Educational Chatbots

Oumayma EL Hit[1]

[1] *Computer Science Student at Linköping University, oumel125@student.liu.se*

January, 2023

Abstract

Nowadays, chatbots have become widespread and are used in various industries such as e-commerce, healthcare, finance, and education. They can be integrated into websites, messaging platforms, and mobile apps, and can be accessed via voice or text. Many businesses are leveraging chatbots to enhance customer experience, improve efficiency, and reduce operational costs.(1)

Additionally, advancements in natural language processing and machine learning have allowed chatbots to become more sophisticated and able to handle complex queries and provide personalized recommendations. There are different types of chatbots particularly those used for educational purposes.

This project will highlight how to build an educational chatbot from scratch, which can help students revise most asked questions in the field of artificial intelligence as an instance.(2)

Introduction

Educational chatbots are computer programs designed to assist with learning and education. They can be used to provide information, answer questions, offer study material, and even conduct quizzes and assessments. These chatbots can be integrated into learning management systems and mobile learning apps, and can be accessed by students anytime, anywhere. Educational chatbots can help to improve learning outcomes, save time and effort for both students and teachers, and provide customized and interactive learning experiences. They are also being used to provide support for distance and online learning, as well as to assist with language learning and skills training. This paper will present the theoretical background of how to build an educational chatbot namely the NLP methods used and the neural networks of deep learning. Finally, we will discuss some final thoughts and possible improvements.

Training data

Our data is stored in a JSON format file which is commonly used for data storage and transmission. The key terms in the data are "tag", "patterns", and "responses". To use the chatbot, the user's query must be included in the "patterns". The chatbot will then randomly select a reply from the "responses" set to reply to the user's query. The "tag" groups similar patterns and responses into a category, making it easier for the model to predict the pattern's category.

This data is presented in the form of a JSON file, which is commonly used for data storage and transmission. The key terms to note are "tag", "patterns", and "responses". Any user query the chatbot needs to interpret must be included in the "patterns". The chatbot will then generate a reply, randomly selected from the predefined "responses". The "tag" groups similar patterns and responses into a specific category, making it easier for the model to predict the category a pattern belongs to.(3)

We can train a neural network with the data to classify it into a tag from the file, and then select a random response from that tag to display to the user. The more tags, patterns, and responses we add, the stronger the chatbot will become.

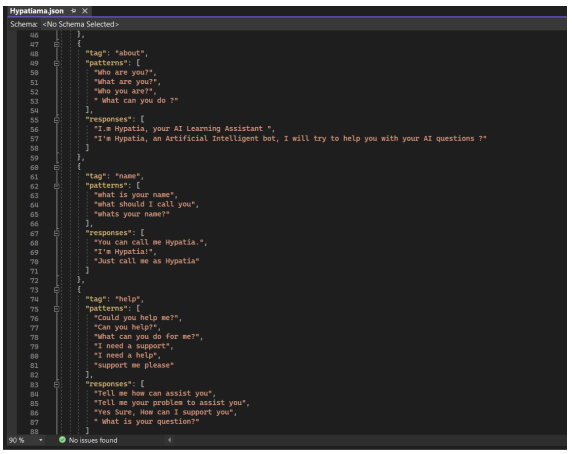


Figure 1 An extract of the Json file of the training data

Figure 2 An other extract of the Json file of the training data

NLP methods

NLP stands for Natural Language Processing, which is a subfield of artificial intelligence and computer science that deals with the interaction between computers and humans in natural language. It involves the development of algorithms and models that can understand, interpret, and generate human language. The goal of NLP is to enable computers to understand and generate human language in a way that is useful and effective for a particular task or application. To process the given text data, The "tag" and "patterns" must be extracted from the file and stored in a list. We'll also create a collection of unique words in the patterns to form a Bag of Words (BoW) vector. To do so, I've created lists to store : all the unique words, all the unique tags, the patterns as well as the corresponding tags. We use different NLP methods to process the data: converting patterns to lower case, tokenization, stemming, vectorization. (4)

Tokenization

Tokenization is a process in Natural Language Processing (NLP) that involves breaking down a sentence or a text document into smaller units, called tokens. Tokens can be words, phrases, symbols, or even individual characters. The purpose of tokenization is to convert a large text document into smaller, manageable units that can be analyzed and processed more easily. Tokenization is a crucial step in NLP preprocessing, as it helps to remove any irrelevant information and transform the text into a format that can be processed by NLP algorithms.

Stemming

Stemming is a process in Natural Language Processing (NLP) that involves reducing words to their base or root form. The goal of stemming is to reduce words to their core structure, so that words with the same meaning but different inflections are treated as the same word. This is useful in NLP tasks such as text classification, information retrieval, and sentiment analysis, where words with similar meaning but different forms can cause problems for the algorithms. Common stemming algorithms include the Porter Stemmer, Snowball Stemmer, and the Lancaster Stemmer. In the code used, we use Lancaster stemmer.

Vectorization

Vectorization is a process in Natural Language Processing (NLP) that involves transforming text data into numerical vectors, which can then be processed by mathematical algorithms. The purpose of vectorization is to convert text data, which is inherently unstructured and difficult to process, into a format that can be easily manipulated and analyzed.(5)

There are several methods for vectorizing text data, including:

(1) One-hot encoding: This method involves creating a vector for each unique word in the text data, where each vector has a length equal to the number of unique words, and only a single element in the vector is set to 1.
(2) Term Frequency-Inverse Document Frequency (TF-IDF): This method involves calculating the frequency of each word in a document, and normalizing the frequency based on the number of times the word appears across all documents.
(3) Word Embeddings: This method involves using deep learning techniques to learn low-dimensional representations of words, where words that are semantically similar are mapped to nearby points in the vector space.

The Bag of Words (BoW) approach will be utilized in the code. BoW provides information on the presence of words within a document. In this scenario, each sentence will be represented by a list based on the number of unique words in the "words" list. The list's positions correspond to unique words from "words". If a sentence contains a particular word, its position will be marked as 1, and if the word is absent, it will be marked as 0.

The deep learning model

Deep learning

Deep Learning is a subfield of machine learning concerned with algorithms inspired by the structure and function of the brain, called artificial neural networks. These algorithms have been used to achieve state-of-the-art results in a wide range of tasks such as image classification, speech recognition, natural language processing, and game playing. Deep Learning models are capable of automatically extracting and learning complex and abstract features from raw data.(1)

With data preprocessing complete, we move on to constructing a model and feeding it the processed data. The network design is relatively simple, consisting of two hidden Fully Connected Layers (FC layers) and one output layer. The final layer utilizes a softmax activation. Experimentation is encouraged to determine the ideal architecture that meets our needs. Additional steps in text preprocessing may also be implemented to enhance results.

With the model set up, it's time to train it with the data. The number of training iterations, known as epochs, is set to 200. However, we have the flexibility to adjust this value to find the best results for our data.

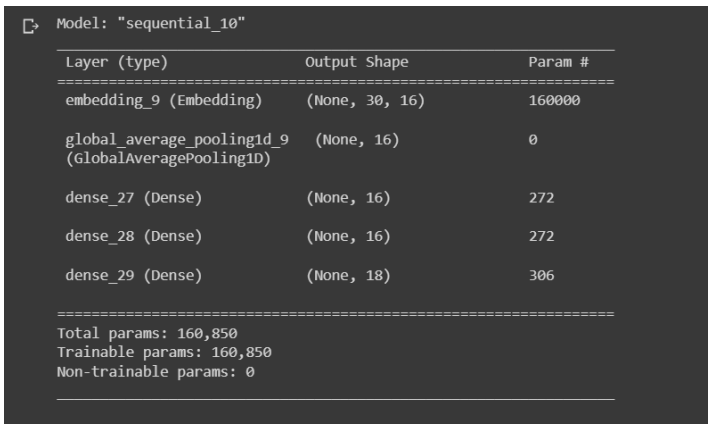
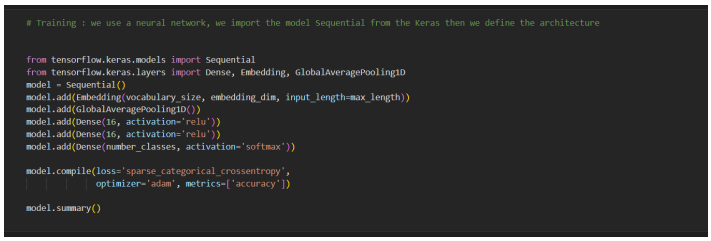


Figure 3 The model structure



```
# Training : we use a neural network, we import the model Sequential from the Keras then we define the architecture

from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense, Embedding, GlobalAveragePooling1D
model = Sequential()
model.add(Embedding(vocabulary_size, embedding_dim, input_length=max_length))
model.add(GlobalAveragePooling1D())
model.add(Dense(16, activation='relu'))
model.add(Dense(16, activation='relu'))
model.add(Dense(number_classes, activation='softmax'))

model.compile(loss='sparse_categorical_crossentropy',
              optimizer='adam', metrics=['accuracy'])

model.summary()
```

