ML Powered Text Auto-Completion and Generation

Dr.P.Kayal
BVRIT Hyderabad
Hyderabad, India
Kayal.p@bvrithyderabad.edu.in

Jilla Nikhitha
BVRIT Hyderabad
Hyderabad, India
nikhithajilla@gmail.com

Parvathaneni Pooja Sri
BVRIT Hyderabad
Hyderabad, India
poojasriparvathaneni@gmail.com

Gottipati Pavithra
BVRIT Hyderabad
Hyderabad, India
pavithragottipati2501@gmail.com

Chunduru Sai Meghana
BVRIT Hyderabad
Hyderabad, India
meghanachunduru22@gmail.com

***Abstract*-A branch of artificial intelligence called "natural language processing" (NLP) helps computers comprehend human language. It combines the strength of linguistics and computer science to think about language rules and grammar and create intelligent systems capable of understanding, dissecting, and isolating meaning from text and speech. Using this concept, the project proposes a system which helps in minimizing the human effort by providing the features like text auto-completion and generation. Auto-completion is the process of completing a word, or a phrase, as we type in a document. The prediction is based on the most likely word among a set of frequently used terms. The text prediction task involves editing text with the fewest number of keystrokes possible. This method is suggesting words that the user intended to write, and the system predictsthe next word related to the previous work. Text generation is a feature which is an enhanced version of Gmail's Smart Reply. This feature helps in generating a template of the mail by classifying the subject line. The goal of our intelligent system is to assist differently abled persons by increasing their typing speed and decreasing the number of keystrokes required to complete a word or a sentence. The suggested method is also very useful for multi-domain professionals who compose content, particularly those who frequently employ extensive, difficult-to-spell language that may be technical or medical in nature, such as medical doctors.**

***Keywords*- NLP, Gmail's Smart Reply, LSTM, RNN, Deep Learning, Text Auto-completion, Text Generation.**

I. INTRODUCTION

In our daily life, we have numerous mediums of communication. There is a fax, telex, text messaging, cell phones, etc. But still, email plays a major role in the communication medium. Email provides us with a wide range of communication across the globe, just a simple internet connection we can communicate with someone who is on the other side of the planet also. Email technology is simple and easy to use, mainly affordable it is a free communication medium. Email not only provides us the communication but it also ensures that there is strict privacy. As Gmail is popular form of communication, we are proposing two features that is Text Auto-completion and Text Generation. These two features would ensure that it is more user convenient. In todays busy world we require a model that would help us in instant messaging. Hence forth our model would be supporting the instant messaging criteria. That would be saving lots of time and effort of user.

II. LITERATURE SURVEY

Yonghui Wu , Benjamin N. Lee, Gagan Bansal, Yuan Cao, Justin Lu, Timothy Sohn, Jackie Tsay, Yinan Wang, Andrew M. Dai, Zhifeng Chen,Shuyuan Zhang, and Mia Xu Chen suggested a unique approach for producing interactive, real-time Gmail suggestions. Smart Compose [1] is a feature that helps users write emails by preventing repetitive typing. During the design and execution of such a complicated and large-scale system, they met a variety of challenges, including model selection, performance evaluation, servicing, and other operational obstacles. Smart Compose is built around a huge neural language model. They created a one-of-a-kind service infrastructure for high-throughput and real-time inference and applied cutting-edge machine learning algorithms for language model training, resulting in high-quality recommendation prediction. Experiment results demonstrate the efficiency of their suggested system architecture and deployment strategy. Gmail is being used to provide this system.

