

TARGETED SENTIMENT ANALYSIS ON TURKISH TEXTS

by

Mustafa Melih Mutlu

B.S., Computer Engineering, Boğaziçi University, 2018

Submitted to the Institute for Graduate Studies in
Science and Engineering in partial fulfillment of
the requirements for the degree of
Master of Science

Graduate Program in Computer Engineering
Boğaziçi University

2022

ACKNOWLEDGEMENTS

First of all, I would like to express my profound gratitude to my advisor, Assoc. Prof. Arzucan Ozgür for her guidance, support and patience during my study.

I would also like to thank my thesis defense jury members, Assoc. Prof. Ahmet Cüneyd Tantığ and Assist. Prof. Suzan Üsküdarlı for participating the defense of this thesis, their time and their suggestions to improve this study.

I am grateful to Abudllatif Köksal for his mentorship, support and all the comments, suggestions about my work.

I would like to offer special thanks to my wife, Merve Yılmaz Mutlu for her endless support during my study and her help and contributions to annotate the dataset.

Lastly, I'd like to thank my family for their patience and support.

ABSTRACT

TARGETED SENTIMENT ANALYSIS ON TURKISH TEXTS

Sentiment Analysis (SA) is one of the Natural Language Processing (NLP) tasks whose goal is to understand subjective information from a piece of text. The increased accessibility to the Internet and thus to social media leads people to create an enormous amount of textual data. This data can store valuable information that waits to be extracted. Targeted Sentiment Analysis (TSA) specifically aims to extract sentiment towards a particular target from a given text. Sentiment analysis in English texts is a well-studied area and mainly requires human-annotated data for training. For languages such as Turkish, there is a lack of such annotated data. In the context of this study, we introduce an annotated dataset that consists of Twitter data in Turkish. It contains almost 4K sentences that are labeled for both sentiment analysis and targeted sentiment analysis. The proposed dataset allows us to train a TSA model on Turkish texts. We propose BERT-based models with different architectures, one of which is to be used as our baseline for TSA and the others are to improve this baseline. We observe that the performance of conventional SA models degrades when used for TSA data. We investigate the performance of several BERT-based architectures for this task. Our best performing model with target markers and max-pooling layer outperforms the F1-score of conventional BERT-based SA models by 13%.

ÖZET

TÜRKÇE METİNLERDE HEDEF ODAKLI DUYGU ANALİZİ

Duygu Analizi, metinden öznel bir bilgiyi elde etmeyi amaçlayan bir Doğal Dil İşleme görevidir. İnternetin, ve dolayısıyla sosyal medyanın, ulaşılabilirliğinin artması insanların büyük miktarlarda metinsel veri üretmesine olanak sağladı. Bu veriler içlerinde çıkarılmayı bekleyen değerli bilgiler barındırıyor olabilir. Duygu analizi bu bilgiyi çıkarmaya odakalanan ve zamanla da popüler olan bir alandır. Bütün metnin duygusunun, o metinde geçen bir hedefe karşı olan duyguyu yansıtmadığı durumlar söz konusu olabilir. Hedef Odaklı Duygu Analizi verilen bir metinden verilen bir hedef kelimeye yönelik olan duyguyu çıkarmayı amaçlar. İngilizce metinlerde duygu analizi çokça çalışılan bir alandır ve çoğunlukla eğitim için insan eliyle işaretlenmiş veriler gereklidir. Türkçe gibi dillerde böyle işaretlenmiş verilerin eksikliği görülür. Bu çalışmanın kapsamında, duygu analizi ve hedef odaklı duygu analizi için işaretlenmiş bir veri kümesi sunuyoruz. Bu veri kümesi yaklaşık 4 bin adet hem duygu analizi hem de hedef odaklı duygu analizi görevlerine göre işaretlenmiş cümle içeriyor. Bu önerilen veri kümesi Türkçe metinlerde hedef odaklı duygu analizi modeli eğitmemize olanak sağlıyor. Önerilen BERT tabanlı modellerimizden bir tanesi temel model olarak, diğerleri ise bu temel modeli geliştirmek üzere eğitilmiştir. Farklı BERT tabanlı model mimarilerinin bu görev için performansını inceliyoruz. Geleneksel duygu analizi modellerinin performansının hedef odaklı duygu analizi görevinde düştüğünü gözlemliyoruz. En iyi sonuç veren hedef belirteçli ve maksimum havuz katmanlı modellerimiz geleneksel BERT tabanlı duygu analizi modellerinden %13 daha iyi F1 skoru verdiğini gözlemliyoruz.

TABLE OF CONTENTS

ACKNOWLEDGEMENTS	iii
ABSTRACT	iv
ÖZET	v
LIST OF FIGURES	viii
LIST OF TABLES	ix
LIST OF SYMBOLS	xi
LIST OF ACRONYMS/ABBREVIATIONS	xii
1. INTRODUCTION	1
1.1. Summary of Contributions	3
1.2. Thesis Outline	4
2. BACKGROUND	5
2.1. Sentiment Analysis Approaches	5
2.1.1. Machine Learning Approach	5
2.1.1.1. Supervised Methods	5
2.1.1.2. Unsupervised Methods	7
2.1.2. Lexicon-based Approach	8
2.1.3. Hybrid Approach	9
2.2. Targeted Sentiment Analysis	10
2.3. Twitter Sentiment Analysis and Challenges	11
2.4. Transformers	13
2.4.1. BERT	15
2.4.2. BERTurk	18
3. RELATED WORK	20
3.1. Turkish Sentiment Analysis	20
3.2. Sentiment Analysis Using Transformer	23
4. DATASETS	26
4.1. BounTi Dataset	26
4.2. New Dataset	26
5. METHODOLOGY	32

5.1. Problem Definition	32
5.2. Input and Output Representations	33
5.3. Tweet Preprocessing	36
5.4. Baseline Models	37
5.5. BERT-based Architectures for Targeted Sentiment Analysis	38
6. RESULTS	42
6.1. Baseline Results	42
6.2. Targeted BERT Model Results	43
6.3. Error Analysis	46
7. CONCLUSION	48
REFERENCES	50

LIST OF FIGURES

Figure 1.1.	Increasing interest in sentiment analysis since 2004 according to Google Trends.	2
Figure 2.1.	Overview of lexicon-based sentiment analysis pipeline.	9
Figure 2.2.	An overview of the transformer architecture.	14
Figure 2.3.	Masked model training with BERT. The model predicts the associated token for each masked token.	16
Figure 2.4.	Classification model with BERT.	17
Figure 4.1.	Distribution of sentiment classes in the dataset.	30
Figure 4.2.	Excel sheet used for labeling data by annotators.	31
Figure 5.1.	Input type representations.	34
Figure 5.2.	Output type representations.	35
Figure 5.3.	Target oriented BERT-based architecture.	40
Figure 5.4.	BERT-based architecture with target pooling.	41

LIST OF TABLES

Table 1.1.	An example sentence that shows different sentiment for different targets.	3
Table 2.1.	BERT _{Turk} vs. BERT _{base}	18
Table 4.1.	Sample tweets and their labels from BounTi test dataset.	27
Table 4.2.	List of target words/brands chosen to create the proposed dataset.	28
Table 4.3.	Sample data with different sentence and targeted sentiment. The targets are shown in italic.	29
Table 4.4.	Sample spam tweets that are removed from the dataset.	29
Table 4.5.	Label distribution for each dataset.	30
Table 5.1.	Examples of original tweets and their processed versions.	36
Table 5.2.	Baseline model and BounTi model results with different datasets for the traditional sentiment analysis task.	38
Table 5.3.	Sample sentences with target token [TAR].	39
Table 6.1.	Baseline model and BounTi model results with test dataset against targeted sentiment.	42
Table 6.2.	Results of targeted BERT models on the test dataset in terms of targeted sentiment.	43

Table 6.3.	Results of targeted BERT models on the test dataset in terms of sentence sentiment.	44
Table 6.4.	Overall results for TSA.	45
Table 6.5.	Results for TSA with data whose targeted and sentence sentiment are different.	46



LIST OF SYMBOLS

a	Aspect
a_{ij}	j_{th} aspect of the entity e_i
D	Document
E	Entity set
e_i	i_{th} entity
h_k	k_{th} holder of the sentiment
P	Probability function
s_{ijkl}	Sentiment at τ_l about a_{ij} of e_i held by h_k
t	Target
T	Target set
t_s	s_{th} target
w_i	i_{th} word
τ_l	Time when the sentiment is held

LIST OF ACRONYMS/ABBREVIATIONS

ABSA	Aspect-Based Sentiment Analysis
BERT	Bidirectional Encoder Representation from Transformer
BERT-FC	BERT with fully connected layer
BPE	Byte-Pair Encoding
DL	Deep Learning
DNN	Deep Neural Networks
GPU	Graphics Processing Unit
LSTM	Long-Short Term Memory
mBERT	Multilingual BERT
ML	Machine Learning
MLM	Masked Language Modeling
NB	Naïve Bayes
NER	Named-Entity Recognition
NLI	Natural Language Inference
NLP	Natural Language Processing
NSP	Next Sentence Prediction
POS	Part-of-speech
RNN	Recurrent Neural Network
SA	Sentiment Analysis
SVM	Support Vector Machines
T-BERT	Targeted BERT
TD-BERT	Target-Dependent BERT
TDK	Turkish Language Association
TSA	Targeted Sentiment Analysis
TPU	Tensor Processing Unit
QA	Question Answering

1. INTRODUCTION

Sentiment Analysis (SA) is one of the essential tasks of Natural Language Processing (NLP). It aims to grasp sentiment and opinions out of a given task [1]. By sentiment, the attitude of an author with respect to some topic or the overall tonality of a document is meant [2]. This subjective information which is extracted from a text by SA process can be highly valuable to understand the feelings behind the text.

The task of sentiment analysis can be defined as a classification problem at its core because it simply determines whether a text is positive, negative, or neutral. Therefore, the SA process, in fact, includes steps that are similar to the ones that are used for classification tasks [3].

The increasing availability of the Internet and the growing number of online platforms allowed people to easily generate online contents. This led to the creation of a huge amount of data in general. Textual data takes a significant portion of this generated online data. Social media, online reviews, blogs, etc. contain countless opinions about various topics such as brands, products, politicians, and more. The sentimental information which resides in that online data makes sentiment analysis an interesting topic to work on. It's not an area that draws the attention of only academic researchers but also companies and governments [3].

Due to the value of mining the people's opinions from text, there is an increasing interest in sentiment analysis. Figure 1.1 demonstrates the worldwide search interest of sentiment analysis according to Google searches. The date with the highest number of sentiment analysis searches is set to 100 and other dates are scaled with respect to this date. As can be seen in the figure, interest in sentiment analysis has increased since 2004. It has become one of the rapidly growing research areas when considering the number of SA papers published recently [4].

There are different text data sources to be used to develop a sentiment analysis process. Social media is among the most preferred source to gather text data. Twitter is widely used for sentiment classification in the literature [5] [6] [7] [8]. In this thesis, we also chose to use tweets for the purpose of sentiment analysis.

The problem with traditional sentiment analysis is that it cannot grasp the different attitudes to multiple aspects in one given text. For instance, if the given text is *“I like the food here, but the service is terrible”*, then sentiment analysis methods would not be able to detect opposite sentiment for *“food”* and *“service”*. [9]

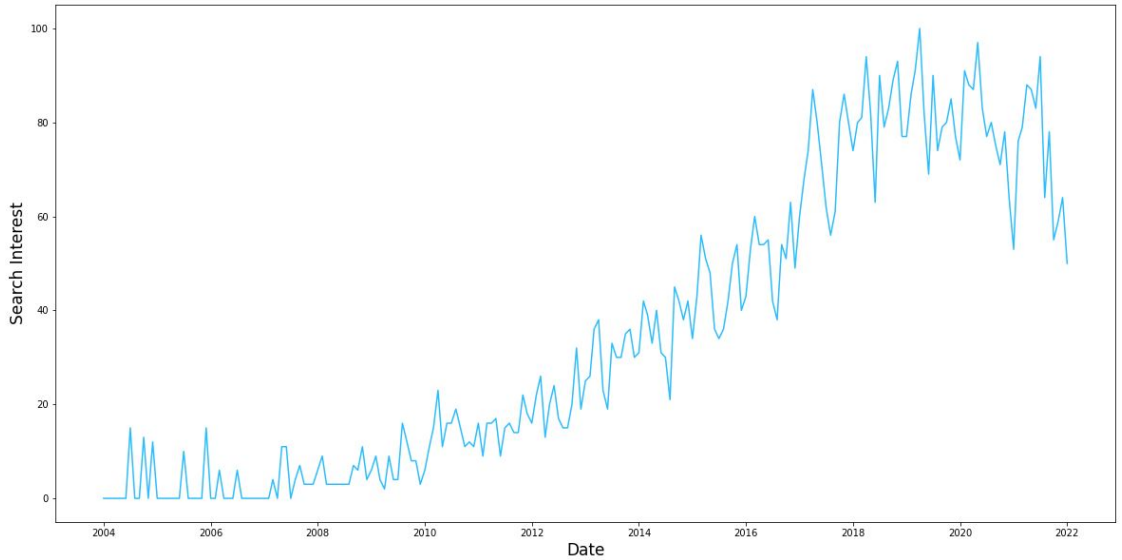


Figure 1.1. Increasing interest in sentiment analysis since 2004 according to Google Trends.

In the sentence shown in Table 1.1, it’s seen that sentiment can change according to the target. If the target is chosen as *“phone”*, then the sentiment towards to target word is positive since it’s stated as great. However, if the target word is *“laptop”*, its sentiment would be quite the opposite. Targeted Sentiment Analysis (TSA) aims to overcome this challenge and extracts sentiment from a text towards a given target.

Targeted sentiment analysis has its own challenges while it also inherits some of sentiment analysis’ challenges as well. The first one to mention is the lack of available

datasets. Both TSA and SA require labeled datasets. Collecting data from different kinds of sources and labeling them, which is mostly done manually, are expensive processes. Even though the number of datasets suitable for SA has increased recently with new studies in the SA area, not all SA datasets are usable for TSA [9]. TSA requires more fine-grained datasets. One example would be that labels should reflect the sentiment toward targets and not the overall sentiment of sentences.

Table 1.1. An example sentence that shows different sentiment for different targets.

Sentence	Target	Sentiment
Phones from this brand are great but I don't really like their laptops.	phone	positive
	laptop	negative

English is the most studied language for sentiment analysis [10]. Even though SA methods are improving with recent studies, these methods may not be directly applicable to morphologically rich languages such as Turkish [11]. SA models that perform satisfactorily for English do not seem to work with similar performance for Turkish [11]. While Turkish SA studies lag behind SA for English, there is also a dearth of studies in the literature that focus on targeted sentiment analysis for Turkish.

