EMOTION ANALYSIS ON TURKISH TWEETS

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

SİNEM DEMİRCİ

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
COMPUTER ENGINEERING

FEBRUARY 2014

Approval of the thesis:

EMOTION ANALYSIS ON TURKISH TWEETS

submitted by **SİNEM DEMİRCİ** in partial fulfillment of the requirements for the degree of **Master of Science in Computer Engineering Department, Middle East Technical University** by,

Prof. Dr. Canan Özgen
Dean, Graduate School of **Natural and Applied Sciences** ________________

Prof. Dr. Adnan Yazıcı
Head of Department, **Computer Engineering** ________________

Assoc. Prof. Dr. Pınar Karagöz
Supervisor, **Computer Engineering Department, METU** ________________

Prof. Dr. İsmail Hakkı Toroslu
Co-supervisor, **Computer Engineering Department, METU** ________________

Examining Committee Members:

Assoc. Prof. Dr. Halit Oğuztüzün
Computer Engineering Department, METU ________________

Assoc. Prof. Dr. Pınar Karagöz
Computer Engineering Department, METU ________________

Prof. Dr. İsmail Hakkı Toroslu
Computer Engineering Department, METU ________________

Assist. Prof. Dr. Tuğba Taşkaya Temizel
Information Systems, METU ________________

Dr. Ruket Çakıcı
Computer Engineering Department, METU ________________

Date: ________________

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last Name: SİNEM DEMİRCİ

Signature :

ABSTRACT

EMOTION ANALYSIS ON TURKISH TWEETS

Demirci, Sinem
M.S., Department of Computer Engineering
Supervisor : Assoc. Prof. Dr. Pınar Karagöz
Co-Supervisor : Prof. Dr. İsmail Hakkı Toroslu

February 2014, 54 pages

Automatically detecting emotions in micro-blogs is a new research area which gains importance with the rapid growth of the micro-blogs in the last few years. Mining emotions in micro-blogs has some practical uses which can improve human-computer interaction. As opposed to regular text used in text mining studies, micro-blog entries are short and not well-formed enough to process directly. Also there are some special usages, symbols and conveniences used in micro-blogs which may greatly influence the affect in the text. Therefore, in this thesis, a general framework which considers those deficiencies is suggested and a new data set of Turkish tweets for emotion analysis is constructed.

Keywords: Emotion Analysis, Text Mining, Sentiment Analysis

ÖZ

TÜRKÇE TWEET'LERDE DUYGU ANALİZİ

Demirci, Sinem
Yüksek Lisans, Bilgisayar Mühendisliği Bölümü
Tez Yöneticisi : Doç. Dr. Pınar Karagöz
Ortak Tez Yöneticisi : Prof. Dr. İsmail Hakkı Toroslu

Şubat 2014, 54 sayfa

Mikro-bloglarda duyguların otomatik olarak saptanması, mikro-blogların geçtiğimiz birkaç yıl içerisindeki hızlı büyümesiyle birlikte önem kazanan yeni bir araştırma alanıdır. Mikro-bloglardaki duygu madenciliği insan-bilgisayar etkileşimini geliştirebilecek bazı pratik kullanım alanlarına sahiptir. Metin madenciliği çalışmalarındaki normal metinlere karşın, mikro-blog girdileri kısadır ve doğrudan işlemlenecek kadar düzgün formatlı değildir. Ayrıca mikro-bloglarda kullanılan, metindeki duyguyu büyük ölçüde etkileyebilen özel kullanımlar, semboller ve kolaylıklar vardır. Bu yüzden, bu tezde, bu eksiklikleri göz önünde bulunduran genel bir çerçeve önerilip duygu analizi için yeni bir Türkçe tweet veri seti oluşturulmuştur.

Anahtar Kelimeler: Duygu Analizi, Metin Madenciliği, Düşünce Çözümleme

To my family

Canan Demirci, Fatma Demirci, Cemal Demirci

ACKNOWLEDGMENTS

I would like to thank to my supervisor Assoc. Prof. Dr. Pınar Karagöz for her guidance, support and positive attitude towards me. I would like to thank to my co-supervisor Prof. Dr. İsmail Hakkı Toroslu for his ideas regarding my thesis study.

I would like to thank to my precious family for everything they have done for me. I would not be here if it were not for their love, devotion and sympathy.

I would like to thank to my friend Kerem Hadımlı for cheering me up during the harsh times also for allowing me to benefit from his knowledge and opinions.

I would like to thank to all the people from the office A-301 in Computer Engineering Department in METU for their friendship and making everyday life more joyful during my assistantship.

I would like to thank to my friends Erdal Sivri, Ayşenur Durgut, Işıl Özge Pekel, Tolga Akın, Çiğdem Avcı Salma, Gülseren Gönül, Alaettin Zubaroğlu, Murat Ongan, Gökdeniz Karadağ, İsmail Can Coşkuner for their friendship.

TABLE OF CONTENTS

LIST OF TABLES

TABLES

LIST OF FIGURES

FIGURES

CHAPTER 1

INTRODUCTION

Emotions determine the way humans interact with other humans and the social environment in which they exist. Therefore, they constitute important part in human life. Identifying the polarity of the text has been an active research area for a long time. However, not only the detection of the polarity of text but also the detection of emotion itself embodied in the text have gained importance over time with the aim of improvement in human-computer interaction applications. The aim in this interaction is to understand the humans and respond to the needs of them. With the advent of social networking websites, people began to share their thoughts and emotions with rest of their social group.

Emotion is the manifestation of the subjective experience influenced by a person's internal state and external stimulants. Emotions are widely studied in the fields of psychology, sociology, physiology and medicine. Paul Ekman, a psychologist who is well-known especially for his studies about emotions, identified six basic emotions. These emotions are joy, sadness, anger, fear, disgust and surprise [10]. Basic emotions are universally recognizable even though they have subjective aspects. Other emotions are categorized as the combination of these basic ones.

There are numerous studies on emotion analysis on textual data sources, especially in English language, such as Kozareva et al. [17], Mohammad [22] and Chaffar and Inkpen [5]. On the other hand, there are only a few studies about this topic on Turkish language such as Boynukalın [4]. On Turkish, there are more studies about sentiment analysis than emotion analysis as it is a more established area of research [11, 15].

The Turkish language is an agglutinative language. With the use of derivational suffixes, stem of a word may be converted into a totally different part of speech type such as from noun to verb. These derivations can be applied consecutively more than once [25]. Since each derivational suffix has the possibility of changing the meaning of the word, to obtain the real meaning of a word each derivational suffix must be examined.

Even though the previous research focused on formal data sources such as newspaper headlines and surveys, lately research on informal data sources such as instant messaging [23], blog posts [30] and Twitter [21, 26, 30, 31] is trending. Twitter is a social micro-blogging service which provides users with the option to post and read messages in real time, called tweets. People share their opinions, daily life events and emotions on Twitter. Although there exist many more micro-blogging environments, Twitter is the most popular one. Sheer volume of user generated content make Twitter a favorable domain. Moreover, uniformity of tweets make them efficient entities to process for emotion detection task.

In this thesis, a framework for emotion analysis on informal Turkish text is proposed. It is aimed to classify the six basic emotions from Twitter data focusing on Turkish language. The problems originating from Turkish language and Twitter are aimed to be solved. Alternative methods for components of this framework are explored and their performance results are compared.

The rest of this thesis is organized as follows:

- In Chapter 2, a survey about the related studies is given. The advantages and the drawbacks of the methods applied are summarized.
- In Chapter 3, background information about the concepts and algorithms utilized are given.
- In Chapter 4, details of the proposed method are explained.
- In Chapter 5, experimental results are discussed. The effect of parameter selection is illustrated and performance measurement is shown.
- In Chapter 6, a conclusion is made and possible research issues emerged during

this thesis are presented.

CHAPTER 2

LITERATURE SURVEY

Emotion detection task is applied to different domains including news headlines, blog contents, fairy tales and tweets. Some rule-based methods are devised considering special features of the related domain. Classification algorithms are applied in a supervised learning setting. Feature selection and reduction techniques are utilized.

2.1 Studies on Emotion Analysis on English Texts

Kozareva et al. [17], classified news headlines using 6 basic emotion classes identified by Ekman. They used web search engine results to measure PMI (pointwise mutual information). It is stated that the more the concepts co-occur, the more closely related they are and PMI is a method to get the numeric value of this relevancy. They used hit counts from 3 different web search engines to calculate co-occurrence of the words of the headline and emotion class labels. Results obtained from different search engines are averaged and mapped between 0 and 100. Higher scored emotion class is determined to be the result. This approach has some drawbacks. In the study of Bollegala et al. [3], the use of web search engine hit counts itself is expressed to be unreliable. The first reason is that web search engines count as a hit even if the query words are not related semantically as long as they occur in the same document. The second reason is that words having more than one sense and noise in the web may lead to wrong inferences.

Yang et al. [31] used 4 emotion classes in the study Taiwan Yahoo! Kimo blog posts classification. Positive and negative emotion classes are divided into joy, happiness (positive); and sadness, fear (negative). SVM and CRF (conditional random fields) classifiers are used. CRF is stated to be able to learn the transitions between sentences in a web blog, therefore, it is called context based classification. As a result it outperforms SVM. However, considering emotional sense in the previous sentence for the current one, is not suitable to Twitter domain. Tweets are short text that do not contain more than a few sentences at most.

Neviarouskaya et al. [23] detected 10 classes of emotions and 5 classes of communicative functions in an instant messaging environment. In this study, a rule-based approach taking common abbreviations, emoticons and WordNet Affect lexicon into account is used. For each emotional word a feature vector is constructed showing to which degree the word is related with a class. Therefore, it is a fuzzy approach.

It is shown in [22] with the word-emotion lexicon addition to n-gram approach, remarkable improvements are achieved.

Chaffar and Inkpen [5], applied supervised machine learning techniques to a heterogeneous data set consisting of news headlines, stories and blog posts in 6 basic emotion detection task.

