ile yaptığım görüşmede sorunun modemden olduğu ve modem firmasını aramam gerektiği söylendi.

Evde bulunduğum halde <TIMEX3 tid="t2" type="TIME" value="2021-04-13T10:10:27"> 13/04/2021 - 10:10:27 de </TIMEX3> evde yok gerekçesiyle "iade-tanınmıyor" iade olmuştur.

- Tarih ile zaman ifadeleri arasındaki kelimede "-de", "-den" ekleri olduğunda ayrı ayrı işaretlenir:

<TIMEX3 tid="t1" type="DATE" value="2021-04-13"> 13 Nisan 2021 tarihinde </TIMEX3> <TIMEX3 tid="t1" type="TIME" value="T12:50"> saat 12.50'de </TIMEX3> müşteri hizmetleri ile yaptığım görüşmede sorunun modemden olduğu ve modem firmasını aramam gerektiği söylendi.

- Ay-Gün formatındaki tarih ve haftanın günü ifadeleri yan yana kullanılmış ise ayrı ayrı işaretlenir:

<TIMEX3 tid="t1" type="DATE" value="XXXX-04-08"> 8 Nisan </TIMEX3> <TIMEX3 tid="t2" type="DATE" value="XXXX-WXX-4"> Perşembe günü </TIMEX3> beklenen haber geldi.

- Bir doküman, birden fazla referans tarihe sahip olabilir. Eğer bir dokümanın yayınlanma ve yazılma tarihleri var ise yazdığı tarih referans alınır. Belge içeriğine göre referans tarihi belirlenmesi gerekir:

<TIMEX3 tid="t1" type="DATE" value="1848"> 1848'de </TIMEX3> Marx, <TIMEX3 tid="t2" type="DATE" value="FUTURE_REF"> geleceğin </TIMEX3> itici gücünün siyasetin değil, ekonomi olacağını öngördü.

<TIMEX3 tid="t3" type="TIME" value="2021-04-13TEV"> Dün akşam </TIMEX3> numaramı eklemek istediğimde "<TIMEX3 tid="t4" type="DATE" value="PRESENT_REF"> Şu an </TIMEX3> işleminizi gerçekleştiremiyoruz" diye hata verdi.

- Kalıplaşmış tarih belirten isim tamlamaları, değer belirtilmeden işaretlenir:

Ne zaman kargolanacak diye sorduğumda <TIMEX3> teslimat tarihi </TIMEX3> paylaşılmıyor.

Ürünlerin üzerinde <TIMEX3> tüketim tarihi </TIMEX3> yazmadığını fark ettim.

Faturamın <TIMEX3> son ödeme tarihinde </TIMEX3> ödemeyi yaptım.

<TIMEX3> Sınav haftasındayım </TIMEX3> ve <TIMEX3 tid="t1" type="DATE" value="2021-04-15"> yarın </TIMEX3> da iki tane sınavım var.

Böyle bir süreçte pandemiden kaynaklı olsun, <TIMEX3> çalışma saatlerimiz </TIMEX3> olsun bankaya gitmeye uygun değil.

- Doğumgünü ve yıl dönümleri işaretlenir; tarihi belli değil ise değer belirtilmez:

Ancak tahtın varisi, <TIMEX3 tid="t1" type="DATE" value="XXXX-WXX-5"> Cuma günü </TIMEX3> onuruna düzenlenen <TIMEX3> 50. doğum günü </TIMEX3> partisinde yine de hit oldu.

Eşime çiçek sepetinden <TIMEX3> doğum günü </TIMEX3> hediyesi aldım.

- Herkesin bildiği tatil günleri ve olaylar işaretlenir; zamanı belli değil ise değer belirtilmez:

Kuzenlerim <TIMEX3> Noel'de </TIMEX3> ziyarete geldiler.

- Belirttiği zaman üzerine ittifakın sağlanamayacağı ve kültürel veya bireysel rutinleri belirten zaman ifadeleri işaretlenir; değer belirtilmez:

Güvenlik güçleri, hükümetin <TIMEX3> seçim döneminde </TIMEX3> gösterileri ve büyük toplantıları yasaklamasına rağmen müdahale etmekten kaçındı.

Bu <TIMEX3> pandemi sürecinde </TIMEX3> şubeye gitmek istemiyorum.

- Tarih veya zaman belirten fakat belirttiği an net olmayan ifadeler işaretlenir:

Ancak <TIMEX3 tid="t1" type="TIME" value="XXXX-XX-XXTEV"> o gün akşam </TIMEX3> bankayı arayarak talebimin sistemde olup olmadığını sordum.

<TIMEX3 tid="t2" type="DATE" value="XXXX-WXX"> Hafta içi </TIMEX3> banka <TIMEX3 tid="t3" type="TIME" value="TXX:XX"> açık olduğu saatlerde </TIMEX3> çalışıyorum.

Artık öyle bir duruma geldim ki <TIMEX3 tid="t4" type="DATE" value="XXXX-XX-XX"> aldığım gün </TIMEX3> <TIMEX3 tid="t5" type="DATE" value="XXXX-XX-XX"> en talihsiz günüm </TIMEX3> olarak görüyorum.

Faturalarımı <TIMEX3 tid="t6" type="DATE" value="XXXX-XX-XX"> gününden </TIMEX3> önce öderim.

- Yaşanmış bir olayın aynı zamanda adı olmuş fakat ilgili olayın zamanını da belirten ifadeler işaretlenir:

Ancak <TIMEX3 tid=”t1” type=”DATE” value=“XXXX-09-11”> 11 Eylül </TIMEX3> muhtemelen dünya tarihinin en önemli olaylarından biriydi.

- Küsuratlı olarak belirtilen zaman ifadeleri işaretlenir:

Federal ceza yönergelerini takiben, yargıç ona maksimum <TIMEX3 tid=”t1” type=”DURATION” value="P10Y1M"> 10 yıl 1 ay </TIMEX3> verdi.

<TIMEX3 tid=”t2” type=”DURATION” value="P6.5Y"> Altı buçuk yıl </TIMEX3> hapis yattı.

<TIMEX3 tid=”t3” type=”DURATION” value="P0.5H"> Yarım saat </TIMEX3> sonra dönebilir misiniz dedim.

- Referans tarihi belli olmayan ifadelerde sadece ifadede geçen değer işaretlenir; ilgili formülünde geri kalan kısımlar X ile doldurulur:

<TIMEX3 tid=”t1” type=”TIME” value="XXXX-XX-XXTEV"> Aynı günün akşamında </TIMEX3> teslim edilmiştir.

Kredi kartı borç ödememin <TIMEX3 tid=”t2” type=”DATE” value="XXXX-XX-XX"> son günüydü </TIMEX3>.

<TIMEX3 tid=”t3” type=”DATE” value="XXXX-XX"> 2020 yılının son aylarında </TIMEX3> mont aldım.

<TIMEX3 tid=”t4” type=”TIME” value="XXXX-XX-XXTMO"> Sabah 8’de </TIMEX3> uyandım ve tid=”t5” type=”TIME” value="TXX:XX"> o saatten </TIMEX3> beri ayaktayım

<TIMEX3 tid=”t6” type=”DATE” value="XXXX-12"> Aralık ayını </TIMEX3> seviyorum.

Ancak <TIMEX3 tid=”t7” type=”DATE” value="XXXX-XX-XX"> ay sonunda </TIMEX3> yüksek gelen faturayla öğrendim.

Aboneliğimi iptal ettirdim, <TIMEX3 tid=”t8” type=”DATE” value="XXXX-XX"> kullanmadığım ayın </TIMEX3> ücretini iade alacağım söylendi.

<TIMEX3 tid=”t9” type=”DATE” value="XXXX-XX-XX"> Yaz günü </TIMEX3> kazak giymemi istiyorlar.

2.2.7 İşaretlenmeyen İfade Kuralları

- Sıralama ifadeleri işaretlenmez:

Sonrasında aradığımızda ise takıp numarası veriyoruz denildi.

Daha sonra da birçok mağduriyet yaşadık.

Sonunda ürünü kaybettiklerini kabullendiler.

Markete çıkmıştım, o sırada gelmişler ve hiçbir şekilde arama yapılmadan ürün teslim edilememiştir diye not almışlar.

50 işi aynı anda yapmaya çalışmanız bizim gibi müşterilerin mağdur edersiniz.

Sizle bundan sonra bir daha da alışveriş yapmam.

- Bir şeyin ne kadar erken ve hızlıca olacağını belirten durum zarfları işaretlenmez:

Bu internet altyapısının sokağımıza en kısa zamanda yapılmasını istiyorum.

Bir an önce VDSL altyapının sağlanmasını talep ediyorum.

Hemen iptal edip yeniden almaya çalışınca ne yazık ki paramın hemen yatmadığını unutmuştum.

- Ölçülemeyen süreler işaretlenmez:

Bu süreç içerisinde kurye şirketini defalarca aradım.

Kısa bir süre sonra bonus cezası yansımış.

Bu hattımı uzun zamandır kullanıyorum.

Sonra da sürekli hattınız kapanır diye tehdit ediyor.

- Belirsiz zaman birimleriyle yapılan atıflar işaretlenmez:

Şu zor zamanların da üstesinden geleceğimize eminim.

Çok daha zor dönemleri atlattık bunu da atlatırız.

Telefonumun garanti süresi doldu.

Geçici olarak yurtdışındayım.

- Olumsuzluklar ve var olmayan zamanlara yapılan atıflar işaretlenmez:

Çünkü hiçbir zaman vaktinde teslim yapmadılar.

Bir daha asla alışveriş yapmayı düşünmüyorum.

Geleceğini söyledi ama hala gelmedi.

- Belirli periyot dönemi verilmeyen frekans ifadeleri işaretlenmez:

2 kez mail atmama rağmen dönüş olmuyor.

2 defa çağrı merkezini aradım acil diye.

Her aradığımda bayilerine ulaşamadıklarını bu yüzden iadeyi yapamadıklarını söylüyorlar.

Her seferinde bana düzeltildi denmesine rağmen hiç düzeltilme yapılmadı.

Yine her zamanki gibi size bu platformdan ulaşabileceğimi bildiğimden buraya yazıyorum.

Yaşadığım hız sorunu ve belirli aralıkla yaşanan kopmalar nedeniyle hizmet alamıyorum.

- Yazımsal olarak zaman ifade etse de kullanıldığı isim halinde başka bir şeyi çağrıştıran ifadeler işaretlenmez:

George Orwell'ın 1984 kitabını lisede okumuştum.

Zonguldak Perşembe Hacıkadı şubenizdeki ATM sürekli arızalı.

Dakika hakkım bitince ücretsiz aramalarda faturama ücret yansıtıldı.

Fevzi Çakmak şubesi benim aylığıma kesinti yapmıştır.

- "pandemi" ifadesi sonrası "süreci", "dönemi", "zamanı" gibi dönem anlamı taşıyan bir kelime yoksa ifade işaretlenmez:

Pandemi dolayısıyla eşyalarımıza gösterdiğiniz muamele böyle olmamalı.

