Sigma J Eng & Nat Sci 38 (3), 2020, 1581-1592

Sigma Journal of Engineering and Natural Sciences
Sigma Mühendislik ve Fen Bilimleri Dergisi

Research Article

CLASSIFICATION OF TURKISH TWEETS BY DOCUMENT VECTORS AND INVESTIGATION OF THE EFFECTS OF PARAMETER CHANGES ON CLASSIFICATION SUCCESS

Metin BİLGİN*[1]

[1]*Bursa Uludağ University, Department of Computer Engineering, BURSA;* ORCID: 0000-0002-4216-0542

Received: 12.11.2019 Revised: 11.05.2020 Accepted: 13.06.2020

ABSTRACT

Natural language processing is an artificial intelligence field which is gaining in popularity in recent years. To make an emotional deduction from texts related to an issue, or classify documents are of great importance considering the increasing data size in today's world. Understanding and interpreting written texts is a feature that pertains to people. But, it is possible to deduce from texts or classify texts using natural language processing which is a sub-branch of machine learning and artificial intelligence. In this study, both text classification was made on Turkish tweets, and text classification success of method parameter changes was investigated using two different methods of the algorithm mentioned as document vectors in the literature. It was found in the study that as well as higher accuracy values were obtained by the DBoW (Distributed Bag of Words) method than DM (Distributed Memory) method; higher accuracy values were also obtained by DBoW-NS (Negative Sampling) architecture than others.
Keywords: Text classification, natural language processing, document vectors, doc2vec, sentiment analysis, deep learning.

1. INTRODUCTION

Natural language processing is to collect information regarding the comprehension of human language and development of the ability to use it, so that it can be used to understand and process tasks, and to develop tools and techniques to use in this manner [1].

Natural language processing has developed as a branch of artificial intelligence and has become increasingly popular with the development of social media and web technologies. It contributes to the solution of many language-related problems, such as automated translation systems, question-and-answer systems, summarizing a topic, semantic role labeling, sentiment analysis, correction of spelling errors, and text classification.

Text classification (TC), is the assignment of texts formed in the natural language to predetermined groups by an expert. TC studies have started with the works of Maron in the 1960s [2]. However, since the area has become more and more popular since the 1990s, the development of technology and the subsequent access to strong sources of hardware, it has become a prominent sub-discipline of information systems [3]. TC is being used in many areas

* Corresponding Author: e-mail: metinbilgin@uludag.edu.tr, tel: (224) 275 52 63

such as the automatic classification of text-based documents, filtering of unwanted messages, developing results of search engines, determination of ideas, or author recognition [4].

Sentiment analysis (SA) is the ability to express sentiments, thoughts, ideas, and experiences regarding any topic on social media such as Facebook, Google+, Twitter, etc. SA is the determination of the sentiment reflected in the texts [5]. As an example, we can give positive or negative comments regarding a product, or the determination of the mood of users, and political ideas of the society [6]. Sentiment analysis studies categorize the content of every part of message content in the database into two (e.g., positive, negative) or more (e.g., very good, good, satisfactory, bad, very bad) categories; thus, SA can be considered a type of TC that in every message, the dominant sentiment represents a category [7].

In Zhang et al.'s work, positive-negative-tagged sentiment analysis was conducted on 100,000 Chinese interpretations of a product obtained from the trial using various classification and learning methods. In the study of two separate experiments by Lexicon and part of speech, the combination of Word2Vec and SVM perf was 89.95% on lexicon basis and 90.30% on part speech [8].

Dickinson et al. emotion analysis was conducted using n-gram and Word2Vec methods to investigate how a firm's stock price change affected the sentiment of the tweets about the product. As a result of the study, 65% success was achieved for the n-gram method and 75% success was achieved in the word2vec method [9].

Tang et al. conducted another emotion analysis on the twitters. Unlike the others in this study, it was tried to show how various word embedding methods changed the success of the system [10].

Polpinij et al. conducted another empirical analysis of hotel reviews. In this study where the Word2Vec method is used, SVM is used as a learning algorithm [11].

Şahin, on the Turkish texts, Word2Vec, and SVM based classification process were done. It was investigated how the processing of root or attached words affected the system performance in the study in which 22,729 Turkish documents were used [12].

