# Improved Generation-based and Improved Search-based Model on Chatbot

1st Yanwei Jiang
*Shanghai Jiao Tong University*
518030910274

1st Bowen Qu
*Shanghai Jiao Tong University*
518030910279

***Abstract*—Chatbots are robots programmed to respond like humans. Today, chatbots have been endowed with the ability to communicate effectively with humans. In this study, we want to know two chatbots Nao talk to each other about a given topic. The chatbot needs to perform speech recognition and then respond like humans to another chatbot. Normal chatbot is consist of two kinds, search-based and generation-based. We want to improve these two kinds of chatbot. We first recurrent the classic sequence to sequence model used by generation-based chatbot. Then, to fix contextual issues in multiple rounds of conversation, we add hierarchical neural network to sequence to sequence model. Hierarchical neural network is a sequence to sequence model but the encoder uses an auxiliary structure. It can include context and speak more logical. We also add some processing to the questions so the conversation will continue around topic. It gets a better performance on Cornell Movie-Dialogues Corpus from loss 1.893 to 0.808. But the sentence is less of smooth and understandable. We improved the search-based chatbot by adding the encoder model in sequence to sequence model. We also add some processing to the questions so the conversation will continue around topic. This improved model get a good performance on our given topics.**

***Index Terms*—chatbot, seq2seq, hierarchical neural network, search-based, generation-based**

## I. Introduction

Chatbots are robots programmed to respond like humans, or in general, the hardware is not essential, chatbot can be even just a software that can respond like humans. Many of the existing robots are specific to answer a human's question, but there are not a type of robot that can talk with each other.

Our work is to build a chatbot which can communicate with another chatbot about a given topic. It can perform speech recognition and then respond like humans, and the conversation go on until a symbol to terminate.

Early in 1950, Alan Turing has already given the famous Turing Test [1], this the prototype of chatbots. He imaged that computer can cheat 30% of adults in a 5-minute conversation. If a computer pass the Turing Test, then we think the computer owns wisdom of people. The Turing Test acquires the computer can chat with humans. This can be recognized as the origin of chatbot.

The first chatbot in the world is created in 1966 in MIT. [2] Joseph Weizenbaum designed a program to imitate a psychologist in clinical treatment. This program was implemented by simple keyword match, then reply in some template designed by people. Although the method is kind of naive now, it still make the world shocked. People are surprised at its outstanding result. This the first search-based chatbot.

After 22 years, ai was riding on a rocket, developing quickly, people reignited the interest of chatbot. Robert et al. developed the UC(UNIX Consultant) chatbot system. [3] This system aimed at solving the problems about UNIX of users. UC can analyse the dialog and find a solution of user's problem and finally express it to the user. This system is widely used in most of the costumer service programs. Just like ELIZA, UC is search-based.

In 1995, Richard S. Wallace released ALICE. [4] This open-source system still works until now, you can find it on *alice-bot.org*. Along with ALICE, a new method, AIML(Artificial Intelligence Markup Language) come into people's eyes. AIML is a form of template of marked language for ai, just like Markdown. [5]. ALICE and AIML brings search-based chatbot popular.

Nowadays, search-based chatbot is the mainstream, but some scientists came up with generation-based chatbot and Knowledge-Graph-based chatbot. These require more ai technology and will be the future destination of chatbot.

In this article, we tried to build a chatbot which can communicate with another about a given topic. It can perform speech recognition and then respond like humans, and the conversation go on until a symbol to terminate. We plan to make an improved generation-based chatbot with hierarchical neural network and make some process to force chatbot to make sentence around topic.

## II. Related Work

Chatbot is a popular field in AI, there are many famous companies devoted to this, such as Apple's Siri, Microsoft's Cortana, Huawei's Xiaoai.

Generally speaking, there are Goal–Driven robots and Non-Goal–Driven robots. The Goal–Driven robots serve for a clear target or a certain group of person, such as customer service robots. Non-Goal-Driven robots differs, its target is ambiguous, not for a certain goal. Such as robots just chat for fun.

If classify by technology, chatting robots can be divided in two categories, search-based robots and generation-based robots. The search-based type need a large database with lots of conversation, when accepting a sentence, querying it in database, then return the proximate answer. While the generation-based robots are established on some machine-learning method. They try to analysis the sentence and generate a proper reply.

Until now, most of the chatbots are search-based, the instances of generation-based are hard to find.

## III. Method

To deal with this problem, we work out two methods to deal with AI vs AI chatbot. One is an improved generation-based chatbot, which use the hierarchical neural network. Another is an improved search-based chatbot, which use encoder and matching.

### A. *improved generation-based chatbot*

We build an improved generation-based chatbot based on sequence to sequence model. But simple sequence to sequence model can not deal with contextual issues in multiple rounds of conversation, so we add hierarchical neural network to build auxiliary structure of recurrent neural network (RNN). For another problem that the conversation cannot revolves around the topic, we do some statement processing to the conversation.

