Available online: December 27, 2017

Commun.Fac.Sci.Univ.Ank.Series A2-A3
Volume 59, Number 2, Pages 69-79(2017)
DOI: 10.1501/commua1-2_0000000105
ISSN 1303-6009
http://communications.science.ankara.edu.tr/index.php?series=A2

SENTIMENT ANALYSIS USING A RANDOM FOREST CLASSIFIER ON TURKISH WEB COMMENTS

NERGIS PERVAN and HACER YALIM KELEŞ

Abstract. Sentiment analysis is an active research area since early 2000s as a field of text classification. Most of the studies in this field focus on the analysis using the text in English language, where the Turkish and the other languages have fallen behind. The purpose of this research is to contribute to the text analysis in Turkish language using the contents that we access through web sites. In particular, we deduce the sentiment behind noisy product reviews and comments in a highly popular commercial web page. In this context, we generate a unique dataset that includes 9100 product review samples for training our classification model. There are different word representation methods that are utilized in sentiment analysis, such as bag-of-words and n-gram models. In this work, we generated our word models using the word2vec algorithm. In this model, each word in the vocabulary is represented as a vector of 300 dimensions. We utilize 70% of our dataset in the training of a Random Forest Model and make binary classification of sentiments as being positive or negative, utilizing the ratings of the user for the product as classification labels. In the highly noisy and unfiltered comments, we achieve an accuracy of 84.23%.

1. Introduction

Sentiment analysis has emerged as an important topic with the increase of social media interactions, use of forums and blogs, sales comments and ratings through e-commerce websites. Sentiment analysis is a field of Natural Language Processing (NLP), which is also referred to as, opinion mining, sentiment classification, opinion extraction, etc. in the literature. The research in this field has started in the early 2000s with the works of [1], [1], [3], [4], [5] and [6]. The use of the term *sentiment analysis* first appeared in [7].

Sentiment analysis is a way to determine the writer's opinions polarity as positive or negative in a piece of text, about a particular topic, and product etc. The field applications contributed to many tasks on the evaluation of consumer products, understanding the impacts of some social events, and evaluating movie reviews. In

Received by the editors: November 11, 2017; Accepted: December 27, 2017.
Key word and phrases: Sentiment Analysis, Word2Vec, Random Forest, Turkish Web Comments.

addition to these, there are even studies that make stock market predictions using some sentiment states on tweets [8].

Feature extraction on texts is a challenging problem; the challenge is on converting the characters, words, sentences or documents to computational units which is useful for sentiment classification. There are recent studies based using deep learning algorithms based on character level text representations [9] [10]. Different word-level algorithms that are effective representations of words for feature extraction on texts are proposed, such as *word2vec* [11], *glove* [12] and *fasttext* [13]. The word vectors have semantic and syntactic meaning of words. Using the pre-trained word vector models are beneficial in terms of time in classifying sentences.

In this work, we first collected customer reviews including misspelling words and meaningless character combinations from e-commerce websites. We used the entire dataset for word embedding and a part of the dataset for classification. The obtained word representation model was used for classifying Turkish reviews as positive or negative with a Random Forest (RF) classifier.

The paper is organized as follows: In Section 2, we explain the materials and the proposed method, in Section 3, we describe our experiments and discuss the results. We conclude the paper in Section 4.

2. MATERIALS AND METHODS

The proposed model is outlined in Figure 1. The first step is data generation that is main contribution to this domain because of lack of Turkish labeled data set. Data preprocessing is the second step is required for noisy refinement to obtain both word embedding vector and training samples. The last step is training classification model

using training samples are obtained with the transformation of labeled data set conjunction with word embedding to review vectors.

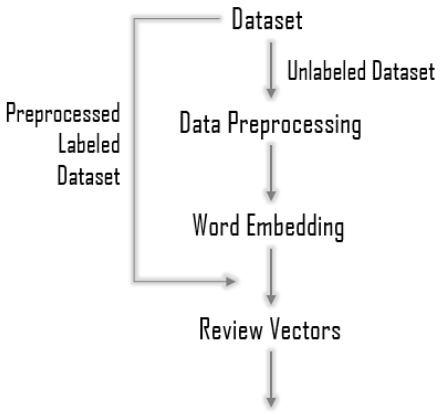

Figure 1. Flow of the method

Dataset Preparation

We collected the dataset from a Turkish e-commerce web site, which includes costumer product reviews in electronics category. This category has many sub-categories such as computers, cell phones, TV, video games, etc. Collected comments are not in ideal shape all the time; they usually contain some misspelling words and meaningless word structures, therefore consist of highly noisy sentences. Some sample sentences from our original dataset collection are shown in Table 1.

