Document Modeling with Hierarchical Deep Learning Approach for Sentiment Classification

Monalisa Ghosh
Research scholar: Dept. of CSE
National Institute of Technology, Durgapur, India
monalisa_05mca@yahoo.com

Goutam Sanyal
HOD: Dept. of CSE
National Institute of Technology, Durgapur, India
nitgsanyal@gmail.com

ABSTRACT
Sentiment analysis has recently been considered as most active research field in NLP domain. Deep learning is a growing trend of machine learning due to its automatic learning capability with impressive results across different NLP task. In this paper a model is proposed to analyze the deep sentiment representation based on CNN and LSTM (modified version of RNN) network. We aim to improve the performance of traditional machine learning method by merging them with deep learning techniques to tackle the challenge of sentiment prediction of massive amount of unsupervised product review dataset. We make our model first learn to sentence representation with CNN. Next, the semantics of sentences are encoded with LSTM network for document representation. We conduct experiments on two review datasets based on movie review with evaluation metric 'accuracy'. The result shows that proposed model outperformed traditional machine learning as well as baseline neural network model

CCS Concepts
- **Computing methodologies~Information extraction**

Keywords
Sentiment Analysis, Deep learning, traditional machine learning, Convolutional neural network (CNN), recurrent neural networks, LSTM, embedding algorithm etc.

1. INTRODUCTION
An opinion is a viewpoint or judgment about a specific thing and act as a key influence to an individual process of decision making. Opinion plays an important role in human being's life, because of People's beliefs and the choices they make are always depending on how others see and evaluate the world.

Sentiment analysis, also known as opinion mining is the process of determining the emotional tunes behind a series of words, in recent years, it has been receiving a lot of attention from the researchers. This field has many interrelated sub problems rather than a single problem to solve, which makes this field more challenging. Sentiment classification process can be done in three levels mainly: document level, sentence level and feature level.

In Document level, the entire document is classified based on the positive or negative opinion expressed by the authors.

Sentiment classification at the sentence level, considers individual sentence to identify whether the sentence is positive or negative. In feature level, classify the sentiment with respect to the specific aspects of entities. In this study, document level sentiment analysis has been taken into consideration.

Sentiment classification generally relies on two types of techniques, i.e., lexicon based and machine learning based techniques.

Machine learning approaches classify the sentiments based on training as well as test data sets. The lexicon based approach doesn't require any prior training data set for sentiment analysis. It uses large amount of linguistic resources with predefined list of words, where each word is associated with a specific sentiment. There are few researchers applied hybrid approaches by combining both approaches to machine learning and lexical to improve the sentiment classification performance. Deep learning has an extremity over the classical machine learning algorithms to perform the task of sentiment analysis, because of its capability to handle the challenges faced by sentiment analysis. In last few years, many researchers have tried to merge the concept of traditional machine learning with deep learning to develop more precise sentiment classifier to improve the classification task on any text content with minimal constraints.

In this paper, we approach a model based on two deep neural methods CNN and LSTM to learn continuous document representation for sentiment classification. The method is on the basis of the principle of compositionality. Specifically, the approach composes document modeling in two steps. In the first step, it uses convolutional neural network (CNN) to produce sentence representations from word representations. Next, LSTM applied for document representation from sentence level. Afterwards, max over pooling technique applied over the feature map and take the maximum value of features. Finally, the pooled features are used in a softmax layer for classification.

The remaining part of this paper is the following.

Section 2 presents the existing works which can relate to our approach. Section 3 discusses the basic Procedure of Sentiment Classification of review dataset. Section 4 addresses the background including LSTM networks and convolution operators. We then describe our architectures for sentence modeling and document modeling. In Section 5, and report experimental results in Section 6.