1.1. Summary of Contributions

This thesis addresses the targeted sentiment analysis problem for Turkish and makes the following contributions:

- (i) A dataset that is suitable for targeted sentiment analysis in Turkish is introduced. It contains approximately 4K sentences, which are manually annotated for both the SA and TSA tasks.
- (ii) New models for improving TSA in Turkish are proposed and compared with state-of-the-art approaches from the literature.
- (iii) The best performing model is publicly shared for further research studies.

1.2. Thesis Outline

In Chapter 2, we cover the subjects that are the backbone of this thesis. It includes a comprehensive investigation of sentiment analysis methods with several approaches without considering the language. Targeted sentiment analysis is also defined and common methods for this task are described. We also briefly mention the challenges of working with Twitter data. Finally, transformer and BERT architectures are overviewed.

In Chapter 3, related works that have similar goals to ours are explained. These works can be related to this thesis in two ways, either to Turkish SA or SA with transformers. Both topics are surveyed separately in this chapter.

In Chapter 4, we explain the processes of data retrieval from Twitter and annotation. The Specifications of our new dataset are included in this chapter. The Turkish SA dataset from [12] which we utilize in this thesis is also presented.

Chapter 5 covers our methodology for the Turkish targeted sentiment analysis task. We introduce our contributions with this work in this area. Each model's architecture that we propose for this task is shown and their training is explained in detail.

We evaluate all the models mentioned in the previous chapter and analyze their results in Chapter 6. The baseline models and the proposed TSA models are compared from multiple aspects.

Chapter 7 concludes this study. We give a brief overall summary of all the work and experiments we conduct.

2. BACKGROUND

In this section, topics that form the basis of this thesis are briefly discussed.

2.1. Sentiment Analysis Approaches

With the increasing interest in sentiment analysis, number of SA studies increases significantly [4]. The high number of works about SA and the interest from various backgrounds make different kinds of approaches to solve SA problems emerge. Since sentiment analysis can be applied to many domains and each domain has its own peculiarities, researchers approach this SA problem from different angles.

Even though this interest in SA leads to numerous techniques, sentiment analysis approaches can be divided into 3 main categories. Those are Machine Learning (ML) approach, Lexicon-based approach and Hybrid approach like in most classification problems [13] [14] [15]. Machine Learning approach employs well-known ML methods to classify the text while considering the language features. It's currently the most widely-used approach in the literature. Lexicon-based approach utilizes lexicons which consist of words that indicate sentiment [16]. Lastly, Hybrid methods combine both methods and leverage them.

2.1.1. Machine Learning Approach

Machine Learning (ML) methods can be used to classify the polarities of text. ML approaches require datasets for both training and testing to learn classes or specifically sentiment in SA context. ML methods can be roughly categorized as supervised and unsupervised methods.

2.1.1.1. Supervised Methods. Supervised methods need training data from a source that has previously been classified correctly [17]. It aims to learn the ability to predict

the sentiment of texts that haven't been seen earlier from labeled texts that have been fed into the system during the training phase [18].

We only cover a few of the widely used supervised methods in this section. The first method is Support Vector Machines (SVM). It has been developed to separate data linearly or non-linearly into categories. SVMs have been applied successfully to different domains of wide variety [19]. According to [20], Support Vector Machines is effective for text classification which makes SVM also suitable for sentiment analysis as well. According to [21], SVM outperforms other ML classification methods which are evaluated on a movie review dataset for sentiment analysis. Li and Li [22] conducted a study that mines opinions from microblog posts using Support Vector Machines. They define a credibility score for each post by extracting some language features. SVM is trained according to these extracted features. They conclude that SVM performs better than other supervised ML methods.

The second supervised Machine Learning method is Naïve Bayes (NB). Naïve Bayes is one of the most commonly used algorithm in the context of text classification [3]. It is based on the famous Bayes Theorem,

$$\mathbf{P}(\text{sentiment} \mid \text{text}) = \frac{\mathbf{P}(\text{sentiment})\mathbf{P}(\text{text} \mid \text{sentiment})}{\mathbf{P}(\text{text})}, \quad (2.1)$$

and relies on the features extracted from datasets. One drawback of NB is that it ignores word positioning so may have a lack of contextual information

Troussas *et al.* [23] employs Naïve Bayes model for sentiment analysis on social media status updates. They use two classes and label status updates as either positive or negative. With the feature extraction methods presented in the paper, they observe that Naïve Bayes classifier performs successfully on their social media dataset.

Deep Neural Networks (DNN) have recently become one of the commonly used methods for text classification. Deep Learning (DL) based methods offer building and training complex systems with the purpose of solving a complex problem with

large enough datasets. DL-based methods such as DNNs don't particularly need to be given pre-defined features determined by humans. Its ability to adapt and resolve the training set features with state-of-the-art performance makes this a popular approach for sentiment analysis [24] [25].

Among the different kinds of Deep Neural Networks, Recurrent Neural Networks (RNN) can be considered as an essential Neural Network for SA tasks. In RNNs, output depends on computations of previous tokens. This makes RNNs to be aware of contextual information. Thus it can provide better sentiment predictions [26]. However, RNNs suffer from vanishing gradients which affects their learning ability with long texts. Hochreiter and Schmidhuber [27] introduced a type of RNN which is named as Long-Short Term Memory (LSTM). LSTM extends RNN architecture by adding new units containing gates which are responsible to control historical flow from previous computations. These gates allow LSTM to decide what to keep in memory and what to remove from memory and what will be going through output. This addition tackles the long-term memorization problem with RNNs.

2.1.1.2. Unsupervised Methods. Sentiment analysis approaches which utilize supervised methods heavily depend on an annotated dataset. This annotation process is usually done by humans. Especially for SA tasks, manually labeling textual data enough to train an adequate supervised model is quite time-consuming and labor-intensive. This expensive cost of collecting and labeling data for supervised methods makes unsupervised methods more appealing.

While unsupervised methods also need a proper dataset to learn classification, the dataset does not have to be annotated. Unsupervised approaches make use of information such as text statistics and word representations [28]. Lexicons that contain sentimental or polarized words are also used for unsupervised learning. For example, in [28] the average sentiment of a text is computed by iterating over the opinion words in the text by making use of a lexicon.

Ma *et al.* [29] experimented the performance of clustering methods to achieve sentiment analysis goals. Clustering methods aim to partition data into non-overlapping sets without knowing any class information of data and which classes these non-overlapping sets represent. In literature, the most popular clustering algorithm for this task is claimed to be K-means [30]. The K-means algorithm starts with arbitrarily or non-arbitrarily selected cluster centroids for each given number of classes. It continues to process by iteratively updating the centroids with respect to similarities between data items. According to [31], K-means achieves the same performance with SVM on sentiment analysis for online forums data.

2.1.2. Lexicon-based Approach

Lexicon-based approaches are one of the two main ways for sentiment classification. Lexicon-based methods for sentiment analysis are dependent on the lexical resources which are list of words representing opinions. Such lexicons are also called as *opinion lexicons*. The main idea behind the lexicon-based methods is a measurement metric called semantic orientation. This measure of subjectivity expresses potential, strength, or likelihood of a sentiment class (positive, negative, neutral, etc.) towards the subject in a text [32].

Lexicon-based approaches infer the sentiment of a text by calculating the semantic orientation of words or phrases in the text and processing these calculations to make predictions [33]. Determining semantic orientations requires pre-defined dictionaries or lexicons for the specific goal. One of the main challenges of lexicon-based approach is generating such lexicons since it requires time and resources.

Lexicons can be created by humans determining and collecting words and their semantic orientations [33], or they can be created automatically by starting with a set of seed words [34] [35]. Such work focuses mainly on adjectives, since they tend to express strong sentiment (i.e. semantic orientation) of a text [33].

For sentiment analysis task, Turney [34] proposed an unsupervised SA method to classify sentiment as either positive or negative. He achieved to teach the model how to analyze a text's sentiment by calculating the semantic orientation of words. The model decides the sentiment by associating the semantic orientations to the two words namely poor and excellent.

One downside of the lexicon-based approach, other than the cost of creating a lexicon, is domain dependency. Due to the fact that meanings of some words may change from one context to another, lexicons might need to be switched, or created from scratch, for a study in a domain different than the one for which the lexicon is generated in the first place. Some common lexicons for sentiment analysis can be listed as WordNet [36], SentiWordNet [37], SenticNet [38].

2.1.3. Hybrid Approach

Hybrid approaches combine both lexicon and machine learning approaches to take advantage of the stability of lexicons and the high accuracy of ML methods. It combines the throughput of lexical analysis with the flexibility of machine learning approaches to cope with ambiguity and integrate the context of sentiment words [39]. One common way to sentiment analysis with the hybrid approach is to take advantage of a lexicon-based approach to label word sentiment to be later used to feed a supervised ML method.

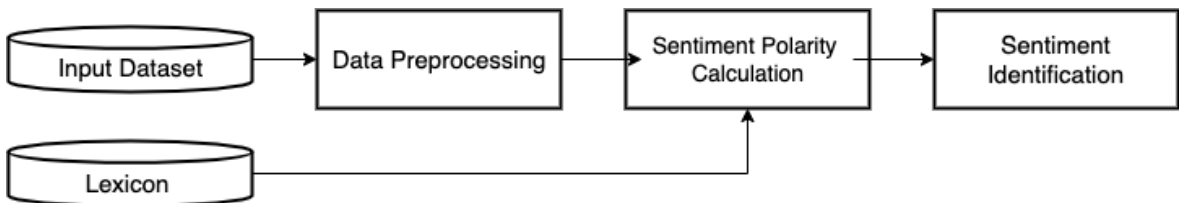


Figure 2.1. Overview of lexicon-based sentiment analysis pipeline.

2.2. Targeted Sentiment Analysis

Targeted Sentiment Analysis (TSA) can be considered as one step further from sentiment analysis. It investigates the opinions towards a specific target entity in a given text [5]. Defining a target and considering that target as the main entity which we're interested in the opinion about is the key concept that differentiates targeted sentiment analysis from traditional sentiment analysis [9].

TSA can be used interchangeably with Aspect-Based Sentiment Analysis (ABSA) in some studies. Even though there isn't a consensus on whether those are the same or not, there are some opinions that state slight differences between TSA and ABSA. Liu [40] defines opinion as

$$\mathbf{Opinion} = (e_i, a_{ij}, s_{ijkl}, h_k, \tau_l) \quad (2.2)$$

where;

- e_i is the entity,
- a_{ij} is an aspect of the entity e_i ,
- h_k is holder of the sentiment,
- τ_l is the time when the sentiment is held,
- s_{ijkl} is the sentiment at the time τ_l about the aspect a_{ij} of the entity e_i held by the holder h_k .

ABSA's objective is to predict the sentiment towards entity e with respect to aspect a . ABSA is interested in sentiment towards (e, a) tuple and picks such tuple as its target, according to Equation (2.2). On the other hand, TSA is not necessarily interested in an entity-aspect pair, but allows the target to be defined only as entity e or only as aspect a or both as a tuple [9].

In the following sample sentence “*The movie was so boring that I had to check on my Facebook all the time!*”, traditional sentiment analysis methods catch the negativity of the words such as *boring* and tend to label the above sentence as a negative review. SA omits the target which the sentence or the opinion is about and may not actually

have the awareness of it. Targeted sentiment analysis is to add that awareness into SA models and consider the target while classifying. In the example sentence above, TSA is likely to yield opposite results depending on the target. If the target is given as *movie*, then it is a negative review about the mentioned movie, and the model results in a negative label. However, there isn't an opinion stated about *Facebook*, so the resulting sentiment is not negative but neutral if the target is chosen as *Facebook*.

Dataset availability has an enabler effect on targeted sentiment analysis studies. It has that effect on sentiment analysis in general. The challenge with TSA datasets is that such datasets are needed to be more fine-grained. Existing SA datasets are not directly applicable for TSA tasks. Recent developments of TSA-suitable datasets have increased the number of researches focused on targeted sentiment analysis [9]. In particular, the datasets released by [41] for one of the tasks in SemEval 2014 have generated a lot of interest and brought more attention to TSA in the research community.

One challenge with targeted sentiment analysis is domain-dependency. TSA Datasets can have properties that make them consist of texts related to pre-defined domains [41] [42]. TSA models may tend to work well in a particular domain while not being satisfactory in other domains. To tackle this issue, creating open domain datasets that are collected from multiple-domain resources such as Twitter [43] [12] or using transfer learning methods to adapt a TSA model to different domains are common approaches in the literature [44].

2.3. Twitter Sentiment Analysis and Challenges

Twitter is one of the most popular social media platforms. It is usually called as a micro-blogging system since short textual contents have the biggest portion of all content created on Twitter. Until 2017, Twitter allowed its users to share posts called tweets with the length of only 140 characters. It has now doubled the limitation and allows 280 characters. Either 140 or 280 characters, Twitter content is usually a few sentences long at most anyway. Twitter's popularity around the world and its rapidly growing daily content make it an interesting and valuable source of textual data for

NLP tasks.

Micro-blogging data such as Twitter brings its own challenges in addition to the already existing one in NLP domain. Kireyev *et al.* [45] mentions the challenges of Twitter data in NLP processes. First of those challenges is the shortness of tweets due to Twitter’s character limitations. The small number of words in tweets also limits the information that can be extracted from a tweet. For sentiment analysis, it is difficult to classify tweets accurately since less word usage often means less contextual knowledge. Character limitations force Twitter users to omit some information by assuming that their followers can catch the context at that particular time the tweet posted, or because already have the context from previous tweets. However, SA systems can not understand that context which can probably influence the predictions from the short texts that exclude crucial information.

Another issue with Twitter data is the usage of informal language. Texts do not necessarily follow grammatical rules and can have missing required punctuations. Additionally, tweets often contain Internet slang and abbreviations or acronyms due to character limits. Even though commonly used acronyms such as *LOL* (laughing out loud), *ASAP* (as soon as possible), etc. can be handled, the Internet community continues to generate new acronyms over time. It’s also quite likely to have some spelling errors which can be accepted and understood by the community and are not considered as errors anymore. Such words do not exist in any lexicon. Those inconsistencies in the language of Twitter can harm the performance of sentiment analysis models.

Local and specific references are a factor that makes SA on Twitter data challenging [45]. It is common to not use proper complete names of people, organizations, brands, etc. This is actually a result of a lack of contextual awareness which comes with character limitations. Tweets can use names in a way that they are somehow shortened or abbreviated. Alternatively, Twitter users only write some part of the names such as only first name, surname, or even nicknames. While a tweet mentions a person, it may not use the person’s name at all, but uses Twitter handles to mention people. Those Twitter handle might not explicitly imply who the person is.

Users can also benefit from some Twitter features in their tweets to reach out to people or communities. In addition to mentions “@”, tweets can have hashtags “#”. Twitter hashtags can provide significant information about a tweet’s context and sentiment while it practically is against most grammatical rules and has no place in formal language. Ignoring the hashtags can be a part of the preprocessing step for sentiment analysis to preserve the actual tweet with relatively formal language. On the other hand, there are studies to utilize Twitter hashtags for the sake of SA purposes [46] [47].