In the study of Wang et al. [30] emotion labeled Twitter data is collected for 7 emotion classes (joy, sadness, anger, love, fear thankfulness and surprise). The initial list of emotion words is expanded with the lexical variants of the emotion hashtags. For example, “surprising” and “surprised” are added to “surprise” class. Emotion words associated with other concepts or domains are excluded from the list to decrease ambiguity. 93.16% of the tweets of the resulting data set is shown to be relevant to the corresponding emotion. The data set is trained with different combinations of n-gram features, part-of-speech tagging and several predefined lexical resources such as WordNet Affect. The aim of this study is to construct automatically annotated data set in short amount of time and measure the performance of classification for different size of data sets. It is shown that by increasing the size of training data from 1000 to 2 million, 22.16% accuracy gain is obtained.

Qadir and Riloff [26] suggested a bootstrapped approach to collect automatically annotated data set for emotion detection task on Twitter. A small list containing 5 seed emotion hashtags for each emotion class is used to collect data from Twitter. A logistic regression classifier is trained with resulting data set using n-gram features. A new unlabeled test set collected from Twitter is classified and the most confident 10 new hashtags are determined. The new hashtags are added to the seed hashtag list and this process is applied iteratively resulting a list of learned hashtags.

Go et al. [12] applied supervised machine learning methods to data retrieved from Twitter and classified the data as either positive or negative according to the sentiment available in tweet. Firstly, emoticons are used as noisy labels for classification ground truth and stripped from the tweets. As features, unigrams, bigrams, unigram combined with bigrams and unigrams combined with POS information are used. As a baseline for comparison, a web service performing sentiment analysis called Twittratr is used. Naïve Bayes, Maximum Entropy and SVM classifiers are compared.

Danisman and Alpkocak [9] compared the performances of Vector Space Model, Naïve Bayes and SVM classifiers using ISEAR data set [28] for 5 emotion classes namely anger, disgust, fear, joy and sadness. Training set is enriched with WordNet Affect and WPARD (Wisconsin Perceptual Attribute Rating Database) data sources. Stop word removal and stemming are applied and tf-idf is selected as feature weighting. 70.2% overall classification accuracy is reached.

Kouloumpis et al. [16] discusses about the difficulties related with identifying the labels for the training data set. They propose a method for constructing the training labels using hashtags in the tweets. They extracted tweets with hashtags which appear more than 1000 times in Edinburgh Twitter corpus and manually grouped them according to their sentiment as three groups (positive, negative, neutral hashtags). As well as this hashtag annotated data set, they also make use of Go et al.'s emoticon annotated data set. They used the emoticon data set to enhance their first data set by appending 19000 tweets from the emoticon data set.

2.2 Studies on Emotion Analysis on Turkish Texts

Boynukalın [4] classified Turkish text using the translation of ISEAR data set and a manually annotated fairy tale data set in the MSc thesis. Except from the emotion classes, emotion levels are tried to be detected. Several combinations of different n-gram features are used. Weighted log likelihood algorithm [24] is utilized to score features and identify the most significant ones.

Kaya et al.[15] applied supervised classification algorithms on Turkish news columns for sentiment classes positive and negative. Except from the SVM, Maximum Entropy and Naïve Bayes classifiers, n-gram based character Language Model is utilized. This language model uses characters instead words as units. Their reasoning is that statistical methods may not yield promising results since Turkish is a morhologically rich language.

Eroğul [11] generated a data set from a Turkish movie review site. The reviews were tagged by their writers with one of the positive, negative or neutral icons. In the generated data set, the review's text and its icon were associated with each other to produce a sentiment-labeled data item. From another movie review site, where users give score to the movie they reviewed, a polarity data set was constructed. Using Zemberek tool as morphological parser, combinations of n-grams and POS information are used for classification task. Regression and one-vs-all techniques are utilized to predict the scores for the polarity labeled data set.

CHAPTER 3

BACKGROUND

3.1 Twitter

Twitter is a widely used worldwide micro-blogging website connecting people with whom they interested in. People share news, opinions, information, the latest news about themselves, etc. Share mechanism is realized via 140-characters long messages called tweets. Visibility of a tweet can be restricted optionally or a tweet can be shared with all users. A tweet can be shared or re-shared with others in real time. The re-sharing act of someone else's tweet is called retweeting. People also have the opportunity to add URLs, certain media such as pictures and videos to their tweets or mention other users in their tweets. Hashtags, words or phrases written with prefix "#", are used to give some contextual cue about the tweet such as main topic, dominant idea or feeling.

Twitter does not only connects people of local, small and closed groups. It also allows connection formation between people and business organizations, government representatives and major news sources. Therefore, it becomes a platform in which users are promoted to share their feedback, thoughts and emotions about products, services, agency and management policies, politics, etc. This makes Twitter a valuable source for sentiment analysis, opinion mining and emotion analysis applications.

3.2 Turkish Morphology

Turkish is an agglutinative language in which new words are generated through addition of derivational suffixes to existing word stems. Theoretically it is possible to generate infinitely many words from a stem by consecutively appending these suffixes. Inflectional suffixes on the other hand does not produce new words. They modify the word so that it indicates tense, person, gender, number, etc.

Multiple morphological analyzers exist for Turkish such as Zemberek [1] and TRmorph [7]. In this study Zemberek is chosen for its simplicity.

3.3 Classification Algorithms

Grouping the instances of a data set has long been the focus of machine learning. When the categories to which instances are aimed to be mapped are known in advance, the problem is regarded as a classification problem.

At the heart of the classification problem, there are test data set, training data set and classifier. Training data set is composed of instances whose class information is available in advance. This information is not available for test data set on the contrary. The classifier is the algorithm modeling the training set so that it manages to infer the class labels of the instances in the test data set. In the modeling process, classifier make use of observable features of the instances.

There are several kinds of classification algorithms. Some differences stem from the feature space modeling in the inference process. While some classifiers use linear predictor functions, others use non-linear ones. There are binary and multi-class classifiers. Binary classification task only involves two classes. Some of the multi-class classifiers are derived from the combinations of binary classifiers whereas some are inherently multi-class. Some of the classifiers produce confidence values along with possible class labels, others do not.

In this thesis, several kinds of classification algorithms are utilized. Bayesian classifier is selected since it gives the probability of the instance to be in a class. SVM is selected to utilize both linear and non-linear models. k-NN, which uses similarities of the instances, is chosen as an inherently multi-class classifier.

3.3.1 Naïve Bayes Classifier

Naïve Bayes classifier is a statistical method based on Bayesian theorem. To calculate the posterior probability of an instance to be in a class, prior probability distributions and likelihood information are used. In an n-dimensional space and m different classes, probabilities of membership of an item $x = (f_1, f_2, ..., f_n)$ are calculated as in Equation 3.1.

$$P(C_i|x) = \frac{P(C_i)P(x|C_i)}{P(x)} \tag{3.1}$$

which can be rewritten as follows:

$$P(C_i|x) = \frac{P(C_i)P(f_1, f_2, ..., f_n|C_i)}{P(f_1, f_2, ..., f_n)} \tag{3.2}$$

It is assumed that the features of x are independent and identically distributed which leads to Equation 3.3.

$$P(C_i|x) = \frac{P(C_i)P(f_1|C_i)P(f_2|C_i)...P(f_n|C_i)}{P(f_1, f_2, ..., f_n)} \tag{3.3}$$

The class with the highest probability is designated to be the class label.

$$\text{Class} = \arg\max_i \big(P(C_i|x)\big) \tag{3.4}$$

Since the denominator is a constant, it is disregarded and the equation is simplified as in Equation 3.5.

$$P(C_i)P(C_i|x) \propto P(C_i)P(f_1|C_i)P(f_2|C_i)...P(f_n|C_i) \tag{3.5}$$

3.3.2 Complement Naïve Bayes Classifier

Complement Naïve Bayes is proposed in order to overcome some deficiencies existing in Naïve Bayes classifiers [27]. A Naïve Bayes classifier is prone to falsely predict the class labels if data set is skewed. The decision boundary is biased when some classes have different number of samples in them than the other classes. Other problem addressed is about the independent feature assumption of Naïve Bayes classifiers. Contributions of dependent features are calculated separately and summed. This situation leads to favoring the classes with dependent variables over the others.

Complement Naïve Bayes is a one-vs-all-but-one method. It constructs a model to get the posterior probability of a sample to be in the set of combination of other classes except the current one. The complement class of the lowest probability one is selected to be the class label.

Naïve Bayes formula is updated by applying the log function as in Equation 3.6. Since logarithm is a monotonic function, the result of the maximization will not be affected.

$$\text{Class} = \arg\max_i \left(\log P(C_i) + \sum_{j=1}^{n} \log P(f_j|C_i) \right) \tag{3.6}$$

The aim of Complement Naïve Bayes classifier is to select the class whose complement does not match well. Therefore, instead of summing apriori with the conditional probability on the current class, subtraction operation is introduced to the equation with the conditional probability on complementary classes. The new equation can be formulized as,

$$\text{Class} = \arg\max_i \left(\log P(C_i) - \sum_{j=1}^{n} \log P(f_j|C_{i'}) \right) \tag{3.7}$$

Note that the equation obtained yields a non-generative formula.

3.3.3 Support Vector Machines

Support vector machines (SVMs)[8] are supervised learning algorithms used in classification and regression tasks. SVM is used as a binary classifier to separate instances with an n-dimensional hyperplane. To achieve this, SVM maps data set into higher or infinite dimensional space using a kernel function. There are several kernel functions some of which are linear, polynomial, sigmoid functions and radial basis function. Aim of SVM is to find an optimal hyperplane which minimizes the error in classification. As a result of SVM, the unique hyperplane which separates the instances of the two classes with a maximum margin is found. For an n-dimensional space, separating hyperplane has n-1 dimensions. Instances from each class that constrain this optimal hyperplane are called support vectors.

