LSTM and GRU neural network performance comparison study

taking Yelp review dataset as an example

Shudong Yang*
School of Information and Business Management
Dalian Neusoft University of Information
Dalian, China
* Corresponding author: yangshudong@neusoft.edu.cn

Xueying Yu
School of Information and Business Management
Dalian Neusoft University of Information
Dalian, China

Ying Zhou
School of Information and Business Management
Dalian Neusoft University of Information
Dalian, China

***Abstract*—Long short-term memory networks(LSTM) and gate recurrent unit networks(GRU) are two popular variants of recurrent neural networks(RNN) with long-term memory. This study compares the performance differences of these two deep learning models, involving two dimensions: dataset size for training, long/short text, and quantitative evaluation on five indicators including running speed, accuracy, recall, F1 value, and AUC. The corpus uses the datasets officially released by Yelp Inc.. In terms of model training speed, GRU is 29.29% faster than LSTM for processing the same dataset; and in terms of performance, GRU performance will surpass LSTM in the scenario of long text and small dataset, and inferior to LSTM in other scenarios. Considering the two dimensions of both performance and computing power cost, the performance-cost ratio of GRU is higher than that of LSTM, which is 23.45%, 27.69%, and 26.95% higher in accuracy ratio, recall ratio, and F1 ratio respectively.**

Keywords-deep learning; long short-term memory (LSTM); gate recurrent unit (GRU); performance evaluation; natural language processing (NLP)

I. Introduction

The difference between recurrent neural networks(RNN) and ordinary artificial neural networks(ANN) is that RNN can process sequence data well. There are several words in a sentence, and a certain word has a certain correlation with the words before and after it, which constitutes sequence data. So RNN can be used to deal with natural language with context[1]. The role of the gradient is to update the weight value of the neural network. If the weight is too small, the gradient will disappear, that is, the hidden layer near the input layer will not continue to learn. On the other hand, if the weight is too large, it will cause the gradient to explode[2]. Affected by this, RNN is sensitive to time step, which means that RNN does not have long-term memory, and will be affected by short-term memory. To solve this problem, long short-term memory networks(LSTM)[3] and gate recurrent unit networks(GRU)[4] were born.

For natural language processing of online reviews, which one is better, LSTM or GRU? What are the usage conditions for LSTM and GRU applications? At present, the relevant research in academia is limited. In order to explore these issues, the following research has been carried out. Chapter II introduces the basic principles, data sets, data preprocessing, model construction and training of LSTM and GRU; Chapter III introduces the experimental environment, experimental results, and experimental data analysis; The last chapter summarizes the comparison between LSTM and GRU deep learning models, and the research prospects.

II. Model building and training

A. Basic principles

LSTM was first introduced by Sepp Hochreiter and Jurgen Schmidhuber in 1997, and has been widely used until now, and many variants have been derived. Compared with ordinary RNN, LSTM adds input gate and forget gate to solve the problem of gradient disappearance and gradient explosion, so that long-term information can be captured, and it can have better performance in long sequence text. The input and output structure of GRU is similar to ordinary RNN, but its internal structure is similar to LSTM[5]. The internal structure comparison of LSTM and GRU is shown in Figure 1 and Figure 2 below.

