Theoretical Questions

1. About embedding

- What is an embedding?
  - A vector representation of categorical data that captures the meaning/context of certain elements in the data
- What are the hyperparameters of an embedding layer?
  - input dimension, output dimension (aka embedding dimension)
- Hom many trainable parameters does an embedding layer contain?
  - vocabulary size * embedding dimension ?? hier ben ik nie zeker

2. Describe how “shuffling” a tf.data.Dataset works. What is the effect of the buffer_size parameter?

- Take the first items of the dataset and put them in the buffer of *buffer_size* until it is filled
- Then, when asked for an item, pull a random item out and replace it with a new one.
- buffer_size determines the "randomness" of the shuffeling, a buffer_size must be large enough for effective shuffling

3. About TF-IDF

- What does TF-IDF stand for?
  - Term Frequency-Inverse Document Frequency
- What are the two factors that determine the TF-IDF score of a word in a document in a corpus of documents. Describe qualitatively.
  - The amount of times a certain word occurs in a document = TF
  - The amount of documents that contain that certain word = IDF
  - Qualitatively: TF looks at the importance of a word within a document, while IDF looks at the importance of a word globally across all documents.
- Is TF-IDF a sparse or a dense representation?
  - Sparse
- What is the benefit of using TF-IDF compared to simple word counts?
  - TF-IDF takes into account not only how often a word appears in a document (TF) but also how unique or important it is across all documents (IDF). This helps in giving more weight to words that are important in a specific document but not too common across the entire corpus.
  - Simple word counts might overemphasize common words like "the" or "and" which might not carry much significance in understanding the content. TF-IDF provides a more nuanced measure of the importance of words in a document

4. De rest -> doen in colab en daar checken