- Bir olaya bağlı zaman belirten ifadeler işaretlenmez; tamlama sonunda "gün", "saat" gibi zaman birimi varsa işaretlenir:

Ürünü aldığımdan beri bir veya iki defa kullanmışımdır.

Ürünü <TIMEX3 tid="t1" type="DATE" value="XXXX-XX-XX"> teslim aldığım günden </TIMEX3> bu yana bir veya iki defa kullanmışımdır.

Müşteri temsilcime aboneliği kabul ettiğim andan itibaren ulaşamıyorum.

Müşteri temsilcime aboneliği <TIMEX3 tid=”t2” type=”TIME” value="TXX:XX"> kabul ettiğim dakikadan </TIMEX3> itibaren ulaşamıyorum.

Ödedikten sonra başka herhangi bir borcum kalıp kalmadığını sordum.

Her aradığımda modem ayarlarını güncellemeye kalkıyorlar.

Müşteri hizmetlerini aradığım zaman hiçbir şekilde muhatap alınmıyorum.

3
SİSTEM TASARIMI

Bu çalışma kapsamında çeşitli internet sayfalarında yer alan şikayet metinlerinden veri kümesi oluşturulmuştur. Serbest metin şeklinde yazılan şikayet metinlerinde sıkça yazım hataları olmaktadır. Bu hataları düzeltmek amacıyla veri kümesi üzerinde ön işlemler yapılmaktadır. Daha sonrasında geliştirilen modelin girdileri olacak vektörler oluşturulmaktadır. Bu kısımda tüm bu süreçlerle birlikte modelin tasarımı özetlenmektedir.

3.1 Ön İşlemler

Metinler üzerinde gerçekleştirilen model kullanılmadan önce metinlerdeki bozuk karakterlerin düzeltilmesine ve modele girdi oluşturmaya yönelik ön işlemler gerçekleştirilmiştir.

İlk adım olarak metinlerde yer alan “ ’ ” ve “ “ ” gibi anormal noktalama işaretleri düzeltilmiştir. Birden fazla kez yer alan boşluklar tek boşluk olacak şekilde silinmiştir. İşlemler sırasında bozulan Türkçe karakterler unicodedate kütüphanesi yardımıyla tespit edilip düzeltilmiştir.

Metinler üzerinde yapılan işlemlerde Zemberek-nlp aracında yer alan fonksiyonlar kullanılmıştır [62]. Cümle bazlı çalışma için metinlerde cümleleri ayrıştırıcı olarak TurkishSentenceExtractor fonksiyonu kullanılmıştır.

Kelimelerin yazım yanlışlarını düzeltmek için Zemberek-nlp aracında yer alan TurkishSentenceNormalizer fonksiyonu kullanılmıştır. Ardından tüm harfler küçük harfe dönüştürülmüştür. Rakamlar sıfır ile değiştirilmiştir. Bu işlemler sonucunda kelimelerin normalize edilmiş hali oluşturulmuştur. Tablo 3.1’de örnek bir metin için yapılan dönüşümler gösterilmiştir.

Metinleri tokenize etmek için Zemberek-nlp aracında yer alan TurkishTokenizer fonksiyonu kullanılmıştır. TurkishMorphology fonksiyonu kullanılarak da kelimelerin morfolojik analizleri yapılmıştır.

Tablo 3.1 Örnek bir metin için yapılan düzeltmeler ve değişiklikler

Metin	Magdur	edildim	gereginin	yapılmasını	talep	ediyom	.
Yazım yanlışı düzeltme	Mağdur	edildim	gereğinin	yapılmasını	talep	ediyorum	.
Küçük harfe dönüştürme	mağdur	edildim	gereğinin	yapılmasını	talep	ediyorum	.
Rakamları sıfır ile değiştirme	mağdur	edildim	gereğinin	yapılmasını	talep	ediyorum	.

Modele girdi olacak vektörleri oluşturma sürecinde kullanmak üzere metinlerde yer alan öğelerden kelime, karakter, ortografik karakter ve morfolojik analiz sonucu değer sözlükleri oluşturulmuştur. Bu sözlüklerdeki değerler, metinlerde yer alma sıklıklarına göre sıralandıktan sonra tekil olarak numaralandırılmıştır. Öğeler için oluşturulacak vektörlerde bu sözlükler kullanılmaktadır.

3.2 Model Girdileri

Modelin girdisi, metinlerde yer alan kelime bazlı oluşturulan 5 farklı vektörün birleştirilmesiyle oluşturulan vektörlerdir. Bu vektörlerin oluşturulma süreçleri, bu kısımda özetlenmiştir.

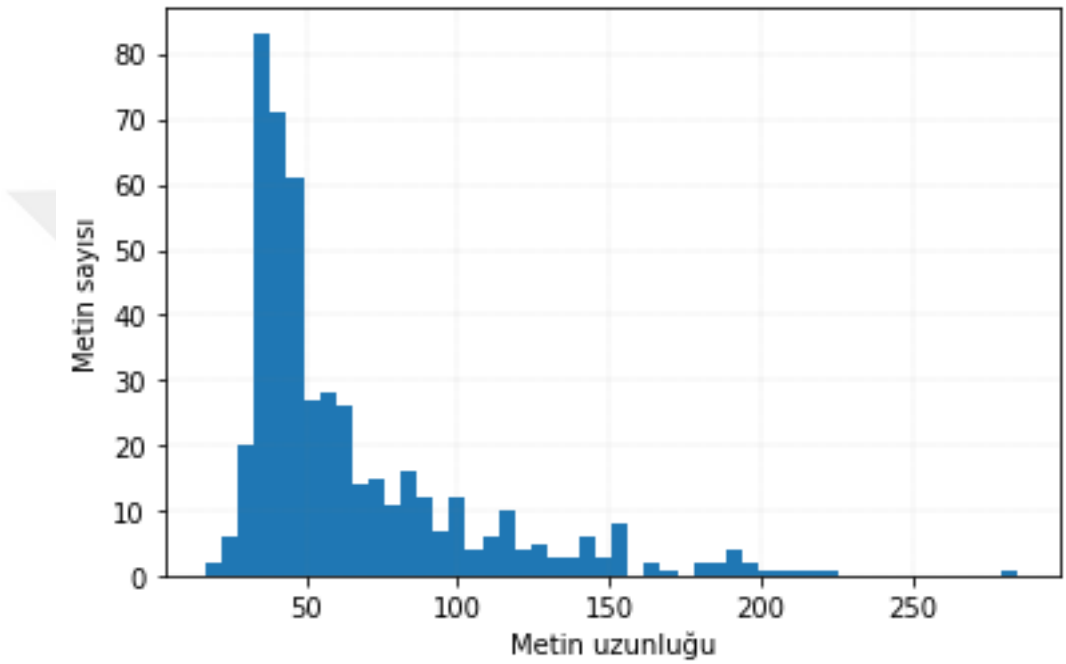
3.2.1 Kelime Vektörleri

Kelime vektörleri, çeşitli bakış açılarıyla oluşturulan kelimeleri temsil eden vektörlerdir. Bu tez çalışmasında, normalize işlemi sonucu elde edilen kelimeler, eğitilmiş ve herkese açık paylaşılan kelime vektörleri sözlüğünde aranarak kelime vektörleri elde edilmiştir. Bulunamayan sözcükler için sözlükte yer alan tüm vektörlerin değerlerinin ortalaması alınarak oluşturulan vektör kullanılmıştır.

Veri kümesi analiz edildiğinde, en uzun şikayet metninin 285 kelimeden oluştuğu görülmüştür. Bu sonuçtan yola çıkarak metinleri oluşturan kelime sayısı 300'e tamamlanmıştır. Metinlerin uzunluklarının dağılımı Şekil 3.1'de gösterilmektedir.

Kelime vektörleri olarak, Türkçe için 3 farklı model ile oluşturulan ve herkese açık vektörler kullanılmıştır. Bunlardan ilki Güngör'ün hedef kelimenin etrafındaki kelimeleri tahmin eden Skip-gram modelini eğiterek oluşturduğu ve paylaştığı vektörlerdir [48]. Skip-gram, hedef kelimenin etrafındaki kelimeleri tahmin ederek kelime vektörlerini oluşturan tahmin temelli bir modeldir [56, 61]. Bu vektörler, değerlendirme sonuçlarında "onurgu" olarak etiketlenmiştir. Bir diğeri, kelimelerin

birlikte yer alma olasılık istatistiklerinden faydalanan GloVe modelini kullanarak oluşturulmuş ve Github'da paylaşılan vektörlerdir [49, 60]. Bu vektörler, değerlendirme sonuçlarında "glove" olarak etiketlenmiştir. Sonuncusu, Skip-gram modelinin bir uzantısı olan fasttext geliştiricilerinin sitelerinde Türkçe için paylaştıkları vektörlerdir [50]. Bu vektörler, değerlendirme sonuçlarında "fasttext" olarak etiketlenmiştir. Tüm bu vektörler, modele girdi oluştururken boyutları 300 olarak alınmıştır.

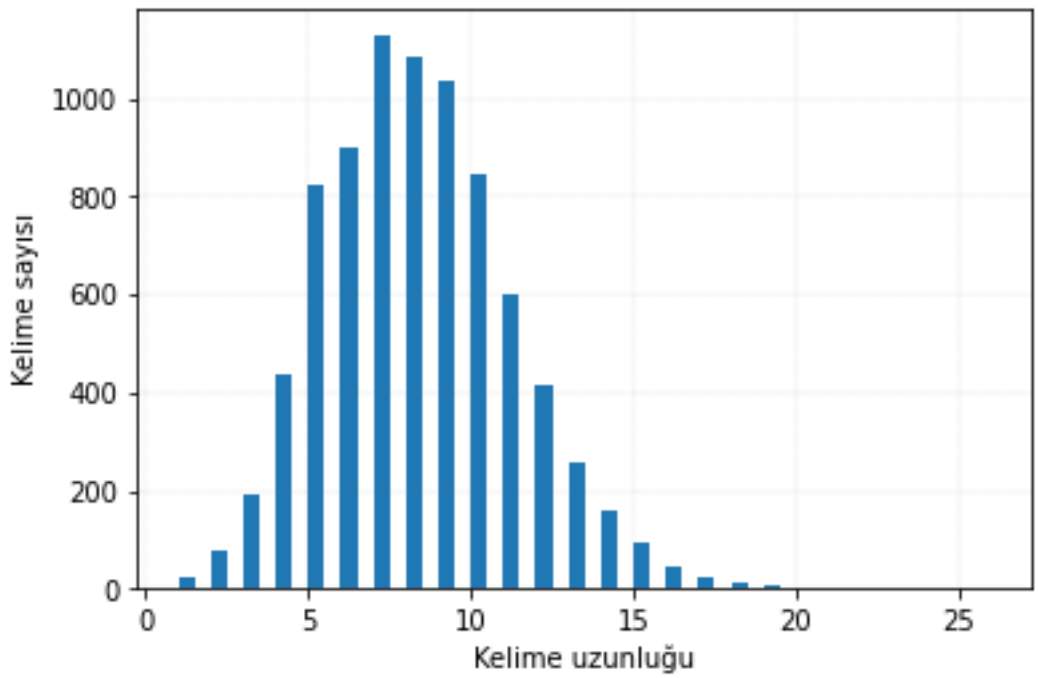


Şekil 3.1 Metin uzunluklarının dağılımı

3.2.2 Karakter Tabanlı Kelime Vektörleri

Metinde bulunduğu şekilde kelimelerde yer alan harf, rakam ve noktalama işaretlerini içeren karakter listeleri oluşturulmuştur. Sonrasında bu listelerde yer alan değerlerin sözlükteki karşılığı olan tekil numaralar kullanılarak karakter tabanlı kelime vektörleri oluşturulmuştur.