*1) sequence to sequence model:* Seq2seq [6] is a family of machine learning approaches used for language processing. Applications include language translation, image captioning, conversational models and text summarization. Seq2seq turns one sequence into another sequence. It does so by use of a recurrent neural network (RNN) or more often LSTM or GRU to avoid the problem of vanishing gradient. The context for each item is the output from the previous step. The primary components are one encoder and one decoder network. The encoder turns each item into a corresponding hidden vector containing the item and its context. The decoder reverses the process, turning the vector into an output item, using the previous output as the input context. Training typically uses a cross-entropy loss function, whereby one output is penalized to the extent that the probability of the succeeding output is less than 1.

Seqence to sequence model is shown in Figure 1.

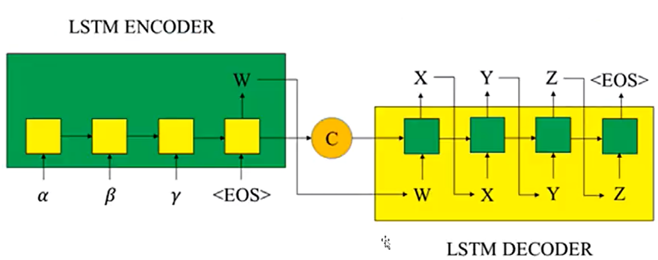


Fig. 1. the sequence to sequence model.

*2) hierarchical neural network:* Hierarchical neural network (HNN) [7] is essentially an Encoder-Decoder framework. The main difference is that the encoder uses an auxiliary structure. Each sentence in the context first uses "Sentence RNN" to encode each word to form an intermediate representation of each sentence. The second-level RNN encodes the intermediate representation results of the first-level sentence RNN according to the order of the sentences in the context. This level of RNN model can be called "context RNN", and its tail node is a hidden layer. The node state information is the semantic encoding of all contexts and the current input Message. This information is used as one of the inputs of each word generated by the decoder so that context information can be considered when generating the word in response to the response.

Hierarchical neural network model is shown in Figure 2.

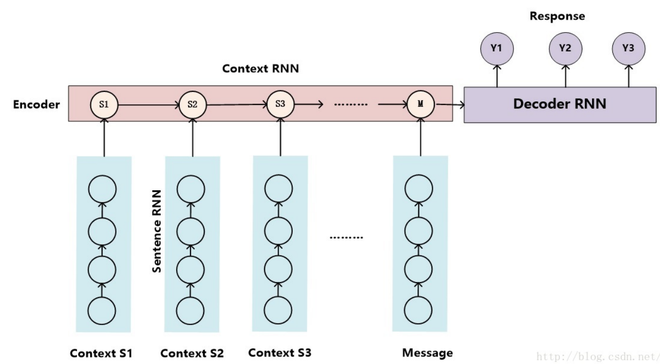


Fig. 2. the hierarchical neural network model.

*3) statement processing:* To ensure the conversation is revolves around the topic, we do some statement processing. First, we start the conversation with the sentence pattern "What so you think about the topic?", so the conversation will start with the topic. Then, we add sentence pattern "As for topic" to the beginning of the question to keep the conversation around the topic.

### B. *improved search-based chatbot*

Another idea is using search-based chatbot. We store a large quantity of conversations and for the topic, when given a topic, we find a conversation that most likely to the topic. We still use the encoder network which has been training in the improved generation-based chatbot to turn input into a vector in 1024 size, then matching input and questions to find which question fit best and use the answer as the answer of the input. For another problem that the conversation cannot revolves around the topic, we do some statement processing to the conversation.

*1) matching:* To matching input with the questions, we calculate Minkowski Distance of them to define the difference between two questions. Then we find the minimum distance and classify it to the question. The answer of the question is the answer of input.

*2) statement processing:* To ensure the conversation is revolves around the topic, we do some statement processing. For one way, we replace some pronoun in the answers with the topic. For another, we delete some universal answers to avoid Safe Response problem, which means that no matter what the user says, the chatbot will respond with a few very common sentences.

TABLE I
IMPROVED GENERATION-BASED MODEL LOSS AND CONVERGENCE EPOCH

| Model | Original model | Improved model | | | | |
|---|---|---|---|---|---|---|
| | | 5000_1 | 10000_1 | 10000_16 | 100000_1 | 100000_32 |
| loss | 1.893 | 4.465 | 4.563 | 0.808 | 4.677 | 1.251 |
| epoch | 48 | 45-undone | 25 | 61 | 4-undone | 31 |

[a] 5000,10000,100000 means number of conversation in data set.
[b] 1,16,32 means batch size.
[c] undone means no trend to convergence.