Turkish alphabet contains some letters which do not exist in the Latin alphabet. Datasets retrieved from online resources can miss those letters, or there can be inconsistencies in the usage of words that contains such letters. Turkish Twitter data is most likely to have that kind of an issue in addition to all other challenges caused by the nature of the Internet language. The root cause of this difficulty can be the wide use of keyboards with English layouts in mobile devices. It is faster to write by simply ignoring Turkish characters. This is a point to consider while conducting Turkish sentiment analysis studies.

Despite all the challenges mentioned, Twitter datasets are preferred by researchers [7] [22]. It is easier to collect tweets and create a dataset programmatically by Twitter API ¹ It brings also some advantages into sentiment analysis area. It makes it easier to gather data related to different domains. SA works benefit from this to develop multi-domain methods and avoid domain-dependency of SA models.

2.4. Transformers

Deep Learning based methods have gained popularity in most of the NLP tasks. Sentiment analysis surely is one of those tasks that DL approach brought a perspective and improved results significantly. As a result, this success of DL approach leads studies to focus on such methods for NLP tasks.

¹<https://developer.twitter.com/en/products/twitter-api>

It's been observed that the performance for most NLP tasks increased with the pre-trained word embeddings. Word embeddings allows to keep the semantics of the words in a given sentence. Word2Vec [48], GloVe [49] and FastText [50] are among the widely used word embedding methods. With these developments, many studies aimed to integrate such word embeddings into their DL models. Especially training a model using RNN architecture with word embeddings is the main strategy in such studies. Even though such strategies have been proven to be effective and word embeddings are powerful tools to achieve NLP goals, those embedding methods are missing context-based mutability. All three word embeddings that we mentioned previously use fixed embeddings for each word regardless of the context that the word is used [51].

RNN challenges mentioned in Section 2.1.1.1 prevents building a pre-trained model for NLP. Such pre-trained models exist in Computer Vision and these allow to achieve tasks by only fine-tuning such models [52] [53]. These RNN issues and the problems of word embeddings were the obstacles, until Vaswani *et al.* introduced transformers [54].

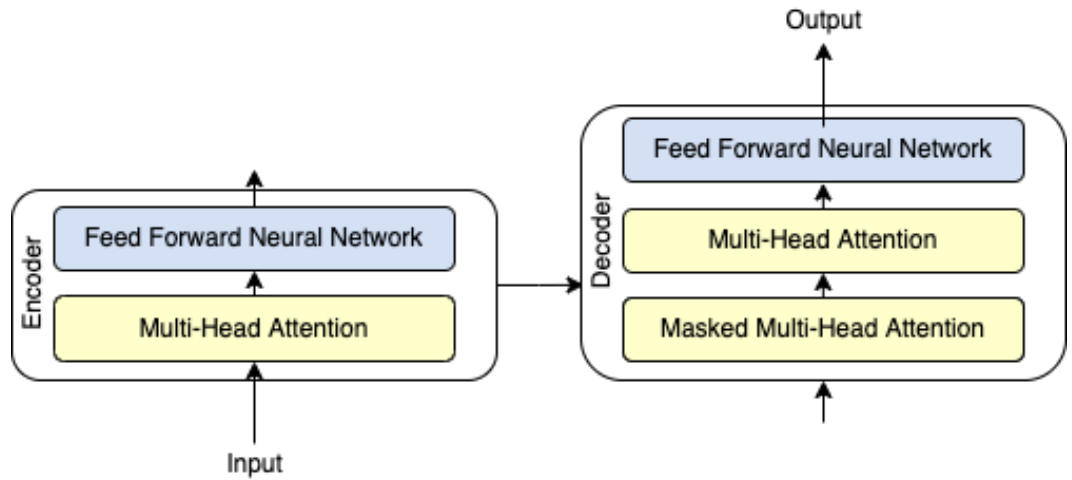


Figure 2.2. An overview of the transformer architecture.

The breakthrough transformer architecture is based on multi-headed self attention layer as shown in Figure 2.2. Two main blocks, encoders and decoders, form transformer. Both blocks have self attention layers followed by a feed-forward neural

network. Transformer has encoder blocks stacked on top of each other and input goes through this encoder stack. Same way, decoders are also stacked on top of decoders and this part's output is the output of the transformer. This novel idea enables to pretrain language models based on transformer by parallelization.

Training a transformer-based language model is a time and resource expensive process. But pretrained Transformer based language models can be adapted to many NLP tasks by fine-tuning. Fine-tuning is done by adding a dense layer after the last hidden state of pretrained model. Later, a supervised learning process is performed to train the additional dense layer. This is a rather inexpensive process than training the model from scratch or only pre-training.

Language models that benefit from transformer architecture are extensively used in recent studies. GPT-2 [55], one of the famous transformer based language models, reached state-of-the-art results when it was introduced. It uses left-to-right modeling and ignores the context from the right. Later Devlin *et al.* introduced a new model called Bidirectional Encoder Representations from Transformer (BERT) [56]. As the name implies, being bidirectional means considering both left and right context in a sentence.

2.4.1. BERT

Among the transformer-based language models, Bidirectional Encoder Representation from Transformer (BERT) [56] is one of the popular attention models. As opposed to the other models, BERT introduces a model which considers both left-to-right and right-to-left contexts. The results show that a bidirectionally trained model can have a deeper sense of context than its predecessors. The novel idea behind BERT is Masked Language Modeling (MLM) that allows training bidirectional models.

Masked Language Modelling is the technique that masks some of the input tokens and then feeds the masked input to BERT. This way trains BERT to predict masked tokens. It's mainly a task to predict the whole sentence. In order to fill the gaps, BERT

needs to be aware of the context from both sides. BERT takes the outputs of masked tokens into consideration and calculates a probability of each word from vocabulary for masked words' positions.

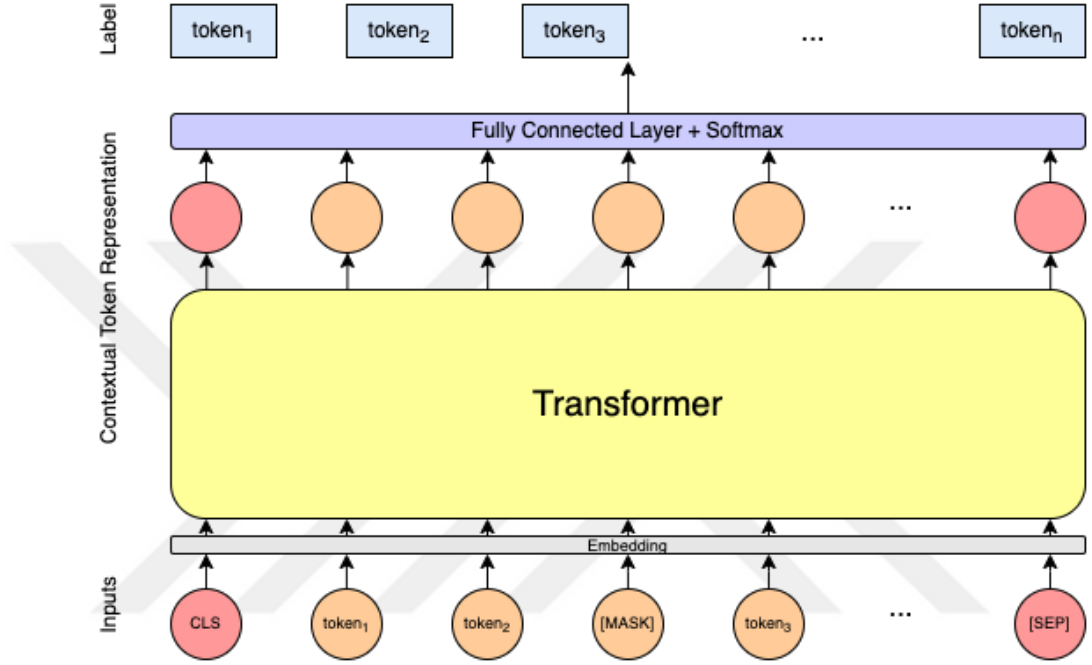


Figure 2.3. Masked model training with BERT. The model predicts the associated token for each masked token.

MLM's process starts with tokenization. BERT uses WordPiece [57] tokenizer that adapts a subword tokenization mechanism for building a vocabulary. WordPiece is based on Byte-Pair Encoding (BPE) [58] which is a method to split words into subword level. It firstly creates a vocabulary with all characters that occur in the text. In the next step, WordPiece tokenizer takes two elements from the vocabulary created in the first step and calculates the likelihood of word piece by combining the selected two elements. It applies this step to all combinations from vocabulary and chooses the piece with the highest likelihood. After this word piece is added into vocabulary, it repeats all these steps until it reaches a predefined threshold. For example, the word “probability” can be tokenized as “probab-##ility”. Even if “probability” has not been seen by the model yet, seeing the words that have similar subwords previously, such as “probably”, helps the model to get an understanding of the word “probability”. This

approach aims to solve problems caused by out of vocabulary words. It also significantly improves the performance with morphologically rich languages, like Turkish [57].

After MLM is done with tokenization, it masks 15% of input tokens. This tokenized and masked sentence is given to transformer encoders for training. What is expected from BERT at this step is estimating the masked tokens. This is done by adding one more layer to the top for classification. The last fully connected layer with softmax calculates the probabilities of being in a specific position for each token from the vocabulary. Figure 2.3 displays an overview of how BERT is pretrained.

To make BERT more powerful for NLP tasks, it's also pretrained for Next Sentence Prediction (NSP). A pretrained model with this understanding is relevant for tasks such as question answering. Two special tokens are introduced to the model for NSP. [CLS] token is added to the beginning of the sentence, [SEP] token is used to emphasize the end of each sentence. The output vector of [CLS] token is used to calculate the probability of whether the second sentence follows the first sentence. Figure 2.4.1 shows the architecture for NSP pretraining.

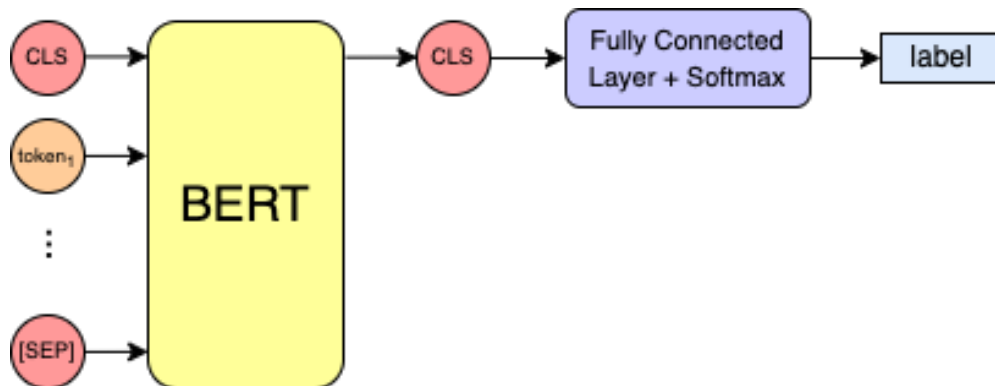


Figure 2.4. Classification model with BERT.

BERT is pretrained with two datasets. Those are BookCorpus [59] with 800 million words and English Wikipedia with 2.5 billion words. The paper [56] introduces two BERT models, $BERT_{base}$ and $BERT_{large}$. $BERT_{base}$ has 12 layers, 12 attention heads and 110 million total parameters while $BERT_{large}$ has 24 layers, 16 attention

heads and 340 million total parameters. The purpose of BERT_{base} is to compare the model with GPT-2. For this reason, a model with similar size with GPT-2 is pretrained alongside BERT_{large}. Both BERT versions outperform GPT-2 on the GLUE benchmark [60].

2.4.2. BERTurk

BERT’s success in many NLP tasks raised the interest in the model for other languages. The original BERT_{base} and BERT_{large} models are trained only for English. The need for such models for other languages results in efforts to train BERT-based models in various languages like Turkish.

Applying BERT to another language means training BERT from scratch. This training work for a particular language has several challenges. Firstly, this operation requires an enormous amount of unannotated or annotated corpus. Moreover, quality hardware such as GPUs and TPUs are needed to train such models.

Table 2.1. BERTurk vs. BERT_{base}.

Specifications	BERTurk	BERT _{base}
Number of Hidden Layers	12	12
Number of Self Attention Layers	12	12
Vocabulary Size	128K	31K
Languages	Turkish	English
Data Source	OSCAR Corpus, OPUS Corpora, Wikipedia, Turkish research corpora	BookCorpus, Wikipedia

Schweter [61] introduced a BERT model, BERTurk which is particularly trained for Turkish with many sources for the dataset. The corpus to train BERTurk is collected

from OSCAR corpus, OPUS corpora, Wikipedia, and a special corpus provided by Turkish researchers. In the end, the final training corpus has approximately 4 billion tokens. There are two versions of BERTurk with different vocabulary sizes, 32K and 128K.



3. RELATED WORK

In this section, we cover researches that are related to the motivation and objective of this thesis. Our methodology includes BERT-based targeted sentiment analysis on Turkish texts. Therefore, BERT-based SA and Turkish SA are briefly explained.

3.1. Turkish Sentiment Analysis

In this section, we investigate researches conducted with the goal of sentiment analysis in Turkish. We aim to show how SA approaches mentioned in Section 2.1 are applied in Turkish for different domains.

It's observed that most of the Turkish SA work applies supervised methods [62]. Various supervised methods are implemented and evaluated for Turkish SA. Despite the popularity of supervised methods, there are efforts to use unsupervised methods in the area. While studies based on unsupervised methods use a lexicon-based approach, work using supervised methods favours the machine learning approach.

Unsupervised studies for Turkish SA are usually based on translating English lexicons to Turkish and using these translated Turkish lexicons for further analyses [63]. Türkmenoğlu and Tantı [64] also form a Turkish lexicon by translating from English. In addition to translation, they add multi-word expressions, words with absence/presence suffixes, and other required words for Turkish. Since Turkish is an agglutinative language that allows adding suffixes at the end of the root word, a preprocessing step is needed as opposed to English SA. Suffixes added to root words can affect part-of-speech (POS) tags and sentiment orientations. The lexicon-based classifier is evaluated on two datasets. One is comprised of Twitter data and the other consists of movie reviews. Twitter data usually is more error-prone and ungrammatical than movie reviews. As a result, the performance of the lexicon-based classifier is 75.2% on the Twitter dataset and 79% on the movie review dataset. Even though it falls behind the ML-based method which is also developed in the same paper, the results show that

lexicon-based approach with added features into the translated lexicon is promising for Turkish SA.

Akgül *et al.* [65] also proposed a lexicon-based unsupervised SA method for Turkish. Their proposed lexicon is constructed by using adjective words from Turkish Language Association (TDK). Adjectives are labeled manually as positive or negative. Additionally, multi-word phrases and jargon words are added to the lexicon. The proposed dataset is built with texts from Twitter and essential preprocessing steps are applied on these data. Their lexicon-based method obtains 72% F1-score on the evaluations.

