Sentiment analysis Using Neural Network Models

Aynur Islamovich Akhmetgaliev, Fail Mubarakovich Gafarov and
Farida Bizyanovna Sitdikova

Abstract--- *The article deals with methods for solving the problem of sentiment analysis based on neural network models of natural language processing. The article considers methods that create a vector representation of words in the n-dimensional vector space, which are based on "Word2Vec", "GloVe", "FastText" technology. Approaches are used in the tasks of classification, sentiment analysis, typo correction, recommendation systems. We present the results of classifications comparison in the problem of sentiment analysis of a multilayer perceptron, a convolutional and recurrent neural network, decision trees (random forest), support vector machine (SVM), naive Bayes classifier (NB), and k-nearest neighbors (K-NN). The results of the classification are presented for three data sets: Twitter messages, reviews of various goods and services, Russian-language news.*

Keywords--- *Sentiment Analysis, Word2Vec, GloVe, FastText, Vector Word Representation, Recurrent Neural Networks, Convolutional Neural Networks.*

I. INTRODUCTION

Sentiment analysis is a class of natural language processing methods designed for automated recognition in texts of emotionally colored vocabulary and emotional evaluation (positive, negative, neutral) [1]. In this paper, sentiment analysis is considered as a special case of the document classification problem. Document classification is one of the tasks of information retrieval, which consists in assigning a document to one of several categories based on the content of the document [2].

Doing automatic sentiment analysis has an important practical application in various areas of human activity: running business, determining political and economic strategies, in assessing the quality of products and services in social networks, blogs of targeted sites. Understanding how negatively or positively people react to the product or service produced allows us to evaluate the future success of sales in the market, as well as to analyze the quality of the PR compaigns conducted. Many large producers need to evaluate how their brand is perceived by consumers.

Sentiment analysis can also be actively used in determining the population’s attitude to various reforms carried out by the government or the electorate’s opinion of a particular candidate for a leadership position. Thus, sentiment analysis is a powerful tool used by scientists, businessmen and politicians.

II. METHODS

Machine learning techniques consider sentiment analysis as a text classification task. These methods use extracting functions such as unigrams, bigrams, word embeddings, first from text, or words , and then documents are vectorized.

Aynur Islamovich Akhmetgaliev, Master Student, Kazan Federal University, Kazan, Kremliovskaya, Russian Federation.
Fail Mubarakovich Gafarov, PhD, Associate Professor, Kazan Federal University, Kazan, Kremliovskaya, Russian Federation.
Farida Bizyanovna Sitdikova, PhD, Associate Professor, Kazan Federal University, Kazan, Kremliovskaya, Russian Federation.
E-mail: farida7777@yandex.ru

Received: 12 Nov 2018 | Revised: 22 Dec 2018 | Accepted: 06 Jan 2019

Vectors are fed to classification models such as SVM, decision trees, or deep neural networks (CNN, RNN). The accuracy of the classification is highly dependent on the size and “quality” of the training data set. This article focuses on methods of supervised learning.

Vector representations of words

Word2Vec. The Google’s Word2Vec tool (set of algorithms) was used to solve the document classification problem created by Thomas Mikolov. The **word2vec** tool takes the text corpus as input and produces vectors representations of words as output. [3] Vectors of similar words have a scalar product close to one. Thus, Word2Vec tool allows you to find words that are close in meaning. In our research the following text corpora have been used: "National corpus of the Russian language" (788 million words), Russian-language news for December 2018 (5 billion words), Wikipedia (2.6 billion words). For the sentiment analysis task Word2Vec vectors were trained on the basis of untagged tweets of 17,639,674 messages. Top 5 words are close to the word “interesting” on the tweeter data: [('good', 0.7640383243560791), ('funny', 0.7558714151382446), ('curious', 0.7503457069396973), ('cognitive', 0.7198209762573242), ('cool', 0.7184607982635498)]

