A Framework for Sentiment Analysis in Turkish: Application to Polarity Detection of Movie Reviews in Turkish

A. Gural Vural[1], B. Barla Cambazoglu[2], Pinar Senkul[1] and Z. Ozge Tokgoz[1]

[1] Middle East Technical University, Ankara, Turkey
[2] Yahoo! Research, Barcelona, Spain

Abstract. Sentiment analysis has been an active research area for quite some time. Unfortunately, most works are specific to the English language. In this work, we present a framework for unsupervised sentiment analysis in Turkish text documents. As part of our framework, we customize the SentiStrength sentiment analysis library by translating its lexicon to Turkish. We apply our framework to the problem of classifying the polarity of movie reviews. For performance evaluation, we use a large corpus of Turkish movie reviews obtained from a popular Turkish social media site. Although our framework is unsupervised, it is demonstrated to achieve a fairly good classification accuracy, approaching the performance of supervised polarity classification techniques.

1 Introduction

The sentiment analysis of user-generated text content in the Web has attracted lots of research interest in the last decade [9]. The research has focused on various aspects of sentiment analysis including extraction [14], classification [10], and retrieval [4] of sentiments. In this work, our focus is primarily on predicting the polarity associated with a short piece of human-generated text, i.e., predicting whether a given piece of text has positive or negative sentiments. This is an important task that finds application in the analysis of online product reviews and user mood detection with implications for monetization in commercial products.

In literature, there are two lines of techniques for assessing the polarity of a text. In the first line of techniques [6, 10], a machine learning model is built using a training corpus containing documents that are labeled as positive or negative. The labels on the training documents can be assigned manually by the editorial teams or, if available, can be automatically inferred from the user ratings (e.g., the star ratings in online product reviews). Once the model is built, it can be used to predict the polarity of previously unseen documents. Most models in literature use n-grams extracted from the documents as the basic features.

In the second line of techniques [11, 12], an editorially created lexicon is used. Each word in the lexicon is associated with some positive and/or negative sentiment score. Certain non-lexical rules (e.g., negation) are further used to modify the sentiment scores of words depending on their current context. The polarity of a piece of text is determined by combining the sentiment scores of its words through some hand-crafted function. The lexicon-based techniques have two advantages over the techniques that rely on machine learning. First, since the lexicon construction is unsupervised, there is no need to obtain labeled training data. Second, since there is no dependency to a particular training corpus, it is easier to develop domain-independent polarity classifiers.

Although most works considered only English, several recent works applied sentiment analysis to other languages. In this work, our focus is on Turkish, which is an agglutinative language that makes the sentiment analysis a relatively more complicated problem. The contributions of our work are the following. We propose a framework for sentiment analysis in text written in Turkish, especially focusing on informal and noisy text found in the Web. We customize a lexicon-based sentiment analysis library, SentiStrength [11, 12], to make it work with Turkish text. We evaluate the performance of our framework using a large dataset containing online movie reviews that are written in Turkish. The experimental results indicate a fairly good accuracy in predicting the polarity of movie reviews.

The rest of the paper is organized as follows. In Section 2, we provide a survey of the previous work on non-English sentiment analysis and polarity detection in movie reviews. Section 3 gives a brief over of the SentiStrength library. The sentiment analysis framework we propose for Turkish is described in Section 4. Section 5 summarizes the details of our data and presents the performance results, as well as some caveats in our work. The paper is concluded in Section 6.

2 Previous Work

The literature on sentiment analysis is vast [9]. Herein, rather than providing a detailed literature overview, we very briefly survey some representative works on sentiment analysis in non-English text. We also provide an overview of prior work applying sentiment analysis to movie reviews since this is the application we use to evaluate our framework.

