Research Progress of RNN Language Model

Jianqiong Xiao*
Educational Information Technical Center
China West Normal University
Nanchong, Sichuan, China

Zhiyong Zhou
Educational Information Technical Center
China West Normal University
Nanchong, Sichuan, China

***Abstract*—Language model is the cornerstone of natural language processing (NLP). The language model based on recurrent neural network (RNN) has become the most widely used language modeling technology because of its strong ability of self-learning and analysis of sequence data. In this paper, RNN language model is combed in detail. Firstly, the development history and application of RNN language model are briefly summarized. Then, the general mathematical description form of language model, the basic structure and calculation process of classical RNN language model are introduced. Then, the research progress of RNN language model in computational complexity and model performance improvement is mainly introduced. Finally, the future research of RNN language model is discussed.**

Keywords—language mode, recurrent neural network (RNN), research progress

I. INTRODUCTION

In recent years, natural language processing (NLP) has made great progress. It has been widely used in machine translation, intelligent Q & A, information retrieval and other applications. As the cornerstone of NLP, language model has also been widely concerned and deeply studied by scholars. Among all kinds of language models, RNN and its variants have become the most widely used language modeling technology because of their powerful ability of processing sequence data and self-learning.

II. DEVELOPMENT HISTORY

Language model has been put forward since 1980s, and it has gone through three stages: expert grammar rule model, statistical language model and neural network language model.

Since the 1990s, Frederick Jelinek, the pioneer of NLP research, and his team put forward the statistical language model and applied it to the speech recognition system, which set off the first climax of language model research and application to practical tasks. However, the application of this language model gradually reveals its fatal defects, which are dimension disaster and sparsity.

In order to overcome the dimension disaster and sparsity of statistical language model, after years of research by bengio and others, in 2003, they proposed a language model based on neural network feedforward neural network model (FFLM) [1]. However, the generalization performance of FFLM model is worrying when it encounters sequence data problems.

In 2010, Tomas mikolv proposed to apply the cyclic neural network model to language modeling (RNNLM) [2], and applied it to various NLP tasks such as speech recognition, machine translation, etc., and achieved the best results at that time, showing the outstanding advantages of the recurrent neural network language model, that is, this language model can capture the characteristics of long sequence of text data, with good generalization ability. Since then, the study of language model has entered a new era. Since then, with the emergence of multiple variants of the cyclic neural network model structure, the recurrent neural network language model, such as springing up, has been continuously improved, and further promote the vigorous development of various NLP applications.

III. MODEL INTRODUCTION

A. Problem Definition

The so-called language model (LM) is to estimate the probability p (x) of a piece of text X by a certain calculation method, and then judge whether the statements appear reasonable or not by the size of the probability, that is, whether they conform to people's grammar habits.

For the probability p (x) of language model, we usually express it in mathematical language. Suppose a piece of text x, which consists of L words, followed by the sequence $< w_1, w_2, \cdots, w_L >$, in order to facilitate visualization, the word W is usually directly marked with text x, that is, the word sequence of text x is recorded as $< x_1, x_2, \cdots, \mathrm{x}_L >$, where $x_i (1 \leq x_i \ll x_L)$ represents the number of the ith word in the word sequence in the dictionary. Then the probability $\mathrm{p}(\mathrm{x}_1, \mathrm{x}_2, \cdots, \mathrm{x}_{\mathrm{l}-1}, \mathrm{x}_L)$ of the whole word sequence of text x is expressed as follows:

$$\mathrm{P}(x_1, x_2, \cdots, x_{l-1}, x_L) = \prod_{i=1}^{L} P(x_l | x_0, x_1, \cdots, x_{l-1}) \qquad (1)$$

Where, x_0 is a placeholder added to the head of the word sequence to mark the beginning of the sequence. x_{L} is the target word, and $(x_0, x_1, \cdots, x_{l-1})$ is the above semantic information of the target word. The performance of the language model is to distinguish according to the probability value calculated by the language model. If it can well distinguish whether the word sequence conforms to the natural language, the language model is what we need.

B. RNN Language Model Architecture

The architecture of RNN language model usually includes three layers: input layer, hidden layer and output layer. Its structure model is shown in Figure 1, which is a cyclic neural network language model expanded by time. In the RNN language model, the probability of each word appearing under the given information semantic condition can be calculated iteratively by RNN in the way of time expansion, and the conditional probability of one word in the

978-1-7281-7005-3/20/$31.00 ©2020 IEEE

Dalian, China
June 27-29, 2020

word sequence can be calculated in each time step. The calculation method of each layer is described in detail below.

