Système de Chatbot de santé basé sur GPT-2

Roberto PETOH TSENE[1] and Walid YAICI[2]

Université Paris Saclay, 78000 Versailles, France
https://www.universite-paris-saclay.fr/

Keywords: GPT 2 · Chat bot · Fine Tuning. · Medical chat bot

1 Répertoire Github

Voici le lien vers le répertoir github de notre projet: GitHub Repository.

2 Introduction

L'avènement de l'intelligence artificielle a ouvert de nouvelles perspectives passionnantes dans le domaine de la santé, transformant la manière dont nous abordons la prévention, le diagnostic et le suivi des conditions médicales. Dans cette ère de progrès technologiques rapides, la mise en œuvre de chatbots alimentés par des modèles de langage avancés représente une frontière prometteuse pour améliorer l'accessibilité et l'efficacité des services de santé. Le présent article détaille le processus de fine-tuning du modèle GPT-2 (Generative Pre-trained Transformer 2) dans le contexte spécifique de la santé, aboutissant à la création d'un chatbot novateur destiné à fournir des informations personnalisées et des réponses contextuelles dans le domaine de la santé. Cette initiative vise à exploiter le potentiel de l'IA pour renforcer l'autonomie des individus dans la gestion de leur santé, tout en suscitant des discussions importantes sur les enjeux éthiques et les perspectives futures de l'utilisation de telles technologies dans le secteur médical.

3 GPT2 Fine tuning

Nous avons fine tuné le modéle GPT2 [3] en s'inspirant de la méthode de l'article Exploring Language Models for Medical Question Answering [1]. Cette derniére consiste a concaténé la question avec la réponse et de passer cette derniére comme données d'entrainement du modéle. ca permet au modéle de faire le lien et de contextualiser la question et la réponse . Étant donné que gpt2 est un modéle de generation de text. durant le test quand on lui donnera une question il nous genera une réponse qui répond au mieux à cette question.

3.1 Le jeu de données

Notre jeux de données est Medquad dekaggle. MedQuAD comprend 47 457 paires de questions-réponses médicales créées à partir de 12 sites web des NIH (par exemple cancer.gov, niddk.nih.gov, GARD, MedlinePlus Health Topics). La collection couvre 37 types de questions (par exemple, traitement, diagnostic, effets secondaires) associées à des maladies, des médicaments et d'autres entités médicales telles que des tests. Le dataset contient 4 colonnes

- Question : contenant une question de santé
- Answer : la réponse à la question
- Source : la source de laquelle l'information a été extraite
- Focus area : le domaine concerné par la question

Dans notre cas, nous utiliserons uniquement les colonnes Question at answer.

3.2 Le modèle de base

Le transformateur de modèle GPT2 [3] avec une tête de modélisation du langage au-dessus (couche linéaire avec des poids liés aux embeddings d'entrée).

Ce modèle est une sous-classe de PyTorch torch.nn.Module et s'utilise comme un module PyTorch normal.

3.3 Configurations de fine tuning

Nous avons 80% du dataset pour le training. Nous avons concatené chaque question et la réponse correspondante. Le training a duré deux époques avec un batch size de 32.

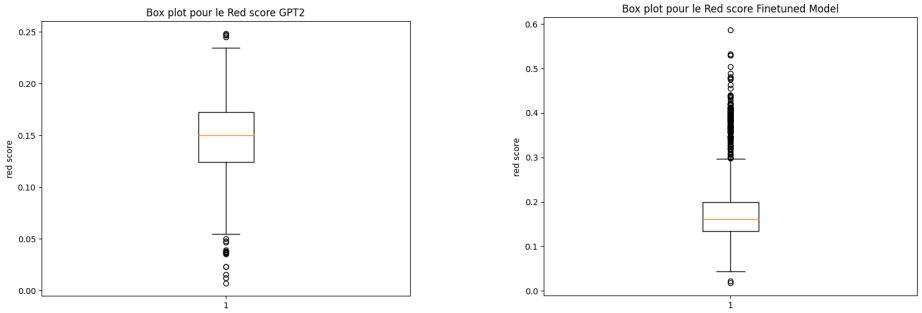
4 Métriques de test

4.1 Blue Score

Le score BLEU (Bilingual Evaluation Understudy) est une mesure couramment utilisée pour évaluer la qualité d'un texte généré par une machine, en particulier dans le contexte du traitement du langage naturel et de la traduction automatique. Il mesure la similarité entre le texte généré et un ou plusieurs textes de référence. Le score est basé sur la précision des n-grammes (séquences de mots) dans le texte généré par rapport au texte de référence, fournissant une mesure quantitative de l'alignement du texte généré avec les références produites par l'homme.

