International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 07 Issue: 02 | Feb 2020 www.irjet.net p-ISSN: 2395-0072

Deep Learning based Chatbot

Ryan Noronha[1], Ankesh Gaikwad[2]

[1,2] *Student, Dept. of Computer Engineering, Fr. C. Rodrigues Institute of Technology, Vashi, Maharashtra, India*

Abstract - *Deep Neural Networks (DNNs) find their application in many modern day systems. From analysing images to predicting weather parameters and many other applications in between. One such application for DNNs is found in chatbots. Chatbots are very useful for answering questions of customers that may require heavy memorisation and questions that maybe very repetitive in nature. Through this paper we would like to share our implementation of a general purpose chatbot that can be used to converse with like another human being. Chatbots require a substantial amount of data to be trained against. For this purpose we are using comment dumps from the popular social networking website, Reddit.*

Key Words: Sequence-to-Sequence, seq2seq, Chatbot, LSTM, DNN, ANN, Deep Learning

1. INTRODUCTION

Deep Neural Networks (DNNs) have proved themselves very valuable in modern day issues. Today, DNNs are used in applications such as forecasting weather and the stock market, non-player characters (NPCs) in video games, object detection in robots and many more [1].

DNNs are powerful models that are capable of performing parallel computations and have achieved excellent results in difficult learning tasks. Despite their might, a significant drawback of DNNs is that they can only be used where large training sets are available; they cannot be used to map a sequence to another sequence, which is the nature of chatbots. Furthermore, DNNs can only be used where the input can be sensibly encoded into feature vectors of fixed dimensionality. Many real world problems are best explained with feature vectors of varying dimensionality i.e. the length of the input vector is not always known beforehand.

To implement this chatbot, we will be using Long Short-Term Memory (LSTM) based neural networks. LSTMs allow us to map a variable length input vector into a fixed-dimensional vector representation. The idea is to use an LSTM to map the variable length input to a fixed-dimensional vector representation, and then use another LSTM to extract the output from that vector.

The model that we will be using is an open source neural machine translator (NMT) developed by Google [3]. This model has been trained for translating English sentences to French [4]. The reason we can use an NMT in a chatbot is because a chatbot can be thought of as a translator that translates a human utterance into a machine generated response [2]. Basically, a chatbot can be visualised as a translator that translates, a sentence to another sentence of the same language (in our case English).

2. SOME EXISTING SYSTEMS

Chatbots are present all around us today. One would find a chatbot as a digital assistant on one's smartphone. Customer service is another sector where one can find chatbots. Chatbots are used to replace some humans for answering questions. In this section we present some existing chatbot systems.

2.1 Amazon Alexa

Alexa is a virtual assistant developed by Amazon.com, Inc. for use in their Amazon Echo devices. Alexa is capable of understanding a large number of languages such as English, German, French, Hindi, Marathi, Punjabi, and many more. Alexa can play music, set alarms and reminders, play audiobooks, provide weather and traffic information [6]. In addition to this, Alexa can also control several smart devices such as lighting, thermostats and more. Additionally, users can also install third party commands, also called as Alexa Skills which can perform tasks that cannot be done by Alexa out of the box.

2.2 Siri

Siri is Apple's offering in the world of virtual assistants. Siri is an integral part of Apple's iOS, iPadOS, watchOS, macOS, and tvOS operating systems. Similar to Amazon Alexa, Siri can also operate on voice commands and can be summoned by saying the phrase "Hey, Siri…". [8]

2.3 Artificial Linguistic Internet Computer Entity (ALICE)

ALICE was developed by Richard Wallace in 1995. It is a free and open source chatbot [9] implemented using an XML Schema called Artificial Intelligence Markup Language (AIML). The chatbot evaluates the human input using some heuristic pattern matching rules.

2.4 IBM Watson

Watson is a question answering bot. This chatbot generates a number of replies to a question and uses rule based AI and other mechanisms to evaluate all the candidate replies and then generates a feature values. It then uses machine

learning to combine these feature scores to determine its top answer[10].

Watson was initially trained using encyclopaedias, dictionaries, structured and unstructured data to build a knowledge base for it. Watson can now automatically update itself through past experiences and also learn from new information published. Under the hood, Watson uses IBM's DeepQA software along with a bunch of open-source and proprietary software.

3. PROPOSED SYSTEM

As with any Machine Learning based system, the initialisation occurs after obtaining sufficient amounts of data that can be used for training the model. In our case, we have obtained data from Reddit comment dumps.

Initially, the raw data (comments and replies) will be moved to a database. This allows us to manage the raw data in a much more efficient manner. We then will pair a comment to a reply that has the highest score. A higher score of the reply generally would mean that the reply is much more preferred over the others and thus is more reliable. The next step will involve removing comments that are too long or are null. All of the aforementioned steps will be done in an SQLite database. Finally, the comment-reply pairs are converted into CSV format using Pandas for use in our model.