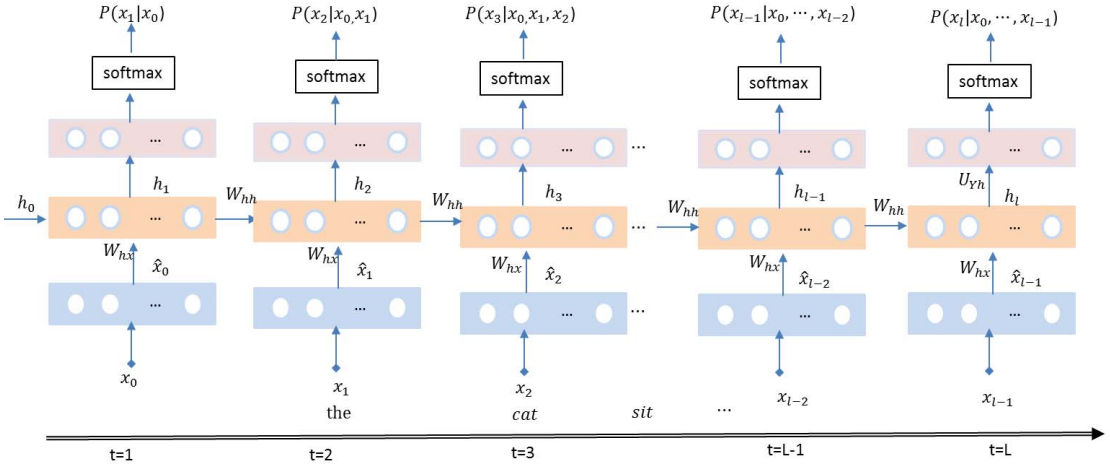

Fig. 1: RNN Language model

1) Input layer

The function of RNN input layer is to read the word sequence of text X in sequence $< x_1, x_2, \cdots, x_L >$ and then map it to a continuous word vector $< \hat{x}_1, \hat{x}_2, \cdots, \hat{x}_L >$ as the input of the current time. In Figure 1, $\hat{x}_{l-1}$) is the vector after the mapping of the word x_{l-1} in the word sequence.

2) Hidden layer

At time t, the vector $\hat{x}_{l-1}$ is the word sequence vector mapped by the input layer, and h_{t-1} is the output of the hidden layer at the previous time step. The output of the hidden layer at the current time is calculated by using the following formula (2).

$$h_t=\sigma(W_{hx}\hat{x}_{l-1}+W_{hh}h_{t-1}+b_h), 1\le t\le L \quad (2)$$

Among them, $W_{hx} \in R^{d_h \times d_w}$ is the weight matrix from the input layer to the hidden layer, and $W_{hh} \in R^{d_h \times d_h}$ is the weight matrix between the current hidden layer and the previous hidden layer. d_h is the number of neurons in the hidden layer, and b_h is the bias vector of the hidden layer. "σ" represents the nonlinear activation function of each element. This method saves the memory features of the above information, and then updates the output of the hidden layer together with the new input. In this way, the sequence can be connected from x_1 to x_L to realize the representation of the above information.

3) Output layer

The output layer takes the expression of the hidden layer at the current time as the input, and then estimates the probability distribution of all words in the vocabulary through softmax. The calculation process is as follows:

$$y_t = softmax(U_{yh}h_t + b_y) \in R^{|V|} \quad (3)$$

$$P(x_l|x_0, x_1, \cdots, x_{l-1}) = y_t(i) \quad (4)$$

Each element in y_t represents the probability that a word in a dictionary may appear in a given context. If the number of the target word in the dictionary is i, that is, $x_i = i$, then the probability that the target word appears in a given context is the ith element of the vector y_t (i). U_{yh} is the weight matrix between the hidden layer and the output layer.

C. Language model evaluation

At present, in the field of NLP, the measurement of a language model is usually based on the measure of perplexity (PPL). Its basic principle is that if the sentence of the test set can be given a higher probability value, it means that the language model is better. Assuming that the test set s contains N word sequences, the confusion PPL on the test set is calculated by formula (5):

$$PPL(S) = 2^{-\frac{1}{K}\sum_{n=1}^{N}\sum_{l=1}^{L_n}\log_2 P(x_l)} \quad (5)$$

Among them, S is the sentence in the test set, L_n is the number of words contained in the sentence S, K is the total number of dictionaries in the test set, and $P(x_l)$ is the probability of the first word. The larger the $P(x_l)$, the smaller the PPL, which means the greater the probability of the sentence, that is, the better the language model.

There are many factors that can affect the fluctuation of the size of the Perplexity. For example, the training data set size has the greatest impact on the PPL. If the training data set is larger, the PPL will be lower. Secondly, all kinds of punctuation in the data will also have a great impact on the PPL of the language model. In addition, the influence of stop words on PPL can not be ignored, but some stop words do not have no effect in some sentences. Therefore, when we evaluate the language model, we need to analyze the specific problems specifically, and we can make accurate judgment by using perplexity and other evaluation methods.

