TRADITIONAL OR DEEP LEARNING FOR SENTIMENT ANALYSIS: A REVIEW

Aadil Gani Ganie
PhD Student, Institute of Information Sciences, University of Miskolc
3515 Miskolc, Miskolc-Egyetemváros, e-mail: aadilganiganie@gmail.com

Samad Dadvandipour
associate professor, Institute of Information Sciences, University of Miskolc
3515 Miskolc, Miskolc-Egyetemváros, e-mail: aitsamad@uni-miskolc.hu

Abstract
Getting the context out of the text is the main objective of sentiment analysis. Today's digital world provides us with many data raw forms: Twitter, Facebook, blogs, etc. Researchers need to convert this raw data into useful information for performing analysis. Many researchers devoted their precious time to get the text's polarity using deep learning and conventional machine learning methods. In this paper, we reviewed both the approaches to gain insight into the work done. This paper will help the researchers to choose the best methods for classifying the text. We pick some of the best articles and critically analyze them in different parameters like dataset used, feature extraction technique, accuracy, and resource utilization.

Keywords: *Machine learning, deep learning, sentiment analysis, NLP*

1. Introduction

Natural language processing with opinion mining helps the researchers explore new ways to comprehend the text's sentiments in a better way. Sentiment analysis is getting context out of the text. Classification and emotion analysis of the text is a prevalent problem of machine learning and is used in many tasks such as product forecasts, movie recommendations, and many others. When humans approach a text to decide whether a portion of it is positive or negative or marked by some other complex emotion, such as surprise or disgust, we use our interpretation of words' emotional intent. We can use text mining software to programmatically approach the emotional content of the text. According to (Feldman, 2013), the analysis of sentiment is a method in which the dataset consists of feelings, behaviors, or evaluations that consider the way a person thinks. Polarity classification can be done at various levels such as Document level, Sentence level, and Aspect or Feature level (Feldman, 2013). Researchers can use any classification level, which suits their model best Sentiment analysis approaches can primarily be classified (Medhat et al., 2014) as machine-learning (Sebastiani, 2002), Lexicon-based (Taboada et al., 2011) hybrid (Prabowo and Thelwall, 2009; Dang et al., 2009), and deep learning approach we consider two most prominent methods, i.e., traditional machine learning and deep learning approach.

In various IR applications, conventional neural networks have also been successfully implemented, e.g., (Shen et al., 2014 a; Shen et al., 2014 b). Deep learning has appeared in many application domains, ranging from Natural language processing and speech recognition to image classification, as a most advanced machine learning tool (Goodfellow et al., 2016) and offer state-of-the-art results. It has also recently become ubiquitous to apply deep learning to sentiment analysis. Like other review papers, we will not define the terms like a neural network, SVM, Random forest, etc., as they are known to everyone. The rest of the paper is categorized into a Literature review, conclusion, and references. We mainly focus on literature review as this is the central part of this paper.

2. Literature review

Sentiment analysis has gained its attention with the advancements in NLP. Multiple techniques have been adopted to classify the sentiment into negative, positive, and neutral. However, there is no substantiated effect on accuracy. Recently many researchers move towards deep learning methods. We will compare the traditional methods with the deep learning methods to overview advancements done in this field.

2.1. Machine learning methods

In (Wawre and Deshmukh, 2016), a classification technique was applied to the movie review dataset; they used only two supervised machine learning algorithms, Naïve Bayes and support vector machine. They consider two sentiments only negative and positive, dropping the neutral sentiment. Results show that Naïve Bayes outperforms SVM. They also conclude that with a more significant dataset accuracy of Naïve Bayes increases.

Table 1. Movie review sentiment analysis at document level

Algorithm	Accuracy	Feature extraction	Dataset	Number of sentiments
SVM	45.71	Document level	IMDB	2
Naïve Bayes	65.75	Document level	IMDB	2

This paper uses a limited number of algorithms and sentiments; the accuracy achieved is significantly less. The reason for less accuracy is feature extraction technique and lack of model tuning. Sentiment analysis was carried out by Gautam, G., & Yadav, D in (Gautam and Yadav, 2014) used tweet dataset with two types of sentiments negative and positive they also did not consider the neutral tweets. The only addition they did compare to (Wawre and Deshmukh, 2016) was to include a feature vector based on adjectives in the data. They also increased the number of algorithms; Support vector machine, Maximum entropy, and Naïve Bayes. WordNet helps in extracting phrases and similarities for the content feature. Among the different algorithms, Semantic analysis proves to be effective with 89.9 percent accuracy.

