Proceedings of the 3rd International Conference on Signal Processing and Machine Learning
DOI: 10.54254/2755-2721/4/2023385

Research on the evolution of neural networks models in NLP

Ziang Zhao

College of Engineering, University of Wisconsin – Madison, 1415 Engineering Drive, Madison, WI, United States of America, 53706

daniel.zhao@wisc.edu

Abstract. Since their first emergence, neural networks have produced significant results and have been an essential tool for solving natural language processing (NLP) problems. This paper aims to provide insight into the development of neural network models and their application in various NLP problems. The efficiency of the different approaches would be analyzed and insight into the different models would be provided. Overall, neural networks have been proven to be essential for solving problems related to NLP, and it is believed that further advancements in newer models would see a surge in performances of questions that were currently considered hard.

Keywords: Neural Network, Natural Language Processing, Algorithms, Machine Learning.

1. Introduction

Neural networks, being an important constitutional method that has been used for solving problems within the field of natural language processing (NLP), has evolved into several different models that each have been utilized for different problems. While the utilization of neural networks has been the central of solving NLP problems for extended periods, there is a lack of an integrated understanding of how the different models work for specific problems within the field. This paper is aiming to provide insight through the different utilization of the different models of neural network and tends to present an overview review of the applications of different neural network models within the field of natural language processing. This paper would start by briefly discussing the basics of neural networks and would continue to elaborate on the different kinds of neural networks, the theory behind the usage of these neural networks, and an analysis on some cases where these neural networks were being used in problems of NLP. The analysis would be based on a few studies on utilizing neural networks on NLP problems and is aiming for analyzing the model being used and the results from the training of these models.

2. Feed-forward neural network

The most basic form of the neural network is often considered to be the feed-forward neural network. The idea of this type of neural network was being first adopted from an analogy of the computational mechanisms of brains. In the analogy, individual neurons were being considered as computational units that would accept certain inputs as features and computes an output. The output is frequently thought to be determined by a non-linear function that takes its input from a weighted sum of the input values. Within a neural network, individual neurons are being connected to one another, where the output from a specific neuron would become the input for one or more additional neurons. When data

DOI: 10.54254/2755-2721/4/2023385

is being presented to this network, both during training and execution, the data would only passes through the layers in sequence from the input in the form of features until output, one layer at a time and would not revisit a previous layer. According to Goldberg [1], neural networks were considered special due to their non-linearity, which could often outperform traditional linear models.

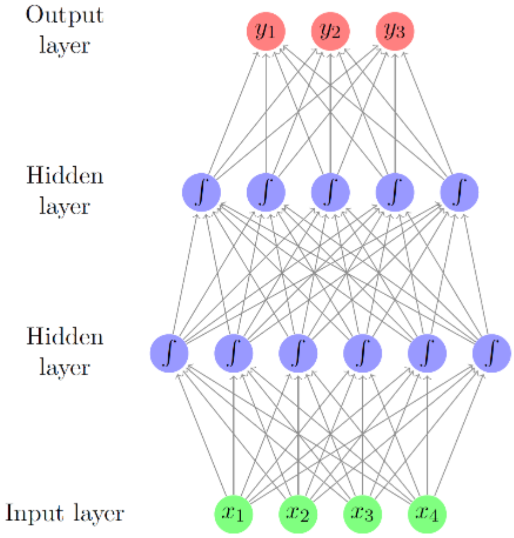

Figure 1. Representation of a two layer feed forward neural network.

Within the field of NLP, one problem that has benefited from neural networks has been parsing. One significant example of this could be parser construction through the work of Chen and Manning [2], where a parser has been constructed through a neural network and have achieved a relative higher accuracy comparing to existing parsers. The features that are being utilized for the training were a dense vector containing information of the word itself, their part-of-speech (POS) labels, and a series of dependency-based labels that were being derived from the arc-standard system, where the parse tree of the inputted sentence is being stored. The neural network is being designed to contain only a single layer that utilizes a cube activation function, which, after taking the weighted sum of the selected features, and then was taken as a cubed result. The output from this single layer of neurons were further being applied with a weighted sum and a SoftMax function to allow for multi-class probabilities. When the training of the neural model is completed, it is shown that the model would correctly shows the outcome at about a 90% accuracy rate for English database examples and over 82% accuracy for a Chinese database example. Furthermore, the model is proved to be much faster in terms of parsing speed than the popular parsers that were being selected as comparisons. The testing results have shown that the neural network model is able to parse at about 150% as fast as the popular parsers. Another similar result comes from the work of Lewis and Steedman [3], who built a Combinatory Categorial Grammar parser that uses discrete features for word and suffix embedding. The neural network that is being used utilized a tanh activation function for its only hidden layer and utilizes a similar SoftMax function for the output layer. The parser itself works comparably better than other models, which indicates that the neural network model is much more efficient and accurate than traditional models.