IV. Research Progress of RNN Language Model

Since 2010, Tomas Mikolv formally proposed the RNN language model, which has been favored by many researchers and started in-depth research. These researches are mainly improved from two aspects: one is the complexity of computation, the other is the improvement of network structure.

A. Improve RNN Computing Complexity

The size of training data set, punctuation marks and stop words all have influence on the ppl of RNN language model. However, in the early RNN language models, only some high-frequency words can be used for modeling. In addition, unknown (UNK) words and stop words have an impact on

computational complexity. Therefore, in view of the above problems, researchers have done a lot of work.

Jean s et al. Use the importance sampling technology [3], first analyze the importance of text features, screen out the corresponding features, and then input the model for training, which reduces the complexity of language model calculation to a certain extent.

Mnih A et al. Proposed the noise contrast estimation (NCE) method to approximate the output layer of softmax [4], which can also alleviate the computational complexity to some extent. However, these methods still have not solved the problems such as UNkown (UNK) words.

In 2015, vVinyals et al. Proposed a "pointer network" [5], which first infers a reasonable known word from the context through a method, and then copies it to replace unk. They applied this method to machine translation and text summarization and achieved good results. However, this method is not suitable for the passage.

B. The Improvement of Network Structure

In order to further improve the performance of RNN language model, more researchers try to start with the improvement of network structure or introduce other technologies.

1) Changes in network structure

It has been proved that when the basic recurrent neural network is used, with the deepening of training depth, its gradient will die out or explode, so it is very difficult to learn the expression of long-distance sequence. In order to overcome the problem of gradient extinction or explosion of basic recurrent neural network, various improved recurrent neural network models are proposed. The most praiseworthy one is the long short term memory (LSTM) recurrent neural network [6] proposed by Hochreiter et al. In 1997. After that, on the basis of LSTM, various versions of network structures such as GRU have been proposed one after another.

In 2012, in the research of Sundermeyer et al., LSTM neural network was introduced into language modeling, and the language model of LSTM neural network was constructed [7]. The improvement of its performance shocked scholars.

The network structure of language modeling is not only changed from basic RNN to LSTM, but also from one-way network structure to two-way network structure. The bi-directional recurrent network structure model can not only learn the above information features of the text, but also learn the following features of the text. For example, Miyamoto et al. [13] and Peter [17] use bidirectional LSTM structure to build the model.

2) Other technologies introduced

On the one hand, researchers improve the network structure, on the other hand, they seek external technology to improve the performance of language model. At present, the external technologies that are introduced into language modeling mainly include: one is the introduction of external information context, such as the application of latent dirichlet allocation (LDA) and knowledge map. The two is the introduction of various technologies, such as Dropout technology, word vector development from word level to character level to the combination of them, attention mechanism (including self-attention mechanism).

a) External context information

In 2012, Tomas mikolov proposed a language model using LDA [8]. In his work, this model can learn context features from external historical information, so it can better capture long-term dependence. On this basis, the follow-up researchers put forward many improved versions.

In Tomas mikolov's LDA language model, only one time scale context information is considered. Morioka et al. Proposed a recurrent neural network language model in 2015, which takes multiple time scale contexts as external information [9].

In 2016, the knowledge map was used in the modeling of language model by Ahn et al. They used the prior knowledge information provided by the knowledge map to predict whether the words generated by the model have the potential. The model reduced the number of uNK words [10].

b) Various technologies introduced

In 2014, dropout technology was introduced to LSTM neural network by Zaremba et al. to model language [11], which greatly improved the performance of LSTM neural network language model and achieved remarkable results in practical application tasks such as machine translation and image description.

The character features of words are also helpful for semantic information. Therefore, in 2016, the character recognition neural network language model [12] was proposed by Kim et al. In 2016. Firstly, the convolution neural network was used to train the character level expression from the word shape, reducing the network parameters, reducing the computational complexity and improving the training speed. On this basis, Miyamoto and Cho use BiLSTM to extract character feature vectors from words, and then combine character level and word level vectors as input. They use a gating strategy to realize the adaptive combination of the two vectors [13].

The regularization and its optimization strategy are used by Merity's team to build a language model based on LSTM [14]. The input of this model uses the word level vector as the feature expression of modeling, and obtains the best experimental effect in the word level modeling task.

In 2014, when the attention mechanism was first applied to the field of NLP by bahdanau et al., attention mechanism was quickly introduced into the construction of RNN language model. Soon, in 2016, it was confirmed by Tran and Mei et al. [15], the introduction of attention mechanism was very helpful to the performance improvement of RNNLM.

In 2018, Hongli Deng et al. Proposed to apply self-attention mechanism to RNN language modeling, and applied her model to sentence expression, and achieved good experimental results [16].