One of the earlier papers for Turkish SA with Machine Learning approach was published by Kaya *et al.* [11]. The research evaluates the performance of many traditional ML methods for the task of sentiment analysis in Turkish. Due to the lack of Turkish SA works at the time of this paper published, they defined a baseline with a simplistic approach. According to the claim in [21], languages have certain strong words that express feelings quite strongly. Creating a list of such words to determine the sentiment of a text can be sufficient to some extent. Thus, they created a list of such strong sentimental words and classified sentences based on whether they have positive or negative words from this list. To improve this baseline, four supervised Machine Learning methods are experimented and their results are evaluated. Those methods are Naïve Bayes, Support Vector Machines, Maximum Entropy, and N-gram language model. These methods are applied to the News domain. A dataset is collected from news data, especially news columns on political subjects. News columnists' work has many strong opinions compared to the other news data which aims to inform readers objectively. One challenge with this dataset that is mentioned in the paper is that there are some sentences which are expressing an opinion in favor of a political view while against the other political angles. Since the study does not focus on targeted sentiment analysis, it only considers the overall sentiment. Several feature extraction methods are also tested with all selected ML methods. Moreover, the importance of those features' frequency is investigated. For each feature extraction and ML method combinations, they repeat experiments with only taking the presence of the feature

into account, and then considering the frequencies of these features. They conclude that N-gram model and Naïve Bayes perform better than the other two methods. Both methods obtain an accuracy of slightly above 70%. An interesting observation is that SVM does not perform for Turkish SA as well as it does for English SA.

On the other hand, Türkmenoğlu and Tantı [64] also execute SVM for Turkish SA and their results are different than the paper by Kaya *et al.* [11]. They utilize the same Twitter and the movie review datasets which are used for their unsupervised method. To extract features from texts, tf-idf method is used. Their experiment covers three ML approaches; SVM, Naïve Bayes, and Decision Trees. Their result show that the accuracy of SVM is 85% on the Twtiter dataset and 89.5% on the movie dataset. Even though there is a slight difference between SVM and NB, SVM is the best performing method among all three.

Due to promising results of Deep Learning methods, Turkish SA studies follow the spreading trend of applying DL-based techniques in sentiment analysis. Among DL-based approaches, RNNs and transformer-based neural networks have been reported to obtain better results compared to traditional ML based approaches. According to [66], RNN, specifically LSTM, architecture outperforms traditional Machine Learning methods in Turkish SA. They obtain a dataset by gathering online reviews in Turkish, and they test RNN and other methods on this dataset. The dataset consists of reviews from an online shopping site and a website for Turkish movie reviews. Ratings of these reviews are used to label data as positive, or negative. Results show that LSTM structure accomplishes 82% accuracy which is higher than other methods, namely NB and logistic regression.

BERT is gaining interest in Turkish SA too. Recently, BERT-based solutions have started to emerge. Köksal *et al.* [61] conducted an experiment utilizing BERT-based model with an introduced dataset called as BounTi that is formed from Twitter. They evaluated different versions of BERT-based models trained for Turkish. The best performing model which is fine-tuned with BounTi dataset has 72.3% accuracy.

3.2. Sentiment Analysis Using Transformer

The transformer architecture and its variations have brought new state-of-the-art results for several Natural Language Processing tasks. How transformer works is explained in Section 2.4. In this section, we demonstrate applications and results from the literature for both traditional SA and TSA tasks.

Sun *et al.* [67] conduct experiments that evaluate the performance of BERT and some LSTM-based methods for targeted aspect-based SA. Among BERT-based models, BERT_{single} is trained for classification with multiple labels as many as the number of given aspects. Additionally, they develop BERT models with auxiliary sentences which transforms classification task to next sentence prediction task. Additional sentences convert ABSA task to either question answering (QA) or natural language inference (NLI) task, it depends on the form of added sentences. For QA approach, there are two types of auxiliary sentences, QA-M and QA-B. QA-M alters original text to a question such as *“What do you think of this aspect of that entity?”*. QA-B method adds three additional sentences in the forms of *“the polarity of this aspect of that entity is postive”*, *“the polarity of this aspect of that entity is negative”* and *“the polarity of this aspect of that entity is none”*. It aims to find the correct sentence by considering this method as binary classification where labels are *“yes”* or *“no”*.

On the other hand, NLI-M and NLI-B types of auxiliary sentences’ goal is to modify ABSA task in a way that it becomes NLI task. NLI types adds pseudo-sentences such as *“(entity - 1 - aspect)”* into original sentences. Similar to QA sentence types, The difference between NLI-B and NLI-M is that the former adds *“(entity - 1 - aspect)”*, whereas the latter adds *“(entity - 1 - aspect - negative)”* or similar pseudo-sentences with other sentiment polarities. Their results show that BERT-based models outperform other DL-based approaches. Moreover, all BERT with auxiliary sentence methods present results better than BERT_{single} for aspect detection and sentiment prediction. They claim that BERT with NLI models provides higher performance, even though all BERT with auxiliary sentence models are quite close to each other.

Gao *et al.* [68] utilize BERT for targeted sentiment analysis. They consider both entities and aspects as their models’ targets. They test multiple architectures based on BERT. All models are evaluated on three datasets. Two of those datasets are from SemEval 2016 [69], laptop and restaurant datasets. The third dataset is a Twitter dataset by [70]. The first two methods do not include any modification on inputs. Their first model which provides a baseline for BERT-based models is a conventional BERT model with fully connected layer (BERT-FC) trained for classification with [CLS] token’s output. Their target-dependent BERT model (TD-BERT) applies max-pooling to outputs of target words. The last two models are combined versions of BERT-FC and TD-BERT. In addition to that, they adapt the auxiliary sentence approach from [67]. One of two variants puts a sentence like “*what do you think of target?*” after the original sentence. Later, the [CLS] output and the max-pooling result of the target word outputs are multiplied. The result is fed into a fully connected layer. Another model variation again uses additional sentences as the earlier variant does. The distinction is that the last BERT model variant concatenates the [CLS] and max-pooled outputs instead of multiplying them. Consequently, all three proposed models achieved new state-of-the-art results on all datasets used in the paper. Results of TD-BERT and other two variants with auxiliary sentences outperform the models from [67]. Even though models with auxiliary sentences are expected to show an improvement, they state that it is not the case. The distinction between those three models is insignificant.

BERT has begun to be a widely-used model for Turkish SA too. In earlier sections, we already mentioned that Köksal *et al.* trained a BERT-based model with Turkish Twitter data for SA task. Another study by [71] assesses the performance of BERT for Turkish texts. The dataset used in this paper consists of movie reviews and hotel reviews datasets. Both datasets are labeled according to ratings associated with the review, they’re not manually annotated. They compare the results of three BERT models. The first model is based on Multilingual BERT (mBERT) which is trained for 104 languages including Turkish. The other two models are based on BERT_{base} and BERT_{large} introduced by Devlin *et al.* [56]. Due to the fact that these BERT models are trained for English, the dataset in this study is first translated into English. Later,

both models are fine-tuned with the translated dataset. In their results, BERT_{large} with only movie review dataset is the best performing among others. Overall, BERT-based models outperform previous methods mentioned in [71].



4. DATASETS

4.1. BounTi Dataset

The BounTi dataset is created for Turkish sentiment analysis. It consists of Twitter data which are about predefined universities. It is manually annotated by considering the sentimental polarities towards the universities by the students of the CMPE 493 course in the Spring 2018 term.

BounTi can be used as a second dataset for comparison. First of all, its source of data is Twitter. Therefore it has sentences with errors, slang, etc., and also this dataset is manually labeled. This means no rating or star is considered as a reflection of sentiment polarity. Tweets are categorized as positive, negative, and neutral. Even though there are slight filtering operations while creating BounTi dataset, it's possible to reach the original texts from tweets. It's because the dataset contains tweet IDs instead of texts themselves. This allows retrieving unprocessed texts for use in other studies.

Another aspect of BounTi is how sentences are annotated. Authors state that the sentiment towards a university that is mentioned in a particular text are regarded as labels. Hence BounTi can be used as a dataset for targeted sentiment analysis.

On the other hand, it has texts only about predefined universities. This can make BounTi perform well for only this specific domain. This domain dependency of the dataset can reduce the performance in other areas.

4.2. New Dataset

Targeted sentiment analysis requires datasets that are labeled for this specific task. As opposed to most existing datasets which are annotated with overall sentence sentiment, dataset for TSA needs to be annotated considering the sentimental polarity

towards a specific target. In the scope of this thesis, a dataset that fits the requirements of Turkish TSA task is created. We propose this dataset to train and test our models and also as a contribution to the Turkish targeted sentiment analysis field.

Table 4.1. Sample tweets and their labels from BounTi test dataset.

Tweet	Sentiment
Bogazicililere duyduğum saygi biraz abarti ama napiyim benim icin diger tum devlet universiteleri ve Bogazici Universitesi	positive
Bogazici universitesi sen ne tatli seysin Oyle :D	positive
Ah Boğaziçi Üniversitesi bana yar olur musun ahh	positive
Koskoca bogazici universitesi , diploma toreni canli yayınlanmıyor uzakta olanlar gormesin zaten di mi ieu bile canli yayinladi toreni	negative

Datasets are collected by online data from a variety of sources. Source matters since each online platform has different formats and kinds of user bases. Some may require or force users to use formal language compared to others. However, some platforms, especially social media platforms, do not have that sort of requirement. These are open to anyone from any background. Grammatical rules, Internet slang or memes, some invented words that are not actually in any dictionary but widely used by online communities are part of such data. These and more challenges explained in Section 2.3 are the nature of social media data.

Despite all these issues, importance of social media is widely acknowledged by companies, organizations, governments, and researchers. Such platforms are already being actively used by anyone who wants to promote or spread a product, an idea, or even a political figure. This democratic nature of social media leads to sharing freely. Hence it concludes that social media has comments, opinions directly from the user and is quite valuable due to the fact that it has this pure feedback from the audience. For these reasons, our goal in this thesis is to build a model to improve sentiment analysis performance on social media data.

Building a dataset that consists of reviews that have a rating associated with each review from online platforms such as e-commerce websites does not require a manual annotation process. Even though creation of such datasets requires less time and resource, we choose to create a manually annotated Twitter dataset. This choice serves our goal of building a sentiment analysis model for social media data. Our proposed dataset consists of Turkish tweets that are fetched by using some predefined target words. As our target, 6 brands and companies are chosen, without any particular reason, except that we assume that they have a sufficient number of tweets with different targeted and sentence sentiment classes. Table 4.3 shows the list of target words.

Table 4.2. List of target words/brands chosen to create the proposed dataset.

Word used to search on Twitter	Description
Hepsiburada	One of the largest e-commerce platforms
Trendyol	One of the largest e-commerce platforms
Turkcell	One of the leading mobile phone operators in Turkey
Coca Cola	A world wide famous beverage company
Whatsapp	Instant messaging application widely used in Turkey
Ekşi Sözlük	Collaborative dictionary-like Turkish website

In total, 3952 tweets are collected. The timestamp of the tweets spans a six months period, between January 2020 and June 2020. Each data has only one of the selected target words. This dataset is manually annotated with three labels which are positive, negative, and neutral. While annotating, two factors are considered, namely sentence sentiment and targeted sentiment. Each item has two labels, sentence sentiment expresses the overall sentiment of the sentence, while targeted sentiment reflects the sentiment to the target in that sentence. Table 4.3 shows the difference between sentence and target sentiment.

The collected tweets are annotated by two annotators separately. In the case of a conflict between annotators, they reevaluated conflicted tweets and reached an

agreement together. In Figure 4.1, the distribution of sentiment polarities in the dataset is shown. It's observed that the number of negative tweets is significantly higher than positive and neutral tweets for each target. We think that the notably high numbers of negative tweets are caused by the customers' tendency to write a review or post a complaint when they had a bad experience. The overall percentages of the positive, negative and neutral classes are 19%, 58% and 23%, respectively.

Table 4.3. Sample data with different sentence and targeted sentiment. The targets are shown in italic.

Tweet	Sentence Sentiment	Targeted Sentiment
<i>turkcell</i> ege bayisi müdürü otostopta ne güzel de denk gelmiştik geçen sene bu zamanlar	positive	neutral
<i>trendyol</i> 'un bildirimlerini kapattığım günden beri kafamda, telefonumda o kadar rahat ki anlatamam.	positive	negative
tamam çalışıyomuş <i>whatsapp</i> bana mesaj gelmiyomuş	negative	positive
aralıklı oruç yaparken <i>coca cola</i> zero içilir mi (sağlığımın önemi yok kola içmek istiyorum)	neutral	positive

Table 4.4. Sample spam tweets that are removed from the dataset.

Tweet
iyi geceler #bebekgen #hepsiburada #n11 #trendyol #iyigeceler #anne #bebek #annebebek #bebegim #kizbebek #erkebebek #bebekgiyim #bebekhediyesi #bebekbeslenme #yeniyl newyear...
daha fazla içerik için takip edin #facebook #instagram #corona #whatsapp #bitcoin #follow

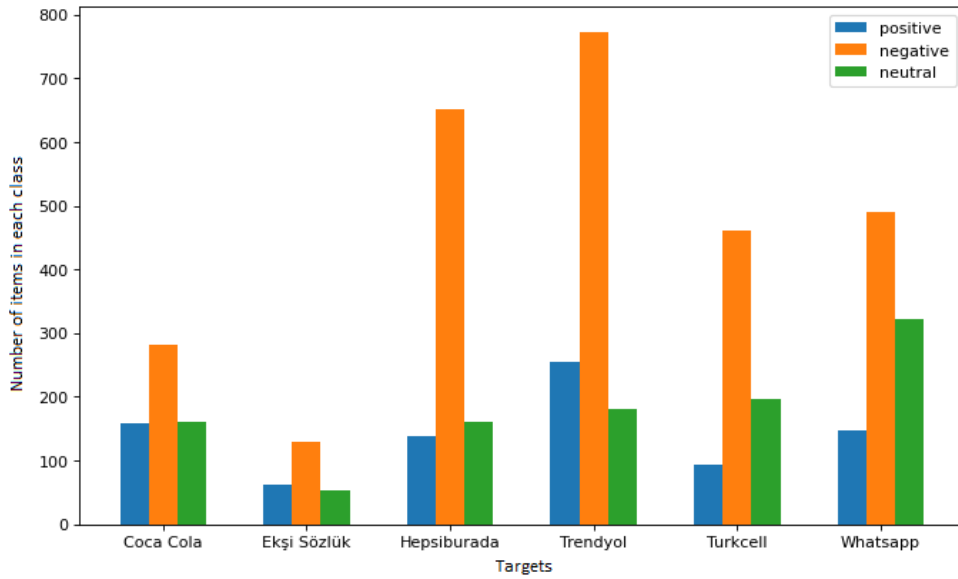


Figure 4.1. Distribution of sentiment classes in the dataset.