4.2 Red Score

ROUGE (Recall-Oriented Understudy for Gisting Evaluation) est une famille de mesures utilisées pour évaluer la qualité des résumés automatiques produits par des systèmes de traitement automatique du langage naturel. ROUGE se concentre principalement sur la comparaison entre les résumés automatiques et les résumés de référence, en mesurant le chevauchement des n-grammes, des séquences de mots et des phrases entre les deux.

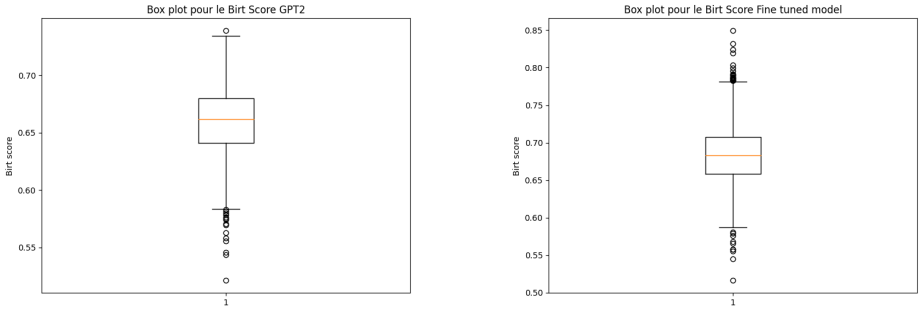


(a) Red score du modèle de base. (b) Red score de notre modèle

Fig. 1: Comparaison du Red score du modèle de base et notre modèle

4.3 Bert Score

Le score BERT (Bidirectional Encoder Representations from Transformers) est une mesure conçue pour évaluer la qualité du texte généré en le comparant directement à des références humaines. Contrairement aux mesures traditionnelles qui s'appuient sur les chevauchements de n-grammes, le score BERT utilise des modèles BERT pré-entraînés pour mesurer la similarité des représentations des mots dans le texte généré et dans le texte de référence, offrant ainsi une évaluation plus nuancée et tenant compte du contexte.



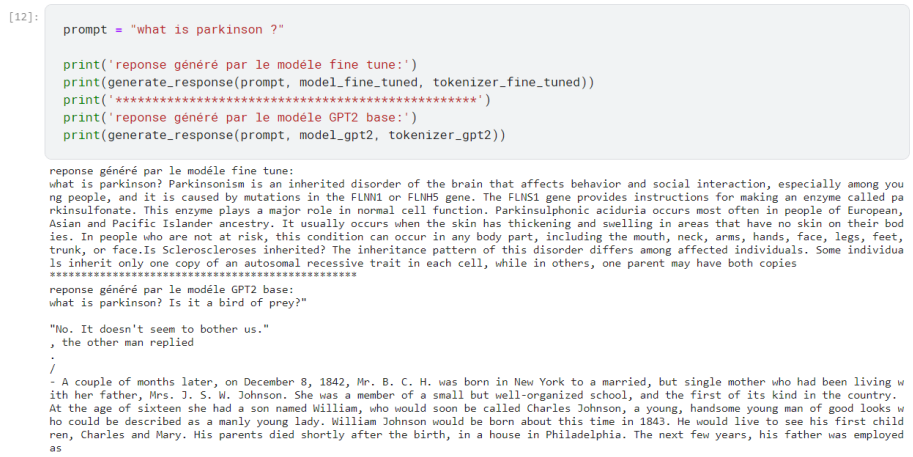
(a) Bert score du modèle de base. (b) Bert score de notre modèle

Fig. 2: Comparaison du Bert score du modèle de base et notre modèle

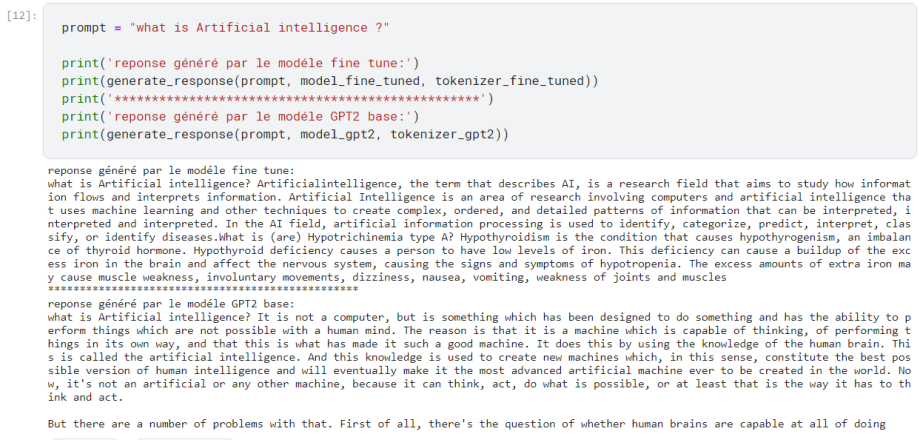
4.4 Human Score

Le score humain fait référence à l'évaluation des résultats du modèle par des annotateurs humains ou des experts. L'évaluation humaine est cruciale pour évaluer