Metinlerde yer alan kelimelerin uzunlukları analiz edildiğinde, en uzun kelimenin 26 karakterli olduğu görülmüştür. Bu sonuçtan yola çıkarak kelimeleri oluşturan karakterlerin sayısı 30'a tamamlanmıştır. Kelimelerin uzunluklarının dağılımı Şekil 3.2'de gösterilmektedir.



Şekil 3.2 Kelime uzunluklarının dağılımı

3.2.3 Ortografik Karakter Tabanlı Kelime Vektörleri

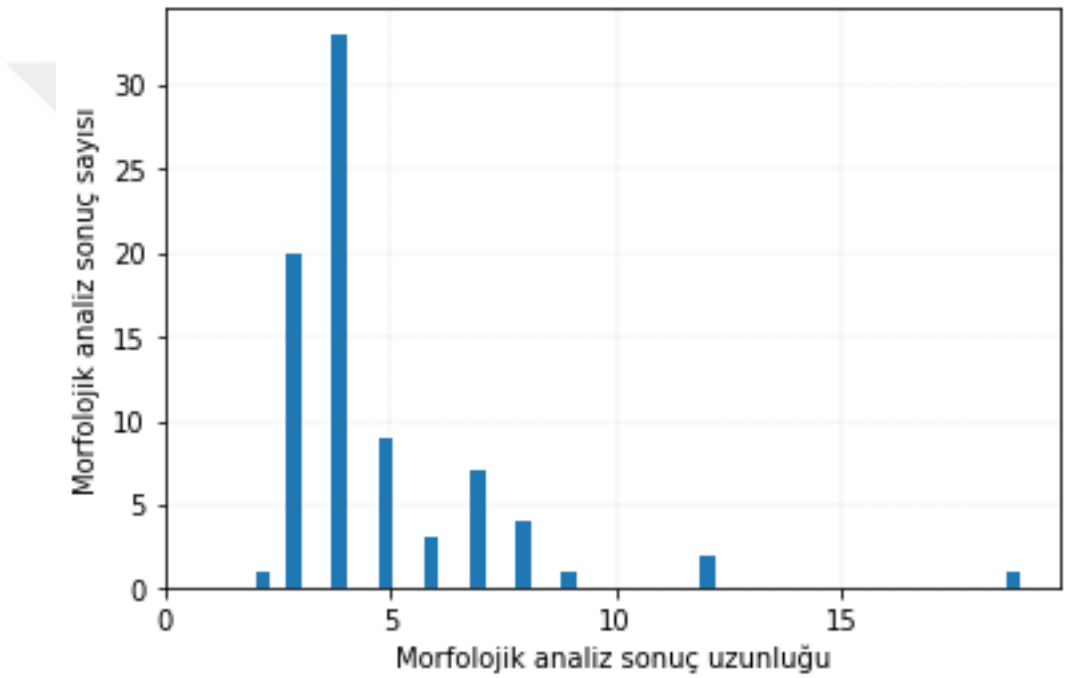
Akkaya'nın geliştirmiş olduğu modele girdi olarak sadece kelime vektörleri kullanıldığında fasttext en başarılı kelime vektörleri olmuştur [36]. Yine aynı çalışmada fasttext kelime vektörleri ile birleştirildiğinde model başarısını en çok artıran, ortografik karakter tabanlı kelime vektörleri olmuştur. Bu sonuçlardan esinlenerek metinlerin ortografik karakter tabanlı kelime vektörleri oluşturulmuştur. Bu amaçla, kelimelerde yer alan büyük harflerin karşılığı "C", küçük harflerin "c", rakamların "n", noktalama işaretlerinin "p", boşlukların "s" olacak şekilde dönüşüm işlemi yapılmıştır. Sonrasında bu değerlerden oluşan kelime bazlı karakter listeleri oluşturulmuştur. Ardından listelerdeki değerlerin ilgili sözlükteki karşılığı olan tekil numaralar kullanılarak ortografik karakter tabanlı kelime vektörleri oluşturulmuştur. Karakter tabanlı kelime vektörleri için yapılan analiz dikkate alınarak ortografik karakter sayısı 30'a tamamlanmıştır.

3.2.4 Morfolojik Analiz Sonucu Tabanlı Kelime Vektörleri

Güngör, morfolojik analizlerin farklı versiyonlarının NER modelinde başarılarını kıyasladığı çalışmasında geliştirdiği biLSTM model üzerinde farklı morfolojik analiz sonucu tabanlı kelime vektörlerinin başarılarını değerlendirmiştir [51]. Paylaştığı sonuçlara göre model en yüksek başarıyı kelime kökünü içermeyen morfolojik

analiz sonucu tabanlı kelime vektörleriyle elde etmiştir. Bu sonuçtan esinlenerek, kelime bazlı olarak morfolojik analiz sonuçları ayrıştırılarak liste haline getirilmiş ve bu listelerden kelime kökleri silinmiştir. Sonrasında bu listelerde yer alan değerlerinin sözlükteki karşılığı olan tekil numaralar kullanılarak morfolojik analiz sonucu tabanlı kelime vektörleri oluşturulmuştur.

Metinlerde yer alan kelimelerin morfolojik analiz sonuçları analiz edildiğinde, en uzun morfolojik analiz sonucunun 19 öğe içerdiği görülmüştür. Bu sonuçtan yola çıkarak her bir kelime için morfolojik analiz sonuçlarını oluşturan öğe sayısı 20'ye tamamlanmıştır. Sonuç öğelerinin sayılarının dağılımı Şekil 3.3'te gösterilmektedir.



Şekil 3.3 Morfolojik analiz sonuç uzunluklarının dağılımı

3.2.5 Yazınsal Öznitelik Vektörleri

Güneş, NER kapsamında biLSTM ve sınıflandırıcı olarak Softmax kullanarak geliştirdiği 7 model üzerinde girdi olarak kelime vektörlerine yazınsal özelliklerini eklediğinde başarısının artırdığını gözlemlemiştir [52]. Şeker, NER kapsamında CRF tabanlı geliştirdiği sistemde zaman ve numara ifadeleri için ekstra öznitelikler çıkarmıştır: sayıların bulunduğu aralığın kategori değeri, yüzde işareti veya "yüzde" ifadesi olması, kelimenin "saat" ifadesi olması, iki nokta üst üste işaretini içermesi [35]. Bu özniteliklerden sayıların bulunduğu aralığın kategori değerini çıkardığında modelin tarih ifadelerini tespit başarısı en fazla düşüşü göstermiştir. Saat

ifadelerinde ise iki nokta üst üste işaretini içermesi özniteliğini çıkardığında modelin başarısındaki en yüksek düşüş gözlenmiştir. Bu sonuçlardan esinlenerek kelimelerin 15 farklı yazınsal özniteliği belirlenmiştir. Öznitelikler ve alabileceği değerler aşağıda listelenmiştir:

- Tüm karakterler harf; 0 veya 1
- Tüm karakterler rakam; 0 veya 1
- Tüm karakterler noktalama işareti; 0 veya 1
- Tüm karakterler harf ve rakamlardan oluşuyor; 0 veya 1
- Tüm karakterler harf ve noktalama işaretlerinden oluşuyor; 0 veya 1
- Tüm karakterler rakam ve noktalama işaretlerinden oluşuyor; 0 veya 1
- Tüm karakterler harf, rakam ve noktalama işaretlerinden oluşuyor; 0 veya 1
- İlk karakter büyük harf; 0 veya 1
- Tüm karakterler büyük harf; 0 veya 1
- Tüm karakterler küçük harf; 0 veya 1
- Sayının bulunduğu aralığın kategori değeri: 1-12 arası ise 1, 13-31 arası ise 2, 32-60 arası ise 3, 1990-2020 arası ise 4, diğerleri ise 5
- Saniye, dakika, saat, gün, hafta, ay, yıl ve yüzyıl ifadelerinden birini içermesi; 0 veya 1
- İki nokta üst üste noktalama işareti içermesi; 0 veya 1
- Tre noktalama işareti içermesi; 0 veya 1
- Nokta noktalama işareti içermesi; 0 veya 1

Kelimelerin listelenen öznitelikleri çıkarılarak alabildiği değerler üzerinden kodlaması yapılmış ve bu kodlar üzerinden 15 uzunluğunda yazınsal öznitelik vektörleri oluşturulmuştur.

3.2.6 BERT Kelime Öznitelikleri

Devlin ve arkadaşları, Transformer adı verilen dönüştürücü model mimarisini kullanarak BERT adını verdikleri dil modelini geliştirdiler [57]. BERT, yayınlandığı döneme kadar olan modellerden farklı olarak cümleyi sağdan sola ve soldan sağa değerlendirerek kelimelerin metin içerisinde aralarındaki ilişkiyi daha iyi öğrenebilmektedir. Model, BookCorpus ve İngilizce Wikipedia verisi kullanılarak maskeli dil modeli ve sonraki cümleyi tahmin adı verilen iki teknikle eğitilmiştir.

Makalede doğal dil işleme alanındaki 11 farklı konuda model sonuç değerleri paylaşılmıştır. Ayrıca modelinin çıktısı olan 13 katman değerleri üzerinden kelimeleri temsilen oluşturulacak vektörleri kullanmanın, modeli ince ayar yaparak kullanmaya göre avantajlı olduğundan bahsedilmektedir. Bu amaçla tez kapsamında daha önceden eğitilmiş BERTurk (uncased 128k) modelini kullanarak metinler tokenize edilmiş ve model çıktısı oluşturulmuştur [65]. Sonrasında metinlerdeki her bir kelime için modelin çıktıları üzerinden 5 farklı kelime vektörü oluşturulmuştur: ilk katman değerleri, son katman değerleri, sondan ikinci katman değerleri, son 12 katman değerlerinin toplamı ve son 4 katman değerlerinin toplamı. Bu vektörler oluşturulurken, tokenize işlemi sonucu bir kelime için birden fazla token oluştuğu durumlarda ilk token değerleri dikkate alınmıştır.