While trying to find the optimal hyperplane, SVM allows error by using a positive valued cost variable C in order to avoid over-fitting problem. If SVM forces all instances to be in the correct class, then the new data can be misclassified. Too much tolerance on the other hand, leads to training errors. In Figure 3.1, a separating hyperplane and support vectors are shown.

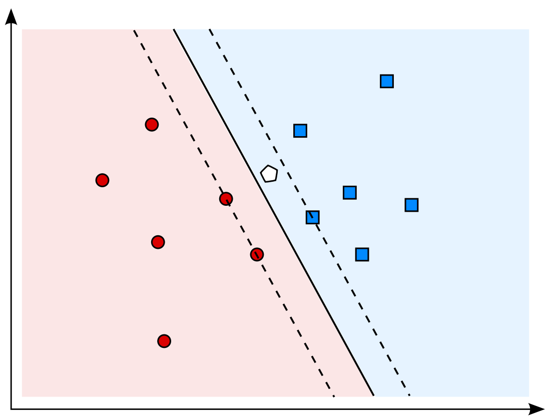

Figure 3.1: SVM example. The dashed lines show the largest margin between the two classes. The solid line shows the separating hyperplane

Some of the kernel functions are given below in which γ, r, and d are parameters of the related kernel and $\mathbf{x}_i$ values are instances from input domain [14]:

- Linear Kernel: $K(\mathbf{x}_i, \mathbf{x}_j) = \mathbf{x}_i^T \mathbf{x}_j$

- Polynomial Kernel: $K(\mathbf{x}_i, \mathbf{x}_j) = (\gamma \mathbf{x}_i^T \mathbf{x}_j)^d, \gamma > 0$

- Radial Basis Kernel (RBF): $K(\mathbf{x}_i, \mathbf{x}_j) = \exp(-\gamma \|\mathbf{x}_i - \mathbf{x}_j\|^2), \gamma > 0$

- Sigmoid Kernel: $K(\mathbf{x}_i, \mathbf{x}_j) = \tanh(\gamma \mathbf{x}_i^T \mathbf{x}_j + r)$

In order to get optimum results from SVM, correct values for kernel parameters and C should be selected. One of the mostly used methods to achieve this is to try out different values of parameter combinations considering parameter boundaries, which is called grid search and cross validate the classification results. k-fold cross validation is a statistical analysis in which data set is divided into k partitions and each time different $k-1$ partitions are used as training set and the other partition is used to measure classification accuracy. This is done to find a model well generalizing to different data and overcome the problem of over-fitting. Problem with this approach is that it is very time consuming. To speed up the process, parameter selection step size is increased and finer-grained parameter search is applied only on the vicinity of the better results obtained from the first search.

For different kinds of data sets different kinds of kernel functions are chosen. If it is a high-dimensional data set, linear kernel, which does not map input space into higher dimensional one, can be used. Non-linear kernel mapping does not provide performance gain over linear one in such cases. [14] In the text classification task, since data is very high-dimensional, linear kernel mapping is preferable.

SVM is a binary classifier but with some modifications it can be used in multiclass cases. There are two alternatives, one-versus-all and one-versus-one. In the first approach, instances are tested to be in one class or in the other classes. There will be m binary classification tasks in total, where m is the number of classes. Each instance is determined to be in the class with highest output score. In the second approach each instance is tested with every possible binary classifier pairs for each class. There will be $m(m-1)/2$ binary classification tasks. Each instance is assigned to the class for which its output score sum is maximum.

3.3.4 K-Nearest Neighbors Algorithm

K-nearest neighbors algorithm is a basic machine learning algorithm used for classification and regression. This classification method requires no training. A similarity metric is defined to compare instances. When a new instance comes, its k nearest neighbors according to the similarity metric are selected. The new instance to be classified is simply assigned to the class of the majority among these k nearest neighbors.

In Figure 3.2, a training data set with two data classes can be seen. The most similar 5 instances are selected to the unknown test instance. Since the most of the instances are from the blue class, new instance is determined to be in blue class.

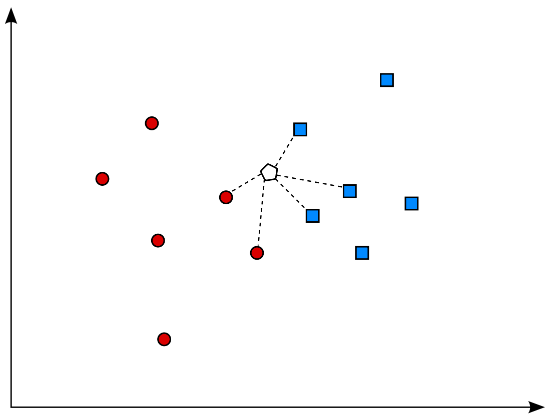

Figure 3.2: k-nn with $k = 5$. The unknown item is assigned to the blue class.

In this algorithm the selection of k is crucial. If k is selected too small, the classification results prone to noise. On the other hand, is k is selected too large, local variations are disregarded and global knowledge will dominate results. To select the most efficient k value, cross validation can be utilized.

3.4 Feature Selection Methods

Feature selection methods are means of selecting a subset of features from the original feature set. These methods aim to reduce the feature space by eliminating indifferent and redundant features as much as possible.

3.4.1 Information Gain

Information Gain [18] is used to measure the significance of a feature. When a feature is excluded from the data set, the resulting training matrix and the original training matrix are used to calculate the information loss caused by the removal of that feature. The feature which leads to the highest loss is considered to be the most significant feature.

3.4.2 WordNet Affect

WordNet [20] is a lexical database of English words organised as groups of synonym sets which are called synsets. WordNet Affect [29] is derived from WordNet Domains [2] which is an extension to WordNet with the addition of domain knowledge. WordNet Affect contains a subset of synsets that are related with affective concepts. These synsets contain nouns, verbs, adjectives and adverbs that have affective meaning. In this synset list, there are some affective labels called a-labels. There are different categories of a-labels such as emotion, mood, cognitive state, physical state, emotion eliciting situation, attitude, and others.

3.5 Tools

Various tools are utilized in the implementation of this thesis. In this section, these tools are described.

3.5.1 Zemberek

Zemberek [1] is an open source natural language processing tool for Turkic languages. It is implemented using Java programming language. Zemberek provides basic morphological parser operations including spell checking, stemming, parsing, mistyped word correction and word suggestion.

Zemberek returns a list of possible morphological analyses for a given input word. There is no strict order among the list items. Each analysis is composed of the stem, the part of speech information of the stem and the list of ordered morphemes. An example analysis of a sample word is given in Figure 3.3.

```
yapılacak:
[ Kok:yap, Tip:FIIL | Ekler:FIIL_KOK, FIIL_EDILGEN_IL, FIIL_DONUSUM_ECEK]
[ Kok:yap, Tip:FIIL | Ekler:FIIL_KOK, FIIL_EDILGEN_IL, FIIL_GELECEKZAMAN_ECEK]
```

Figure 3.3: Example Zemberek Analysis

Zemberek can correct simple spelling mistakes made in the stem and the morphemes. It can correct at most 1 letter insertion, deletion, change or swapped letters. When Zemberek is queried to correct a word, it recommends a list of grammatically correct words which are similar to fed input.

3.5.2 LIBSVM

LIBSVM [6] is a widely used Support Vector Machines library and it includes implementations for regression and classification tasks. Parameter selection, training and testing steps are provided by the library for SVM classification. Several different kernel functions are also available.

LIBSVM can be used for both binary and multiclass classifications. Multiclass version uses one-vs-one approach and employs $n * (n-1)/2$ binary classifiers where n is the total number of classes. Majority vote is applied in decision-making.

The default kernel function in LIBSVM is the radial basis function and it needs two different parameters which are cost C and γ. While C value affects the rigidness of the margin in training step, γ affects the separating hyperplane shape. As paramaters play an essential role in the performance of the classifier, they should be selected properly. The grid search script which is distributed with LIBSVM searches over a grid of C and γ values on the training set using 5-fold cross validation to obtain the best parameters. For the multiclass case, the script assumes same pair of C and γ values for all binary classifiers and returns a single result for each parameter.

The parameter values obtained from grid search script is used to construct a model from the training set using the LIBSVM training tool. The class labels of the elements in the test set is predicted using the LIBSVM prediction tool.

3.5.3 WEKA

WEKA [13] is a widely used open source machine learning algorithms tool. It provides various algorithms for data preprocessing, feature selection, clustering ,classification, regression and data visualization. WEKA is developed using Java programming language and can be easily integrated to other java applications. It can also be easily extended with new algorithms. The stand-alone graphical user interface of WEKA facilitates experimentation.

WEKA uses Attribute-Relation File Format (ARFF). It is a simple ASCII text file format. It supports both numeric and nominal attributes.

CHAPTER 4

PROPOSED METHOD FOR EMOTION ANALYSIS ON TURKISH TWEETS

Emotion detection task on micro-blog entries can be considered as a classification problem. In classification problem, a set of categories, a set of instances with membership information and a set of instances without membership information exist. Group labels of the instances with unknown categories are aimed to be discovered based on the known set of labeled instances called training set. In this study, each emotion becomes the group label, annotated data sets become the training sets and new tweets to categorize become the test set.

Micro-blog entries are much shorter in length than the regular text data and they have some extra properties to be considered separately such as hashtags in tweets. There exist several studies for English attaining promising results. However, unlike English, a morphologically rich language like Turkish needs further strategies to achieve higher performance than applying basic text mining techniques.

The steps of the proposed method can be summarized as follows:

Data Collection: Data is collected from Twitter since there are no suitable data sets available for emotion analysis in Turkish. A list of words and pharases is compiled for each emotional class. Keyword based search is applied to construct a data set with ready to use class labels.

Preprocessing: Properties which do not contain possible emotional content such as links and user names are removed. Words forming hashtags are extracted. Morphological analysis is performed. Some rule-based methods are applied to correct mistyped words.

