InPars

1. Steps

1. Positive query generation via GPT-3.5-Turbo ✅
    - zero-shot using 1k sampled docs with seed 20
    - **prompt**: *Generate a short and objective query in the way a human user would in search engines (based on trec-covid dataset) that would help him find more information about the main topic on the following document:*
2. Negative query generation via BM25 ✅
    - for each query obtained via GPT-3.5-Turbo (previous step), the top 1k documents were retrieved via BM25, and then 5 retrieved docs were random chosen to compose the collection of negative documents.
3. Binary Classifier Training 😵
4. Two-phase Re-Ranking: BM25 + ranking of relevant documents via the score of the classifier trained in step 3. 😵

Note: The symbols ✅ and 😵 are used to indicate completion and incomplete tasks, respectively.

2. Concepts

- Augmented data (queries) using LLMs
  - prompt engineering
- Fine-tune using synthetic data (queries)

3. Tricks

- used `getpass()` (didn't know) or simple `input` for keys
- LangChain FTW! ✌️
- used chatgpt do generate some prompts that would generate good queries
  - Sure, here are some prompts that can be used to instruct ChatGPT to generate synthetic queries for documents:
    1. "Can you generate queries that would help a user find information related to this document's content?"
    2. "What are some alternative ways to ask for information covered in this document?"
    3. "What are some common search queries that someone might use to find this document?"
    4. "Can you suggest queries that would help a user identify the main points covered in this document?"
    5. "Can you generate queries that would help a user connect this document to other relevant resources on this topic?"

4. Interesting/Unexpected

- chatgpt was unable to generate good code snippets for langchain since it doesn't know about it
- low cost with gpt-3 💰
- prompt tuning 🔧
- **Borela** - interesting analysis on presentation

4.1. Prompt

prompt:

```
Generate a short and objective query in the way a human user would in search engines (based on trec-covid dataset) that would help him find more information about the main topic on the following document:
```

title: Automatic Detection and Quantification of Tree-in-Bud (TIB) Opacities from CT Scans

text: This study presents a novel computer-assisted detection (CAD) system for automatically detecting and precisely quantifying abnormal nodular branching opacities in chest computed tomography (CT), termed tree-in-bud (TIB) opacities by radiology literature. ...

query from gpt-3.5-turbo:

```
What is the Automatic Detection and Quantification of Tree-in-Bud (TIB) Opacities from CT Scans and how does the developed CAD system work?
```

ps: tried changing the prompt so it would generate queries more similar to the trec-covid dataset by adding "(based on trec-covid dataset)", generate longer queries in general.

5. A "basic" doubt that you or your colleagues may have

- is it a good idea to filter documents based on their relevant queries in some way to improve the synthetic data?