FastText. FastText is a way to build a vector representations from Facebook. In FastText, as in the Word2Vec model, skip-gram or cbow approaches are used. The main idea is the same as in Word2Vec: to maximize the probability (softmax) to meet the central word with contextual or vice versa, running through the corpus with a given window width. The main difference between FastText and word2vec or Glove is that each word is represented as a n-gram character set. Let us give an example with n = 3, the word “school” is divided into the following 3-grams <sc, sch, cho, hoo, ool, ol>. The authors of FastText recommend choosing n from 3 to 6, the optimal n depends on the language, body and task. [4] Suppose we have a dictionary $G_w \subset \{1, \dots, G\}$ n-gram, then we associate a vector representation z_g with every n-gram g. And we represent the word as the sum of the vector representations of its n-grams. [5] Then the proximity of two words in FastText is defined as

$$s(w, c) = \sum_{g \in G_w} z_g^T \, v_c$$

We should note that learning in FastText is faster than in the word2vec model.

Here are top 5 words similar to the word ‘good’ in FastText which was trained on the news text corpus: [('excellent', 0.7849140167236328), ('fair', 0.7302890419960022), ('bad', 0.7082219123840332), (‘superb’, 0.6944824457168579), ('lovely', 0.6733499765396118)].

GloVe. Global Vectors is a model for distributed vector representation of words. It is a unsupervised learning algorithm. Training is performed on an aggregated global frequency matrix of words (co-occurrence matrix), which is built from the corpus. GloVe is just like Word2Vec allows you to find words that are close in meaning, that is, word vectors closest in value will be next to each other in vector space.

The first stage is the construction of the so-called co-occurance matrix - the matrix showing how often words meet each other in a given corpus, where the matrix element X_{ij} is the number of occurrences of the word ***j*** next to ***i*** in context. The GloVe model learns on non-zero matrix elements. To complete the given matrix one pass through

Received: 12 Nov 2018 | Revised: 22 Dec 2018 | Accepted: 06 Jan 2019

the entire corpus is necessary. Suppose $X_i = \sum_k X_{ik}$ is the number of times when any word was found next to the word i.

$P_{\downarrow}ij = P(j \dashv| \ i) = X_{\downarrow}ij/X_{\downarrow}i$ is the probability that the word ***j*** appears in the context of the word ***i*** **.** Finding word vectors u_i for ***i*** = 1, W, where W is the size of the dictionary, is equivalent to the solution the minimization task of the function presented below [6].

$$J = \frac{1}{2} \sum_{i,j=1}^{W} f(P_{ij})(u_i^T u_j - logP_{ij})^2$$

Top 5 words that are similar to the word “furniture” in GloVe, trained on the corpus of feedbacks: [('furniture', 0.5776834487915039), ('interiors', 0.4908282160758972), ('glamorous', 0.4803837537765503), ('mattresses', 0.4794725179672241), ('design', 0.4584319591522217)].

III. Results

Text classification by using sentiment analysis. First you need to get a vector representation of the words using any of the technologies (Word2vec, GloVe, FastText) for each word from the corpus dictionary. Next, we find the average fixed-dimension vector for each document from the training set, (that is, summing up all the word vectors of the document and dividing by the number of words in the document). The obtained aggregated vectors of documents with the corresponding labels of belonging to a particular class are submitted for training to the classifier. Next accuracy of the classification is determined on the test set . But when finding the average vector the word order in the document is not taken into consideration. Therefore, when summing up word vectors, you can multiply them by some “weight”, for example, the term TF-IDF. When doing the sentiment analysis of a text, experiments were conducted with a multilayer perceptron with two hidden layers of 6 neurons per layer. The dimension of the input vector was equal to the dimension of the document vector. The number of neurons in the output layer was 2 or 3, depending on how many classes of sentiment classification the separation took place. As a training method, the method of stochastic gradient descent was used. We also used the classical methods of Data-mining, such as the SVM support vector machine [7], the k-nearest neighbors algorithm (k-NN), the Random Forest (Random Forest), the Naive Bayes Classifier (Naive Bayes). As classifiers, convolutional and recurrent neural networks were used. The convolutional network shows good results in image recognition and is currently actively used in natural language processing tasks.