3. Convolutional neural network

Even though a simple feed-forward neural network sometimes may be enough for problem-solving of problems within the field of NLP, sometimes it is necessary for the sequence of the data that was being used for training to be important, such as in problems of sentiment analysis within NLP. While traditional feed-forward neural networks would help to solve these kinds of problems, they would require significantly longer time. This is when the idea of a convolutional neural network being come

DOI: 10.54254/2755-2721/4/2023385

to place. A convolutional neural network is like a traditional feed-forward neural network, however, not all input features are fully connected towards neurons. Instead, a convolutional neural network would utilize convolution layers to allow local features to be merged and produce a feature vector from the local input. Furthermore, a pooling layer is also often placed after the convolution, mainly being utilized to reduce the dimensions of the outputted features. Through the utilization of convolutional neural networks, the details of local features could be extracted and thus help to evaluate the training results better.

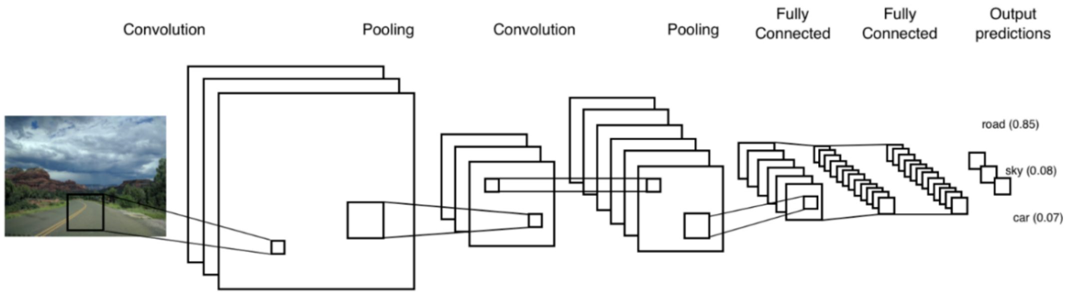

Figure 2. A convolution neural network that is used in image analysis.

While convolution neural networks were traditionally being utilized within the field of computer vision, its ability to extract local features could also be utilized for problems within NLP. An example would be the example that we talked about in the previous paragraph, i.e., the sentiment analysis. To be more specific, convolution neural network has been useful in solving the problem of sentence modeling, as shown by the work of Kalchbrenner et al. [5]. In their work, they purposed the utilization of a dynamic k-max pooling strategy, in which, instead of the traditional form of pooling where a fixed number of the most active features were being selected after the convolution of a local area, the number of features were selected during pooling have been dynamic and was related to the length of the input. Through utilizing these methods and accompanied by the convolution neural network, their trained model was able to achieve overall about a 87.4% accuracy when predicting the sentiment of tweet, which is shown to be better performing than any other of the models being provided as comparison. A similar result could be seen from the aspect-based multilingual sentiment analysis model completed by Ruder et al. [6]. The convolution neural network would directly take a padded input as its features and would apply convolution and pooling towards the text until the result is being processed by a SoftMax output activation function. After the training process, when being placed in different language settings, the model tends to perform well majority of the time, and would nearly always holds an accuracy of more than 70% despite the different languages, which was largely attributed to the language independence that occurred from the training of the models. Overall, the convolution neural network has produced relative greater results comparing to traditional neural network as a great improvement within the field.

4. Recurrent neural network

Despite the largely accuracy increase of the utilization of convolution neural networks, they might not always be the case where we only want to look at the most salient features of the input features within problems of NLP. In certain cases, we would want the sequence structure of the input to be preserved, which is where recurrent neural network would be a better choice. A recurrent neural network could be considered more like a traditional neural network in terms of the layers, except that they would also accept information from a previous sequential input, allowing the entire sequence to be processed while preserving the properties of the sequence. The information would cycle through the neural network, which has allowed the feature to be extracted more easily when processing sequential data [7].

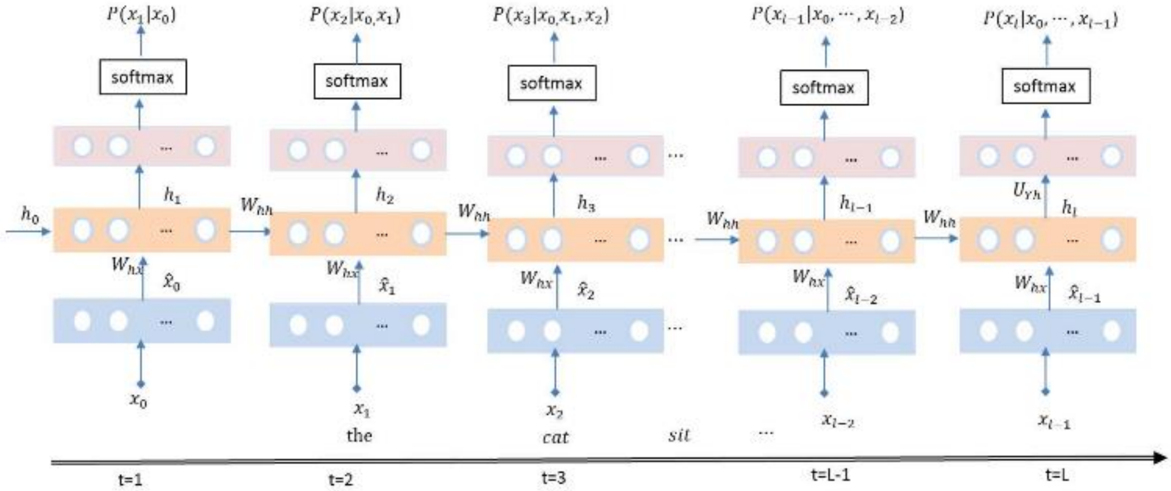

Figure 3. Image of representing a RNN model for NLP questions.

Since information would be cycled through the neural network and allows for the sequence to be understood as information [8], an example for utilizing the network model would be the example of the work by Dodal and Kulkarni [9], which was an information retrieval system for multiple languages. This system would take the input of a sentence and produce a ranked output of the number of items available. The system was developed utilizing a deep learning recurrent neural network, which usually contains multiple hidden layers. After the input words were being transformed into a vector representation of different features, the features were being extracted and matched against the information and would output the rank based on a calculated ranked score from the features.After the network has been trained, a test of 100 English queries have produced an accuracy of over 60% for the mapping to both languages (Hindi and Marathi). While this accuracy might not seem to be high, it is worth noting that the accuracy would be much higher when the input would be in a correct grammatical structure.

While a recurrent neural network might be enough for solving specific problems, it still has its own problems, one of the most prominent being the problem of long-term dependence. Since the data were being led and computed by the model through cycles, it is very likely that the data that was being placed earlier into the system would have less weight as the number of cycles of the data increases, which is going to be a problem as an old output might not be correctly identified by the model. Fortunately, a recurrent neural network variant known as Long Short Term Memory (LSTM) was attempting to solve the problem [10-12]. LSTM has a relative similar structure of the network as a recurrent neural network, but there has been an additional concept of a “forget gate”, where it would gradually learned to forget specific information by only filtering specific information from the previous output into the next cycle, namely to reduce the problem of long-term dependency.

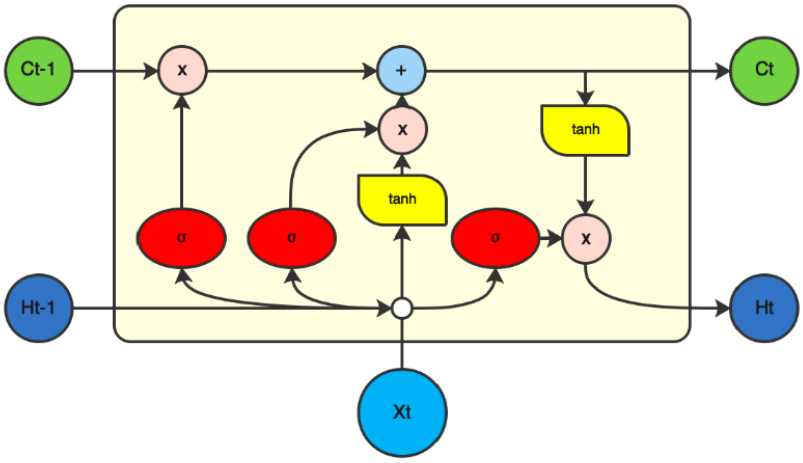

Figure 4. Image Representation of the LSTM Model.