Most sentiment analysis techniques developed so far are for English. In recent years, however, several works focused on non-English languages. Atteveldt et al. [1] used machine learning techniques to automatically determine the polarity of political news stories in Dutch. They extracted lexical and syntactic features besides three different clusterings of similar words based on annotated material. Ghorbel and Jacot [5] devised a supervised learning strategy using linguistic features obtained through part-of-speech tagging and chunking as well as semantic orientation of words obtained from the SentiWordNet sentiment analysis tool [2] to classify the polarity of movie reviews in French. Since SentiWordNet is for English, the authors translated the French words to English before getting their semantic orientation. Zhang et al. [15] addressed the challenges that are unique to the Chinese language. They evaluated a rule-based polarity classification approach against different machine learning approaches. Hiroshi et al. [6] developed a sentiment analysis system using a transfer-based machine translation engine and applied it to Japanese. In literature, the research on sentiment analysis in

Turkish is limited. To the best of our knowledge, a detailed analysis is presented only in Erogul's master thesis [3], which rely on supervised machine learning for polarity classification. Our work differs from [3] as we use a lexicon-based approach, completely unsupervised and independent of the problem domain.

A number of works applied sentiment analysis to predict the polarity of movie reviews. Turney [13] used an unsupervised learning technique based on the estimated semantic orientation of extracted phrases. He classified the reviews as "recommended" or "not recommended" according to their average semantic orientation. A prediction accuracy of 65.8% is reported for a collection of 120 movie reviews. Pang et al. [10] compared the performance of different machine learning techniques on movie reviews taken from the IMDB movie database. The SVM classifier is shown to yield better performance than the other classifiers. The used features included unigrams, bigrams, part of speech information, and the position of the terms in the text. Among these feature types, unigrams were found to yield better performance. In a recent work, Oghina et al. [8] tried to predict the movie ratings based on the feedback obtained from different social media channels like Twitter and YouTube. Kennedy and Inkpen [7] combined machine learning with a simple technique based on counting the positive/negative words in the movie reviews, showing further improvements over both techniques. Erogul's aforementioned work [3] also uses a movie review dataset for the performance evaluation. That work reports 85% prediction accuracy in a binary (positive and negative classes) classification scenario.

3 SentiStrength

Our customized version of SentiStrength is the most important module in the sentiment analysis framework we designed for Turkish. Herein, we briefly summarize the features of the original SentiStrength library. The framework itself and our modifications on SentiStrength are presented in Section 4.

SentiStrength is a lexicon-based sentiment analysis library developed by Thelwall et al. [11, 12]. Given a short piece of text written in English, the library generates a positive and a negative sentiment score for each word in the text. The positive scores range from $+1$ (neutral) to $+5$ (extremely positive) while the negative scores range from -1 (neutral) to -5 (extremely negative).[3] The final positive (negative) sentiment score for the input text is computed by taking the maximum (minimum) of the positive (negative) sentiment score of the words in the text. In addition, the library can also produce binary labels about the polarity of the text. Table 1 shows some sample English sentences and the sentiment scores produced by SentiStrength. Interested readers may try the tool on the SentiStrength site to get further insight.[4]

To compute the sentiment scores of individual words, SentiStrength relies on a number of editorially created word lists:

[3] There is both a positive and a negative score because the input text may contain sentiments in both directions (e.g., "I love you, but I also hate you.") [12].

[4] SentiStrength, `http://sentistrength.wlv.ac.uk/`.

Table 1. Sentiment scores generated by SentiStrength for sample English sentences

Sentence	Positive score	Negative score	Binary prediction
I am going to the school	+1	−1	+1
I like to play chess	+2	−1	+1
I do not like to play chess	+1	−1	+1
I feel sorry for missing the class	+1	−2	−1
I hate your brother	+1	−4	−1
I do not hate your brother	+1	−1	+1
I really love you, but dislike your sister	+4	−3	+1

- *Sentimental word list* contains more than 2500 words together with their associated sentiment scores. The sentimental words and their sentiment scores are compiled by human editors. The list also includes some regular expressions, e.g., the pattern "amaz∗" covers multiple words such as "amaze", "amazed", "amazement", "amazing", "amazingly".
- *Booster word list* contains words that strengthen or weaken the sentiment associated with the succeeding non-neutral words, e.g., the word "good" has a positive sentiment score of +2, whereas in "extremely good", its score becomes +3 due to the preceding booster word "extremely".
- *Idiom list* contains some common phrases. The sentiment scores of individual words in the phrase are overridden, e.g., "how are you" has a sentiment score of +2, instead of a neutral score of +1.
- *Negation word list* contains a few negation words. If a negation word is followed by a positive word, the positive sentiment score is multiplied by −0.5. If a negation word is followed by a negative word, the negative sentiment is turned into neutral. The reader may find related examples in Table 1.
- *Emoticon list* contains some common emoticons which are associated with sentiment scores, e.g., ":)" has a score of +2.