Table 2. Sentiment analysis using uni-gram approach

Algorithm	Accuracy	Feature extraction	Dataset	Number of sentiments
SVM	85.4	Uni-gram	Tweeter	2
Naïve Bayes	88.2	Uni-gram	Tweeter	2
Maximum Entropy	83.9	Uni-gram	Tweeter	2
Sematic Analysis	89.9	Uni-gram	Tweeter	2

This paper also neglected the neutral tweets, and feature extraction has been limited to uni-gram only. For better accuracy, bi-gram, tri-gram, and n-gram prove useful because of easement in context extraction. They used only one dataset, which again cannot specify the legitimacy of any algorithm's accuracy. (Le and Nguyen, 2015) Added a new feature package on the social networking platform focused on Knowledge Gain, Bigram, Object-oriented extraction techniques in sentiment analysis. The two Naïve Bayes and Support vector machine algorithms were also used by the researchers, with a bi-gram approach for feature extraction instead of the uni-gram method. (Gautam and Yadav, 2014) In (Le and Nguyen, 2015), they used three datasets to evaluate the model; algorithms with their accuracies are tabled below.

Table 3. Tweet Sentiment analysis

Algorithm	Accuracy	Feature extraction	Dataset	Number of sentiments
SVM	79.54	Uni-gram, bi-gram, object-oriented	Tweet	2
Naïve Bayes	79.58	Document-level	Tweet	2

As we can observe SVM classifier outperforms Naïve Bayes in this model. It may be due to the proper training of models with various datasets and various feature extraction techniques.

(Neethu and Rajasree, 2013) In the tweets, they dealt mostly with misspelling and slang. An efficient feature vector is generated to deal with these problems by doing feature extraction in two steps after proper pre-processing phase. In the first step, particular features of Twitter are extracted and applied to the function vector. Afterwards, these characteristics are stripped from tweets, and extraction of features is performed again as though it were done on the regular text (Samad and Gani, 2020). These features are introduced by the function vector as well. Using different classifiers, such as Nave Bayes, SVM, Maximum Entropy, and Ensemble classifiers, the precision of the feature vector classification is checked. For the latest feature vector, all these classifiers have almost comparable precision.

Table 4. Tweet sentiment analysis using uni-gram approach

Algorithm	Accuracy	Feature extraction	Dataset	Number of sentiments
SVM	90	Uni-gram	Tweet	2
Naïve Bayes	89.8	Uni-gram	Tweet	2
Maximum Entropy	90	Uni-gram	Tweet	2
Ensemble	90	Uni-gram	Tweet	2

Hasan et al. in (Hasan et al., 2018) By comparing sentiment lexicons (W-WSD, SentiWordNet, TextBlob), they developed a new way of classifying sentiment tweets, which can better be embraced by sentiment analysis. With two machine-learning algorithms, Naïve Bayes and SVM, they validated three of the sentiment analysis lexicons. With W-WSD, Naïve Bayes showed the highest precision of 79 percent, while SVM showed 70 percent precision.

(Singh et al., 2017) used four machine learning algorithms Naïve Bayes, OneR, BFTree, and J-48, to optimize sentiment analysis. Three datasets were used, two from IMBD and one from Amazon. The efficacy of these four sentiment classification models is tested and compared. The Naïve Bayes find learning to be very easy, while OneR appears to be more promising in producing 91.3 percent accuracy, 97 percent F-measure accuracy.

Table 5

Algorithm	Accuracy	Feature extraction	Dataset	Number of sentiments
SVM	70	Uni-gram, Sentence level	Tweet	3
Naïve Bayes	79	Uni-gram, Sentence level	Tweet	3

Table 6. Tweet sentiment analysis with two sentiments

Algorithm	Accuracy	Feature extraction	Dataset	Number of sentiments
Naïve Bayes	85.24	Uni-gram	Tweet	2
J-48	89.73	Uni-gram	Tweet	2
BFTree	90.07	Uni-gram	Tweet	2
OneR	92.34	Uni-gram	Tweet	2

2.2. Deep learning

After examining the above papers extensively, we observed that most researchers used a limited number of datasets to train and test their model. Also, they did not consider the neutral data points as well. The machine learning technique used is supervised in all the papers none of them tried un-supervised learning techniques like the Knn algorithm. Traditional machine learning algorithms did not achieve a satisfactory level in the case of accuracy. The feature extraction technique used is also the same in all the papers; they prefer to use the uni-gram approach, approaches like BOW, Word2vec, OneHotshot encoding, and TF-IDF considered. Now we will move towards deep learning methods.

Ramadhani, A. M., & Goo, H. S in (Ramadhani and Goo, 2017) used Korean and English language for text processing or sentiment analysis using deep learning model. The specification of the network is:

- Feedforward Neural Network
- Using Mean Square Unit and the Stochastic gradient descent
- 3 Hidden Layer
- The input is 100 neurons
- Using ReLU and the sigmoid function activation

This experiment used 1,000 dataset of each negative and positive; the total data points are 4,000. The 100-epoch experiment train uses a learning rate of 0.1 and 0.001. To build the network, the experiment used the Tensorflow. The model showed 77.45 percent accuracy on train data and 75.03 percent accuracy on test data. (Severyn and Moschitti, 2015) This article explains our deep learning framework for tweet analysis of sentiment. This research's crucial contribution is a new paradigm for initializing the coevolutionary neural network's parameter weights, which is essential for training model while avoiding the need to introduce any additional features. In short, we use an unsupervised neural language processing model to train initial word embedding that is further optimized on a small supervised corpus by deep learning model. The network's pre-trained parameters are used at the final stage to initialize the model. The unsupervised model's output is fed as input to the supervised training data newly made available by the official Twitter Sentiment Analysis system assessment campaign organized by Semeval-2015. The network comprises a single convolutional layer followed by a non-linearity, max pooling, and a soft-max classification layer. We divided the deep learning models into CNN's, Word Embedding, LSTM (Long Short-term memory), Recurrent Neural network, and DBN's.

2.2.1. CNN's

The work done by (Kim, 2014) is the most prevalent CNN model for the classification of sentence-level sentiment. The author conducted an experiment with CNN built on top of pre-trained word2vec. The experimental results show that deep learning can be used by pre-trained vectors as an excellent feature extractor for NLP tasks. Inspired by these observations, Zhang and Wallace proposed a one-layer CNNN architecture for sentence classification (Zhang and Wallace, 2015).

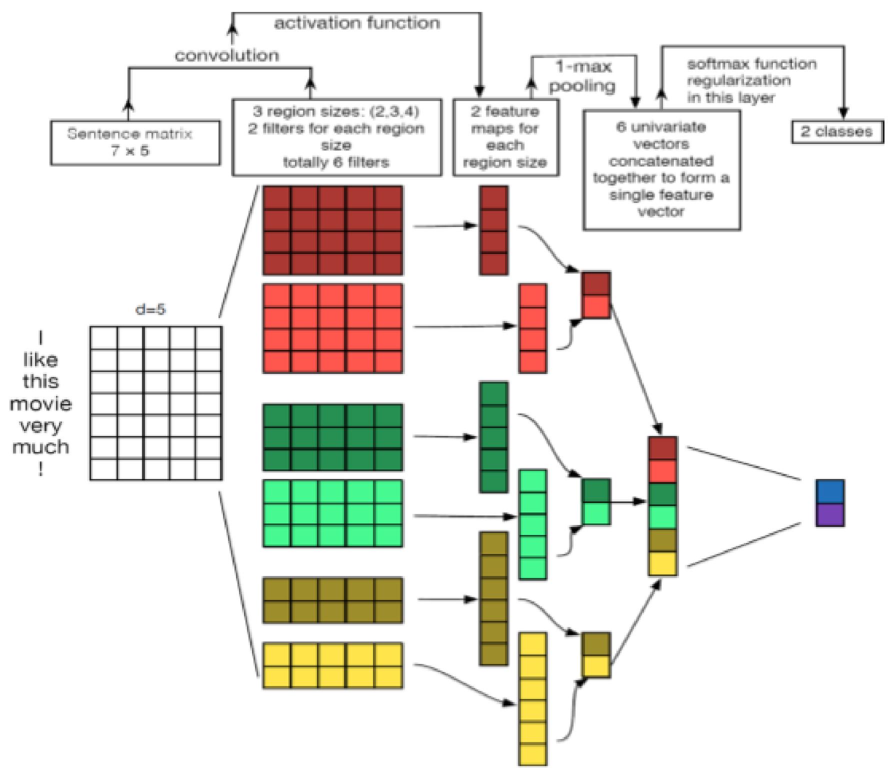

Figure 1. *CNN architecture for sentiment classification* (Zhang and Wallace, 2015)

2.2.2. Word embedding

One of the popular techniques for learning word embedding is Word2Vec (Joulin et al., 2016). They use an existing neural network before moving it into a deep learning algorithm to process a text. Embedding can be done using the Skip Gram model and the Bagofwords Typical model (CBOW). GlobalVectors similarly produce the vector encoding of a word (GloVe) (Faruqui et al., 2016). The advantage of the Glove model is that, as the implementation can be parallelized, it can be easily trained on more data. But char2vec (Sun et al., 2019) learns embedding related to of character of a word from the other side, instead of learning the full word's embedding. (Xu et al., 2018) Suggested a model to learn sentiment embedments using sentiment intensity scores from sentiment lexicons.

2.2.3. Recurrent Neural Networks

The time factor for handling the components in a series is considered by RNN. RNN efficiency relies not only on the current input, but also on the output calculated from the previously hidden state of a network.

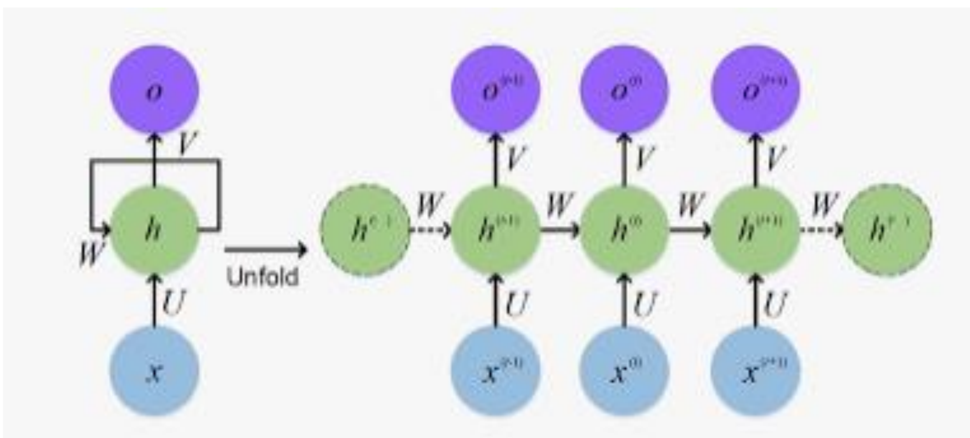

Figure 2. *Recurrent Neural Network*

2.2.4. LSTM

In regular RNN, long short-term memory can manage the vanishing gradient and can capture long-term dependencies.

2.2.5. DBN's

Jin (Naja and Mohamed, 2017) Implemented DBNs with delta rule for sentiment classification on ten sentiment datasets. For fine-tuning the weights, the Delta rule uses gradient descent in a single layer neural network. To distinguish sentences, Ruangkanokmas et al. used DBNs with feature selection (DBNFS) in (Ruangkanokmas et al., 2016). Emotional analysis using attention-base network analysis is used in Yuan et al. (Chen et al., 2018), Zhang et al. (Li et al., 2019), Jiang et al. (Hailong et al., 2014), and Song et al. (Yoo et al., 2018). Also, capsule networks are becoming popular for various text classification tasks in natural language processing (Ke et al., 2019; Yang et al., 2019; Kim and Jeong, 2019).