Feature Vector Construction and Feature Selection: Different combinations of n-grams, part of speech tagging, emoticons and punctuation marks is tried out with term count and tf-idf statistics. Information gain method is applied to select the most significant features for classification task. A constructed Turkish emotional word lexicon is used to filter out non-emotional content from tweets and hence constitutes a feature selection mechanism.

Classification: Several supervised machine learning algorithms are applied to classify the data.

Basic steps of the method are shown in Figure 4.1.

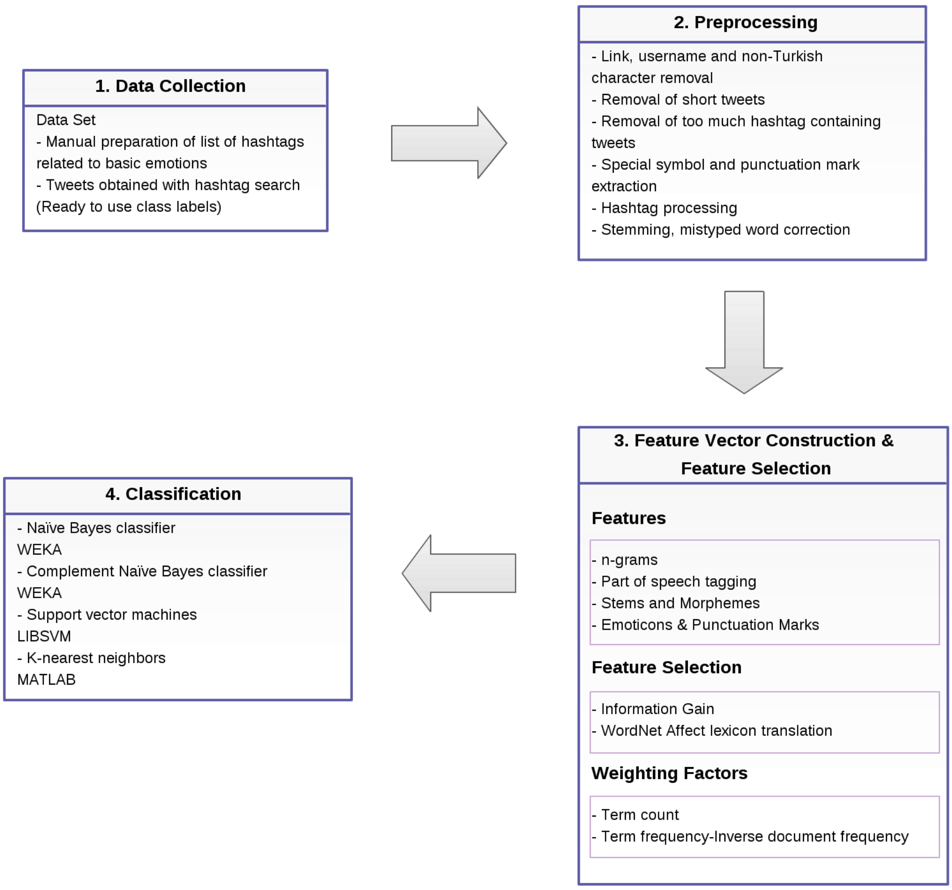

Figure 4.1: Steps of the Method

4.1 Data Collection

Emotion analysis task is a newly emerged field especially for Turkish language. Since there are no available data sets for us to use, we need to gather data. Using two different approaches, data collection process is carried out. Two different data sets are generated with these approaches.

In the study of Mohammad [21], Twitter search mechanism for hashtags is used and some English tweets including one of the 6 basic emotion hashtag words are collected. Those hashtag search words are anger, disgust, fear, joy, sadness and surprise. In this thesis, a similar approach is used. Turkish version of this list is constructed. The list is not only composed of this six basic emotion hashtag words. It is expanded to include synonyms as well and the final list is constructed with the addition of the verb and adjective forms of the basic emotions hashtag words. Moreover, some phrases including these emotion words are added. These words and phrases and their translations are given in Table 4.1 through Table 4.6. This data set contains self-tagged tweets and can be used in classification. In order to obtain the tweets, Twitter streaming API[1] is used.

Table 4.1: Anger Class Hashtag List and Translations

Hashtag	Translation
#kızdım	I am mad
#kızgınım	I am mad
#sinirlendim	I am angry
#sinirbozucu	frustrating
#öfkeliyim	I am angry
#çokkızıyorum	I get very angry
#kızıyorum	I am angry
#sinirleniyorum	I get angry
#öfke	anger
#sinir	anger

[1] https://dev.twitter.com/docs/streaming-apis

Table 4.2: Disgust Class Hashtag List and Translations

Hashtag	Translation
#iğrenç	disgusting
#iğrendim	I am disgusted
#iğreniyorum	I am disgusted
#iğrençtir	it is disgusting
#iğrençti	it was disgusting
#iğrençsiniz	you are disgusting
#iğrençsin	you are disgusting
#iğrenç	disgusting
#ıyy	exclamation expression for disgust
#ıyk	exclamation expression for disgust
#tiksinç	disgusting
#tiksindim	I am disgusted
#tiksindirici	disgusting
#tiksiniyorum	I am disgusted
#tiksinmek	to be disgusted

4.1.1 Data Set Description

In order to build a data set, initially a pool of tweets is collected with hashtag keyword search using Twitter Streaming API as mentioned before. After an initial preprocessing step, details of which are explained in Section 4.2, an equal number of tweets for each emotion class are randomly selected from this larger pool of tweets to form a balanced data set. Data distributions are shown and information about the content of tweets is given in Table 4.7 for each class separately. The table contains average hashtag count and word count per tweet. The words are not checked grammatically; each group of letters that does not contain any whitespace character is considered as a word. There are 1000 tweets in each class adding up to 6000 tweets in total for the whole data set.

4.2 Preprocessing

Twitter data is formatted to a certain extent. Tweets are 140 characters long, may or may not include URL, pictures or other media. These are removed from the tweets.

Table 4.3: Fear Class Hashtag List and Translations

Hashtag	Translation
#korku	fear
#korktum	I was feared
#korkunç	scary
#korkuyorum	I fear
#korkunç	scary
#ürkütücü	scary
#kaygılıyım	I am concerned
#kaygı	concern
#endişeliyim	I am worried
#endişe	worry
#tırstım	I am chickened out
#tırsıyorum	I am chickened out

Table 4.4: Joy Class Hashtag List and Translations

Hashtag	Translation
#mutlu	happy
#mutluluk	happiness
#mutluyum	I am happy
#mutluyumçünkü	I am happy because
#sevinçliyim	I am joyful
#neşeliyim	I am joyful

For each tweet, characters that are not in Turkish alphabet or in punctuation mark list are removed. In Twitter, a user may mention another user by using the format "@<username>" in a tweet. These are also removed from the tweet itself.

Some punctuation marks and some special character sequences (such as emoticons) are extracted so they could be passed to the feature extraction phase. A more detailed explanation can be found in Section 4.3.1, Feature Vector Construction.

As mentioned before, the data set is composed of self-tagged tweets. In this self-tagged tweets class labels are derived from the predefined list of hashtags existing inside the tweet. These hashtags are stripped off from the tweet to prevent bias in classification.

Table 4.5: Sadness Class Hashtag List and Translations

Hashtag	Translation
#üzgün	sad
#üzgünüm	I am sad
#üzüntü	sadness
#hüzün	sadness
#hüzünlü	sad
#mutsuzum	I am unhappy
#mutsuzumçünkü	I am unhappy because
#mutsuzluk	unhappiness

Table 4.6: Surpised Class Hashtag List and Translations

Hashtag	Translation
#şaşırdım	I was surprised
#şaşırtıcı	surprising
#şaşkınım	I am surprised
#oha, #ohaa, #ohaaa	exclamation expression for surprise
#yuh, #yuhh	exclamation expression for surprise
#çüş	exclamation expression for surprise

4.2.1 Filtering Tweets

A tweet which is originally posted by a user may be reposted by other users. Those tweets, which are called retweets, starts with the prefix "RT @<username>" indicating they are not directly written by the account owner instead they are written by the user "username". Therefore, retweets are disregarded and removed from the data set.

After initial preprocessing and retweet removal, if a tweet contains less than 3 words, or contains more than 3 hashtags, that tweet is not processed further and also removed from the data set. If the tweet contains less than 3 words, then it is very short and it is not expected to contain much information. On the other side, too many hashtags pose a similar problem. The aim of the users adding hashtags to tweets is to give a focus point about their intention. However, when there are many hashtags, the tweet focuses on many things at once thus emotions may be scattered, containing little information.

Another problem are the tweets that are almost identical. If a tweet resembles another

Table 4.7: Data Set Description

Class Label	Avg. Hashtag Count	Avg. Word Count	Tweet Count
Anger	0.29	12.14	1000
Fear	0.25	11.26	1000
Disgust	0.18	10.17	1000
Joy	1.38	6.44	1000
Sadness	0.30	8.32	1000
Surprised	0.28	10.17	1000
All	0.45	9.75	6000

tweet in the data set, that tweet is not added to the data set. When a new tweet is to be added to the data set, its Levenshtein distance[19] with all the tweets are calculated. If its edit distance with any of the tweets is greater than 90% of its length, then the tweet is discarded.

4.2.2 Stop Word Removal

There are some Turkish words whose presence may not contribute to the emotion analysis task since they do not have any emotional content. Therefore, removing them from the tweets may increase the system performance. Such words are gathered in order to form a stop word list.

A static stop word list may not be sufficient for Turkish language because of its agglutinative nature. Words can be changed with the addition of suffixes. Therefore, some types of words are taken into consideration as well as items of a collected stop word list.

Pronouns and textual numbers are regarded as stop words. After preprocessing step, they are cleaned out from the tweets. Zemberek morphological analyzer is used to detect the types of the words, including textual numbers. A hand-crafted list of conjunctions and prepositions along with some other words is utilized. The list is shown in Appendix A.