According to [12], the first version of the library could predict positive sentiments with 60.6% accuracy and negative sentiments with 72.8% accuracy, both using five-grade score scales. The latest version of SentiStrength [11], which contains an extended lexicon, is reported to have good performance on six diverse social web data sets, achieving prediction accuracies comparable to the machine-learned polarity prediction techniques. In [11] and [12], also a supervised version of SentiStrength is developed. In the supervised version, the sentiment scores assigned to the words by human editors are tried to be fine tuned using a labeled training corpus. Since earlier works [11, 12] do not report significant performance improvements over the unsupervised version, in our work, we do not make an attempt to create and evaluate a supervised version of SentiStrength.

4 Sentiment Analysis Framework

The motivation behind creating a sentiment analysis framework specific to Turkish, rather than using an existing framework for English, is due to certain differences between Turkish and English. These differences can be summarized as

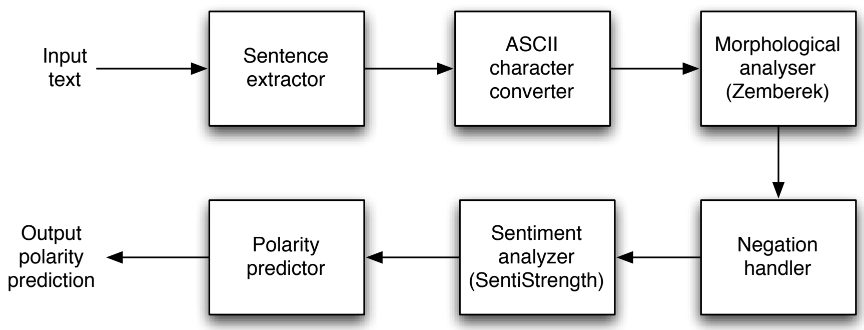

Fig. 1. The pipeline of modules in our sentiment analysis framework.

follows. First, Turkish is an agglutinative language, i.e., new and arbitrarily long words can be created by adding many suffixes to a root word. The added suffixes may change the polarity of words. In practice, it is not feasible to detect and add all variants of Turkish words into the sentimental word list. Second, negation words usually occur after the negated word. This is different than English, where negation words typically precede the word they negate. Moreover, in Turkish, the negation word can be in the form of a suffix ("-ma") within the word. Finally, Turkish has several letters that are missing in English ("ç", "ğ", "ı", "ö", "ş", "ü"). In informal writing on the Web, people tend to substitute these Turkish letters with the closest ASCII English letters ("c", "g", "i", "o", "s", "u"). This creates complication in identifying the words.

We have designed and implemented a sentiment analysis framework taking into account the above-mentioned differences. Our framework consists of a pipeline of several software modules, each providing some input to the succeeding module in the pipeline. The input to the framework is a piece of text written in Turkish and the output is a prediction about the polarity of the sentiments in the text, i.e., either a positive or a negative class prediction.[5] The pipeline for the proposed framework is illustrated in Fig. 1. In what follows, we describe the modules in this pipeline. Table 2 shows the execution of these modules for a sample input text.

– *Sentence extractor:* This is a simple module which splits the input text into sentences based on certain sentence separators (i.e., ".!?"). Each sentence is then passed to the next module as a separate input.
– *ASCII character converter:* Each word in the input sentence is looked up in a dictionary and checked for spelling errors. If a corresponding term is not found in the dictionary or there is a spelling error, the term is passed as input to an ASCII-tolerant parser to see if the word is written using ASCII character substitution. At this step, the parser may rewrite the term by substituting certain characters (e.g., "guzel" becomes "güzel").

[5] We do not consider the neutral class and break the ties in favor of the negative class.

