International Journal of Synthetic Emotions
Volume 9 • Issue 1 • January-June 2018

Sentiment Analysis in the Light of LSTM Recurrent Neural Networks

Subarno Pal, Academy of Technology, Hooghly, India
Soumadip Ghosh, Academy of Technology, Hooghly, India
Amitava Nag, Academy of Technology, Hooghly, India

ABSTRACT

Long short-term memory (LSTM) is a special type of recurrent neural network (RNN) architecture that was designed over simple RNNs for modeling temporal sequences and their long-range dependencies more accurately. In this article, the authors work with different types of LSTM architectures for sentiment analysis of movie reviews. It has been showed that LSTM RNNs are more effective than deep neural networks and conventional RNNs for sentiment analysis. Here, the authors explore different architectures associated with LSTM models to study their relative performance on sentiment analysis. A simple LSTM is first constructed and its performance is studied. On subsequent stages, the LSTM layer is stacked one upon another which shows an increase in accuracy. Later the LSTM layers were made bidirectional to convey data both forward and backward in the network. The authors hereby show that a layered deep LSTM with bidirectional connections has better performance in terms of accuracy compared to the simpler versions of LSTM used here.

KEYWORDS

Bidirectional LSTM, Long Short-Term Memory, LSTM, Recurrent Neural Network, Sentiment Analysis

1. INTRODUCTION

Sentiment analysis is a computational method of identifying or categorizing opinions expressed in a text, which is also one of the very active fields of research (Manning et al., 2008). Text obtained from different sources like user reviews and micro blogs express user's view or attitude towards the particular product or event etc. Sentiment analysis of small text is challenging because they are contextually limited. Decisions are taken on the basis of limited number of words used by the user. We deal with Sentiment Analysis as a supervised learning process where each data element (text reviews) are labeled as either 'positive' or 'negative' (Pang, Lee, & Vaithyanathan, 2002). Machine learning models are trained with word embeddings on these datasets and their accuracy is measured on the basis of their performance.

Artificial neuron network, a computational model developed on the basis of the structure and functions of biological neural networks, has achieved huge success over other machine learning techniques in sentiment analysis (Yoon, 2014; Socher, Pennington, Huang, Ng, & Manning, 2011; Xiong, Zhong, & Socher, 2002). Deep neural networks (DNNs) have recently achieved significant performance gains in a variety of NLP tasks such as language modeling (Bengio, Ducharme, Vincent, & Jauvin, 2003), sentiment analysis (Socher et al., 2013), syntactic parsing (Collobert & Weston, 2008), and machine translation (Lee, Cho, & Hofmann, 2016). A recurrent neural network

DOI: 10.4018/IJSE.2018010103

(RNN) is a special type of neural network, where connections are made between units which form a directed cycle, which allows it to exhibit a dynamic temporal behavior for the model. An RNN has an Input layer, variable number of hidden layers and finally one output layer. Basic RNNs are a network of neuron-like nodes, each with a directed (one-way) connection to every other node in which all connection (synapse) has a modifiable real-valued weight. These weights are constantly updated through successive iterations of the neural network. RNN are mostly used in hand writing recognition and speech recognition. For text classification purpose RNN is certainly more effective than any other variations of neural networks in practice.

A special variation of RNN, long short term memory (LSTM) networks is discussed. LSTM showed a striking accuracy in language modeling and speech recognition. We will be varying different forms of LSTM for our text classification purpose. A LSTM network contains LSTM units along with the input and output network layer units. A LSTM unit is capable of remembering values for either long or short time periods (Hochreiter & Schmidhuber, 1997) and it uses no activation function within its recurrent components. The stored value is not iteratively squashed over time, thus solving the vanishing gradient problem. LSTM blocks contain three or four "gates" that control information flow, implemented using the logistic function to compute a value between 0 and 1. An "input" gate controls the extent to which a new value flows into the memory, a "forget" gate controls the extent to which a value remains in memory, and an "output" gate controls the extent to which the value in memory is used to compute the output activation of the block.

