The Impact of NLP on Turkish Sentiment Analysis

Ezgi Yıldırım
Istanbul Technical University
yildirimez@itu.edu.tr

Fatih Samet Çetin
Turkcell Global Bilgi
fatih.cetin@global-bilgi.com.tr

Gülşen Eryiğit
Istanbul Technical University
gulsenc@itu.edu.tr

Tanel Temel
Turkcell Global Bilgi
tanel.temel @global-bilgi.com.tr

ABSTRACT

Sentiment analysis on English texts is a highly popular and well-studied topic. On the other hand, the research in this field for morphologically rich languages is still in its infancy. Turkish is an agglutinative language with a very rich morphological structure. For the first time in the literature, this paper investigates and reports the impact of the natural language preprocessing layers on the sentiment analysis of Turkish social media texts. The experiments show that the sentiment analysis performance may be improved by nearly 5 percentage points yielding a success ratio of 78.83% on the used data set.

1 Introduction

Sentiment analysis has become a very popular research area because of needs to track and manage population tendency. Many companies today work on this area in order to meet customer expectations and demands. Social microblogging platforms (e.g. Twitter and Facebook) offer an opportunity to get huge amount of easily accessible and processable data. Users of micro-blogging platforms write about their personal lives, their own opinions about political cases, economic changes, companies and their products.

With the emergence of social media platforms, the sentiment analysis studies are shifted from document level analysis [4, 18, 19] towards sentence or phrase level analysis [14, 22, 12, 23, 21]. Recent years showed that syntactic and/or semantic analysis outperforms baseline sentiment analysis methods in many areas such as aspect-based and comparative opinion mining [9, 13, 2]. In order to reach this level of analysis, many other natural language preprocessing stages are required; i.e. tokenization, normalization, parts-of-speech tagging etc...

As in all other natural language processing (NLP) problems, the most widely studied language for sentiment analysis is English. However, studies for morphologically rich languages are not mature yet. Abdul-Mageed et al. [1] used a supervised, two-stage classification approach employing morphological, dialectal, genre specific features besides basic ones for a morphologically rich language, Arabic. Jang and Shin [10] proposes an approach for agglutinative languages and test their method on Korean short movie reviews and news articles. Wiegand et al. [20] investigate the impact of negation in sentiment analysis of German.

In the literature, it has been shown several times that Turkish, due to its highly inflectional and derivational structure, poses many different problems for different NLP tasks when compared to morphologically poor languages. By this property, previous NLP research on Turkish language pioneered the studies for many similar languages. On the other hand, sentiment analysis studies for Turkish are very preliminary; although there exist a couple of studies on sentiment classification of movie reviews, political news, fairytales [17, 11, 3, 15], there exist very few studies on sentiment analysis of social media posts [5,6].

With the emergence of new tools dealing with automatic language processing of social media texts [8], it is now becoming possible to integrate them into higher level applications; i.e. sentiment analysis in our case. But, the following issues still reside as open questions:

1. the impacts of each NLP layers on sentiment analysis.
2. information (e.g. stems, main POS tags, inflectional features) to use from the outputs of beneficial layers.

In this paper, for the first time in the literature, we investigate and report the impact of the preprocessing layers (namely, tokenization, normalization, morphological analysis and disambiguation) on the sentiment analysis of Turkish social media texts. In order to show the maximum sentiment analysis performance to be achieved with flawless NLP tools, we used a hand-annotated sentiment corpus with gold-standard linguistic features.

2 Turkish

Turkish is an agglutinative language where each stem may be inflected by multiple suffixes. Every new suffix concatenation may change the meaning of the word or redefine its syntactic role within the sentence. This feature of Turkish yields to relatively long words (having higher number of characters when compared to other languages). As an ordinary example of this situation, the Turkish word “yapabilirmişcesine” can be translated as “as if he/she is able to do” into English. In addition, the example shows that the same English statement is expressed by a lesser word count (smaller mean sentence length) in the Turkish side. Therefore, semantic analysis of Turkish social media texts is more risky to be defeated by the erroneous writings within this informal domain. The various problems observed in the Turkish Tweets are presented in detail in [16]; these are mainly the missing vowels, diacritics, usage of emoticons, slang words, emo-style writings, spoken accents and high occurrence of spelling errors. The lower word count within a sentence leads to strict dependencies between words in Turkish and the only one single misspelled word can ruin the understandability of the whole sentence. This indicates the importance of normalization preprocessing stage for Turkish differently from English.

