WordNet-Based Text Categorization Using Convolutional Neural Networks

K. Premchander, S. S. V. N. Sarma, K. Vaishali, P. Vijaypal Reddy, M. Anjaneyulu and S. Nagaprasad

Abstract Text Categorization is a task of assigning documents to a fixed number of predefined categories. Concept is the grouping of semantically related items under a unique name. Dimensionality space and sparsity of the document representation can be reduced using concept generation. Conceptual representation of a text can be generated using WordNet. In this paper, an empirical evolution using Convolutional Neural Networks (CNN) for text categorization has been performed. The Convolutional Neural Networks exploit the one-dimensional structures of the text such as words, concepts, word embeddings, and concept embeddings to improve the categorical label prediction. The Reuter's dataset is evaluated with Convolutional Neural Networks on four categories of data. The representation of a text with word embeddings and concept embeddings together results to a better classification performance using CNN compared with word embeddings and concept embeddings individually.

K. Premchander (✉) · M. Anjaneyulu
Department of Computer Science, Dravidian University, Kuppam, India
e-mail: kpc.1279@gmail.com

M. Anjaneyulu
e-mail: anjan.lingam1@gmail.com

S. S. V. N. Sarma
Department of CSE, Vaagdevi College of Engineering, Warangal, India
e-mail: ssvn.sarma@gmail.com

K. Vaishali
Department of CSE, Jyothismathi Institute of Technology and Sciences, Karimnagar, India
e-mail: vaishali5599@gmail.com

P. Vijaypal Reddy
Department of CSE, Matrusri Engineering College, Hyderabad, India
e-mail: drvijayapalreddy@gmail.com

S. Nagaprasad
S.R.R. Government Arts & Science College, Karimnagar, India
e-mail: nagkanna80@gmail.com

B. Tiwari et al. (eds.), *Proceedings of International Conference on Recent Advancement on Computer and Communication*, Lecture Notes in Networks and Systems 34, https://doi.org/10.1007/978-981-10-8198-9_25

Keywords Text categorization · Convolutional Neural Networks
Word embeddings · Concept embeddings · WordNet

1 Introduction

With the advent of Internet, the usage of Internet users was a big explosion in the history of information technology. As the availability of information increased and people were unable to utilize large amounts of information. Text Categorization is the main source for handling and organizing text data in which it assigns one or more class labels to a document according to their content. WordNet contains a set of synsets. A synsets are group of words having similar meaning. In WordNet, it establishes different relationships such as hyponym, hyponymy, or ISA relation among synsets. WordNet can be used in various applications such as Natural Language Processing (NLP), Text Processing, and Artificial Intelligence.

Deep Neural Networks have been the inspiration to various NLP tasks, and the Recursive NN considers the semantics of a sentence through a tree structure which reduces the effectiveness when we want to consider the whole document. To find a solution to this problem, in latest studies the Convolution Neural Network (CNN) model is used for NLP. The problem of high dimensionality and sparsity of data is addressed using Deep Neural Networks [1]. Word embedding is a generation of concepts from words. There are many tools available for word embeddings such as word2vec, Sen2Vec, and Glove. Word embeddings are an important concept in deep neural networks. In Bag-of-words model, the document is represented as a vector which contains words and their weights. The word embedding is used to generate concept vectors for a given word vectors. By using concept vectors, a semantic relationship among the objects is established.

The paper has organized into five sections. The related work has explained in Sect. 2. The proposed model is described, the detailed flow of work explained in Sect. 3. The description about the dataset, the performance evaluation measures, and the experimental evaluations are presented in Sect. 4. The inference from the obtained results and possible extensions to the proposed work is presented in Sect. 5.

2 Related Work

The text documents are represented using linear approaches such as bag-of-words approach and *n*-gram approach as in [1, 2]. But, the nonlinear approaches are proved to be more effective for text categorization in [3, 4]. In this paper, the focus is on Convolutional Neural Network approach for text categorization as proposed in [5].