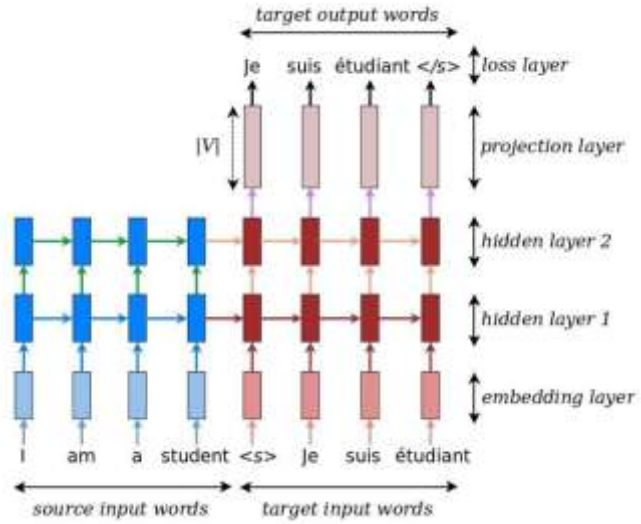

Fig -1: Architecture of Google's NMT

After the data is processed and converted into CSV, the data is fed to our Deep Learning model that is a Sequence-to-Sequence (seq2seq) model. The architecture of our seq2seq model is taken from Google's research (Fig. 1) [3]. This was one of the first Neural Machine Translators (NMT) and was state of the art. If we take a high level view of the NMT model, it has an encoder, decoder and intermediate steps as its main components. It uses embedding, so we have to first make a list of "vocabulary" that we wish our model should be able to read and understand.

After passing it through the embedding, the transformed input is then fed to an encoder. The encoder takes the comment as its input, passes it through an LSTM network which can take variable length inputs. It then maps this input to a fixed length vector which is the output of the encoder. This output of the LSTM is then passed to the decoder which takes this output and tries to find a mapping between this fixed length vector and generate a reply. The purpose of the memory is to be able to remember the sequence of the characters entered. In this way the model learns to predict the next sequence using accuracy metrics such as BLEU in order to determine the best possible sequence of characters that best matches the input sequence.

4. IMPLEMENTAION

The project is an attempt to create a system wherein a user can communicate effectively with a Deep Learning based bot based upon the seq2seq model. The user can use the web application which has a User Interface very similar to that of a chat messenger and interact with the chatbot. Each and every text message is displayed onto the screen and the replies given by the chatbot can be analysed. Details of each of the modules is given below.

4.1 Data Preprocessing

The data we obtained was from Reddit comment dumps. Comments in Reddit are structured such that a comment can have multiple replies and those replies in turn could have multiple replies and this can go on. Our goal was to obtain the immediate replies to every comment with the maximum score. The score is the sum of upvotes and downvotes for the reply. The steps carried out to preprocess the data are as follows.

4.1.1 Moving Over Relevant Data

The raw data obtained is in the form of JSON strings containing multiple parameters such as the ID of the commenter, the subreddit it was posted to, the ID of the comment and so on. The data that is relevant to our task are the comments, the replies, and scores. The comments were moved to a database containing two tables: (1) comments and (2) replies. The comments were directly put into the database in one step and the replies were moved in later. The replies were only added if the key matched with that of the comment. Additionally, during this step we removed comments or replies that either were too long or were deleted.

4.1.2 Pairing Comments and Replies

Now that the comments and replies were in a database, they are much easier to query. In this step, we take a comment and all the replies to that comment. The reply with the highest score is chosen and moved over to another table containing the comment and the reply with the highest score.

4.1.3 Converting to CSV Format

Now that we have the data that we need, we now convert the table into two CSV files which can then be used for creating training and testing sets. One CSV file contains comments and the other contains the corresponding reply.

4.1.4 Creating Training and Testing Data

Now we split the comments into two sections, one for training and another for testing. For this we have chosen a 60-40 split of the data i.e. 60% will be used for training and the other 40% will be used for testing the model.

4.2 The Chatbot Engine

As mentioned previously, we have used Google's NMT which was one of the first Deep Learning based translators which could translate an English input to French. The reason why we use a translator for a chatbot is because a chatbot can be thought of as a translator that translates one English statement to another English statement. For that reason, we have repurposed the NMT to work as a chatbot.

4.3 The User Interface

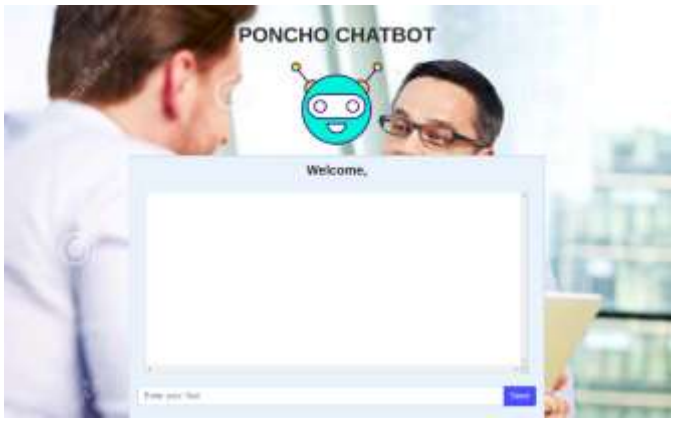

Fig -2: The UI when user just enters the chat

The User Interface (UI) is a very simple and familiar looking design. It looks and works like any other major messenger platform such as WhatsApp and Facebook Messenger. The UI was built using HTML, CSS and JavaScript. We have used Bootstrap to make the UI much easier to customise. Fig. 2 shows the UI when the user just boots up the chatbot.

