

# Deep Sentiment Analysis using a Graph-based Text Representation

Kayvan Bijari<sup>a</sup>, Hadi Zare<sup>a,\*</sup>, Hadi Veisi<sup>a</sup>, Emad Kebriaei<sup>a</sup>

<sup>a</sup>*Faculty of New Sciences and Technologies, University of Tehran, North Kargar Street, Tehran, Iran*

---

## Abstract

Social media brings about new ways of communication among people and is influencing trading strategies in the market. The popularity of social networks produces a large collection of unstructured data such as text and image in a variety of disciplines like business and health. The main element of social media arises as text which provokes a set of challenges for traditional information retrieval and natural language processing tools. Informal language, spelling errors, abbreviations, and special characters are typical in social media posts. These features lead to a prohibitively large vocabulary size for text mining methods. Another problem with traditional social text mining techniques is that they fail to take semantic relations into account, which is essential in a domain of applications such as event detection, opinion mining, and news recommendation. This paper set out to employ a network-based viewpoint on text documents and investigate the usefulness of graph representation to exploit word relations and semantics of the textual data. Moreover, the proposed approach makes use of a random walker to extract deep features of a graph to facilitate the task of document classification. The experimental results indicate that the proposed approach defeats the earlier sentiment analysis methods based on several benchmark datasets, and it generalizes well on different datasets without dependency for pre-trained word embeddings.

---

\*Corresponding author

*Email addresses:* `kayvan.bijari@ut.ac.ir` (Kayvan Bijari), `h.zare@ut.ac.ir` (Hadi Zare), `h.veisi@ut.ac.ir` (Hadi Veisi), `emad.kebriaei@ut.ac.ir` (Emad Kebriaei)

*Keywords:* Sentiment Analysis, Graph Representation, Representation Learning, Feature Learning, Conventional Neural Networks

---

## 1. Introduction

Text messages are very ubiquitous and they are transferred every day throughout social media, blogs, wikis, news headlines, and other online collaborative media. Accordingly, a prime step in text mining applications is to extract interesting patterns and features, from this supply of unstructured data. Feature extraction can be considered as the core of social media mining tasks such as sentiment analysis, event detection, and news recommendation [2].

In the literature, sentiment analysis tends to be used to refer to the task of classifying the polarity of a given piece of text at the document, sentence, feature, or aspect level [23]. There are various applications on a variety of domains which utilize sentiment analysis, in this regard one can mention applying the sentiment analysis for political reviews to estimate the general viewpoint of the parties [43], predicting stock market prices based on sentiment analysis by utilizing the different financial news data [5], and making use of the sentiment analysis to recognize the current medical and psychological status for a community [23].

Machine learning algorithms and statistical learning techniques have been rising in a variety of scientific fields [9, 10]. A number of machine learning techniques have been proposed to perform the task of sentiment analysis. As one of the powerful sub-domains of machine learning in recent years, deep learning models are emerging as a persuasive computational tool, they have affected many research areas and can be traced in many applications. With respect to the deep learning, textual deep representation models attempt to discover and present intricate syntactic and semantic representations of texts, automatically from data without any hand-made feature engineering. Deep learning methods coupled with deep feature representation techniques have improved the state-of-the-art models in various machine learning tasks such as sentiment analysis [25,

30] and text summarization [45].

Inspired by the recent advances in feature learning and deep learning methods, it is determined that inherent features can be learned from the raw structure of data using learning algorithms. This technique is called representation learning which aids to promote and advance functionality of machine learning methods. to put it differently, representation learning is able to map or convert raw data into a set of features which are considerably more distinctive for machine learning algorithms.

The purpose of this research is to propose a novel approach which takes advantages of graph-based representation of documents integrated with representation learning through the Convolutional Neural Networks (CNN) [34]. Graph representation of documents reveals intrinsic and deep features compared to the traditional feature representation methods like bag-of-words (BOW) [24]. Unlike conventional methods, a graph can contain every aspect of a given text, as an example in the bag-of-words representation model the general suggestion is to remove stop words from the texts, but they can convey some meaning which may be valuable for sentiment analysis. For the purpose of graph representation, each individual word is depicted as a node in the graph and the interactions between different nodes are modeled via undirected, directed, and weighted edges.

The overall structure of this paper is as follows. Section 2 begins by reviewing the related works of sentiment analysis and presents the basic idea behind the proposed approach. Section 3 discusses the methodology of the proposed method and demonstrates how graph representation and feature learning are used to perform sentiment analysis. A brief introduction of the standard datasets and experimental results of the proposed approach versus some well-known algorithms is given in Section 4. Eventually, Section 5 ends the paper with a conclusion and some insights for future works.

## 2. Related Works and Basic Idea

### 2.1. Related Works

Over the last few years, broad research on sentiment analysis through supervised [28], semi-supervised [19], and unsupervised [11] machine learning techniques have been done. Go et al. [13] were among the firsts who applied distant supervision technique to train a machine learning algorithms based on emoticons for sentiment classification. Lately, a number of works have been published in this field, researchers in the field of natural language processing carried out a variety of new algorithms to perform sentiment analysis [40]. Subsequently, some of the distinguished works are further discussed in this section.