Table 2. The execution of the modules in the pipeline for a sample input text

Module	Output of the module
Original input text	"bu film cok guzel degildi. hic kimseye tavsiye etmem."
Sentence splitter	"bu film cok guzel degildi." and "hic kimseye tavsiye etmem."
ASCII converter	"bu film çok güzel değildi." and "hiç kimseye tavsiye etmem."
Morphological analyzer	"bu film çok güzel değil." and "hiç kimse tavsiye et-me-m."
Negation handler	"bu film çok güzel değil." and "hiç kimse tavsiye _NOT_ etmek."
Sentiment analyzer	"bu film çok güzel[3] değil [*-0.5 negated multiplier]." and "[-1 emoticon]hiç kimse tavsiye[4] _NOT_ [*-0.5 negated multiplier] etmek."
Polarity predictor	(in all three methods, the polarity is predicted as negative)
(`sentence-binary`)	-1 and -1
(`sentence-max/min`)	$(+1, -2)$ and $(+1, -2)$
(`word-sum`)	-1.5 and -3

- *Morphological analyzer:* We then perform morphological analysis on the words in the sentence. To this end, we use the Zemberek library, which is an open source, platform-independent, and general-purpose natural language processing library for Turkic languages.[6] Zemberek's morphological analyzer basically finds all possible root forms and suffixes of a given word. We always assume that the first morphological analysis result of Zemberek is the correct one and use that. After the morphological analysis, certain suffixes are removed from the selected word form. This is because some suffixes (e.g., tense and person suffixes) are not valuable for sentiment analysis.
- *Negation handler:* The negation takes places in Turkish most often in two forms, either in the form of a separate word negating one of the preceding words (e.g., "güzel değil" ("not nice")) or in the form of the "-ma" suffix, which is a part of the negated word (e.g., "olmayacak" ("it will not happen")). To handle the negations of the first form, we rely on a SentiStrength feature, which we will briefly describe later. To handle the second form of negations, we modify the sentence and introduce an artificial keyword[7] before the negated word. This artificial word is added to the negation word list of our customized version of the SentiStrength library.
- *Sentiment analyzer:* As mentioned before, we customized the lexicon files of the SentiStrength library by translating them to Turkish. The translation is performed by human editors, who also added to the lists new words that were missing in the original SentiStrength. Table 3 shows the number of entries in the original (English) and customized (Turkish) versions of SentiStrength. Other than the changes in the lexicon files, we did not perform any modification in the scoring logic of SentiStrength as the codes of the library are not publicly available. To cope with the first form of nega-

[6] Zemberek 2, `http://code.google.com/p/zemberek/`.
[7] We use "_NOT_" as the keyword.

Table 3. Number of lexicon entries in different lists of the original (English) and modified (Turkish) SentiStrength library

List	English version	Turkish version
Sentimental word list	2,546	872
Booster word list	27	13
Negation word list	16	4
Idiom list	9	38

tion words in Turkish, SentiStrength is initialized with a special parameter (`-negatingWordsOccurAfterSentiment`) to negate sentimental words before as well as after the negation words. SentiStrength by default applies negation to the words within a window of length 1. Our experiments indicated that a window size of 3 gives the best accuracy for Turkish, e.g., in the sentence "güzel film değil" ("it is not a nice movie"), "güzel" is affected from negation although it is not right before the negation word "değil" ("not").

- *Polarity predictor:* This module takes as input the sentiment scores associated with each word in the initial input text as well as the information about sentence splitting. The polarity of the input piece of text is determined according to the sentiment score assigned to the text. In our work, we evaluate three different approaches, which we refer to as `sentence-binary`, `sentence-max/min`, and `word-sum`:
  - `sentence-binary`: For each sentence in the original input text, we use the binary score (i.e., $+1$ or -1) generated by SentiStrength. The sum of the scores over all sentences gives the sentiment score of the text.
  - `sentence-max/min`: For each sentence, we use the maximum of positive word scores and the minimum of negative word scores that are computed by SentiStrength. We simply compute the average of these scores, separately for the positive and negative scores. The sum of the average of positive scores and the average of the negative scores gives the sentiment score of the text.
  - `word-sum`: We use the sum of the sentiment scores of all words in the input text as its sentiment score.[8]

  In all three scoring techniques, if the sentiment score of the text is positive, then its polarity is predicted as positive; otherwise, it is predicted as negative.

5 Experiments

5.1 Dataset

To evaluate our sentiment analysis framework, we try to predict the polarity of online movie reviews written in Turkish. The review data is obtained from a movie site called Beyazperde,[9] a well-known website that provides information about movies. Beyazperde allows its users to enter comments on movies and

[8] The sentiment scores of individual word in the text can be obtained from SentiStrength via the `-explain` option.
[9] Beyazperde, `http://www.beyazperde.com`.