ICDSP 2018, February 25–27, 2018, Tokyo, Japan

ACM ISBN 978-1-4503-6402-7/18/02…$15.00
DOI: https://doi.org/10.1145/3193025.3193046

2. RELATED WORK

Neural network models based on deep learning have achieved great prosperity in many NLP related tasks. Document level Sentiment classification is a basic problem in sentiment analysis with the aims of identifying the sentiment label of a document. The first successful architecture of deep learning is based on word modeling with semantic vector space [1][2][3]. They introduced word embedding technique, where each word is represented as a real valued vector. The researcher [4] described a neural bag of words model that uses the dynamic k-max pooling operator. This model achieves good performance on sentiment classification task without feature engineering. Kim [5] applied simple convolutional neural networks with static vectors and got excellent results on different datasets.

Zhang et al.[6] in 2015 proposed a character-level CNN for text classification and achieve competitive results. Different types of LSTM models are applied in previous research. Specially, the model which encode the assumption and hypothesis separately with two LSTMs [8], a shared LSTM applied by Rockt ̈aschel et al[7].The tree structured LSTMs proposed by Tai et al. for sentiment classification. Few researchers combined the CNN and LSTM model for sentence classification [9]. The hierarchical structure of CNN and LSTM also applied by Tang et al.[10]. They first use of CNN or LSTM to produce sentence representations from word representations. Next, gated recurrent neural network is applied to encode semantics of sentences for document modeling.

Tai et al. (2015) [11] proposed to combine the standard LSTM to tree-structured topologies and got the superior results over a sequential model of LSTM. In [12], the researcher proposes three different models for multi-task learning with recurrent neural networks.

Vo et al.[13] introduced a Vietnamese corpus for sentiment classification, which collected from Vietnamese commercial web pages. They applied CNN and LSTM to generate information channels for Vietnamese sentiment analysis.

Wei et al. approached [14] a transfer learning framework based on a convolutional neural network and a long short-term memory model. This architecture automatically identify whether a post expresses confusion, determine the urgency of the post and finally polarity will be classified.

3. THE BASIC PROCEDURE

In our work we follow the basic procedure to perform sentiment classification by machine based learning method. The classification method summarized into several steps as described below.

A. The review dataset is preprocessed in such a way that supervised learning algorithm can be applied. Data processing is required to remove noisy, inconsistent and incomplete by considering tokenization, stop words removal, stemming method.

B. The features considered for this proposal are Unigram features and two-word (bi tagged) feature in POS based fix pattern. Composite feature set created by combining unigram and bi-tagged feature.

C. Tf-Idf method used to assign a particular score for each individual feature and top ranked features will be considered as input to supervised ML algorithm

D. Finally, train the supervised machine learning classifier SVM and NB with the different feature vector for classification the dataset. SVM and NB classifier considered as baseline method for evaluate our proposed model.

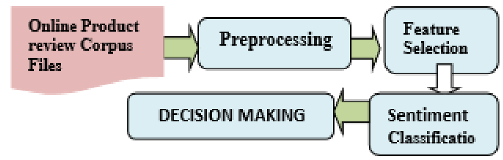

Figure 1. Architecture of the basic framework for sentiment classification using machine learning method.

4. PROPOSED MODEL

Long Short-Term Memory (LSTM) network performs surprisingly well in previous studies regarding sentiment classification problem. This is a modified version of the recurrent neural networks by handling the issue to model long range dependency. The key element of this deep learning network is memory cells, where the information can be stored. In our approach, to model the document semantic representations, we adopt Long Short-Term Memory (LSTM) network and convolutional neural network (CNN) through a hierarchical structure combined of word level and sentence level.

We consider CNN with multiple convolutional filters to catch local features [10] from every possible window of different size, where each filter considered as a feature detector. Max-pooling layer is used in our method to maintain the uniformity in the size of the sentence vector, because the output length of convolution layer based on the length of input sentence. The performance of this layer for classification task is better in comparison with simple averaging. We prefer the CNN and LSTM are very latest composition model for document level as well as sentence level classification [5][4][15]. In our approach, we don't require any external parser or parsing result to capture long distance dependencies for high quality sentence representation. In recent years, CNN based model [23] performs surprisingly well in wide range of NLP task such as sentence and document modeling for class prediction [4][5],object recognition [17],and other traditional NLP tasks [16] etc.