, Gokhan Tur, Yue Weng, Franziska Bell, and Huaixiu Zheng introduced a significant enhancement called One-click-chat (OCC) to the Uber in-app chat system [2]. OCC incorporates smart answers to enable driver-partners to promptly responds to message from riders. This feature utilizes machine learning techniques to select appropriate smart replies based on the content of the conversation. Unlike traditional smart reply systems, OCC mainly two important parts: intent recognition and reply retrieval. It was specifically designed for mobile applications that require concise and non-standard

notifications. The system pairs intents with corresponding replies based on the popularity of chat messages derived from previous data. Various embedding and classification methods for intent detection were tested, and an unsupervised distributed embedding with a nearest-neighbour classifier was chosen for implementation. This approach is highly scalable and requires only a small amount of labelled training data. It is also simple to design and deploy in production, similar to deep learning designs, such as the word-level convolutional neural network, are comparable. Overall, the system detects intent with a high accuracy rate of 76%. This technique is currently being used in English-speaking countries, where smart replies are used in 71% of in-app talks between passengers and driver partners to enhance efficient communication.

A personalized chat bot[3], proposed by Rajeev Gupta, Ranganath Kondapally, and Chakrapani Ravi Kiran, aims to have the same personality as its associated user. The key feature of this chat bot is its ability to respond in a manner consistent with the user's style when interacting with other people's messages. Previous approaches to giving chat bots personalities relied on explicit user profiles containing persona descriptions or personal data. However, in real-world scenarios, users may be reluctant to provide detailed persona descriptions, and gathering explicit user profiles requires significant manual effort. To address this issue, the researchers introduced IMP Chat, a retrieval-based personalized chat bot paradigm that infers an implicit user profile from the user's chat history.

The implicit user profile, according to this approach, is more flexible and accessible compared to explicit profiles. IMP Chat leverages the user's prior responses to construct language models that capture the user's unique speaking style. It also explores the conditional relationships between each user post-response pair to predict their customized preferences. When aggregating personalized preferences, more weight is given to historically relevant pairs that are topically connected to the current query, as they are dynamic and context-sensitive. Extensive experiments on two large datasets demonstrate that the proposed method outperforms all baseline models. The approach matches response candidates with personalized language style and preferences, combining the two matching signals to determine the final ranking score.

III. EXISTING SYSTEM

A smart reply system is a tool that automatically generates brief and contextually appropriate responses to emails or messages. By utilizing machine learning algorithms and natural language processing techniques, it analyses incoming messages and suggests suitable replies. These systems are commonly found in various communication platforms like chat bots, messaging apps, and email clients. They offer users pre-written response options that are concise and tailored to reflect the style and meaning of the original message. Smart reply systems have gained popularity due to their ability to enhance productivity and streamline communication processes. They eliminate the need for users to type out complete responses, resulting in faster interactions and more efficient communication. However, it's important to note that while smart reply systems can be helpful, they may not always capture the intricacies of complex conversations and may not be appropriate in all situations.

IV. PROPOSED SYSTEM

A text generation system is a type of software or model designed to generate human-like text in response to prompts or input. Machine learning algorithms and natural language processing (NLP) techniques are employed to understand and produce coherent and context ually relevant writing.

An efficient tool for automating and streamlining email composition is a text generation system tailored for Gmail. This system takes the subject line as input and generates the body of the email. By leveraging NLP and machine learning, such a system can produce well-structured and contextually appropriate content based on the provided subject line.

Using this approach, users simply need to provide the subject line, and the text generation model will generate a paragraph or complete email body. This can be particularly helpful for users who frequently send similar types of emails or those who need assistance in crafting their messages. By utilizing this system, users can save time and effort in composing emails.

V. MODULES

A. Text Auto-Completion

The user interface feature called "auto-complete" is a convenient tool that suggests words or phrases to users as they type, saving them from having to type the entire text manually. It is a common feature found in modern applications and serves several purposes. Auto-complete aims to speed up typing, assist individuals with typing difficulties, prevent spelling errors, and facilitate information retrieval by providing available options. The concept of auto-complete dates back to the Reactive Keyboard introduced by Witten and Daraghs in 1983, and although various approaches have emerged since then, the core idea remains the same. This feature can be found in word processors, programming editors, web browsers, email clients, web applications, mobile phone interfaces, Unix terminals, and more. The usefulness of auto-complete systems depends on the specific context and how they are utilized. While they can greatly assist in situations such as source code editing, technical manuals, and legal papers, they may be less effective for general natural language input. Many mobile devices incorporate auto-complete features

that expedite typing through predictive algorithms and the redundancy present in languages.