les aspects de la génération de texte que les mesures automatisées peuvent ne pas saisir pleinement, tels que la fluidité, la cohérence et l'adéquation globale du contenu généré. Les notes humaines fournissent des informations précieuses sur les aspects subjectifs de la qualité de la langue, garantissant que les résultats du modèle correspondent aux attentes et aux préférences humaines.



(a) Test avec une question de santé.



(b) Test avec une question en dehors de la santé.

Fig. 3: Comparaison des résultats avec une question de santé et non

4.5 LLM Score

La metrique des LLM se base sur l'evaluation donnée par un LLM.
C'est l'evaluation qui se rapproche le plus de l'evaluation humaine. Mais cette derniére se fait de maniére automatique.

Dans notre cas nous avons utilisé GTP3, l'evaluation consiste à donner au modéle la question avec la réponse idéal et ensuite lui donner deux proposition de text généré, une avec gpt2 et une autre avec gpt finetuné et on lui demande de choisir la réponse la plus pertinente.
Cette metrique est celle qui se raproche le plus de l'evaluation humaine et qu'on peut faire automatiquement.

Pour notre test on a testé sur 100 questions différentes du dataset et pour chaque question nous avons demandé au LLM GPT3 de choisir la méilleur réponse à la question en se basant sur la réponse de référence. Ensuite on fait une moyenne du nombre de fois ou le modéle à privilegié la réponse donnée par le modéle fine tune.
Dans notre cas dans 77% du temps, GPT3 a trouvé notre réponse plus pertinente.
Voici le pseudo algorithme: Algorithme1.

Algorithm 1 Comparaison de réponses générées par des modèles de langage

1: **Input:** $client, question, reference, ft\_generate, gpt2\_generate$
2: $prompt\_template \leftarrow$ "Question: question
Reference answer: reference
Answer #1: ft_generate
Answer #2: gpt2_generate
Given the question and the reference answer, state whether Answer #1 or Answer #2 provides the more accurate answer that is the most closer to Reference answer.
if Answer #1 return Answer #1,
else if Answer #2 return Answer #2,
you have to give an answer"
3: $response \leftarrow client.chat.completions.create(\ model = "gpt-3.5-turbo",\ prompt = prompt\_template,\ temperature = 0)$
4: **Output:** $response.choices[0].text.strip()$

5 Interpretation des résultats

Le tableau 1 résume les résultats Obtenu avec les différentes métriques.

Pour ce qui est des metriques syntaxique, c'est a dire, le Blue et Red Score, les résultats obtenus par le modéle GPT2 de base ou bien le modéle fine tune

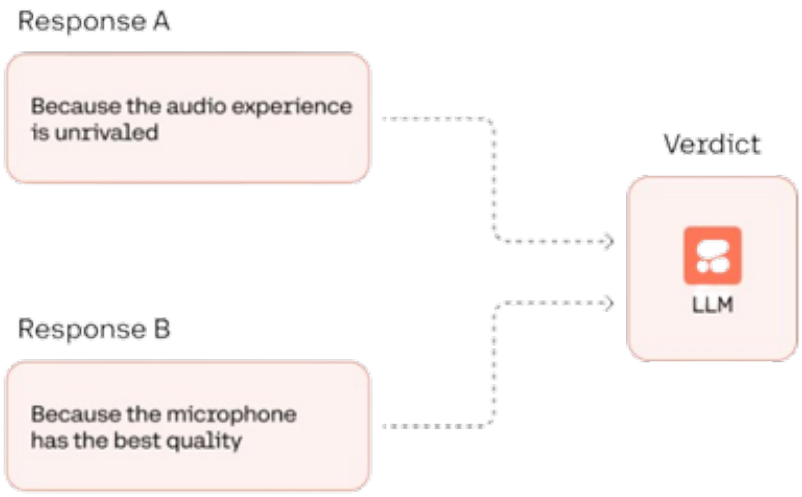


Fig. 4: LLM score

sont trés similaires avec un légrer avantage pour notre modéle sur le Red Score. Ceci veut dire que syntaxiquement parlant on ne peut pas juger quel modéle est le meilleur.

