Designing and Implementing Conversational Intelligent Chat-bot Using Natural Language Processing

Asoke Nath*[1], Rupamita Sarkar[1], Swastik Mitra[1], Rohitaswa Pradhan[1]

[1]Department of Computer Science, St. Xavier's College (Autonomous), Kolkata, India

ABSTRACT

In the early days of Artificial Intelligence, it was observed that tasks which humans consider 'natural' and 'commonplace', such as Natural Language Understanding, Natural Language Generation and Vision were the most difficult task to carry over to computers. Nevertheless, attempts to crack the proverbial NLP nut were made, initially with methods that fall under 'Symbolic NLP'. One of the products of this era was ELIZA. At present the most promising forays into the world of NLP are provided by 'Neural NLP', which uses Representation Learning and Deep Neural networks to model, understand and generate natural language. In the present paper the authors tried to develop a Conversational Intelligent Chatbot, a program that can chat with a user about any conceivable topic, without having domain-specific knowledge programmed into it. This is a challenging task, as it involves both 'Natural Language Understanding' (the task of converting natural language user input into representations that a machine can understand) and subsequently 'Natural Language Generation' (the task of generating an appropriate response to the user input in natural language). Several approaches exist for building conversational chatbots. In the present paper, two models have been used and their performance has been compared and contrasted. The first model is purely generative and uses a Transformer-based architecture. The second model is retrieval-based, and uses Deep Neural Networks.

Keywords: Natural Language Processing, Natural Language Understanding, Natural Language Generation, Deep Neural Networks, Artificial Intelligence, Transformer Model, Intelligent Agent, Chatbot.

I. INTRODUCTION

Natural Languages are those that came into existence for the purpose of being used by humans in order to interact with one another. Such languages have evolved alongside humans, without any conscious planning. **Natural Language Processing** resides at the overlap between Linguistics, Computer Science and Artificial Intelligence and involves understanding, analysing, processing and manipulating Natural Language data.

Building a Conversational Bot using Artificial Intelligence is a problem in the field of Natural

Language Processing, whose aim is to construct an intelligent agent which is able to hold a conversation in human understandable language, with another human or in some cases, another intelligent agent.

The field of conversational chatbots can prove to be a gateway for machines to understand humans and human connections better, since language is one of our largest tools of communication and for improvement in quality of conversation it is a must for a chatbot to master a language and explore its depths. Chatting experience can further be enhanced by enabling text-based sentiment analysis in a chatbot model which shall help the bot to better understand the sentiment and feelings behind a user message.

Chatbots are currently seeing extensive implementation in online customer services, appointment booking (doctor's appointment, business appointments etc.), providing psychological counselling, and question answering. Bots like Facebook messenger chatbot allows businesses to communicate effectively with their customers.

Such chatbots can range from simple domain-specific question answering bots to more random chatter bot which can even simulate personality traits such as empathy, sense of humour, compassion etc.

II. LITERATURE REVIEW

The generative chatbot built for this paper uses a transformer architecture and the Retrieval based bot uses a Deep Neural Net architecture. A generative Transformer model takes each word of an input sequence, processes them in parallel and outputs the probability of the next word in the output sequence. The Transformer model implemented in this paper is based directly on the paper by Vaswani et. al., 2017 [1], which first brought transformers into existence. The