One aspect of the dataset is that both sentiment and targeted sentiment are actually the same for most of the data. Only in the 21% of tweets, targeted sentiment differs from overall sentence sentiment. This means that the rest of the dataset is similar to a standard sentiment analysis dataset. Although imbalanced, the dataset is split into test and train sets with approximately close distributions (See Table 4.5).

Table 4.5. Label distribution for each dataset.

Dataset	Positive	Negative	Neutral
Train	17.5%	59,5%	23%
Test	19.8%	61.2%	19%
Overall	19%	58%	23%

In Section 2.3, challenges working with Twitter data are explained. We're faced with similar challenges with this dataset. Preprocessing and filtering steps are applied to the dataset to resolve these issues. Firstly, spam tweets such as the ones that use popular keywords to get attraction are completely manually removed during annotation process. Example spam tweets are shown in Table 4.4. URLs and mentions "@" are also deleted from tweets, but the rest of the tweets are kept. We decided to keep

hashtags for two reasons: it's shown that hashtags express sentiment [46] [72] and some tweets use targets as hashtags instead of writing the word directly. In such cases, only hashtag signs (#) are removed. Nevertheless, there is not a preprocessing method to implement for some challenges that originated from the nature of Twitter data. The dataset includes many sorts of ungrammatical errors, slang, acronyms, etc. Note that tweets can also include multiple targets, even though they're annotated with only one target.

Timestamp	Text	Targeted Sentiment	Sentence Sentiment
2020-01-18T21:31:50.000Z	23 kasım'da aldığım ürünü bırakın teslim etmeyi nerde olduğunu dahi bilmiyorlar. teslim etmeden de teslim edilmiş gibi gösteriyorlar. 13 aralık ta gelmiş ürün haberim yok eğleniyormusunuz #trendyol	-	-
2020-01-18T21:15:50.000Z	umarım siparişlerimi kış bitmeden gönderirsiniz. trendyol	-	-
2020-01-18T21:08:57.000Z	trendyol her bildirim attığında fakfakirim diye ağlıyorum	n	-
2020-01-18T21:00:49.000Z	trendyol bu ara parasız kaldı galiba gece saat 22.30 da sipariş verilen ürünü 23.57 de kargoya vermişler alkışlıyorum	+	+
2020-01-18T20:47:01.000Z	trendyol ikide bir favori ürününe indirim geldi diye bildirim yolluyor açıp bakıyorum eski fiyatından daha pahalı puuuuuu	-	-
2020-01-18T20:21:37.000Z	lan trendyol 2 ay oldu ürünü ver http://şikayetvar.com'a kadar üye oldum senin yüzünden ver lan kulaklığımmmmmm	-	-
2020-01-19T22:31:12.000Z	nobody: trendyol yorumları: bir beden büyük alın	n	n
2020-01-19T22:12:10.000Z	trendyol batağına düştüm ne varsa alıyorum	-	-
2020-01-19T22:04:03.000Z	üstünüzdekiler trendyol'dan mı ??	n	n
2020-01-19T22:01:57.000Z	düştük yine trendyol batağına	-	-
2020-01-27T22:34:09.000Z	trendyol dan alışveriş yapınca çok heyecanlanıyorummm	+	+
2020-01-19T21:25:06.000Z	yves rocher trendyol'a gelmiş	+	+
2020-01-19T20:54:16.000Z	canım sıkkin olduğunda trendyol hepsiburada n11 gittigidiyor etc yani tüm online alışveriş dünyasına girip saçma şeyler almamaya niyet ettim amin	+	-
2020-01-19T20:33:09.000Z	tam trendyol dan kendime pijama takımı alıyordum kili bir baktımm sırtında eşşekkk kadar bride yazıyor	n	-
2020-01-19T17:33:32.000Z	kendimden çok özür dileyerek trendyol'a giriş yapıpıyordum	-	-
2020-01-27T21:20:40.000Z	trendyol favorilerim çok güzelsin ama hiç sepete ekleyemiyordum	+	-
2020-01-19T16:40:13.000Z	sürekli trendyol'a girip bir şeyler alıyorum diye uygulamasını telefondan sildim bu defa safariden girip duruyorum tutamıyorum kendimi sonum beni korkutuyor	n	-
2020-01-20T23:02:18.000Z	trendyol beni bari şu an sal	-	-
2020-01-28T23:10:03.000Z	sabah sanki ben kalkmıyacağım gibi relax bi şekilde trendyol vb. sitelere bakıyorum	n	-
2020-01-20T21:45:09.000Z	ocak 2'de verdiğim sipariş hala gelmedi. hazırlanma aşmasında bile değil. ocak 17 son gün gözüküyordu. kış bitmeden yollamayı düşünüyormusunuz trendyol?	-	-
2020-01-20T21:19:14.000Z	artık instagram da değil de trendyol da gezinmeye başladığın zaman anlıyorsun ayın 15'inin çok zor geleceğini	n	-
2020-01-20T21:17:34.000Z	trendyol yine sen kazandın	+	+
2020-01-20T21:03:10.000Z	trendyol ciğerimi sömürdün sömürdün	-	-
2020-01-20T21:01:37.000Z	trendyol satıcı pazaryeri, ömrümü yedinnnn . mutabıkız falan değiliz.	-	-
2020-01-20T21:00:49.000Z	bot sipariş etmiştim çizme gönderen koca yürekli trendyol	-	-
2020-01-21T23:53:32.000Z	trendyol paramı iade etmiş aferin böyle yola gel diyarbakırlıyız biz	+	+
2020-01-21T23:52:13.000Z	ayakkabı çanta bakacakken olmayan evim için yemek takımı neversim falan bakarken buldum kendimi. peki çift kişilik neversim takımının 50 tl olup tek kişilik olanının 55 tl olması? sen bir bataklıkıns trendyol.....	-	-
2020-01-21T23:33:11.000Z	trendyol şaka maka bilinçaltıma işledi sanırım. hiç bişey almayacağın halde aptal saptal tüm kategorileri geziyorum	-	-
2020-01-21T22:44:28.000Z	koca yok indirim çok #trendyol	+	+
2020-01-21T21:01:24.000Z	keşke trendyol reklamı gibi her yerde karşıma çıksan	-	-
2020-01-21T20:29:02.000Z	trendyol ev alışveriş günleri... aşık ettin az önce beni bir piring süzgecine	+	+
2020-01-22T23:03:07.000Z	düşüp çıkamadığım bataklar: star wars, lucifer, trendyol.	-	-
2020-01-22T22:39:49.000Z	ulan trendyol olumsuz yorumları reddedip yayınlıyorsun ne uyanıksın	-	-
2020-01-22T22:29:41.000Z	trendyol yüzünden gece gece oturduğum yerden 200 tlm gitti bir ay boyunca sehpa yicem artık yapacak biş yok.....	-	-
2020-01-22T22:14:36.000Z	ayın sonu ve benim 4'üne kadar 300 tlyle ayakta kalmam gerek ve ben trendyol dan alışveriş yapıyorum ne tarafa yığılabılırım	n	-

Figure 4.2. Excel sheet used for labeling data by annotators.

5. METHODOLOGY

In this chapter, we define our problem explicitly, propose our method and discuss its results for targeted sentiment analysis on Turkish texts. We study TSA with BERT-based [56] models. Specifically, BERTurk [61] is fine-tuned for Turkish TSA and compared with the results of different techniques. To begin with, the inability of traditional sentiment analysis models to determine targeted sentiment is shown. Then we define a baseline for Turkish TSA and aim to improve this baseline results. Subsequently, our proposed models are introduced with target markers and, the results of multiple models are compared.

5.1. Problem Definition

Equation (2.2) shows the formal definition of an opinion. It is defined as a quintuple like $(e_i, a_{ij}, s_{ijkl}, h_k, \tau_l)$ [40]. Opinion target can be defined either directly as an entity or a tree where root is an entity (i.e. object) and leaves are its aspects (i.e. features) [9]. In this study, entities are the targets and their aspects are ignored.

With this assumption, Equation (2.2) can be redefined for targeted sentiment analysis. The earlier quintuple can be modified by adding target t_i instead of entity e_i and removing aspects. Targeted opinion definition can be stated as

$$\text{Targeted Opinion} = (t_i, s_{ikl}, h_k, \tau_l) \quad (5.1)$$

where;

- t_i is the target,
- h_k is holder of the sentiment,
- τ_l is the time when the sentiment is held,
- s_{ikl} is the sentiment at the time τ_l towards the target t_i held by the holder h_k .

The objective of targeted sentiment analysis is to find all that $(t_i, s_{ikl}, h_k, \tau_l)$

quadruples in document D such that:

$$D = \{w_1, w_2, \dots, w_k\} \text{ where each } w \text{ is a word; } k \in \mathbb{Z}^+. \quad (5.2)$$

$$E = \{e_1, e_2, \dots, e_l\} \text{ where each } e \text{ is an entity; } l \in \mathbb{Z}^+. \quad (5.3)$$

$$e_i = \{w_m, w_{m+1}, \dots, w_{m+n}\} \text{ where } e_i \text{ is an entity, } w_i \in D; m, n, i \in \mathbb{Z}^+. \quad (5.4)$$

$$T = \{t_1, t_2, \dots, t_p\} \text{ where each } t \text{ is a target; } t_j \in E; p, j \in \mathbb{Z}^+. \quad (5.5)$$

In the scope of this thesis, each target t corresponds to only one entity from the set of entities E in document D and ignores aspects of that entity. It is assumed that the text document and the targets are given. Our objective is to extract the sentiment s towards target t in document D . Sentiment holders and time are not considered.

5.2. Input and Output Representations

Soares and his co-workers introduced a novel method with transformer structure in relation extraction field [73]. The key idea behind this work is marking the entities with additional tokens before feeding the BERT model with the input. They made two contributions to the field of relation extraction. One of them is investigating different architectures for the relation encoders based on BERT. The second goal of the work is to prove that these relation encoders can be learned from widely available distant supervision.

In this section, we briefly explain their first contribution with different architectures since it's related to the objective of this thesis. BERT is chosen as the base of their work due to its state-of-the-art performance. They perform experiments with different input representations and network architectures to extract relation representations. As input types which are analyzed in the paper, they picked standard, positional embeddings, and entity markers. As different variations of the BERT architecture, three output types are tested. Those are getting the output of [CLS] token, mention pooling, and entity start. Combinations of those input and output types are evaluated on four datasets: SemEval 2010 [74], KBP37 [75], TACRED [76], FewRel 5-way-1-shot [77].

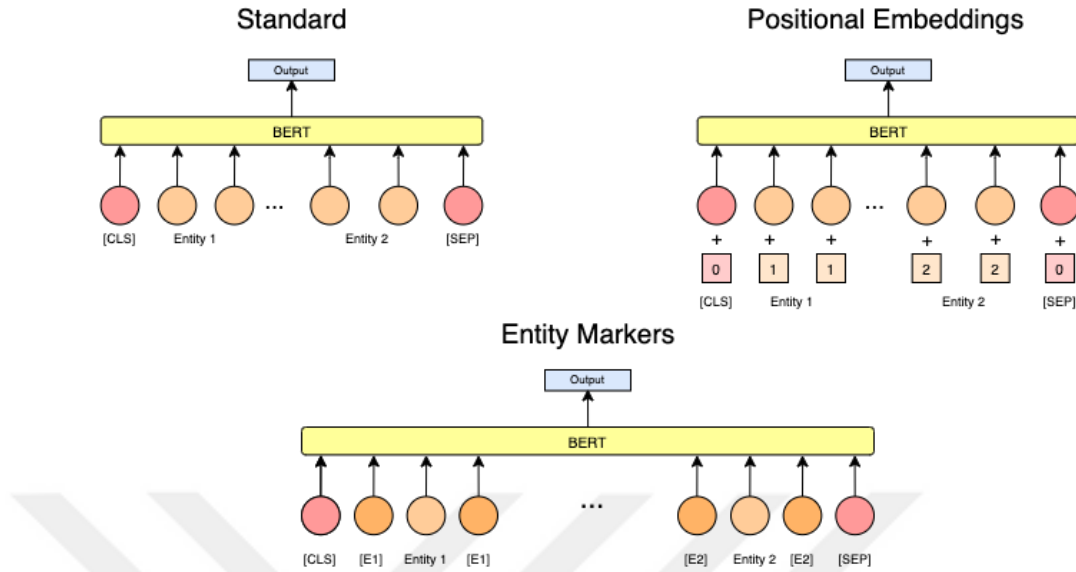


Figure 5.1. Input type representations.

Among the entity representation options, the standard way is to simply provide the inputs into the BERT model, without any modifications. BERT is believed to be an architecture to identify entities and grasp the relation between those without any explicit indicator. The standard input type is used as a baseline. The next input type is positional embeddings. It adds new segmentation embeddings to resolve the unawareness of the entities in the standard input type. The last type is a way to explicitly mark entities by adding additional entity tokens into the starts and the ends of entities. Figure 5.1 shows an overview of the architecture with different input types.

Using the output of the [CLS] token for classification is a method from the original BERT paper [56]. It's a reserved token specifically for gathering the context of the entire sentence, therefore it's suitable to be used for classification. Entity mention pooling obtains the result by max-pooling the outputs of entity tokens instead of the first token which is [CLS]. After max-pooling outputs of two entities separately, they concatenate those two pooled results and get a single vector representation. On the other hand, entity start method only concatenates the outputs of entity start tokens which are introduced with the entity marker input type. Figure 5.2 is a simplistic representation of the mentioned output types.

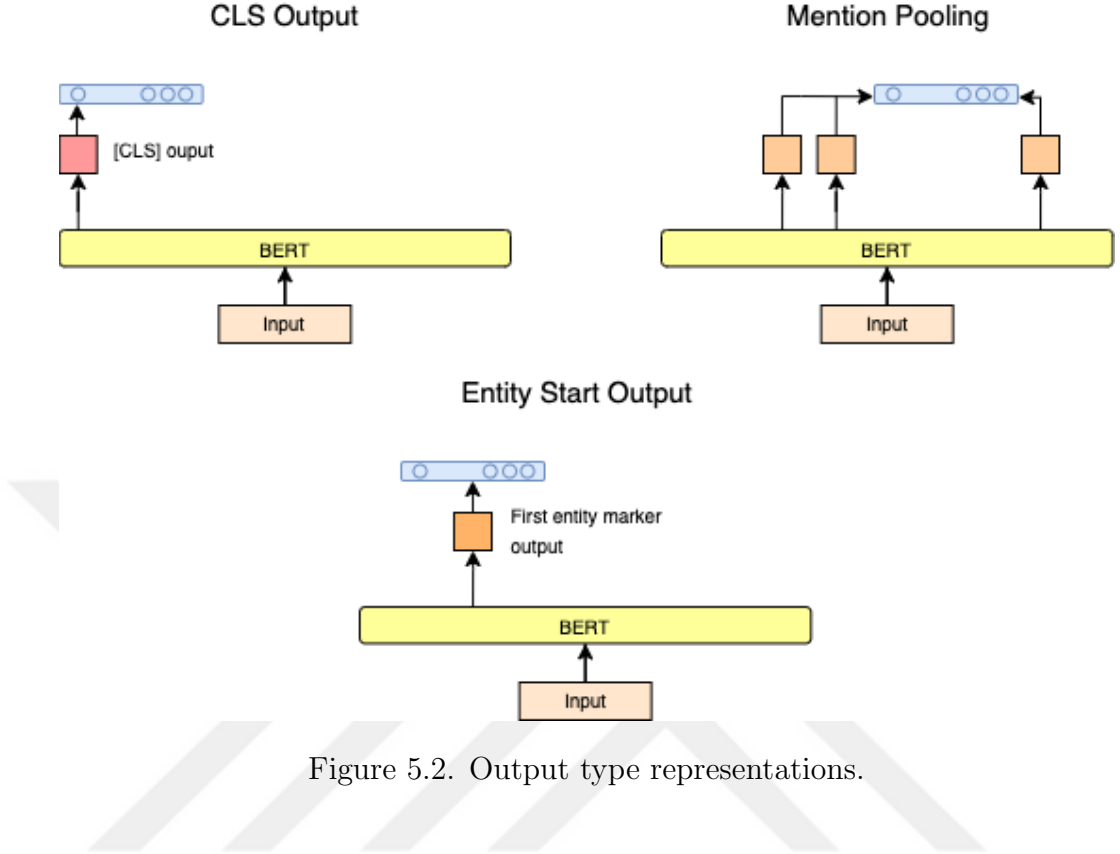


Figure 5.2. Output type representations.