Xue et al. performed emotion analysis using Word2Vec on Sina Weibo, a Chinese microblogging site. The other side of the work is a model for building an emotion dictionary using Word2Vec. Later, it was aimed to determine the sentiment tendency of documents using this dictionary. As a result of the study, success was obtained as 0.94 for positive documents, 0.96 for negative documents, and 0.85 for neutral documents [13].

Bilgin and Şentürk, on the other hand, have implemented semantic supervised sentiment analysis on Twitter for both English and Turkish. Two versions of the Doc2Vec algorithm, DBoW, and DB, were used in this study. As a result, DBoW showed higher performance than DM [14].

For the sentiment analysis they have made, Bilgin and Köktaş have used word vectors on a three-class data set. They have tested the performance of the system using two different machine learning methods, based on a Word2Vec approach [15].

Bilgin and Köktaş have carried out a study on sentiment analysis and have used 3000 Turkish tweets. They had used Term Weighted and Word Vectors as methods. Word Vectors were shown to be better than Term Weight in this study [16]. As well as these studies, lots of studies were carried out on Turkish Tweets [17-20].

This study is on Doc2Vec methods that are improved after Word2Vec. Doc2Vec is a novel approach which little study in it. One of the novel aspects of this study that the parameter changes are investigated on the accuracy metric. Also, 2 different methods of Doc2Vec are detailed examined, and present the pros and cons of the methods. To use of Turkish tweets created a reference point for studies on Turkish. Another novel aspect is one of the first studies to use Doc2Vec for Turkish.

2. EXPERIMENT AL METHODS

2.1. Doc2Vec

The method of word vectors was suggested to represent words as vectors in an n-dimensional space, and thus calculate the distances between words to determine the semantic similarities [21]. Word2Vec is an approach that uses a neural-network approach to embed words. This model is trained with a large set of texts and forms a unique vector for every word. The attribute of these unique vectors is that words with similar meanings form similar vectors. It has two methods: CBoW (Continuous Bag of Words) and Skip-gram. CBoW uses the context around a word to guess the word, whereas Skip-gram tries to guess the words by encircling the words that have a fixed window size. Skip-Gram can produce better results for sparsely used words [22]. CBoW and Skip-Gram methods are illustrated in Figure 2.

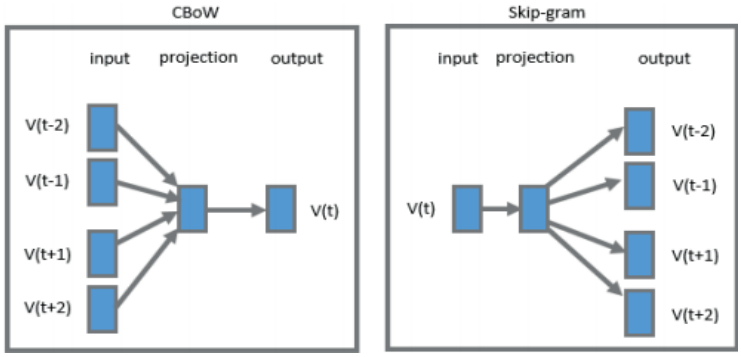

Figure 1. Word2Vec (CBoW and Skip-gram) [14]

Word2Vec has a version that is developed to embed the word sequence, called Paragraf2Vec in some sources. The purpose of this model is to use the word vectors model to represent larger blocks of text (sentence, paragraph, or document) as vectors [23]. The model that uses unsupervised learning has two different methods named DBoW and DM. DBoW uses the same method as Skip-gram, whereas DM uses the same method as CBoW. Having more parameters and consequently being more complex, DM can produce better results compared to DBoW. DBoW is simpler compared to DM and instead of the sequence of words, it is based on the sequence of words in a paragraph. Different from Skip-gram, DBoW replaces the document vector that represents the input. Differently from CBoW, DM uses the document vector as well as contextual words to guess the target word [24]. DM and DBoW methods are demonstrated in Figure 3.

Each of DBoW and DM architectures uses one of HS (hierarchical softmax) or NS training algorithms for learning. While HS training algorithm generally produces good results in low-frequency words, NS produces better results in high- frequency words. The parameters used for the Doc2Vec algorithm are given in Table 1.