3.3 Yöntemler

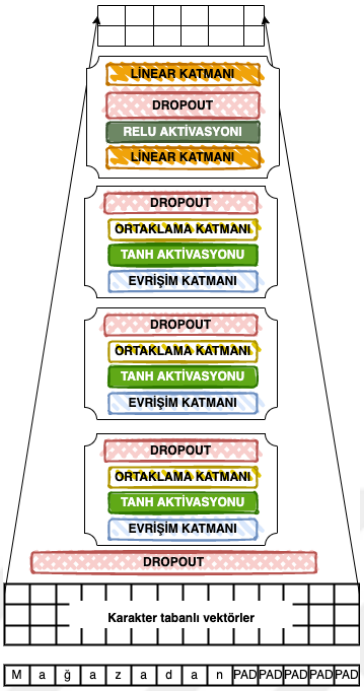
Tez kapsamında karakter ve ortografik karakter tabanlı kelime vektörlerini oluşturmak için karakter tabanlı CNN modeli geliştirilmiştir. Morfolojik analiz sonucu tabanlı kelime vektörlerini oluşturmak için öğe tabanlı biLSTM modeli geliştirilmiştir. Zaman ifadelerini tespit eden model olarak sınıflandırıcı katmanı CRF olacak şekilde biLSTM kullanılmıştır. Bu kısımda, kullanılan bu modellerin detayları özetlenmektedir.

3.3.1 Karakter Tabanlı CNN Modeli

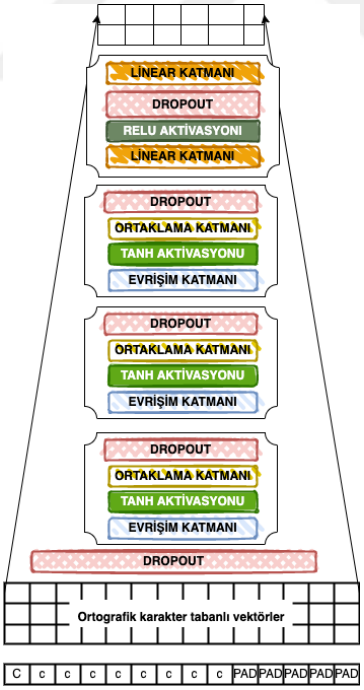
CNN, denetimli öğrenme amacıyla kullanılan çok katmanlı spesifik bir yapay sinir ağı türüdür [53]. Evrişim, aktivasyon, ortaklama ve tam bağlantı olmak üzere 3 katmandan oluşur.

Metinde yer aldığı şekilde kelimelerin içerdiği karakterlerden oluşturulan listeler, CNN modelinde girdi olarak kullanılmıştır. Modele girdi olacak vektörler oluşturulurken Türkçe NER alanında yapılan çalışmalarda kullanılan hiper parametre değerleri dikkate alınarak vektör boyutu 50, gizli katman sayısı 200 olarak alınmıştır. Modelin evrişim ve linear katmanlarının ağırlık parametrelerinin ilk değerlerini belirlemek için, Glorot'un önerdiği tekdüze yöntemi kullanılmıştır [43]. Modeldeki önyargı değerleri, sabit olarak sıfıra eşitlenmiştir. Model çıktısı, zaman ifadelerini tespit eden modele karakter tabanlı kelime vektörleri olarak girdi

oluşturmaktadır. Geliştirilen modelin mimarisi, kelime karakter listesi ve ortografik karakter listesi için sırasıyla Şekil 3.4 ve Şekil 3.5'te yer almaktadır.



Şekil 3.4 Karakter listeleri için CNN modeli



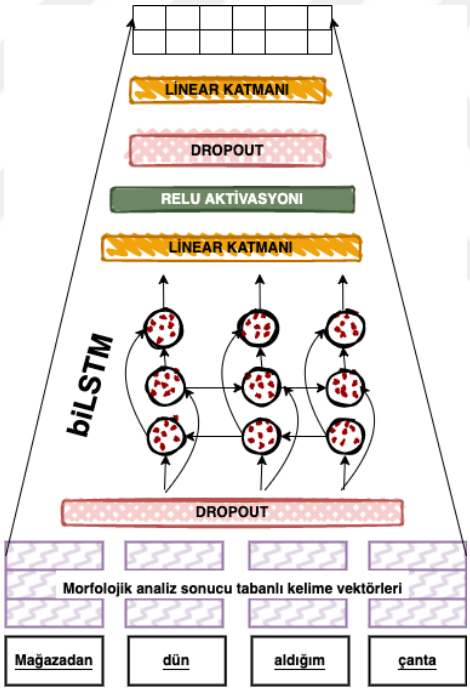
Şekil 3.5 Ortografik karakter listeleri için CNN modeli

3.3.2 Öğe Tabanlı biLSTM Modeli

Yinelemeli Sinir Ağları (RNN), bir dizin üzerinde sırasıyla ilerlerken girdisine ek olarak bir önceki adımın çıktısını da girdi olarak kullanan bir yapay sinir ağ sınıfıdır.

Uzun Kısa Süreli Bellek (LSTM), RNN'in bir versiyonu olmakla birlikte farklı olarak tek nöron katmanı yerine kapı olarak da ifade edilen 4 katmandan oluşan yapıya sahiptir [54]. biLSTM ise, dizinde çift taraflı bilgi çıkarımını sağlamak amacıyla ileri ve geri olarak her iki yönlü ilerleyen iki adet LSTM yapısından oluşmaktadır.

Morfolojik analiz sonuçlarının ayrıştırılmasıyla oluşturulan listeler, geliştirilen öğe tabanlı biLSTM modelinde girdi olarak kullanılmıştır. Modelin biLSTM kısmındaki ağırlık parametrelerinin ilk değerleri için ortogonal matris yöntemi kullanılmıştır. Linear katmanlarında ise ağırlık parametrelerinin ilk değerleri için Glorot'un önerdiği tekdüze yöntemi kullanılmıştır. Modeldeki önyargı değerleri, sabit olarak sıfıra eşitlenmiştir. Model çıktısı, zaman ifadelerini tespit eden modele morfolojik analiz sonucu tabanlı kelime vektörleri olarak girdi oluşturmaktadır. Geliştirilen modelin mimarisi Şekil 3.6'da yer almaktadır.



Şekil 3.6 Morfolojik analiz sonuçları listeleri için biLSTM modeli

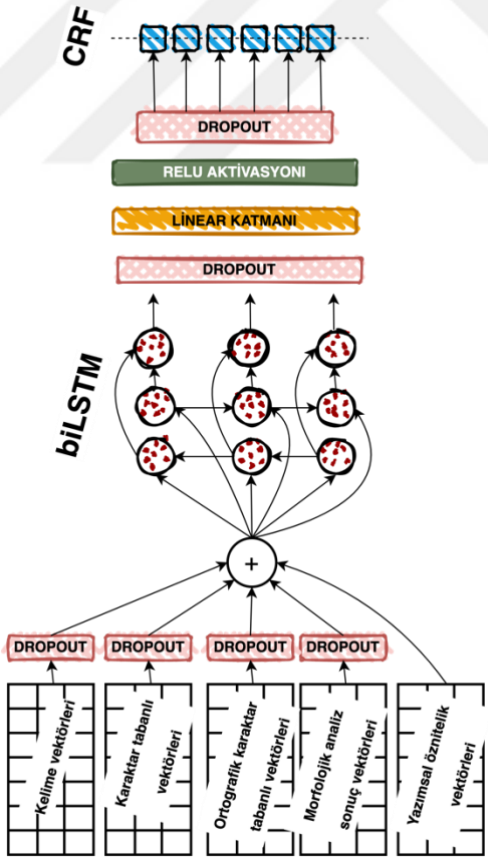
3.3.3 biLSTM ve CRF ile Zaman İfadelerini Tespit Modeli

CRF, sıralı girdi üzerinde her birime etiket atayarak en yüksek olasılıklı etiket dizilimini belirlemek için kullanılan bir olasılık modelidir [55]. Modelde, bir birimden diğerine geçişin etiket sıralaması olarak olasılığı ile birimin girdilerinden elde edilecek etiketlerin olasılıkları hesaplanır. Bunlar üzerinden dizilimin her

birimine atanacak etiketler sonucu elde edilecek etiket dizilimi için en yüksek olasılıkla olanı tespit edilir.

Zaman ifadelerini otomatik tespit edebilmek amacıyla tez kapsamında oluşturulan modelde, oluşturulan kelime vektörler biLSTM yapısına girdi oluşturur. Ardından Linear ve ReLu aktivasyon katmnaları yer almaktadır. Modelin son ve sınıflandırıcı katmanı olarak CRF ile her bir kelimenin etiketlemesi yapılır. Geliştirilen modelin mimarisi Şekil 3.7'de yer almaktadır.

Modelin biLSTM kısmındaki ağırlık parametrelerinin ilk değerleri için ortogonal matris yöntemi kullanılmıştır. Linear katmanlarında ise ağırlık parametrelerinin ilk değerleri için Glorot'un önerdiği tekdüze yöntemi kullanılmıştır. Modeldeki önyargı değerleri, Jozefowicz'in çalışması esas alınarak unut kapısı 1 ve diğerleri sıfıra eşitlenmiştir [63]. Modelin kodlama aşamasında Huang'ın paylamış olduğu çalışmasının kodlarından faydalanılmıştır [64].



Şekil 3.7 Zaman ifadelerini tespit için biLSTM + CRF modeli

Modelin eğitiminde öğrenme katsayısı 0,015 ve öğrenme katsayı değişimi 0,05 olarak alınmıştır. Vektörlerin birleştirilmesi sonrası uygulanan dropout katmanında oran 0.33, diğer dropout katmanlarında ise 0,5 olarak alınmıştır. Eğitim için 100 koşu yapılmıştır. Veri kümesi, 10 boyutunda mini-batch halinde kullanılmıştır. Etiketlenen veri kümesinin %80'i eğitim sırasında kullanmak için ayrılmış, geri kalan %20'si ise modelin başarısını ölçerken kullanılmıştır. Eğitimde kullanılan veri, çalışmanın başka hiçbir aşamasında kullanılmamıştır.

4
DENEYSEL SONUÇLAR

Geliştirilen model, oluşturulan girdilerin çeşitli kombinasyonlarıyla çalıştırılarak 3 farklı başarı skoru üzerinden değerlendirilmiştir.

Tez çalışması boyunca kullanılacak platform olarak PyTorch tercih edilmiştir.

Yapılan testlerin doğruluk, kesinlik, hassasiyet ve F1 skor değerleri hesaplanmıştır. Başarı sıralama ve değerlendirmesinde F1 skor değeri dikkate alınmıştır. Doğruluk, modelin doğru tahmin ettiklerinin veri kümesine oranıdır. Kesinlik, modelin pozitif olarak tahmin ettikleri içerisinde gerçekten pozitif olanların oranıdır. Hassasiyet, gerçekten pozitif olanlar içerisinde modelin pozitif olarak tahmin ettiklerinin oranıdır. Gerçek değerler ile modelin tahmin değerlerinin yer aldığı karışıklık matrisinde yer alan değer ifadeleri Tablo 4.1'de paylaşılmıştır:

Tablo 4.1 Karışıklık matrisi

		TAHMİN EDİLEN DEĞER	
		Etiketlenenler	**Etiketlenmeyenler**
GERÇEK DEĞER	**Etiketlenenler**	*GERÇEK POZİTİF*	*YANLIŞ NEGATİF*
	Etiketlenmeyenler	*YANLIŞ POZİTİF*	*GERÇEK NEGATİF*

Karışıklık matrisinde yer alan değerler üzerinden kullanılacak başarı değerlerinin bağıntıları Eşitlik 2.1-2.4'deki gibi olmalıdır.

$$Doğruluk = \frac{Gerçek\ Pozitif + Gerçek\ Negatif}{Gerçek\ Pozitif + Yanlış\ Pozitif + Gerçek\ Negatif + Yanlış\ Negatif} \quad (2.1)$$

$$Kesinlik = \frac{Gerçek\ Pozitif}{Gerçek\ Pozitif + Yanlış\ Pozitif} \quad (2.2)$$

$$Hassasiyet = \frac{Gerçek\ Pozitif}{Gerçek\ Pozitif + Yanlış\ Negatif} \quad (2.3)$$

$$F1\ skor = 2\ x\ \frac{Kesinlik\ x\ Hassasiyet}{Kesinlik + Hassasiyet} \quad (2.4)$$

Deneysel sonuçlar, modelin başarısını ölçmek için ayrılan veri kümesinde kelime bazlı olarak (2.1), (2.2), (2.3) ve (2.4) denklemlerine göre değerler ölçülerek paylaşılmıştır. Ayrıca Reimers ve Gurevych'ın önerisi dikkate alınarak başarı

değerleri, yapılan koşulların ortalaması ve standart sapmaları üzerinden raporlanmıştır.