Figure 4 The model structure code

Demonstration

Here are some conversations with the chatbot:

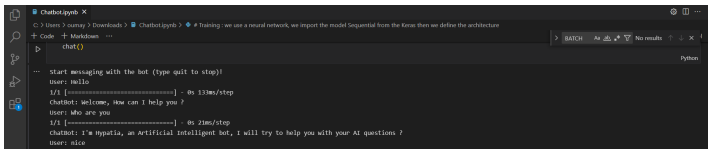


Figure 5 Examples of conversations with the chatbot

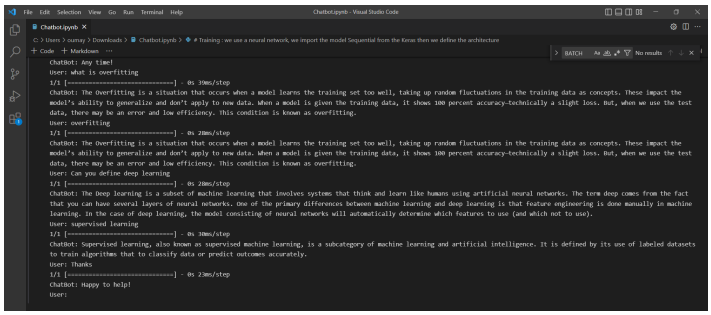


Figure 6 Another example of a discussion with the chatbot

Evaluation and results analysis:

The evaluation of the chatbot is based on its accuracy and its speed. It is highly accurate when the question asked by the user exists in the training data or when it is similar to an already existing question, but the chatbot is unable to generalize to new, unseen data . This means that the chatbot may perform well on the training data, but have poor performance on real-world data that it has not seen before. The goal of this project is to show and understand the way chatbots are built using deep learning techniques. We focus in this project on the machine learning behind chatbots(that's why a small dataset is used instead of a large one).

Possible improvements

In conclusion, the chatbot project has been successfully completed and is functioning as intended. However, there are ample opportunities for further improvement and customization :

(1) We can connect the chatbot to a database or integrate it with a website for greater functionality.
(2) Additionally, we can consider exploring alternative vectorization methods, such as TF-IDF or Word2Vec, as these may enhance the model's accuracy in predicting tags.
(3) we can add more different NLP methods.
(4) we can implement an application or a design for the chatbot

Conclusion

The future of educational chatbots is expected to be bright, with a growing demand for personalized, accessible, and convenient learning experiences. Chatbots have the potential to revolutionize the way education is delivered, by providing 24/7 access to educational content and resources, conducting real-time assessments, and offering individualized learning paths. Additionally, advancements in natural language processing (NLP) and machine learning will enable chatbots to provide even more accurate and engaging learning experiences. However, it's important to note that chatbots should be seen as a supplement to traditional education, not a replacement, and should be used in conjunction with human teachers to maximize their effectiveness.

The aim of the project was to build a simple chatbot, from scratch, which can help in the educational field, particularly in the field of artificial intelligence as an example, by providing summaries, definitions of some methods or terms, provide the difference between two models as an example. We did manage to succesfully build a chatbot but there are many aspects that might be improved or developed mainly the training data, the NLP methods used as well as the design of the chatbot. (1)

References

1 10 Of The Best Use Cases Of Educational Chatbots In 2023 DHEERAJ DANI. https://www.kommunicate.io/blog/10-best-use-cases-of-educational-chatbots/. 2022.

2 10 of the Most Innovative Chatbots on the Web Dan Shewan. https://www.wordstream.com/blog/ws/chatbots. 2022.

3 Pratheesh Shivaprasad ;How To Build a Basic Chatbot from Scratch. https://towardsdatascience.com/how-to-build-a-basic-chatbot-from-scratch-f63a2ccf5262. 2020.

4 Natural Language Processing With Python's NLTK Package Joanna Jablonski. https://realpython.com/nltk python/: :text=natural %20language %20processing%20(nlp) %20is,and%20contains%20human %2dreadable/%20text.

5 Understanding NLP Word Embeddings — Text Vectorization Prabhu. https://towardsdatascience.com/understanding-nlp-word-embeddings-text-vectorization-1a23744f7223: :text=word %20embeddings %20or %20word %20vectorization,into %20numbers %20are %20called %20vectorization. 2019.