Stacked LSTM is a particular model where one LSTM layer is stacked upon another to form a stack of LSTMs in the network. We preferred to form a stack of three LSTM layers to form a deep RNN. The detailed structure of the LSTM models will be described further. We in our study discuss the performance of various types of RNN and LSTMs on a large the IMDB dataset of reviews.

2. RELATED WORKS

Sentiment Analysis has been a favorite chapter for researcher for quite a long time. We discuss some of the important related works in this context.

In Socher et al. (2013b), the authors propose the Recursive Neural Tensor Network (RNTN) architecture, which represents a phrase through word vectors with a parse tree and then compute vectors for higher nodes in the tree using the same function, when trained on the new tree-bank, this model outperformed all previous methods on several metrics.

Santos et al. (n.d.) propose a new deep convolution neural network that exploits from character- to sentence-level information to perform sentiment analysis of short texts. Their approach for two corpora of two different domains: the Stanford Sentiment Tree-bank (SSTb), which contains sentences, from movie reviews; and the Stanford Twitter Sentimentcorpus (STS), which contains Twitter messages.

Mihalcea et al. (2007) make use of English corpora to train sentence-level subjectivity classifiers in Romanian language using two approaches, which they claimed can be applied to any language, and not only Romanian. In the first approach, they use a bilingual dictionary to translate an existing English lexicon to build a target language subjectivity lexicon. In the other one, they generate a subjectivity-annotated corpus in a target language by projecting annotations from an automatically-annotated English corpus. In Zhou et al. (2016) the authors propose an attention-based LSTM network to learn the document presentations of reviews in English and Chinese exploring word vectors as text representation.

Hochrieter and Schmidhuber (1997) pointed out that recurrent backpropagation or simple neural networks are extremely inefficient or fail miserably to learn information that is largely extended over time. long short term memory networks – usually just called "LSTMs" a special kind of RNN, capable of learning long-term dependencies, was proposed by them. LSTMs work tremendously well on a large variety of problems, and are now widely used.

Our work is quite different from the others because we would try to deploy different types of LSTMs in Sentiment Analysis. Sak et al. (n.d.) in their work for speech recognition, presented a novel LSTM based RNN architectures which make more effective use of model parameters to train acoustic models for large vocabulary speech recognition. Comparison between LSTM, RNN and Deep Neural Network models at various numbers of parameters and configurations, which showed that LSTM models converge quickly and give state of the art speech recognition performance for relatively small sized models. In our study we compare different forms of RNN specially LSTMs for sentiment analysis and show the relative result.

3. PROPOSED METHODOLOGY

3.1. Dataset Description

Given a sentence as a review we compute the users view or perspective towards any product or event in that sentence. For this experimental purpose we use Keras (Francois et al., 2015) IMDB dataset of 25,000 movies reviews for training and same for testing from IMDB, each labeled by sentiment (positive/negative). Reviews have been preprocessed, and each review is encoded as a sequence of word indexes (integers).Files must be in MS Word only and should be formatted for direct printing, using the CRC MS Word provided. Figures and tables should be embedded and not supplied separately.

3.2. Data Preprocessing and Cleaning

Reviews have been preprocessed in Keras IMDB dataset, each review is encoded as a sequence of word indexes (integers). For convenience, words are indexed by overall frequency in the dataset, so that for instance the integer “4” encodes the 4th most frequent word in the data. This allows for quick filtering operations on the basis of frequency of the words, like selecting the most frequent keywords etc. The data after being preprocessed is sent into Embedding layer which is further sent to LSTM layers and the activation layer that predicts the output.

Embedding layer: The layer that is common to every model we shown is the embed-ding layer. According to Keras documentation embedding layer turns positive integers (indexes) into dense vectors of fixed size e.g. [[4], [20]] -> [[0.25, 0.1], [0.6, -0.2]].

