Review of LSTM in TensorFlow for NLP tasks

Abstract

In this review, a comprehensive review of the LSTM method in TensorFlow is presented. The innovative part of this paper is that not only the theory basics and implementation of LSTM is discussed, the performance of the LSTM on NLP tasks in TensorFlow are compared with both other techniques and other deep learning frameworks, such as Pytorch. This will provide meaningful information to NLP researchers and users.

1. Introduction

Natural language processing (NLP), which deals with any task related to natural language, is one of the most important fields for machine learning and artificial intelligence. NLP tasks such as sentiment analysis, machine translation, language modeling, text classification, text summarization, text generation, speech reorganization all fall under the scope of NLP. a series of NLP tasks.

TensorFlow is an open source software library for high performance computation. TensorFlow, as a popular deep learning framework specified in Neural network method , and is a powerful tool to deal with NLP tasks. In this review, we will focus on the Recurrent neural network (RNN), specifically the Long-Short Term Memory (LSTM). First, the reason to use LSTM in NLP is discussed in section 2.1 ; the implementation of LSTM is briefly introduced in section 2.2 ; the performance of LSTM with TensorFlow is then compared with other techniques and other deep learning framework in section 2.3.

2. LSTM

2.1 Why LSTM?

The tradition structure of the neural networks usually consists of the input layer, the hidden layer and the output layer. In such setup, the former input is usually not connected to the later input. For the task of NLP, for example, the understanding of each word is not adequate to understand the meaning of the whole sentence. In the meantime, the sequence of each word may also affect the meaning of a sentence. For example, “Rocket is faster than planes”, and however if we rearrange the order, the meaning will be completely different.

In RNN, the output of an output layer can be connected to the next input layer, so it can be trained together. Thus, it becomes an effective method to process sequenced data, such as text data. However, the traditional RNN is with short-term memory and is not capable of processing long input.

LSTM stands for Long-Short Term Memory and it's a type of recurrent neutral network, which is proposed by Hochreiter and Schmidhuber. As with other neural network methods, it has multiple hidden layers, but the relevant information will be kept as it passes through every layer and the irrelevant information will be discarded. So, it's an effective tool in memorizing the important and relevant information and performs well I processing long sequenced text data.

LSTM has three main gates:

1. Forget gate: This gate will decide if the information is relevant or discarded.
2. Input gate: This gate helps to identify the important information and stores the data after the forget gate discarded the irrelevant information.
3. Cell state: The gained information will be used to update the new cell state:
4. Output gate: This gate decides what the next hidden state should be and what information the hidden state should carry.

2.2 LSTM implementation in TensorFlow

TensorFlow has implemented built in RNN functions, which make it easy to construct a LSTM layer. We will take the text classification problem of a movie data set as an example to illustrate the steps:

1. In TensorFlow, create a token dictionary and tokenize the words using the Tokenizer function.
2. Apply padding to the token-represented sequences for consistent length.
3. Create the embedding matrix and set up the embedding layer.
4. Call the Bidirectional function to run the LSTM model.

2.3 Performance of LSTM

In this section, the performance of LSTM in NLP task will be discussed.

2.3.1 Comparison with other methodology

As mentioned above, LSTM has better performance in processing long sequence text data. Malik and Kumar (2018) performed the sentiment analysis of twitter data using both LSTM method and the Naïve Bayes method. It is found that LSTM achieve a higher accuracy, precision and recall. In another research implemented by Zhang (2020), LSTM is shown to have a higher accuracy than traditional RNN model and is more suitable for long sequence text analysis.

2.3.2 Comparison with other deep learning frameworks

In this section, we will review the performance of LSTM method in TensorFlow comparing with other deep leaning frameworks.

Elshwi.et.al (2021) implemented a comprehensive experimental evaluation of deep learning frameworks. It is shown that Keras, Pytorch, TensorFlow have achieves similar accuracy using LSTM on IMDB reviews. TensorFlow has shown the fastest training time in two of three training data sets. In another research (Braun, 2018), it is shown that the training speed is similar across three frameworks (TensorFlow, PyTorch and Keras) with well-optimized LSTM implementations. With less-optimized and more customizable LSTM implementations, then PyTorch is about 1.5x faster than TensorFlow.

3. Conclusion

In summary, this review systematically introduced the LSTM implementation and its performance in TensorFlow to deal with NLP tasks. From the comparison with other methodology, it is shown LSTM with TensorFlow achieves a higher accuracy. Comparing other popular deep learning frameworks, LSTM implementation generally performs well in both accuracy and training speed, but the actual results may depend on the configuration and optimization.

4. References

Malik, V. and Kumar, A., 2018. Analysis of Twitter Data Using Deep Learning Approach: LSTM. International Journal on Recent and Innovation Trends in Computing and Communication, 6(4), pp.144-149.

Zhang, Z., 2020, March. Sentiment Analysis of Chinese Commodity Reviews Based on Deep Learning. In International Conference on Modern Educational Technology and Innovation and Entrepreneurship (ICMETIE 2020) (pp. 22-28). Atlantis Press.

Elshawi, R., Wahab, A., Barnawi, A. and Sakr, S., 2021. DLBench: a comprehensive experimental evaluation of deep learning frameworks. Cluster Computing, pp.1-22.

Braun, S., 2018. LSTM benchmarks for deep learning frameworks. arXiv preprint arXiv:1806.01818.