Once a message has been typed in, the browser performs an AJAX call to the chatbot engine which takes in the message, processes it to obtain all the messages. A list of messages is returned by the chatbot that also contains the score for the message. The message with the highest score is chosen and is returned back to the browser. The browser then displays the reply to the message in the chat box. Fig. 3 shows the message being sent by the user and receiving a reply from the chatbot.

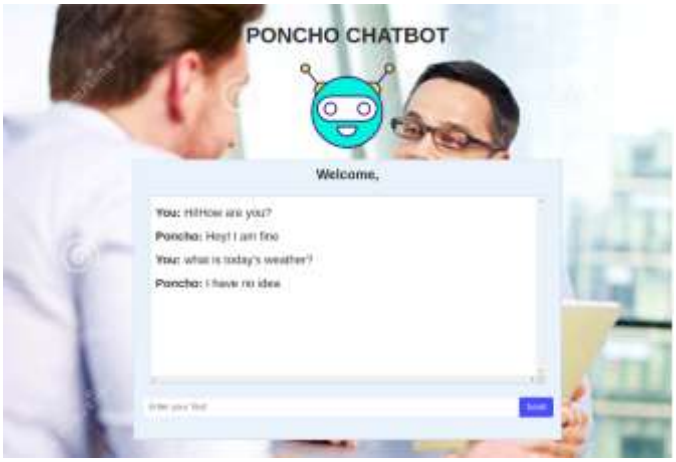

Fig -3: Sample responses generated by the chatbot

5. CONCLUSIONS

With the increase in the amount of data and computational power, humans have taken a huge leap towards giving computers the capability to do what a human can do. By building this chatbot we have managed to replicate human communication to a certain extent. Chatbots have not only found extensive usage in the customer care, finance, and healthcare industries to help executives understand the intent of the user and make business decisions but also to automate the entire system to effectively reduce cost. A chatbot can help customers make E-commerce purchases, monitor employees or customer's satisfaction.

Although chatbots are really powerful, they come with some problems that keep them from becoming the perfect virtual assistants. Some of these issues are easily solvable, for example figuring out how to integrate the chatbot into the app domain and in the personal domain. Other issues, like creating chatbots with human like intelligence are a lot more complex to solve. Chatbots are still far away from passing the Turing test [12]. This means that human intervention will be needed whenever chatbots fall through for quite some time.

We hope that this gives you some idea of the role that chatbots can feasibly play in the future of software and why there is no doubt that they are the future.

REFERENCES

[1] 8 Inspirational Applications of Deep Learning https://machinelearningmastery.com/inspirational-applications-deep-learning/.

[2] Chatbots approaches: Sequence-to-Sequence VS Reinforcement Learning https://medium.com/opla/chatbots-approaches-battle-part-1-sequence-to-sequence-vsreinforcement-learning-731d3db75e0e

[3] The implementation of Google's NMT in TensorFlow: https://github.com/tensorflow/nmt.

[4] Google's Neural Machine Translation System: Bridging the Gap between Human and Machine Translation. arXiv:1609.08144v2

[5] AI And Chatbots Are Transforming The Customer Experience. https://www.forbes.com/sites/shephyken/2017/07/15/ai-and-chatbots-are-transforming-thecustomer-experience/#44fa9d7c41f7. Accessed: 27-02-2020.

[6] Amazon Alexa is the home assistant you never knew you needed.

https://www.dailydot.com/debug/amazon-alexa/. Accessed: 27-02-2020

[7] What are Amazon Alexa skills? https://www.gearbrain.com/what-are-amazon-alexaskills-2471456002.html. Accessed: 27-02-2020.

[8] Apple's website about Siri: https://www.apple.com/siri/. Accessed: 27-02-2020.

[9] Dr. Richard Wallace's Github repository of ALICE: https://github.com/drwallace/aimlen-us-foundation-alice.

[10] James William Murdock IV's reply to "How does IBM's Watson work?" on Quora. https://www.quora.com/How-does-IBMs-Watson-work?share=1. Accessed: 27-02-2020.

[11] It's (not) elementary: How Watson works. https://www.pcworld.com/article/3128401/its-not-elementary-how-watson-works.html. Accessed: 27-02-2020.

[12] How Close Are Chatbots To Passing The Turing Test? https://chatbotsmagazine.com/how-close-are-chatbots-to-pass-turing-test-33f27b18305e. Accessed: 28-02-2020.