4.2.3 Mistyped Word Correction

Zemberek provides correction up to 1 character for stem and 1 character for morphemes. However, this correction may not be sufficient for the words in tweets. People tend to make many writing errors and sometimes write some words intentionally distorted. For example, people use “güzeeeeel” to express “güzel” (beautiful) with repeated ‘e’ character and this usage adds emphasis on the word. Zemberek fails to correct such cases. Since Turkish language does not have words where there are more than same 2 letters that are adjacent, such cases are replaced with one letter and then the word is fed to Zemberek.

In addition to this, Turkish conjunction words “de” and “da” that should be written separately from the preceding word, often written jointly. Words ending with “de”, “da”, “te” and “ta” are checked to detect such cases. If Zemberek fails to process the word, the string suffix is trimmed and the new word is checked with Zemberek again.

4.2.4 Morphological Analysis for Feature Enhancement

Since Turkish is an agglutinative language, morphemes may contain important information and stems by themselves may not be sufficient to classify tweets. Morphemes affect the meaning of the original stems vastly.

As using stems by themselves may miss important information, information obtained from morphemes are used to modify the tweet. Modification is done by injecting new keywords or modifying the stems. This ensures morpheme information is not lost.

In this section, methods for enhancing features with information obtained from morphemes are discussed.

4.2.4.1 Negation Handling

Given a word within the tweet, stem of the word is extracted. If the stem contains one of the Turkish negation suffixes (FIIL_OLUMSUZ_ME: gel*me* (*do not come*), FIIL_OLUMSUZ_SIZIN: dinlenmek*sizin* (*without resting*), ISIM_YOKLUK_SIZ:

keyif*siz* (*indisposed*)), that stem is appended a "_" character at the beginning. The original word in the tweet is replaced with this modified stem.

In Turkish the word "değil" is used to negate the word that comes before it (gitmiş *değil* (*not gone*)). Such cases are also handled and the word "değil" is removed from the tweet, appending a "_" character to the previous word's stem.

Note that double-negations need to be handled properly as positive, such as "keyif*siz değil*" (*not indisposed*).

4.2.4.2 Special Morpheme Handling

In Turkish, some morphemes give the word a strong emotional meaning. By examining these morphemes, they can be used to enhance feature vectors, which in turn might increase classification performance. The feature vectors are enhanced by introducing some special keywords in the tweet, before the related words.

The modifications based on morphemes are listed below.

> Morpheme: ISIM_KUCULTME_CEGIZ, example: "kız*cağız*" (as in "poor little girl"); "KUC" keyword is inserted
>
> Morpheme: FIIL_YETERSIZLIK_E, example: "gid*e*medim" (as in "I was not able to go"); "YT" keyword is inserted
>
> Morpheme: FIIL_EMIR_O_SIN, example: "git*sin*" (as in "make him/her go away"); "EMIR" keyword is inserted
>
> Morpheme: FIIL_ZORUNLULUK_MELI, example: "bitir*meli*yim" (as in "I have to finish"); "ZRN" keyword is inserted

Note that Zemberek morphological analyzer may return multiple analyses for a single word. However, the first analysis may not contain the morpheme identified as FIIL_ZORUNLULUK_MELI even though most of the time this is the correct analysis. In order to resolve this shortcoming, verbs that include the strings "-meli" or "-malı" are assumed to contain FIIL_ZORUNLULUK_MELI morpheme.

Except from the special morphemes, adjectives are generally used in sentences that convey emotions. For that reason, keyword “SFT” is inserted before the adjectives in a tweet to mark their locations.

Homonym words, which are written same but with different meanings, are a major cause of disambiguity in natural languages. The Turkish homonym word “kız” has the meanings “girl” as a noun and “to get angry” as a verb. For emotion analysis purpose this word is a strong word that needs to be handled correctly. Therefore, the stem is replaced with “kızN”, to differentiate it, if it is identified as a noun by Zemberek.

Another deficiency of Zemberek is that it may not always return the correct stem. For example, the Turkish words “korkmak” (“to be afraid”) and “korkutmak” (“to scare someone”) share the same stem “kork”. Still, for these two words Zemberek returns “kork” and “korkut” respectively. As the meaning of these words are important in this study, the stem “korkut” is replaced with “kork” to resolve this issue.

4.2.4.3 Hashtag Handling

Hashtags are splitted to group of 3 characters (#saçdökülmesi (#hairloss): “saç”, “dök”, “ülm”, “esi”) in order to capture at least some of the words. This approach is only used in the word n-gram approach. Letter n-gram approach just ignores the hashtag symbol and treats the hashtag as a normal word.

4.3 Feature Vector Construction and Feature Selection

The tweets are needed to be represented as numerical vectors in order to classify them into emotion classes. Feature vector construction step is applied to convert textual tweets into suitable vector forms. Vector forms can be huge and sparse, feature selection methods are utilized.

4.3.1 Feature Vector Construction

Several different features are extracted from the data set to construct suitable feature vectors for the classification task. The first approach uses word n-grams and the second approach uses letter n-grams. Punctuation marks (?, ! and .) and emoticons given in Table 4.8 are processed separately. The resulting vectors are weighted using term count and tf-idf feature weightings.

Table 4.8: Emoticon List

:) :-) =) (: (-: (=
;) ;-) (; (-;
;D ;-D ;d ;-d
:D :d =D D:
:p :P :-p :-P
:-(:( ):)-:
:'(:'-( )':)-':
:S :s s: S:
:o :O :-O :-o o: O:
:/ :-/

The first method uses word n-grams to construct the feature vectors. For a given tweet, first preprocessing is applied. Stop words such as pronouns and numbers are removed. Then, mistyped words are processed so that they can be analyzed properly by Zemberek. Emoticons and punctuation marks are temporarily removed for later use.

The preprocessed tweet is then tokenized to words. Hashtag tokens are split into 3 letter words. Other words are analyzed with Zemberek and replaced with their stems. At this point, the morphological information is utilized to insert additional keyword in-between the stems (as described in Section 4.2.4). During this step, the stems might be modified as well. This results in an ordered list of stems (and also keywords and parts of hashtags). This resulting list of stems is used to extract n-grams and construct a term count vector. Afterwards, the counts of emoticons and punctuations marks are appended to this term count vector.

The second method uses letter n-grams to construct the feature vectors. For a given tweet, emoticons and punctuation marks are temporarily removed for later use. The remaining tweet is tokenized into words. For each word, letter n-grams are extracted separately to construct a term count vector for the whole tweet. Term counts of emoticons and punctuations marks are appended.

In both methods, after all the tweets are processed, the term count vectors are used to calculate tf-idf vectors.

4.3.2 Feature Selection

Two different feature selection methods are used in order to reduce the number of features used in classification. The first method is information gain. The second method uses WordNet Affect lexicon complementing it with a English-Turkish dictionary.

Information gain is realized with the WEKA tool. When the related training data set is fed to the feature selection algorithm, the information gain for each feature is calculated. The features are sorted according to their information gain scores and the highest ranked n features are chosen. n is the parameter of the method indicating number of features to be selected.

The major problem with WordNet Affect lexicon is that it is not provided for Turkish language. To overcome this problem, the lexicon had be processed.

WordNet Affect contains numeric synset identifiers. In order to derive English words, these identifiers are mapped to WordNet. Once English words are extracted, these words are looked up from an online English-Turkish dictionary [2].

While translating a word, the dictionary may return multiple translations in the target language. In this returned list, although some translations consist of a single Turkish word, some translations consist of multiple words. The translations containing only a single word are taken as they are. The translations that have multiple words require special treatment. If the translations contains two words and the second word is a Turkish helper verb ("*something* etmek" = "to do *something*", "*something* olmak" = "to be *something*", "*something* yapmak" = "to make *something*"), then the second

[2] www.zargan.com

word is discarded and the first word is added to the translation list. Other translations are discarded.

In order to reduce the feature set, the generated Turkish WordNet Affect translation list is utilized. The features that do not appear in the list are dropped. Note that if there is an associated marker keyword, as explained in Section 4.3.1, it is also dropped. Emoticons and punctuation marks are preserved.

4.4 Classification

Classification is performed on feature vectors. Different feature vector types are constructed for different experiments. Naïve Bayes Classifier, Complement Naïve Bayes Classifier, Support Vector Machines and k-Nearest Neighbors are utilized in experiments.

Preprocessed and enhanced tweets (as described in Section 4.2) are used to construct word n-grams vectors to use them as features in classification step. Unigrams, bigrams and trigrams are used independently.

Letter n-grams are extracted from preprocessed but not enhanced tweets. A tweet is split into words. Letter n-grams are constructed from each word separately and placed in the same feature vector. Case-sensitivity is considered and experiments are conducted with case-insensitive and case-sensitive n-grams. Letter Unigrams, bigrams and trigrams are used independently.

These feature vectors are expanded with the emoticons and punctuation marks features. There are 10 features for emoticons. These emoticon groups are shown in Table 4.8. Also there are 3 features for punctuation marks for '.', '?' and '!'. These 13 features are appended to the end of n-gram feature vectors previously constructed.

Each term of feature vectors are weighted using both term count and tf-idf. Term count indicates the number of times the feature is encountered in a tweet. Tf-idf considers both term frequency and document frequency. Term frequency is calculated as the term count divided by sum of all terms' counts. Document frequency is calculated as the number of tweets containing the term divided by total number of tweets. Tf-idf is the ratio of the term frequency to logarithm of the document frequency.

Using letter n-grams and word n-grams cause the count of columns in feature vectors to be too many. It can be seen in Table 4.9. In order to deal with these enormous number of columns, information gain is used to select the most significant features and remove the others. Experiments are performed using different number of selected features. Another method used to select features involves employing WordNet Affect lexicon. By translating the lexicon to Turkish language, an affective word list is obtained. The word n-grams which do not appear in the affective list are discarded from the feature vectors leaving emotional features back.