Let's discuss about how this combination of LSTM and CNN network use to learn document semantic representations. At first we define word vector representation. The generic word vector as a pre-trained can be extract [2] [1] with embedding learning algorithm (word2vec). Sometimes word vector captured in an unsupervised way with semantic and syntactic information. In our approach, we consider the word vectors that are not trained. In general, we assume a product p ε P contain a review from an user. This review can be represent as a document d with N number of sentences {s1,s2,…,.sN}.Then consider, the length of the sentence J-th is l_J. The J-th sentence sJ be formed of l_J words as $\{w_1^J, w_2^J, \ldots w_{l_J}^J\}$, convolutional filters of CNN apply to extract local features from sentence sJ.

4.1 Word Level

Each word is represented as continuous and real valued vector and we embedded the word in a sentence into low dimensional space [21]. This process is known as word embedding [18]. In this way, each word w_i^J is mapped to its embedding representation $w_i^J \in R^d$ and the filter $F \in R^{d \times |L|}$, here d is the dimension of word vector and L is the size of the window.

LSTM network can produce hidden state representation [19]. For the given word w_1^J of sentence sJ, the current state and hidden state of the cell are respectively c_t^J and h_t^J can be updated with the previous cell state c_{t-1}^J and hidden h_{t-1}^J at time step $\boldsymbol{t}$ in the following manner.

$$i_t^J = \sigma (W_i . [h_{t-1}^J ; w_t^J] + b_i) \quad (1)$$
$$f_t^J = \sigma (W_f . [h_{t-1}^J ; w_t^J] + b_f) \quad (2)$$
$$o_t^J = \sigma (W_o . [h_{t-1}^J ; w_t^J] + b_o) \quad (3)$$
$$\hat{c}_t^J = \tanh (W_c . [h_{t-1}^J ; w_t^J] + b_c) \quad (4)$$
$$c_t^J = f_t^J \odot c_{t-1}^J + i_i^J \odot \hat{c}_t^J \quad (5)$$
$$h_t^J = o_t^J \odot \tanh(c_t^J) \quad (6)$$

Where σ denotes the logistic sigmoid function and ʘ represents the element-wise multiplication. Note that (i,f,o) are gate activation with the forget gate ft, the input gate it, and the output gate ot. These gates decide how to update the cell from one state to another state. $W_{\{i,f,o,c\}}$,$b_{\{ i,f,o,c\}}$ are the set of parameters that we have to train.

4.2 Sentence Modeling

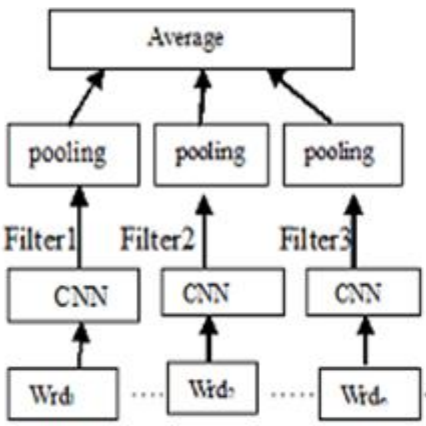

Figure 2. An example of sentence modeling

CNN composed of multiple layers with a number of shared parameters. Each layer performs a specific task of alternating its input into useful representation. We try CNN, which follows as the second layer to compose the sentence model. This convolution neural network consists of multiple convolutional filters of different width. In this work, we apply three convolutional filters of different width in order to fetch the local semantics of N-grams such as unigrams, bigrams and trigrams in a sentence. Suppose, a sentence as input of n words $[w_1,w_2,w_3\dots,w_k,\dots w_n]$. Let us consider p and b are shared parameters and l be the window size of a filter F. For CNN layer w_i ε R^d be the embedding representation of word w_i with dimension d. In general, the input of a linear layer is the concatenation of word embeddings as $w_{i:i+l-1} = \{w_i, w_{i+1}, \dots w_{i+l-1}\}$ ε R^{dl}. Now we can determine the output of linear layer as follows

