Integrated Smart Response System using Deep Learning

Spurthi N Anjan[1], Dr. G T Raju[2]

[1]*Computer Science and Engineering, RNS Institute of Technology, Channasandra, Bangalore, India*

[2]*Professor, Head, Computer Science and Engineering, RNS Institute of Technology, Bangalore.*

Abstract—The proposed system is an end-to-end method for automatically generating short responses, called Smart Reply. The model presents a computationally efficient machine-learned method for natural language response suggestion. It generates semantically diverse suggestions that can be used as complete responses with just one tap. The system is currently used in Inbox by Gmail and is responsible for assisting with 10% of all responses. The main aim is to design a flexible architecture that can learn representations useful for the tasks, thus avoiding excessive task specific feature engineering (and therefore disregarding a lot of prior knowledge).It is designed to work at very high throughput and process hundreds of millions of messages daily. The system exploits state-of-the-art, large-scale deep learning. The model is used to describe the architecture of the system as well as the challenges that was faced while building it, like response diversity and scalability. An optimized search finds response suggestions. The method is evaluated in a large-scale commercial e-mail application, Inbox by Gmail. Compared to a sequence-to-sequence approach, the new system achieves the same quality at a small fraction of the computational requirements and latency.

Keywords—Deep Learning, Machine Learning Algorithms, Prediction system, smart response

I. INTRODUCTION

Message is one of most popular modes of communications in today's digital world. Despite the recent increase in the use of social networks, messages remain the primary means of connection and also a mode of sharing information for billions of users around the world. With the rapid increase, processing and responding to incoming messages has become increasingly challenging for users. More number of people communicates with each other through their Digit appliance. Even in social media like WhatsApp, Face book, Instagram and twitter are few ones to name; people have their informal and public conversion on these platforms. With messages increasingly rapid, the processing of all the messages that come through has become difficult for users. Responding to the whole message is also very time-consuming. More specifically, could brief responses be suggested if necessary, just one tap away. Can we help users compose these short messages? It can be done with appropriate classification to assist users in automated responses, with correct classification and timely prioritization. The proposed methodology will have the provision to use unsupervised learning subroutines for accurate response prediction. In this methodology it would provide a concise, highly structured framework that would generate automatic responses to all incoming messages. It can also be used efficient to assist the user in managing messages.

Smart response provides automated responses, and like any AI powered feature, it can be correct at times or off the mark at different instances. It can bring smart reply functionality to chat/messages. Responses can be smart and your calendar information. Responses also trigger messages based on keywords. Any message such as "urgent" can trigger a sound to notify you of an important message, for example. The Messages app will generate short messages or smart responses automatically based on the conversation's recent messages. It's a feature that can save time. It uses the power of machine learning to suggest entire sentence as you start typing.

Smart Reply utilizes Artificial Intelligence (AI) and Machine Learning (ML) to offer auto-complete recommendations to enable to compose messages quicker. It utilizes the intensity of Machine Learning (ML) to recommend whole sentence as you begin composing.

II. LITERATURE SURVEY

One of the most reliable methods of communication is online messages. While reviewing any of the literature, its main focus should be on the research related to methodologies of message interaction along with the core techniques that can be used for an effective response system. With regard to the familiarity of the user with emails, there are five main activities that include the interaction of the user with an email system. These activities are Flow, Triage, Task Management, and Archive and Retrieve (Cadiz et al., 2001) [1] and are essential for understanding email process workflow. User interaction with a message system actually serves as a classification basis; as automated response must follow a similar algorithm interaction routine to generate a human like reaction. Based on their interaction message can be classified into two main categories which are Prioritize and Archives. Prioritizes tend to keep in check the message inflow and outflow, keeping control over their database, while archives save information for later use to

avoid missing important messages. Initial user class categorization helps to analyze the unique interactions between the two classes. Prioritizes managed to keep absolute control of their inbox while maintaining archived information for later use, ensuring that important messages were not missed.The studies showed that every day an average email user sends and receives emails to reach a point of overload and congestion of information. They proposed semantic web based approach to manage congestion, but for initial classification, the technique required manual annotations.

