基于Attention机制的LSTM语义模型安卓聊天机器人的实现

信息152班：行习铭 指导教师：郭改慧

（陕西科技大学文理学院 陕西 西安 710021）

**Realization of Attention-Based LSTM Semantic Model Android Chatbot**

**ABSTRACT：**As an important branch of natural language processing, intelligent chatbot is of great significance to the development of human-computer interaction. We design scheme of mandarin chatbot, an open-domain chatbot software, that integrates the joint results of Information Retrieval (IR) and attentive Sequence to Sequence (Seq2Seq) based generation models while both encoding and decoding are realized with Bidirectional LSTM. Using a professional domain database to retrieve professional terms, and movie dialogue set to generate mandarin reply when retrieval failure or user chit-chat. Empirical study shows that our chatbot can generate grammatically correct and content-wise appropriate responses to most of the input text, outperforms both IR and generation based models.

1 Introduction

Let people communicate with a machine is the fundamental challenges in artificial intelligence (AI). If a person cannot tell whether he is communicating with a machine or a person, the machine has intelligence(Turing, 1950).The research of conversational robot can be traced back to Eliza (Weizenbaum,1966), Parry (Colby,1975), and Alice (Wallace, 2009), such chatbots were designed to mimic human behavior in a text-based conversation within a controlled scope by hand-crafted rules.Since the 1990s, a lot of research has been done on task-completion conversational systems based on data-driven or machine-learned approaches (Hemphill et al., 1990; Price, 1990; Dahl et al., 1994; Walker et al., 2001, 2002). Their performance is excellent only within domains that have well-defined schemas (Glass et al., 1995; Walker et al., 2001; Raux et al., 2005; Andreani et al., 2006; Tur and de Mori, 2011; Wang et al., 2011).

In the past decade, virtual personal assistants have emerged as the most prominent research focus in goal-oriented conversational systems, which allow users to speak naturally in order to accomplish tasks more effectively, such as Apple’s Siri ([https://www.apple.com/ios/siri/), Microsoft’s Cortana (https://www.microsoft.com/enus/cortana/](https://www.apple.com/ios/siri/),%20%20Microsoft's%20Cortana%20(https://www.microsoft.com/enus/cortana/)), Google Assistant, Facebook M (<https://developers.facebook.com/blog/post/2016/04/12/>), and Amazon’s Alexa (<https://developer.amazon.com/alexa/>). These virtual personal assistants are often deployed on mobile devices and are designed to answer a wide range of questions.

To offer better user experience, it is necessary to design an open-domain chatbot software, we propose a hybrid approach that integrates both IR and generation models. Specifically, for a question, we first use an IR model to retrieve technical term in a sentence and calculated similarity, if not satisfied and then using an attentive Seq2Seq model generate answer: (see Fig. 1 for the detailed process).

Our paper makes the following contributions:

• We propose a novel hybrid approach that uses an attentive Seq2Seq model to meet both professional and chit-chat requirements.

• We conducted a set of experiments to assess the approach. Results show that our approach outperforms both IR and generation.

• We compared our chatbot engine with a public mandarin chatbot. Evidence suggests that our software has a better performance.

• We launched our chatbot for a real-world application.

The rest of the paper is structured as follows: Section 2 presents our IR model, attentive seq2seq model in Section 3, and Section 4 concludes our work.

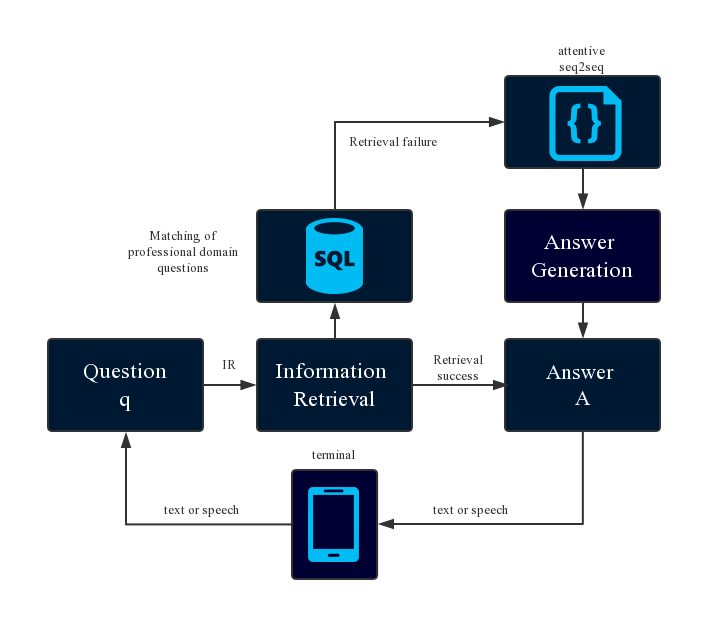


Figure 1: Overview of our hybrid approach.