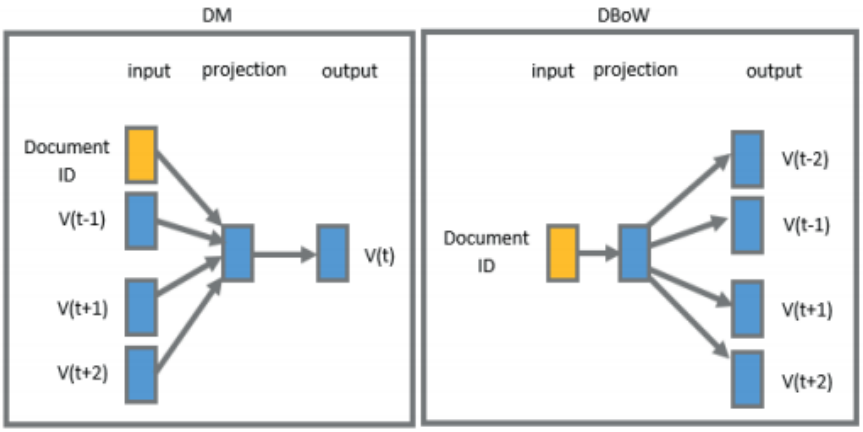

Figure 2. Doc2Vec (DM and DBoW) [14]

Table 1. Parameters and Tasks for Doc2Vec

Parameter	**Task**
-size	Length of vector
window	Number of the neighbor node
-hs	Learning Algorithm (hs=1 HS, hs=0 NS)
-negative	Number of Negative samples
-alpha	Learning Rate (alpha=0.025)
dm	Method (dm=1 DM, dm=0 DBoW)

A software was developed using the Gensim library in the Python programming language for the work to be done. The software we developed is designed to be supervised learning and able to train the system. Training and testing steps have been carried out by using DBoW and DM algorithms. Also, the accuracy value for the test set was calculated with the software, and confusion matrices were created.

2.2. Dataset

A three-class dataset, created by Amasyalı et al., was used in the study [25]. Randomly 10.000 tweets were selected for train and 2.500 tweets for the test from approximately 12.500 tweets. The information concerning the dataset is given in Table 2.

Table 2. Dataset's Properties

	Positive	**Negative**	**Neutral**
Train Set	3500	3250	3250
Test Set	1000	750	750

3. RESULTS AND DISCUSSION

In this study, different methods of the Doc2Vec algorithm were used for the classification of tweets we had. The optimum success of the system was tried to be measured by two different methods, including DM and DBoW, by making changes on parameters, and the effects of

parameter changes on success were investigated. The results are given in Table 3-4, and the graphs of the results in Figure 3-10.

Table 3. The Results for accuracy metric (-size =100)

DBoW	**-window**					
-negative	**3**	**4**	**5**	**6**	**7**	**-hs**
1	52.92	54.32	53.36	53.04	53.24	HS
	52.8	52.96	52.52	54.32	53.12	NS
2	52.08	52.36	53.64	52.2	52.96	HS
	55.68	54.84	54.88	54.36	55.8	NS
3	52.56	53.52	52.88	53.08	52.8	HS
	57.12	56.92	57.16	56.36	56	NS
4	52.2	51.44	52	52.08	53.32	HS
	57.56	57.16	56	56.16	57.28	NS
5	54.04	51.84	52.64	53.12	52.6	HS
	57.92	58.04	57.16	57.04	57.20	NS
DM	**-window**					
-negative	**3**	**4**	**5**	**6**	**7**	**-hs**
1	39.96	40.04	40.04	40.16	40.2	**HS**
	40.96	40.52	40.12	40.2	40.2	**NS**
2	40.08	40.08	40.08	40.04	40.04	**HS**
	41.04	40.96	40.2	40.2	40.2	**NS**
3	40.76	40.12	40.08	40.2	40.04	**HS**
	42.36	40.92	40.08	40.2	40.2	**NS**
4	39.92	40.16	40.32	40.2	40.2	**HS**
	42.6	40.96	40.04	40.2	40.2	**NS**
5	40.44	40.04	40.2	40.2	40.2	**HS**
	42.48	41.68	40.04	40.2	40.2	**NS**

The graph of the accuracy values obtained when DBoW and DM methods used the parameters of size=100, negative=1-5, window=3- 7 for HS ve NS architectures is given in Figure 3-6.