In [73], several combinations of these input and output types are evaluated. The best results are obtained when entity markers are added to the input and when the outputs of the starting entity markers are taken as output from BERT. This model obtains the best F1-scores on all datasets. However, there is not a significant difference between the results of the best three performing combinations. One common point in those best three combinations is that all of them use entity markers as their input types.

Motivated by the results of [73], we aim to adopt some of these input and output type combinations for targeted sentiment analysis on Turkish texts. This thesis evaluates multiple architectures using BERT with different input and output types and compares the results.

5.3. Tweet Preprocessing

Tweets that are used to fine-tune BERT models in this thesis are preprocessed before they're used in model tuning. We do not conduct any serious preprocessing steps due to BERT-based model's ability to understand natural language without requiring an important preprocessing phase. Nonetheless, we decided to apply some steps with the intention of creating a more useful dataset that does not contain any meaningless, spam tweets from both real and bot users.

First condition to gather tweets that can be suitable for our dataset is that tweets should not be too short. For this reason, among the tweets retrieved via Twitter API, the ones with less than three words are removed. Spam tweets are also deleted manually. In the remaining tweets, any HTTP/HTTPS links are removed from original text. If a hashtag contains target word, only hashtag sign (#) is removed and the rest is kept as it is. However, if hashtag words are not related to the target word, they are completely removed from tweets. Mentions are also kept but mention symbols (@) are removed since they point to target words most of the time.

Table 5.1. Examples of original tweets and their processed versions.

Original Text	Processed Text
#trendyol sabaha bi sorunun kalmasın yoksa kafayı yicem	trendyol sabaha bi sorunun kalmasın yoksa kafayı yicem
babamla sıradan bir whatsapp konuşmamız yok #aliciakeys #goodjob	babamla sıradan bir whatsapp konuşmamız yok
super bowl'daki pepsi reklamı gizli coca cola reklamı. https://youtu.be/ddadu4-a7io	super bowl'daki pepsi reklamı gizli coca cola reklamı.

5.4. Baseline Models

We define two baseline models in order to demonstrate the drawbacks of traditional sentiment analysis models. One of those two models is the BERT-based model from BounTi [12]. The second one is also a BERT-based SA model, but it has been trained in the scope of this thesis with our new dataset. Both models have similar architectures but are fine-tuned with different datasets.

The BounTi ² paper performs experiments for the SA task with various BERT-based models trained for Turkish. We chose to benefit from the one which is based on BERTurk since their results show that it performs better. Their SA models and dataset are publicly shared and the models can be used with different datasets as well.

For our baseline model, BERTurk model is fine-tuned with our new dataset. Huggingface libraries are used for implementation [78]. 80% of the dataset is used for training and the rest is reserved as a test set. 15% of the training dataset is used for validation. The model architecture is built as in the BERT paper [56]. The input is tokenized by BERT tokenizer and given to the model. Among the outputs of each token, the first output of the last hidden layer which corresponds to the output of the [CLS] token is used for classification.

In the training phase, batch size is chosen as 24. Hyperparameters like weight decay, learning rate, and warm-up steps are selected as 0.1, $1e-5$, and 300 respectively. As optimizer, AdamW is used.

Table 5.2 shows the results of both base models with different datasets. Both models are trained with sentence sentiment as output labels. Therefore, BounTi and our baseline models aim to extract general sentiment from texts instead of targeted sentiment.

²<https://github.com/boun-tabi/BounTi-Turkish-Sentiment-Analysis>

Table 5.2. Baseline model and BounTi model results with different datasets for the traditional sentiment analysis task.

Proposed Test Dataset				
	Accuracy	Precision	Recall	F1
Baseline Model	0.714	0.648	0.655	0.647
BounTi Model	0.614	0.501	0.580	0.554
BounTi Test Dataset				
Baseline Model	0.563	0.538	0.578	0.542
BounTi Model	0.723	0.692	0.729	0.701

5.5. BERT-based Architectures for Targeted Sentiment Analysis

In this section, we propose architectures with the purpose of targeted sentiment analysis. Our approach to achieve this task is to adopt the approaches originally proposed for the relation extraction task in [73]. BERT-based model architectures that we believe can be suitable for TSA are implemented and evaluated.

We train and compare four variants of BERT-based models. Our first approach is similar to traditional sentiment analysis architectures. A simple overview of the model is shown in Figure 2.4.1. This model differs from our baseline BERT model in output labels. Targeted sentiment is used to fine-tune this model with pretrained BERTurk. While the baseline model predicts overall sentence sentiment, this model is target oriented. Therefore this variant can be considered as Targeted BERT (T-BERT).

On top of T-BERT, we introduce special tokens which mark targets as an additional improvement. This input representation method includes adding a special BERT token at the beginning and end of target words. We select our special token as [TAR] which represents the target and is similar to [CLS] or [SEP] tokens. This [TAR] token needs to be added to the tokenizer. The tokenizer should be familiar with the added special tokens so that it does not tokenize the tokens and keep them as a single token.

For implementation, we benefit from Huggingface transformer library [78]. In order to make the pretrained model appropriate for our special token, the [TAR] token is appended to the token list of the tokenizer of the BERTurk model, and the model’s vocabulary is resized. After the model and its tokenizer are adapted to handle the target tokens, we added the start and end target tokens to the input texts.

Table 5.3. Sample sentences with target token [TAR].

Sentences with Target Markers
teşekkürler [TAR]coca cola[TAR] lime, biz limonlu kola sevenleri pepsi twist’ten kurtardığın için.
canımdan çok sevdiğim kardeşim erva(8) [TAR]whatsapp[TAR] profiline benimle olan fotoğrafını koymuş
ben gerçek bi [TAR]trendyol[TAR] bağımlısıyım
koronavirüs [TAR]whatsapp[TAR]’ı da vurdu...
’internetten baktım. bu tweet dışında [TAR]ekşi sözlük[TAR]’te de veriliyor bu sonuçlar. ancak, henüz açıklanmamış.

With this input representation, our model is trained with target sentiment. This BERT model with target markers (T-BERT_{marked}) has the same architecture as T-BERT which is shown in Figure 2.4.1.

The third proposed variant is target oriented BERT model with target marker representation. The main idea of this model is to take the output of the start target marker. This method corresponds to the model with entity markers as input type and entity start as output method from [73].

Until this point, all models we trained in this thesis use the [CLS] output to classify sentences. According to [56], the [CLS] token is able to understand overall sentence context. Most of the works we mentioned in earlier sections also follow similar steps with the BERT paper [12] [71]. It has been shown that BERT indeed has the ability to have an overall context knowledge with special tokens like [CLS] [56]. Therefore,

we believe that it can be possible to know the context related to or towards target words with a kind of special token. [TAR] token is already implemented and added into the model. However, earlier model, T-BERT_{marked}, determines class labels with respect to the first output token which corresponds to [CLS]. Instead of this, we test whether retrieving the output of [TAR] from the last layer of hidden states can improve BERT's understanding of the sentiment towards target words. Figure 5.3 demonstrates the architecture of the third model which is briefly a BERT model with token markers as input and start target token as output (T-BERT_{marked}-TS).

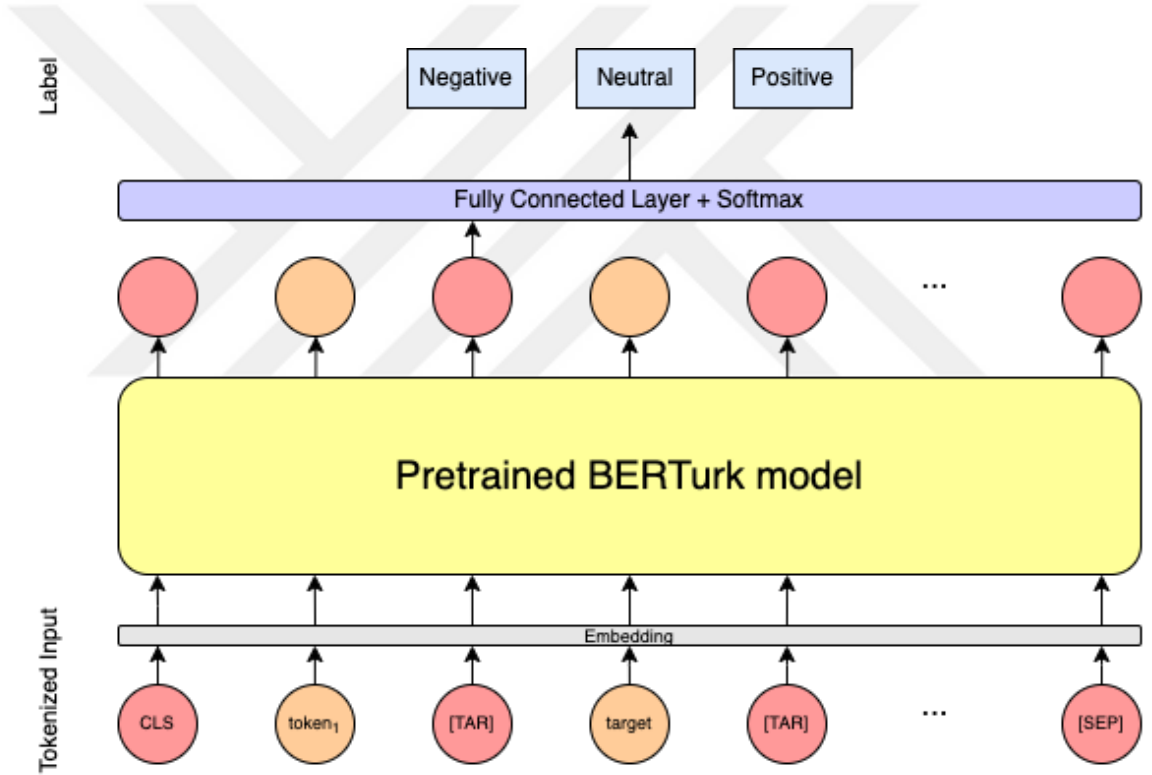


Figure 5.3. Target oriented BERT-based architecture.

The last architecture that we build for the TSA task is identical to T-BERT_{marked}-TS except for the way it handles model outputs. Its inputs are the sentences with target markers like T-BERT and T-BERT_{marked}-TS. Here, we obtain the final output by max-pooling the outputs of word pieces related to the target token and [TAR] tokens from the last hidden state. With this model (T-BERT_{marked}-MP), we assess whether other outputs originating from the target words bring a sentimental value to the model's result. Figure 5.4 shows the BERT architecture with target markers and

pooled output.

All models explained in this section are fine-tuned with identical hyperparameters. Batch size, weight decay, and learning rate are selected as 24, 0.1, and $1e - 5$ respectively. They're all fine-tuned in 20 epochs with 300 warm-up steps.

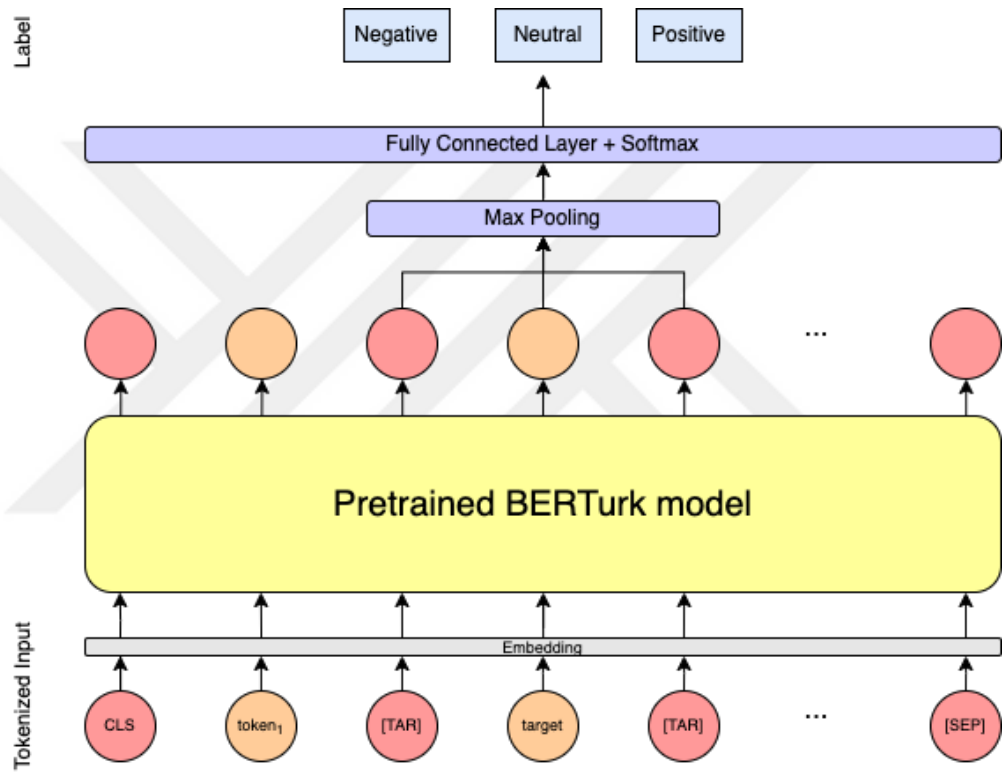


Figure 5.4. BERT-based architecture with target pooling.

6. RESULTS

In this section, we compare the performance of different models explained in Chapter 5 for Turkish targeted sentiment analysis. We provide a baseline BERT model and a BERT-based sentiment analysis model from related work. Later, our proposed models to improve this baseline for Turkish TSA task are investigated. As Dodge *et al.* show BERT can produce different results with different seeds [79]. Therefore, the results are presented by taking averages of 10 runs.