TABLE II
IMPROVED SEARCH-BASED MODEL CONVERSATION WITH TOPIC SUIT

| Robot | Sequences |
|---|---|
| R2: | what the hell's suit matter. |
| R1: | how could i forget about you you are the only person i know. |
| R2: | the young are eating the old something that usually does not happen here. |
| R1: | suit orders. |
| R2: | nobody whose name you want me to say mr young i promise you. |

## IV. EXPERIMENT

### A. Experimental Setup

Our data set is using Cornell Movie-Dialogues Corpus [8], for it has contextual issues in multiple rounds of conversation. This corpus contains a large metadata-rich collection of fictional conversations extracted from raw movie scripts. This corpus contains 220,579 conversational exchanges between 10,292 pairs of movie characters, involves 9,035 characters from 617 movies and in total 304,713 utterances.

For data processing, we choose a certain number of conversation in no more than 15 sentences and each sentence in no more than 25 words. Build dictionary to store words frequency, and remove the words which frequency is less than 15. Then numbering each word and turn the sentence of type string into integer list. Sort questions and answers by questions' length to reduce the time of training. Filling the questions into matrix and it is ready to inputs for the network.

Our network setup and details of models we applied in our system development is as follows. Training max epoch is 100, the size of one batch is 16 or 32. As for the structure of encoder, the size of each RNN size is 512, the number of the layers of RNN in encoder is 3. As for embedding, the size of encoder embedding is 1024 and the size of decoder embedding is 1024. As for the optimization, learning rate is 0.01, min learning rate is 0.0001 and learning rate decay is 0.9, keep probability is 0.5.

We build and train the network in environment Python3.5 and TensorFlow1.0.0 on GPU TITAN X, load and use the network in Python3.6 and TensorFlow1.13.1.

For search-based chatbot, we use encoder model which is build in first half sequence to sequence model. We put all questions in to the model and set keep probability 0.9 to avoid random and keep the conversation diversification. Store the vector in a table, and put input into the encoder and find the minimum Minkowski Distance question. The answer of the question is output.

### B. Experimental Result

We evaluate our proposed solutions by loss and convergence epoch. Our loss is get sequence loss in model sequence to sequence. Sequence loss means first get softmax, then get cross-entropy, finally get the means of loss. Convergence epoch means the model loss convergent in which epoch.

We change parameters data set scale and batch size to try to get a best model, the result is in Table I. We found that only data set size 10000 and batch size 16 and data set size 100000 and batch size 32 get a better loss than original model. And the data set size 10000 and batch size 16 has the best loss.

As for loss, we get a great success. But as for sentences smooth and understandable we meet some trouble. When we test the model in normal case, we find that the result is not satisfy our expected. The model always repeats several words with no relationship when communicating with each others. We think it may caused by unfitted data set and complex network. Although it use second encoder to combine context chat message, the most important question information has been compressed and the indeed information we need to get is dropped. Another reason is the data set is from movie dialogues and it contains a large amount of folk and safe answer.

Consider the improved generation-based chatbot, the improved search-based chatbot performs better for its answer is generate from the real sentence. An example which gives topic "suit" is in Table II.

## V. CONCLUSION

In this study, we use data set Cornell Movie-Dialogues Corpus to train an AI vs AI chatbot Nao.

We first improved the sequence to sequence model to build an improved generate chatbot with hierarchical neural network and make some process to force chatbot to make sentence around topic. We successful reduce loss from 1.893 to 0.808, but the sentence is neither smooth nor understandable.

So we get a second method to make an improved search-based chatbot with encoder in sequence to sequence and matching input with questions. Then substitute pron with topic words. It get a good result on our test examples.

## REFERENCES


[1] Turing A M. Computing machinery and intelligence[M]//Parsing the turing test. Springer, Dordrecht, 2009: 23-65.

[2] Weizenbaum J. ELIZA—a computer program for the study of natural language communication between man and machine[J]. Communications of the ACM, 1966, 9(1): 36-45.

[3] Wilensky R, Chin D N, Luria M, et al. The Berkeley UNIX consultant project[J]. Computational Linguistics, 1988, 14(4): 35-84.

[4] Wikipedia, ALICE. https://en.wikipedia.org/wiki/Artificial_Linguistic_Internet_Computer_Entity

[5] John Gruber, Markdown, https://en.wikipedia.org/wiki/Markdown

[6] Wadhwa, Mani (2018-12-05). "seq2seq model in Machine Learning". GeeksforGeeks. Retrieved 2019-12-17.

[7] Serban I, Sordoni A, Bengio Y, et al. Building end-to-end dialogue systems using generative hierarchical neural network models[C]//Proceedings of the AAAI Conference on Artificial Intelligence. 2016, 30(1).

[8] Danescu-Niculescu-Mizil+Lee:11a,Cristian Danescu-Niculescu-Mizil and Lillian Lee,Chameleons in imagined conversations:A new approach to understanding coordination of linguistic style in dialogs. Proceedings of the Workshop on Cognitive Modeling and Computational Linguistics, ACL 2011