The dataset includes a total of 93922 consumer reviews. For supervised model generation, we need a label for each review that represents the sentiment behind the review. We automatically determined labels by using the information encoded with the star ratings that is provided by the user together with each review. For example, if a customer likes a product very much, he/she gives a rating using five or four stars. We associate the labels of the comments with their star ratings. There are up to five star categories, i.e. from one to five, for each comment but in training, we used only certain sentiments in the reviews that have either one star, i.e. a negative sentiment, or five star, i.e. a positive sentiment. We discarded in between ratings in our dataset. We approximately balanced the number of positive and negative reviews in the dataset.

TABLE 1. Sample review sentences from our original dataset. * indicates correct spelling of reviews.

Reviews
bu telefon ***mutish*** *kendime aldim oneririm* * bu telefon müthiş kendime aldım öneririm
Tlfnumdan *cok memnunum kaldim teşekkür ler* *Telefonumdan çok memnun kaldım teşekkürler
tlf *çok güzel* ***herkeze*** *tavsiye ederim.bu paralara alınabilecek en iyi* ***tlfn****..* * Telefon çok güzel herkese tavsiye ederim bu paralara alınabilecek en iyi telefon.
*Geçikmeli kargo ama üründe sıkıntı yok yine****de tşkr*** * Gecikmeli kargo ama üründe sıkıntı yok yine de teşekkürler
hoparlor eywallah ta *basa* ***glince*** *bu ne ya çok kötü* ***bi*** *bas sistemi son sese açınca berbat beklentimin altında* ***bi*** *ürün beğenmedim* * hoparlör eyvallah da basa gelince bu ne ya çok kötü bir bas sistemi son sese açınca berbat beklentimin altında bir ürün beğenmedim

Data Preprocessing

Data preprocessing is an important and necessary step before training models and requires different attention and care in different languages, especially for Turkish language. In opinion mining problems, we need to remove uninformative characters, words, phrases, etc., hence redundancy, from the dataset. The data refinement helps reducing the dimensionality and noise in the data in terms of word representation; hence helps easy generation of word embedding and training classification models.

Since we collected the dataset from the web, each review contains html tags. We started data preprocessing by removing these tags from the data. We also removed all auxiliary characters except for the Turkish letters. In some cases the characters may express the sentiment behind the review when the combinations of characters correspond to some emoticons, like smiley, laughing, crying, etc. To unify the word representations with a simple format, we converted all the remaining letters to lowercase representation. This helps us interpreting the content in a case-insensitive way. We also apply word reduction, i.e. the word lists in a review contains a set of unique words. Tokenization is the task of splitting text based on a character (blank character, punctuation) to tokens that is a piece of character sequences. In preprocessing, we removed punctuations before tokenization, and then we generated each sentence to list of words. In addition, lemmatization is a commonly used process in data preparation; yet for Turkish language lemmatization is not necessary because it is a morphologically rich language. Turkish language consist of suffixes which

changes the meaning of a word negatively, hence we did not use any lemmatization. Moreover, a recent study shows that lemmatization is not necessary for English language sentiment analysis neither [14].

Word Embedding

There are a few different algorithms for word embedding. We used word2vec representation that is proposed by Google researchers in 2013. Word2vec model training is fast and efficient. The model generates a feature vector representation for each word in a vector space. There are two ways to obtain the model; first is using a pre-trained model which is generated by Google using 100 billion words in English language. These features have 300 dimensions and in total there are 30 million unique words. The second way is training a model from scratch using your own dataset in a language you want. Since we want to generate Turkish word models, we trained word2vec model using all the data in our Turkish dataset. We used *Gensim (Generate Similar)* library of Python to implement vector-space modelling and topic modelling. The word2vec model takes sentences, which are the reviews for the obtained data set, as the input and learns word vector representations. We set some parameters for model implementation as follows. The word2vec algorithm supplies continuous bag-of-words (CBOW) and skip gram architecture for producing word vector representations. In *Gensim* default architecture is CBOW and we used it; we generated the dimensionality of the feature vectors for each learned unique words to *300*, set the maximum distance between target word and words around the target word within a sentence to *10* and ignored all words with an occurrence less than *40* times.

