IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

Self Learning Bot

20201CAI0152 – P. Arun Teja 20201CAI0096 – S. Ganesh) 20201CAI0104 – M. Maheswar Reddy
20201CAI0131-A. Karthik Sharma
Prof. Dr. Asif Mohammed H.B
(Computer Science/Presidency University, India)
(Computer Science/Presidency University, India)
(Computer Science/Presidency University, India)
(Computer Science/Presidency University ,India)
(Computer Science/Presidency University, India)

Abstract: In this Digital decade, Imagine having a friendly assistant who is always ready to help and make your Customer Service or any other service easier for you than ever before. An interactive tool that is designed to understand and respond to your messages or voice commands in a way that feels like chatting with a human. Our approach is to interact with users providing a more dynamic and reliable system. It should serve as a bridge between you and the digital world which truly understands your needs and has the ability to go beyond basic responses. This understanding empowers the chatbot to take initiative and make decisions on your behalf. It aims to measure tasks not just by completing them but by the ease and efficiency with which they are accomplished.

Background: Over the past years, we have witnessed many changes in artificial intelligence and how humans interact with it. Rule-based bots require us to program or give them certain rules to follow, and they do not communicate with humans effectively. In the realm of conversational agents and virtual assistants, the demand for intelligent or self-improving bots has grown to a large extent. Users can now expect more natural and personalized interactions with machines, leading to the development of self-learning bots that can understand user intent, adapt to preference, and continuously evolve.

Materials and Methods: In this research, the Data Collection involves a systematic approach that selects materials, methodologies, and algorithms to materials to improve and make increase the adaptation process . Diverse datasets that contain user interactions, historical data, and contextual information. Methods include mechanisms that calculate the probability of intent action or response to be given to the user. This may involve scraping online conversations, user queries, and any other data sources necessary for training. Methods like tokenization, lemmatizing the words, and normalization make data suitable for training. Cleaning data to remove noise and irrelevant information. Model Selection includes frameworks such as TensorFlow and Natural Language Tool Kit, and deep learning Neural Networks as models that are to be trained. Training the selected model with the dataset and fine–tune it to minimize the difference between predicted and actual outputs.

Results: The accurate responses for the user and adaptability to the user dynamic environments where user behavior, language, and preference change over time. The main result of this is user satisfaction which enhances their easier experience. The Mic that is provided could help with the necessary actions that can be done with voice commands to the bot so it can perform the necessary things such as translation, surfing Google, and playing references for the service you are looking.

Conclusion: In Conclusion, the development and deployment of self-learning bots represent a significant leap forward in the realm of intelligent and interactive systems. The journey of creating a self-learning bot involves meticulous attention to many key components of user experience. The Continuous improvement loop ensures that self-learning bots remain at the forefront of technological advancement.

Key Word: Self-learning bot, Natural language processing, Natural language Understanding, Deep learning.

I. INTRODUCTION

A self-learning chatbot is like a conversation agent that can attract huge interest from users as they can perform different tasks which are meant for you in a simple and easiest way. For business purposes, self-learning bots are capable of having human-like conversations with a user by receiving and sending text messages for the purpose of automating business processes. Self-learning bots are advanced forms of normal chatbots. It uses deep learning and Natural Language Processing. These types of bots are reliably made to make the service easy to the customer or clients for their respective business. These bots can have several capabilities like task-oriented or response-oriented to allow to increase the efficiency of business. It is a human-like partner that interacts with us and redirects us to some sites we need or do some tasks we need hands-free which causes people to draw near to clients.

NLU is the ability of the chatbot to understand what a human says or writes. It is the process of transforming text or speech into structured information that a machine can understand. NLU follows three specific concepts: entities, context, and intents. Entities represent a concept to the chatbot, such as a refund system in our e-commerce chatbot. Context means the situation or background of the conversation. If a natural language understanding algorithm detects a request but has no history of the conversation, it will not be able to remember the request and give the response. Intents mean the goals or expectations of the user when they make a request or ask a question. For example, the user says "Send a request". NLP is the branch of artificial intelligence that allows more natural human-to-computer communication by linking human and machine language. NLP bots are designed to convert the text or speech inputs of the user into structured data. The data is then used to choose a relevant answer. It consists of tokenization, sentiment analysis, normalization, entity recognition, and dependency parsing. These are techniques that help the bot process and analyze the natural language.