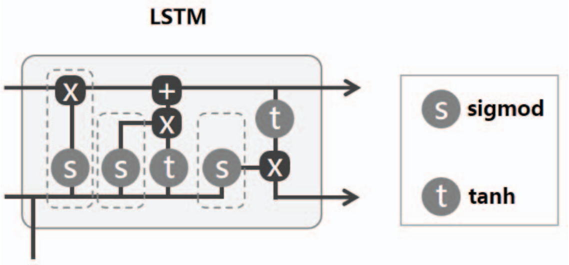

Figure 1. Internal structure of LSTM

978-1-7281-8149-3/20/$31.00 ©2020 IEEE
DOI 10.1109/IWECAI50956.2020.00027

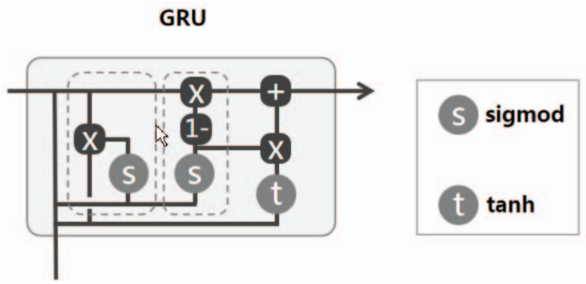

Figure 2. Internal structure of GRU

Neither LSTM nor GRU can encode information from back to front. In scenes with finer classification granularity, such as the five-category tasks of strong commendatory term, weak commendatory term, neutrality, weak derogatory term, and strong derogatory term, attention should be paid to the interaction between degree words, emotion words, and negative words. Bi-directional long short-term memory(Bi-LSTM) and Bi-directional gate recurrent unit(Bi-GRU) solves this problem. They are formed by stacking forward and backward LSTM or GRU to better capture bidirectional semantic dependence. Bi-LSTM or Bi-GRU is usually better than LSTM or GRU, but the training will be more time-consuming.

B. Dataset

Yelp is a global online review network, including restaurants, shopping malls, hotels, and tourism businesses around the world. Users can rate merchants, write reviews, and exchange service experiences on it[6][7]. The Yelp open source dataset is the official dataset for scientific research and machine learning competitions produced by Yelp Inc.. The latest version of the dataset in 2020 contains 8.12 million reviews of 1.96 million users, and the review objects cover 210,000 merchants worldwide.

C. Data preprocessing

The sub-datasets of pictures, merchants, users, etc. in the original dataset are not directly related to this study, so only the review sub-dataset is retained, that contains fields such as review_id, user_id, business_id, stars, date, text, useful, funny, cool, etc. Slicing extracts 10,000 comment corpora and converts JSON format to CSV format.

In the pre-processing stage, non-English comments are filtered through scripts, and then stars, useful, funny, and cool are cross-validated to remove abnormal comments. The emotional polarity of comments is automatically marked through the script. The rule is that stars greater than 3 are considered positive, otherwise they are considered negative. After that, a corpus with more than 200 characters is regarded as a long text, otherwise a short text.

Finally, four datasets were sorted out, namely long text & small data set, long text & large dataset, short text & small dataset, and short text & large dataset.

D. Deep learning model architecture

In order to reduce interference factors as much as possible, the LSTM deep learning model and GRU deep learning model used in this research share the same architecture, as shown in Figure 3 below, where the "X" module can be configured as LSTM or GRU. Firstly, word embedding is processed for the pre-processed corpus. Secondly, the first layer of the neural network is composed of a combination of forward and backward LSTM or GRU, which can better capture bidirectional semantic dependence[8]. In order to reduce overfitting and improve model generalization ability, a dropout layer[9] is added after the first layer of the neural network. Then there is a layer of neural network, which can be configured as LSTM or GRU, because more neural network layers can improve the abstract processing ability[10]. The activation function used Rectified Linear Unit (ReLU), and finally the final output is obtained through the softmax function.

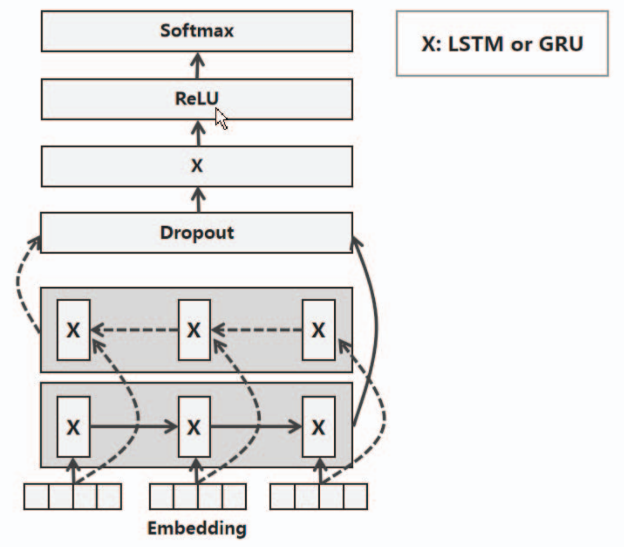

Figure 3. Model architecture with replaceable module "X"

III. EXPERIMENTAL DATA ANALYSIS

A. Experimental environment

The operating system used for the experiment is Windows 7 64Bit Professional Edition; the integrated development environment (IDE) used PyCharm 2019.1 Community Edition; and the development language is Python version 3.7.