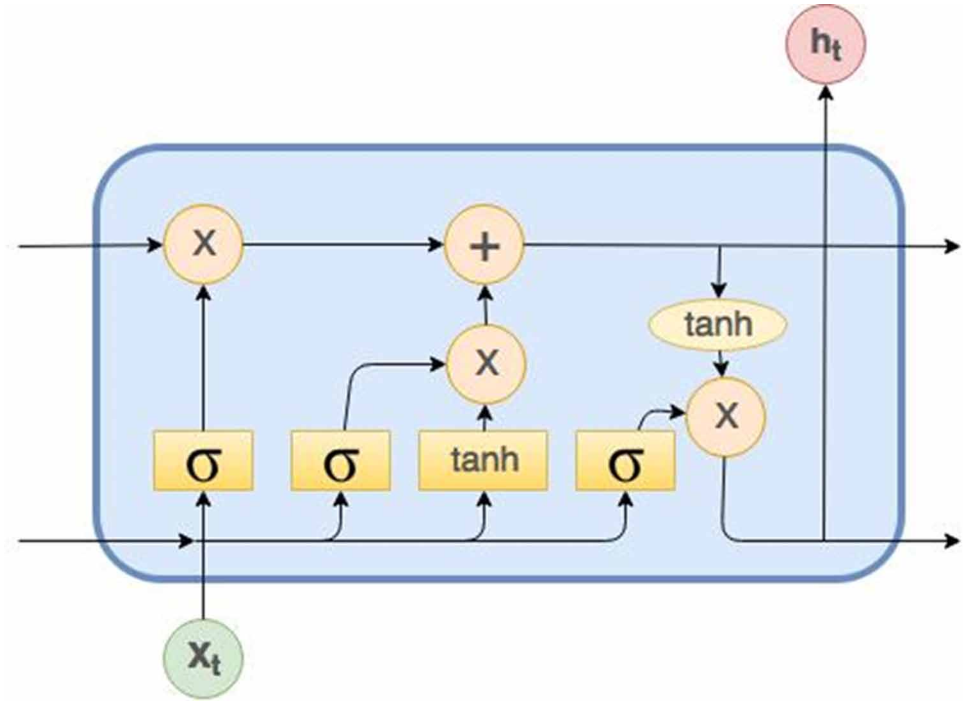
3.3. LSTM Network

The LSTM consists of units or memory blocks in the recurrent hidden layer, which contains memory cells with self-connections storing the temporal state of the network. In addition to this the network has special multiplicative units called gates to control the flow of information in the network. Each memory block in the original architecture contained an input gate and an output gate (Gers, Schmidhuber, & Cummins, 2000). The input gate controls the flow of input activations into the memory cell and the output gate controls the output flow of cell activations into the rest of the network. The forget gate was added to the memory block that scales the internal state of the cell before adding it as input to the cell through the self-recurrent connection of the cell, therefore adaptively forgetting or resetting the cell’s memory. In addition, the modern LSTM architecture might also contain peephole connections from its internal cells to the gates in the same cell which help to learn precise timing of the outputs (Gers, Schraudolph, & Schmidhuber, 2003) (see Figure 1).

An LSTM network computes a mapping from an input sequence $x = (x_1, ..., x_T)$ to an output sequence $h = (h_1, ..., h_T)$ by calculating the network unit activations using the following equations (“Understanding LSTM Networks,” 2015) iteratively from t = 1 to T:

$$f_t = \sigma(W_f \cdot \left[h_{t-1}, x_t\right] + b_t \quad (1)$$

Figure 1. Single LSTM cell



$$i_t = \sigma(W_i \cdot \left[h_{t-1}, x_t\right] + b_i \quad (2)$$

$$\tilde{C}_t = \tanh(W_c \cdot \left[h_{t-1}, x_t\right] + b_{ct} \quad (3)$$

$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t \quad (4)$$

$$o_t = \sigma\left(W_o \cdot \left[h_{t-1}, x_t\right] + b_o\right) \quad (5)$$

$$h_t = o_t * \tanh\left(C_t\right) \quad (6)$$

where W denotes the weight matrices, Ct is the cell state and b is the input bias vector. And the i, f, o are the input, forget and the output gate layer. Cell out activation function in this paper is tanh.

The output layer is the last layer in the network which is used to predict the sentiment. The output later only consists of the activation function for the prediction, and softmax activation function is used in this case.