A good example of recurrent neural network and LSTM being put into usage within the field of NLP would be of the part-of-speech (POS) tagging, according to the model built by Bahcevan et al. [13]. The POS tagging task was designed for the language Turkish, and it utilized both the recurrent neural network and LSTM briefly discussed above. After the words were being placed into the models separately, both models have produced promising accuracy on the tagging process, with both the neural network and the LSTM model achieving an accuracy nearly to be 80%, which were quite promising as the data that was being used to train was not large, but they were still far more efficient comparing to traditional models that utilizes hand-crafted features or linguistic rules.

5. Conclusion

Neural networks have been powerful tools that could be utilized within problems of NLP, providing efficient solutions towards problems within the field. As the neural network evolves from its most straight-forward feed-forward neural network, towards a slightly complicated form of convolution neural network, and finally to a recursive neural network, more and more complex problems within NLP could be solved through the utilization of these networks, ranging from parser design towards sentiment analysis and finally POS tagging. Even though this paper might act as an overview of the development of NLP, only selected examples were being discussed and analyzed, and it would be important to notice that the changes within models would not be a chronological development, rather than just different tools for different problems. As more and more advanced neural network models were being purposed, it is safe to assume that there would be more problems related to NLP that would see a great improvement in its performance.

Acknowledgment

This is to acknowledge and thank all of the organizations and individuals helped me for completing this research. I would like to thank University of Wisconsin – Madison for providing me with the resources available to complete the research, which allows me to collect the cases for writing this paper. I would also like to thank Professor Vipul Goyal from Carnegie Mellon University for giving me some insight on the development of neural networks and suggested the idea of tracing this development within a specific field of artificial intelligence, which leads to the completion of this paper.

References

[1] Y. Goldberg, ‘Neural network methods for natural language processing’, Synthesis lectures on human language technologies, vol. 10, Issue. 1, pp. 1–309, 2017.
[2] D. Chen and C. D. Manning, “A fast and accurate dependency parser using neural networks,” in

DOI: 10.54254/2755-2721/4/2023385

Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP), 2014, pp. 740–750.
[3] M. Lewis and M. Steedman, “Improved CCG parsing with semi-supervised supertagging,” Transactions of the Association for Computational Linguistics, vol. 2, pp. 327–338, 2014.
[4] M. M. Lopez and J. Kalita, “Deep Learning applied to NLP,” arXiv preprint arXiv:1703.03091, 2017.
[5] N. Kalchbrenner, E. Grefenstette, and P. Blunsom, “A convolutional neural network for modelling sentences,” arXiv preprint arXiv:1404.2188, 2014.
[6] S. Ruder, P. Ghaffari, and J. G. Breslin, “Insight-1 at semeval-2016 task 5: Deep learning for multilingual aspect-based sentiment analysis,” arXiv preprint arXiv:1609.02748, 2016.
[7] X. Peng, "A Comparative Study of Neural Network for Text Classification," 2020 IEEE Conference on Telecommunications, Optics and Computer Science (TOCS), 2020, pp. 214-218, doi: 10.1109/TOCS50858.2020.9339702.
[8] J. Xiao and Z. Zhou, "Research Progress of RNN Language Model," 2020 IEEE International Conference on Artificial Intelligence and Computer Applications (ICAICA), 2020, pp. 1285-1288, doi: 10.1109/ICAICA50127.2020.9182390.
[9] S. S. Dodal and P. V. Kulkarni, "Multi-Lingual Information Retrieval Using Deep Learning," 2018 9th International Conference on Computing, Communication and Networking Technologies (ICCCNT), 2018, pp. 1-6, doi: 10.1109/ICCCNT.2018.8493789.
[10] Staudemeyer, R.C. and E.R. Morris, Understanding LSTM--a tutorial into Long Short-Term Memory Recurrent Neural Networks. arXiv preprint arXiv:1909.09586, 2019.
[11] Sak, H., A.W. Senior, and F. Beaufays, Long short-term memory recurrent neural network architectures for large scale acoustic modeling. 2014.
[12] Hochreiter, S. and J. Schmidhuber, Long short-term memory. Neural computation, 1997. 9(8): p. 1735-1780.
[13] C. A. Bahcevan, E. Kutlu and T. Yildiz, "Deep Neural Network Architecture for Part-of-Speech Tagging for Turkish Language," 2018 3rd International Conference on Computer Science and Engineering (UBMK), 2018, pp. 235-238, doi: 10.1109/UBMK.2018.8566272.