Messages are basically a collection of electronic text based words, and with machine learning methods it can be used for superior performance in classification of electronic documents. The application of artificial intelligence techniques can be used to build the architecture and used machine learning routines to support these changes in real-time targeting issues like reply prediction, attachment prediction and summary keyword generation.

In a review study by Jackson et al. (2012) on natural language processing techniques, authors investigated whether natural language processing techniques can be used to fully automate knowledge extraction from emails. This study reports on four generations of knowledge sharing building systems and discusses the effectiveness of knowledge extraction for these four generations (Jackson et al., 2012) [3].Machine learning techniques, data mining and natural language processing (NLP) work together to automatically identify patterns from electronic documents to assist in classifying them into intended categories (Almsman et al., 2012) [2].Classifiers of Naïve Bayes can be trained efficiently. The nature of the applied probability model characterizes performance there. A small data set of training is sufficient to estimate the statistics required to accurately classify and categorize. Information extraction and appropriate template allocation must be robust enough to use efficient probabilistic processes to handle a wide range of textual data.

Machine learning is used as predictive systems for providing better responses. Both Machine learning algorithms and RRN are being used more nowadays, since it's most reliable and robust in prediction systems that help provide a deeper insight into data. Using previous data, the Prediction systems will consider user data when processing information for the prediction of smart reply. To design a machine learning system that helps proposing Smart Response Reply, a new method and system for suggestion for automated response of messages and also to maintain an intelligent instant message response system, our proposed algorithm is formulated to provide an integrated instant message classification and template matching methodology. The next section presents a detailed discussion on algorithm architecture.

III. ARCHITECTURE

In this paper, the model is used to showcase the demonstrated implementation of a deep learning based smart response suggestion generator to enable user convenience on various interaction media, such as text messages, emails, and further uses in chatbot response helpers for support teams. The algorithm works on a pre trained NLP + LSTM model hosted in a cloud installation accessible by means of an authorized API.

The query from the user end is sent across via the API and an intent classifier generates the intent for the query under pre-defined classes, such as "Greet", "Assertion", "Question", "Activity", and follows. Post the intent classification, the query is passed through the pre trained model to generate suggested responses. Based on the confidence parameter for each response, we will rank the responses and send back through the API.

Natural Language Processing is used to understand computers, in way where computers understands and replies to a human using natural language. Ex: NLP is the engine behind Google Translate NLP can be used develop organization, speech recognition. It can also derive meaning from other foreign languages.

The model would use multilayer Long Short-Term Memory (LSTM) which is used to map the input sequence to the fixed dimensionality of the vector. Similar way another Long Short-Term Memory (LSTM) is used to decode the sequence from the vector. Each of the input would read at a single time to get a fixed representation of the vector's dimension and this would in turn used to obtain output vector. The second LSTM is a RNN model but it would be conditioned on the input sequence. The model would learn on the data and provides output very efficiently since there would be lag between the inputs and the output received.

The system is first trained using neural networks to process the incoming message and then generate the top probably responses. A neural network based LSTM model is setup to take the incoming message and classify it based on the intent and then generates probably responses.

At the most basic level, the sequence-to-sequence LSTM model is trained to take an incoming message I and the repository of all possible messages R, to generate a subset of applicable responses r in the following function -

$$r = FNP\ (R|o)$$

Where the limit of the function extends from 0 -> R

Model uses an encoder RNN in our LSTM model to encode phrases or blocks of words into representations and then use a decoder to generate the output natural language sentences.

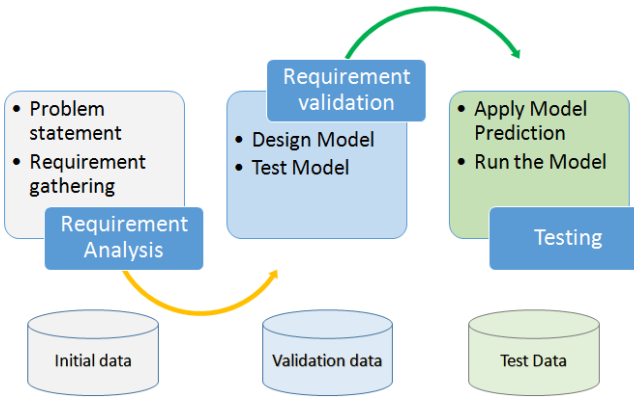

Figure 1 – Architecture Diagram

The model is trained against a huge dataset of vocabulary, consisting of the most frequently used English words and phrases. Additionally, including an extra recurrent projection layer in the neural network, improved the quality as well efficiency in terms of the time taken to converge.