POS tagging task for other languages is performed in two stages for Turkish: morphological analysis and morphological disambiguation. Morphological analysis of a single word can produce several possible analyses regardless of the context in sentence. However, only one of them is correct in its context. The correct analysis can be selected by morphological disambiguation process on the morphological analysis results. Linguistic information about the word and possible relations with other words in the sentence can be extracted from the correct analysis.

3 The Used Data Set

For this study, we collected a twitter Turkish sentiment corpus mainly from the telecommunication domain. The data is retrieved from the Twitter API by querying a predetermined list of keywords. The time frame of the collected data was between May, 10th of 2012 and July, 7th of 2013. We refined the corpus from non-Turkish tweets through a language specifier based on a “Language Detection Library for Java”[1]. For the manual annotation of our corpus, we used TURKSENT [7] - a sentiment annotation tool

[1] It is available on https://code.google.com/p/language-detection/

which allows us to annotate the corpus on the following layers: general and target based sentiment, text normalization, morphology and syntax. For this study, we used only the general sentiment, the normalization and the morphological annotation layers of the tool.

Since the sentiment annotations depend on subjective decisions of the human annotators, we applied an inter-annotator agreement filter to increase the confidence level of our sentiment annotations. Our final dataset consists of 12790 tweets manually normalized, morphologically analyzed and classified between 3 sentiments (3541 positive, 4249 negative and 5000 neutral) agreed by two human annotators.

4 Feature Extraction Methods

In this study, we treat the sentiment detection of a tweet as a multi-class classification problem. We used support vector machines (SVM) in order to classify the tweets into one of the three classes (positive, negative, and neutral). When we extract unigrams from all collected data without preprocessing and feature filtering, we get 97472 unique features. This amount of features is extremely huge for machine learning algorithms, because more features ends up with more training time and more resources. In addition to time and resource constraints, irrelevant features may also ruin the steady nature of the trained model. Since feature extraction is an indispensable stage of machine learning algorithms, we applied an extraction method utilizing Inverse Document Frequency (IDF). While Term Frequency is easier and simpler than IDF calculation, it is not convenient if there are lots of recurring parts of texts which are the case for our study. Tweets are treated as single documents while calculating the document frequencies in IDF. After the calculation of IDF values of all unigram features, we filter them according to our proposed filtering algorithm MinClosestTh given below.

MinClosestTh. A small IDF value indicates a characteristic feature for a given class. But, in order for a feature to be discriminative between different classes, the difference between its IDF values should be bigger than a given threshold. In other words, a feature having similar IDF values for two classes does not help for the discrimination of these classes. For example, a stopword or a keyword which is used to retrieve data from Twitter API will have similar small IDF values for all classes. In the light of these observations, after testing with several feature extraction methods[1], we found that MinClosestTh performed the best. In this approach (Equation 1), we find the difference between the smallest and the second smallest IDF[2] value for a feature among all classes. The features, falling outside of this threshold are removed from the feature set.

$$|minIDF - medianIDF| > threshold \qquad (1)$$

Figure 1 shows the histogram of |*minIDF* – *medianIDF*| difference distributions. One should notice that a determined threshold value will also determine the number of features to be used in the experiments; all the words entering to the bins greater than the threshold value will be included into the feature set. In order to select a good threshold value for further experiments, we investigate the sentiment analysis performance with different threshold values (0.1, 0.15, 0.3, 0.5 and 0.8). These are given in small line chart in Figure 1. As seen from the figure, the maximum f-measure is achieved at 0.15. F-measure is not the only metric to select the optimum threshold since the total feature count should also be considered. For example, the number of features in the feature set is 18536 when the threshold is chosen 0.1 and 3685 when 0.15. Although the difference between f-measures is

[1] Due to space constraints, we only provide here our best model.

[2] Since we have only three classes, the second smallest IDF is represented as medianIDF in Equation 1.

not dramatic, the lesser number of features is preferable. We selected 0.15 for further experiments since as seen from the figure, the performance drops consistently without having any important difference in feature counts.