II. METHODOLOGY RESEARCH

Natural Language Processing: NLP is a way for computers to understand and use human language, both written and spoken. It helps computers break down text into smaller parts, and figure out what they mean, and what the speaker or writer wants to say. Computers learn from a lot of text data how to do things like finding out how people feel, translating between languages, and creating new text. NLP makes possible many things we use every day, like chatbots, virtual assistants, search engines, language translation, sentiment analysis, text summarization, and more. NLP is always improving with new methods from deep learning, making computers more capable of understanding language as humans do.

Deep Learning: Deep learning is a type of machine learning that uses artificial brain-like networks to learn from data. It can find hidden patterns and features that are hard to see otherwise. Deep learning networks have many layers that change the input data into more and more useful information. To train these networks, we need to tweak millions of settings until they can make good predictions. Some common types of deep learning networks are CNNs for pictures and RNNs for words or sounds. Deep learning has done amazing things in making computers see, understand, and talk. But it also has some problems, like needing a lot of data, power, and explanation.

Seq2Seq: Seq2Seq models are good for changing one kind of data into another, like words into another language or sentences into a summary. They have two parts: an encoder and a decoder. The encoder reads the input data and makes a small piece of information that has everything important in it. This is called the context vector. The decoder uses the context vector to make the output data, one piece at a time. It also looks at what it has made before. To teach Seq2Seq models, we give them examples of input and output data and help them get better at matching them.

Seq2Seq models can do many things, like making chatbots, turning speech into text, and any problem that needs changing data.

Pretrained GPT: A pretrained GPT (Generative Pretrained Transformer) is a language model that has been trained on a large amount of diverse text data before any specific task. This helps it capture broad linguistic patterns. It uses a transfer learning approach, where knowledge gained from the initial training is fine-tuned for specific tasks, saving computational resources and time. GPT is built on the transformer architecture, which utilizes attention mechanisms to capture long-range dependencies and relationships between words in a sequence. Self-attention allows each word to focus on different parts of the input sequence, enabling the model to understand context and relationships effectively. GPT is **generative**, meaning it can autonomously create coherent and contextually relevant text based on a given prompt or context. The input text is **tokenized** into smaller units, usually words or sub-words, and the model predicts the likelihood of the next token given the preceding context during training. Pretrained GPT models find applications in various natural language processing tasks, such as text completion, question answering, and language generation, demonstrating state-of-the-art performance in multiple domains.

Frameworks :

TensorFlow and Keras: TensorFlow is an open-source machine learning framework developed by Google. It's like a powerful toolbox that helps you teach your computer to recognize patterns in data so it can make predictions or perform tasks on its own. TensorFlow is versatile and can handle various machine-learning tasks, from recognizing images to translating languages. It allows you to build and train models by providing your computer with a set of tools and instructions on how to learn from examples. TensorFlow is excellent for tasks that involve lots of data and complex patterns, and it can work on different types of devices, from your laptop to large servers.

Keras is a user-friendly interface or a simplified set of instructions built on top of TensorFlow. It simplifies the process of building and training neural networks (a type of model within machine learning) using TensorFlow. Keras is designed to be intuitive and beginner-friendly, making it easier for you to create powerful models without diving into all the complexities of TensorFlow. So, if you want to build a neural network to recognize patterns in data, Keras is the tool that makes it a bit more like assembling blocks following certain rules.