As a sub-domain of information retrieval and natural language processing, sentiment analysis or opinion mining can be viewed from different levels of granularity namely, sentence level, document level, and aspect level; from the point of view of sentence level, Liu’s works can be mentioned as one of the pioneers in this field [18]. Works by Pan and Lee can also be considered in which document level of sentiment analysis is examined [29]. Lately aspect level of sentiment analysis has attracted more attention, research by Agarwal can be listed in this regard [1].

Graph-based representation techniques for sentiment analysis have been used in a variety of research works. [26] considered text corpus as a labeled directed graph in which words represent nodes, and edges denote syntactic relation between the words. They proposed a new path-constrained graph walk method in which the graph walk process is guided by high-level knowledge about essential edge sequences. They showed that the graph walk algorithm results in better performance and is more scalable. In the same way, [44] suggested the word-graph sentiment analysis approach. In the model, they proposed a well-defined graph structure along with several graph similarity methods, afterward, the model extracts feature vectors to be used in the task of polarity classification. Furthermore, [14] proposed a graph-based semi-supervised algorithm to perform sentiment classification through solving an optimization problem, their

model suits situations in which data labels are sparse.

Deep learning methods are also operating properly on the field of sentiment analysis, [37] proposed a semi-supervised approach based on recursive autoencoders to foresee sentiment of a given sentence. The system learns vector representation for phrases and exploits the recursive nature of sentences. They have also proposed a matrix-vector recursive neural network model for semantic compositionality. It is able to learn compositional vector representations for expressions and sentences with discretionary length [36]. To clarify, the vector model catches the intrinsic significance of the component parts of sentences, while the matrix takes the importance of neighboring words and expressions into account. In another effort [38] proposed recursive neural tensor network (RNTN) which represents a phrase through word vectors and a parse tree. The model computes vectors of nodes in the tree-based composition function.

More importantly, other deep architectures have been used for natural language processing tasks [6]. [8] used a convolutional neural network for the semantic role labeling task in order to avoid excessive task-specific feature engineering. In another attempt [7], a convolutional network with similar architecture is used for syntactic parsing. Also, [33] used a convolutional neural network to extract document features and then employed multiple-kernel learning (MKL) for sentiment analysis. In another work [32] long short-term memory is applied to extracts contextual information from the surrounding sentences.

Unlike deep learning methods, which use neural networks to transform feature space into high dimensional vectors, general practices for sentiment analysis take advantage of basic machine learning methods. Indeed, [41] ensembles a collection of machine learning techniques along with n-grams to predict sentiment of a document. Additionally, evolutionary algorithms have been utilized for several optimization problems [4], ALGA [21] makes use of evolutionary computation to determine optimal sentiment lexicons which leads to a better performance.

## 2.2. Motivation

In the field of natural language processing, bag-of-word representation is one of the very well-known means to represent the features of a document. However, it is insufficient to describe the features of a given document due to a number of limitations such as lacking word relations, scalability issues, and neglecting semantics [12]. In order to mitigate these shortcomings, some other representation techniques are proposed to model textual documents [42]. These methods are able to take into account a variety of linguistic, semantic, and grammatical features of a document.

The decency of solutions which a machine learning algorithms produce for a task such as classification, deeply relies upon the way features are represented in the solution area. Various feature representations methods are able to entangle or neglect some distinctive features behind the data. This is where feature selection and feature engineering methods come into play and try to facilitate and augment the functionality of machine learning algorithms [46].

Feature engineering methods accompanying domain-specific expertise can be used to modify basic representations and extract explanatory features for the machine learning algorithms. On the other hand, new challenges in data presentation, advancements in artificial intelligence, and probabilistic models drive the need for representation learning techniques and feature learning methods. Feature learning can be defined as a transformation of raw data input to a new representation that can be adequately exploited in different learning algorithms [3].

As was pointed out in the introduction to this paper, the main idea of the proposed method is to render sentences in a document as a graph and then analyze the graphs using network representation learning approaches. In this regard, the proposed method entails three main phases namely, graph representation, feature learning, and classification. A more detailed account of the components of the proposed method are further addressed in Section 3.

### 3. Elements of the proposed method

The proposed method consists of three principal building blocks which will be explained further in parts. initially, textual documents are quite pre-processed and transformed into word-graphs. Secondary, by making use of feature learning methods, inherent and intrinsic characteristics of textual graphs will be determined in the representation learning phase. Eventually, a convolutional neural network is trained based on the extracted features and performs sentiment classification task. Figure 1 explains the work-flow of the proposed method.