$$O= p. w_{i:i+l-1}+b \quad (7)$$

Where p ε $R^{l_o \times dl}$ and b ε $R^{l_o \times dl}$ are the shared parameters, l_o considered as output length of linear layer. Next, to extract the global features of a sentence, we feed the output of linear layer to the max-over-time pooling layer. A max-over-time pooling layer is added on top of the convolution neural network. Finally, the pooled features are used in a softmax layer for classification.

4.3. Document Modeling:

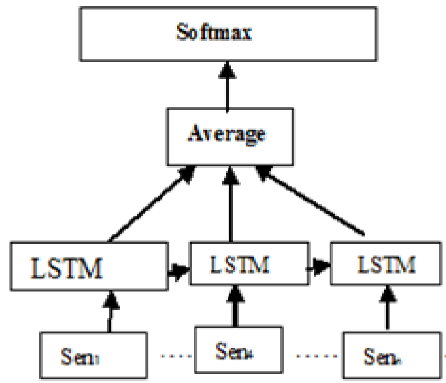

Figure 3. An example of document modeling

Our proposed architecture is not limited to sentence modeling, the obtained sentence vectors are provided to compose the document model. Let's consider the processed document as input to our model, where the document consists of n number of sentences $[s_1,s_2,s_3,\dots,s_n]$. We apply LSTM for document composition in the same way as implemented for sentence modeling. The hidden state of LSTM network feed forward to the mean pooling layer. In this way, we get the mean value for the sentences.

CNN will be placed just top of LSTM like sentence modeling. Finally, the classification task can be completed with pool features by the max pooling and softmax layer.

5. EXPERIMENTS

5.1 Dataset: In this section, we conduct experiment to evaluate the effectiveness of our model on various benchmark data sets. Our main task is to perform sentiment classification on these different domain data set. Specifically, we use movie review and movie review datasets to evaluate the performance of our model and compare to existing base line model. The evaluation metric of these two datasets are *Accuracy.* We consider 80% of the data for training purpose, 10% for validation, and the remaining 10% for testing, unless stated otherwise.

SST2: Stanford Sentiment Treebank is an extension of movie review dataset. It is same as SST1, but this review dataset includes only binary labels.

IMDB The movie review data set contains 5,331 positive and 5,331 negative reviews, movie reviews [22]

5.2 Pre-processing:

Tokenization or segmentation: It can be done by splitting documents into a list of words. In our experiment we use the *Stanford Tokenizer* to obtain the tokens.

Removal of stop words: Some of the high frequency stop words are to be removed (prepositions, irrelevant words).

5.3 Experimental Settings:

We implement our model based on Python library. The hyper-parameter settings of the deep learning neural networks may depends on the data set being used for the experiment. For product review data set, we considered hyper parameters such as number of filters, filter length in CNN; memory dimension in LSTM; which layer to apply, etc. In our proposed architecture, we use one CNN layer and 1 LSTM layer. The filter size Or filter length 1,2,3 are used for single convolutional layer to captured the local features.

5.4 Baseline Methods

We compare our proposed model with the following baseline methods including traditional machine learning approaches such as SVM,NB etc. for document level sentiment classification.

5.4.1. Traditional paradigm:

- SVM+Ngrams: SVMs based methods are elaborate [10] with Unigram, Bigram and Unigram + Bigram as features.
- NB + Ngrams: Naïve Bayes classifier with Unigram features and Bigram features [4] also considered as baseline method.
- NBSVM-bi: SVM and Multinomial NB with bigram features are specified in [20] previous research.
- Text Features are defined based on [24], which includes word and character ngrams, sentiment lexicon features, cluster features etc.