Consider how CNN is used in the task of classifying documents. Suppose we built a vector representation of the words for each term from the dictionary using one of the Word2Vec, GloVe or FastText models. We compose a matrix from these word vectors, the so-called embedding layer, for each document from the training set. Since CNN accepts a fixed-length matrix as input, we trim our documents to a fixed length. If, when analyzing, the number of words exceeded the dimension of the matrix, the remaining words were removed. The length of the document is chosen so that a large percentage of all the words of the texts in the formed body are covered.

Thus, the dimensions of the matrix n x k, n are the fixed number of words in the document, k is the dimension of the vector. We have used various architectures of convolutional neural networks.[8] One of the most accurate ones was a network in which filters of dimensions from 2-5 to 10 convolutional layers for each filter were used. After using the convolutional layers, the max-pooling operation was applied. Then the layers were concatenated into a common vector, which was fed to the input of a fully connected layer of 40. [9]

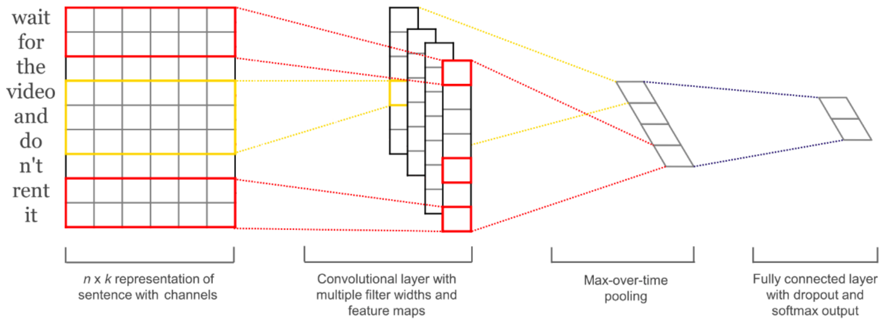
wait
for
the
video
and
do
n't
rent
it
n x k representation of sentence with channels
Convolutional layer with multiple filter widths and feature maps
Max-over-time pooling
Fully connected layer with dropout and softmax output

Fig.1: CNN. The classification uses convolutions of dimension 1

We used 10 epochs of training and also 5 additional ones with epoch weight in which the epoch had received the highest F1-score. The volume of training and test samples was 80 and 20 percent, respectively. And the size of the validation sample was taken 25% of the training one. As a classifier, the Long Short Term Memory network LSTM network was also used (a kind of recurrent neural network that keeps and considers the previous information). Some words sequence is transmitted as the input of the RNN network. Memorization means that the RNN performs the same task for each element of the sequence, where each output depends on previous calculations. The length of the time steps is determined by the length of the input. For example, if the word sequence being processed is a 4-word sentence, then the RNN will be deployed into a neural network with 4 time layers.[10] One layer corresponds to a word. The sequence of word vectors of the document was fed to the input of the LSTM, where the number of time layers coincided with a fixed size of terms in the document. RNN training requires a large amount of data. RNN is prone to retraining, so the number of training epochs is normally not large.

IV. DISCUSSION

Experiments in sentiment analysis classification were performed on 3 different data sets, which are described below.