Table 4. Properties of the movie review dataset used in the experiments

Property	Positive reviews	Negative reviews
Number of reviews	30,000	30,000
Reviews including an emoticon	4,093	2,477
Average number of words	36.02	37.07
Average number of sentences	3.75	3.82
Average word length	6.03	5.92

Table 5. Performance results (over all review instances)

	`sentence-binary`	`sentence-max/min`	`word-sum`
Accuracy	70.39%	74.83%	75.90%
True positive rate	36.89%	39.83%	40.70%
False positive rate	13.11%	10.17%	9.30%
True negative rate	33.50%	35.00%	35.30%
False negative rate	16.50%	15.00%	14.70%

state their opinion about the movie by selecting an icon (positive or negative), which forms the ground-truth polarity labels in our data. For the experiments, we picked a random sample of positive and negative reviews, each with equal number of documents. The properties of our dataset is shown in Table 4.

5.2 Results

We evaluate the performance in terms of accuracy, i.e., the ratio of the number of reviews whose polarity is correctly predicted to the total number of reviews. Table 5 reports the accuracy values for the three scoring techniques mentioned in Section 4, as well as the true/false positive/negative rates. In this experiment, we activate all modules in the processing pipeline. According to the table, the `word-sum` scoring technique is the best performing technique, while `sentence-binary` performs considerably worse than the other two scoring techniques. This result indicates that a fine-grain (at the word level) aggregation of the sentiment scores is more promising. In Table 5, we also observe that the prediction performance is better for positive reviews. This is in contrast to what is reported by Thelwall et al. [12] for an English dataset. Overall, our performance does not reach the performance of Erogul's supervised machine learning approach on the same data (85% accuracy). However, given that our technique is unsupervised and independent of the problem domain, we believe that the accuracy achieved by our framework (76% accuracy) is promising.

Table 6 shows the accuracies when the ASCII conversion or morphological analysis modules are turned off. We note that turning off the morphological analysis also turns off the negation handling for the within-word negations. According to the table, most of the achieved accuracy is due to the customized SentiStrength library. Nevertheless, including the ASCII conversion and morphological analysis modules in the framework brings reasonable improvement. In particular, for negative reviews, the accuracy increases by about 5% for all three scoring techniques when morphological analysis is turned on. Turning the ASCII conversion module on seems to help more in case of positive reviews.

Table 6. Accuracy when some modules are turned off

Inst.	Modules	`sentence-binary`	`sentence-max/min`	`word-sum`
All	All modules	70.39%	74.83%	75.90%
	No ASCII conversion	69.48%	73.72%	74.67%
	No morphological analysis	67.34%	71.53%	72.30%
+	All modules	73.78%	79.67%	81.40%
	No ASCII conversion	72.36%	77.29%	78.98%
	No morphological analysis	72.12%	78.12%	79.21%
−	All modules	67.00%	70.00%	70.39%
	No ASCII conversion	66.60%	70.15%	70.36%
	No morphological analysis	62.56%	64.95%	65.40%

5.3 Caveats

Our framework is not perfect. In this section, we summarize some of the issues we encountered when using the linguistic tools in our framework. First, since the movie reviews are written using a rather informal language, the likelihood of typos is high. Most often the typos are intentional and are hard to detect by linguistic tools. The spell-checking algorithm we used can handle up to three misplaced or wrong characters in the root and two misplaced or wrong characters in the word suffixes. This check improves the accuracy up to only 0.3%, but the execution time of the overall system increases about 10 times. Second, as stated before, some of the reviews are written using ASCII letters, replacing original Turkish letters. The ASCII converter we used cannot always detect the correct version of the words, as it has no domain knowledge. For instance, "yas" ("mourning") having a negative sentiment score of -3 is the ASCII version of "yaş" ("age") which has neutral sentiment. Our conversion tool fails in certain cases like this. Third, we observed difficulties in sentence splitting due to the misuse of punctuation. This results in wrong interpretations of the negation. For example, the review "filmde hic bir sey yok. manasiz bir film", which contains negative sentiments, gains a positive sentiment if the periods are omitted as in "filmde hic bir sey yok manasiz bir film" ("yok" negates "manasiz").