Classification

For classification, we first need to generate training samples in a suitable form. We used 9100 training and 3700 test samples for total of 12800 samples. Words in all samples are represented by our word2vec model with a 300-dimensional vector corresponding to that word. Each review becomes a two dimensional tensor, i.e. $N_i \times 300$, here N_i is the number of words in review *i*. We need to create a fixed size vector representation for each review for training, yet the number of words in each review, N_i, are different, For this purpose, we computed the averages of the vectors for each review to represent as a point in the feature space of 300 dimensions. The subset of the attained vectors that we prepared for training with labels as negative and positive sentiments are utilized by a RF classifier in training. The generated model is then used in testing to classify a given, i.e. unseen, Turkish sentiment as a positive or negative sentiment.

3. EXPERIMENTAL RESULTS

This section shows the results of our word2vec model and RF model.

Training word2vec Model Results

First, we used 65525 unlabeled comments in Turkish to initialize word embeddings. After we tuned parameters required for *Gensim* tool, as a result of unsupervised learning, we attained 4328-word vector representations. When we upgrade the number of word samples to 93922 and regenerate the word vector representations, a total of 5711 unique word vectors are generated in about 7 seconds. When we use these word models, our RF classification accuracy also increases by 1.59%.

In Figure 2, we show the produced word vectors in a 2d space. Although individual words are not readable in this figure, we can see some word clusters. When we zoom in, the clusters contain similar words in terms of semantic in Figure 2 and syntactic in Figure 3 (The content is best read in the digital form). In Figure 3 the conjugations of word 'yapmak' ('do' in English) are close to each other such as 'yapmam'('I do not'), 'yapmayı' ('doing'), 'yapacağım' ('I will do'), 'yaparım' ('I do'), 'yapan' ('who do'), 'yapamayacağım' ('I will not be able to). Although they seem far away on the graph, the values in the two dimensions are very close to each other.

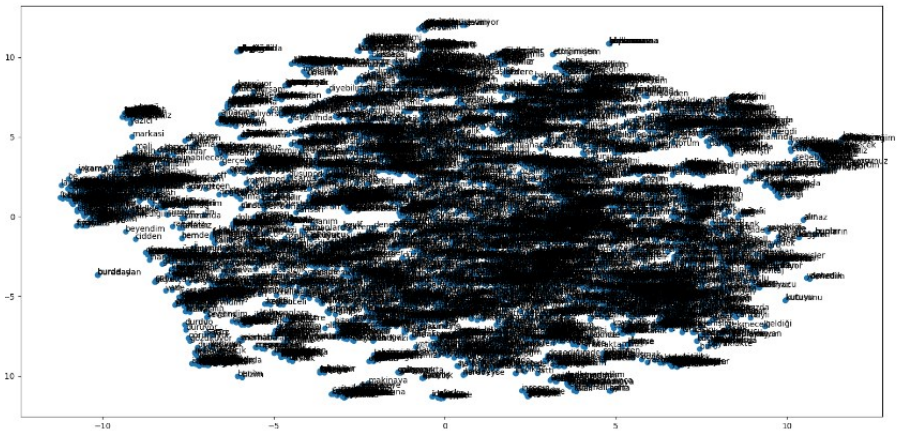

FIGURE 2. Distribution of the Turkish word2vec model words in two-dimensional vector space.

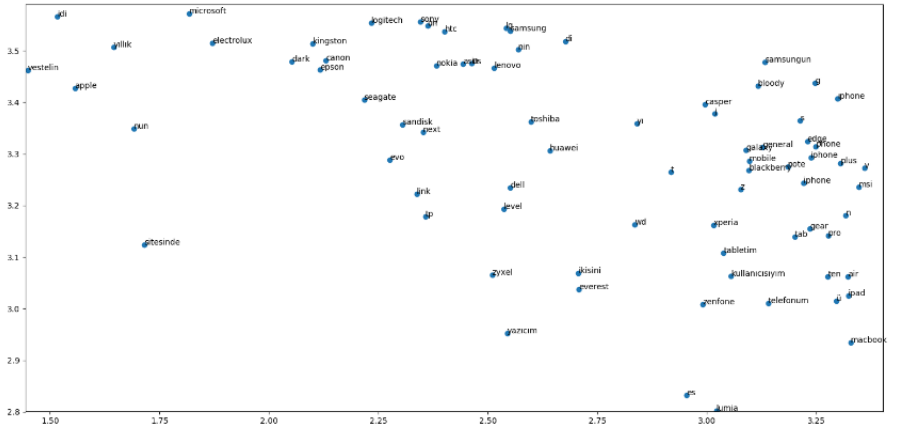

FIGURE 3. Projections of words that have similar semantics.