A selection of reviews. A special bot was written to collect text reviews from the *irecommend.ru* website, passing through various categories: cinema, books, cars, beauty and health, tourism, equipment, animals. Each review had a rating from 1 to 5. Reviews with a rating of less than 4 were negative, the rest were positive. Thus, a sample of 47,847 reviews was collected. Out of the whole review the most significant part (conclusions and summary) were selected. The training sample size was 27759 examples, the validation sample size was 9253 examples. Before starting the training superfluous symbols and words that do not have semantic value (stop words) were removed from the sample, for example, particles, prepositions, punctuation marks, and symbols from other languages. All words were reduced to lower case, to normal form with the help of special morphological analyzers. During the testing all data pass the same preliminary processing algorithm as the training data. For each review, an aggregated vector was found for all types of words vectorization described in W2V, FastText, Glove. Also statistical methods Bag of Words, TF-IDF, n-grams were used. The results of the classification of reviews are presented in the table. F1- score was used as a metric of quality classification.

Table 1: F1-score on the reviews test

	Bag of Words	N - grams	TF-IDF	W2V (avg)	FastText (avg)	GloVe (avg)
MLP	0.65	0.73	0.81	0.78	0.77	0.76
RF	0.62	0.72	0.79	0.76	0.74	0.75
NB	0.62	0.72	0.74	0.74	0.73	0.73
KNN	0.64	0.73	0.77	0.75	0.73	0.74
SVC	0.65	0.74	**0.81**	0.78	0.77	0.77
CNN	0.64	0.72	0.78	***0.81***	**0.80**	***0.78***
RNN (LSTM)	0.62	0.70	0.75	0.76	0.75	0.75

Thus, the most accurate was the classification with TF-IDF methods, with the SVM classifier (support vector machine) and Word2Vec vector representation in conjunction with the convolutional neural network. They have given the best F1-score.

The recurrent neural network has not produced the best result, since it requires much more training data for training than other classifiers.

The news sample. The news sample was found at the kaggle.com machine learning competition.[11] The dataset was divided into 3 classes: negative, positive, neutral news. As a result, neutral news data prevailed in the sample, so some negative news was added from the “Accident” amend item of the Ria News news portal. The training sample contained 18798 news. The best result was given by the statistical TF-IDF method with the vector of support SVM vectors**.**

Table 2: F1-score for a set of news data

	Bag of Words	N - grams	TF-IDF	W2V (avg)	FastText (avg)	GloVe (avg)
MLP	0.55	0.63	0.72	0.73	0.71	0.67
RF	0.53	0.62	0.70	0.71	0.70	0.68
NB	0.52	0.62	0.68	0.72	0.69	0.69
KNN	0.54	0.63	0.69	0.70	0.70	0.69
SVC	0.55	0.64	**0.77**	0.73	0.73	0.70
CNN	0.52	0.62	0.70	*0.73*	0.73	0.72
RNN (LSTM)	0.52	0.60	0.65	0.66	0.63	0.62

Received: 12 Nov 2018 | Revised: 22 Dec 2018 | Accepted: 06 Jan 2019

Let us consider Fig.2. Error matrix for selecting news from the SVM classifier. As it can be seen from the error matrix, errors occur most often when classifying neutral news. The classifier considers neutral news to be positive or negative, since the definition of sentiment analysis is a relatively subjective classification, even a human being does not always cope with this task correctly. But a classifier with minimal error distinguishes negative news from positive ones.

Selection of Twitter messages. A sample of Russian-language social networking messages on Twitter was found in source [12]. It contains 15 million records obtained using the Twitter API. Vector representations of words for W2V, FastText, Glove. Training sample size 183,521 have been trained on it. Training sample size was as big as 183,521 tweets.

Table 3: F1-score for the Twitter dataset

	Bag of Words	N - grams	TF-IDF	W2V (avg)	FastText (avg)	GloVe (avg)
MLP	0.63	0.65	0.75	0.75	0.75	0.74
RF	0.62	0.64	0.74	0.74	0.74	0.73
NB	0.62	0.64	0.74	0.73	0.72	0.74
KNN	0.60	0.63	0.73	0.74	0.73	0.73
SVC	0.63	0.65	**0.76**	0.75	0.75	0.74
CNN	0.62	0.62	0.74	**0.77**	**0.77**	***0.74***
RNN (LSTM)	0.61	0.60	0.72	0.76	0.75	0.74

Thus, the best F1-score 0.77 was shown by a convolutional neural network with a W2V vector representation. Also, a close result was shown by SVM with static TF-IDF. Data sets may contain some errors, that is, examples from the sample contain instances that are incorrectly marked, which affects the accuracy of the classification.