Table 4.9: n-gram Counts

	Case-Sensitive	Case-Insensitive
letter bigram	1824	812
letter trigram	9825	6741
word unigram	-	5554
word bigram	-	35010
word trigram	-	39121

4.4.1 Naïve Bayes and Complement Naïve Bayes Classification

Naïve Bayes Classifiers are utilized in most text classification tasks. In this study, WEKA tool is used to apply both Naïve Bayes and Complement Naïve Bayes. Since Complement Naïve Bayes was devised to overcome some deficiencies of Naïve Bayes, it is expected to achieve better results.

4.4.2 SVM Classification

SVM classifier uses a kernel function to map a low dimensional feature space to a higher dimensional space to separate classes. In this study, radial basis function kernel is used. To classify with RBF kernel, cost parameter C and γ are needed. In order to obtain the best parameters, grid search mechanism is utilized with the LIBSVM tool. Grid search tool applies a coarse search over an two dimensional parameter space.

After a maxima region is detected, a fine grained search is applied around the vicinity of that region to obtain the best parameter pair.

SVM classifier needs it input feature vectors to be normalized. Each feature of the training set is normalized between -1 and 1 linearly. The scaling constants are stored to be later used in the normalization of the test set.

4.4.3 K-Nearest Neighbors Classification

K-NN algorithm uses k number of closest neighbors in the training set to determine the class label of the test set item. In this study, for k-NN algorithm knnclassify function of MATLAB is utilized. The best k value is selected with cross validation in the training set. The k value whose classification accuracy is the highest is selected to be the parameter of the algorithm. Majority rule is applied to determine the emotion class of the test set item.

CHAPTER 5

EXPERIMENTS

In this chapter, experimental results are given and discussed. Experiments are conducted on the the data set generated as a part of this thesis. Aim is to classify tweets as one of the "anger", "sadness", "fear", "joy", "disgust" and "surprised" classes. The whole data set is divided into a training set and a test set. Classifiers are trained on the training set and classification performances are measured for the test set.

5.1 Naïve Bayes Classifier Experiments

Table 5.1: Naïve Bayes Classification Results for Letter Unigrams and Bigrams

		Accuracy
77 features (Unigram, case-sensitive, emoticons and punctuation marks)	**term count**	32.25%
	tf-idf	32.83%
45 features (Unigram, case-insensitive, emoticons and punctuation marks)	**term count**	28.42%
	tf-idf	32.92%
1840 features (Bigram, case-sensitive, emoticons and punctuation marks)	**term count**	38.17%
	tf-idf	**40.17%**
825 features (Bigram, case-insensitive, emoticons and punctuation marks)	**term count**	38.25%
	tf-idf	39.42%

The results of the Naïve Bayes Classifier is shown in the Table 5.1, Table 5.2 and Table 5.3. As presented in these tables, in some cases term count is slightly better in other cases tf-idf weighting is better. For this classifier, highest accuracy is obtained in letter trigram with tf-idf weighting. Word bigrams and trigrams do not yield notable results. Since tweets are short in length, n-grams become more unique as we increase

Table 5.2: Naïve Bayes Classification Results for Letter Trigrams with Information Gain Feature Selection

		Accuracy
2400 features (Information gain, trigram, case-sensitive, emoticons and punctuation marks)	**term count**	47.50%
	tf-idf	45.83%
1600 features (Information gain, trigram, case-sensitive, emoticons and punctuation marks)	**term count**	47.42%
	tf-idf	46.33%
800 features (Information gain, trigram, case-sensitive, emoticons and punctuation marks)	**term count**	47.83%
	tf-idf	47.00%
2400 features (Information gain, trigram, case-insensitive, emoticons and punctuation marks)	**term count**	46.33%
	tf-idf	44.08%
1600 features (Information gain, trigram, case-insensitive, emoticons and punctuation marks)	**term count**	46.67%
	tf-idf	45.33%
800 features (Information gain, trigram, case-insensitive, emoticons and punctuation marks)	**term count**	**48.08%**
	tf-idf	46.58%

the n count. Therefore, classification performance decreases.

5.2 Complement Naïve Bayes Classifier Experiments

The results of the Complement Naïve Bayes Classifier is shown in the Table 5.4, Table 5.5 and Table 5.6. Complement Naïve Bayes Classifier outperforms the result obtained with Naïve Bayes Classifier in most of the cases. As in the Naïve Bayes Classifier, this classifier achieves the best result with letter trigrams. In most of the cases, it performs better with the term count.

5.3 Support Vector Machines Classifier Experiments

The results of the SVM Classifier is shown in the Table 5.7, Table 5.8 and Table 5.9. In order to obtain best results from SVM classifier parameter selection must be performed. In the parameter selection grid.py is utilized on the training set with cross validation. The best parameters are then used to train the classifier. Trained model is used to predict the labels of the test data set. The best results are obtained via this classifier with the unigram word features which are selected using information gain. For this best case configuration, also WordNet Affect feature selection is tested and

Table 5.3: Naïve Bayes Classification Results for Word n-grams with Information Gain Feature Selection

		Accuracy
2400 features (Information gain, unigram, case-insensitive, emoticons and punctuation marks)	**term count**	43.42%
	tf-idf	43.58%
1600 features (Information gain, unigram, case-insensitive, emoticons and punctuation marks)	**term count**	43.25%
	tf-idf	44.58%
800 features (Information gain, unigram, case-insensitive, emoticons and punctuation marks)	**term count**	42.67%
	tf-idf	**44.92%**
2400 features (Information gain, bigram, case-insensitive, emoticons and punctuation marks)	**term count**	31.58%
	tf-idf	31.58%
1600 features (Information gain, bigram, case-insensitive, emoticons and punctuation marks)	**term count**	31.50%
	tf-idf	31.58%
800 features (Information gain, bigram, case-insensitive, emoticons and punctuation marks)	**term count**	31.50%
	tf-idf	31.58%
2400 features (Information gain, trigram, case-insensitive, emoticons and punctuation marks)	**term count**	25.42%
	tf-idf	26.50%
1600 features (Information gain, trigram, case-insensitive, emoticons and punctuation marks)	**term count**	25.42%
	tf-idf	26.60%
800 features (Information gain, trigram, case-insensitive, emoticons and punctuation marks)	**term count**	25.42%
	tf-idf	26.60%

the result is shown in the Table 5.9.

5.4 K-Nearest Neighbors Classifier Experiments

The results of the k-NN Classifier is shown in the Table 5.10, Table 5.11 and Table 5.12. Cross validation is used to select the best k value for each experiment. For a range of k values training set is divided into 10 parts. For each partition classification accuracy is calculated and the best performed k value is selected. Then, that k value is used in classification and predict the class labels for the test data set. For distance metric "euclidean" distance is utilized. The classification results generally are worse than the other classifiers. The best results are obtained with word unigrams approach.

Table 5.4: Complement Naïve Bayes Classification Results for Letter Unigrams and Bigrams

		Accuracy
77 features (Unigram, case-sensitive, emoticons and punctuation marks)	**term count**	35.92%
	tf-idf	30.67%
45 features (Unigram, case-insensitive, emoticons and punctuation marks)	**term count**	28.08%
	tf-idf	30.67%
1840 features (Bigram, case-sensitive, emoticons and punctuation marks)	**term count**	**48.08%**
	tf-idf	42.67%
825 features (Bigram, case-insensitive, emoticons and punctuation marks)	**term count**	47.33%
	tf-idf	44.32%

Table 5.5: Complement Naïve Bayes Classification Results for Letter Trigrams with Information Gain Feature Selection

		Accuracy
2400 features (Information gain, trigram, case-sensitive, emoticons and punctuation marks)	**term count**	50.75%
	tf-idf	46.58%
1600 features (Information gain, trigram, case-sensitive, emoticons and punctuation marks)	**term count**	49.50%
	tf-idf	46.00%
800 features (Information gain, trigram, case-sensitive, emoticons and punctuation marks)	**term count**	49.75%
	tf-idf	46.58%
2400 features (Information gain, trigram, case-insensitive, emoticons and punctuation marks)	**term count**	**53.67%**
	tf-idf	49.75%
1600 features (Information gain, trigram, case-insensitive, emoticons and punctuation marks)	**term count**	52.08%
	tf-idf	49.00%
800 features (Information gain, trigram, case-insensitive, emoticons and punctuation marks)	**term count**	50.83%
	tf-idf	48.00%

5.5 Discussion

In this section, a comparison between the baseline algorithm and the algorithm suggested in this study is made. The general outcomes and deficiencies of the algorithm are discussed.

5.5.1 Baseline Algorithm

For the baseline to the task of classification of tweets in Turkish language, study of Boynukalın [4] is chosen since there are limited number of studies for emotion anal-

Table 5.6: Complement Naïve Bayes Classification Results for Word n-grams with Information Gain Feature Selection

		Accuracy
2400 features (Information gain, unigram, case-insensitive, emoticons and punctuation marks)	**term count**	47.17%
	tf-idf	43.83%
1600 features (Information gain, unigram, case-insensitive, emoticons and punctuation marks)	**term count**	**47.50%**
	tf-idf	44.58%
800 features (Information gain, unigram, case-insensitive, emoticons and punctuation marks)	**term count**	46.75%
	tf-idf	45.25%
2400 features (Information gain, bigram, case-insensitive, emoticons and punctuation marks)	**term count**	33.25%
	tf-idf	32.92%
1600 features (Information gain, bigram, case-insensitive, emoticons and punctuation marks)	**term count**	32.92%
	tf-idf	32.92%
800 features (Information gain, bigram, case-insensitive, emoticons and punctuation marks)	**term count**	32.83%
	tf-idf	32.67%
2400 features (Information gain, trigram, case-insensitive, emoticons and punctuation marks)	**term count**	24.00%
	tf-idf	24.00%
1600 features (Information gain, trigram, case-insensitive, emoticons and punctuation marks)	**term count**	24.00%
	tf-idf	24.00%
800 features (Information gain, trigram, case-insensitive, emoticons and punctuation marks)	**term count**	24.00%
	tf-idf	24.00%

ysis in Turkish. For various configurations of parameters, the experiments are conducted and results are shown in Table 5.13 for comparison. Although in the baseline study promising results are achieved, for Twitter domain accuracy of the classification did not catch up with the results of the original study and the results achieved in this work. The reason lies in the domain differences. Data set of the baseline study is composed of longer text than the tweets to infer emotion. It has fewer spelling mistakes to handle. The method proposed in this study takes punctuation marks, emoticons and some special morphems into account to improve results. It considers correcting specific mistyped patterns in addition to using Zemberek's word correction algorithm.