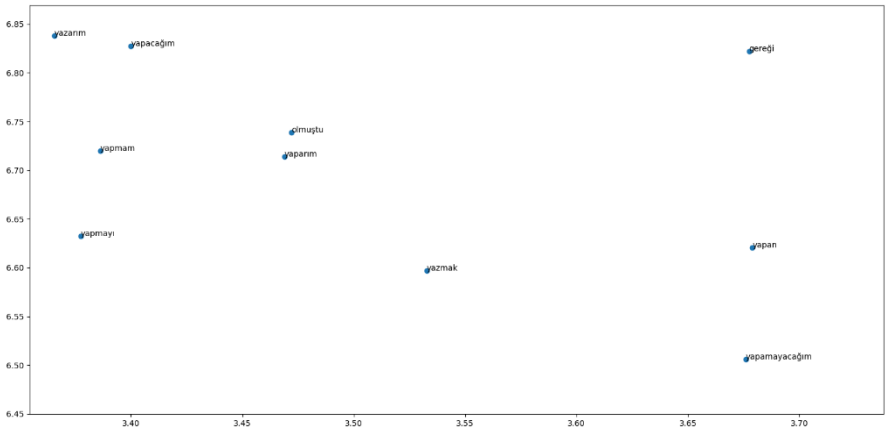

FIGURE 4. Projections of words that have similar syntax.

One of the problems of the word embedding model is that some opposite semantic words are found to be similar to each other in their word vector representations. There are recent studies that aims to project not only semantic and syntactic but also sentiment content of text before creating a model [15], [16]. [17] emphasizes the same problem with an approach distinctly using existing word embedding model.

Classification Results

We used 200 decision trees in the training of our random forest classifier. We tested our model with 3100 test samples and obtained 84.23% accuracy.

The sentiment of the text in the obtained dataset is semantically noisy. In Table 2 we show some semantically conflicting samples from the test set. Despite the positive opinion it implies hidden a negative meaning, the review is ranked as negative, and our model classified it correctly as negative. Similarly, in the second example, the word 'inanılmaz'('incredible') is misleading, i.e. have a positive meaning, and resulted in an incorrect estimation. In third review, the beginning of the sentence has positive sentiment and the rest of the sentence goes on negative; yet in this case our model could infer the true meaning. The last sample is misclassified although it clearly contains a positive sentiment. This sentence has an implied semantic meaning that exaggerates the usefulness of the product by saying it very implicitly, i.e. telling that the previous products are not products at all. So, RF model is having difficulty to mine the implied meaning behind the sentiments.

TABLE 2. Test samples.

Reviews	Target	Prediction
program yüklenince ram doluyor fazla program kurulmazsa ideal (ram is full when program is loaded, ideal if no more programs are installed)	Negative	Negative
İnanılmaz gürültülü bir yazıcı. (An incredibly noisy printer.)	Negative	Positive
İşinizi görecek bir alet fakat mouse la normal mouse gibi iş yapamıyoruz alışık deyilim açıkçası (a tool that serves your needs but with this mouse we can not work like (we do with) a normal mouse, actually I am not used to.)	Positive	Positive
Bunu kullandıktan sonra önceden kullandıklarım ne idi diye insan düşünmekten kendini alamıyor kalite mükemmel (After using this, people can not get away from thinking what I used before quality is perfect)	Positive	Negative

4. Conclusion

In this paper, our purpose is to contribute to Turkish language sentiment analysis using a machine learning approach. First, we collected a unique dataset, which was a primary challenge in Turkish language analysis, since there were no publicly available dataset. We applied a set of preprocessing techniques to create a useful training and test samples. We generated word vector representations using our dataset with an unsupervised learning algorithm, i.e. word2vec model generation algorithm. These representations are used in training an RF model for Turkish sentiment classification. We achieved 84.23% classification accuracy with our test samples.

