



Projet de fin de module

Artificial Intelligence

SABRINE CHAACHOUA

ESTIAM E5



DEVELOPPEMENT D'UN CHATBOT SUR LES JEUX VIDEO

Dans le cadre de ce projet, notre objectif est de concevoir un chatbot spécialisé dans le domaine des jeux vidéo, en utilisant l'API de ChatGPT pour générer des réponses pertinentes.

OBJECTIF DU TP

L'objectif principal de ce tp est de développer un chatbot simple qui interagit avec les utilisateurs dans le domaine spécifique des jeux vidéo. Nous nous appuierons sur l'API de ChatGPT pour générer des réponses.

IL FAUT SAVOIR QUE

- Le langage de programmation Python sera utilisé pour implémenter le chatbot.
- Étant donné que le chatbot interagira avec l'API de ChatGPT via des requêtes HTTP, il faut donc savoir quelques bases sur la manière d'effectuer des requêtes HTTP.
- Il faut avoir accès à l'API de ChatGPT ou à une API de traitement du langage naturel (NLP) similaire pour pouvoir envoyer des requêtes et recevoir des réponses du modèle de langage.

CHOIX DU SUJET

Nous avons opté pour les jeux vidéo comme domaine d'intérêt, compte tenu de leur popularité et de leur diversité.

CONFIGURATION DE L'ENVIRONNEMENT

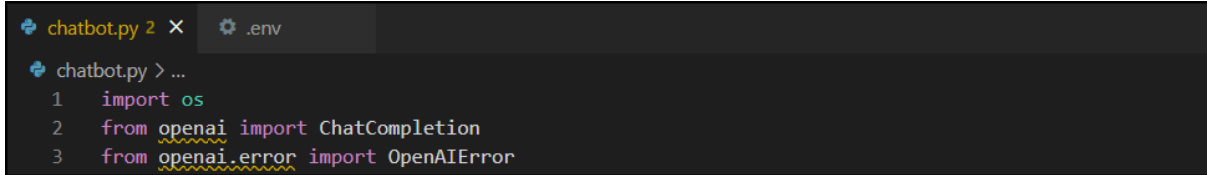
Nous nous assurons d'abord d'avoir Python installé sur notre système, ainsi que les bibliothèques nécessaires pour le tp.

ETAPES DU DEVELOPPEMENT

Notre projet consiste à créer un chatbot fonctionnel qui peut interagir avec les utilisateurs. Nous devons écrire un script Python qui permet de saisir une question de l'utilisateur, d'envoyer cette question à l'API de ChatGPT et d'afficher la réponse obtenue à l'utilisateur. Il est important de prendre en compte la gestion des erreurs de requête et le nettoyage des réponses pour une meilleure lisibilité.

IMPORTATION DES MODULES

Nous importons les modules nécessaires à notre script Python. Le module `os` est utilisé pour interagir avec le système d'exploitation, tandis que le module `openai` nous donne accès à l'API de ChatGPT. L'importation de l'exception `OpenAIError` à partir du module `openai.error` nous permet de gérer les erreurs liées à l'API.



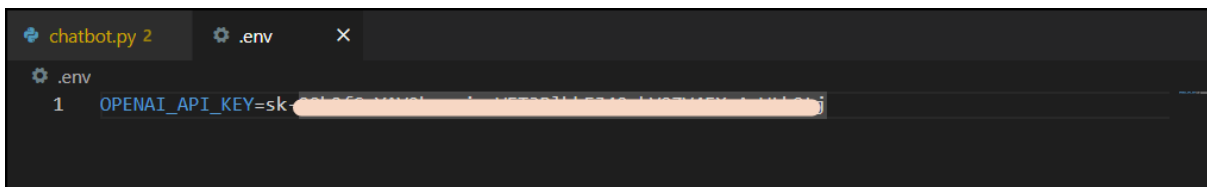
```

chatbot.py 2 x .env
chatbot.py > ...
1 import os
2 from openai import ChatCompletion
3 from openai.error import OpenAIError

```

CONFIGURATION DE LA CLE API

Nous avons configuré notre script pour récupérer la clé API à partir d'un fichier .env, qui stocke des variables d'environnement sensibles. Cette approche garantit que la clé API reste confidentielle et n'est pas exposée dans le code source.