5.5.2 Overview of Experimental Results

In this study, Naïve Bayes, Complement Naïve Bayes, Support Vector Machines and K-Nearest Neigbors classification algorithms are tested. Since feature space is large, information gain is used to select the most distinctive features from the feature set.

Table 5.7: SVM Classification Results for Letter Unigrams and Bigrams

		Accuracy
77 features (Unigram, case-sensitive, emoticons and punctuation marks)	**term count**	41.92%
	tf-idf	41.33%
45 features (Unigram, case-insensitive, emoticons and punctuation marks)	**term count**	36.00%
	tf-idf	38.08%
1840 features (Bigram, case-sensitive, emoticons and punctuation marks)	**term count**	**51.92%**
	tf-idf	50.75%
825 features (Bigram, case-insensitive, emoticons and punctuation marks)	**term count**	50.92%
	tf-idf	50.67%

Table 5.8: SVM Classification Results for Letter Trigrams with Information Gain Feature Selection

		Accuracy
2400 features (Information gain, trigram, case-sensitive, emoticons and punctuation marks)	**term count**	55.50%
	tf-idf	56.75%
1600 features (Information gain, trigram, case-sensitive, emoticons and punctuation marks)	**term count**	54.00%
	tf-idf	55.50%
800 features (Information gain, trigram, case-sensitive, emoticons and punctuation marks)	**term count**	54.83%
	tf-idf	55.50%
2400 features (Information gain, trigram, case-insensitive, emoticons and punctuation marks)	**term count**	**56.83%**
	tf-idf	55.25%
1600 features (Information gain, trigram, case-insensitive, emoticons and punctuation marks)	**term count**	54.83%
	tf-idf	56.08%
800 features (Information gain, trigram, case-insensitive, emoticons and punctuation marks)	**term count**	53.83%
	tf-idf	55.83%

Also, WordNet Affect lexicon is employed to discard non-emotional words and hence resulting in fewer features to deal with. Term count and tf-idf weighting are utilized.

The best results are obtained from Support Vector Machines with word unigrams and 800 features obtained via information gain. Confusion matrix for the best case is shown in Table 5.14. According to the confusion matrix, the highest confusion occurs between anger and surprise classes. 42 of the anger class items are labelled as disgust class. The least confusion occurs between joy and disgust classes. 1 of the joy class items is labelled as disgust class. When confused test set items are further investigated, it is seen that some of these tweets do not exactly indicate one specific emotion, rather they may involve mixed emotions or more than one emotion.

Table 5.9: SVM Classification Results for Word n-grams with Information Gain Feature Selection and WordNet Affect

		Accuracy
2400 features (Information gain, unigram, case-insensitive, emoticons and punctuation marks)	**term count**	66.67%
	tf-idf	48.42%
1600 features (Information gain, unigram, case-insensitive, emoticons and punctuation marks)	**term count**	68.33%
	tf-idf	47.33%
800 features (Information gain, unigram, case-insensitive, emoticons and punctuation marks)	**term count**	**69.92%**
	tf-idf	48.33%
600 features (Information gain, unigram, case-insensitive, emoticons and punctuation marks)	**term count**	46.92%
	tf-idf	48.33%
3755 features (WordNet Affect, unigram, case-insensitive, emoticons and punctuation marks)	**term count**	46.67%

Table 5.10: k-NN Classification Results for Letter Unigrams and Bigrams

		Accuracy
77 features (Unigram, case-sensitive, emoticons and punctuation marks)	**term count**	33.08%
	tf-idf	33.33%
45 features (Unigram, case-insensitive, emoticons and punctuation marks)	**term count**	30.42%
	tf-idf	33.00%
1840 features (Bigram, case-sensitive, emoticons and punctuation marks)	**term count**	36.50%
	tf-idf	30.00%
825 features (Bigram, case-insensitive, emoticons and punctuation marks)	**term count**	**36.58%**
	tf-idf	31.50%

Since data set items are automatically collected using hashtag search, this situation is expected. In some cases, tweets indicate sarcasm which leads to wrong emotional inference. Also, since tweets use considerably different style than formal texts, it is not really possible to discard all the proper names. Some Twitter users do not capitalize the first letter of a proper name. Therefore, some nouns such as “mutlu”, which means “happy”, can not be distinguished from the common surname “Mutlu”. Some example tweets from the test set, their preassigned emotional classes and the class labels obtained from the best classification are shown in Appendix A. The list contains both correct and incorrect classification results.

Feature selection method using WordNet Affect lexicon does not surpass the results obtained from classification using information gain. This is an expected outcome. The original lexicon is in English and some of the words have more than one mean-

Table 5.11: k-NN Classification Results for Letter Trigrams with Information Gain Feature Selection

		Accuracy
2400 features (Information gain, trigram, case-sensitive, emoticons and punctuation marks)	**term count**	35.00%
	tf-idf	36.25%
1600 features (Information gain, trigram, case-sensitive, emoticons and punctuation marks)	**term count**	37.92%
	tf-idf	38.08%
800 features (Information gain, trigram, case-sensitive, emoticons and punctuation marks)	**term count**	39.92%
	tf-idf	**40.25%**
2400 features (Information gain, trigram, case-insensitive, emoticons and punctuation marks)	**term count**	36.75%
	tf-idf	35.25%
1600 features (Information gain, trigram, case-insensitive, emoticons and punctuation marks)	**term count**	35.67%
	tf-idf	36.92%
800 features (Information gain, trigram, case-insensitive, emoticons and punctuation marks)	**term count**	39.17%
	tf-idf	40.00%

ing. The other meanings of the words are not emotional in some cases and this situation results in non-emotional words in the translated list. In addition to this, some of the Turkish translations of the lexicon words have homonyms which are not emotional. Therefore, the resulting Turkish words list contains both emotional and non-emotional words. The list also lacks some emotional Turkish words which do not have direct translation in English language which leads to missing out on some of the important features.

Table 5.12: k-NN Classification Results for Word n-grams with Information Gain Feature Selection

		Accuracy
2400 features (Information gain, unigram, case-insensitive, emoticons and punctuation marks)	**term count**	38.42%
	tf-idf	36.33%
1600 features (Information gain, unigram, case-insensitive, emoticons and punctuation marks)	**term count**	39.50%
	tf-idf	38.50%
800 features (Information gain, unigram, case-insensitive, emoticons and punctuation marks)	**term count**	40.58%
	tf-idf	**42.67%**
2400 features (Information gain, bigram, case-insensitive, emoticons and punctuation marks)	**term count**	23.33%
	tf-idf	25.92%
1600 features (Information gain, bigram, case-insensitive, emoticons and punctuation marks)	**term count**	23.17%
	tf-idf	26.75%
800 features (Information gain, bigram, case-insensitive, emoticons and punctuation marks)	**term count**	22.83%
	tf-idf	26.67%
2400 features (Information gain, trigram, case-insensitive, emoticons and punctuation marks)	**term count**	18.08%
	tf-idf	19.08%
1600 features (Information gain, trigram, case-insensitive, emoticons and punctuation marks)	**term count**	18.33%
	tf-idf	19.17%
800 features (Information gain, trigram, case-insensitive, emoticons and punctuation marks)	**term count**	18.17%
	tf-idf	19.00%

Table 5.13: Baseline Algorithm Classification Results for Term Count

	Accuracy
Total=1200 features (100 unigrams, 100 bigrams from each class with WLLR)	26.58%
Total=1800 features (200 unigrams, 100 bigrams from each class with WLLR)	28.33%
Total=2400 features (200 unigrams, 100 bigrams, 100 trigrams with WLLR)	28.50%
Total=3000 features (200 unigrams, 200 bigrams, 100 trigrams with WLLR)	29.50%

Table 5.14: Confusion Matrix for SVM Classification with Word n-grams (Term Count, Information Gain 800 Features)

Class	Anger	Sadness	Fear	Disgust	Surprise	Joy
Anger	81	14	24	31	42	8
Sadness	12	159	4	6	5	14
Fear	6	8	160	10	12	4
Disgust	16	5	6	161	10	2
Surprise	25	9	13	16	126	11
Joy	3	40	2	1	2	152

CHAPTER 6

CONCLUSION AND FUTURE WORK

6.1 Conclusion

In this thesis, emotion analysis on Turkish tweets have been studied. Several different machine learning methods were employed to correctly classify tweets according to their affective content. Experimental results have shown that it is possible to apply classification algorithms to informal short texts and obtain promising results.

Although sentiment analysis is a research topic which is being studied for long time, a finer grained emotion analysis is more of a recent topic especially in Turkish language. Therefore, a ready to use data set tagged with emotion classes was not available for use during this study. Initially, Ekman's list of basic emotions was chosen as the emotion classes to be used in the learning algorithms. The list was composed of the emotions anger, disgust, fear, joy, sadness and suprise. Then, employing a keyword based search on Twitter, a data set was constructed. This keyword based search ensured that the data set was labelled with emotion classes. The data set consisted 1000 tweets for each emotion class adding up to 6000 tweets in total.

Preprocessing step started with removal of non-Turkish letters and user name mentions. This step also included elimination of retweets, which were tweets posted again by some other person and elimination of very short tweets, which were possibly unsuitable to derive emotion from. Mistyped word correction phase was carried out using Zemberek and introducing a rule-based correction algorithm. Morphological analysis covered stemming, incorporation of negations and handling of some special case morphemes. Getting rid of the irrelevant parts and enhancing the significant parts were crucial for the success of the classification step because these steps allowed us to represent the data concisely and extract the knowledge to be fed to classifiers.