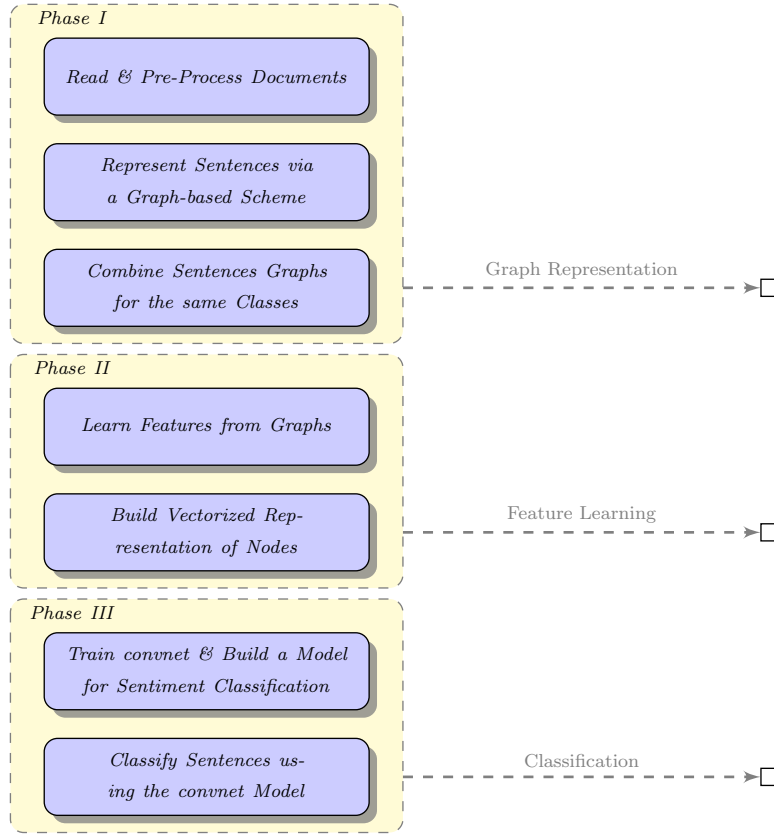


Figure 1: work-flow of the proposed sentiment classification approach

### 3.1. Graph Representation

In the era of big data, text is one of the most ubiquitous forms of storing data and metadata. Data representation is the vital step for the feature extraction phase in data mining. Hence, a proper text representation model which can considerably picture inherent characteristics of textual data, is still an ongoing challenge. Due to simplicity and shortcomings of traditional models such as the vector space model, offering new models is highly valued. Some disadvantages of classical models such as bag-of-words model can be listed as follows [12]:

- Meaning of words in the text and textual structure cannot be accurately represented.
- Words in the text are considered independent from each other.
- Word's sequences, co-occurring, and other relations in a corpus is neglected.

In general terms, words are organized into clauses, sentences, and paragraphs to define the meaning for a document. Furthermore, their occurring, ordering, positioning, and the relationship between different components of the document are important and valuable to understand the document in details.

Graph-based text representation can be acknowledged as one of the genuine solutions for the aforementioned deficiencies. A text document can be represented as a graph in many ways. In a graph, nodes denote features and edges outline the relationship among different nodes. Although there exist various graph-based document representation models [44], the co-occurrence graph of words is an effective way to represent the relationship of one term over the other in the social media contents such as Twitter or short text messages. The co-occurrence graph is called word-graph in the rest of the paper.

Word-graph is defined as follows: given a sentence  $S$ , let  $W$  be the set of all words in the sentence  $S$ . A Graph  $G(E, W)$  is constructed such that any  $w_i, w_j \in W$  are connected by  $e_k \in E$ , *if*  $\exists R_l$  *s.t.*  $R_l(w_i, w_j) \in R$ .



In other words, in the graph  $G$  any word in the sentence is treated as a single vertex. Any two vertices are connected via the edge  $e_k$ , if there exists a connection between them governed by the relation  $R$ . The relation  $R$  is satisfied if its corresponding lexical units co-occur within a window of maximum  $N$  words, where  $N$  can be set to any value (typically between two and ten words seems to be fine in based on different trade-offs). Figure 2 presents the graph of a sample sentence with word-window with size 3. Relation  $R$  in this graph is satisfied when two nodes are within a window with the maximum length of 3.

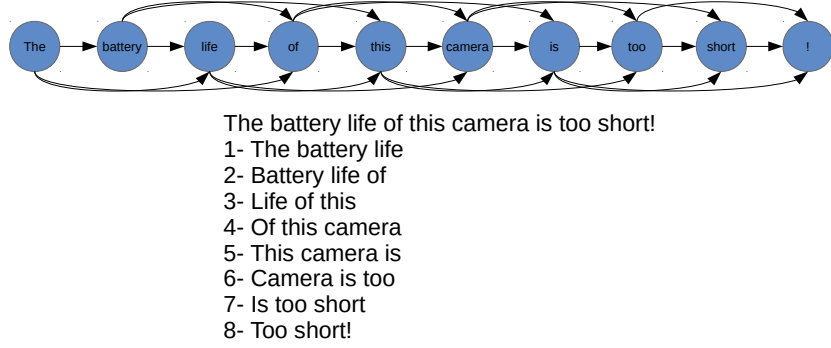


Figure 2: A sample sentence graph with word-window 3, and sub-sentences which each window give importance to.