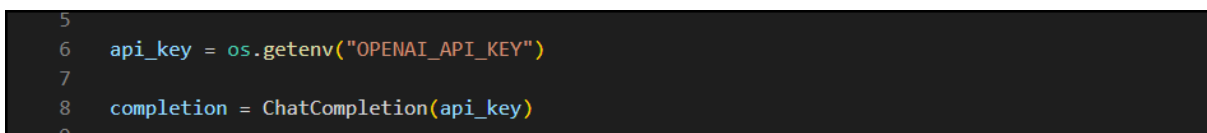
```

.env
1 OPENAI_API_KEY=sk-

```

INITIALISATION DU CLIENT OPENAI

Une fois que la clé API est récupérée du fichier .env, nous initialisons le client OpenAI en utilisant cette clé. L'initialisation du client nous permettra d'envoyer des requêtes à l'API de ChatGPT pour générer des réponses.



```

5
6 api_key = os.getenv("OPENAI_API_KEY")
7
8 completion = ChatCompletion(api_key)
9

```

FONCTION POUR LE CHATBOT

- La fonction principale chatbot_jeux_video() gère l'interaction entre l'utilisateur et le chatbot en lui permettant de poser des questions sur les jeux vidéo. Le chatbot accueille l'utilisateur avec un message de bienvenue et entre ensuite dans une boucle d'interaction où il attend que l'utilisateur saisisse une question.
- Le chatbot utilise la fonction send_question(question) pour envoyer cette question à l'API de ChatGPT. Cette fonction envoie la question à l'API en utilisant le modèle de langage gpt-3.5-turbo, qui est capable de produire des réponses sur une variété de sujets. Une fois la réponse reçue de l'API, elle est renvoyée au chatbot pour être affichée à l'utilisateur.

EXECUTION DU CHATBOT

Pour exécuter notre chatbot, nous utilisons une clause `if __name__ == "__main__":` pour vérifier si le script est exécuté en tant que programme principal. Si c'est le cas, nous appelons la fonction `chatbot_jeux_video()` pour démarrer le chatbot. Cela lance notre chatbot et permet à l'utilisateur de poser des questions sur les jeux vidéo. L'utilisateur peut interagir avec le chatbot jusqu'à ce qu'il décide de quitter en arrêtant l'exécution du script.

```

chatbot.py 2 X .env
chatbot.py > _
1 import os
2 from openai import ChatCompletion
3 from openai.error import OpenAIError
4
5
6 api_key = os.getenv("OPENAI_API_KEY")
7
8 completion = ChatCompletion(api_key)
9
10 predefined_responses = {
11     "Quel est le jeu le plus populaire en ce moment ?": "Le jeu le plus populaire en ce moment est Fortnite, avec des millions de joueurs actifs chaque jour.",
12     "Quels sont les meilleurs jeux de rôle (RPG) sortis cette année ?": "Parmi les meilleurs RPG sortis cette année, on trouve Cyberpunk 2077, Final Fantasy VII Remake et Persona 5 Royal.",
13     "Quels sont les jeux indépendants les plus appréciés par les joueurs ?": "Certains des jeux indépendants les plus appréciés incluent Hollow Knight, Celeste et Stardew Valley."
14 }
15
16 def chatbot_jeux_video():
17     print("Bienvenue dans le chatbot jeux vidéo ! Posez vos questions sur les jeux vidéo.")
18     while True:
19         question = input("Votre question : ")
20         if question in predefined_responses:
21             print("La réponse est : " + predefined_responses[question])
22         else:
23             response = send_question(question)
24             print("La réponse est : " + response)
25
26 def send_question(question):
27     try:
28         completion_response = completion.create(
29             model="gpt-3.5-turbo",
30             messages=[
31                 ("role": "system", "content": "Tu es un assistant de jeux vidéos."),
32                 ("role": "user", "content": question),
33             ]
34         )
35
36         response = completion_response.choices[0].message.content
37         return response
38     except OpenAIError as e:
39         print("Une erreur s'est produite lors de la communication avec l'API : ", e)
40         return "Désolé, une erreur est survenue. Veuillez réessayer plus tard."
41
42 if __name__ == "__main__":
43     chatbot_jeux_video()

```

RESULTAT

Grâce à la compréhension des prérequis, à la configuration de l'environnement de développement, à la compréhension de l'API et à la création du chatbot de base, nous avons pu créer un agent conversationnel capable d'interagir de manière efficace et pertinente avec les utilisateurs. En envisageant des améliorations potentielles, nous pourrions continuer à développer et à perfectionner notre chatbot.

```

PS C:\Users\Sacha\OneDrive\Bureau\Examen ML> python .\chatbot.py
Bienvenue dans le chatbot jeux vidéo ! Posez vos questions sur les jeux vidéo.
Votre question : Quel est le jeu le plus populaire en ce moment ?
La réponse est : Le jeu le plus populaire en ce moment est Fortnite, avec des millions de joueurs actifs chaque jour.
Votre question : Quels sont les jeux indépendants les plus appréciés par les joueurs ?
La réponse est : Certains des jeux indépendants les plus appréciés incluent Hollow Knight, Celeste et Stardew Valley.

```