3.4. Deep LSTM Network

In Deep Neural Networks (DNN) with deeper architectures, deep LSTM RNNs have been successfully used for speech recognition (Mohamed, Dahl, & Hinton, 2012). Deep LSTM RNNs are built by stacking multiple LSTM layers. LSTM RNNs are already deep architectures. The depth in deep LSTM RNNs has an additional meaning, the input to the network at a given time step goes through multiple LSTM layers in addition to propagation through time and LSTM layers. Benefit of using Deep LSTM RNNs to normal LSTM, they can better use parameters by distributing them over the space through multiple layers. Deep LSTM model used in this paper consists of one embedding layer for input, three consecutive LSTM layers finally followed by two dense layers or output layer. Deep LSTM helps to remove the over fitting of the network. Deep LSTM are computationally more costly but present a very reasonable model for the task. A deep LSTM structure stacked with three LSTM layers is shown in Figure 2.

3.5. Bidirectional Deep LSTM Network

The principle of Bidirectional Deep LSTM (BDLSTM) is to split the neurons of a regular LSTM into two directions, one for positive time direction forward states, and another for negative time direction backward states, the output are not connected to inputs of the opposite direction states. By using two time directions, input information from the past and future of the current time frame can be used unlike standard LSTM which requires the delays for including future information. Our model here consisted of same Deep LSTM structure only difference was every LSTM cell was made bidirectional, so that the propagation of the signal might be in both the forward and the backward ways. A dropout of 0.2 is used in all the LSTM layers. The LSTM layers are followed by two dense layers having 'relu' and 'sigmoid' as the activation function (see Figure 3).

4. EXPERIMENTAL RESULTS ANALYSIS

The experiment was performed on 25000 IMDB movie reviews for training and the same number for testing purpose. By keeping all the parameters constant, the complexity of the model was increased and the changes in the result was noted. As we made the LSTM deep the over fitting of the network was removed, and the test set accuracy (in percentage) was increased as well. Detailed results are given in the following Table 1.

Figure 2. Deep LSTM Network

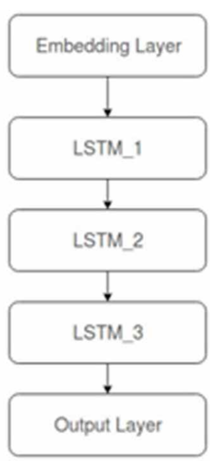


Figure 3. (a) LSTM Network; (b) Bidirectional LSTM Network

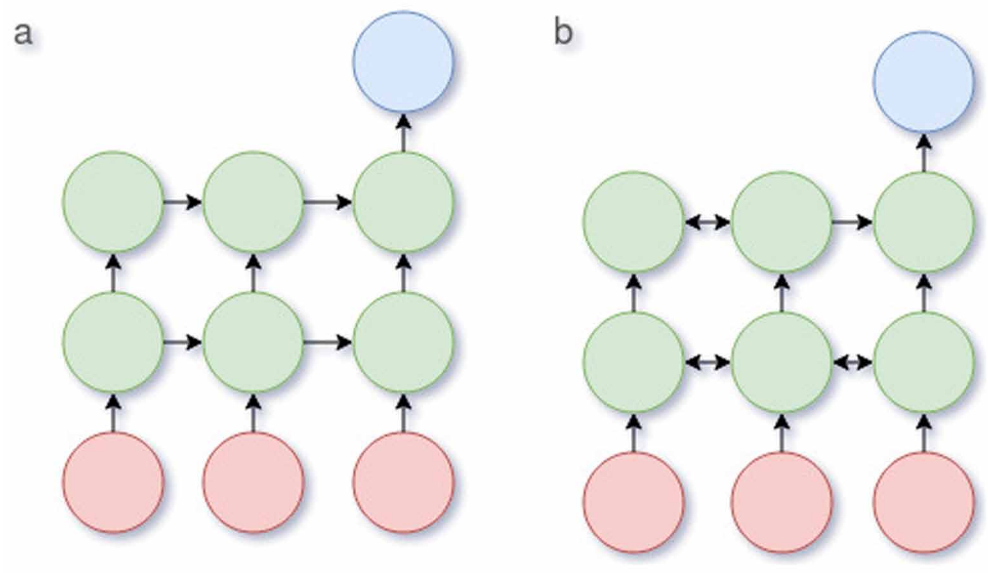


Table 1. Experimental Results Analysis

MODEL	Validation Accuracy (%)	Validation Loss
Conventional LSTM	80.92	1.1013
Deep LSTM	81.32	0.9295
Bidirectional Deep LSTM	83.83	0.8285