### 3.2. Feature Learning

In order to perform well on a given learning task, any (un)supervised machine learning algorithm requires a set of informative, distinguishing, and independent features. One typical solution in this regard is to feed the algorithms with hand-engineered domain-specific features based on human ingenuity and expert knowledge. However, feature engineering designates algorithm's lack of efficiency to entangle and organize the discriminative features from the data. Moreover, feature engineering not only requires tedious efforts and labor, but it is also designed for specific tasks and can not be efficiently generalized across other tasks [16]. Accordingly, in order to broaden the scope and applicability of machine learning algorithms for different jobs, it would be much beneficial to

make machine learning algorithms less dependent on feature engineering techniques.

An alternative to feature engineering is to enable algorithms to learn features of their given task based on learning techniques. As one of the new tools in machine learning, representation learning and feature learning enables machines and algorithms to learn features on their own directly from data. In this regard, features are extracted by exploiting learning techniques and making transformation on raw data for the given task. Feature learning allows a machine to learn specific tasks as well as its features and obviates the use of feature engineering [3].

Node embedding is vectorized representation of nodes of each graph, and it is trained via feature learning algorithms to pay more attention to the important nodes and relations while paying less to the unimportant ones. To be more specific, in the proposed method as a novel feature learning algorithm, node2vec, is used to reveal innate and essential information of a given text graph [16], then a conventional neural network is used to learn and classify text graphs.

Node2vec [16] together with Deepwalk [31] are two well-known algorithms for representation learning on the graph structure. The main goal of such algorithms is to pay more attention to the important nodes and relations while paying less to the unimportant ones. In other words, a feature learning algorithm is used to reveal the innate and essential information of a given graph.

Node2vec is a semi-supervised algorithm which lately presented for scalable feature learning in graph networks. The purpose of the algorithm is to optimize a graph-based objective function using stochastic gradient descent. Using a random walk manner in finding a flexible notation of neighborhoods, the node2vec algorithm returns feature representation, node embeddings, that maximize the likelihood of preserving network neighborhoods of nodes [16]. In the proposed method representation learning was done based on the node2vec framework. The work-flow of the feature learning in the proposed algorithm is further discussed in the following.

Feature learning in networks is formulated as a maximum likelihood opti-

mization problem. let  $G = (V, E)$  be a given (un)directed word-graph. let  $f : v \rightarrow R^d$  be the mapping function from nodes to feature representation which is to be learned for a distinguished task.  $d$  is a parameter which designates the number of dimensions of the feature to be represented. Equivalently,  $f$  is a matrix of size  $|v| \times d$  parameters. for every node in the graph  $u \in V$ , a neighborhood  $N_{S(u)} \subset V$  is defined.

The following optimization function which attempts to maximize the log-probability of observing neighborhood  $N_S(u)$  for node  $u$ , is defined as equation (1).

$$\max_f \sum_{u \in V} \log(P(N_S(u)|f(u))) \quad (1)$$

To make sure that the equation (1) is tractable, two standard assumptions need to be made.

- Conditional independence. Likelihood is factorized in such a way that the likelihood of observing a neighborhood node is independent of observing any other neighborhood. According to this assumption,  $P(N_S(u)|f(u))$  can be rewritten as equation (2).

$$P(N_S(u)|f(u)) = \prod_{n_i \in N_S(u)} P(n_i|f(u)) \quad (2)$$

- Symmetry in feature space. A source node and neighborhood node have a symmetric impact on each other. Based upon this assumption  $P(n_i|f(u))$  is calculated using equation (3) in which conditional likelihood of every source-neighborhood node can be parametrized by the dot product of their features.

$$P(n_i|f(u)) = \frac{\exp(f(n_i) \cdot f(u))}{\sum_{u \in V} \exp(f(v) \cdot f(u))} \quad (3)$$

Based on these two assumptions, the objective function in equation (1) can be simplified,

$$\max_f \sum_{u \in V} \left[ -\log(Z_u) + \sum_{n_i \in N_S(u)} f(n_i) \cdot f(u) \right] \quad (4)$$

where, per-node partition function,  $Z_u = \sum_{v \in V} \exp(f(u) \cdot f(v))$ . Equation (4) is then optimized using stochastic gradient ascent over model parameters defining in the features  $f$ .

The neighborhoods  $N_S(u)$  are not restricted to direct neighbors, but it is generated using sampling strategy  $S$ . There are many search strategies to generate neighborhood  $N_S(u)$  for a given node  $u$ , simple strategies include breadth-first sampling which samples immediate neighbors, and depth-first sampling which seeks to sample neighbors with the most distant from the source. For a better exploration of the graph structure, a random walk manner is used as a sampling strategy which smoothly interpolates between BFS and DFS strategies. In this regard, given a source node  $u$ , a random walk of length  $l$  is simulated. Let  $c_i$  be the  $i$ -th node in the walk, starting with  $c_0 = u$ . Other nodes in the walk, are generated using the following equation (5).

$$P(c_i = x | c_{i-1} = v) = \begin{cases} \frac{\pi_{vx}}{Z} & \text{if } (v, x) \text{ in } E, \\ 0 & \text{otherwise.} \end{cases} \quad (5)$$

where  $\pi_{vx}$  is the transition probability between given nodes  $v$  and  $x$ , and  $Z$  is the normalizing constant.