In 2018, Peter et al. Constructed an Elmo model based on BiLSTM-RNNLM [17], which was pre-trained on a large text corpus through the functions obtained from the internal state learning of the deep bidirectional language model (BiLM), which is a new deep context word representation. Experiments show that this model can significantly improve the performance of the language model.

C. PPL Comparison of RNN Language Models

Table I summarizes the ppl values of the above RNN

language models on the data set PTB test set. The data set PTB is preprocessed by reference [2]. Among them, the first 0-20 parts of the data set are set for training set, with 42068 statements and about 1M words. 21-22 are data set as the verification set, with 3370 statements, and the rest are set for test, with 3761 statements.

TABLE I PPL COMPARISON OF RNN LANGUAGE MODEL ON DATA SET PTB TEST SET

Models	External context information	External technology	PPL on test set
Basic RNNLM	-	-	*224*
Basic LSMLM	-	-	*203*
RNNLM	*LDA*	-	*207*
RNNLM	*Knowledge map*	-	*182*
LSTMLM	-	*Regularization*	*121*
LSTMLM	-	*Char+Word*	*107*
RNNLM	-	*Attention*	*124*
RNNLM	-	*Self-Attention*	*78*
LSTMLM	-	*Self-Attention*	*73*
BiLSTM-RNNLM	*Depth context*	*Self-Attention*	*68*

From table I, it can be seen that the performance of the language model based on the recurrent neural network is also improving from the initial simple structure to the continuously improved model.

V. FUTURE OUTLOOKS

These studies have done a lot of research on the performance improvement of RNN language model in different ways, but how to build a better language model in the end-to-end deep RNN architecture is a problem worthy of researchers' consideration. In addition, in the current language model building, the introduction of external prior knowledge mainly comes from the text, which does not involve whether image, voice and other information can be integrated? It's a direction worth considering.

ACKNOWLEDGMENT

This research was supported by the Innovation Team of China West Normal University under Grant No.No.CXTD2017-6) and Excellence Fund of China West Normal University (No.17Y183)

REFERENCES

[1] Y. Bengio, R. Ducharme, P. Vincent, et al., "A neural probabilistic language model", *Journal of machine learning research*, 3 (Feb):1137–1155, 2003.

[2] T. Mikolov, M. Karafiat, L. Burget,et.al., "Recurrent neural network based language model"//Eleventh Annual Conference of the International Speech Communication Association, pp. 1045–1048, 2010.

[3] S. Jean, K. Cho, R. Memisevic,et.al., "On using very large target vocabulary for neural machine translation", arXiv preprint arXiv:1412.2007,2014.

[4] A. Mnih, Y. W.The, "A fast and simple algorithm for training neural probabilistic language models", arXiv preprintar Xiv:1206.6426,2012.

[5] O. Vinyals, M. Fortunato, N. Jaitly, Pointer networks//Advances in Neural Information Processing Systems, pp. 2692–2700, 2015.

[6] S. Hochreiter, J. Schmidhuber, "Long short-term memory", *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[7] M. Sundermeyer, R. Schluter, H. Ney, LSTM neural networks for language modeling//Conference of the International Speech Communication Association, pp. 194–197, 2012.

[8] T. Mikolov, G. Zweig, Context dependent recurrent neural network language model//Spoken Language Technology Workshop, pp. 234–239, 2012.

[9] T. Morioka, T. Iwata, T. Hori, et.al., Multiscale recurrent neural network based language model//Sixteenth Annual Conference of the International Speech Communication Association, 2015.

[10] S. Ahn, H. Choi, T. Parnamaa, et.al., A neural knowledge language mode.arXiv preprint arXiv:1608.00318,2016.

[11] W. Zaremba, I. Sutskever, O. Vinyals, Recurrent neural network regularization, arXiv preprint arXiv:1409.2329,2014.

[12] Y. Kim, Jernitey , D. Sontag, et.al., Character-aware neural language models//AAAI Conference on Artificial Intelligence, pp. 2741–2749, 2016.

[13] Y. Miyamoto, K. Cho, "Gated word-character recurrent language model", *Proceeding s of the 2016 Conference on Empirical Methodsin Natural Language Processing*, pp. 1992–1997, 2016.

[14] S. Merity, S. Keskarn, R. Socher, Regularizing and optimizing LSTM language models, arXiv preprint arXiv:1708.02182,2017.

[15] T. D. Q. Vinh, T.-A. N. Pham, C. Gao, and L. L. Xiao, Attention-based Group Recommendation, 2016.

[16] H.L. Deng, L. Zhang, L.T. Wang, "Global context dependent recurrent neural network language model with sparse feature learning", *Neural Computing and Applications*, 2018.

[17] M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, Deep contextualized word representations, arXiv preprint arXiv:1802.05365V2