Table 4. The Results for (-size =500)

DBoW	**-window**					
-negative	**3**	**4**	**5**	**6**	**7**	**-hs**
1	42.56	42.8	43.8	42.96	42.76	**HS**
	40.12	40.20	40.16	40.16	40.24	**NS**
2	43.16	42.64	42.4	43.96	42.28	**HS**
	40.48	40.24	40.24	40.16	40.36	**NS**
3	42.48	42.88	42.52	43.64	42.92	**HS**
	40.2	40.4	40.6	40.8	40.32	**NS**
4	43.88	43.56	43.28	42.96	43.56	**HS**
	40.64	40.64	40.84	40.84	41.2	**NS**
5	42.32	43.16	43.88	43.48	44	**HS**
	41.24	41.28	41.12	41.28	40.92	**NS**
DM	**-window**					
-negative	**3**	**4**	**5**	**6**	**7**	**-hs**
1	39.28	40.76	40.08	39.52	39.84	**HS**
	40.2	40.2	40.2	40.2	40.2	**NS**
2	39.8	39.76	40.68	40.68	39.16	**HS**
	40.2	40.2	40.2	40.2	40.2	**NS**
3	39.2	40.4	40.28	39.8	39.92	**HS**
	40.2	40.2	40.2	40.2	40.2	**NS**
4	40.32	39.28	40.08	40.16	39.2	**HS**
	40.2	40.2	402	40.2	40.2	**NS**
5	39.48	40.04	39.92	39.36	39.76	**HS**
	40.2	40.2	402	40.2	40.2	**NS**

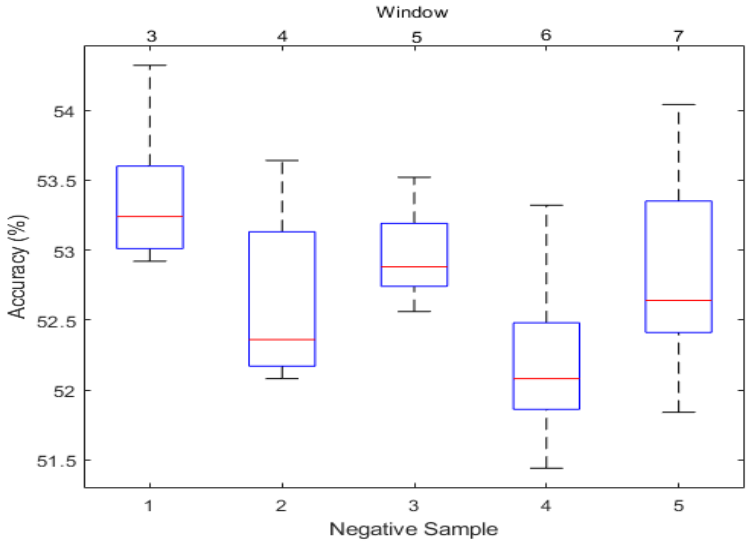

Figure 3. The Results for DBoW-HS (-size=100)

Since HS learning algorithm was applied for the DBoW method; increasing and decreasing accuracy values were obtained depending on the number of negative samples and on the change in the window. The change in the window by the same number of negative samples caused oscillation in the accuracy value.

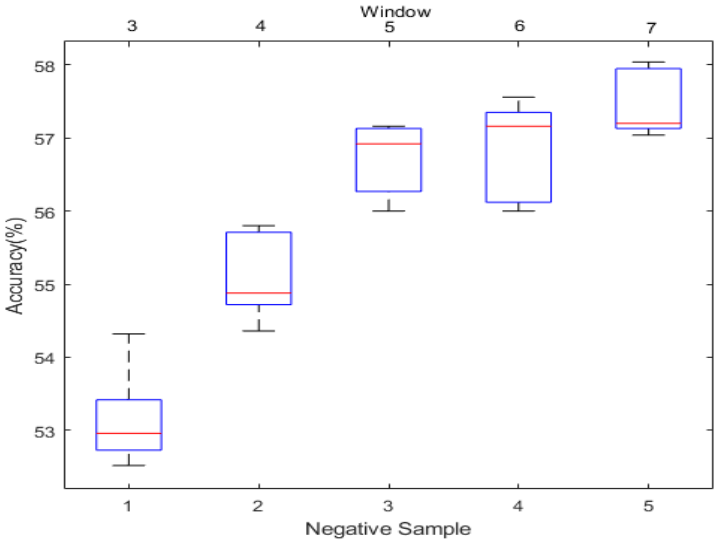

Figure 4. The Results for DBoW-NS (-size=100)

Since NS learning algorithm was applied for the DBoW method; increasing accuracy values were obtained depending on the number of negative samples and on the change in the window. The change in the window by the same number of negative samples caused little oscillation in the accuracy value.