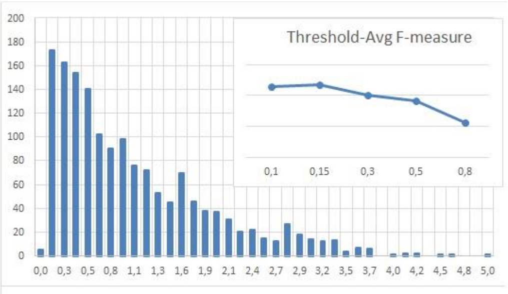

Figure 1. minIDF - medianIDF Histogram and Related Performance

5 Natural Language Processing Layers

Turkish is an agglutinative language and stems can be transformed theoretically to unlimited number of variations with derivational affixes. Moreover, all these different variations of a word may not make a difference on sentiment classification of tweets. Therefore, we want to polarize features which have the similar impact on sentiment to the same pole, and make explicit the difference between poles. We applied mainly three different NLP preprocessing layers (explained in previous sections) to transform features from original versions to the desired representations. Below we give the information extracted from the output of these layers.

Normalization. We used the normalized forms of the words before extracting the features. For instance, “tşkkrler” is normalized as “teşekkürler” (thanks).

Stemming. Stems of words have more general coverage than surface forms. To match different surface forms of a word into one simple token, we used stemming by deleting all inflectional groups and tags from its correct morphological analysis. For instance, “uzmanlar” (specialists), “uzmanlığı” (his/her specialty), “uzmanlık” (specialty) are derived from the same stem “uzman” (specialist). All three forms are turned into their stem “uzman”.

Negation. As stated in [20], the detection of negation needs extra treatment in morphologically rich languages where the negation may be realized within the word with an affixation rather than a separate individual word. The case holds very frequently for Turkish, that’s why our motivation in this section is to model the negation for sentiment analysis.

Negative indicators -such as the inflectional tags at the output of morphological analysis: “+Neg”, “+WithoutHavingDoneSo” (like in use of regardless of, or without stopping)- have a power to turn meaning of words into opposite. For instance, “çekmiyor” (meaning “there is no signal” for the the telco domain) has a morphological analysis such as “çek+Verb+Neg+Prog1+A3sg” where the stem “çek” translated literally as to pull into in English. If a feature will be extracted from this word we represent it as “çek+Neg”. In addition, negation word, “değil” (means to not in English), has the same negative effect on preceding words. We put negation tag if a word contains negative indicators, or has “değil” as its successor. For instance, “iyi değil” (not good) is represented as “iyi+Neg”. Furthermore, we added negation tag to the adjective if its successor is a negative verb. “Net göremiyorum.” (I can’t see clearly.) is transformed to “Net+Neg gör+Neg”. When a word achieved double negation tag because of these conditions, we removed all the negation tags belonging to this word. For example, “sessiz değil” (not silent - “siz” suffix matches with less, like use in noiseless.) converted to “ses”, not to “ses+Neg+Neg”.

Using adjectives. We performed extra effort for adjectives in this research, because of the general belief that adjectives have a direct

Model #	Model Name	Avg. F-measure	Accuracy	Feature #
1	no_normalization – no_preprocessing	73.38	73.72	78025
2	normalization	78.05	78.28	39788
3	normalization-stem	78.35	78.63	17855
4	normalization-stem-neg	78.83	79.09	18493
5	normalization-stem-neg-adj	77.93	78.27	23613

Table 1. Sentiment Analysis Experiments Results

impact on sentiment analysis in comparison with other word types. We added adjectives to the feature set without exposure them to filtering by feature extraction methods defined previously. Even if we applied any of the other NLP preprocessing methods on adjectives just like any other word types, we also used surface form of adjectives as an additional feature instead of using only preprocessed versions. For example, we represent the adjective "tatsız" (tasteless) with two different features, "tat+Neg" (taste+Neg) and "tatsız".

6 Experiments and Discussions

In all of our experiments, we used SVM with linear kernel. In order to increase the confidence level of sentiment analysis, we applied 10-fold-cross-validation. The results are presented in terms of macro average of all iterations in Table 1.

We tested with 5 different NLP preprocessing models where each of them is the addition of a new processing layer on top of the previous one.