Pour ce qui est des métriques sémntiques, à savoir le score de Bert et le LLM score, on remarque que notre modéle pérforme mieux.
En effet pour ce qui du score de Bert, celui ci permet de dire que notre modéle génére des réponse qui sont sémantiquement plus proches de la réponse références, en se basant sur les embeddings générés par un modéle de Bert et en comparant ses derniers avec les embeddings de la réponse de référence.

Pour ce qui est du LLM Score, c'est le score qui se rapproche le plus d'une evaluation humaine.
Le fait est que GPT3 à estimé que dans 77% des cas notre modéle génére des réponse plus pertinente que le modéle de base, permet d'affirmer que notre modéle arrive à mieux répondre aux questions en relation avec le domaine medical.

Pour ce qui est de l'evaluation humaine, nous avons remarqué que comme le predisait le LLM score notre modéle arrive à mieux répondre aux questions médicales.
Cepedant, nous avons aussi remarqué que pour les questions générales (hors domaine medical), le modéle GPT2 de base fournissait une méilleur réponse.

Ceci est du au fait que les réponses de notre modéle aux questions générales introduisent souvent des notions médicales qui ne sont pas forcément pertinentes. C'est la qu'on peut voir l'influence qu'a eux le fine tuning du modéle GPT2 de Base.

Table 1: Comparaison des résultats entre le modèle de base et le notre.

Métriques	Modèle de base	Notre Modèle
Blue score	0.45	0.445
Red Score	moy: 0.145 med: 0.150 std: 0.037	moy: 0.186 med: 0.161 std: 0.090
Bert score	moy: 0.658 med: 0.661 std: 0.0306	moy: 0.686, med: 0.683, std: 0.043
LLM score	33%	77%

6 Chatbot

Nous avons conçu une bibliothèque pour notre système de chatbot avec streamlit [2]. Streamlit est une bibliothèque open source en Python qui permet de créer rapidement des applications web interactives pour l'analyse de données et la visualisation.

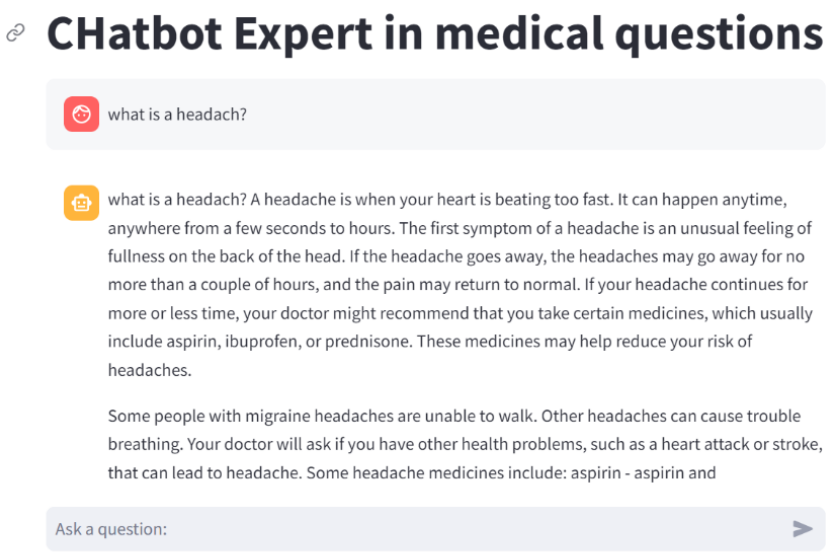


Fig. 5: Interface du chatbot

7 Conculsion

Pour conclure on peut dire que la tache de fine tunning du modéle GPT2 sur un dataset du domaine medical, et la creation d'une interface pour interagir avec le modéle à permit d'avoir un chatbot pertient qui répond aux questions du domaine medical avec une meilleur précision que le modéle de base. Cependant nous avons soulignée les limites de la méthode de fine tunning que nous avons utilisé en ce qui concerne les questions générales.
Une perspective pour améliorer ce modéle serait donc d'utiliser la technique LORA (Low-Rank Adaptation) [5].

References

1. Niraj Yagnik : Exploring Language Models for Medical Question Answering systems. 21 Jan 2024 https://doi.org/2401.11389v1 [cs.CL]
2. `https://streamlit.io/`
3. Page d'accueil PyTorch GPT2 Hugging Face, `https://huggingface.co/transformers/v3.0.2/model_doc/gpt2.html`
4. Page d'acceuil Medquad dataset, `https://www.kaggle.com/datasets/jpmiller/layoutlm`
5. Presentation de la technique LORA, `https://huggingface.co/blog/lora`