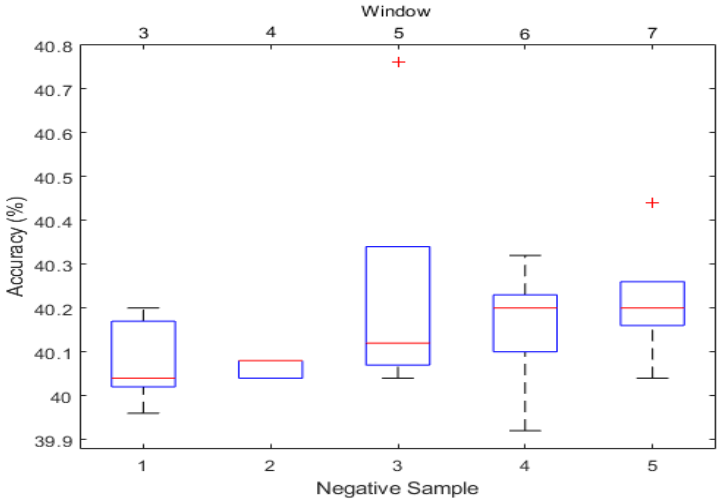

Figure 5. The Results for DM-HS (-size=100)

Since HS learning algorithm was applied for DM method; increasing and decreasing accuracy values were obtained depending on the number of negative samples and on the change in the window. The change in the window by the same number of negative samples caused changeable oscillation in the accuracy value.

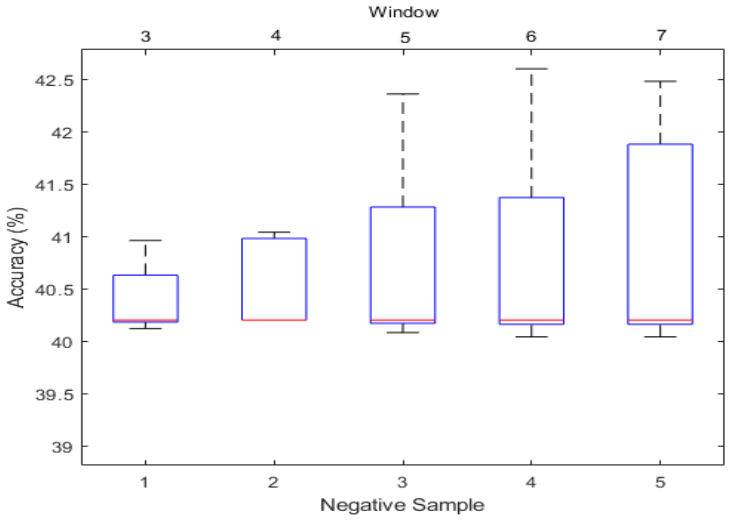

Figure 6. The Results for DM-NS (-size=100)

Since NS learning algorithm was applied for DM method; increasing accuracy values were obtained depending on the number of negative samples and on the change in the window. The change in the window by the same number of negative samples caused oscillation increasingly in the accuracy value.

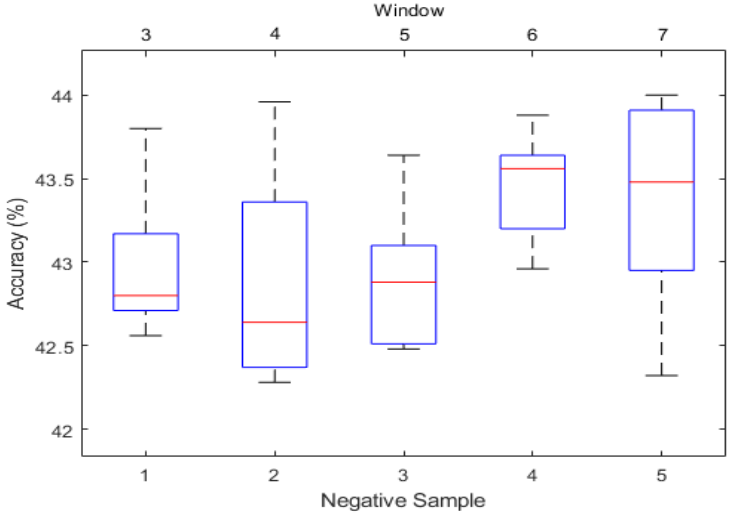

Figure 7. The Results for DBoW-HS (-size=500)

