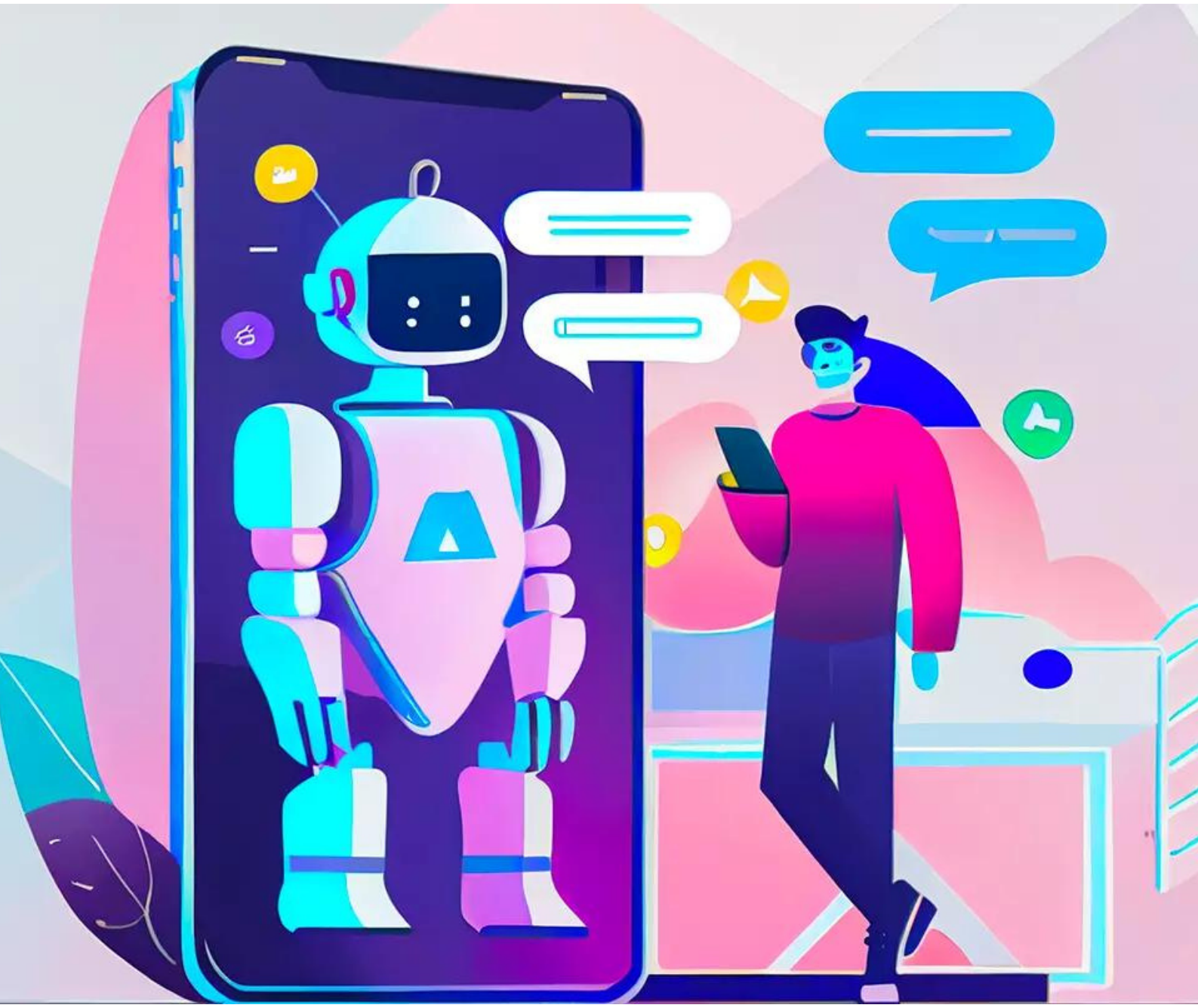
TI101:My first chatBot



By :
Paul Gournay and Vrej KARAKOZIAN
Promo 2028

PARIS PANTHÉON-ASSAS UNIVERSITÉ

https://github.com/sheepfd/Pychatbot-Karakozian-Gournay-Int1

Table of content

Introduction

Hello, everyone! Today, I'm excited to present to you a fascinating project that delves into the world of text analysis, an essential aspect of developing intelligent systems such as chatbots and generative artificial intelligences. Our project focuses on a method based on word occurrences to generate intelligent responses from a corpus of texts, and it involves several key steps in text processing.