Feature vectors were constructed using n-gram approach. Features were weighted with both term count and tf-idf to observe the effect of feature weighting method to classification success. The most significant features were selected and the other features were discarded using information gain and utilizing WordNet Affect lexicon in separate experiments. In this way, the effect of different feature selection methods were able to be investigated. Naïve Bayes, Complement Naïve Bayes, SVM and K-NN classifiers were experimented with to see the effect of classification algorithm. It could be concluded that feature weighting method was not distinctive for the most cases. However, feature selection method information gain was better than the WordNet Affect lexicon use. SVM classifier surpassed the other ones achieving 69.92% classification accuracy with 800 features.

This thesis proposed that with further morphological analysis, special feature enhancing methods and incorporation of informal text features such as emoticons could boost the performance of emotion classification task. With the comparison to the baseline algorithm proposed for formal text emotion analysis, it has been shown that formal text and informal short text differed greatly and required special methods.

6.2 Future Work

This study can be used to analyze individual Twitter users over time to see how their emotional status changes according to the content of their tweets. Public reactions toward some specific events or issues can be observed and trends can be determined.

For future work, hand-crafted keyword list used to retrieve emotional tweets might be improved to reach a wide range of diverse tweets. After construction of a list of core keywords, Zemberek may be utilized to generate new keywords from the core set with a certain agglutinative process. A data set containing more diverse and a greater number of tweets may better generalize the tweets in Turkish.

In the data set construction, since an approach using emotional keyword search is utilized, the data set lacked the emotionally neutral tweets. For future work, tweets that do not constitute affective meaning might be incorporated into the data set. A task to classify a stream of tweets emotionally might be more efficiently carried out if non-emotional tweets are automatically eliminated from the start.

REFERENCES

[1] A. A. Akın and M. D. Akın. Zemberek, an open source nlp framework for turkic languages. *Structure*, 2007.

[2] L. Bentivogli, P. Forner, B. Magnini, and E. Pianta. Revising wordnet domains hierarchy: Semantics, coverage, and balancing. In *in Proceedings of COLING 2004 Workshop on Multilingual Linguistic Resources*, pages 101–108, 2004.

[3] D. Bollegala, Y. Matsuo, and M. Ishizuka. Measuring semantic similarity between words using web search engines. *www*, 7:757–766, 2007.

[4] Z. Boynukalın. Emotion analysis of turkish texts by using machine learning methods. Master's thesis, Middle East Technical University, 2012.

[5] S. Chaffar and D. Inkpen. Using a heterogeneous dataset for emotion analysis in text. In *Advances in Artificial Intelligence*, pages 62–67. Springer, 2011.

[6] C.-C. Chang and C.-J. Lin. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2:27:1–27:27, 2011. Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm` [last accessed on 2013-11-14].

[7] C. Cöltekin. A freely available morphological analyzer for turkish. In *LREC*, 2010.

[8] C. Cortes and V. Vapnik. Support-vector networks. *Machine learning*, 20(3):273–297, 1995.

[9] T. Danisman and A. Alpkocak. Feeler: Emotion classification of text using vector space model. *AISB 2008 Convention Communication, Interaction and Social Intelligence*, 1:53, 2008.

[10] P. Ekman. Basic emotions. *Handbook of cognition and emotion*, 4:5–60, 1999.

[11] U. Eroğul. Sentiment analysis in turkish. Master's thesis, Middle East Technical University, 2009.

[12] A. Go, R. Bhayani, and L. Huang. Twitter sentiment classification using distant supervision. *CS224N Project Report, Stanford*, pages 1–12, 2009.

[13] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten. The weka data mining software: an update. *ACM SIGKDD explorations newsletter*, 11(1):10–18, 2009.

[14] C.-W. Hsu, C.-C. Chang, C.-J. Lin, et al. A practical guide to support vector classification, 2003.

[15] M. Kaya, G. Fidan, and I. H. Toroslu. Sentiment analysis of turkish political news. In *Proceedings of the The 2012 IEEE/WIC/ACM International Joint Conferences on Web Intelligence and Intelligent Agent Technology-Volume 01*, pages 174–180. IEEE Computer Society, 2012.

[16] E. Kouloumpis, T. Wilson, and J. Moore. Twitter sentiment analysis: The good the bad and the omg! *ICWSM*, 11:538–541, 2011.

[17] Z. Kozareva, B. Navarro, S. Vázquez, and A. Montoyo. Ua-zbsa: a headline emotion classification through web information. In *Proceedings of the 4th International Workshop on Semantic Evaluations*, pages 334–337. Association for Computational Linguistics, 2007.

[18] S. Kullback and R. A. Leibler. On information and sufficiency. *The Annals of Mathematical Statistics*, pages 79–86, 1951.

[19] V. I. Levenshtein. Binary codes capable of correcting deletions, insertions and reversals. In *Soviet physics doklady*, volume 10, page 707, 1966.

[20] G. A. Miller. Wordnet: A lexical database for english. *COMMUNICATIONS OF THE ACM*, 38:39–41, 1995.

[21] S. Mohammad. #emotional tweets. In **SEM 2012: The First Joint Conference on Lexical and Computational Semantics – Volume 1: Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation (SemEval 2012)*, pages 246–255, Montréal, Canada, 7-8 June 2012. Association for Computational Linguistics.

[22] S. Mohammad. Portable features for classifying emotional text. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 587–591, Montréal, Canada, June 2012. Association for Computational Linguistics.

[23] A. Neviarouskaya, H. Prendinger, and M. Ishizuka. Analysis of affect expressed through the evolving language of online communication. In *Proceedings of the 12th international conference on Intelligent user interfaces*, pages 278–281. ACM, 2007.

[24] K. Nigam, A. K. McCallum, S. Thrun, and T. Mitchell. Text classification from labeled and unlabeled documents using em. *Machine learning*, 39(2-3):103–134, 2000.

[25] K. Oflazer. Two-level description of turkish morphology. In *Proceedings of the sixth conference on European chapter of the Association for Computational Linguistics*, EACL '93, pages 472–472, Stroudsburg, PA, USA, 1993. Association for Computational Linguistics.

[26] A. Qadir and E. Riloff. Bootstrapped learning of emotion hashtags# hashtags4you. *WASSA 2013*, page 2, 2013.

[27] J. D. Rennie, L. Shih, J. Teevan, D. R. Karger, et al. Tackling the poor assumptions of naive bayes text classifiers. In *ICML*, volume 3, pages 616–623. Washington DC), 2003.

[28] K. R. Scherer and H. G. Wallbott. Evidence for universality and cultural variation of differential emotion response patterning. *Journal of personality and social psychology*, 66(2):310, 1994.

[29] C. Strapparava and A. Valitutti. WordNet-Affect: an affective extension of WordNet. In *Proceedings of LREC*, volume 4, pages 1083–1086, 2004.

[30] W. Wang, L. Chen, K. Thirunarayan, and A. P. Sheth. Harnessing twitter" big data" for automatic emotion identification. In *Privacy, Security, Risk and Trust (PASSAT), 2012 International Conference on and 2012 International Confernece on Social Computing (SocialCom)*, pages 587–592. IEEE, 2012.

[31] C. Yang, K. H.-Y. Lin, and H.-H. Chen. Emotion classification using web blog corpora. In *Web Intelligence, IEEE/WIC/ACM International Conference on*, pages 275–278. IEEE, 2007.

APPENDIX A

LISTS

A.1 Stop Words and Example Classification Results

Table A.1: Stop Word List

bey	ettiği	kimi	pek
burada	ettiğini	mı	şey
da	falan	mi	şeyden
de	gibi	mu	şeyi
dolayısıyla	hangi	mü	şeyler
edecek	herhangi	nasıl	şöyle
eden	ile	ne	tüm
ederek	ilgili	nerde	üzere
edilecek	ise	nerede	var
ediliyor	işte	nereye	vardı
edilmesi	itibariyle	olan	ve
ediyor	ki	olsa	veya
en	kim	olup	ya
etmesi	kimden	olursa	yani
etti	kime	öyle	

Table A.2: Example Classification Result List

Original Tweet	**Class Label**	**Class Prediction**
Tamam bende yaptım ama gelmiyorum geç kalıcam dememek telefonunu açmayıp ima etmek #sinirbozucu	anger	anger
Tam tweet atmaya kalkiyorum ve twitter tanrilari tweet atmami istemiyorcasina "sunucular suan mesgul" cevabi aliyorum. #sinirbozucu	anger	surprised
İstanbulun pisliğinden, gürültüsünden, trafiğinden, kalabalığından #tiksiniyorum	disgust	disgust
insanlarin dış goruntusuyle dalga gecip kendini bi ..... zannedenlerden #tiksiniyorum	disgust	anger
istanbul'da motosiklete bindigim zaman trafikteki dort tekerlekli canavarlardan #korkuyorum	fear	fear
Dün Ceylanpınar'da Suriye'den atıln mermiler yüzünden 1 çocuk öldü 1 çocuk yarlndı.Bu ülkede insan hayatnn bu kdr ucuz olmsndn #korkuyorum	fear	disgust
Tee kastamonulardan gelmis tadindan yenmez :) #helva #cide #mutluluk *evde*	joy	joy
#çatalzeytin #summer #çay #mutluluk. Özledim o günlerii ..	joy	sadness
#mutsuzum çünkü bu berbat eğitim sistemi sayesinde lise yıllarım dershane okul ve ev üçgeni arasında mekik dokumakla geçiyor 7*	sadness	sadness
Kişi hem imamlık yapıp hem de Rocker olamaz mı? Özgürlük sadece işinize gelen şeyler değildir. #mutsuzum #IronMaiden	sadness	anger
eet adamlarin herseyi degisik cikista pasaport kontrolu de yok...:) ama bu sefer angaryadan kurtariyorlar saolsunlar... #hayret	surprised	surprised
Giyecek hiç bir şeyim yok diyen bir kız #Ohaaadiyorum beni de giyseydin gerçi yakışırdım üstüne ((:	surprised	disgust