Since HS learning algorithm was applied for the DBoW method; similarly accuracy values were obtained depending on the number of negative samples and on the change in the window. The change in the window by the same number of negative samples caused changeable oscillation in the accuracy value.

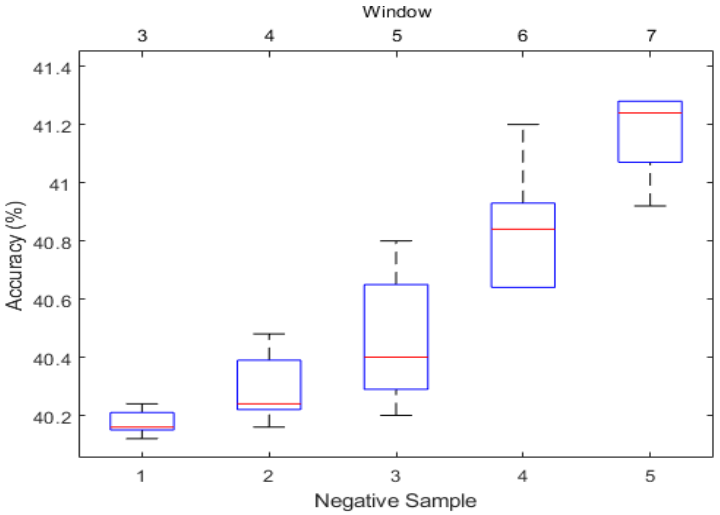

Figure 8. The Results for DBoW-NS (-size=500)

Since NS learning algorithm was applied for the DBoW method; increasing accuracy values were obtained depending on the number of negative samples and on the change in the window. The change in the window by the same number of negative samples caused little changeable oscillation in the accuracy value.

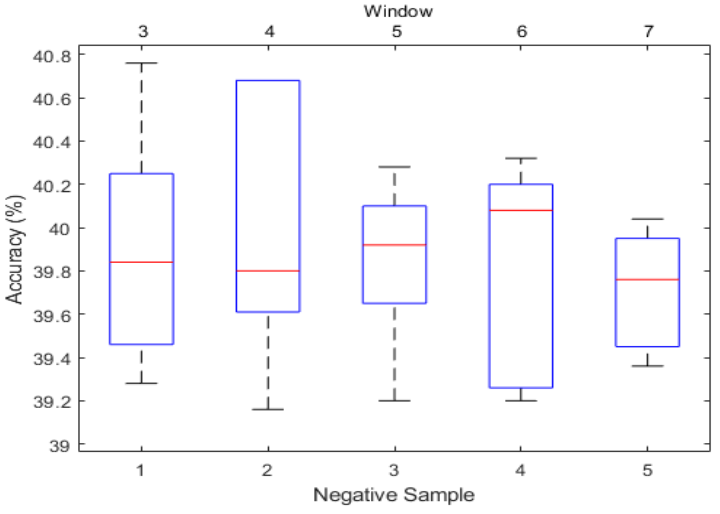

Figure 9. The Results for DM-HS (-size=500)

Since HS learning algorithm was applied for the DM method; decreasing accuracy values were obtained depending on the number of negative samples and on the change in the window. The change in the window by the same number of negative samples caused changeable oscillation in the accuracy value.