### 3.3. ConvNet based Sentiment Classification

Adopted from neurons of the animal’s visual cortex, ConvNets or conventional neural networks is a biologically inspired variant of a feed-forward neural network [34]. ConvNets have shown to be highly effective in many research areas such as image classification and pattern recognition tasks [35]. They have also been successful in other fields of research such as neuroscience [17] and bioinformatics [20].

Similar to the general architecture of neural networks, ConvNets are comprised of neurons, learning weights, and biases. Each neuron receives several inputs, takes a weighted sum over them, passes it through an activation function at its next layer and responds with an output. The whole network contains a

loss function to direct the network through its optimal goal, All settings that will apply on the basic neural network [15], is likewise applicable to ConvNets.

Apart from computer vision or image classification, ConvNets are applicable for sentiment and document classification. In this regard, inputs for the deep algorithms are sentences or documents which are represented in form of a matrix. Each row of the matrix corresponds to one token or a word. In other words, each row is a vectorized representation of the word, and the whole matrix will represent a sentence or a document. In the deep learning based approaches, these vectors are low-dimensional word embedding resulted from approaches such as word2vec [25] or GloVe [30].

In the proposed method, a slight variant of ConvNet architecture of Kim [22] and Collobert [8] is used for sentiment classification of sentences. Let  $n_i \in \mathbb{R}^d$  the d-dimensional node embedding corresponding to  $i$ -th node in a word-graph of a given sentence. It should be noted that sentences are padded beforehand where it was necessary in order to make sure that they have the same length.

For convolution operation, a *filter*  $w \in \mathbb{R}^{x \times d}$  is applied to nodes to produce a new feature,  $c_i$  in equation (6), form a set of  $x$  nodes.

$$c_i = f(w \cdot n_{i:i+x-1} + b) \quad (6)$$

Where  $b$  is bias term and  $f$  is a non-linear function such as hyperbolic tangent. This filter is applied to any possible nodes in the graph of a sentence  $\{g_1, g_2, g_3, \dots, g_x\}$  to create the *feature map*  $c$  in equation (7).

$$c = [c_1, c_2, c_3, \dots, c_x] \quad (7)$$

Afterwards, a max-over-time pooling operation [8] is performed over the feature map and takes maximum value,  $\hat{c} = \max\{c\}$ , as a feature corresponding to this particular filter. This idea is to capture and keep the most important features for each estimated map. Furthermore, this max-pooling deals with the uncertain length of sentences which were padded previously.

The above description was a procedure in which a feature is extracted from

a single filter. The ConvNet model utilizes multiple features each with varying window-sizes to extract diverse features. Eventually, these features fabricate next to the last layer and are passed into a fully connected softmax layer which yields the likelihood probability over the sentiment labels. Figure 3 reveals the architecture of the proposed method accompanying its different parts.

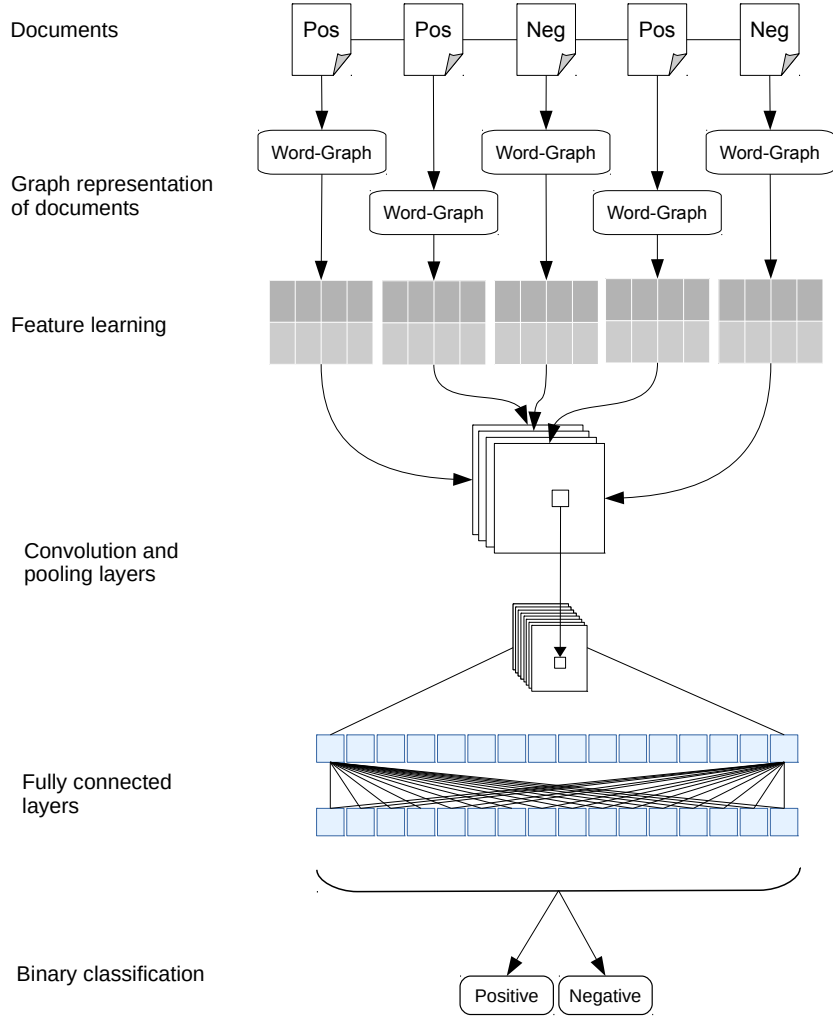


Figure 3: The model architecture of a multi-channel CNN network for sample documents. First, documents are converted into word-graphs. Then, using a feature learning algorithm, node2vec, structure of the graph is transformed into a set of meaningful features. Afterward, via convolution and max-pooling layers, the CNN learns distinguishing features of each document, and eventually, a fully connected softmax layer performs the sentiment classification.

#### 4. Experimental Results

This section is devoted to the experimental results of the proposed method on a set of public benchmark datasets for sentiment classification. In this regard,

first, an introduction to the benchmark datasets and some statistics is given. Then, the performance of the proposed method would be evaluated compared to some well-known machine learning techniques.

#### *4.1. Datasets*

An essential part of examining a sentiment analysis algorithm is to have a comprehensive dataset or corpus to learn from, as well as a test dataset to make sure that the accuracy of your algorithm meets the expected standards. The proposed method was investigated on different datasets which are taken from Twitter and other well-known social networking sites. These datasets are “HCR”, “Stanford”, “Michigan”, “SemEval”, and “IMDB”. These datasets are briefly introduced in the following.

##### *4.1.1. Health-care reform (HCR)*

The tweets of this dataset are collected using the hash-tag “#hcr” in March of 2010 [39]. In this corpus, only the tweets labeled as negative or positive are considered. This dataset consists of 1286 tweets, from which 369 are positive and 917 are negative.

##### *4.1.2. Stanford*

The Stanford Twitter dataset was originally collected by Go et al. [13] this test dataset contains 177 negative and 182 positive tweets.

##### *4.1.3. Michigan*

This data set was collected for a contest in university of Michigan. In this corpus each document is a sentence extracted from social media or blogs, sentences are labeled as positive or negative. The Michigan sentiment analysis corpus contains totally 7086 sentences which 3091 samples are negative and 3995 positive samples.



#### 4.1.4. *SemEval*

The SemEval-2016 corpus [27] was built for Twitter sentiment analysis task in the Semantic Evaluation of Systems challenge (SemEval-2016). 14247 tweets were retrieved for this dataset, of which 4094 tweets are negative and the rest 10153 tweets categorized as positive.

#### 4.1.5. *IMDB*

10,000 positive, 10,000 negative full text movie reviews. Sampled from original Internet movie review database of movies reviews. Table 1 briefly summarizes the datasets which are being used for evaluation of the proposed method.

Table 1: Distribution of negative, positive samples in the given datasets, which will be used for evaluation.

Dataset	HCR	Stanford	Michigan	SemEval	IMDB
Positive	369	182	3995	10153	10,000
Negative	917	177	3091	4094	10,000
Total	1286	359	7086	14247	20,000

#### 4.2. *Results*

Performance of the proposed method is compared to support vector machine (SVM) and conventional neural network (CNN) for short sentences which is using pre-trained Google word embeddings [22]. Table 2 presents the results of the different methods and indicates the superiority of the proposed method over its counterparts.

Table 2: Experimental results on given datasets

Method	Negative class (%)			Positive class (%)			Overall (%)	
	precision	recall	F1	precision	recall	F1	accuracy	F1
<b>HCR</b>								
Proposed	89.11	88.60	81.31	85.17	84.32	84.20	85.71	82.12
SVM(linear)	80.21	91.40	85.01	67.12	45.23	54.24	76.01	76.74
CNN+w2v	75.39	78.69	77.71	40.91	36.49	38.52	66.53	65.94
<b>Stanford</b>								
Proposed	86.38	90.37	91.29	77.46	56.45	65.52	83.71	78.72
SVM(linear)	79.21	100.0	88.40	00.00	00.00	00.00	79.20	70.04
CNN+w2v	79.96	99.59	88.70	22.22	0.56	0.95	79.72	71.10
<b>Michigan</b>								
Proposed	98.89	98.75	98.41	98.82	98.14	98.26	98.41	98.73
SVM(linear)	99.51	91.51	97.50	98.56	98.14	99.62	98.73	98.72
CNN+w2v	95.64	93.43	94.58	95.12	96.73	95.46	95.31	95.34
<b>SemEval</b>								
Proposed	90.80	80.35	84.81	87.32	92.24	90.76	87.69	87.78
SVM(linear)	77.91	61.97	69.06	85.74	92.89	89.17	83.95	83.36
CNN+w2v	57.87	42.26	46.97	78.85	85.13	81.87	72.50	71.98
<b>IMDB</b>								
Proposed	87.42	90.85	88.31	86.25	86.80	86.60	86.07	87.27
SVM(linear)	77.37	76.01	76.69	75.70	77.07	76.38	76.53	76.54
CNN+w2v	81.84	82.35	81.29	82.31	82.32	81.01	79.97	81.11