4.1 Kelime Vektörleri

Geliştirilen modele girdi olarak sadece kelime vektörlerini alarak 3 farklı kelime vektörünün başarısı ölçülmüştür. Ayrıca, önceden eğitilmiş kelime vektörlerini kullanmanın etkisini de göstermek için rastgele değerler ile oluşturulan kelime vektörlerinin başarısı ölçülmüştür. Rastgele kelime vektörlerinde ilk değerler olarak He'nin önerdiği tekdüze yöntemi kullanılmıştır. Vektör bazlı başarı değerleri Tablo 4.2'de listelenmektedir.

Tablo 4.2 Kelime vektörlerinin başarı değerleri

Kullanılan Kelime Vektörleri	**Doğruluk**	**Kesinlik**	**Hassasiyet**	**F1 Skor**	**Rastgele Vektörlere Göre Katkısı**
rastgele	%93,00 ± 0,00	%87,00 ± 0,00	%93,00 ± 0,00	%90,00 ± 0,00	
onurgu	%95,21 ± 0,01	%93,47 ± 0,02	%95,21 ± 0,01	%93,88 ± 0,01	+3,88
glove	%95,63 ± 0,01	%94,10 ± 0,01	%95,63 ± 0,01	%94,51 ± 0,01	+4,51
fasttext	%94,28 ± 0,01	%91,47 ± 0,02	%94,28 ± 0,01	%92,27 ± 0,01	+2,27

Tablo 4.2'deki değerler incelendiğinde kelime vektörlerinin başarıları kıyaslandığında en başarılı kelime vektörlerinin "glove" olduğu gözlemlenmektedir. Ayrıca, en az başarılı olan "fasttext" kelime vektörleri bile, rastgele oluşturulan kelime vektörlerinden daha başarılı olduğu görülmektedir. Bu sonuç ile, önceden eğitilmiş kelime vektörlerinin model başarısı üzerindeki etkisi gösterilmiştir.

Kelime vektörlerinin başarısı üzerinde veri kümesinde bulunan kelimelerin vektörler içerisinde bulunmasının etkisini araştırmak için bulunan ve bulunmayan veri kümesi kelime sayıları ile veri kümesinde kapsama yüzdesi Tablo 4.3'te listelenmektedir. Tabloya son sütun olarak kullanıldıkları modelin F1 skor değerleri eklenmiştir.

Tablo 4.3 Kelime vektörlerinde bulunma oranları

Kullanılan Kelime Vektörleri	**Bulunan Kelime Sayısı**	**Toplam Bulunan Kelime Oranı**	**Bulunan Kelimelerin Metinleri Kapsama Oranı**	**Etiketlenen Kelimelerin Bulunma Oranı**	**F1 Skor**
onurgu	7.359	%90,12	%95,80	%89,76	%93,88 ± 0,01
glove	7.078	%86,68	%95,53	%87,71	%94,51 ± 0,01
fasttext	7.739	%94,77	%98,37	%92,15	%92,27 ± 0,01

Tablo 4.3'teki değerler incelendiğinde, bulunma ve kapsama yüzdesi en düşük olan "glove" kelime vektörleri, model başarısında en üst sırada yer almaktadır. Kapsama yüzdesi en yüksek olan "fasttext" kelime vektörleri ise, 3 model arasında en başarısızı olmuştur. Bu araştırmayla, kelime vektörlerini oluştururken kullanılan yöntemin model başarısı üzerinde etkili olabileceği çıkarımına ulaşılmaktadır.

Kelime vektörlerinin model çıktılarının gerçek ve tahmin etiket sayılarının karışıklık matrisleri token bazında Tablo 4.4, Tablo 4.5 ve Tablo 4.6'da gösterilmektedir. Tablolardaki karışıklık matrisleri incelendiğinde, tüm vektörlerin saat ve tekrar ifadelerinde başarısız olduğu görülmektedir. "fasttext" kelime vektörlerinin kullanıldığı model, ifadeleri zaman ifadesi olmadığı şeklinde etiketlemeye meyilli olduğu gözlemlenmektedir. "glove" vektörleri, tarih ifadelerini, "onurgu" kelime vektörleri ise süre ifadelerini belirlemede daha başarılı gözükmektedir.

Tablo 4.4 onurgu kelime vektörlerinin etiket bazlı karışıklık matrisi

		TAHMİN EDİLEN DEĞER				
		<unk>	*TARİH*	*SAAT*	*SÜRE*	*TEKRAR*
GERÇEK DEĞER	*<unk>*	5.932	2	0	12	0
	TARİH	108	65	0	25	0
	SAAT	16	13	0	9	0
	SÜRE	13	2	0	135	0
	TEKRAR	25	0	0	19	0

Tablo 4.5 glove kelime vektörlerinin etiket bazlı karışıklık matrisi

		TAHMİN EDİLEN DEĞER				
		<unk>	*TARİH*	*SAAT*	*SÜRE*	*TEKRAR*
GERÇEK DEĞER	*<unk>*	5.934	4	0	8	0
	TARİH	115	73	0	10	0
	SAAT	20	11	0	7	0
	SÜRE	30	0	0	120	0
	TEKRAR	35	0	0	9	0

Tablo 4.6 fasttext kelime vektörlerinin etiket bazlı karışıklık matrisi

		TAHMİN EDİLEN DEĞER				
		<unk>	*TARİH*	*SAAT*	*SÜRE*	*TEKRAR*
GERÇEK DEĞER	*<unk>*	5.935	0	0	11	0
	TARİH	154	0	0	44	0
	SAAT	22	0	0	16	0
	SÜRE	51	0	0	99	0
	TEKRAR	35	0	0	9	0

4.2 İkili Vektörler

Kelime vektörlerine ilaveten geliştirilen karakter tabanlı CNN modeli ile oluşturulan karakter tabanlı kelime vektörleri (char) ve ortografik karakter tabanlı kelime vektörleri (orth), geliştirilen öğe tabanlı biLSTM modeli ile oluşturulan morfolojik analiz sonucu tabanlı kelime vektörleri (morph) ile yazınsal öznitelik vektörleri (fea) eklendiğinde modelin başarı değerleri Tablo 4.7'de listelenmektedir.

Tablo 4.7 Kelime vektörleri ve diğer vektörleri ile başarı değerleri

Kullanılan Vektörler	Doğruluk	Kesinlik	Hassasiyet	F1 Skor	Katkısı
onurgu + char	%95,24 ± 0,01	%93,37 ± 0,02	%95,24 ± 0,01	%93,94 ± 0,01	+0,06
onurgu + orth	%95,42 ± 0,01	%93,64 ± 0,02	%95,42 ± 0,01	%94,19 ± 0,01	+0,31
onurgu + morph	%95,25 ± 0,01	%93,45 ± 0,02	%95,25 ± 0,01	%93,93 ± 0,01	+0,05
onurgu + fea	%95,42 ± 0,01	%93,82 ± 0,02	%95,42 ± 0,01	%94,23 ± 0,01	+0,35
glove + char	%95,66 ± 0,01	%94,24 ± 0,01	%95,66 ± 0,01	%94,65 ± 0,01	+0,14
glove + orth	%95,73 ± 0,01	%94,30 ± 0,01	%95,73 ± 0,01	%94,63 ± 0,01	+0,12
glove + morph	%95,78 ± 0,01	%94,16 ± 0,01	%95,78 ± 0,01	%94,69 ± 0,01	+0,18
glove + fea	%96,05 ± 0,01	%94,45 ± 0,01	%96,05 ± 0,01	%94,95 ± 0,01	+0,44
fasttext + char	%94,42 ± 0,01	%91,85 ± 0,02	%94,42 ± 0,01	%92,50 ± 0,01	+0,23
fasttext + orth	%94,42 ± 0,01	%91,86 ± 0,01	%94,42 ± 0,01	%92,43 ± 0,01	+0,16
fasttext + morph	%94,35 ± 0,01	%91,64 ± 0,01	%94,35 ± 0,01	%92,37 ± 0,01	+0,10
fasttext + fea	%94,74 ± 0,01	%92,09 ± 0,01	%94,74 ± 0,01	%92,96 ± 0,01	+0,69

Tablo 4.7'deki değerler incelendiğinde modelin kelime vektörleri ile oluşturulan herhangi bir başka vektörü beraber kullanıldığında modelin başarısının arttığı

gözlemlenmektedir. 3.2.5'te detaylıca anlatılan yazınsal öznitelik vektörlerini kullanmanın, hangi kelime vektörlerinin kullanıldığından bağımsız olarak model başarısını en çok artıran olduğu görülmüştür. 3.2.2, 3.2.3 ve 3.2.4 başlıklarında detaylıca anlatılan karakter, ortografik karakter, morfolojik analiz sonucu tabanlı kelime vektörlerinin önceden eğitilmiş kelime vektörlerine katkısı, kullanılan kelime vektörüne göre değişiklik göstermektedir. "onurgu" kelime vektörleri için en çok başarıya etki eden bir diğer vektör ortografik karakter tabanlı kelime vektörleri olurken, "glove" kelime vektörleri için morfolojik analiz sonucu tabanlı kelime vektörleri, "fasttext" kelime vektörleri için ise karakter tabanlı kelime vektörleri olmuştur.

4.3 Morfolojik Analiz Sonucu Tabanlı Kelime Vektörleri

Bir kelime, cümledeki konumuna göre farklı morfolojik analiz sonucuna sahip olabilmektedir. Kendinden önce ve sonra gelen kelimelere göre bu sonuçlardan en doğrusu seçilmesi gerekmektedir. Bu amaçla, tez kapsamında morfolojik analiz sonuçlarından en doğru olanını seçmek için Zemberek-nlp aracında yer alan bestAnalysis fonksiyonu kullanılmaktadır. Bu fonksiyonun çıktı değişkenliğini göstermek amacıyla paragraf halinde bulunan veri kümesi, Zemberek-nlp aracında yer alan TurkishSentenceExtractor fonksiyonu kullanılarak cümlelere bölünmüştür. Ardından tez yöntemleri takip edilerek "glove" kelime vektörleri ile 3.2.4'te detaylıca anlatılan morfolojik analiz sonucu tabanlı kelime vektörleri birlikte kullanılarak elde edilen model başarısı Tablo 4.8'de gösterilmektedir.

