Intelligent Chatbot Using RNN

Harsha Potluri[1],Prof. Ajaykumar T. Shah[2]
[1] Dept of Computer Engineering
[2]HOD, Dept of Computer Engineering
[1, 2] Alpha College of Engineering and Technology

Abstract- *A chatbot is an intelligent system that identifies the input and provides output based on the provided training data. Recurrent Neural Networks provide continuous context to a conversation which allows prediction of next word. The output of the previous step is fed as input to next step to maintain context. RNN's are widely used for neural networks that require the context to be carried forward. There are so many different architectures developed for RNN's. We will be implementing LSTM RNN for our proposed system. This particular architecture consists of LSTM and attention model implementation.*

Keywords- Python, Keras, Tensorflow, Recurrent Neural Network, LSTM's, Bi-RNN's, Attention model, NLP.

I. INTRODUCTION

Neural Networks have been fascination for humanity for at least 5 decades when psychologist Frank Rosenblatt introduced Perceptron to ideate human thought process in its simplest form. Since then the idea of a synthetic recreation of human mind has been in research and development. The nearest resemblance to a thought process is Neural Network that is being currently used and being improved to develop programs that can perform certain task by itself. There are many types of Neural Networks that are used for different tasks like image identification or text summarization, automation and many more. In this we will be focusing on RNN's (Recurrent Neural Networks).

Recurrent Neural Network (RNN) is the popular architecture used in Natural Language Processing (NLP) tasks since it is a very suitable structure to process input text of varying length [1]. An RNN is a neural network that provides its output to the next state allowing the information to persist in any given cell. RNN's are useful for this particular fact and they provide continual context retainment through network. RNN utilizes words by converting the text into tokens which furthermore, comprises of text vectors, which form a matrix [2]. Matrix input is very easily consumed by any neural network. The process of learning for any Neural Network (RNN in this case) is to identify weightage, biases and activation value for any given RNN architecture.

LSTM (Long Short-Term Memory) neural network has an objective of predicting target words based on context, they also require large vocabulary computation during training. LSTM are capable of learning long term dependencies. And these are very efficient for learning context and grammar which is essentially important for a chatbot. LSTM may contain a singular activation or have complex structure with multiple activations.

Attention Model provides us to highlight importance to certain areas in image or text. This decreases the computational overhead created by unnecessary objects be it text or a background in an image. Which in turn increases efficiency and resources are saved.

II. LITERATURE REWIEW

There has been ongoing research on this topic for years and quite a few development has been done in the field of neural network. There are various Methods to implement encoding of the input sentences: LSTM based models, bi-directional LSTM's, GRU's, CNN's and many more architectures. GRU's cannot use the information from future tokens which is one of the major drawbacks for context heavy chatbots [3]. While CNN's do not encode the position or orientation of the object which is important for grammatical information for the chatbot [3]. Bi-directional LSTM can utilize previous and future context as well as provides importance to the order of the sentence.

In one research, they used bidirectional LSTM, and they fed one LSTM network with the sentence words from left to right, and another from right to left. They were fed sets of left-to-right and right-to-left context word embeddings(vector) [4]. They used two separate vectors one consisting of word vector while the other consisted of context. These are to differentiate the words with context and computed separate and then the final target embedding is provided while considering the context.

In another research the proposed language model is a form of standard feed-forward neural network language model (NNLM) [5]. They implement attention mechanism for their model which focuses on the task of sentence-level

summarization. It incorporates less linguistic structure than comparable abstractive summarization approaches, but can easily scale to train on a large amount of data [5]. Since this system takes no vocabulary conversion of the generated summary it is trained on document-summary pair.

Keras is a Neural Networks API, written in Python and capable of running on top of TensorFlow, CNTK, or Theano [6]. Keras provides deep learning library that allows us to create complex Neural Networks easily without much hassle. It provides functions that can easily be computed on the CPU itself and decreases the time of computation. It allows for easy and fast and supports both convolutional networks and recurrent networks, as well as combinations of the two.

III. STUDYAND FINDINGS

This system consists of bi-directional LSTM implemented on first layer which allows the system to remain context aware and will not change the tone of dialogue. It may provide confusing answers due to highly unreliable data fed during the training time. For more accurate results we have to input highly monitored and serious conversed data. these kinds of data are not available on for singular entities or really difficult to come across. Bi-directional LSTM can provide better results for more uniform data.

The attention model is implemented after the LSTM layer to decrease the computation in second layer which allows for easier identification of important words and provides output based on those highlighted words. The Bidirectional LSTM's are consisted of two layers one with the regular input and the other with reversed input. The more the intricate the layers the more the accurate the outputs. The attention layers are quite handy to provide words with importance a higher weight and activation value.

The vocabulary of the present system consists of only 50,000 words since it causes more time and resources for exceptional outputs and may take over 2 to 3 weeks for more optimal results. The available vocabulary is also quite large and hence it will provide bot with decent language knowledge.

The training will require external GPU to compute such large network. So, an external resource will be implemented for running the model.

This system is also one user deployed in its current state and will require a separate database to keep track of user and their data. Since the task is costly it will be revisited in next iteration/version of the development.

The system is currently only for the desktop over 4gb ram and i3 processors since they do require computation. They are currently only being considered for a laptop/PC interface.

The user should also be able to understand and communicate in English language since the data on which it was trained on was in English. The bot may also use some slangs which can become a barrier in communication. Also, the personality of bot is undefined and may come off as strongly opinionated and offensive. But since we cannot remove all the offensive data some might have passed through the filter and can become problematic for certain words. A filter is present but some words may slip through filter and lad in its vocabulary.

IV. FUTURE ENHANCEMENT

The future enhancements can include a better interface of the application. Since humans get attracted to better looking and colorful the interface/GUI should have better outlook.

The application can be multi login and have user chat history retainment for each user. This will be implemented with better database with fast transactions from direct server.

The size of vocabulary can be increased to a 100,000 so the bot will have more topics and become broader conversationalist with more accurate usage of words. The training can be also increased for the bot to have better topics and opinion broadens. A web scraper can also be implemented for the bot to have up to date knowledge of happenings in real world.

V. CONCLUSION

This system is currently self-sufficient and can be further improved with better resources. Efficient model considering the given resources. The system is adequately accurate and provides quite an efficiency. The system is user friendly and has further margin of improvement. The proposed model is not very costly for given training requirements.

VI. AKNOWLEDGEMENT

I would like to express sincere thanks to Prof. Ajaykumar T. Shah, Head of Department of Computer Engineering, Alpha College of Engineering and Technology for their complete support and guidance throughout the project for successful completion. I would also like to thank every person that was involved in support and guidance in this endeavor.

REFERENCES

[1] http://colah.github.io/posts/2015-08-Understanding-LSTMs/
[2] Peng Zhou, Zhenyu Qi, Suncong Zheng, Jiaming Xu, Hongyun Bao, Bo Xu for "Text Classification Improved by Integrating Bidirectional LSTM with Two-dimensional Max Pooling" in arXiv:1611.06639, 21 Nov 2016
[3] Yang Liu, Chengjie Sun, Lei Lin and Xiaolong Wang for "Learning Natural Language Inference using Bidirectional LSTM model and Inner-Attention" in arXiv:1605.09090, 30 May 2016
[4] Oren Melamud, Jacob Goldberger, Ido Dagan for "context2vec: Learning Generic Context Embedding with Bidirectional-LSTM" in Bar-Ilan University
[5] Alexander M. Rush, Sumit Chopra, Jason Weston for "A Neural Attention Model for Abstractive Sentence Summarization"
[6] https://keras.io/