5. CONCLUSION

A model consisting of Bidirectional Deep LSTM is presented for sentiment analysis of movie review. With the increase of complexity of the network the validation set accuracy increased and there by validation loss dropped. But increase in complexity clearly also increases the computational cost of the network. Future works might be done on optimization of the Bidirectional Deep LSTM for better increase in performance of the system. Different variations on the model like changing the activation function, varying the number of stacked LSTM layers, varying the input and output dimensions of the network might also increase the accuracy of the system.

REFERENCES

Bengio, Y., Ducharme, R., Vincent, P., & Jauvin, C. (2003). A neural probabilistic language model. *JMLR*, *3*(Feb), 1137–1155.

Chollet, F. (2015). Keras. Gitub. Retrieved from https://github.com/fchollet/keras

Collobert, R., & Weston, J. (2008, July). A unified architecture for natural language processing: Deep neural networks with multitask learning. In *Proceedings of the 25th international conference on Machine learning* (pp. 160-167). ACM.

Gers, F. A., Schmidhuber, J., & Cummins, F. (2000). Learning to forget: Continual prediction with LSTM. *Neural Computation*, *12*(10), 2451–2471. doi:10.1162/089976600300015015 PMID:11032042

Gers, F. A., Schraudolph, N. N., & Schmidhuber, J. (2003). Learning precise timing with LSTM recurrent networks. *Journal of Machine Learning Research*, *3*, 115–143.

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, *9*(8), 1735–1780. doi:10.1162/neco.1997.9.8.1735 PMID:9377276

Kim, Y. (2014). Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)* (pp. 1746–1751). doi:10.3115/v1/D14-1181

Lee, J., Cho, K., & Hofmann, T. (2016). Fully character-level neural machine translation without explicit segmentation. arXiv:1610.03017

Mihalcea, R., Banea, C., & Wiebe, J. (2007). Learning multilingual subjective language via cross-lingual projections. In *Proceedings of the 45th annual meeting of the association of computational linguistics* (pp. 976-983).

Nogueira dos Santos, C., & Gatti, M. Deep Convolutional Neural Networks for Senti-ment Analysis of Short Texts. In *International Conference on Computational Linguistics* (pp. 69-78).

Pang, B., Lee, L., & Vaithyanathan, S. (2002). Thumbs up! sentiment classification using machine learning techniques. In *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing* (pp. 79–86).

Rahman Mohamed, A., Dahl, G. E., & Hinton, G. E. (2012). Acoustic modelling using deep belief networks. *IEEE Transactions on Audio, Speech, and Language Processing*, *20*(1), 14–22. doi:10.1109/TASL.2011.2109382

Sak, H., Senior, A., & Beaufays, F. (2014). Long short-term memory recurrent neural network architectures for large scale acoustic modeling. In *Fifteenth annual conference of the international speech communication association*.

Schütze, H., Manning, C. D., & Raghavan, P. (2008). *Introduction to information retrieval*. Cambridge University Press.

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 2011 Conference on Empirical Methods in Natural Language Processing* (pp. 151–161).

Socher, R., Perelygin, A., Wu, J., Chuang, J., Manning, C. D., Ng, A. Y., & Potts, C. (2013). Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing* (pp. 1631–1642).

Understanding LSTM Networks. (2015). Retrieved from http://colah.github.io/posts/2015-08-Understanding-LSTMs/

Xiong, C., Zhong, V., & Socher, R. (2016). Dynamic coattention networks for question answering. arXiv:1611.01604

Zhou, X., Wan, X., & Xiao, J. (2016). Attention-based LSTM Network for Cross-Lingual Sentiment Classification. In *Proceedings of EMNLP* (pp. 247–256). doi:10.18653/v1/D16-1024