We would like to demonstrate and discuss the performance of the traditional SA models for TSA tasks with different models. Afterward, the performance of targeted sentiment models are shown for the SA and TSA tasks. Finally, we discuss the presented results and conduct an error analysis.

6.1. Baseline Results

As a baseline model, a BERT-based sentiment analysis model is trained as explained in Chapter 5. Additionally, we added the BounTi [12] model into our evaluations since it is trained in an identical way with our baseline model. Their performance for the SA task is shown in Table 5.2.

Table 6.1. Baseline model and BounTi model results with test dataset against targeted sentiment.

Test Dataset with Targeted Sentiment				
	Accuracy	Precision	Recall	F1
Baseline Model	0.640	0.607	0.584	0.591
BounTi Model	0.543	0.501	0.496	0.498

Table 6.1 demonstrates both models' results with the new dataset which is proposed in this study. These results are obtained by picking targeted sentiment as target

label. Thus, these are the results of traditionally trained SA models for the Turkish TSA task. Both models are run with the test dataset since the training dataset is used during training our baseline model, even though the training data has not been used by the BounTi model.

Comparing the results from Table 5.2 and Table 6.1, we see that the performance of both models decreased. The BounTi model obtains 55% F1-score for sentence sentiment. Nonetheless, its performance drops to 49% when it's tested with targeted sentiment. Likewise, our baseline model's F1-score is 64% for sentence sentiment. On the other hand, it is 59% for the TSA task. These declines in results clearly show that traditional SA models are not powerful enough in targeted sentiment analysis tasks.

6.2. Targeted BERT Model Results

We conduct experiments with BERT variants to achieve both targeted sentiment analysis and traditional sentiment analysis. TSA results of the proposed BERT models are presented in Table 6.2. As can be seen in the results, all variants outperform base models.

Table 6.2. Results of targeted BERT models on the test dataset in terms of targeted sentiment.

Test Dataset with Targeted Sentiment				
	Accuracy	Precision	Recall	F1
T-BERT	0.641	0.608	0.614	0.610
T-BERT _{marked}	0.692	0.664	0.659	0.659
T-BERT _{marked} -TS	0.674	0.648	0.662	0.653
T-BERT _{marked} -MP	0.685	0.665	0.684	0.669

Among all variants, T-BERT_{marked}-MP achieves the best results with 67%. T-BERT_{marked} and T-BERT_{marked}-TS come later than T-BERT_{marked}-MP, while T-BERT

is relatively the worst performing model with 61% F1-score.

T-BERT_{marked}-TS and T-BERT_{marked} obtain performance quite close to each other, the difference between those models is insignificant. They both have approximately 65% F1-scores. One thing which is safe to conclude according to these results is that all three models with target markers outperform T-BERT. Hence, adding target markers is a successful attempt to improve performance. We believe that it allows gathering whole sentimental targeted context into one token. Therefore, it results in better classification.

Moreover, we test the performance of these four variants for sentence sentiment too (See Table 6.3). Predicting sentence sentiment is the task that traditional SA models are trained for. It's seen that targeted BERT models lose some of their performance when they're used for SA task instead of TSA. None of the T-BERT variants performs as well as the baseline BERT model. This is fairly expected since T-BERT variants are trained with targeted sentiment. T-BERT_{marked}-MP is observed to have the best results for TSA, whereas it does not show this performance for SA. It is fairly expected that T-BERT_{marked}-MP considers most knowledge related to token words. This improves the ability to estimate targeted sentiment. This improved performance of T-BERT_{marked}-MP for TSA leads to loss of understanding of sentence sentiment.

Table 6.3. Results of targeted BERT models on the test dataset in terms of sentence sentiment.

Test Dataset with Sentence Sentiment				
	Accuracy	Precision	Recall	F1
T-BERT	0.610	0.560	0.586	0.555
T-BERT _{marked}	0.661	0.591	0.620	0.599
T-BERT _{marked} -TS	0.616	0.572	0.604	0.567
T-BERT _{marked} -MP	0.598	0.574	0.611	0.563

It's the same issue for T-BERT_{marked}-Ts as well. That models' architecture follows a target oriented path. This results in improved scores for TSA, but decreased performance for SA. Overall, T-BERT variants' unsatisfactory results for SA are understandable since they are not trained for it. Our purpose in comparing models trained for targeted sentiment analysis with a traditional sentiment analysis model is to clearly show the shift from understanding sentence sentiment to targeted sentiment.

As explained in Chapter 4, our dataset mostly contains sentences with the same targeted and sentence sentiment. 80% of the whole dataset consists of such sentences. Only 20% of these sentences from our new dataset can demonstrate the distinction between targeted and sentence sentiment classification. If the targeted sentiment of a sentence is the same as overall sentence sentiment, then traditional models may seem to accurately predict targeted sentiment. However, such sentences do not show how accurate the predictions from neither TSA nor SA models are.

Table 6.4. Overall results for TSA.

	Accuracy	Precision	Recall	F1
Baseline Model	0.640	0.607	0.584	0.591
BounTi Model	0.543	0.501	0.496	0.498
T-BERT	0.641	0.608	0.614	0.610
T-BERT _{marked}	0.692	0.664	0.659	0.659
T-BERT _{marked} -TS	0.674	0.648	0.662	0.653
T-BERT _{marked} -MP	0.685	0.665	0.684	0.669

For this reason, we created a subset of our new dataset such that all sentences have different targeted and sentence sentiment. Table 6.5 shows the results for the TSA task with this subset of data. We observe that the improvements are obvious with this subset. Baseline models' F1-score decreases dramatically to 25%, and it's 23% for BounTi model. Targeted BERT model with the lowest scores (40% F1-score) outperforms both models.

Comparisons of T-BERT models provide more information about their performance. T-BERT_{marked}-TS achieves better targeted sentiment predictions with 46% F1-score. T-BERT_{marked}-MP performs almost as good as T-BERT_{marked}-TS. We again observe that target markers improve TSA performance significantly on Turkish sentences. Additionally, taking target related outputs into account to predict labels has a good influence on results as well. It's either only target token or all outputs of target words and tokens, using target related output rather than [CLS] output outperforms other methods. In conclusion, we see that models that have both target markers in the input and use target related outputs can perform almost twice as well as traditional sentiment analysis models.

Table 6.5. Results for TSA with data whose targeted and sentence sentiment are different.

	Accuracy	Precision	Recall	F1
Baseline Model	0.295	0.313	0.298	0.256
BounTi Model	0.267	0.266	0.297	0.233
T-BERT	0.400	0.427	0.407	0.401
T-BERT _{marked}	0.492	0.455	0.470	0.428
T-BERT _{marked} -TS	0.556	0.475	0.477	0.459
T-BERT _{marked} -MP	0.563	0.453	0.458	0.444

6.3. Error Analysis

As explained in Chapter 4, the dataset consists of mostly negative tweets. Even if the dataset is used as balanced as possible in the training phase, these significant differences between the number of negative tweets and the others either lead us to use only a part of the dataset with balanced sentiment classes or the models tend to predict positive and neutral sentiment classes as negative.

Moreover, sentence and targeted sentiment are the same for 79% of the dataset. Thus, if a model that can predict sentence sentiment is used for the TSA task, it may be incorrectly considered capable of estimating targeted sentiment. Even the performance of targeted sentiment analysis models increases when they are tested with the entire test dataset, which consists of tweets that have mostly similar sentence and targeted sentiment. It is clear that targeted sentiment analysis models perform significantly better than traditional sentiment analysis models for TSA task. However, they may still be biased to some extent in favor of sentence sentiment.

Multiple targets with conflicting targeted sentiment in the same sentence is also a problem to consider. There are cases where a sentence has more than one target and each target has a different targeted sentiment. For example, in a comparison, the sentiment toward one target may actually depend on the sentiment of another target in the same sentence. In this work, only one target is considered for each sentence and the sentence is annotated only with respect to that one target. Target markers are also used for only one target in the sentence and other possible targets are ignored. The lack of proper treatment of such cases in this thesis may affect the performance of all models.

7. CONCLUSION

In this study, we aim to develop a model to improve results for targeted sentiment analysis on Turkish texts. We proposed a new dataset for the Turkish TSA task. The dataset is manually labeled according to sentence sentiment and targeted sentiment. Our next and main contribution is investigating and applying transformer-based methods for the Turkish targeted sentiment analysis task.

This study first covers the subjects that are essential to make a progress in our work. Sentiment analysis approaches are explained comprehensively and we mentioned the performance of these approaches for English SA tasks. Brief descriptions of the transformer architecture as well as challenges of working with Twitter data are also provided in this thesis. Especially, Bidirectional Encoder Representations from Transformer (BERT) is important to understand and a key for our models. We surveyed the studies from the literature related to Turkish sentiment analysis and the applications of transformer-based models for the SA task.

Our first contribution to Turkish TSA is forming a dataset that enables us to test existing models and build novel models in this area. Our source of data is Twitter so that we collected tweets about multiple brands and companies. Using several target words in the dataset is expected to prevent our dataset from domain dependency. As a result, we proposed a new Turkish dataset that consists of nearly 4K tweets and is suitable for both traditional SA and targeted SA with two labels for each sentence.

We investigated BERT-based models with different kinds of architectures for our goal. Our proposed approaches are motivated by the study [73]. We attempted to adapt input representations and output handling methods which have been shown to be effective in the relation extraction area. For the input representation, we introduced target markers that are special tokens placed just before and after target words. Furthermore, outputs of [CLS] token, first target tokens, and max-pooling all target tokens are evaluated and compared.

We observed that the addition of target token [TAR]’s into input sentences improves the performance notably, comparing it to our BERT-based baseline model. While all four proposed targeted BERT-based models outperform both our baseline models, T-BERT_{marked}-MP, which is the max-pooling model with target markers, is the best performing model with 67% F1-score on our dataset.

We tested all models on a subset of our dataset which contains only sentences with different sentence and targeted sentiment. This corresponds to approximately 20% of the whole dataset. Our purpose in this experiment with the subset data is to demonstrate the improvement on targeted sentiment predictions clearly. We showed that our best performing BERT-based model outperforms the baseline with a wide margin. It increased the F1-score from 25% to 46%.

For the future work, we plan to expand our proposed dataset so that it will have a larger number of tweets with different targeted and sentence sentiment. More data with different sentiment can be used in the training phase so that the models learn the targeted sentiment better. Additionally, sentences with conflicting sentiment for a given target can be processed to make more accurate predictions. Multiple targets in the same sentence is an aspect that can increase performance if handled properly.

These targeted sentiment analysis models can be combined with a Named-Entity Recognition (NER) system to create an enhanced pipeline for the TSA task. NER and TSA systems can work end-to-end. First, the NER system is required to recognise the targets in each sentence in the preprocessing phase and can reduce the manual work for target marking. Later, the processed input can pass through the TSA models.

REFERENCES

1. Liu, B., *Sentiment Analysis: Mining Opinions, Sentiments, and Emotions*, Cambridge University Press, 2015.
2. Mukherjee, S. and P. Bhattacharyya, “Sentiment Analysis : A Literature Survey”, *arXiv preprint arXiv:1304.4520*, 2013.
3. Birjali, M., M. Kasri and A. Beni-Hssane, “A Comprehensive Survey on Sentiment Analysis: Approaches, Challenges and Trends”, *Knowledge-Based Systems*, Vol. 226, p. 107134, 2021.
4. Mäntylä, M. V., D. Graziotin and M. Kuuttila, “The Evolution of Sentiment Analysis—A Review of Research Topics, Venues, and Top Cited Papers”, *Computer Science Review*, Vol. 27, pp. 16–32, 2018.
5. Jiang, L., M. Yu, M. Zhou, X. Liu and T. Zhao, “Target-dependent Twitter Sentiment Classification”, *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pp. 151–160, Association for Computational Linguistics, Portland, Oregon USA, 2011.
6. Ren, Y., Y. Zhang, M. Zhang and D. Ji, “Context-Sensitive Twitter Sentiment Classification using Neural Network”, *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 30, Phoenix, Arizona USA, 2016.
7. Tang, D., F. Wei, B. Qin, T. Liu and M. Zhou, “Cooooo!!!: A Deep Learning System for Twitter Sentiment Classification”, *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, pp. 208–212, Dublin, Ireland, 2014.
8. Severyn, A. and A. Moschitti, “Unitn: Training Deep Convolutional Neural Network for Twitter Sentiment Classification”, *Proceedings of the 9th International*

- Workshop on Semantic Evaluation (SemEval 2015)*, pp. 464–469, Denver, Colorado USA, 2015.
9. Pei, J., A. Sun and C. Li, “Targeted Sentiment Analysis: A Data-Driven Categorization”, *arXiv preprint arXiv:1905.03423*, 2019.
 10. Dashtipour, K., S. Poria, A. Hussain, E. Cambria, A. Y. Hawalah, A. Gelbukh and Q. Zhou, “Multilingual Sentiment Analysis: State of the Art and Independent Comparison of Techniques”, *Cognitive Computation*, Vol. 8, No. 4, pp. 757–771, 2016.
 11. Kaya, M., G. Fidan and I. Toroslu, “Sentiment Analysis of Turkish Political News”, *Proceedings - 2012 IEEE/WIC/ACM International Conference on Web Intelligence, WI 2012*, pp. 174–180, Macau, China, 2012.
 12. Köksal, A. and A. Özgür, “Twitter Dataset and Evaluation of Transformers for Turkish Sentiment Analysis”, *2021 29th Signal Processing and Communications Applications Conference (SIU)*, 2021.
 13. Maynard, D. and A. Funk, “Automatic Detection of Political Opinions in Tweets”, *The Semantic Web: ESWC 2011 Workshops: Workshops at the 8th Extended Semantic Web Conference, Revised Selected Papers*, Vol. 7117, p. 88, Springer Science & Business Media, 2012.
 14. Thakkar, H. and D. Patel, “Approaches for Sentiment Analysis on Twitter: A State-of-Art Study”, *arXiv preprint arXiv:1512.01043*, 2015.
 15. Madhoushi, Z., A. R. Hamdan and S. Zainudin, “Sentiment Analysis Techniques in Recent Works”, *2015 Science and Information Conference (SAI)*, pp. 288–291, 2015.
 16. Jurek, A., M. D. Mulvenna and Y. Bi, “Improved Lexicon-based Sentiment Analysis for Social Media Analytics”, *Security Informatics*, Vol. 4, No. 1, pp. 1–13,

2015.