5.4.2. Neural network Paradigm:

- CNN-rand, CNN-static, CNN-nonstatic and CNN-multichannel: These four variants of the CNN model proposed by [5] based on the different usage of word vectors for sentiment classification purpose.
- Standard-LSTM: Standard Long Short Term Memory Network by [11] considered as baseline to compare with our approach.

6. RESULT AND DISCUSSION

The experimental results of our proposed model on different datasets are shown in Table 1. This result focuses its effectiveness in comparison with other baseline method. We evaluate each dataset with the metric 'Accuracy' and the best method in each dataset will be marked in bold.

Table 1. Comparisons with baseline models on movie review dataset. Binary is a 2-classification task. The first block contains other baseline methods of traditional approach. The second blockare methods related to convolutional neural networks. The third block contains CNN method for word representation and Char representation. The fourth block contains methods using LSTM. The last block is our model.

Model	**SST-2**	IMDB
SVM+Ngram (Socher et al., 2013)	79.4	--
NB+Ngrams (Kalchbrenner et al., 2014)	80.5	--
NBSVM-bi (Wang and Manning, 2012)	---	91.2
SVM + text feature (Kiritchenko et al)	87.8	—
CNN-rand (Kim, 2014)	82.7	—
CNN-static(Kim, 2014)	86.8	—
CNN-nonstatic (Kim, 2014)	87.2	—
CNN-multichannel(Kim, 2014)	85.1	—
Standard-LSTM (Tai et al.,2015)	86.7	----
bi-LSTM (Tai et al., 2015)	86.8	---
SA-LSTM (Dai and Le, 2015)	88.1	92.8
LSTM+CNN (Our implementation)	86.7	88.9
CNN+Ngram (Our implementation)	89.7	93.2

From Table 1, we can see that the neural network approach (CNN, LSTM) outperform the traditional methods for different datasets. The neural network approach more effective in composing the semantic representation of text data. However, it must say that SVM classifier is extremely strong method with Ngram features than other baseline method in comparison.

CNN based approach achieve better accuracy when comparing the Convolutional Neural Networks and Recurrent neural network (RNN) to Recursive NNs using the movie review data set. This is because RNN suffers from the vanishing gradient and Gradient Explosion problems. The improved version of RNN such as Long Short Term Memory Models (LSTM) works surprisingly very well for sentence as well as document modeling.

7. CONCLUSION:

In this paper, we introduce neural network model () for document level sentiment classification. The approach is to first learns sentence representation with Long Short Term Memory network and after that the semantics of sentences are encoded with convolutional neural network for document representation. We conduct experiments on two review datasets based on movie review with evaluation metric 'accuracy'. The result shows that proposed model outperformed traditional machine learning as well as baseline neural network model with better accuracy on sentiment classification dataset.

8. REFERENCE:

[1] J. Pennington, R. Socher, and Christopher D Manning, "Glove: Global vectors for word rep-resentation" Proceedings of EMNLP, 12:1532–1543,2014.

[2] D. Bahdanau, K. Cho, and Y. Ben-gio. "Neural machine translation by jointly learning to align and translate". arXiv preprint arXiv:1409.0473, 2014.

[3] T. Mikolov, I. Sutskever, K. Chen, G. S Cor-rado, and J. Dean. "Distributed representa-tions of words and phrases and their compositionality". In Proceedings of NIPS, pp. 3111–3119, 2013.

[4] N. Kalchbrenner, E. Grefenstette, and P. Blun-som. "A convolutional neural network for mod-elling sentences". arXiv preprint arXiv:1404.2188,2014.

[5] Y. Kim. "Convolutional neural networks for sen-tence classification". In Proceedings of EMNLP, pp. 1746–1751, Doha, Qatar, October,2014.