V. SUMMARY

The paper considered methods for solving the problem of document classification, in particular, the task of sentiment analysis. Standard statistical models have been compared with modern models of vector representations of words, such as Word2Vec, FastText, Glove. Vector representations of words on various Russian-language corpore have been obtained. The properties of vectors were checked to find similar words by their meaning. We have used 3 training samples: a sample of reviews on various goods and services, a selection of Twitter messages, a selection of news. In the task of sentiment analysis, various types of classifiers were compared: MLP (multi-layer neural network), Random Forest (Random forest), NB (Naive Bayes classifier), KNN (K-neighbor method), SVM (support vector method), CNN (convolutional neural network), LSTM (recurrent network with long short-term memory).

VI. CONCLUSIONS

The results of our research has demonstrated that a convolutional neural network with a bunch of Word2Vec shows the best accuracy (F1-score) on medium-sized samples. FastText and Glove also give similar results. The statistical TF-IDF method also gives good accuracy in conjunction with the SVM reference vector method on all data sets used in the work. Thus, statistical methods are not always inferior to methods based on the vector representation of words. Note that the definition of sentiment analysis is inherently a subjective task. Classifiers are trained according to the intuition of the user who has tagged this data. Therefore, in the task of a text sentiment analysis, an acceptable accuracy is considered to be the F1-score more than 0.70. We have managed to exceed such accuracy in this work on all the presented training samples. In the future, it is possible to experiment with various architectures of convolutional neural network, changing the number of training epochs, the size of the training set, changing the parameters of training methods, as well as changing the various text preprocessing and text tokenization methods etc.

ACKNOWLEDGEMENTS

The work is performed according to the Russian Government Program of Competitive Growth of Kazan Federal University.

BIBLIOGRAPHY

[1] Kim, E., & Klinger, R. (2018). A survey on sentiment and emotion analysis for computational literary studies. *arXiv preprint arXiv:1808.03137*.

[2] Abdullah, M., & Zamil, M. G. (2018). The Effectiveness of Classification on Information Retrieval System (Case Study). *arXiv preprint arXiv:1804.00566*.

[3] Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., & Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, 3111-3119.

[4] Bojanowski, P., Grave, E., Joulin, A., & Mikolov, T. (2017). Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, *5*, 135-146.

[5] Joulin, A., Grave, E., Bojanowski, P., & Mikolov, T. (2016). Bag of tricks for efficient text classification. *arXiv preprint arXiv:1607.01759*.

[6] Pennington, J., Socher, R., & Manning, C. (2014). Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, 1532-1543.

[7] Bishop, C.M. (2006). *Pattern recognition and machine learning*. Springer.

[8] Zhang, Y., & Wallace, B. (2015). A sensitivity analysis of (and practitioners' guide to) convolutional neural networks for sentence classification. *arXiv preprint arXiv:1510.03820*.

[9] Kim, Y. (2014). Convolutional neural networks for sentence classification. *arXiv preprint arXiv:1408.5882*.

[10] Zhang, L., Wang, S., & Liu, B. (2018). Deep learning for sentiment analysis: A survey. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, *8*(4), e1253.

[11] Sentiment Analysis in Russian [URL]: https://www.kaggle.com/c/sentiment-analysis-in-russian/leaderboard (Accessed: 2.06.19).

[12] Y.V. Rubtzova Postroyeniye korpusa textov dlya natroyki tonovogo klassifikatora. №1(109) – Pp.72-78. [URL]: http://study.mokoron.com (Accessed: 2.06.19).

Received: 12 Nov 2018 | Revised: 22 Dec 2018 | Accepted: 06 Jan 2019