The obtained word vector models and the dataset will be used in our future researches and will be made publicly available. Our future direction is training a sequence based deep learning model that can reveal complex and hidden semantic meanings behind the sentences.

REFERENCES

[1] Wiebe, J. "Learning Subjective Adjectives from Corpora", *Proceedings of the Seventeenth National Conference on Artificial Intelligence and Twelfth*

Conference on Innovative Applications of Artificial Intelligence, July 30- August 03 (2000): 735-740.

[2] Das, S.R. and Chen, M. Y. 2001. "Yahoo! for Amazon: Extracting Market Sentiment from Stock Message Boards". *In Proceedings of the 8th Asia Pacific Finance Association Annual Conference,* (2001).

[3] Morinaga, S., Yamanishi, K., Tateishi, K. and Fukushima, T. "Mining Product Reputations on the Web". *In Proceedings of ACM SIGKDD International Conference on Knowledge Discovery and Data Mining,* (2002).

[4] Tong, R. M. "An Operational System for Detecting and Tracking Opinions in On-Line Discussion". *In Proceedings of SIGIR Workshop on Operational Text Classification,* (2001).

[5] Pang, B., Lee, L. and Vaithyanathan. S. "Thumbs up? Sentiment Classification Using Machine Learning Techniques". *In Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP),* (2002): 79–86.

[6] Turney, P. 2002, "Thumbs Up or Thumbs Down? Semantic Orientation Applied to Unsupervised Classification of Reviews". *In Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics,* (2002): 417–424.

[7] Nasukawa, T. and Yi, Jeonghee. "Sentiment analysis: Capturing Favorability Using Natural Language Processing". *In Proceedings of the KCAP-03, 2nd Intl. Conf. on Knowledge Capture,* (2003).

[8] Bollen, J., Mao, H. and Zeng, X. 2010. "Twitter Mood Predicts the Stock Market". *Journal of Computational Science*, (2010): 2(1), 1–8.

[9] Kim, Y., Jernite, Y., Sontag, D. and Rush, A. "Character-Aware Neural Language Models". *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence (AAAI-16),* (2016).

[10] Zhang, X., Zhao, J. and LeCun, Y. "Character-level Convolutional Networks for Text Classification*". In Proceedings of NIPS,* (2015).

[11] Mikolov, T., Sutskever, I., Chen, K., Corrado, G., and Dean, J. "Distributed Representations of Words and Phrases and their Compositionality". *In Proceedings of NIPS,* (2013).

[12] Pennington, J., Socher, R., and Manning, C. D. 2014. "Glove: Global Vectors for Word Representation". *Proceedings of the Empiricial Methods in Natural Language Processing (EMNLP),* (2014): 12.

[13] Bojanowski, P., Grave, E., Joulin, A. and Mikolov, T. "Enriching Word Vectors with Subword Information". *arXiv preprint,* (2016): 1607.04606.

[14] Camacho-Collados, J. and Pilehvar, M.T. "On the Role of Text Preprocessing in Neural Network Architectures: An Evaluation Study on Text Categorization and Sentiment Analysis". *arXiv preprint,* (2017): *1707.01780*

[15] Lan, M., Zhang, Z., Lu, Y., and Wu, J. 2016. "Three Convolutional Neural Network-Based Models for Learning Sentiment Word Vectors towards Sentiment Analysis". *In Proceedings of the 2016 International Joint Conference on Neural Networks (IJCNN-16),* (2016): 3172-3179.

[16] Tang, D., Wei, F., Qin, B., Yang, N., Liu, T., and Zhou, M. 2016. "Sentiment Embeddings with Applications to Sentiment Analysis". *IEEE Trans. Knowl. Data Eng.*, (2015): 28 (2), 496-509.

[17] Yu, L.-C., Wang J., Lai, K. R. and Zhang X. "Refining Word Embeddings for Sentiment Analysis". *In Proceedings of the Conference on Empirical Methods in Natural Language Processing*, (2017): 545-550.

Current Address: Nergis PERVAN: Department of Computer Engineering, Ankara University, Ankara 06830, TURKEY
E-mail Address: *nergispervan@gmail.com*
ORCID: *https://orcid.org/0000-0003-3241-6812*

Current Address: Hacer YALIM KELEŞ : Department of Computer Engineering, Ankara University, Ankara 06830, TURKEY
E-mail Address: *hkeles@ankara.edu.tr*
ORCID: *https://orcid.org/0000-0002-1671-4126*