Nexthink AI Software Engineer assignment

First, we thank you for the time and effort you will invest in this assignment. We believe that this exercise will help us understand your technical skills and expertise in the field of artifical intelligence and machine learning.

You have seven days to complete the assignment. We estimate that the work required should take only a few hours at most. A jupyter notebook is provided as a starting point, but you are free to use any other format.

Please pay attention to the code and the description of your approach and results. The code should be well-documented and easy to understand. Additionally, we expect that you will provide a clear description of your approach to the problem and the rationale behind your methodology. In case of doubt, you can take any assumption you think is reasonable but make sure you explain it well. Lastly, we encourage you to present your results in an easy-to-understand manner, including visualizations and other appropriate metrics.

We wish you the best of luck in completing the exercises, and we look forward to reviewing your submission.

1. Uncovering topics behind news articles

You are a tech-savvy journalist tasked with classifying news articles into categories. To save time, you decide to use your machine learning skills to automate this process.

Dataset

We provide a dataset (`news.jsonl`) containing around 210k news headlines between 2012 and 2022 from HuffPost. It contains the following attributes:

- `link`: link to the original news article.
- `headline`: the headline of the news article.
- `category`: category in which the article was published.
- `short_description`: Abstract of the news article.
- `authors`: list of authors who contributed to the article.
- `date`: publication date of the article.

The objective is to automatically determine the categories behind news articles. You will solely make use of the `headline` attribute **OR** the `short_description` attribute.

> For both parts below we expect you to explain and show that your solution works as expected (e.g., through metrics on a test dataset)

```
import pandas as pd
news_df = pd.read_json("data/news.jsonl", lines=True)
news_df.head()
```

Known categories

You will first assume that you know the categories (e.g., the unique values of the `category` attribute). Train a model able to correctly classify the headline or description of news articles into the correct category.

```
# TODO: provide your solution here.
```

Unkown categories

You will next assume that the categories are unknown (e.g., you are NOT allowed to use the `category` attribute). However, you CAN assume that the number of categories is known.

Your solution should: - Identify news headlines that belong to the same category - Provide a human-understandable representation of each category

```
# TODO: provide your solution here.
```

2. Detecting complex query operations in natural language questions

You are building a pipeline that translates user-questions in corresponding SQL queries. As part of this pipeline, you need to build a classifier that detects complex query operations in a question.

Dataset

We provide an excerpt of the Spider dataset containing around 1k pairs of natural language questions and their corresponding SQL query. It contains the following attributes:

- `question`: the natural language question.
- `query`: the SQL query corresponding to the question.
- `col_names`: description of the content of the columns
- `col_names_original`: names of the columns in the database schema
- `has_join`: if the query contains a join.
- `has_groupby`: if the query contains a "group by" operation.
- `has_orderby`: if the query contains an "order by" operation.

```
import pandas as pd
fname ="./data/queries.json"
df = pd.read_json(fname)
df.head()
```

Objectives

1) Choose **one** of the "join", "group by" or "order by" operators, and build a binary classifier that, given a natural language question, detects if the

corresponding query will use this operator. You may NOT use the "query" field.

We expect you to explain and show that your solution works as expected (e.g., through metrics on a test dataset)

2) Describe in details how you would proceed if you had to identify which column from the schema (provided in the `col_names_original` in the dataset) needs to be put under the "group by" or "order by" statement. You can implement your solution but it is not mandatory