## 5. Conclusion and Future Work

The main goal of the current study was to determine the usefulness of graph representation learning methods for the task of sentiment analysis. In the proposed approach, graph-based representation was embedded in continuous latent features of documents to incorporate them in a learning scenario. Moreover, deep learning architecture is employed to perform the sentiment classification. The experimental results revealed the superiority of the proposed approach versus its competitors. This ongoing applied field of research has several directions that could be followed for future practice, including, but not limited to, employing other graph-based representation techniques to extract characteristics of a network, exploiting preprocessing methods to enrich the initial features of the system, and use other innate information in a social media to enhance sentiment analysis techniques.

## References

## References

- [1] Agarwal, A., Biadys, F., & Mckeown, K. R. (2009). Contextual phrase-level polarity analysis using lexical affect scoring and syntactic n-grams. In *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics* (pp. 24–32). Association for Computational Linguistics.
- [2] Aggarwal, C. C. (2018). *Machine Learning for Text*. (1st ed.). New York, NY: Springer.
- [3] Bengio, Y., Courville, A., & Vincent, P. (2013). Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, 35, 1798–1828.
- [4] Bijari, K., Zare, H., Veisi, H., & Bobarshad, H. (). Memory-enriched big bang–big crunch optimization algorithm for data clustering. *Neural Computing and Applications*, (pp. 1–11).

- [5] Bollen, J., Mao, H., & Zeng, X. (2011). Twitter mood predicts the stock market. *Journal of computational science*, 2, 1–8.
- [6] Chen, T., Xu, R., He, Y., & Wang, X. (2017). Improving sentiment analysis via sentence type classification using bilstm-crf and cnn. *Expert Systems with Applications*, 72, 221–230.
- [7] Collobert, R. (2011). Deep learning for efficient discriminative parsing. In *AISTATS* (pp. 224–232). volume 15.
- [8] Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K., & Kuksa, P. (2011). Natural language processing (almost) from scratch. *Journal of Machine Learning Research*, 12, 2493–2537.
- [9] Detmer, F. J., Chung, B. J., Mut, F., Pritz, M., Slawski, M., Hamzei-Sichani, F., Kallmes, D., Putman, C., Jimenez, C., & Cebal, J. R. (2018). Development of a statistical model for discrimination of rupture status in posterior communicating artery aneurysms. *Acta Neurochirurgica*, (pp. 1–10).
- [10] Eshtay, M., Faris, H., & Obeid, N. (2018). Improving extreme learning machine by competitive swarm optimization and its application for medical diagnosis problems. *Expert Systems with Applications*, 104, 134–152.
- [11] García-Pablos, A., Cuadros, M., & Rigau, G. (2018). W2vlda: almost unsupervised system for aspect based sentiment analysis. *Expert Systems with Applications*, 91, 127–137.
- [12] Gharavi, E., Bijari, K., Zahirnia, K., & Veisi, H. (2016). A deep learning approach to persian plagiarism detection. In *FIRE (Working Notes)* (pp. 154–159).
- [13] Go, A., Bhayani, R., & Huang, L. (2009). Twitter sentiment classification using distant supervision. *CS224N Project Report, Stanford*, 1.

- [14] Goldberg, A. B., & Zhu, X. (2006). Seeing stars when there aren’t many stars: graph-based semi-supervised learning for sentiment categorization. In *Proceedings of the First Workshop on Graph Based Methods for Natural Language Processing* (pp. 45–52). Association for Computational Linguistics.
- [15] Goodfellow, I., Bengio, Y., Courville, A., & Bengio, Y. (2016). *Deep learning* volume 1. MIT press Cambridge.
- [16] Grover, A., & Leskovec, J. (2016). node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 855–864). ACM.
- [17] Güçlü, U., & van Gerven, M. A. (2015). Deep neural networks reveal a gradient in the complexity of neural representations across the ventral stream. *Journal of Neuroscience*, 35, 10005–10014.
- [18] Hu, M., & Liu, B. (2004). Mining and summarizing customer reviews. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 168–177). ACM.
- [19] Hussain, A., & Cambria, E. (2018). Semi-supervised learning for big social data analysis. *Neurocomputing*, 275, 1662–1673.
- [20] Ji, S., Xu, W., Yang, M., & Yu, K. (2013). 3d convolutional neural networks for human action recognition. *IEEE transactions on pattern analysis and machine intelligence*, 35, 221–231.
- [21] Keshavarz, H., & Abadeh, M. S. (2017). Alga: Adaptive lexicon learning using genetic algorithm for sentiment analysis of microblogs. *Knowledge-Based Systems*, 122, 1–16.
- [22] Kim, Y. (2014). Convolutional neural networks for sentence classification. (pp. 1746–1751). Doha, Qatar: Association for Computational Linguistics.

- [23] Liu, B. (2012). Sentiment analysis and opinion mining. *Synthesis lectures on human language technologies*, 5, 1–167.
- [24] Manning, C. D., Manning, C. D., & Schütze, H. (1999). *Foundations of statistical natural language processing*. MIT press.
- [25] Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, .
- [26] Minkov, E., & Cohen, W. W. (2008). Learning graph walk based similarity measures for parsed text. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing* (pp. 907–916). Association for Computational Linguistics.
- [27] Nakov, P., Ritter, A., Rosenthal, S., Sebastiani, F., & Stoyanov, V. (2016). Semeval-2016 task 4: Sentiment analysis in twitter. *Proceedings of SemEval*, (pp. 1–18).
- [28] Oneto, L., Bisio, F., Cambria, E., & Anguita, D. (2016). Statistical learning theory and elm for big social data analysis. *IEEE Computational Intelligence in TelligenCe mAGazine*, 11, 45–55.
- [29] Pang, B., & Lee, L. (2004). A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In *Proceedings of the 42nd annual meeting on Association for Computational Linguistics* (p. 271). Association for Computational Linguistics.
- [30] Pennington, J., Socher, R., & Manning, C. D. (2014). Glove: Global vectors for word representation. In *EMNLP* (pp. 1532–1543). volume 14.
- [31] Perozzi, B., Al-Rfou, R., & Skiena, S. (2014). Deepwalk: Online learning of social representations. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 701–710). ACM.

- [32] Poria, S., Cambria, E., Hazarika, D., Majumder, N., Zadeh, A., & Morency, L.-P. (2017). Context-dependent sentiment analysis in user-generated videos. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)* (pp. 873–883). volume 1.
- [33] Poria, S., Chaturvedi, I., Cambria, E., & Hussain, A. (2016). Convolutional mkl based multimodal emotion recognition and sentiment analysis. In *Data Mining (ICDM), 2016 IEEE 16th International Conference on* (pp. 439–448). IEEE.
- [34] Schmidhuber, J. (2015). Deep learning in neural networks: An overview. *Neural networks*, 61, 85–117.
- [35] Sharif Razavian, A., Azizpour, H., Sullivan, J., & Carlsson, S. (2014). Cnn features off-the-shelf: an astounding baseline for recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops* (pp. 806–813).
- [36] Socher, R., Huval, B., Manning, C. D., & Ng, A. Y. (2012). Semantic compositionality through recursive matrix-vector spaces. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning* (pp. 1201–1211). Association for Computational Linguistics.
- [37] Socher, R., Pennington, J., Huang, E. H., Ng, A. Y., & Manning, C. D. (2011). Semi-supervised recursive autoencoders for predicting sentiment distributions. In *Proceedings of the conference on empirical methods in natural language processing* (pp. 151–161). Association for Computational Linguistics.
- [38] Socher, R., Perelygin, A., Wu, J. Y., Chuang, J., Manning, C. D., Ng, A. Y., Potts, C. et al. (2013). Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the conference*

on empirical methods in natural language processing (EMNLP) (p. 1642).  
Citeseer volume 1631.

- [39] Speriosu, M., Sudan, N., Upadhyay, S., & Baldridge, J. (2011). Twitter polarity classification with label propagation over lexical links and the follower graph. In *Proceedings of the First workshop on Unsupervised Learning in NLP* (pp. 53–63). Association for Computational Linguistics.
- [40] Taboada, M., Brooke, J., Tofiloski, M., Voll, K., & Stede, M. (2011). Lexicon-Based Methods for Sentiment Analysis. *Computational Linguistics*, 37, 267–307. doi:10.1162/COLI\_a\_00049.
- [41] Tripathy, A., Agrawal, A., & Rath, S. K. (2016). Classification of sentiment reviews using n-gram machine learning approach. *Expert Systems with Applications*, 57, 117–126.
- [42] Tsivtsivadze, E., Pahikkala, T., Boberg, J., & Salakoski, T. (2006). Locality-convolution kernel and its application to dependency parse ranking. In *International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems* (pp. 610–618). Springer.
- [43] Tumasjan, A., Sprenger, T. O., Sandner, P. G., & Welpe, I. M. (2010). Predicting elections with twitter: What 140 characters reveal about political sentiment. *ICWSM*, 10, 178–185.
- [44] Violos, J., Tserpes, K., Psomakelis, E., Psychas, K., & Varvarigou, T. A. (2016). Sentiment analysis using word-graphs. In *WIMS* (p. 22).
- [45] Yousefi-Azar, M., & Hamey, L. (2017). Text summarization using unsupervised deep learning. *Expert Systems with Applications*, 68, 93–105.
- [46] Zare, H., & Niazi, M. (2016). Relevant based structure learning for feature selection. *Engineering Applications of Artificial Intelligence*, 55, 93–102.