2 IR Model

The chatbot implementation is roughly divided into IR model and generative models. The IR chatbot model needs to build a large database system to contain data about possible problems for users, who, after giving their own questions, will respond to the most similar match ingested in the database based on the problem.

Our retrieval model uses search techniques to find the most similar questions for each input, and then gets a matching answer. By word segmentation, we build an inverted index of all the problems in the database by mapping each word to the problem set that contains that word. Given a problem, we split it into a set of words, delete the stop words, extend the set with their synonyms, and call back a set of QA candidate pairs with the improved set. We then used BM25 (Robertson et al., 2009) to calculate the similarity between the input question and the retrieval question, and took the paired answers of the most similar questions as the answers. The formula is as follows:

 (1)

Where weight and function  are typically defined as:

 (2)

 (3)

Where function  is the correlation value of each word  in the user input statement with related  in the entire database. is the frequency of in the database.In the QA task, The longer the length of correlation reply  is, the largeris, and the smaller the correlation is with .

3 Attentive Seq2Seq Model

We present an overview of our approach in Fig. 1. At first, we construct a QA knowledge base from the professional domain database. Based on this QA knowledge database, we then develop two models: an IR model, a generation based model. There are two points to be noted: (1) all the two models are based on words (i.e., word segmentation is needed): the input features of IR model are words, while those of generation model are word embeddings; (2) our generation based model is built on the Seq2Seq structure. Given an input question q, the procedure of our approach is as follows:

3.1 Encoder-Decoder

Generation-based approaches have recently made great progress due to advancements in deep learning. In this approach, an encoder-decoder-based neural network model is used (Sutskever et al., 2014; Bahdanau et al., 2015). First, the message from the user and the contextual information are encoded into representation vectors, usually by a long short-term memory (LSTM) (Hochreiter and Schmidhuber, 1997) recurrent neural network (RNN). These representation vectors are then fed into a decoder, usually another LSTM, to generate the response word by word (Vinyals and Le, 2015).

In this paper, the user's questions are taken as the source sequence of the chatbot model, and the answers returned by the system are used as the target sequence of the model. The Encoder-Decoder framework is a relatively mature pattern for dealing with from sequence to sequence issues. Figure 2 abstracts the application of the Encoder-Decoder framework in the field of natural language processing into a common processing model, i.e. one sequence is converted to another. For the question-and-answer sequence pair, the Encoder-Decoder framework generates the target sequence Y with the input of the original sequence X and continuously changes the model parameters to enhance this possibility.

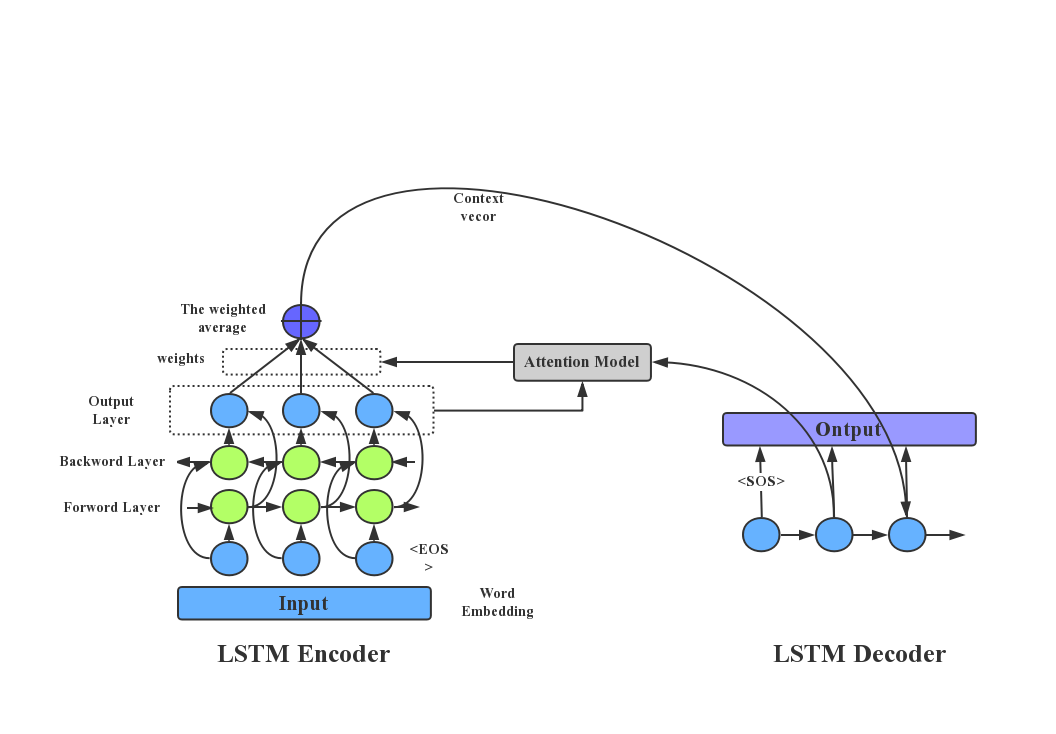
****

Fig. 2 illustrates such an encoder-decoder framework.