For Text categorization, the documents are represented with set of features such unigrams, bigrams, *n*-grams. But the traditional methods to represent the document using bag-of-words representation suffer with the problem of identifying the semantically relationships among the terms in the document. There are some features such as second-order *n*-gram tree structures [6], proposed to capture the semantic relations among the terms in the document. But these features are suffered with the problem of data sparsity which reduces the performance of the classifiers. Nowadays, the developments in the deep neural networks and word embeddings lead to address the problems such as data sparsity in NLP tasks. As in [7, 8], word embeddings capture the semantic and syntactic relations among the terms in the document. As proposed in [9], the Recursive Neural Network (RNN) is more effective for sentence representation in semantic space. But RNN uses tree structures to represent the sentence in a document which is not suitable for long sentences. Another drawback is its heavy time complexity. RNN model stores the semantics of the term for each word using hidden layers as in [10].

Text Categorization contains three topics such as feature engineering, feature selection and machine learning algorithms. The Bag-of-Words (BOW) model is used for feature engineering. Some other features such as noun phrases, POS tagging have proposed in [11] and tree kernels in [12]. Identification of the suitable features from the documents can improve the performance the classification system. The commonly used process for text classification is elimination of stop words from the document. There are some approaches such as information gain, chi-square indexing, mutual information is used to identify the importance of the features in [13]. There are various machine learning algorithms which are used to build a learning model for classification. These methods lead to the problem of data sparsity. Deep neural networks as in [14] and representation learning in [15] are proposed to come out from the high dimensionality space and sparsity of data problems in the document representation as in [6, 16].

The representation of a word in the form of a neuron is known as embedding of word in the form of a vector. The word embedding is used to measure semantic relationship between two words using word vectors. With word embeddings in neural networks, the performance of classification models is improved. As in [17], semi-supervised recursive auto-encoders are used to identify sentiment terms from the sentences. As in [18], RNN is used to predict the paragraph detection. As in [19], the sentiments in tensor networks are explored using recursive neural tensor networks. As in [20], the language models are built using RNN. In [21], RNN is used for dialogue act classification.

3 Proposed Work

The proposed model consists of various phases such as preprocessing the training and testing dataset, constructing a vector space model using word embeddings and concept embeddings of the document and finally building a classification model

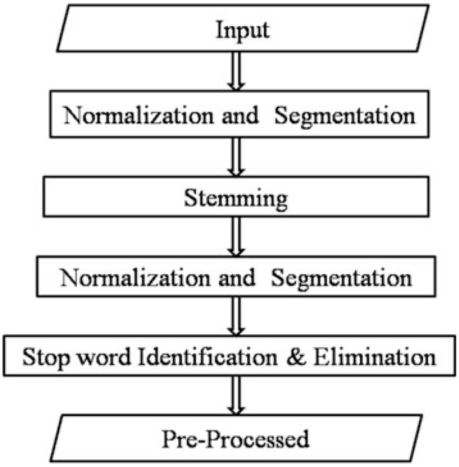
Input
Normalization and Segmentation
Stemming
Normalization and Segmentation
Stop word Identification & Elimination
Pre-Processed

Fig. 1 Preprocessing for the document

using CNN and assigning a class label to the test document using the classification model. The various steps are explained as follows:

3.1 Preprocessing

There are four steps involved in preprocessing the documents. In the first step, the non-content words are removed from the text. In the second step, the words are converted into their root forms. In third step, Part-Of-Speech (POS) tagging is assigned to each of the words. In the last step, stop words are removed from the text. The flow is shown in Fig. 1.

3.2 Proposed Model

The proposed model has presented in Fig. 2.

3.3 Convolutional Neural Network

Convolutional Neural Network (CNN) was proposed for handling various applications on digital image processing in [5]. CNN is a feed-forward neural network which contains a set of layers with a combination of pooling layers. In CNN, data are presented in the form of dimensional vectors in each layer. These low-dimensional vectors are called word embeddings. Using these embeddings, the

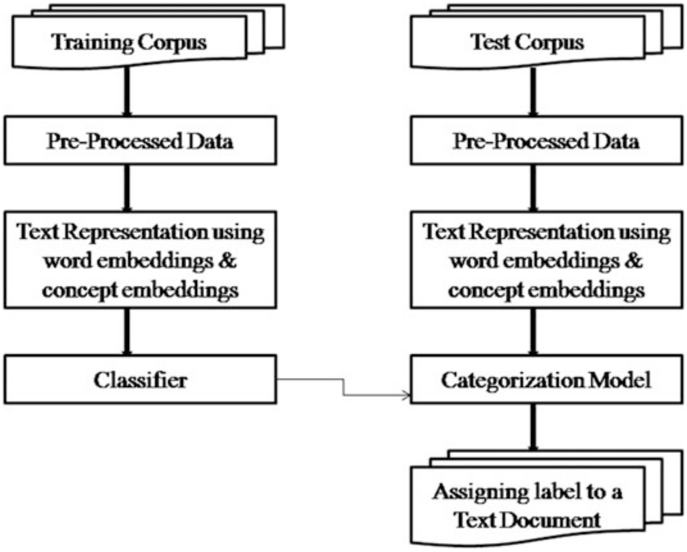
Training Corpus
Pre-Processed Data
Text Representation using word embeddings & concept embeddings
Classifier
Test Corpus
Pre-Processed Data
Text Representation using word embeddings & concept embeddings
Categorization Model
Assigning label to a Text Document

Fig. 2 Proposed model for text categorization

features which are presented in various layers can be identified. In CNN, the document is represented as a sequence of layers. The data presented in the layers are grouped into small regions. These regions are represented using low-dimensional vectors which are called word embeddings. The pooling layer in CNN combines the region of a word embeddings to represent a document vector to achieve maximum value per component. This method is proved to be better in performance for various applications compared with existing approaches.

3.4 Word2Vec

Each word in the text can be represented as a dimensional vector. These vectors are called word embeddings. Representation of words with its corresponding word embeddings is called Word2Vec model. The Word2Vec model is basically of two types such as Continuous Bag-Of-Words model (CBOW) and the Skip-Gram Bag-of-words model (SBOW). In CBOW model, the target words are derived from source context words, and in SBOW model, source words are predicted from the target words. CBOW model generates a smoothed curve on distributional information. The CBOW model is more useful for smaller datasets. SBOW model is more useful in the context of larger datasets. For text categorization, CBOW model is used to produce concept and word embeddings.

3.5 WordNet

WordNet is a thesaurus for the English language. It has many applications in various fields such as NLP, text mining, and information retrieval. WordNet is useful to find the semantic relationship among the words in a document. Many algorithms consider the height and depth of a word in the WordNet using synsets to get the closeness among the words based on its meaning. WordNet-based texts categorization has two stages. The first stage is learning phase, in which we get a new text by combining the terms with their relevant concepts. This enables to create categorical profiles based on characteristic features, and the second stage relates to the classification phase in which weights are given to the features in the categorical profiles.

3.6 Algorithm

Input: Training dataset and Test dataset

Step 1: Preprocess the data for both training and test datasets using various preprocessing techniques.
Step 2: Identify content terms from the training dataset and test dataset.
Step 3: Identify unique concepts using WordNet from content terms.
Step 4: Generate word embeddings and concept embeddings for content words and concepts derived from WordNet using Word2vec.
Step 4: Represent each document of training and test datasets in vector space model using word embeddings and concept embeddings.
Step 5: Construct a classifier using vector space model of documents with convolution neural networks.
Step 6: Identify the class label of test document by inputting the vector space model to the classification model.

4 Evaluation and Discussions

In this paper, Reuter's dataset is used to carry the experiments to label the documents with predefined categories. The precision, recall, and F_1 measures are used to measure the performance of proposed classification model.

4.1 Dataset Description

In this paper, the experiments were performed on the Reuter's dataset. It contains four categories of dataset namely CRAN, CISI, CACM, and MED. For empirical evaluations, 800 documents are considered based on the minimum number of sentences in the document. From 800 documents, 640 documents were considered as training set and the remaining were considered as test set. After applying various preprocessing techniques, the vector representation of the documents with their word embeddings and concept embeddings are inputted to CNN model for classification model generation.

4.2 Evaluation Measures

The performance of the obtained classification model is measured using precision, recall, and F_1 measures. The formulas for calculating precision, recall, and F_1 measures are as follows:

$$\text{Precision} = \frac{X}{X+Y}$$

$$\text{Recall} = \frac{X}{X+Z}$$

$$F_1 = \frac{2 * \text{Recall} * \text{Precision}}{\text{Recall} + \text{Precision}}$$

X is the number of documents retrieved and relevant, Y shows the number of documents retrieved but not relevant, Z is the number of documents relevant but not retrieved for a given query. F_1 measure is calculated using precision and recall.

4.3 Results

The efficiency of a classifier is measured on the test set by using precision, recall, and F_1 measures. Out of 800 documents, 640 documents are considered as training set and the remaining 160 are documents as test set. The results of our experiments are given in Table 1.