Tkinter: Tkinter is a framework in Python that helps create graphical user interfaces (GUIs) for journaling applications. It provides a range of widgets like buttons, labels, and entry fields that can be easily added to a window. Tkinter uses a geometry manager for layout management, allowing the organization of widgets within the GUI. Event handling is facilitated, enabling the definition of functions triggered by user actions like button clicks. To integrate Tkinter into a journaling application, one could design a GUI with entry fields for date, title, and content, along with buttons for saving and viewing entries. The framework's simplicity makes it accessible for creating an intuitive interface for users to interact with their journal entries. The example code snippet demonstrates how to build a basic journal entry GUI using Tkinter, with entry fields, labels, and a save button arranged using the grid manager. This foundation can be expanded upon to meet specific requirements for journaling functionalities.

III. Result (11 Bold)

TF-IDF (Term Frequency-Inverse Document Frequency):
Term Frequency (TF):

$$TF(t, d) = \frac{\text{Number of times term t appears in document d}}{\text{Total number of terms in a document d}}$$

Inverse Document Frequency (IDF):

$$IDF(t, D) = \log\left(\frac{\text{Total number transform computes the TF-of documents in the corpus N}}{\text{Number of documents containing term t+1}}\right)$$

TF-IDF:

- **TF-IDF(t,d,D) = TF(t,d) x IDF(t,d)**
- **Tf-idf-Vectorizer is used to compute the TF-IDF representation.**
- **IDF vectors for the user input and candidate answers.**

Cosine Similarity:

$$\text{Cosine Similarity}(A,B) = \frac{A.B}{|A|.|B|}$$

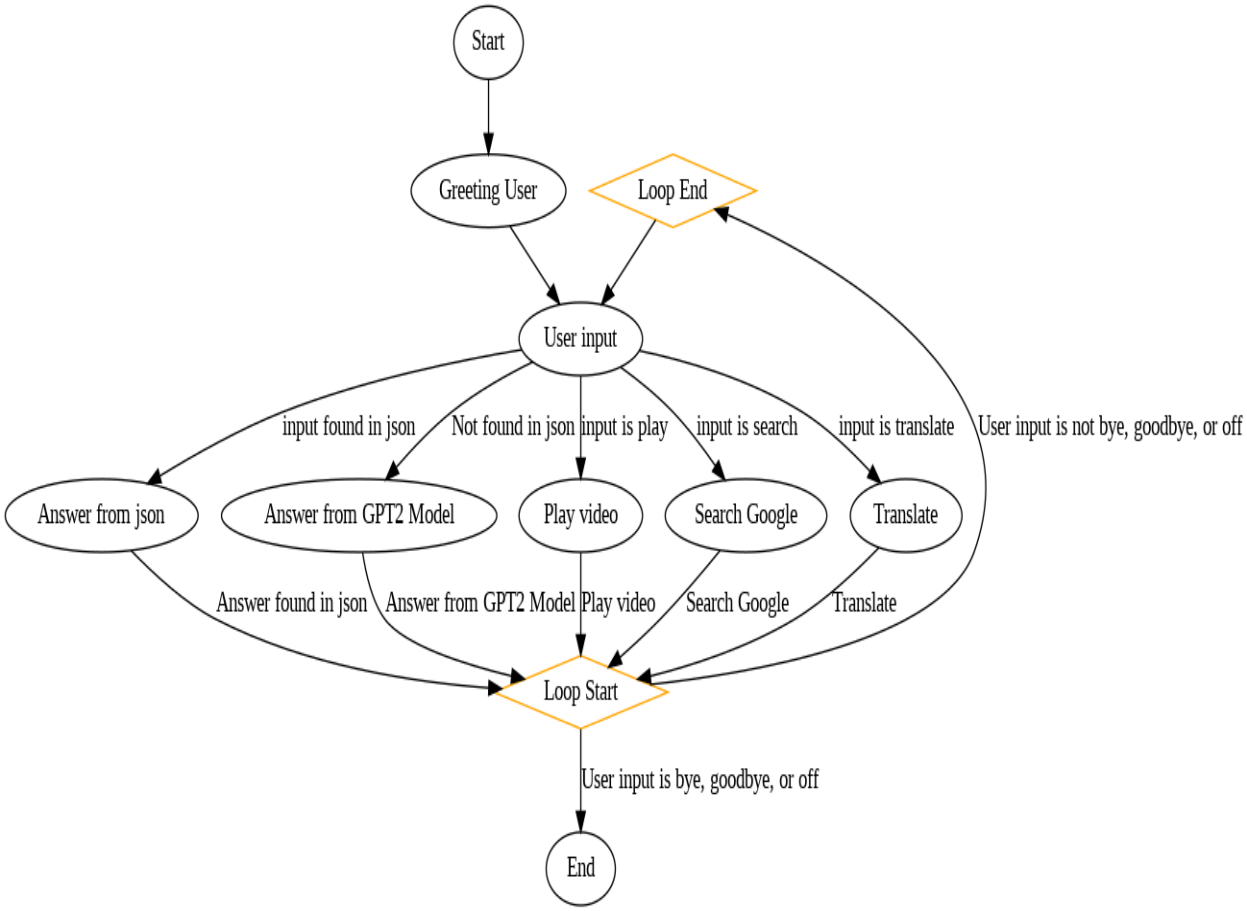
Where:

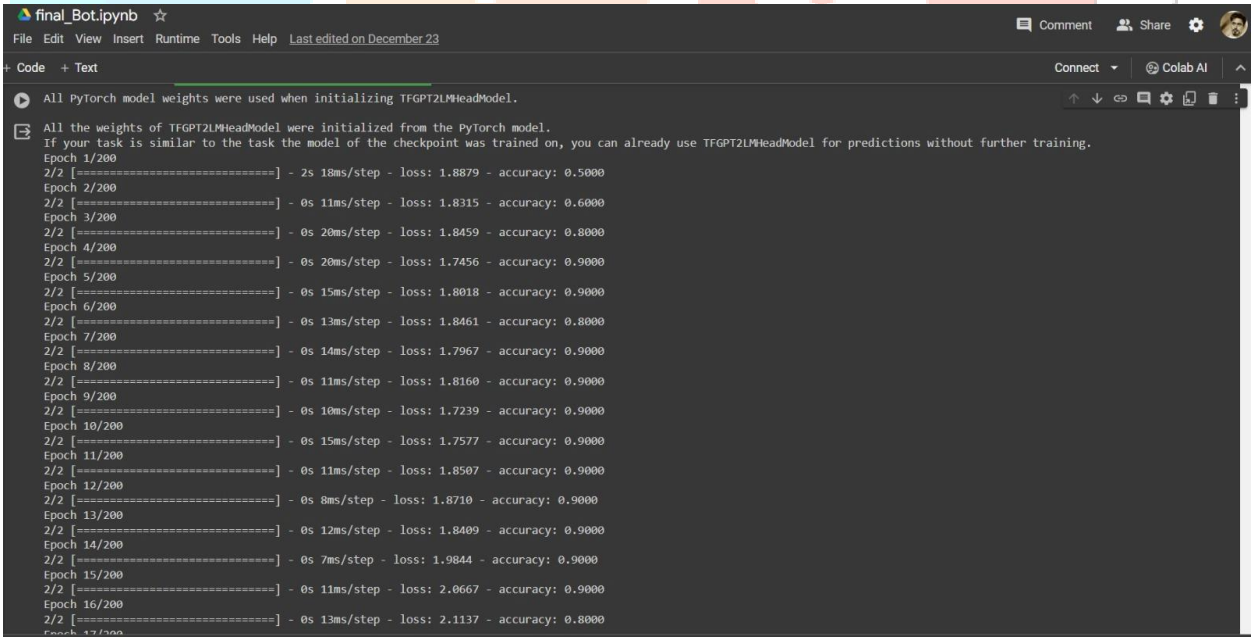
- **A.B is the dot product of vectors A and B**
- **|A| and |B| are the magnitudes (Euclidean norms) of vectors A and B**
- **Cosine similarity(vectors) calculates the cosine similarity between the TF-IDF vectors of the user input and candidate answers.**
- **The cosine similarity values are then used as relevancy scores, indicating how similar each candidate's answer is to the user input.**