6 Conclusions

In this paper, we proposed a framework for unsupervised sentiment analysis in Turkish text documents. Our framework used various linguistic tools as well as the customized version of the SentiStrength sentiment analysis tool. We evaluated the performance of our framework by applying it to the problem of polarity prediction of movie reviews. The experiments over a large corpus of Turkish movie reviews indicate reasonable prediction accuracy.

As the future work, we plan to evaluate the performance of our framework over other social media datasets. In addition, to optimize the sentiment word strengths, the framework will be tested against human evaluated texts. We also plan to make the customized SentiStrength library freely available to the research community, to enable the reproducibility of our findings and to support the research on sentiment analysis in Turkish.

Acknowledgments. This work is supported by grant number TUBITAK-112E002, TUBITAK. We would like to thank Umut Erogul for providing us the Turkish movie review dataset used in this work.

References

1. V. Atteveldt, J. Kleinnijenhuis, N. Ruigrok, and S. Schlobach. Good news or bad news? conducting sentiment analysis on dutch text to distinguish between positive and negative relations. *J. Inf. Technol. Polit.*, 5(1):73–94, 2008.
2. A. E. S. Baccianella and F. Sebastiani. Sentiwordnet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining. In *Proc. 7th Conf. Int'l Language Resources and Evaluation*, 2010.
3. U. Erogul. Sentiment analysis in turkish. Master's thesis, Middle East Technical University, 2009.
4. S. Gerani, M. J. Carman, and F. Crestani. Proximity-based opinion retrieval. In *Proc. 33rd Int'l ACM SIGIR Conf. Research and Development in Information Retrieval*, pages 403–410, 2010.
5. H. Ghorbel and D. Jacot. Sentiment analysis of french movie reviews. In V. Pallotta, A. Soro, and E. Vargiu, editors, *Advances in Distributed Agent-Based Retrieval Tools*, volume 361 of *Studies in Computational Intelligence*, pages 97–108. Springer Berlin / Heidelberg, 2011.
6. K. Hiroshi, N. Tetsuya, and W. Hideo. Deeper sentiment analysis using machine translation technology. In *Proc. 20th Int'l Conf. Computational Linguistics*, pages 494–500, 2004.
7. A. Kennedy and D. Inkpen. Sentiment classification of movie reviews using contextual valence shifters. *Computational Intelligence*, 22(2):110–125, 2006.
8. A. Oghina, M. Breuss, M. Tsagkias, and M. de Rijke. Predicting imdb movie ratings using social media. In *Proc. 34th European Conf. Information Retrieval*, 2012.
9. B. Pang and L. Lee. Opinion mining and sentiment analysis. *Found. Trends Inf. Retr.*, 2:1–135, 2008.
10. B. Pang, L. Lee, and S. Vaithyanathan. Thumbs up? sentiment classification using machine learning techniques. In *Proc. ACL-02 Conf. Empirical Methods in Natural Language Processing*, pages 79–86, 2002.
11. M. Thelwall, K. Buckley, and G. Paltoglou. Sentiment strength detection for the social web. *J. Am. Soc. Inf. Sci. Technol.*, 63(1):163–173, 2012.
12. M. Thelwall, K. Buckley, G. Paltoglou, D. Cai, and A. Kappas. Sentiment strength detection in short informal text. *J. Am. Soc. Inf. Sci. Technol.*, 61(12):2544–2558, 2010.
13. P. D. Turney. Thumbs up or thumbs down?: semantic orientation applied to unsupervised classification of reviews. In *Proc. 40th Annual Meeting on Association for Computational Linguistics*, pages 417–424, 2002.
14. J. Yi, T. Nasukawa, R. Bunescu, and W. Niblack. Sentiment analyzer: extracting sentiments about a given topic using natural language processing techniques. In *Proc. 3rd IEEE Int'l Conf. Data Mining*, pages 427–434, 2003.
15. C. Zhang, D. Zeng, J. Li, F.-Y. Wang, and W. Zuo. Sentiment analysis of Chinese documents: From sentence to document level. *J. Am. Soc. Inf. Sci. Technol.*, 60(12):2474–2487, 2009.