Transformer model discussed in this paper had been trained on the WMT 2014 English-German dataset, which is an English to German translation dataset consisting of about 4.5 million sentence pairs and on the significantly larger WMT2014 English-French translation dataset consisting of 36M sentences and split tokens into a 32000 word-piece vocabulary. This is also the base model for many architectures performing different NLP tasks, including building chatbots. This model does quite well on Neural Machine Translation tasks, with a **BLEU** score of 41.0 on the big model trained on English-French dataset. BLEU score is a popular metric in NLP for evaluating the quality of a generated sequence by computing the number of n-grams in the machine generated sequence matching against the **n-grams** in the target sequence provided by a human. An n-gram here is a n-word ordered set which is a subsequence of any given sequence. In the Neural Machine Translation task, a model learns to relate words/phrases of the input language to words/phrases of the target language. When it comes to building a chatbot, this particular base model needs to be altered to do the job. The *Blenderbot* (Roller et. al., 2020) [2] uses the transformer of [1] as its base model and builds three different architectures on top of this base model: a **retrieval-based** architecture, a **generative** architecture and a **retrieve-and-refine** model. The retrieval-based architecture of *Blenderbot* uses two poly-encoder models (Humeau et. al., 2020) [3] which is a pre-trained architecture with an additional pre-trained attention mechanism for increasing the number of global feature representations. Also, it uses a ranking system based on cross-entropy loss of one correct sample and multiple negative samples. The generative model is inspired by **ParlAI** (Miller et. al., 2017) [4], which comprises multiple **tasks** and **agents** where the tasks are APIs representing various problems and the agents are applied to solve these

tasks. The tasks can be of type conversation logs, online or reinforcement learning tasks, and also real language and simulated tasks. ParlAI is based on the three main concepts of **world-** this is an environment which can have two conversing agents and it can also be an environment with multiple conversing agents; **agent-** can be a human, a hard coded bot or a system trained with machine learning and an agent can speak i.e., converse; **teacher-** an entity that converses with the learner to help it learn to speak. For decoding from the set of output word probabilities and deciding on the output word in *Blenderbot*, the **Beam Search** decoding has been experimented. In Beam Search, instead of always choosing the word with highest probability (greedy search), a beam length **b** is decided. For the first output word, the **b** highest probability words are chosen. For each of these **b** first words, **b** words are chosen such that the combined probability of the previous and the new word is amongst the **b** highest probability combinations. This goes on until all the possible word sequences reach the maximum number of words allowed in the output, at which point, the word sequence with the highest probability (product of the probabilities of each constituent word) is selected for output. For sampling from decoder outputs, **top-k** and **search and rank** approaches have been tried out. The **top-k** approach has been used in the Transformer model chatbot discussed in this paper and the top-k method was used in **Holtzman et. al., (2020)** [14]. This paper comes up with the **top-k** sampling approach while trying to discuss a very common problem faced by purely generative text generation models and this problem is called **Neural Text Degeneration**, and it entails the repetition of the same word or phrase in the output sequence, such that the frequency of this repetition makes the sequence look like an incoherent loop. The top-K approach randomly samples an output word from a set of **k** highest

probability words. This reduces the chance of text degeneration due to repetition quite a bit.

III. OUR PROPOSED ALGORITHM

This paper discusses two algorithms: **the transformer algorithm for purely generative chatbot** and the **Deep Neural Net algorithm for retrieval-based bot**.

A. Transformer:

The transformer model uses two different data sets, the first is the **Cornell Movie Dialog Corpus**. The next dataset is a **custom dataset created from multiple famous books and novels** (Including Sherlock Holmes, The Shining, The Picture of Dorian Grey etc). This custom dataset has been used to increase the size of the training dataset as well as for providing well-structured text generation examples for the Transformer.

The transformer model is based on the model discussed in the paper: Attention is All You Need **(Vaswani et. al., 2017)**. The basic modules of this architecture are briefed as follows:

1) Preprocessing: The sequences with less than 5 words are removed and all words that occur less than 50 times in the individual datasets are also removed. The sequences are then tokenized and padded.

2) Word Embedding: Word embeddings have been created with a dimension **d_model = 128**. This dimension has been maintained for the parameter matrices throughout the architecture. The code snippet in Fig. shows the creation of word embedding in Tensorflow.

3) Positional Encodings: Positional embeddings are used for capturing information regarding the absolute position of a word in a sequence as well as the relative position of each word with respect to the other words in the

sequence. Sine and Cosine waves have been used for capturing the positional information of even and odd positions respectively.

4) Multi Headed Self Attention: The task of the self-attention layer is to find out the relationship of each word in a sequence with all the other words in that sequence. The input to the self-attention is the positional embedding matrix (in case of encoder) and the masked positional embedding matrix (in case of decoder). Three copies of the input are made. What it does is take a single word of a sequence and computes the degree of relevance (attention) of every other word in the sequence with that word. A single self-attention mechanism is called a head. The transformer model uses multiple such self-attention heads in both encoder and decoder.

5) Look Ahead Mask: The solution to this is the **look-ahead mask** that creates a matrix which is to be added to the output of every self-attention head. This mask is designed in such a way that for every word (with respect to which self-attention is computed at any instant), all the words coming after it in the sequence will be masked out.

Total trainable parameters: 1,539,157

B. Retrieval Based Bot:

The authors created the intents dataset for this bot and preprocessed it using the *Natural Language ToolKit* (NLTK). The retrieval-based bot has its data in JavaScript Object Notation (JSON) format stored in a file called **conversation_intents.json**.

```
transformer_model.summary()
```

```
Model: "transformer"

Layer (type)                  Output Shape           Param #      Connected to
==================================================================================
enc_inputs (InputLayer)       [(None, None)]         0

dec_inputs (InputLayer)       [(None, None)]         0

lambda_3 (Lambda)             (None, 1, 1, None)     0            enc_inputs[0][0]

encoder_stack (Functional)    (None, 25, 128)        701312       enc_inputs[0][0]
                                                                  lambda_3[0][0]

lambda_4 (Lambda)             (None, 1, 1, None)     0            dec_inputs[0][0]

lambda_5 (Lambda)             (None, 1, 1, None)     0            enc_inputs[0][0]

decoder_stack (Functional)    (None, 25, 128)        834432       dec_inputs[0][0]
                                                                  encoder_stack[0][0]
                                                                  lambda_4[0][0]
                                                                  lambda_5[0][0]

final_outputs (Dense)         (None, 25, 3413)       440277       decoder_stack[0][0]
==================================================================================
Total params: 1,539,157
Trainable params: 1,539,157
Non-trainable params: 0
```

Figure 1. Transformer Model Summary

The various intent classes in the dataset include: **purpose**, **feelings**, **family**, **politics**, **suicidal**, **sadness**, **covid** etc. There are a total 38 intents in the dataset. The model used for this bot is a Deep Neural Network model with alternating Dense and Dropout layers. All the non-final Dense layers use ReLU activation and the final Dense layer uses Softmax activation function.

It is a classification model that classifies the user's input into one of the 38 intent classes, following which a random response is selected from the set of predefined responses. This randomly chosen response is then displayed as the chatbot's reply.

Total Trainable parameters: 51,174

IV. RESULTS AND DISCUSSION

Transformer:
- **epochs** - 10,
- **loss (sparse categorical cross-entropy)** - 3.8488,
- **accuracy (sparse categorical accuracy)**- 0.1979

Retrieval Based bot:
- **epochs** – 200,
- **loss** - 0.2081,
- **accuracy** – 0.9142

i used to rule the world

sure as door this himself before take her every find we ever other house many

seas would rise when i gave the word

hands find had room always might all and under most but think life such no

one-minute i held the key then the walls were closed on me and i discovered that my castles stand upon pillars of salt and pillars of sand

yet through any day at work always made their house right little found now against

her eyes and words are so icy oh but she burns like rum on fire . hard and fast and angry as she can be i walk my days on a wire

what turned how you home might something time never ever it's other her place last

Conversation of Purely Generative Transformer Bot where the normal fonts and messages from user and the bold fonts are replies from bot

```
You: Hello there
Marnie: Hello, please tell me your name user
You: This is Beckham
Marnie: OK! Hola HUMAN, how can I help you?
You: How many senses do you have?
Marnie: I am a nihilistic bot
You: Who built you?
Marnie: A few jobless and depressed postgrad students built me.
You: What is your model?
Marnie: A retrieval based model built with encoder-decoder.
You: Do you understand what I am saying?
Marnie: Well I would not be a very clever AI if I did not would I?
You: Tell me a joke then
Marnie: Man: (to friend) I'm taking my wife on an African Safari. Friend: Wow! What would you do if a
vicious lion attacked your wife? Man: Nothing. Friend: Nothing? You wouldn't do anything? Man: Too ri
ght. I'd let the stupid lion fend for himself!
You: Can you prove you are self-aware?
Marnie: That is an interesting question, can you prove that you are?
```

Figure 2. questions that test the *self-awareness* of the bot

```
Marnie: I am a work in progress.
You: Who is the Prime Minister of UK?
Marnie: Good! Hi HUMAN, how can I help you?
You: What is the capital of Germany?
Marnie: OK! Hola HUMAN, how can I help you?
You: Who is the winner of Major League Baseball?
Marnie: Good! Hi HUMAN, how can I help you?
You: Which nation won the maximum Olympic gold medal?
Marnie: Great! Hi HUMAN! How can I help?
You: Do you know Ellon Musk?
Marnie: Coronaviruses are a large family of viruses that are known to cause illness ranging from the
common cold to more severe diseases such as Middle East Respiratory Syndrome (MERS) and Severe Acute
Respiratory Syndrome (SARS)
You: This is wrong
Marnie: I will always lend you an ear , and a shoulder
```

Figure 3. Some imperfect replies from the bot

```
GO! Bot is running!
You: Hello
Marnie: OK! Hola HUMAN, how can I help you?
You: The user is Rahul
Marnie: Good! Hi HUMAN, how can I help you?
You: Can you tell me something about COVID-19
Marnie: Coronaviruses are a large family of viruses that are known to cause illness ranging from the
common cold to more severe diseases such as Middle East Respiratory Syndrome (MERS) and Severe Acute
Respiratory Syndrome (SARS)
You: Is there any vaccine for Corona Virus?
Marnie: Yes. There are three COVID-19 vaccines for which certain national regulatory authorities have
authorized the use. None have yet received WHO EUL/PQ authorization but we expect an assessment on th
e Pfizer vaccine by the end of December and for some other candidates soon thereafter.
You: Can you give me more information about symptoms of COVID-19?
Marnie: The most common symptoms are Fever, Dry Cough, Fatigue. Other less common symptoms are Loss o
f taste or smell, Nasal Congestion, Headache, Nausea, and Muscle pain. Some of the sever symptoms are
Shortness of breath, Loss of appetite, High Temperature, Persisten pain in chest.
```

Figure 4. Bot's awareness about COVID-19

C.   Reason for the Performance Difference

There is an obvious difference in the quality of conversation of the Generative Chatbot and the

Retrieval Based Chatbot. The Retrieval based bot is superior when it comes to providing grammatically correct and meaningful replies. The replies are meaningful with respect to the user's message so long as the bot can successfully classify the intent of the user message. On the other hand, the generative chatbot is uttering unintelligible sentences. The reason for this is given in the comparison table below:

1) The **purely generative Transformer Bot** literally **generates the output sequence one word at a time, using no outside help.** All it has is the input sequence and its own internal mechanism to rely upon. This is why a purely generative chatbot falls under the category of **Language Models**, which are models that predict the probability of each word in the

sequence. Chatbots as well as other text generators fall under this category.

Retrieval based bots are provided with a set of intents which serve as topics on which the chatbot is capable of holding conversation. For each topic there are a set of sequences which serve as the examples of the kind of user messages the bot might expect and using these the **bot is trained to classify the user message under the correct intent.** So the task of this chatbot is not to generate an output sequence word by word, but to figure out which topic the user message falls under. **Once classified, it randomly selects one of the reply strings from the set of reply strings provided for that intent and outputs it.**

```
model.summary()

Model: "sequential"

_____
Layer (type)            Output Shape             Param #
=================================================================
dense (Dense)           (None, 128)              40448

_____
dropout (Dropout)       (None, 128)              0

_____
dense_1 (Dense)         (None, 64)               8256

_____
dropout_1 (Dropout)     (None, 64)               0

_____
dense_2 (Dense)         (None, 38)               2470
=================================================================
Total params: 51,174
Trainable params: 51,174
Non-trainable params: 0
```

Figure 5. Model Summary of Retrieval Based Deep Neural Net Bot

2) **The final layer of the generative chatbot has to compute the probability of each word in the vocabulary of the corpus.** If all words are kept, this number can reach well over **20,000**. For this model, the authors eliminated all words below a predetermined threshold frequency and there

were still **3412 words.** So **the final Dense layer of the model had 3412 units and had to compute the probability for each unit** i.e., probability for the word that unit represents. This adds to the difficulty of making such predictions.

The final layer of the retrieval based bot has to compute probability for **38 intents classes**. This **significantly reduces the number of parameters** needed and also makes it easier for the model to learn to classify correctly.

3) A **transformer model** has different kinds of modules with different functionalities. The SOTA (State Of The Art) Transformer models that are used for text generation and as chatbots all have a huge number of parameters. **Facebook's Blenderbot for example have variations that have been pre trained using 256M parameters and 622M parameters [1]**. To parallelize computation on such a huge volume of parameters (**tremendously large matrices**), it requires **memory and computational resources like multiple distributed GPU or TPU** that at this moment only the largest corporates can support and even then such bots are often a hybrid of retrieval based and generative. **The transformer bot built in this paper has less than 2M parameters (1,539,157 to be precise, with 6 layers for each Encoder and each Decoder Stack; 8 heads for each of the two multiheaded attention and multiple, dense, dropout and normalization in both Encoder and Decoder)**, and **trained on a single remote TPU** provided by google colab. So it is fair to say that **the resources available for this model do not allow it to reach its full potential but at the same time the Transformer Chatbot built for this project demonstrates the functioning of such models.**

The Retrieval based bot is built using a sequential model with 51,174 parameters and 6 layers. The model is simpler and smaller and the considerably less number of parameters makes it **trainable on a CPU**. As a result, the model learns fast with very little demand for resources.

4) To save time, it was necessary to cut short the size of the training dataset of the **Generative Transformer Bot**. **The number of training examples in our final corpus was 63846**. Even then the time it **took for a single epoch to complete was 380s i.e., 6 minutes**. We have limited the **total number of epochs** of the final model to **10** and the **total time taken** to train the model for 10 epochs was **a little over 1 hr. When trained on the entire dataset (size of which is over 2,000,00), a single epoch took over 1 hr**. The size of our dataset was very small compared to even those Transformer models that are considered as "small". For example, **the size of our data corpus is 6 MB** whereas a small RoBERTa-like model (which is a special variant of the Transformer) needs 3 GB sized corpus to train on [13]. The reason for requiring this huge dataset is that this particular chatbot is to be trained to understand human language and generate text on its own and from scratch. This is the same as a human child learning to build comprehensive and coherent sentences on its own. While a SOTA model still takes way less time than a human child needs to learn to generate meaningful texts, it still requires too many resources which is very hard to provide for students/practitioners/scholars.

The Retrieval based bot was trained with 200 epochs and each epoch took less than a minute to complete. It does not need to learn to generate text from scratch. It only needs to learn to classify the user's message into one of the classes available to it, based on the word patterns present in the user message. **For a Retrieval based model it is important to have a good variation of user messages for each class and to be very particular when deciding which user message goes under which intent class, so as to minimize the chance of misclassification. A Retrieval based model is capable of holding meaningful conversations even when trained with very few parameters and on minimal computational and memory resources, provided**

that the Training dataset is well planned and the intents and patterns are made to be specific and the ambiguity is kept to the minimum.

How can a conversational bot be infused with a sense of personality? The **PERSONA-CHAT dataset** [15] has been created with help of crowd workers who were asked to chat in pairs and attempt for a natural conversation. The conversations held between each pair had different tones (topic wise and mood wise), called profiles. Different **personas** were also collected and for each persona there were five profile sentences. These personas were for training a conversational bot to infuse with the essences of individual personalities. Here each persona is essentially a short character summary and the profile sentences assigned to each persona, define that particular character. This dataset can be used either for creating a dialog agent capable of showing different personalities or for training models that, given a dialog history, will make predictions on the persona of that dialog.

V. CONCLUSION AND FUTURE SCOPE

A. Limitations
1) Retrieval-Based Model
- **Does not do well on topics that are outside its scope** - In this context, "scope" means the intents that are defined for the chatbot. For instance, a retrieval-based chatbot might have an intent called "name" which indicates that the user is asking for the chatbot's name. A retrieval-based chatbot tries to classify the intent of a user message into one of those that are pre-programmed into it. However, the user might enter a message such that the chatbot fails to find an intent that fits the user message. This can easily happen, as it is impossible for the programmer to foresee all the kinds of messages

that a user can provide. In such a case, a retrieval-based chatbot fails to provide meaningful output.

- **Might repeat replies for multiple questions that belong to the same topic** - The best that can be done to prevent a retrieval-based chatbot from repeating the exact same response for a question on a given topic (intent), is to provide multiple possible replies under each topic (intent) and have it choose one at random to give as a reply. Even under this scheme, since the number of possible replies under each intent is not very high, there can be frequent repetition of response, when asked multiple questions on the same topic.

2) Transformer-Based Model
- **Might not provide grammatically correct or even relevant replies** - As seen in the "Results and Discussion" section, the transformer-based chatbot gives nonsensical output that is neither related to the user's query, nor grammatically correct. The reason for this is closely linked to the following limitation.

- **Requires intensive computational resources to train to desired performance** - In this paper, the term "desired performance" simply means "coherent replies somewhat linked to the user's message". Yet, even to achieve this simple milestone, a transformer-based bot requires massive computational resources. A detailed breakdown of this is given in the table in section C in Results and Discussion. Hence, it is usually only feasible for big corporations having massive computational assets, to implement production-level transformer-based chatbots, and even then, they use hybrid models combining the two classes viz. Retrieval-based and Transformer-based.

3) B. Scope for Future Improvement

- **Increasing scale** - For the transformer-based model, increasing the corpus size, parameters and computational resources (resources used to train) could make a huge difference to its performance. As for the retrieval-based chatbot, more intents could be added to make the chatbot able to answer a wider range of questions.

- **Using a hybrid model** - Even production-level generative chatbots are not purely generative. They use retrieval-based models in addition, to get the best of both worlds. For instance, one strategy could be to let the retrieval-based model reply if it is able to classify the intent of a user message, but if it is unable to classify the intent, then to let the generative model generate a reply.

- **Better User Interface** - An application with a user-interface like those of mobile chat applications could be designed. Further, the processing of the user message and generation of replies could be moved to a server, requiring a user to only install a front-end application, reducing the processing power needed by an end-user to run the application. Of course, in that case, the server would receive requests in the form of user messages and send back replies.

- **Text-To-Speech and Speech-To-Text** - For users who require assistive technologies (visually impaired users) or simply for lazy convenience, a user could speak their message instead of typing it and the application would automatically convert it into text. Similarly, when the chatbot replies to a user's message, the reply could be read out aloud by the application for the user to hear.

- **Performance Measurement using NLP Metrics -** In the future, the performance of this Chatbot could be more accurately measured using metrics such as BLEU score(Bilingual Evaluation Understudy Score), perplexity, etc.

VI. ACKNOWLEDGEMENT

VII.   REFERENCES

[1]. Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, llia Polosukhin, "Attention is All You Need", Page-2-6, 2017, Conference on Neural Information Processing Systems, Long Beach, CA, USA.

[2]. Stephen Roller, Emily Dinan, Naman Goyal, Da Ju, Mary Williamson, Yinhan Liu, Jing Xu, Myle Ott, Kurt Shuster, Eric M. Smith, Y-Lan Boureau, Jason Weston, "Recipes for building an open-domain chatbot", Facebook AI Research, 30th April, 2020.

[3]. Humeau et al, Facebook AI research, 25th March, 2020.

[4]. Miller et al, Facebook AI research, 2017.

[5]. Denny Britz, "Recurrent Neural Network Tutorial, Part 4 – Implementing a GRU/LSTM RNN with Python and Theano.," WILDML, 27-Oct-2015.

[6]. Dzmitry Bahdanau, KyungHyun Cho, Yoshua Bengio, "Neural Machine Translation By Jointly Learning to Align and Translation", Page-3-4, 2015, Conference paper at ICLR.

[7]. "Neural Machine Translation with Attention.", tensorflow.org.

[8]. Amir Ali, Muhammad Zain Amin, "Conversational AI Chatbot Based on Encoder-Decoder Architectures with Attention Mechanism", Page-8-10, 2019, Artificial Intelligence Festival 2.0, NED University of Engineering and Technology, Karachi, Pakistan.

[9]. Abonia Sojasingarayar, "Seq2Seq AI Chatbot with Attention Mechanism", Page-3-10, 2020, IA School/University, Boulogne-Billancourt, France.

[10]. J.Prassanna, Khadar Nawas K, Christy Jackson J, Prabakaran R, Sakkaravarthi Ramanath, "Towards Building a Neural Conversation Chatbot through Seq2Seq Model", International Journal of Scientific & Technology Research, ISSN 2277-8616, Vol-9, Issue-3, Page-2, 2020.

[11]. A. Gillioz, J. Casas, E. Mugellini and O. A. Khaled, "Overview of the Transformer-based Models for NLP Tasks," 15th Conference on Computer Science and Information Systems (FedCSIS), Sofia, Bulgaria, 2020, pp. 179-183, doi: 10.15439/2020F20.

[12]. Julien Chaumond, Patrick von Platen, Orestis Floros, Anthony MOI, Aditya Malte, "How to train a new language model from scratch using Transformers and Tokenizers", blog/01_how_to_train.ipynb at master · huggingface/blog (github.com), 11th March, 2021.

[13]. Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, Yejin Choi, "The Curious Case Of Neural Text DeGeneration", Conference Paper at ICLR, 14th February 2020.

[14]. Saizheng Zhang, Emily Dinan, Jack Urbanek, Arthur Szlam, Douwe Kiela, Jason Weston, "Personalizing Dialogue Agents: I have a dog, do you have pets too?", Montreal Institute for Learning Algorithms, MILA and Facebook AI Research, 25th September.

AUTHOR'S PROFILE

Dr. Asoke Nath is working as Associate Professor in the department of Computer Science, St. Xavier's College (Autonomous), Kolkata. He is engaged in research work in the field of Cryptography and Network Security, Steganography, Green Computing, Big Data Analytics, Li-Fi technology, Mathematical modelling of Social Area Networks, MOOCs, etc.

He has published 253 research articles in different journals and conference proceedings.

Rupamita Sarkar is a Postgraduate student of Computer Science at St. Xavier's College (Autonomous), Kolkata. Her interests include Data Science and Machine Learning, with a particular leaning towards Natural Language Processing.

Swastik Mitra is a Postgraduate student of Computer Science at St. Xavier's College (Autonomous), Kolkata. He is interested in research work in the field of Machine Learning and Artificial Intelligence. Specifically, his topics of interest are Deep Learning and Natural Language Processing. He also shares an interest in Data Science, Big Data Analytics, Algorithms, and Distributed Systems.

Rohitaswa Pradhan is a Postgraduate student of Computer Science at St. Xavier's College (Autonomous), Kolkata. He is interested in the fields of Analysis of Algorithms, Data Science and Machine Learning. He is also enthusiastic about various Software Development technologies and Development Methodologies.

Cite this Article

Asoke Nath, Rupamita Sarkar, Swastik Mitra, Rohitaswa Pradhan , "Designing and Implementing Conversational Intelligent Chat-bot Using Natural Language Processing ", International Journal of Scientific Research in Computer Science, Engineering and Information Technology (IJSRCSEIT), ISSN : 2456-3307, Volume 7 Issue 3, pp. 262-266, May-June 2021. Available at doi : https://doi.org/10.32628/CSEIT217351
Journal URL : https://ijsrcseit.com/CSEIT217351