The first line of the table (no_normalization – no_preprocessing) presents our baseline model. This test is performed on the original version of the data set, in other words without applying any preprocessing during the selection of the feature set. The further experiments are evaluated according to their preceding experiments, and the performance improvement of the best model is reported with respect to this baseline.

Table 1 shows that the normalization stage (Model #2) contributes to the sentiment analysis, and increases the overall success by about 5 percentage points. On the other hand, although the addition of the stemming (Model #3) results in a slight improvement on top of Model #2, this improvement is not proven to be statistically significant according to McNemar's test. Despite this, Model #3 is considered very valuable since the total number of features is almost reduced by 50% (39788→17855). As a result, the lesser number of features provide us the ability to train our classifier by using less time and less resource as we mentioned in Section 4. This yields the possibility of adding more valuable training data to our machine learning algorithm, especially for active learning experiments.

Our final two experiments (Model #4 & Model #5) deal with the addition of some morphological features into sentiment analysis (detailed in Section 5). Although with the addition of negation (Model #4), we observed a slight improvement in the results, this improvement is again not statistically significant whereas it also increases the total number of selected features. A similar case holds for Model #5 again with no statistical significance, but this time with a small decrease.

As the conclusion, in this study, we showed that normalization is an indispensable stage for sentiment analysis whereas stemming is also very valuable for further studies (e.g. active learning). However, our tested model for the addition of morphological information into the system does not seem well-fitted for this domain. Nevertheless, we may not conclude that the morphological information such as negation has no impact on sentiment analysis. We rather sense that we need to make further research on the inclusion of morphological features such as using them as separate features instead of the approach defined in here (the concatenation: Stem+Neg).

7 Conclusion and Future Work

Feature extraction methods provide us to decrease training time of classifiers, and also they have a positive impact on sentiment analysis success rate. We achieved higher sentiment analysis success rate with less number of features. In addition, we showed how the normalization improves the sentiment analysis on Turkish social media posts. With the normalization preprocessing, we increased the success rate of sentiment analysis from 73.38% to 78.05%, which is the 6.36% relative improvement. By the addition of morphological features we saw a slight improvement from 78.05% to 78.83% which is not statistically significant according to McNemar. However, stemming, which is the first morphological feature that we applied, is dramatically reduced the number of features as an advantage of ability to train models with more data. For our future studies, we will work on developing automatic NLP tools to make use of morphological information. Thereby, we want to build an environment for further linguistic analysis, such as syntax and semantics. We expect to increase sentiment analysis success by such deep analyzes of language.

8 Acknowledgments

This work is accomplished as part of a TUBITAK-TEYDEB (The Scientific and Technological Research Council of Turkey – Technology and Innovation Funding Programs Directorate) project (grant number: 3120605) in “Turkcell Global Bilgi” Information Technology Department. The authors want to thank Ozan Can Arkan for his valuable support during system development.

9 References

[1] **Muhammad Abdul-Mageed, Mona Diab, and Sandra Kübler.** 2014. Samar: Subjectivity and sentiment analysis for Arabic social media. Computer Speech & Language, 28(1):20–37.

[2] **Alexandra Balahur, Rada Mihalcea, and Andrés Montoyo.** 2014. Computational approaches to subjectivity and sentiment analysis: Present and envisaged methods and applications. Computer Speech & Language, 28(1):1–6.

[3] **Zeynep Boynukalin.** 2012. Emotion analysis of Turkish texts by using machine learning methods. Ms.

[4] **Rebecca F Bruce and Janyce M Wiebe.** 1999. Recognizing subjectivity: a case study in manual tagging. Natural Language Engineering, 5(2):187–205.

[5] **Mahmut Çetin and M Fatih Amasyali.** 2013. Active learning for Turkish sentiment analysis. In Innovations in Intelligent Systems and Applications (INISTA), 2013 IEEE International Symposium on, pages 1–4. IEEE.

[6] **Mahmut Çetin and M Fatih Amasyali.** 2013. Supervised and traditional term weighting methods for sentiment analysis. In Signal Processing and Communications Applications Conference (SIU), 2013 21st, pages 1–4. IEEE.

[7] **Gülşen Eryiğit, Fatih Samet Çetin, Meltem Yanik, Tanel Temel, and Ilyas Çiçekli.** 2013. Turksent: A sentiment annotation tool for social media. In Proceedings of the 7th Linguistic Annotation Workshop and Interoperability with Discourse, pages 131–134, Sofia, Bulgaria, August. Association for Computational Linguistics.