Goals and target

The primary goal of this project is to design a system capable of answering questions based on the frequency of words in a given corpus. By employing the TF-IDF (Term Frequency-Inverse Document Frequency) method, our application aims to provide insightful responses by understanding the importance of words within a collection of documents. Another goal

Functional presentation

We use a lot of function, in our project and here is a list of the most importants ones :

- list_of_files : create a list of the name of files in the chosed folder

- Cleanedfile : create all the new files that are cleaned to be used in other functions

- tf : count how much the word is present in a file

- idf : compute the idf score of each unique word in the file

- Tf_idf : product between tf and idf that shom how much important a word is. We use first a dictionarie and then create the matrix with this dictionarie of arrays

- TF_IDF_question_dico and TF_IDF_question_mat : used to calculate the tf_idf of the question

- similarity : calculate the similarity of two vector using scalar_product function and calculate_norm function

- most_relevant_document : uses similarity function to tell us wich document in the folder is the most similar to our question

Difficulties encountered and solutions provided

First of all, the thing with what we had the more difficulties with is technical difficulties with the usage of psalm and github that are very difficult to understand.

Next, we also had a lot of difficulties using an making the matrix for TF_IDF for example because it was more natural for us to use a dictionary to have for each score the word in the keys.
Also, converting our dictionaries into matrix have made them to have random order that makes the computation of similarity very difficult.

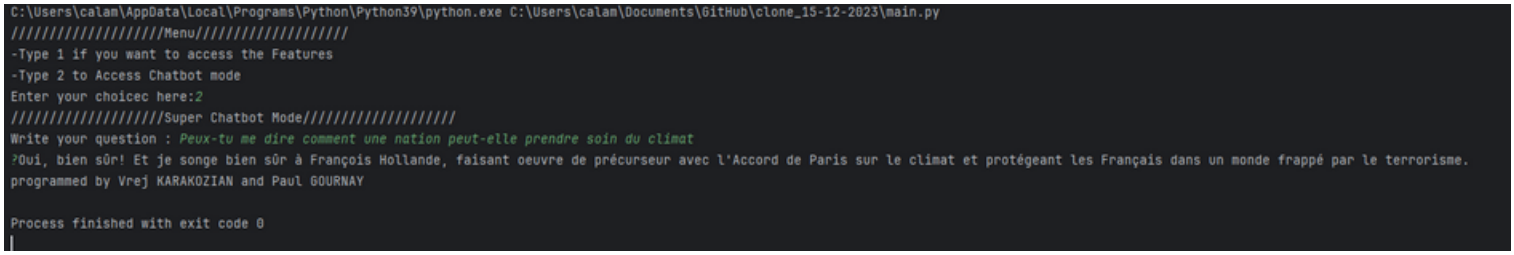
That's why we decided to add the tf_idf score of the question in the dictionary of the tf_idf of all the corpus so all words are in the same index when we create the matrix used in the computation of similarity

We also had trouble to understand what we had to do with the questions of the instruction file

Finally we had trouble with the IDF function, we didn’t know if we had to put +1 after the LOG.

Results presentation:

Like you can see in this screenshot we can chose between the features mode or the Chatbot Mode and then chose the number of the feature we want or ask the question we want to ask.



There is some issues with our code because for some question none of the word are in the corpus so there is no answer.
One of the solution to this problem is to add more file to our corpus.

Conclusion:

Throughout the course of this project, our team gained valuable insights and knowledge. This endeavor introduced us to novel concepts that significantly enhanced our understanding of analyzing large datasets more effectively. The inclusion of concepts such as the TF-IDF score and cosine similarity required us to adopt a different mindset to approach and efficiently address the challenges presented.
A pivotal aspect of our success was the strategic decision to break down the program into smaller, independent functions. This approach proved essential in navigating through the complexities of the project and achieving successful outcomes.
Moreover, the project underscored the importance of utilizing version control software, such as git. This tool not only facilitated seamless collaboration but also enhanced our organizational efficiency through features like branches and pull requests. Overall, the project served as an excellent platform for acquiring new concepts and skills, contributing to our continuous learning journey.