IV. WORK FLOW

The Smart Reply framework interfaces a couple of recurrent neural systems, one used to encode an incoming messages, the other predicts the reply, to make an immediate cycle. The encoding system devours the expressions of the incoming message each one in turn, and delivers a vector (a rundown of numbers). This vector catches the substance of what is being said without getting hung up on style .The second system begins from this idea vector and blends a syntactically right answer single word at once, similar to its composing it out. Incredibly, the detailed operation of each system is totally adapted, just via preparing the model to anticipate likely reply.

A. Input Data

The data considered would be the pair of messages and the responses to it, where where X is an incoming message and Y is a Boolean true or false based on whether or not a message has replied or not. For the positive reply, only the messages that were replied to from a mobile device are considered. With the user's message, datasets for training and testing is created. The data is spilt into 90% and 10% which is used for both training and testing. The training data, validation data, testing data are unique.

B. Prediction Process

Pre-processed trained data would undergo a classification phase where each data is assigned a vector value using the algorithm. The Model consists of the data which has both incoming messages and reply to each. Language detection, Tokenization, Padding is done to maintain same length.

C. Result Data

In the training, the model process each pair of sentences is converted to tensors. The model would calculate the losses. The pair of sentence is given as an input to the model to predict the output. The model would calculate the losses for each true word and update respectively. While each pair of sentence is given has input, the highest value obtained would be considered to the output. Hence the model can compare the prediction with true sentence.

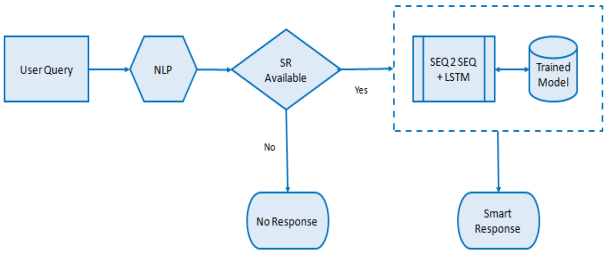

Figure 2 -Process Workflow

V. IMPLEMENTATION AND RESULTS

Step 1: User receives a text.

Step 2: The text is sent to the backend model for processing.

Step 3: The backend model preprocesses and encodes the message, creates a forecast and send a response.

Step 4: The API retrieves the data and follows reply retrieval policy to filter the best of 3 replies.

Step 5: User receives the response from the API.

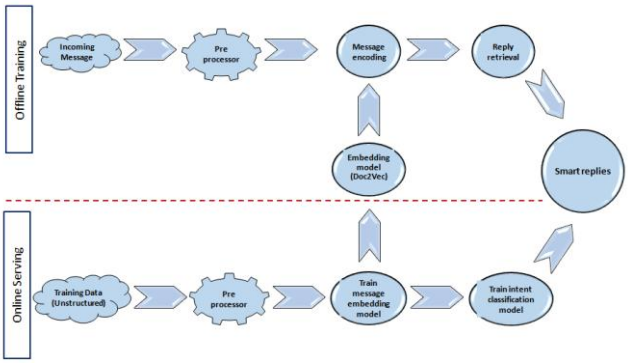

Figure3 – Flow Diagram

The above Flow diagram shows us how the system would work both on online services and Offline Training. The program is written in Python and implemented using Tensor Board. The above model is implemented and the results are displayed using a Web Interface.

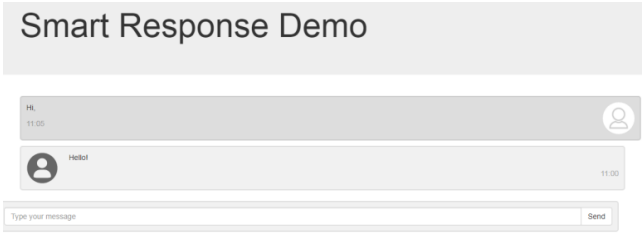

Figure 4 – Initial Screen

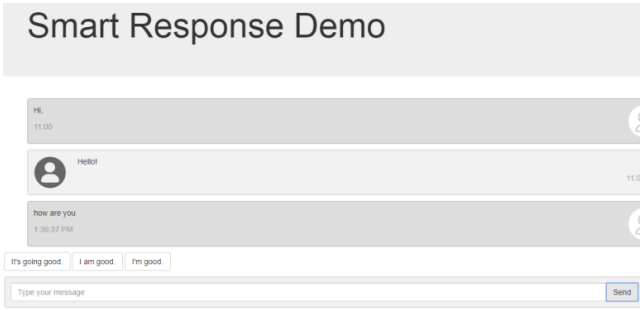

Figure 5 – Smart Reply

VI. CONCLUSION

Smart Reply is mechanically generating short and complete messages. The smart Reply model will predict full responses, given to any incoming message. Deep networks use totally different element for reading inputs and pre computing illustration of attainable responses and therefore the design allows an extremely economical runtime search.

Smart Reply application is evaluated mistreatment Live experiments with production traffic enabled a series of enhancements that resulted in an exceedingly system of upper quality other system. The mechanically generated response is entirely new or it may use the static message and conjointly embody appended content and phrases supported the delineated techniques. Sensitive knowledge is protected by permitting users to specify privacy levels with relevancy content, context and contacts.

As way as additional analysis is worried, the most aim is to implement this model in an exceedingly appropriate email application to check its efficaciousness by activity performance parameters e.g., Response accuracy, precise content filling, guide connation of associate degree economical response system. Several technical challenges within the dataset like abbreviations, writing system mistakes, many synonyms associate degreed equivocalness and propose an approach to supply solutions to the technical challenges

ACKNOWLEDGEMENT

I would like to thank Dr. G. T. Raju, Vice Principal, Prof. and Head, Department of CSE, RNSIT, Bangalore for his valuable suggestions and expert advice throughout this paper.

REFERENCES

[1] Cadiz, J.J., L. Dabbish, A. Gupta and G.D. Venolia, 2001. Supporting email workflow. Microsoft Res

[2] ALmomani, A., T.C. Wan, A. Altaher, A. Manasrah and E. ALmomaniet al., 2012. Evolving fuzzy neural network for phishing emails detection. J. Computer Sci., 8: 1099-1107. DOI: 10.3844/jcssp.2012.1099.1107.

[3] Jackson, W.T., S. Tedmori and H. Hinde, 2012. The boundaries of natural language processing techniques in extracting knowledge from emails. J. Emerg. Technol. Web Intell., 4: 119-119.

[4] McCallum, A. and K. Nigam, 2003. A comparison of event models for naïve bayes text classification. J. Mach. Learn. Res

[5] MATTHEW HENDERSON, RAMI AL-RFOU, BRIAN STROPE, YUN-HSUAN SUNG,L ´ ASZL´O LUKA´CS, RUIQI GUO, SANJIV KUMAR, BALINT MIKLOS, And RAY KURZWEIL, For Efficient Natural Language Response Suggestion For Smart Reply

[6] Al-Alwani, A. (2015). Improving email response in an email management system using natural language 507 processing based probabilistic methods. Journal of Computer Science, 11(1):109.

[7] Kannan, A., Kurach, K., Ravi, S., Kaufmann, T., Tomkins, A., Miklos, B., Corrado, G., Luka´cs, L., Ganea, 523 M., Young, P., et al. (2016). Smart reply: Automated response suggestion for email. In Proceedings of 524 the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pages525 955–964. ACM.

[8] I. V. Serban, A. Sordoni, Y. Bengio, A. Courville, and J. Pineau. Building end-to-end dialogue systems using generative hierarchical neural network models. In Conference on Artificial Intelligence. AAAI, 2016.