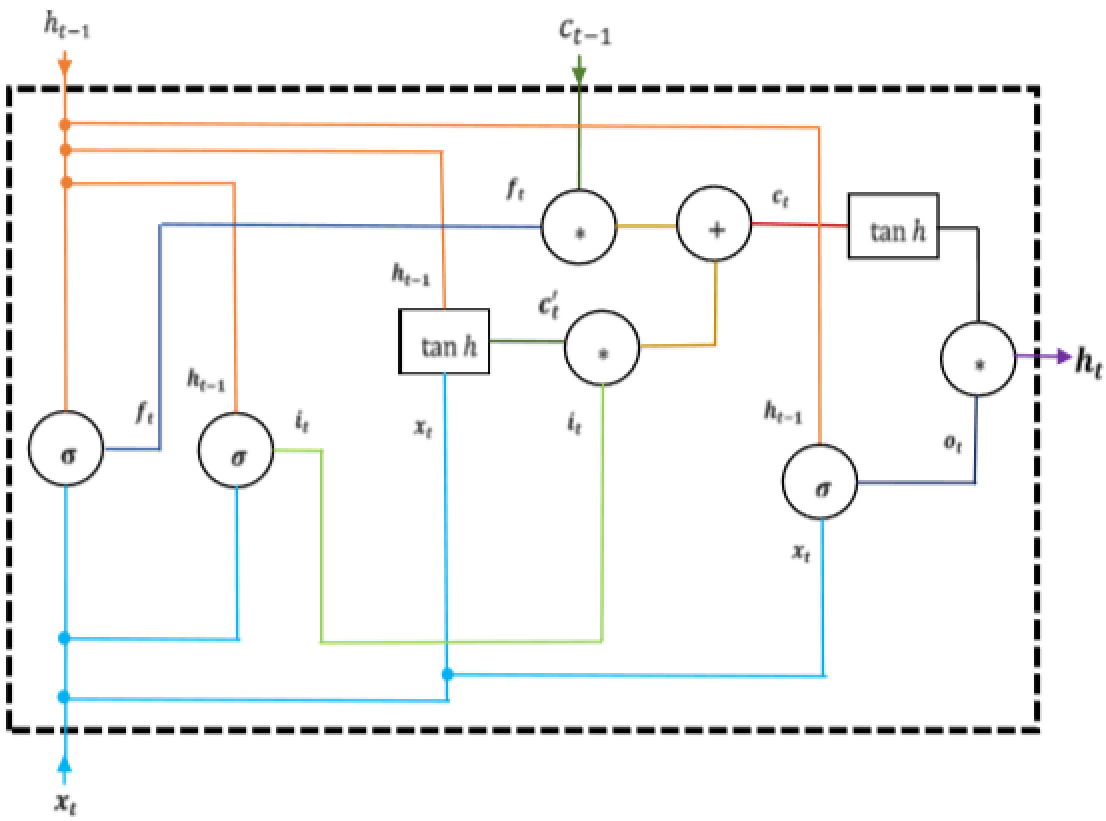

Figure 3. *Architecture of LSTM* (Yadav and Vishwakarma, 2020)

While deep learning techniques show promising results in sentimental research, there are some drawbacks; in order to ensure better results, deep learning networks need a significant number of labeled data for training. It is difficult to determine the real reason for the neural network to forecast a specific sentiment in a body of the text by pointing at weights in multiple elements, where we know what features are selected to forecast a particular feeling, unlike conventional machine learning or lexical approaches. This makes it hard for several researchers to comprehend the mechanism for predicting neural networks and to function as a “black box”. Choosing optimum conditions is also a tricky job. Deep learning methods have been resource-intensive due to a large number of parameters.

3. Conclusion

We reviewed deep learning and traditional machine learning models’ papers for sentiment analysis; most machine learning models either used Naïve Bayes or SVM algorithms to classify the sentiments and number of datasets to train and test the model limited most of the researchers used only one dataset. Traditional machine learning models showed good accuracy but the feature extraction technique used was also traditional, due to which they did not achieve satisfactory results. The deep learning approach, on the other, proves to be more effective due to advanced feature extraction techniques like Word2Vec, GloVe, etc. However, deep learning is resource-intensive. They require GPU and CPU for useful and timely training and testing of data. There is a tradeoff between the traditional machine learning approach and the deep learning approach in speed and accuracy. Deep learning methods showed good accuracy but resource-intensive, while traditional methods showed little less accurate than conventional, but they are not resourced intensive. Researchers can choose any of the two approaches based on their needs and resource availability.

4. Acknowledgements

We are grateful to University of Miskolc, particularly informatics department for providing us such an opportunity to work with them.