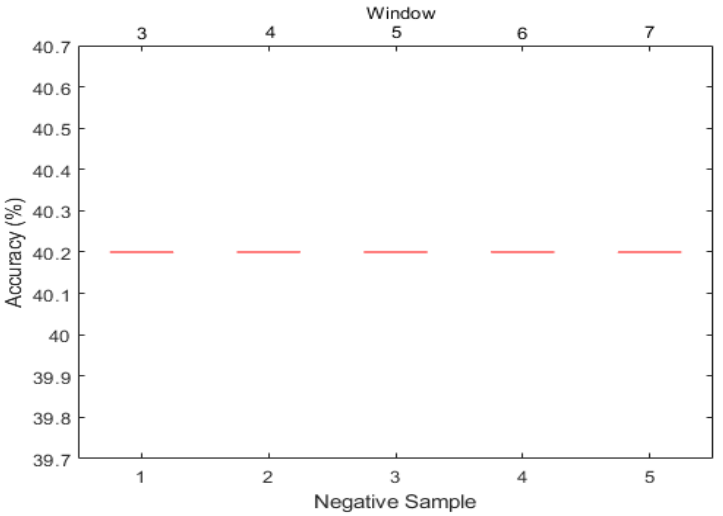

Figure 10. The Results for DM-NS (-size=500)

Since NS learning algorithm was applied for the DM method; likely accuracy values were obtained depending on the number of negative samples and on the change in the window. The change in the window by the same number of negative samples caused oscillation in the accuracy value.

In the study in which vector length variation was investigated; DBoW reacted negatively to the increase in the size, and accuracy value decreased. Although NS architecture managed to produce better values when the window parameter was increased along with the size parameter, these values were not better than those obtained with size=100. Although similar results were obtained in DBoW with HS and NS, with size = 100 and negative = 1; NS architecture produced better results as long as the negative value increased and even DBoW reached the highest accuracy value of 58.04 with the parameters of size=100, negative=5, window=4. In DM, both HS and NS reacted similarly to the size increase and produced similar results. The oscillation slightly increased in both DM-HS and DM-NS when size=500 in comparison to when size=100. When particularly an evaluation, independent of size, is made, it can be said that the accuracy rate of DM-NS was found 40.2, and this rate didn't oscillate much from the changes in both negative and window parameters, and it produced the same results.

Considering the findings of the study, it can be said that; DBoW achieved higher accuracy values than DM. DBoW-NS reacted more negatively to the increase in size than DBoW-HS. As a result of the study, it was seen that the highest accuracy value for DBoW was obtained both when size=100, window=3-4-5, negative=3-4-5, and when NS architecture was used. DM gave similar reactions to the changes in size, window, and negative parameters and produced similar accuracy values without oscillating so much. The highest accuracy value of 42.48 was obtained with size=100, window=3,negative=1.

The reason why better results were obtained with DBoW method than DM method is that the calculation cost of DBoW is less in small spaces and in parallel to that, higher accuracy values

are expected to be obtained. As space, where processing was done, grew, process complexity increased, and accordingly, the accuracy values of DBoW approximated to DM. The same situation is valid for HS and NS architectures. NS produced better results than HS as the size parameter increased. As the size parameter increased, the values obtained by the DBoW method for NS were less than those obtained for HS. In the DM method, a little change occurred in the parameter-dependent accuracy values.

4. CONCLUSIONS

In this study, text classification was performed through the data obtained from twitter using two different methods of the Doc2Vec algorithm. A train was given using supervised learning infrastructure and then tests were conducted for a 3-class dataset by using the Gensim library in Python programming language. SVM algorithm used for the classification. Different learning architectures and different parameter values were used to find the most appropriate parameter range for the system. The success of the system was tried to be increased with these changes.
As a result of the study, it was seen that better results were obtained by DBoW method than DM method. Higher accuracy values were obtained by NS architecture than HS. The accuracy of the optimized parameter values obtained in this study will be tested in future studies to be performed on the data sets of different sizes.