From the results, it is observed that the learning a classification model using word embeddings gains better precision, recall, and F_1 measure values compared with classification model with concept embeddings only. But combining word embeddings and concept embeddings together to train the Convolutional Neural Network leads to best classification performance.

Table 1 Precision, recall, and F_1 measure values using Convolution Neural Networks approach for word embeddings, concept embeddings and with their combination

Document representation	Precision	Recall	F_1 measure
Word embeddings	0.85	0.92	0.88
Concept embeddings	0.79	0.84	0.81
Word embeddings + Concept embeddings	0.89	0.95	**0.92**

5 Conclusions and Future Scope

The proposed model captures contextual information and constructs the representation of text using a Convolutional Neural Network for Text Categorization. It demonstrates that our model of Convolutional Neural Network gives best results using four different Reuter's datasets. In this paper, a new approach for Text Categorization is proposed by considering concept embeddings using WordNet. The experimental results with Reuters 21,578 dataset proved that the background knowledge to establish the relationships between words leads to effective classification performance. A possible extension to the proposed work is utilization of more suitable weighting techniques for representation of terms and concepts. It is also required to experiment with various possible Deep Neural Network approaches for different term representation techniques.

References

1. Thorsten: TC with SVM and Learn relevant features, ECML (1998)
2. Yang, E.T.: Semi supervised RNN classification of text with word embedding. JMLR Res. **5**, 361–397 (2004)
3. Dai, A.M., Le, Q.V.: Semi-supervised sequence learning. Adv. Neural Inf. Process. Syst. 3079–3087 (2015)
4. Zhang, X., Zhao, J., LeCun, Y.: Character-level convolutional networks for text classification. Adv. Neural Inf. Process. Syst. 649–657 (2015)
5. Johnson, R., Zhang, T.: Semi-supervised convolutional neural networks for text categorization via region embedding. Adv. Neural Inf. Process. Syst. 919–927 (2015)
6. Aggarwal, C.C., Zhai, C.: A survey of text classification algorithms. Mining text data, 163–222 (2012)
7. Dinu, G.: Predict a systematic compare of context counting using context predict semantic vector. ACL, 238–247 (2012)
8. Vincent, P.: ANN probabilistic model of a language. JMLR **3**, 1137, 1155 (2003)
9. Bengio, Y., Courville, A., Vincent, P.: Representation learning: a review and new perspectives. IEEE Trans. Pattern Anal. Mach. Intell. **35**(8), 1798–1828 (2013)
10. Bottou, L.: Learning of gradient in networks using CNN. In: Proceedings on Neuro-Nımes, vol. 91 (1999)
11. Bloehdorn, S., Hotho, A.: Boosting for text classification with semantic features. In: WebKDD, pp. 149–166 (2004)
12. Johnson, M.: Maxent discriminative re-ranking and Coarse-to-fine n-best parsing. In: Association for Computational Linguistics, pp. 173–180 (2005)

13. Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K., Kuksa, P.: Natural language processing (almost) from scratch. J. Mach. Learn. Res. **12**, 2493–2537 (2011)
14. Cover, T.M., Thomas, J.A.: Elements of Information Theory. Wiley (2012)
15. Glänzel, Wolfgang, Thijs, Bart: Using ‘core documents’ for detecting and labelling new emerging topics. Scientometrics **91**(2), 399–416 (2012)
16. Hinton, G.E, Salakhutdinov, R.R.: Reducing the dimensionality of data with neural networks. Science, **313**(5786), 504–507 (2006)
17. Huang, E.H., Socher, R., Manning, C.D., Ng, A.Y.: Improving word representations via global context and multiple word prototypes. In: Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers, vol. 1, pp. 873–882. Association for Computational Linguistics (2012)
18. Kalchbrenner, N., Blunsom, P.: Recurrent convolutional neural networks for discourse compositionality. arXiv preprint arXiv:1306.3584 (2013)
19. Klementiev, A., Titov, I., Bhattarai, B.: Inducing crosslingual distributed representations of words (2012)
20. Mikonos, T.: Distributed representations of sentences and docs. ICML (2014)
21. Sutskever, I.: Distributional representations of words and phrases and their composite. NIPS, 3111–3119 (2013)
22. Mikolov, T., Yih, W., Zweig, G.: Linguistic regularities in continuous space word representations. In hlt-Naacl **13**, 746–751 (2013)