References

[1] Wawre, S. V. & Deshmukh, S. N. (2016). Sentiment classification using machine learning techniques. *International Journal of Science and Research (IJSR)*, 5 (4), 819–821. **https://doi.org/10.21275/v5i4.NOV162724**

[2] Gautam, G. & Yadav, D.: *Sentiment analysis of Twitter data using machine learning approaches and semantic analysis.* 2014 Seventh International Conference on Contemporary Computing (IC3), IEEE, pp. 437–442. **https://doi.org/10.1109/IC3.2014.6897213**

[3] Le, B. & Nguyen, H. (2015). Twitter sentiment analysis using machine learning techniques. *Advanced Computational Methods for Knowledge Engineering*, pp. 279–289. Springer, Cham. **https://doi.org/10.1007/978-3-319-17996-4_25**

[4] Neethu, M. S. & Rajasree, R.: *Sentiment analysis in Twitter using machine learning techniques*. 2013 Fourth International Conference on Computing, Communications and Networking Technologies (ICCCNT), IEEE, pp. 1–5. **https://doi.org/10.1109/ICCCNT.2013.6726818**

[5] Hasan, A., Moin, S., Karim, A. & Shamshirband, S. (2018). Machine learning-based sentiment analysis for Twitter accounts. *Mathematical and Computational Applications*, 23 (1), 11. **https://doi.org/10.3390/mca23010011**

[6] Singh, J., Singh, G. & Singh, R. (2017). Optimization of sentiment analysis using machine learning classifiers. *Human-centric Computing and Information Sciences*, 7 (1), 32. **https://doi.org/10.1186/s13673-017-0116-3**

[7] Ramadhani, A. M. & Goo, H. S.: *Twitter sentiment analysis using deep learning methods*. 2017 7th International annual engineering seminar (InAES), IEEE, pp. 1–4. **https://doi.org/10.1109/INAES.2017.8068556**

[8] Severyn, A. & Moschitti, A. (2015). Twitter sentiment analysis with deep convolutional neural networks. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval* (pp. 959–962). **https://doi.org/10.1145/2766462.2767830**

[9] Kim, Y. (2014). Convolutional neural networks for sentence classification. *arXiv preprint arXiv: 1408.5882*. **https://doi.org/10.3115/v1/D14-1181**

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

[11] Joulin, A., Grave, E., Bojanowski, P. & Mikolov, T. (2016). Bag of tricks for efficient text classification. *arXiv preprint arXiv:1607.01759*. **https://doi.org/10.18653/v1/E17-2068**

[12] Faruqui, M., Tsvetkov, Y., Rastogi, P., & Dyer, C. (2016). Problems with an evaluation of word embeddings using word similarity tasks. *arXiv preprint arXiv:1605.02276*. **https://doi.org/10.18653/v1/W16-2506**

[13] Sun, C., Qiu, X. & Huang, X. (2019). Vcwe: Visual character-enhanced word embeddings. *arXiv preprint arXiv:1902.08795*.

[14] Xu, H., Liu, B., Shu, L. & Yu, P. S. (2018). Double embeddings and CNN-based sequence labeling for aspect extraction. *arXiv preprint arXiv:1805.04601*. **https://doi.org/10.18653/v1/P18-2094**

[15] Yadav, A. & Vishwakarma, D. K. (2020). Sentiment analysis using deep learning architectures: a review. *Artificial Intelligence Review*, 53 (6), 4335–4385. **https://doi.org/10.1007/s10462-019-09794-5**

[16] Huang, G., Liu, Z., Van Der Maaten, L. & Weinberger, K. Q. (2017). Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 4700–4708). **https://doi.org/10.1109/CVPR.2017.243**

[17] Naja, M. M. F. & Mohamed, M. I. I. (2017). Analysis of Systematic Data Mining Approaches for Achieving Competitive Advantage by Monitoring Social Media.

[18] Ruangkanokmas, P., Achalakul, T. & Akkarajitsakul, K.: Deep belief networks with feature selection for sentiment classification. 2016 7th International Conference on Intelligent Systems, Modelling and Simulation (ISMS), *IEEE*, pp. 9–14. **https://doi.org/10.1109/ISMS.2016.9**

[19] Chen, Y., Yuan, J., Yu, Q. & Luo, J. (2018). Twitter sentiment analysis via bi-sense emoji embedding and attention-based LSTM. In *Proceedings of the 26th ACM international conference on Multimedia* (pp. 117–125). **https://doi.org/10.1145/3240508.3240533**

[20] Li, X., Bing, L., Zhang, W. & Lam, W. (2019). Exploiting BERT for end-to-end aspect-based sentiment analysis. *arXiv preprint arXiv:1910.00883*. **https://doi.org/10.18653/v1/D19-5505**

[21] Hailong, Z., Wenyan, G. & Bo, J.: Machine learning and lexicon-based methods for sentiment classification: A survey. 2014 11th web information system and application conference, *IEEE*, pp. 262–265.