The neural network (LSTM) in the model calculates the conditional probability according to the following steps: first, the input sequence, through the layerl LSTM nerve unit in Encoder, obtains a vector representation of C of a fixed dimension from the final hidden layer state calculation, and then calculates the probability of the output sequence according to the standard LSTM-LM formula. The initial hidden layer state of the LSTM is a vector for the input sequence representing C:

 (4)

In this equation, each probability distribution  is represented by an activation function model for all words in the dictionary. Also, all sentences are represented by the sentence finale “<EOS>”, which allows the model to define a probability distribution of all possible lengths of the output sequence.

3.2 Word Embedding

Word Embedding, each word is used to convert to a real number, and each real number corresponds to a specific word in the dictionary. It is a technique for learning deep words in a low-dimensional word vector space, and by expanding the vocabulary, it can greatly increase the training speed, because some information is shared by embedding words very close lying in the space.

3.3 Attention Model

The core of Attention structure is to pay "Attention" to relevant source content through model decoder stage, so that a direct and short connection can be established between target sentence and source sentence, and the information fault between chatbot model and user can be solved. The basic idea of the attention model is to change the Encoder-Decoder structure network which relies on an internal fixed length vector limit. In fact, the attention model is a similarity vector. The closer the current input is to the target state, the larger the current input weight will be, indicating that the current output is more dependent on the current input.

After the attention mechanism is added, the encoder encodes the input sequence into a vector sequence, and the vector with the largest weight is calculated by the model as the input of the decoder. Thus, the calculation formula of the output sequence in RNN is

 (5)

The attention model can reduce the data dimension, reduce the computational burden of processing high-dimensional data, and make the model more focused on finding useful information in the input data with the current output, improving the quality of the output by selecting a subset of the inputs.

4 Conclusion

Based on the generative technical framework used by the mainstream implementation chatbot, this paper mainly uses the end-to-end model of Encoder-Decoder, which omits the intermediate lexical analysis and syntactic analysis, and reduces the excessive assumptions and guesses of the sequence. Very efficient. In the implementation of the Encoder-Decoder framework and combined with the LSTM neuron network, on the basis of which word Embedding word embedding, Attention mechanism, Beam Search cluster search algorithm, etc. were added to solve the information transmission, Personality consistency and answering diversity questions. Finally, this paper designed an Android-based chatbot software.

Acknowledge

The authors would like to thank my mentor and predecessor in academic. We would also like to thank reviewers for their valuable comments. References.

Reference

[1] Warren S.McCulloch, Walter Pitts. A logical calculus of the ideas immanent in nervous activity[J]. The bulletin of mathematical biophysics, 1943, 5(4):115-133.

[2] 张冲.基于Attention-based LSTM模型的文本技术的研究[D]. 南京:南京大学, 2016.

[3] 周志华.机器学习[M]. 北京:清华大学出版社, 2016.

[4] Srivastava N, Hinton G, 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.

[5] Lecun Y, Bottou L, Bengio Y, et al. Gradient-based learning applied to document recognition[J]. Proceedings of the IEEE, 1998, 86(11):2278-2324.

[6] Cho K, Van Merrienboer B, Gulcehre C, et al. Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation[J]. CoRR, 2014: abs/1406.1078.

[7] Sutskever I, Vinyals O, Le Q V . Sequence to Sequence Learning with Neural Networks[J]. CoRR, 2014: abs/1409.3215.

[8] Luong M T, Pham H , Manning C D . Effective Approaches to Attention-based Neural Machine Translation[J]. CoRR, 2015: abs/1508.04025.

[9] 戴怡琳, 刘功申.智能聊天机器人的技术综述[J]. 计算机科学与应用, 2018, 8(6): 918-929.

[10] Srivastava N, Hinton G, 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.

[11] Koehn P. Pharaoh: A Beam Search Decoder for Phrase-Based Statistical Machine Translation Models[J]. Washington, DC, 2004. 115-124.

[12] Boden M. A guide to recurrent neural networks and backpropagation[J]. Electrical Engineering, 2001, 2001(2):1-10.

[13] Qiu M, Li F L, Wang S, et al. AliMe Chat: A Sequence to Sequence and Rerank based Chatbot Engine[C]. Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers). 2017.

[14] Sutskever I, Vinyals O, Le Q V. Sequence to Sequence Learning with Neural Networks[J]. Advances in Neural Information Processing Systems, 2014, 3104-3112.

[15] Yao K, Zweig G, Hwang M, et al. Recurrent Neural Networks for Language Understanding[J]. ISCA Interspeech, Lyon, France, 2013, pp.2524-2528.