[6] X. Zhang, J. Zhao, and Y. LeCun. ."Character-level convolutional networks for text classification". arXiv preprint arXiv:1509.01626,2015.

[7] T. Rockt ̈aschel, E. Grefenstette, K. Moritz Hermann, Tom ́aˇs Koˇcisk`y, and P. Blunsom, "Reasoning about entailment with neural attention". In Proceedings of the 2016 ICLR, San Juan, Puerto Rico,2016.

[8] Samuel R Bowman, Gabor Angeli, Christopher Potts, and Christopher D Manning, "A large annotated corpus for learning natural language inference". In Proceedings of the 2015 EMNLP, pp 22–32, Lisbon, 2015, Portugal.

[9] S Lai, L Xu, K Liu, and J Zhao, "Recurrent convolutional neural networks for text classification" In Twenty-Ninth AAAI Conference on Artificial Intelligence,2015.

[10] D. Tang, B. Qin, and T. Liu. "Document modeling with gated recurrent neural network for sentiment classification". In Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, pp. 1422–1432,2015

[11] K. Sheng Tai, R. Socher, and Christopher D. Manning. "Improved semantic representations from tree-structured long short-term memory networks". In Proc. ACL.

[12] Liu et al., Pengfei Liu, Xipeng Qiu, Jifan Chen, and Xuanjing Huang. Deep Fusion LSTMs for Text Semantic Matching. In ACL, 2016.

[13] Quan Hoang Vo, Huy Tien Nguyen, Bac Le, Minh-Le Nguyen, "Multi-channel LSTM-CNN model for Vietnamese sentiment analysis" Proceedings of KSE October,2017.

[14] X. Wei, H. Lin, L. Yang and Y. Yu, "A Convolution-LSTM-Based Deep Neural Network for Cross-Domain MOOC Forum Post Classification", Information **2017**, 8, 92; doi:10.3390/info8030092.

[15] Rie Johnson and Tong Zhang, "Effective use of word order for text categorization with convolutional neural networks". NAACL,2015.

[16] Ronan Collobert and Jason Weston, "A unified architecture for natural language processing: Deep neural networks with multitask learning". In Proceedings of ICML, pp. 160–167. ACM,2008.

[17] Y. LeCun, L ́eon Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition". Proceedings of the IEEE, 86(11): pp.2278–2324.1998

[18] Yoshua Bengio, R ́ejean Ducharme, Pascal Vincent, and Christian Janvin. 2003. "A neural probabilistic language model. Journal of Machine Learning Research",

[19] S. Hochreiter and J. Schmidhuber. "Long short-term memory. Neural computation", 9(8):pp.1735–1780, 1997.

[20] Sida Wang and Christopher D Manning, "Baselines and bigrams: Simple, good sentiment and topic classification". In Proceedings of ACL: Short Papers-Volume 2, pp. 90–94. Association for Computational Linguistics,2012.

[21] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Cor-rado, and Jeff Dean, "Distributed representations of words and phrases and their compositionality" In Proceedings of NIPS, pp. 3111–3119,2013.

[22] Peng, Long & Ding., "Feature selection based on mutual information: criteria of max-dependency", max-relevance, and min-redundancy. IEEE Trans. Pattern Anal. Mach.Intell.27:1226-1238,2005.

[23] Mingbo Ma, Liang Huang, Bing Xiang, and Bowen Zhou,. "Dependency-based convolutional neural networks for sentence embedding". In Proceedings of ACL-IJCNLP, volume 2, pp. 692-700, 2015.

[24] Svetlana Kiritchenko, Xiaodan Zhu, Colin Cherry, and Saif Mohammad. "Detecting aspects and sentiment in customer reviews". In Proceedings of the 8th Internation- Workshop on Semantic Evaluation (SemEval 2014) Nrccanada 2014, pp. 437–442, 2014.