[22] Yoo, S., Song, J. & Jeong, O. (2018). Social media contents based sentiment analysis and prediction system. *Expert Systems with Applications*, 105, pp. 102–111. **https://doi.org/10.1016/j.eswa.2018.03.055**

[23] Ke, P., Ji, H., Liu, S., Zhu, X. & Huang, M. (2019). Sentilr: Linguistic knowledge enhanced language representation for sentiment analysis. *arXiv preprint arXiv:1911.02493*.

[24] Yang, M., Jiang, Q., Shen, Y., Wu, Q., Zhao, Z. & Zhou, W. (2019). Hierarchical human-like strategy for aspect-level sentiment classification with sentiment linguistic knowledge and reinforcement learning. *Neural Networks*, 117, pp. 240–248. **https://doi.org/10.1016/j.neunet.2019.05.021**

[25] Kim, H. & Jeong, Y. S. (2019). Sentiment classification using convolutional neural networks. *Applied Sciences*, 9 (11), 2347. **https://doi.org/10.3390/app9112347**

[26] Feldman, R. (2013). Techniques and applications for sentiment analysis. *Communications of the ACM*, 56 (4), pp. 82–89. **https://doi.org/10.1145/2436256.2436274**

[27] Vohra, M. S., Teraiya, J. B. (2016) A Comparative Study of Sentiment Analysis Techniques. *Journal of Information, Knowledge, and Research in Computer Engineering*, 2 (2), pp. 313–317.

[28] Medhat, W., Hassan, A. & Korashy, H. (2014). Sentiment analysis algorithms and applications: A survey. *Ain Shams Engineering Journal*, 5 (4), pp. 1093–1113. **https://doi.org/10.1016/j.asej.2014.04.011**

[29] Sebastiani, F. (2002). Machine learning in automated text categorization. *ACM computing surveys (CSUR)*, 34 (1), pp. 1–47. **https://doi.org/10.1145/505282.505283**

[30] Taboada, M., Brooke, J., Tofiloski, M., Voll, K. & Stede, M. (2011). Lexicon-based methods for sentiment analysis. *Computational linguistics*, 37 (2), pp. 267–307. **https://doi.org/10.1162/COLI_a_00049**

[31] Prabowo, R. & Thelwall, M. (2009). Sentiment analysis: A combined approach. *Journal of Informetrics*, 3 (2), pp. 143–157. **https://doi.org/10.1016/j.joi.2009.01.003**

[32] Dang, Y., Zhang, Y. & Chen, H. (2009). A lexicon-enhanced method for sentiment classification: An experiment on online product reviews. *IEEE Intelligent Systems*, 25 (4), pp. 46–53. **https://doi.org/10.1109/MIS.2009.105**

[33] Shen, Y., He, X., Gao, J., Deng, L. & Mesnil, G. (2014 a). A latent semantic model with the convolutional-pooling structure for information retrieval. In *Proceedings of the 23rd ACM international conference on conference on information and knowledge management* (pp. 101–110). **https://doi.org/10.1145/2661829.2661935**

[34] Shen, Y., He, X., Gao, J., Deng, L. & Mesnil, G. (2014 b). Learning semantic representations using convolutional neural networks for web search. In *Proceedings of the 23rd international conference on world wide web* (pp. 373–374). **https://doi.org/10.1145/2567948.2577348**

[35] Goodfellow, I., Bengio, Y., Courville, A. & Bengio, Y. (2016). *Deep learning* (Vol. 1, No. 2). Cambridge: MIT Press.

[36] Samad, D., Gani, G. A. (2020). Analyzing and predicting spear-phishing using machine learning methods. *Multidiszciplináris Tudományok*, 10 (4), pp. 262–273. **https://doi.org/10.35925/j.multi.2020.4.30**