REFERENCES

[1] Chowdhury. G.G., "Natural language processing", Annual review of information science and technology,37,1, 51-89, 2005.
[2] Maron, M.E., "Automatic indexing: an experimental inquiry", Journal of the ACM,8,3,404-417,1961.
[3] Fabrizio, S., "Machine learning in automated text categorization", ACM computing surveys,34,1,1-47,2001.
[4] Dalal, M.K., Mukesh, A. Z., "Automatic Text Classification: A Technical Review", International Journal of Computer Applications, 28,2,37-40, 2011.
[5] Sommer, S.,Schieber, A., Hilbert, A., Heinrich K., "Analyzing customer sentiments in microblogs–A topic-model-based approach for Twitter datasets", Americas conference on information systems (AMCIS),Detroit, Michigan, USA, (2011), 1-7.
[6] Liu, B, Lei, Z., "Mining Text Data: A survey of opinion mining and sentiment analysis", Mining Text Data, Springer, Boston, USA, 2012, 415-463.
[7] Prabowo, R., Thelwall, M., "Sentiment analysis: A combined approach", Journal of Informetrics, 3,2, 143-157, 2009.
[8] Zhang, D., Xu H., Su, Z., Xu, Y., "Chinese comments sentiment classification based on word2vec and SVMperf", Expert Systems with Applications, 42,4,1857-1863,2014.
[9] Dickinson, B., Wei, H., "Sentiment analysis of investor opinions on twitter", Social Networking, 4,3, 62-71,2015.
[10] Tang, D., Wei, F., Yang, N., Zhou, M., Liu, T., Qin, B., "Learning sentiment-specific word embedding for twitter sentiment classification", 52nd Annual Meeting of the Association for Computational Linguistics, Baltimore, Maryland, USA, (2014), 1555-1565.
[11] Polpinij. J., Natthakit, S., Paphonput, S., "Word2Vec Approach for Sentiment Classification Relating to Hotel Reviews", 13th. International Conference on Computing and Information Technology, Bangkok, Thailand, (2017), 308-316.
[12] Şahin, G., "Turkish document classification based on Word2Vec and SVM classifier", Signal Processing and Communications Applications Conference, Antalya, Turkey, (2017), 1-4.

[13] Xue, B., Chen, F. and Zhan, S., "A study on sentiment computing and classification of sina weibo with word2vec", IEEE International Congress on Big Data, Alaska, USA, (2014), 358-363.

[14] Bilgin, M. and Şentürk, İ. F., "Sentiment analysis on Twitter data with semi-supervised Doc2Vec", 2nd. International Conference on Computer Science and Engineering, Antalya, Turkey, (2017), 661-666.

[15] Bilgin, M. and Köktaş, H., "Word2Vec Based Sentiment Analysis for Turkish Texts", International Conference on Engineering Technologies, Konya, Turkey, (2017), 106-109.

[16] Bilgin, M. and Köktaş, H., "Sentiment Analysis with Term Weighting and Word Vectors", International Arab Journal of Information Technology, 16,5, 953-959, 2019.

[17] Ayata, D., Saraçlar, M. and Özgür, A., "Turkish tweet sentiment analysis with word embedding and machine learning", 25th Signal Processing and Communications Applications Conference (SIU), Antalya, Turkey, (2017), 1-4.

[18] Yüksel, A.E., Türkmen, Y.A., Özgür, A. and Altınel, B., "Turkish Tweet Classification with Transformer Encoder, International Conference on Recent Advances in Natural Language Processing", Varna, Bulgaria, (2019), 1380-387.

[19] Akkol, E., Alıcı, S., Aydın, C. and Tarhan, Ç., "What Happened in Turkey After Booking.com Limitation: Sentiment Analysis of Tweets via Text Mining", Economic and Financial Challenges for Balkan and Eastern European Countries, 291-301, 2020.

[20] Akkol, E., Alıcı, S., Aydın, C. and Tarhan, Ç., "Sentiment Analysis of How Turkish Customers Affected by PayPal Closure", Economic and Financial Challenges for Balkan and Eastern European Countries, 303-313, 2020.

[21] Mikolov. T., Chen, K., Corrado, G., Dean, J., "Efficient estimation of word representations in vector space", International Conference on Learning Representations, Scottsdale, Arizona, USA, (2013), 1-12.

[22] Campr, M., Karel, J., "Comparing semantic models for evaluating automatic document summarization", International Conference on Text, Speech and Dialogue, Pilsen, Czech Republic, (2015), 252-260.

[23] Kamkarhaghighi. M., Masoud, M., "Content Tree Word Embedding for document representation", Expert Systems with Applications, 90,241-249,2017.

[24] Le. Q., Mikolov, T., "Distributed representations of sentences and documents", International Conference on Machine Learning, Beijing, China, (2014),1188-1196.

[25] Amasyalı, M.F., Taşköprü, H., Çalışkan, K. (2019) Duygu durum Analizinde Kelimeler, Anlamlar, Karakterler [Internet] Yıldız Technical University. Available from:http://www.kemik.yildiz.edu.tr/data/File/17bintweet.zip [accessed November 10, 2019].