[8] **Gülşen Eryiğit.** 2014. ITU Turkish NLP web service. In Proceedings of the Demonstrations at the 14th Conference of the European Chapter of the Association for Computational Linguistics (EACL), Gothenburg, Sweden, April. Association for Computational Linguistics.

[9] **Minqing Hu and Bing Liu.** 2004. Mining and summarizing customer reviews. In Proceedings of the Tenth ACM SIGKDD International

Conference on Knowledge Discovery and Data Mining, KDD '04, pages 168–177, New York, NY, USA. ACM.

[10] **Hayeon Jang and Hyopil Shin.** 2010. Language specific sentiment analysis in morphologically rich languages. In Proceedings of the 23rd International Conference on Computational Linguistics: Posters, COLING '10, pages 498–506, Stroudsburg, PA, USA. Association for Computational Linguistics.

[11] **Mesut Kaya, Guven Fidan, and I Hakkı Toroslu.** 2013. Transfer learning using twitter data for improving sentiment classification of Turkish political news. In Information Sciences and Systems 2013, pages 139–148. Springer.

[12] **Soo-Min Kim and Eduard Hovy.** 2004. Determining the sentiment of opinions. In Proceedings of the 20th international conference on Computational Linguistics, page 1367. Association for Computational Linguistics.

[13] **Bing Liu.** 2012. Sentiment analysis and opinion mining. Synthesis Lectures on Human Language Technologies, 5(1):1–167.

[14] **Satoshi Morinaga, Kenji Yamanishi, Kenji Tateishi, and Toshikazu Fukushima.** 2002. Mining product reputations on the web. In Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining, pages 341–349. ACM.

[15] **Sadi Evren Seker and Khaled Al-naami.** 2013. Sentimental analysis on Turkish blogs via ensemble classifier. In Proceedings Of The 2013 International Conference On Data Mining. DMIN.

[16] **Dilara Torunoğlu and Gülşen Eryiğit.** 2014. A cascaded approach for social media text normalization of Turkish. In 5th Workshop on Language Analysis for Social Media (LASM) at EACL, Gothenburg, Sweden, April. Association for Computational Linguistics.

[17] **A Gural Vural, B Barla Cambazoglu, Pinar Senkul, and Z Ozge Tokgoz.** 2013. A framework for sentiment analysis in Turkish: Application to polarity detection of movie reviews in Turkish. In Computer and Information Sciences III, pages 437–445. Springer.

[18] **Janyce M Wiebe, Rebecca F Bruce, and Thomas P O'Hara.** 1999. Development and use of a gold standard data set for subjectivity classifications. In Proceedings of the 37th annual meeting of the Association for Computational Linguistics on Computational Linguistics, pages 246–253. Association for Computational Linguistics.

[19] **Janyce Wiebe.** 2000. Learning subjective adjectives from corpora. In AAAI/IAAI, pages 735–740.

[20] **Michael Wiegand, Alexandra Balahur, Benjamin Roth, Dietrich Klakow, and Andrés Montoyo.** 2010. A survey on the role of negation in sentiment analysis. In Proceedings of the Workshop on Negation and Speculation in Natural Language Processing, NeSp-NLP'10, pages 60–68, Stroudsburg, PA, USA. Association for Computational Linguistics.

[21] **Theresa Wilson, Janyce Wiebe, and Paul Hoffmann.** 2005. Recognizing contextual polarity in phrase level sentiment analysis. In Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing, HLT '05, pages 347–354, Stroudsburg, PA, USA. Association for Computational Linguistics.

[22] **Jeonghee Yi, Tetsuya Nasukawa, Razvan Bunescu, and Wayne Niblack.** 2003. Sentiment analyzer: Extracting sentiments about a given topic using natural language processing techniques. In Data Mining, 2003. ICDM 2003. Third IEEE International Conference on, pages 427–434. IEEE.

[23] **Hong Yu and Vasileios Hatzivassiloglou.** 2003. Towards answering opinion questions: Separating facts from opinions and identifying the polarity of opinion sentences. In Proceedings of the 2003 conference on Empirical methods in natural language processing, pages 129–136. Association for Computational Linguistics.