B. Text Generation

Text generation, also known as natural language generation, is the process of creating text that resembles human-written material. It falls under the subfield of natural language processing (NLP) and utilizes artificial intelligence and computational linguistics to automatically generate documents in natural languages for specific communication purposes. This field has been evolving since the 1970s and is considered a part of NLP. Ongoing efforts in NLP involve developing deep learning models, such as generative adversarial networks (GANs), which consist of a generator and a discriminator and are used to generate artificial text outputs. One application of text generation is in email communication, where users provide the subject line and the model generates the content of the email based on that subject. Text auto-completion and text generation work together to enhance user convenience in email messaging, with the subject line guiding the body of the email and auto-complete predicting and completing words. These functionalities enable efficient and instant messaging through emails.

VI. ARCHITECTURE

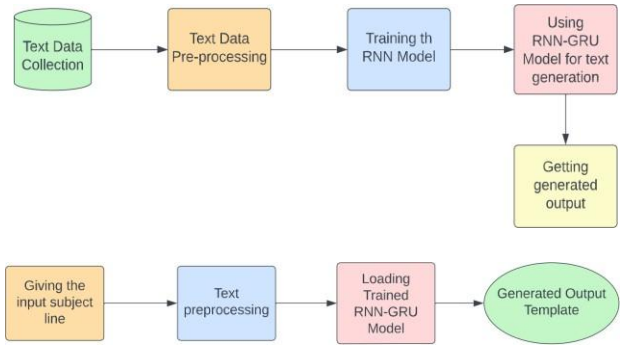

Fig. 6.1. Text Generation Project Flow

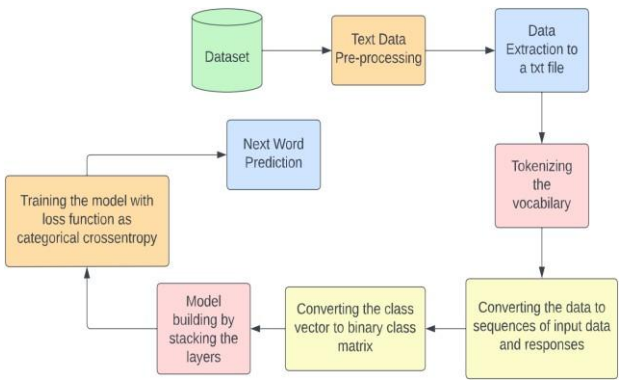

Fig. 6.2. Text Auto-Completion Project Flow

VII. ALGORITHM

A. Recurrent Neural Networks

A recurrent neural network (RNN) is a form of artificial neural network that is specifically built to process sequential data, such as time series or natural language. RNNs excel at capturing patterns and dependencies within sequential data by maintaining internal states or memories.

One key feature of RNNs is their ability to retain information from previous time steps and incorporate it into the current step. This enables the network to learn from the entire sequence, rather than solely relying on the current input, and make predictions based on this accumulated knowledge. Another advantage of RNNs is their flexibility in handling inputs of varying lengths, allowing them to effectively model the temporal dynamics of the data.

In an RNN, each step in the sequence corresponds to a specific time step, and recurrent units, often in the form of memory cells, play a crucial role in updating the network's internal state based on the input at each time step. This allows information to flow across the network and be propagated through different time steps, as each step's output can be feeded back as input to the subsequent step.

B. Long Short-Term Memory

LSTM architecture is designed to overcome the vanishing gradient problem often encountered in recurrent neural networks (RNNs). LSTMs excel in tasks that involve understanding and modelling the context and temporal dynamics of sequential data. The key feature of LSTM networks is their memory cell, which is responsible for storing and updating data over multiple time steps. It consists of three essential components: the input gate, forget gate, and output gate. These gates control the flow of information into and out of the memory cell, enabling selective retention or forgetting of information.