17. Sathya, R. and A. Abraham, “Comparison of Supervised and Unsupervised Learning Algorithms for Pattern Classification”, *International Journal of Advanced Research in Artificial Intelligence*, Vol. 2, No. 2, pp. 34–38, 2013.
18. Alloghani, M., D. Al-Jumeily Obe, J. Mustafina, A. Hussain and A. Aljaaf, *A Systematic Review on Supervised and Unsupervised Machine Learning Algorithms for Data Science*, pp. 3–21, 2020.
19. Evgeniou, T. and M. Pontil, “Support Vector Machines: Theory and Applications”, *Advanced Course on Artificial Intelligence*, pp. 249–257, Springer, 1999.
20. Joachims, T., “Text Categorization with Support Vector Machines: Learning with Many Relevant Features”, *Proceedings of the 10th European Conference on Machine Learning*, p. 137–142, Chemnitz, Germany, 1998.
21. Pang, B., L. Lee and S. Vaithyanathan, “Thumbs up? Sentiment Classification using Machine Learning Techniques”, *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing*, pp. 79–86, Association for Computational Linguistics, Philadelphia, USA, 2002.
22. Li, Y.-M. and T.-Y. Li, “Deriving market intelligence from microblogs”, *Decision Support Systems*, Vol. 55, No. 1, pp. 206–217, 2013.
23. Troussas, C., M. Virvou, K. J. Espinosa, K. Llaguno and J. Caro, “Sentiment Analysis of Facebook Statuses Using Naive Bayes Classifier for Language Learning”, *4th International Conference on Information, Intelligence, Systems and Applications, IISA 2013*, pp. 1–6, 2013.
24. Zhang, L., S. Wang and B. Liu, “Deep Learning for Sentiment Analysis: A Survey”, *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, Vol. 8, No. 4, p. e1253, 2018.

25. dos Santos, C. and M. Gatti, “Deep Convolutional Neural Networks for Sentiment Analysis of Short Texts”, *Proceedings of the 25th International Conference on Computational Linguistics: Technical Papers*, pp. 69–78, Dublin City University and Association for Computational Linguistics, Dublin, Ireland, 2014.
26. Baktha, K. and B. K. Tripathy, “Investigation of Recurrent Neural Networks in the Field of Sentiment Analysis”, *2017 International Conference on Communication and Signal Processing (ICCSP)*, pp. 2047–2050, 2017.
27. Hochreiter, S. and J. Schmidhuber, “Long Short-Term Memory”, *Neural Computation*, Vol. 9, No. 8, pp. 1735–1780, 1997.
28. Sankar, H. and V. Subramaniaswamy, “Investigating Sentiment Analysis Using Machine Learning Approach”, *2017 International Conference on Intelligent Sustainable Systems (ICISS)*, pp. 87–92, 2017.
29. Ma, B., H. Yuan and Y. Wu, “Exploring Performance of Clustering Methods on Document Sentiment Analysis”, *Journal of Information Science*, Vol. 43, No. 1, pp. 54–74, 2017.
30. Al-Harbi, S. H. and V. J. Rayward-Smith, “Adapting K-Means for Supervised Clustering”, *Applied Intelligence*, Vol. 24, No. 3, pp. 219–226, 2006.
31. Li, N. and D. D. Wu, “Using Text Mining and Sentiment Analysis for Online Forums Hotspot Detection and Forecast”, *Decision Support Systems*, Vol. 48, No. 2, pp. 354–368, 2010.
32. Osgood, C. E., G. J. Suci and P. H. Tannenbaum, *The Measurement of Meaning*, 47, University of Illinois press, 1957.
33. Taboada, M., J. Brooke, M. Tofiloski, K. Voll and M. Stede, “Lexicon-Based Methods for Sentiment Analysis”, *Computational Linguistics*, Vol. 37, No. 2, pp. 267–307, 2011.

34. Turney, P. D., “Thumbs Up or Thumbs Down? Semantic Orientation Applied to Unsupervised Classification of Reviews”, *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, pp. 417–424, Philadelphia, USA, 2002.
35. Hatzivassiloglou, V. and K. McKeown, “Predicting the semantic orientation of adjectives”, *35th Annual Meeting of the Association for Computational Linguistics and 8th Conference of the European Chapter of the Association for Computational Linguistics*, pp. 174–181, Madrid, Spain, 1997.
36. Miller, G. A., R. Beckwith, C. Fellbaum, D. Gross and K. J. Miller, “Introduction to WordNet: An On-line Lexical Database*”, *International Journal of Lexicography*, Vol. 3, No. 4, pp. 235–244, 1990.
37. Baccianella, S., A. Esuli and F. Sebastiani, “SentiWordNet 3.0: An Enhanced Lexical Resource for Sentiment Analysis and Opinion Mining”, *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC’10)*, European Language Resources Association (ELRA), Valletta, Malta, 2010.
38. Cambria, E., Y. Li, F. Z. Xing, S. Poria and K. Kwok, “SenticNet 6: Ensemble Application of Symbolic and Subsymbolic AI for Sentiment Analysis”, *Proceedings of the 29th ACM International Conference on Information & Knowledge Management, CIKM ’20*, p. 105–114, Association for Computing Machinery, New York, NY, USA, 2020.
39. Gupta, I. and N. Joshi, “Enhanced Twitter Sentiment Analysis Using Hybrid Approach and by Accounting Local Contextual Semantic”, *Journal of Intelligent Systems*, Vol. 29, No. 1, 2019.
40. Liu, B., “Sentiment Analysis and Subjectivity”, *Handbook of Natural Language Processing, Second Edition*, Vol. 2, pp. 627–666, 2010.
41. Pontiki, M., D. Galanis, J. Pavlopoulos, H. Papageorgiou, I. Androutsopoulos

- and S. Manandhar, “SemEval-2014 Task 4: Aspect Based Sentiment Analysis”, *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, pp. 27–35, Association for Computational Linguistics, Dublin, Ireland, 2014.
42. Saeidi, M., G. Bouchard, M. Liakata and S. Riedel, “SentiHood: Targeted Aspect Based Sentiment Analysis Dataset for Urban Neighbourhoods”, *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pp. 1546–1556, The COLING 2016 Organizing Committee, Osaka, Japan, 2016.
 43. Dong, L., F. Wei, C. Tan, D. Tang, M. Zhou and K. Xu, “Adaptive Recursive Neural Network for Target-dependent Twitter Sentiment Classification”, *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 49–54, Association for Computational Linguistics, Baltimore, Maryland USA, 2014.
 44. Liu, R., Y. Shi, C. Ji and M. Jia, “A Survey of Sentiment Analysis Based on Transfer Learning”, *IEEE Access*, Vol. 7, pp. 85401–85412, 2019.
 45. Kireyev, K., L. Palen and K. Anderson, “Applications of Topics Models to Analysis of Disaster-Related Twitter Data”, *Neural Information Processing Systems (NIPS) Workshop on Applications for Topic Models: Text and Beyond*, Vol. 1, 2009.
 46. Alfina, I., D. Sigmawaty, F. Nurhidayati and A. N. Hidayanto, “Utilizing Hashtags for Sentiment Analysis of Tweets in The Political Domain”, *Proceedings of the 9th International Conference on Machine Learning and Computing, ICMLC 2017*, p. 43–47, Association for Computing Machinery, New York, NY, USA, 2017.
 47. Rill, S., D. Reinelt, J. Scheidt and R. V. Zicari, “PoliTwI: Early Detection of Emerging Political Topics on Twitter and the Impact on Concept-Level Sentiment Analysis”, *Knowledge-Based Systems*, Vol. 69, No. 1, pp. 24–33, 2014.
 48. Mikolov, T., K. Chen, G. Corrado and J. Dean, “Efficient Estimation of Word

- Representations in Vector Space”, *arXiv preprint arXiv:1301.3781*, 2013.
49. Pennington, J., R. Socher and C. D. Manning, “Glove: Global Vectors for Word Representation”, *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1532–1543, Doha, Qatar, 2014.
 50. Mikolov, T., E. Grave, P. Bojanowski, C. Puhersch and A. Joulin, “Advances in Pre-Training Distributed Word Representations”, *Proceedings of the International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan, 2018.
 51. Mishev, K., A. Gjorgjevikj, I. Vodenska, L. T. Chitkushev and D. Trajanov, “Evaluation of Sentiment Analysis in Finance: From Lexicons to Transformers”, *IEEE Access*, Vol. 8, pp. 131662–131682, 2020.
 52. Simonyan, K. and A. Zisserman, “Very Deep Convolutional Networks for Large-Scale Image Recognition”, *arXiv preprint arXiv:1409.1556*, 2014.
 53. He, K., X. Zhang, S. Ren and J. Sun, “Deep Residual Learning for Image Recognition”, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770–778, Las Vegas, NV, USA, 2016.
 54. Vaswani, A., N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser and I. Polosukhin, “Attention Is All You Need”, *Advances in Neural Information Processing Systems*, Vol. 30, 2017.
 55. Radford, A., J. Wu, R. Child, D. Luan, D. Amodei and I. Sutskever, “Language Models Are Unsupervised Multitask Learners”, *OpenAI Blog*, Vol. 1, No. 8, p. 9, 2019.
 56. Devlin, J., M.-W. Chang, K. Lee and 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*

- Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 4171–4186, Association for Computational Linguistics, Minneapolis, Minnesota USA, 2019.
57. Wu, Y., M. Schuster, Z. Chen, Q. Le, M. Norouzi, W. Macherey, M. Krikun, Y. Cao, Q. Gao, K. Macherey, J. Klingner, A. Shah, M. Johnson, X. Liu, u. Kaiser, S. Gouws, Y. Kato, T. Kudo, H. Kazawa and J. Dean, “Google’s Neural Machine Translation System: Bridging the Gap Between Human and Machine Translation”, *arXiv preprint arXiv:1609.08144*, 2016.
 58. Sennrich, R., B. Haddow and A. Birch, “Neural Machine Translation of Rare Words with Subword Units”, *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1715–1725, Berlin, Germany, 2016.
 59. Zhu, Y., R. Kiros, R. Zemel, R. Salakhutdinov, R. Urtasun, A. Torralba and S. Fidler, “Aligning Books and Movies: Towards Story-Like Visual Explanations by Watching Movies and Reading Books”, *Proceedings of the IEEE International Conference on Computer Vision*, pp. 19–27, Santiago, Chile, 2015.
 60. Wang, A., A. Singh, J. Michael, F. Hill, O. Levy and S. Bowman, “GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding”, *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pp. 353–355, Brussels, Belgium, 2018.
 61. Schweter, S., “BERTurk - BERT models for Turkish”, <https://zenodo.org/record/3770924>, accessed in January 2022.
 62. Aydın, C. R. and T. Güngör, “Sentiment Analysis in Turkish: Supervised, Semi-supervised, and Unsupervised Techniques”, *Natural Language Engineering*, Vol. 27, No. 4, pp. 455–483, 2021.
 63. Vural, A. G., B. B. Cambazoglu, P. Senkul and Z. O. Tokgoz, “A Framework for

- Sentiment Analysis in Turkish: Application to Polarity Detection of Movie Reviews in Turkish”, E. Gelenbe and R. Lent (Editors), *Computer and Information Sciences III*, pp. 437–445, Springer London, London, UK, 2013.
64. Türkmenoğlu, C. and A. C. Tantı, “Sentiment Analysis in Turkish Media”, *Workshop on Issues of Sentiment Discovery and Opinion Mining, International Conference on Machine Learning (ICML)*, Beijing, China, 2014.
 65. Akgül, E., C. Ertano and B. Diri, “Sentiment Analysis with Twitter”, *Pamukkale University Journal of Engineering Sciences*, Vol. 22, pp. 106–110, 2016.
 66. Ciftci, B. and M. S. Apaydin, “A Deep Learning Approach to Sentiment Analysis in Turkish”, *2018 International Conference on Artificial Intelligence and Data Processing (IDAP)*, pp. 1–5, 2018.
 67. Sun, C., L. Huang and X. Qiu, “Utilizing BERT for Aspect-Based Sentiment Analysis via Constructing Auxiliary Sentence”, *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 380–385, Minneapolis, Minnesota USA, 2019.
 68. Gao, Z., A. Feng, X. Song and X. Wu, “Target-Dependent Sentiment Classification with BERT”, *IEEE Access*, Vol. 7, pp. 154290–154299, 2019.
 69. Pontiki, M., D. Galanis, H. Papageorgiou, I. Androutsopoulos, S. Manandhar, A. Mohammad, M. Al-Ayyoub, Y. Zhao, B. Qin, O. De Clercq *et al.*, “SemEval-2016 Task 5: Aspect Based Sentiment Analysis”, *Proceedings of the 10th International Workshop on Semantic Evaluation*, pp. 19–30, 2016.
 70. Dong, L., F. Wei, C. Tan, D. Tang, M. Zhou and K. Xu, “Adaptive Recursive Neural Network for Target-Dependent Twitter Sentiment Classification”, *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (volume 2: Short papers)*, pp. 49–54, Baltimore, Maryland USA, 2014.

71. Acikalin, U. U., B. Bardak and M. Kutlu, “Turkish Sentiment Analysis Using BERT”, *2020 28th Signal Processing and Communications Applications Conference (SIU)*, pp. 1–4, 2020.
72. Celebi, A. and A. Özgür, “Segmenting Hashtags and Analyzing Their Grammatical Structure”, *Journal of the Association for Information Science and Technology*, Vol. 69, No. 5, pp. 675–686, 2018.
73. Soares, L. B., N. Fitzgerald, J. Ling and T. Kwiatkowski, “Matching the Blanks: Distributional Similarity for Relation Learning”, *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 2895–2905, Florence, Italy, 2019.
74. Hendrickx, I., S. N. Kim, Z. Kozareva, P. Nakov, D. Ó. Séaghdha, S. Padó, M. Pennacchiotti, L. Romano and S. Szpakowicz, “SemEval-2010 Task 8: Multi-Way Classification of Semantic Relations Between Pairs of Nominals”, *Proceedings of the 5th International Workshop on Semantic Evaluation*, pp. 33–38, Uppsala, Sweden, 2010.
75. Zeng, D., K. Liu, S. Lai, G. Zhou and J. Zhao, “Relation Classification via Convolutional Deep Neural Network”, *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pp. 2335–2344, Dublin, Ireland, 2014.
76. Zhang, Y., V. Zhong, D. Chen, G. Angeli and C. D. Manning, “Position-Aware Attention and Supervised Data Improve Slot Filling”, *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pp. 35–45, Copenhagen, Denmark, 2017.
77. Han, X., H. Zhu, P. Yu, Z. Wang, Y. Yao, Z. Liu and M. Sun, “FewRel: A Large-Scale Supervised Few-Shot Relation Classification Dataset with State-of-the-Art Evaluation”, *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pp. 4803–4809, Brussels, Belgium, 2018.

78. Wolf, T., L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz, J. Davison, S. Shleifer, P. von Platen, C. Ma, Y. Jernite, J. Plu, C. Xu, T. L. Scao, S. Gugger, M. Drame, Q. Lhoest and A. M. Rush, “Transformers: State-of-the-Art Natural Language Processing”, *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pp. 38–45, Online, 2020.
79. Dodge, J., G. Ilharco, R. Schwartz, A. Farhadi, H. Hajishirzi and N. Smith, “Fine-tuning Pretrained Language Models: Weight Initializations, Data Orders, and Early Stopping”, *arXiv preprint arXiv:2002.06305*, 2020.