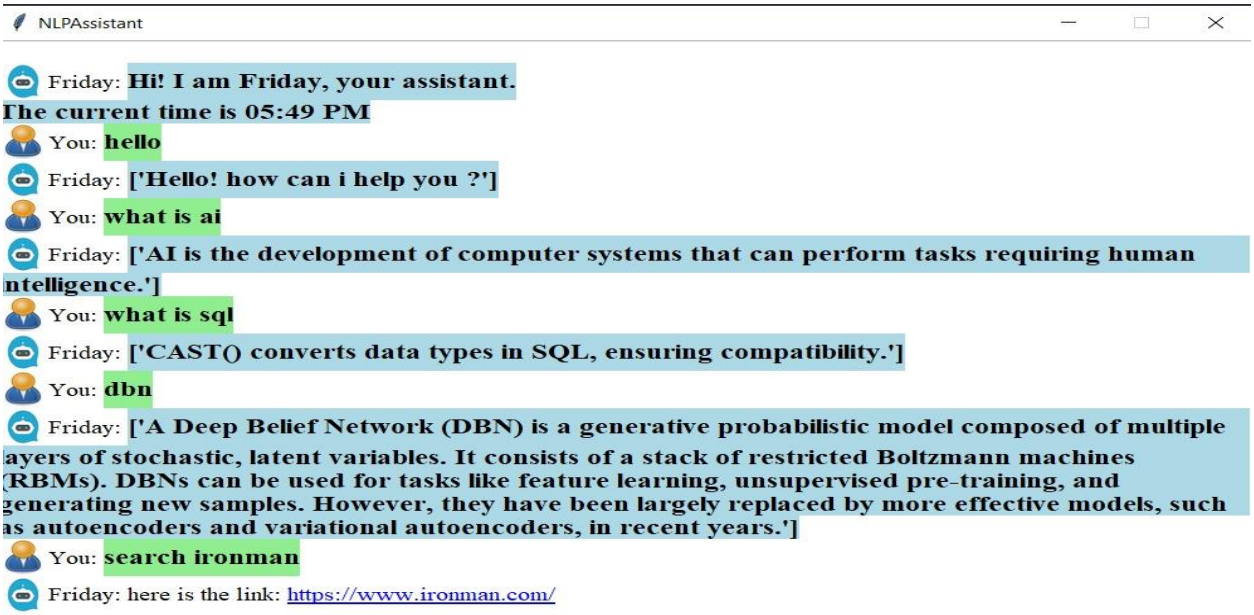
Bag of Words:

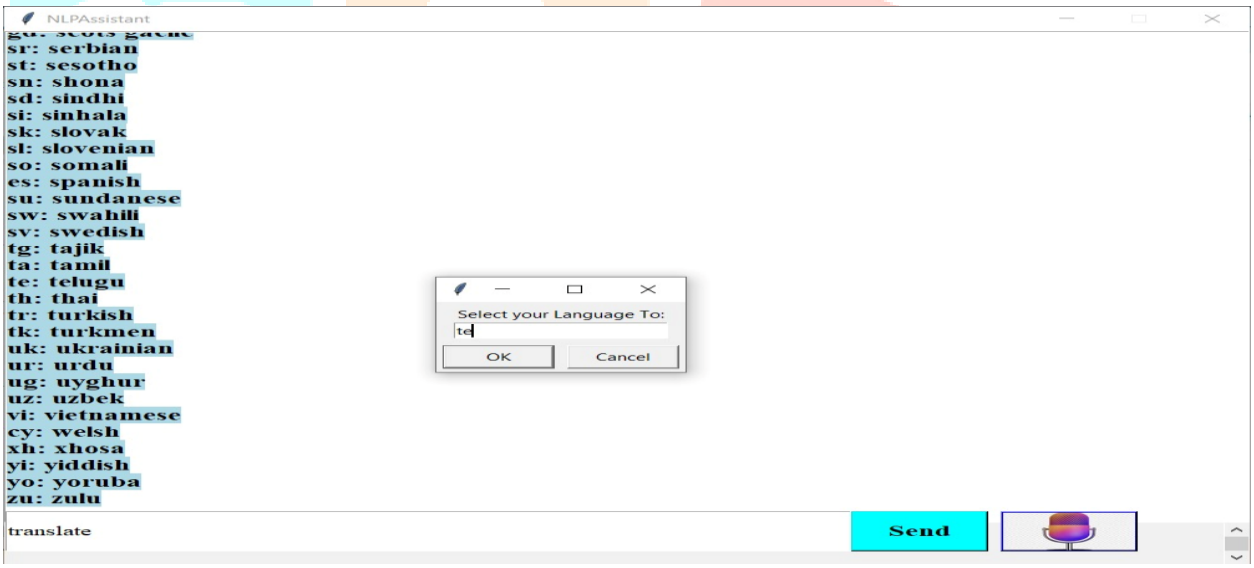
The bag-of-words model is a technique used in Natural Language Processing (NLP) to preprocess text. It converts text into an unordered collection of words, disregarding grammar and word order. This technique is used to keep a count of the total occurrences of the most frequently used words. The bag-of-words model is used to facilitate text analysis and enables efficient techniques like Term Frequency-Inverse Document Frequency (TF-IDF) and cosine similarity for similarity comparisons.