The input gate controls how much fresh data is stored in the memory cell. Based on the current input and the previous hidden state, it generates a value between 0 and 1, signifying the amount of new information to be preserved. The forget gate determines how much of the existing memory cell content should be discarded. It accepts the current and previous hidden states as inputs and returns a value between 0 and 1 indicating the amount of knowledge to forget..

The output gate governs the amount of data to be read from the memory cell. It scales the contents of the memory cell and passes them as the current hidden state, which is used for the final prediction. The output gate considers the current input and the previous hidden state in its decision-making.

VIII. RESULTS

Suggestions generated while typing the subject line

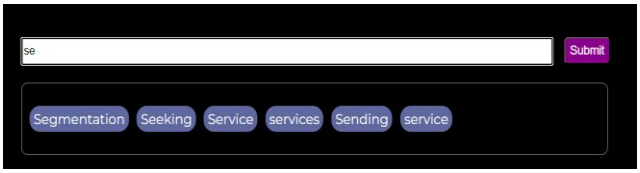

Fig. 8.1 Text Auto-Complete Suggestions

Template Generation

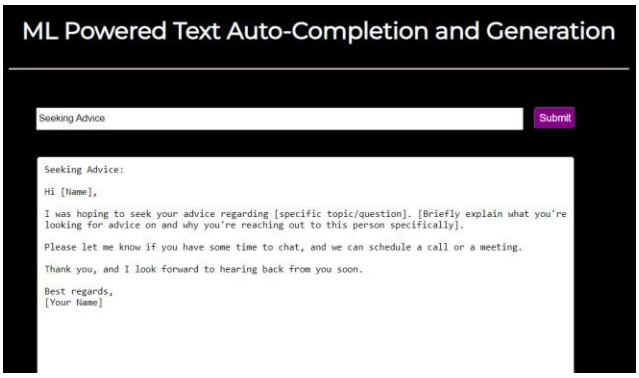

Fig. 8.2. Template generation as per the input

Next Word Prediction in the body

Fig. 8.3. Next Word Prediction in the result body

IX. CONCLUSION

To sum up, our ML-based approach for text auto-completion and generation offers significant advancements in the field of natural language processing. By combining deep learning techniques, fine-tuning, and personalized training, we have developed a powerful algorithm that produces authentic and relevant text output. The versatility and potential applications of this algorithm make it a valuable tool across various domains, allowing users to efficiently generate coherent and contextually suitable text content.

X. FUTURE SCOPE

The future holds great promise for ML-powered text auto-completion and generation. There are several areas that require further development to enhance the contextual understanding, personalization, and adaptability of these systems. This includes incorporating multiple modalities, addressing concerns related to bias and ethics, creating interactive and transparent systems, enabling real-time collaboration, and developing domain-specific applications. These advancements will lead to more accurate, flexible, and adaptable text generation systems that can seamlessly integrate into various domains and applications.

REFERENCES

[1] Mia Xu Chen, Benjamin N Lee, Gagan Bansal, Yuan Cao, Shuyuan Zhang, Justin Lu, Jackie Tsay, Yinan Wang, Andrew M. Dai, Zhifeng Chen, Timothy Sohn, and Yonghui Wu. 2019. “Gmail Smart Compose: Real-Time Assisted Writing”. In The 25th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD ’19), August 4–8, 2019, Anchorage, AK, USA.

[2] Yue Weng, Huaixiu Zheng, Franziska Bell, and Gokhan Tur. 2019. “OCC: A Smart Reply System for Efficient In-App Communications”. In Proceedings of The 25th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Anchorage, AK, USA, August 4–8, 2019 (KDD ’19).

[3] Rajeev Gupta, Ranganath Kondapally, Chakrapani Ravi Kiran. “Impersonation: Modeling Persona in Smart Responses to Email”. IJCAI 2018 conference. Workshop on Humanizing AI.

[4] Shao T., Chen H., Chen W. “Query auto-completion based on word2vec semantic similarity”. J. Phys. Conf. Ser. 1004(1), 12–18 (2018).

[5] Aaron Jaech and Mari Ostendorf. 2018. “Personalized Language Model for Query Auto-Completion”. In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics.