MBA AI Y1 - 2025

Language Technology - Group Assignment

Data

All the data is located in this shared folder on SurfDrive:
`https://surfdrive.surf.nl/files/index.php/s/hmLgIqc74L6IJ9W`

You are given access to 3 datasets:

- Malawi: Original dataset from Claudia Orellana-Rodriguez, a collection of news articles written in English, reporting on the 2019 floods in Malawi
- Amazon Fine Food Reviews: Original dataset from Kaggle, a collection of customer reviews of food-related items sold on Amazon
- EN_Crawl: Original dataset from Common Crawl, a collection of web pages automatically crawled by a web indexer. This sample contains (mainly) websites in English. **Warning**: unfiltered content

You are also provided with the text extract of 2 published papers, you don't need to download the PDF and extract the text.

- Attention is all you need: Vaswani et al., "Attention is all you need", 2017
- BERT Devlin et al., BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, 2018

You are given access to a Milvus datastore:

- Cahiers du Football: Chunks of articles from the website Les Cahiers du Football, embedded with OpenAI `text-embedding-3-small`

Finally there is a notebook named `LLM_endpoint_STUDENT.ipynb`, showing you how to connect to the GenAI models and Milvus Datastore.

Question:	1	2	3	4	Total
Points:	25	25	25	25	100
Score:					

25 1. In this question, you will work with lexical representations of text, study the effect of pre-processing on the vocabulary.

Follow these instructions to answer the questions:

1. There are 3 datasets:
   - Amazon Reviews: Load the dataframe from the file `amazon-reviews.parquet`
   - Malawi: Load the dataframe from the file `malawi.parquet`
   - EN Crawl: Load the dataframe from the file `en_crawl.parquet`
2. Use `CountVectorizer` and `TfidfVectorizer` from `sklearn`
3. Use regular expression for pattern filtering

Answer the following question with the Amazon Reviews dataset:

Q1 (10pts) What is the size of vocabulary with the following pre-processing: lowercase and lemmatization with SpaCy `en_core_web_sm` ?

Answer this last question with the Amazon Reviews dataset:

Q2 (15pts) Create a machine learning model to predict the review score from the text

- Restrict yourself to 50000 random samples from the dataset
- Make it a binary classification: is the score 5 or not ?
- Create a 2-stage `sklearn` pipeline:
  1. `TfidfVectorizer`: lowercase, filter by document frequency: at least in 5 documents, in less than 90% of the documents
  2. `LogisticRegression`: maximum 1000 iterations
- Explore the following hyperparameters:
  * Logistic Regression regularization parameter `C`
  * Restrict to only tokens made of at least 3 ASCII letters (ie. letters from `a` to `b`)
  * Using TFIDF or only TF (check the documentation for the parameter `use_idf` of the class `TfidfVectorizer`
  * Considering unigrams only, or unigrams + bigrams

25 2. In this question, you will extract entities from text and gather some basic statistics.

Follow these instructions to answer the questions:

1. There is 1 dataset:
   - Malawi: Load the dataframe from the file `malawi.parquet`
2. Use `en_core_web_sm` from `spacy`
3. Use regular expression for pattern filtering

Answer the following questions with the Malawi dataset:

Q1 (10pts) Build a dataframe of all entity categories that appear in the corpus, as well as the number of usage of each category

Q2 (15pts) Build a dataframe of VIPs: identify all the mentions of people in the original corpus, grouped together by their lowercase text. Display the TOP-10 mentioned persons by number of mentions. In the report, comment on those most-mentioned people, the accuracy of the NER model and ways forward to remedy to the issue.

25 3. In this question, you will apply models from the Huggingface Hub, to build a Question-Answering bot based on semantic search and span extraction.

Follow these instructions to answer the questions:

1. Use the published articles BERT: file `bert.txt`
2. Use `en_core_web_sm` from `spacy` for tokenization, sentence tokenization
3. Use `sentence-transformers` for text encoding
4. Use `pipeline` from the library `transformers` for QA
5. Split the article into chunks of 2 sentences
6. Create a vector store, one per chunk, by encoding each chunk with the model `"sentence-transformers/multi-qa-mpnet-base-cos-v1"`
7. For each question, the process is the following:
   - Identify in the article's vector store the 3 chunks that are the most similar to the question (use the built-in method `similarity` of every sentence-transformers model
   - If no chunk has a similarity above 0.3, answer `"I can't answer this question"`
   - Use these chunks as the context for 2 different QA models: `"distilbert/distilbert-base-cased-distilled-squad"` and `"deepset/tinyroberta-squad2"`

Here are the questions:

ID	**Question**	**Article**	**Answer**
B1	What kind of attention does BERT use ?	BERT	”cross-attention” ”self-attention used within a single sequence and cross-attention”
B2	Is it difficult to fine-tune ?	BERT	fine-tuning is relatively inexpensive
B3	How much chocolate does it need ?	BERT	I can't answer this question

Table 1: Questions

Although there are ”expected answers” provided, you will not be graded on the capacity of the model to deliver exactly these answers. When your model is unable to provide a reasonable answer, it indicates that there is something to check either in the chunking (all of the text is in chunks?), or the similarity method, or the selection of the 3 most relevant chunks, etc.

25 4. In this question, you will use Generative AI to build a chatbot grounded in articles from a french website about football.

Pre-requisites:

- Get your group API KEY
- Run the notebook `LLM_endpoint_STUDENT.ipynb`
- Check that all cells run properly

The website Les Cahiers du Football is a french website about football with a large collection of articles on topics around the sport itself, its history including major events and players, its place in society and culture, its links with the business world.

For this exercise, neither prior knowledge of the French language nor a prior interest in football is required. In fact, you will see how you can explore a corpus in an unknown language by using Generative AI and your own language.

In this exercise, you will build a RAG-based chatbot that will answer the questions of users in any language, based only on the content of this website.

You are provided with the following elements:

- Access to multiple Generative AI models through a single API endpoint (see notebook)
- Access to a text embedding endpoint through an API (see notebook)
- Access to a Milvus vectorstore (see notebook)
  - Articles from the website have been crawled
  - These articles have been split into chunks of approximately 400 words
  - These chunks have been embedded using `text-embedding-3-small` with the embedding endpoint

You do not need to encode the chunks of the articles, this is already done. The Milvus vector store you have access to contains already 62068 embedded chunks, together with the original text and some metadata, such as the URL of the article the chunk comes from, the title of this article.

Prepare your chatbot in the notebook:

- Setup the system prompt to check whether the user query relates to football or not
- If not about football, decline politely
- Use an LLM to separate the user query in 2 parts:
  1. The question itself
  2. The formatting instructions
- Use an LLM to rephrase the question, consider multiple facets
- Retrieve relevant chunks relative to the question
- Use the question, the relevant chunks, the formatting instructions and deliver an answer that cites the sources

You can try the chatbot on questions like the following ones, that you can reword in your own mother tongue:

- Tell me in 4 bullet points what are the issues of video assisted refereeing
- Tell me 3 things I should know about Johan Cruyf (incorrect spelling is intentional)

In the report, make observations on the capacity to provide answers, on the variations you observe with different languages: what is the tone? how formal is the answer? does language affect the formatting?