Tablo 4.8 Morfolojik analiz girdisine göre başarı değerleri

Kullanım Şekli	Doğruluk	Kesinlik	Hassasiyet	F1 Skor
Paragraf bazlı	%95,78 ± 0,01	%94,16 ± 0,01	%95,78 ± 0,01	%94,69 ± 0,01
Cümle bazlı	%94,00 ± 0,00	%88,00 ± 0,00	%94,00 ± 0,00	%91,00 ± 0,00

Tablo 4.8'deki F1 skor değerleri incelendiğinde, metinleri cümlelere ayrıştırdıktan sonra elde edilen model başarısı, metin halinde elde edilen model başarısından düşük olduğu gözlemlenmiştir. Her iki kullanım şekli arasında, metinlerin cümlelere ayrıştırılması dışında hiçbir değişiklik yapılmamıştır. Bu durumda modelin başarısındaki düşüşün sebebi olarak, metinlerdeki cümleleri ayrıştırmadaki hatalar

ve bu hatalardan dolayı cümlelerin morfolojik analiz başarısının düşmesi olduğu çıkarımı yapılmıştır.

4.4 Yazınsal Öznitelik Vektörleri

En başarılı olan "glove" kelime vektörlerini kullanarak, belirlenen 15 farklı yazınsal özniteliğin her birinin model başarısına katkısını gözlemlemek için her seferinde bir özniteliği çıkarılarak model başarısına etkisi ölçülmüştür. Her bir öznitelik eksiltilerek 15 farklı model koşulmuştur. Bu modellerin başarı değerleri Tablo 4.9'da gösterilmektedir. Tabloda, aşağıda listelenen öznitelik liste değerleri kullanılmıştır:

- Tüm karakterler harf; ***(1)***
- Tüm karakterler rakam; ***(2)***
- Tüm karakterler noktalama işareti; ***(3)***
- Tüm karakterler harf ve rakamlardan oluşuyor; ***(4)***
- Tüm karakterler harf ve noktalama işaretlerinden oluşuyor; ***(5)***
- Tüm karakterler rakam ve noktalama işaretlerinden oluşuyor; ***(6)***
- Tüm karakterler harf, rakam ve noktalama işaretlerinden oluşuyor; ***(7)***
- İlk karakter büyük harf; ***(8)***
- Tüm karakterler büyük harf; ***(9)***
- Tüm karakterler küçük harf; ***(10)***
- Sayının bulunduğu aralığın kategori değeri: ***(11)***
- Saniye, dakika, saat, gün, hafta, ay, yıl ve yüzyıl ifadelerinden birini içermesi; ***(12)***
- İki nokta üst üste noktalama işareti içermesi; ***(13)***
- Tre noktalama işareti içermesi; ***(14)***
- Nokta noktalama işareti içermesi; ***(15)***

Tablo 4.9 Yazınsal özniteliklerin model başarısına etkileri

Kullanılan Vektörler	Doğruluk	Kesinlik	Hassasiyet	F1 Skor	Katkısı
glove + fea	%96,05 ± 0,01	%94,45 ± 0,01	%96,05 ± 0,01	%94,95 ± 0,01	
- (1)	%95,91 ± 0,01	%94,36 ± 0,01	%95,91 ± 0,01	%94,84 ± 0,01	-0,11
- (2)	%96,15 ± 0,01	%95,02 ± 0,01	%96,15 ± 0,01	%95,15 ± 0,01	+0,20
- (3)	%96,54 ± 0,01	%95,64 ± 0,01	%96,54 ± 0,01	%95,71 ± 0,01	+0,76
- (4)	%95,95 ± 0,01	%94,41 ± 0,01	%95,95 ± 0,01	%94,91 ± 0,01	-0,04
- (5)	%95,85 ± 0,01	%94,33 ± 0,01	%95,85 ± 0,01	%94,83 ± 0,01	-0,12
- (6)	%95,77 ± 0,01	%94,66 ± 0,01	%95,77 ± 0,01	%94,99 ± 0,01	+0,04
- (7)	%95,98 ± 0,01	%94,43 ± 0,01	%95,98 ± 0,01	%94,89 ± 0,01	-0,06
-(8)	%95,99 ± 0,01	%94,49 ± 0,01	%95,99 ± 0,01	%94,93 ± 0,01	-0,02
-(9)	%95,70 ± 0,01	%94,59 ± 0,01	%95,70 ± 0,01	%94,93 ± 0,01	-0,02
-(10)	%95,91 ± 0,01	%94,41 ± 0,01	%95,91 ± 0,01	%94,94 ± 0,01	-0,01
-(11)	%95,76 ± 0,01	%94,61 ± 0,01	%95,76 ± 0,01	%94,92 ± 0,01	-0,03
-(12)	%96,15 ± 0,01	%95,05 ± 0,01	%96,15 ± 0,01	%95,28 ± 0,01	+0,33
-(13)	%95,85 ± 0,01	%94,36 ± 0,01	%95,85 ± 0,01	%94,76 ± 0,01	-0,19
-(14)	%95,94 ± 0,01	%94,36 ± 0,01	%95,94 ± 0,01	%94,84 ± 0,01	-0,11
-(15)	%95,85 ± 0,01	%94,32 ± 0,01	%95,85 ± 0,01	%94,78 ± 0,01	-0,17

Tablo 4.9'daki F1 skor değerleri incelendiğinde, model başarısının en çok iki nokta üst üste noktalama işareti içermesi özniteliği olmadığında düştüğü gözlemlenmiştir; model başarı değerinde 0,19 bir düşüş kaydedilmiştir. Bir sonraki en çok etki eden öznitelik, nokta noktalama işareti içermesi olmuştur; model başarı değerinde 0,17 bir düşüş kaydedilmiştir. Tablodaki değerler incelendiğinde, tüm karakterlerin

noktalama işareti olduğunu belirten ile zaman birimleri ifadelerinden birisini içermesini belirten özniteliklerinin yokluğunda model başarısının %94,95'ten sırasıyla %95,71 ve %95,28 değerlerine çıktığı gözlemlenmiştir; sırasıyla 0,76 ve 0,33 bir artış kaydedilmiştir. Bu sonuç ile, bu öznitelikleri kullanmanın modelde bazı değerleri ezberlemeye sebep olduğundan modeli yanılttığı değerlendirmesi yapılmıştır.

4.5 Glove Kelime Vektörlerinin Diğer Vektörlerle Birlikte Kullanılması

Tek başına ve ortografik karakter tabanlı kelime vektörleriyle modelin en başarılı sonuçlara ulaştığı "glove" kelime vektörüne diğer vektörlerin çeşitli kombinasyonlar ile eklenmesiyle elde edilen model başarı değerleri Tablo 4.10'da listelenmektedir.

Tablo 4.10 "glove" kelime vektörleri ile diğer vektörlerin birlikte başarı değerleri

Kullanılan Vektörler	**Doğruluk**	**Kesinlik**	**Hassasiyet**	**F1 Skor**	**"glove" Kelime Vektörlere Göre Katkısı**
glove + char + orth	%95,68 ± 0,01	%94,24 ± 0,01	%95,68 ± 0,01	%94,64 ± 0,01	+0,13
glove + char + morph	%95,81 ± 0,01	%94,31 ± 0,01	%95,81 ± 0,01	%94,89 ± 0,01	+0,38
glove + char + fea	%95,87 ± 0,01	%94,41 ± 0,01	%95,87 ± 0,01	%94,89 ± 0,01	+0,38
glove + orth + morph	%95,66 ± 0,01	%94,14 ± 0,01	%95,66 ± 0,01	%94,54 ± 0,01	+0,03

Kullanılan Vektörler	**Doğruluk**	**Kesinlik**	**Hassasiyet**	**F1 Skor**	**"glove" Kelime Vektörlere Göre Katkısı**
glove + orth + fea	%95,99 ± 0,01	%94,31 ± 0,01	%95,99 ± 0,01	%94,93 ± 0,01	+0,42
glove + morph + fea	%96,08 ± 0,01	%94,46 ± 0,01	%96,08 ± 0,01	%94,98 ± 0,01	+0,47
glove + char + orth + morph	%95,69 ± 0,01	%94,29 ± 0,01	%95,69 ± 0,01	%94,73 ± 0,01	+0,22
glove + char + orth + fea	%95,95 ± 0,01	%94,42 ± 0,01	%95,95 ± 0,01	%94,94 ± 0,01	+0,43
glove + char + morph + fea	%95,88 ± 0,01	%94,35 ± 0,01	%95,88 ± 0,01	%94,76 ± 0,01	+0,25
glove + orth + morph + fea	%95,91 ± 0,01	%94,42 ± 0,01	%95,91 ± 0,01	%94,89 ± 0,01	+0,38
glove + char + orth + morph + fea	%95,95 ± 0,01	%94,37 ± 0,01	%95,95 ± 0,01	%94,93 ± 0,01	+0,42

Tablo 4.10'daki değerler incelendiğinde, "glove" kelime vektörleriyle morfolojik analiz sonucu tabanlı kelime vektörleri ile yazınsal öznitelik vektörleri birlikte kullanıldığında modelin en başarılı olduğu gözlemlenmektedir. Diğer vektör kombinasyonları incelendiğinde, ortografik karakter tabanlı kelime vektörleri ve

yazınsal öznitelik vektörlerinin “glove” kelime vektörleriyle kullanılmasının model başarısına en çok katkıyı yaptığı görülmektedir.

Karakter ve ortografik karakter tabanlı kelime vektörlerinin birlikte kullanılmasının, tek başına kullanıldıkları kombinasyonların üzerine pek katkı yapmadığı görülmektedir, dolayısıyla ikisinin model başarısına katkıda aynı etkiyi yaptığı sonucuna ulaşılmıştır.

Değerlendirilen kombinasyonlarda en çok katkıyı yazınsal öznitelik vektörlerinin yaptığı görülmektedir. Herhangi bir kombinasyona dahil olduklarında model başarısının en yüksek artışı gösterdiği görülmektedir.

En başarılı model olan “glove" kelime vektörlerinin morfolojik analiz sonucu tabanlı kelime vektörleri ve yazınsal öznitelik vektörleri ile kullanıldığı modelin (glove + morph + fea) gerçek ve tahmin ettiği etiket sayıları karmaşıklık matrisi token bazında Tablo 4.11’de gösterilmektedir.

Tablo 4.11 glove + morph + fea modelinin etiket bazlı karışıklık matrisi

		TAHMİN EDİLEN DEĞER				
		<unk>	*TARİH*	*SAAT*	*SÜRE*	*TEKRAR*
GERÇEK DEĞER	*<unk>*	5.931	11	0	4	0
	TARİH	67	127	0	12	0
	SAAT	22	6	0	2	0
	SÜRE	21	2	0	127	0
	TEKRAR	27	0	0	17	0

Tablo 4.11’deki değerler incelendiğinde, modelin saat ve tekrar ifadelerini tespitte başarısız olduğu ve bu ifadeleri zaman ifadesi olmadığı şeklinde etiketlemeye meyilli olduğu gözlemlenmiştir. Tarih ve süre ifadelerinin tespitinde ise girdi olarak sadece “glove” kelime vektörlerinin kullanıldığı modelden daha başarılı olduğu görülmektedir.

Veri kümesinde bulunan metinlerden örnek olarak 3 tanesinde yer alan kelimelerin etiketleri ve yine aynı modeli kullanarak elde edilen etiket tahminleri Tablo 4.12, 4.13 ve 4.14'te verilmiştir.

Tablo 4.12 Örnek bir metin için kelimelerin etiketleri ve etiket tahminleri

Metin	Geçen	sene	internet	üzerinden	kredi	çektim	.
Tokenize edilmiş metin	Gecen	sene	internet	uzerinden	kredi	cektim	.
Etiket	*TARİH*	*TARİH*	*<unk>*	*<unk>*	*<unk>*	*<unk>*	*<unk>*
Tahmini Etiket	*TARİH*	*TARİH*	*<unk>*	*<unk>*	*<unk>*	*<unk>*	*<unk>*

Tablo 4.13 Örnek bir metin için kelimelerin etiketleri ve etiket tahminleri

Metin	Fakat	birkaç	gündür	banka	tarafından	aranıyorum	.
Tokenize edilmiş metin	Fakat	birkac	gundur	banka	tarafından	aranıyorum	.
Etiket	*<unk>*	*SÜRE*	*SÜRE*	*<unk>*	*<unk>*	*<unk>*	*<unk>*
Tahmini Etiket	*<unk>*	*SÜRE*	*SÜRE*	*<unk>*	*<unk>*	*<unk>*	*<unk>*

Tablo 4.14 Örnek bir metin için kelimelerin etiketleri ve etiket tahminleri

Metin	Çalışma	saatlerimiz	bankaya	gitmeye	uygun	değil	.
Tokenize edilmiş metin	Calısma	saatlerimiz	bankaya	gitmeye	uygun	degil	.
Etiket	*SAAT*	*SAAT*	*<unk>*	*<unk>*	*<unk>*	*<unk>*	*<unk>*
Tahmini Etiket	*<unk>*	*<unk>*	*<unk>*	*<unk>*	*<unk>*	*<unk>*	*<unk>*

Tablo 4.12'te yer alan örnek metni incelediğimizde tarih ifadelerinin doğru şekilde tahmin edildiği, Tablo 4.13'te yer alan örnek metinde süre ifadelerinin doğru şekilde tahmin edildiği, Tablo 4.14'te yer alan örnek metinde ise saat ifadelerinin zaman ifadelerinden biri olarak tahmin edilemediği görülmektedir.

4.6 BERT Kelime Özniteliklerinin Kullanılması

3.2.6 konu başlığı altında anlatıldığı şekilde BERTurk modeli çıktıları üzerinden oluşturulan çeşitli kelime vektörleri modelde girdi olarak kullanıldığında elde

edilen başarı değerleri ve tezde yapılan çalışmalarda en başarılı sonuç veren (glove + morph + fea) vektör birleşiminin başarısına göre katkısı Tablo 4.15'te verilmiştir.

Tablo 4.15 BERT kelime öznitelikleri ile başarı değerleri

Kullanılan Çıktılar	**Doğruluk**	**Kesinlik**	**Hassasiyet**	**F1 Skor**	***glove + morph + fea Göre Katkısı***
İlk gizli katman	%97,64 ± 0,01	%96,51 ± 0,01	%97,64 ± 0,01	%96,78 ± 0,01	+1,80
Son gizli katman	%97,83 ± 0,01	%96,69 ± 0,01	%97,83 ± 0,01	%96,90 ± 0,01	+1,92
Sondan 2. gizli katman	%97,74 ± 0,01	%96,68 ± 0,01	%97,74 ± 0,01	%96,89 ± 0,01	+1,91
Son 12 katman toplamı	%95,71 ± 0,01	%94,36 ± 0,01	%95,71 ± 0,01	%94,65 ± 0,01	-0,33
Son 4 katman toplamı	%97,90 ± 0,01	%96,85 ± 0,01	%97,90 ± 0,01	%97,38 ± 0,01	+2,40

Tablo 4.15'teki F1 skor değerleri incelendiğinde, geliştirilen biLSTM ve CRF etiketleme modelinin, BERTurk model çıktılarından son 4 gizli katmanda yer alan değerleri kullandığında, en başarılı olduğu gözlemlenmiştir. Ayrıca, BERTurk modelinden kelime öznitelik çıkarma yöntemiyle elde edilen vektörlerin, en başarılı kelime vektörleri kombinasyonunu oluşturan glove kelime vektörleri ile morfolojik analiz sonucu tabanlı kelime vektörleri ve yazınsal öznitelik vektörlerinin (glove + morp + fea) birlikte kullanıldığında elde edilen başarıdan daha başarılı sonuçlar verdiği görülmüştür. Kelimelerin çeşitli özellikleri üzerinden elde edilen vektörlerin BERT modelindeki Transformer yapısı kadar başarılı olamadığı sonucuna ulaşılmıştır.

Tablo 4.16 BERT son 4 gizli katman modelinin etiket bazlı karışıklık matrisi

		TAHMİN EDİLEN DEĞER				
		<unk>	*TARİH*	*SAAT*	*SÜRE*	*TEKRAR*
GERÇEK DEĞER	*<unk>*	5.919	26	0	1	0
	TARİH	8	196	0	2	0
	SAAT	3	27	0	0	0
	SÜRE	10	6	0	133	0
	TEKRAR	11	10	0	23	0

Tablo 4.16'daki değerler incelendiğinde, son 4 gizli katmanın çıktıları kullanıldığında modelin tarih ve süre ifadelerinin tespitinde daha başarılı olurken saat ve tekrar ifadelerinin tespitinde aynı şekilde başarısız olduğu gözlemlenmiştir. BERT'in Transformers yapısı sayesinde kelimelerin metin içerisindeki yerini metin konsepti açısından daha iyi yakaladığı için daha başarılı olduğu düşünülmektedir.

5
SONUÇ VE ÖNERİLER

Bu tez çalışmasında Türkçe metinlerde zaman ifadelerini tespiti için ilk ANN tabanlı model geliştirilmiş ve kelime vektörleri kullanılarak başarısı ölçülmüştür.

Yapılan çalışma sonucunda önceden eğitilmiş kelime vektörlerinin, küçük bir veri setinde rastgele değerlerle başlayıp eğitilerek oluşturulacak kelime vektörleri ile kıyaslandığında model başarısına etkileyici bir katkısı olduğu gözlemlenmiştir. Farklı yöntemlerle eğitilen kelime vektörlerinin, kelimelerin farklı yönlerini içermesinden dolayı model başarısına katkılarının farklılık gösterdiği görülmüştür. Yapılan analizde, kelime vektörlerinin veri kümesini kapsama oranı ile kullanıldıklarında elde edilen model başarısı arasında doğru orantı olmadığı tespit edilmiştir.

Kirli metinler üzerinde çalışırken morfolojik analiz için konunun dile getirildiği tüm paragraf yerine cümleler kullanıldığında morfolojik analiz başarısının düşüşüne bağlı olarak analiz sonucu tabanlı kelime vektörlerinin kullanıldığı model başarısının da düştüğü gözlemlenmiştir. Bu tespit sonrası yapılan tüm çalışmalarda paragraf halinde metinlerin morfolojik analiz sonuçları kullanılmıştır.

Önceden eğitilmiş kelime vektörleri ile karakter tabanlı kelime vektörleri (char) ve ortografik karakter tabanlı kelime vektörleri (orth), geliştirilen öğe tabanlı biLSTM modeli ile oluşturulan morfolojik analiz sonucu tabanlı kelime vektörleri (morph) ve yazınsal öznitelik vektörlerinden (fea) birisi birlikte kullanılmasının model başarısını artırdığı gözlemlenmiştir. İkili vektör kullanımlarında en çok katkıyı yazınsal özniteliklerin verdiği görülmüştür.

Üzerine çalışılan konu özelinde çıkarılan yazınsal öznitelik vektörlerinin model başarısına etkileyici bir katkısı olduğu gözlemlenmiştir. Ancak çıkarılacak her yazınsal özniteliğin model başarısını ileriye taşımadığı, bazen modeli ezbere yönlendireceğinden model başarısını düşürdüğü görülmüştür.

BERTurk model çıktısı olan 13 katmanda yer alan değerlerin farklı şekillerde kullanımı sonucu elde edilen kelime vektörleri kullanıldığında en başarılı sonuçların elde edildiği gözlemlenmiştir. Farklı katman kombinasyonları kıyaslandığında, son 4 gizli katmanların çıktıları ile en iyi sonucun alındığı görülmüştür. BERTurk model çıktılarının daha başarılı olmasında, diğer modellerden farklı olarak kelimelerin

metin içerisinde iki yönlü eğitilmesiyle aralarındaki ilişkinin öğrenilmesinden kaynaklandığı düşünülmektedir.

Yapılan bu çalışmanın, etiketlenen kelimelerin doküman tarihi üzerinden normalize edilmiş tarihlerinin belirlenmesi, metinlerdeki olayların tespit edilmesi, tespit edilen olay ve zaman ifadeleri üzerinden metnin zaman çizelgesinin çıkarılması çalışmalarına ilham olacağı ümit edilmektedir.

Gelecekte sınıflandırıcı katmanında kelimenin öncesi ve sonrasında gelen kelimeler de dikkate alınarak çalışmanın geliştirilmesi planlanmaktadır.

KAYNAKÇA

[1] A. Setzer ve R. Gaizauskas, "Annotating Events and Temporal Information in Newswire Texts," Proceedings of the Second International Conference on Language Resources and Evaluation, 2000.

[2] L. Ferro, L. Gerber, I. Mani, B. Sundheim ve G. Wilson, "TIDES 2005 Standard for the Annotation of Temporal Expressions", MITRE Corporation, 2005.

[3] J. Pustejovsky, J. Castaño, R. Ingria, R. Saurí, R. Gaizauskas, A. Setzer ve G. Katz, "Timeml: Robust specification of event and temporal expressions in text," New Directions in Question Answering, 2003, p 28-34.

[4] R. Saurí, J. Littman, R. Gaizauskas, A. Setzer ve J. Pustejovsky, "TimeML Annotation Guidelines, Version 1.2.1", Technical report, 2006.

[5] M. Verhagen, I. Mani, R. Sauri, J. Littman, R. Knippen, S. B. Jang, J. Littman, A. Rumshisky, J. Phillips ve J. Pustejovsky, "Automating Temporal Annotation with TARSQI," Proceedings of the ACL Interactive Poster and Demonstration Sessions, 2005, p 81–84.

[6] M. Verhagen, R. Saurí, T. Caselli ve J. Pustejovsky, "SemEval-2010 Task 13: TempEval-2," Proceedings of the 5th International Workshop on Semantic Evaluation, 2010, p 57-62.

[7] J. Strötgen ve M. Gertz, "HeidelTime: High Quality Rule-Based Extraction and Normalization of Temporal Expressions," Association for Computational Linguistics, 2010, p 321–324.