In summary, the bag-of-words model is a technique used in NLP to preprocess text by converting it into a bag of words, which keeps a count of the total occurrences of the most frequently used words. It is used to facilitate text analysis and enables efficient techniques like TF-IDF and cosine similarity for similarity comparisons.


Start
Greeting User
Loop End
User input
input found in json
Not found in json
input is play
input is search
input is translate
User input is not bye, goodbye, or off
Answer from json
Answer from GPT2 Model
Play video
Search Google
Translate
Answer found in json
Answer from GPT2 Model
Play video
Search Google
Translate
Loop Start
User input is bye, goodbye, or off
End


final_Bot.ipynb
File Edit View Insert Runtime Tools Help Last edited on December 23
Comment
Share
Code
Text
Connect
Colab AI
All PyTorch model weights were used when initializing TFGPT2LMHeadModel.
All the weights of TFGPT2LMHeadModel were initialized from the PyTorch model.
If your task is similar to the task the model of the checkpoint was trained on, you can already use TFGPT2LMHeadModel for predictions without further training.
Epoch 1/200
2/2 [==========] - 2s 18ms/step - loss: 1.8879 - accuracy: 0.5000
Epoch 2/200
2/2 [==========] - 0s 11ms/step - loss: 1.8315 - accuracy: 0.6000
Epoch 3/200
2/2 [==========] - 0s 20ms/step - loss: 1.8459 - accuracy: 0.8000
Epoch 4/200
2/2 [==========] - 0s 20ms/step - loss: 1.7456 - accuracy: 0.9000
Epoch 5/200
2/2 [==========] - 0s 15ms/step - loss: 1.8018 - accuracy: 0.9000
Epoch 6/200
2/2 [==========] - 0s 13ms/step - loss: 1.8461 - accuracy: 0.8000
Epoch 7/200
2/2 [==========] - 0s 14ms/step - loss: 1.7967 - accuracy: 0.9000
Epoch 8/200
2/2 [==========] - 0s 11ms/step - loss: 1.8160 - accuracy: 0.9000
Epoch 9/200
2/2 [==========] - 0s 10ms/step - loss: 1.7239 - accuracy: 0.9000
Epoch 10/200
2/2 [==========] - 0s 15ms/step - loss: 1.7577 - accuracy: 0.9000
Epoch 11/200
2/2 [==========] - 0s 11ms/step - loss: 1.8507 - accuracy: 0.9000
Epoch 12/200
2/2 [==========] - 0s 8ms/step - loss: 1.8710 - accuracy: 0.9000
Epoch 13/200
2/2 [==========] - 0s 12ms/step - loss: 1.8409 - accuracy: 0.9000
Epoch 14/200
2/2 [==========] - 0s 7ms/step - loss: 1.9844 - accuracy: 0.9000
Epoch 15/200
2/2 [==========] - 0s 11ms/step - loss: 2.0667 - accuracy: 0.9000
Epoch 16/200
2/2 [==========] - 0s 13ms/step - loss: 2.1137 - accuracy: 0.8000