This study is a binary classification, so in the model compilation stage, the binary cross entropy is selected for the loss function. The optimizer chose Adam, which is suitable for large-scale corpus scenarios[11]. Hyperparameter optimization cannot be optimized by gradient descent like ordinary parameters, and it is a combinatorial optimization problem.

In view of the small number of model layers, the grid search method was used to adjust the parameters. According to the different combinations of these hyperparameters, the LSTM model and the GRU model of the same architecture are trained respectively, and then the performance of these models is tested, and a set of the best performance configuration is selected as the hyperparameters of both models.

B. *Experimental results*

A total of four datasets of long text & small dataset, long text & large dataset, short text & small dataset, and short text & large dataset were trained with LSTM and GRU models respectively. The experimental results are shown in Table 1 below:

TABLE I. EXPERIMENTAL DATA SUMMARY

Patterns	Simulation [a]	Accuracy	Recall	F1	AUC
GRU_long_2K	129.41	0.6897	0.8513	0.8451	0.6605
GRU_long_700	128.13	0.7699	0.7744	0.7747	0.7714
GRU_short_2K	130.02	0.6763	0.8462	0.8288	0.6108
GRU_short_700	129.11	0.7761	0.7795	0.7803	0.7802
LSTM_long_2K	166.32	0.7500	0.8769	0.8787	0.7646
LSTM_long_700	171.78	0.7596	0.7538	0.7548	0.7635
LSTM_short_2K	165.07	0.7492	0.8718	0.8643	0.6970
LSTM_short_700	164.83	0.7909	0.7897	0.7907	0.7965

a. Simulation time (second)

From the perspective of model training speed, the average running time of the LSTM model is 167.00 seconds. The time of the GRU model is 129.17 seconds. So the training speed of the GRU model is 29.29% faster than that of the LSTM. Analyzing the reason from the model principle, GRU can forget and choose memory with one gating, and there are fewer parameters, while LSTM needs to use more gating and more parameters to complete the same task.

Ignoring the influencing factors of the length of the corpus, in the scenario of large datasets, the average performance of GRU on the four indicators of accuracy rate, recall rate, F1 value, and AUC is 91.12%, 97.07%, 96.04%, and 86.98% of LSTM respectively. On the other hand, in the scenario of small datasets, the average performance of GRU on the four indicators of accuracy, recall, F1 value, and AUC is 99.71%, 100.67%, 100.61%, and 99.46% of LSTM respectively. Therefore, GRU is more suitable for smaller datasets than LSTM.

Ignoring the influencing factors of the dataset size, in the long text scenario, the average performance of the GRU on the four indicators of accuracy rate, recall rate, F1 value, and AUC is 96.69%, 99.69%, 99.16%, 93.70% of LSTM, respectively. In the short text scenario, the average performance of GRU on the four indicators of accuracy rate, recall rate, F1 value, and AUC is 94.31%, 97.85%, 97.23%, and 93.14% of LSTM, respectively. Therefore, in long text processing, the performance of GRU is close to LSTM, but in short text processing, GRU is not as good as LSTM, especially in terms of accuracy.

Considering the two influencing factors of dataset size and text length, there are a total of 4 combinations. The performance of GRU and LSTM is shown in Table 2 below. In the scenario of long text and small data set, GRU performance will exceed LSTM, but in other three scenarios it is not as good as LSTM.

TABLE II. GRU TO LSTM PERFORMANCE RATIO

Patterns	Accuracy	Recall	F1	AUC
long_2K	0.9196	0.9708	0.9618	0.8639
short_2K	0.9027	0.9706	0.9589	0.8763
long_700	1.0136	1.0273	1.0264	1.0103
short_700	0.9813	0.9871	0.9868	0.9795

Considering the two dimensions of performance and computing power cost, the performance-cost ratio of GRU is higher than that of LSTM, which is 23.45%, 27.69%, and 26.95% higher in accuracy ratio, recall ratio, and F1 ratio, respectively.

The resources used in the above experiment have been shared on Github at the URL: <https://github.com/g9g99g9g/basic/tree/master/ANN/LSTM_vs_GRU>, which contains a condensed version of the datasets, the source code of data pre-processing, the neural network models and all experimental logs.

IV. CONCLUSION AND RESEARCH PROSPECT

RNN is suitable for processing sequence data for prediction in many research fields, but it is restricted by short-term memory. LSTM and GRU use a gate structure to overcome the impact of short-term memory. The gate structure can regulate the information flow through the sequence chain. They are widely used in speech recognition, speech synthesis and natural language processing.

Traditional research believes that both LSTM and GRU can retain important features through various gates, ensuring that important special features will not be lost during long-term transmission. The structure of GRU is simpler. It has one gate less than LSTM, which reduces matrix multiplication, and GRU can save a lot of time without sacrificing performance.

However, through empirical research, this advantage of GRU only holds in the scenario of long text and small datasets. In other scenarios, compared with LSTM, the performance loss of GRU is more serious. In an age when computing power is no longer a bottleneck, LSTM is actually more suitable in these scenarios.

In the future, whether LSTM or GRU, there will be more and more variants, or even alternatives, to improve their performance and speed at the same time.

ACKNOWLEDGMENT

Firstly, I would like to thank the editors and reviewers gave me lots of helps. Secondly, thanks to my newborn baby who gave me lots of inspiration. Thirdly, I would like to thank my friends and colleagues for their supports.

Last but not the least, I would like to thank my family: my parents, my wife and guide-dog LaBa for supporting me spiritually throughout writing this paper and my life in general.

Thanks for all your encouragement and help.

REFERENCES

[1] Çöltekin Ç, Rama T. Tübingen-oslo at SemEval-2018 task 2: SVMs perform better than RNNs in emoji prediction[C]//Proceedings of The 12th International Workshop on Semantic Evaluation. 2018: 34-38.

[2] Pascanu R , Mikolov T , Bengio Y . On the difficulty of training Recurrent Neural Networks[J]. 2012.

[3] Gers F A . Learning to forget: continual prediction with LSTM[C]// 9th International Conference on Artificial Neural Networks: ICANN '99. IET, 1999.

[4] Wang N, Wang J, Zhang X. YNU-HPCC at SemEval-2018 Task 2: Multi-ensemble Bi-GRU Model with Attention Mechanism for Multilingual Emoji Prediction[C]//Proceedings of The 12th International Workshop on Semantic Evaluation. 2018: 459-465.

[5] Hettiarachchi H, Ranasinghe T. Emoji powered capsule network to detect type and target of offensive posts in social media[C]//Proceedings of RANLP. 2019.

[6] Asghar N . Yelp Dataset Challenge: Review Rating Prediction[J]. 2016.

[7] Cervellini P , Menezes A G , Mago V K . Finding Trendsetters on Yelp Dataset[C]// IEEE Symposium Series on Computational Intelligence 2016. IEEE, 2016.

[8] Coman A C, Zara G, Nechaev Y, et al. Exploiting deep neural networks for tweet-based emoji prediction[C]//International Workshop on Semantic Evaluation. 2018, 4: 1.

[9] Srivastava N, Hinton G E, Krizhevsky A, et al. Dropout: a simple way to prevent neural networks from overfitting[J]. Journal of Machine Learning Research, 2014, 15(1): 1929-1958.

[10] Svozil D, Kvasnicka V, Pospichal J, et al. Introduction to multi-layer feed-forward neural networks[J]. Chemometrics and Intelligent Laboratory Systems, 1997, 39(1): 43-62.

[11] Kingma D P, Ba J. Adam: A Method for Stochastic Optimization[J]. arXiv: Learning, 2014.