[8] N. UzZaman, H. Llorens, L. Derczynski, J. Allen, M. Verhagen ve J. Pustejovsky, "SemEval-2013 Task 1: TempEval-3: Evaluating Time Expressions, Events, and Temporal Relations," Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013), 2013, p 1-9.

[9] A. X. Chang ve C. D. Manning, "SUTIME: A Library for Recognizing and Normalizing Time Expressions," Proceedings of the Eighth International Conference on Language Resources and Evaluation, 2012, p 3735–3740.

[10] N. Chambers, "NavyTime: Event and Time Ordering from Raw Text," Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013), 2013, p 73-77.

[11] J. Strötgen, T. Bögel, J. Zell, A. Armiti, T. V. Canh ve M. Gertz, "Extending HeidelTime for Temporal Expressions Referring to Historic Dates," Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14), 2014, p 2390-2397.

[12] W. Sun, A. Rumshisky ve O. Uzuner, "Evaluating temporal relations in clinical text: 2012 i2b2 Challenge," J Am Med Inform Assoc., cilt 20, no 5, 2013, p 806–813.

[13] K. Hacioglu, Y. Chen ve B. Douglas, "Automatic Time Expression Labeling for English and Chinese Text," Computational Linguistics and Intelligent Text Processing, 2005, p 548-559.

[14] N. UzZaman ve J. F. Allen, "Event and Temporal Expression Extraction from Raw Text: First Step towards a Temporally Aware System," International Journal of Semantic Computing, cilt 04, no 04, 2010, p 487-508.

[15] M. Filannino, G. Brown ve G. Nenadic, "ManTIME: Temporal expression identification and normalization in the TempEval-3 challenge," Proceedings of the Seventh International Workshop on Semantic Evaluation, 2013, p 53–57.

[16] X. Zhong ve E. Cambria, "Time Expression Recognition Using a Constituent-based Tagging Scheme," Proceedings of the 2018 World Wide Web Conference, 2018, p 983-992.

[17] S. Bethard, L. Derczynski, G. Savova, J. Pustejovsky ve M. Verhagen, "SemEval-2015 Task 6: Clinical TempEval," Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015), 2015, p 806-814.

[18] S. Velupillai, D. L. Mowery, S. Abdelrahman, L. Christensen ve W. Chapman, "BluLab: Temporal Information Extraction for the 2015 Clinical TempEval Challenge," Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015), 2015, p 815-819.

[19] S. Bethard, G. Savova, M. Palmer ve J. Pustejovsky, "SemEval-2017 Task 12: Clinical TempEval," Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017), 2017, p 565-572.

[20] S. MacAvaney, A. Cohan ve N. Goharian, "GUIR at SemEval-2017 Task 12: A Framework for Cross-Domain Clinical Temporal Information Extraction," Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017), 2017, p 1024-1029.

[21] W. Ding, G. Gao, L. Shi ve Y. Qu, "A Pattern-Based Approach to Recognizing Time Expressions," Proceedings of the AAAI Conference on Artificial Intelligence, cilt 33, no 01, 2019, p 6335-6342.

[22] E. Laparra, D. Xu ve S. Bethard, "From Characters to Time Intervals: New Paradigms for Evaluation and Neural Parsing of Time Normalizations," Transactions of the Association for Computational Linguistics, cilt 6, 2018, p 343–356.

[23] S. Chen, G. Wang ve B. F. Karlsson, "Exploring Word Representations on Time Expression Recognition," Microsoft Research, 2019.

[24] S. Almasian, D. Aumiller ve M. Gertz, "BERT got a Date: Introducing Transformers to Temporal Tagging," arXiv:2109.14927, 2021.

[25] J. Strötgen, A. Armiti, T. V. Canh, J. Zell ve M. Gertz, "Time for More Languages: Temporal Tagging of Arabic, Italian, Spanish, and Vietnamese," ACM Transactions on Asian Language Information Processing, cilt 13, no 1, 2014, p 1–21.

[26] J. Strötgen ve M. Gertz, "A Baseline Temporal Tagger for all Languages," Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, 2015, p 541-547.

[27] J. Kocoń ve M. Marcińczuk, “Recognition of Polish Temporal Expressions,” Proceedings of Recent Advances in Natural Language Processing, 2015, p 282-290.

[28] B. Altuna, M. J. Aranzabe ve A. D. D. Ilarraza, “EusHeidelTime: Time Expression Extraction and Normalisation for Basque,” Proces. Del Leng. Natural, cilt 59, 2017, p 15-22.

[29] M. Navas-Loro ve V. Rodríguez-Doncel, “Annotador: a temporal tagger for Spanish,” Journal of Intelligent & Fuzzy Systems, cilt 39, no 2, 2020, p 1979-1991.

[30] Y. Karagöl, “Event Ordering in Turkish Texts,” Master Science of Thesis, 2010.

[31] S. Tatar, “Automating Information Extraction Task for Turkish Texts,” Doctor Philosophy of Thesis, 2011.

[32] S. Kazkılınç ve E. Adalı, “Labeling Turkish News Stories with CRF,” 7th International Conference on Application of Information and Communication Technologies, 2013, p 384-388.

[33] D. Küçük ve R. Steinberger, “Experiments to Improve Named Entity Recognition on Turkish Tweets,” Proceedings of the 5th Workshop on Language Analysis for Social Media (LASM), 2014, p 71-78.

[34] D. Küçük ve D. Küçük, “On TimeML-Compliant Temporal Expression Extraction in Turkish,” arXiv:1509.00963, 2015.

[35] G. A. Şeker ve G. Eryiğit, “Extending a CRF-based Named Entity Recognition Model for Turkish Well Formed Text and User Generated Content,” Semantic Web, cilt 8, 2017, p 625-642.

[36] E. K. Akkaya ve B. Can, “Transfer learning for Turkish named entity recognition on noisy text,” Natural Language Engineering, cilt 27, no 1, 2021, p 35-64.

[37] H. K. Çamcı ve G. Eryiğit, “Türkçe Zamansal İfadelerin Yakalanması ve Tanımlanması,” Bilişim Teknolojileri Dergisi, cilt 14, no 3, 2021, p 337- 343.

[38] C.-G. Lim, Y.-S. Jeong ve H.-J. Choi, "Survey of Temporal Information Extraction," Journal of Information Processing Systems, cilt 15, no 4, 2019, p 931-956.

[39] N. Reimers ve I. Gurevych, "Reporting Score Distributions Makes a Difference: Performance Study of LSTM-networks for Sequence Tagging," Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, 2017, p 338-348.

[40] Z. Huang, W. Xu ve K. Yu, "Bidirectional LSTM-CRF Models for Sequence Tagging," arXiv:1508.01991, 2015.

[41] G. Lample, M. Ballesteros, S. Subramanian, K. Kawakami ve C. Dyer, "Neural Architectures for Named Entity Recognition," Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, 2016, p 260–270.

[42] X. Ma ve E. Hovy, "End-to-end Sequence Labeling via Bi-directional LSTM-CNNs-CRF," Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, 2016, p 1064–1074.

[43] X. Glorot ve Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics, 2010, p 249-256.

[44] K. He, X. Zhang, S. Ren ve J. Sun, "Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification," arXiv:1502.01852, 2015.

[45] N. Reimers ve I. Gurevych, "Optimal Hyperparameters for Deep LSTM-Networks for Sequence Labeling Tasks," arXiv:1707.06799, 2017.

[46] J. Yang, S. Liang ve Y. Zhang, "Design Challenges and Misconceptions in Neural Sequence Labeling," cilt Proceedings of the 27th International Conference on Computational Linguistics, 2018, p 3879–3889.

[47] A. Aghaebrahimian ve M. Cieliebak, “Hyperparameter tuning for deep learning in natural language processing,” 4th Swiss Text Analytics Conference, 2019.

[48] O. Güngör ve E. Yıldız, “Linguistic Features in Turkish Word Representations,” 25th Signal Processing and Communications Applications Conference, 2017, p 1-4.

[49] J. Pennington, R. Socher ve C. Manning, “GloVe: Global Vectors for Word Representation,” Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, 2014, p 1532–1543.

[50] P. Bojanowski, E. Grave, A. Joulin ve T. Mikolov, “Enriching Word Vectors with Subword Information,” Transactions of the Association for Computational Linguistics, 2017, p 135-146.

[51] O. Güngör, E. Yıldız, S. Üsküdarlı ve T. Güngör, “Morphological Embeddings for Named Entity Recognition in Morphologically Rich Languages,” arXiv:1706.00506, 2017.

[52] A. Güneş, “Türkçe'de Varlık İsmi Tanıma,” Yüksek Lisans Tezi, 2018.

[53] Y. LeCun, P. Haffner, L. Bottou ve Y. Bengio, “Object Recognition with Gradient-Based Learning,” Shape, Contour and Grouping in Computer Vision, cilt 1681, 1999, p 319-345.

[54] S. Hochreiter ve J. Schmidhuber, “Long Short-Term Memory,” Neural Computation, cilt 9, no 8, 1997, p 1735-1780.

[55] J. D. Lafferty, A. McCallum ve F. C. N. Pereira, “Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data,” Proceedings of the Eighteenth International Conference on Machine Learning, 2001, p 282–289.

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

[57] J. Devlin, M. Chang, K. Lee ve K. Toutanova, “BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding,”, Proceedings of the 2019 Conference of the North American Chapter of the Association for

Computational Lnguistics: Human Language Technologies, cilt 1, 2019, p 4171-4186.

[58] M. Peters, S. Ruder ve N. A. Smith, "To Tune or Not to Tune? Adapting Pretrained Representations to Diverse Tasks", Proceedings of the Fourth Workshop on Repsentation Learning for NLP, 2019, p 7-14.

[59] F. Souza, R. Nogueira ve R. Lotufo, "Portuguese Named Entity Recognition using BERT-CRF", arXiv:1909.10649, 2020.

[60] "TurkishGloVe", erişim: 24 Aralık 2021, https://github.com/inzva/Turkish-GloVe

[61] T. Mikolov, I. Sutskever, K. Chen, G. Corrado ve J. Dean, "Distributed Representations of Words and Phrases and their Compositionality," arXiv:1310.4546, 2013.

[62] "Zemberek-NLP", erişim: 24 Aralık 2021, https://github.com/ahmetaa/zemberek-nlp

[63] R. Jozefowicz, W. Zaremba ve I. Sutskever, "An Empirical Exploration of Recurrent Network Architectures," Proceedings of the 32nd International Conference on International Conference on Machine Learning, cilt 37, 2015, p 2342-2350.

[64] "bi-lstm-crf", erişim: 24 Aralık 2021, https://github.com/jidasheng/bi-lstm-crf

[65] "BERTurk", erişim: 27 Nisan 2022, https://github.com/stefan-it/turkish-bert

TEZDEN ÜRETİLMİŞ YAYINLAR

Konferans Bildirileri

[1] E. Emirali ve M. E. Karslıgil, "Türkçe Metinlerde Zaman İfadelerinin Tespitinde Kelime Vektörlerinin Kullanulması," 30. IEEE Sinya İşleme ve İletişim Uygulamaları Kurultayı, 2022.