NLPAssistant
Friday: Hi! I am Friday, your assistant.
The current time is 05:49 PM
You: hello
Friday: ['Hello! how can i help you ?']
You: what is ai
Friday: ['AI is the development of computer systems that can perform tasks requiring human ntelligence.']
You: what is sql
Friday: ['CAST() converts data types in SQL, ensuring compatibility.']
You: dbn
Friday: ['A Deep Belief Network (DBN) is a generative probabilistic model composed of multiple ayers of stochastic, latent variables. It consists of a stack of restricted Boltzmann machines (RBMs). DBNs can be used for tasks like feature learning, unsupervised pre-training, and generating new samples. However, they have been largely replaced by more effective models, such as autoencoders and variational autoencoders, in recent years.']
You: search ironman
Friday: here is the link: https://www.ironman.com/
Send


NLPAssistant
sr: serbian
st: sesotho
sn: shona
sd: sindhi
si: sinhala
sk: slovak
sl: slovenian
so: somali
es: spanish
su: sundanese
sw: swahili
sv: swedish
tg: tajik
ta: tamil
te: telugu
th: thai
tr: turkish
tk: turkmen
uk: ukrainian
ur: urdu
ug: uyghur
uz: uzbek
vi: vietnamese
cy: welsh
xh: xhosa
yi: yiddish
yo: yoruba
zu: zulu
Select your Language To:
te
OK
Cancel
translate
Send

IV. Conclusion (11 Bold)

In conclusion, the code gives a versatile and interactive chatbot application that leverages advanced technologies such as GPT-2 language modeling, speech recognition, and external APIs. Despite notable functionalities, there are areas for improvement, including refining context understanding, enhancing speech recognition accuracy, and incorporating more sophisticated error handling. The script stands as a foundation for a robust chatbot, offering a dynamic user experience, and future enhancements could amplify its capabilities for effective natural language interactions.

References (11 Bold)

[1]. Self-Learning Conversational AI Chatbot Using Natural Language Processing] - https://ijarsct.co.in/Paper4603.pdf

[2]. [High Accuracy Conversational AI Chatbot Using Deep Recurrent Neural Networks Based on BiLSTM Model - https://ieeexplore.ieee.org/document/9332074

[3]. A self Learning Chat-Bot From User Interactions and Preferences] - https://ieeexplore.ieee.org/document/9120912

[4]. A Survey On Chatbots Using Artificial Intelligence - https://ieeexplore.ieee.org/document/9262366/

[5]. A Survey on Chatbot Implementation in Customer Service Industry through Deep Neural Networks- https://ieeexplore.ieee.org/document/8592630

[6]. E-Commerce Assistance with a Smart Chatbot using Artificial Intelligence - https://ieeexplore.ieee.org/document/9445316

[7]. Gpt2 Model: Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., & Sutskever, I. (2019). Language Models are Few-Shot Learners. arXiv preprint arXiv:2005.14165.

[8]. Transformers: Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., ... & Brew, J. (2019). Hugging Face's Transformers: State-of-the-art Natural Language Processing. arXiv preprint arXiv:1910.03771.

[9]. Spacy: Honnibal, M., & Montani, I. (2017). spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks, and incremental parsing.

[10]. Tkinter: Python Software Foundation. (n.d.). Tkinter - Python interface to Tcl/Tk. Retrieved from https://docs.python.org/3/library/tkinter.html

[11]. Self-Learning Conversational AI Chatbot Using Natural Language Processing - https://ijarsct.co.in/Paper4603.pdf

[12]. A self Learning Chat-Bot From User Interactions and Preferences - https://ieeexplore.ieee.org/document/9120912

[13]. A Survey on Chatbot Implementation in Customer Service Industry through Deep Neural Networks - https://ieeexplore.ieee.org/document/8592630