IRE - Assignment 2 Report

`TermDocumentMatrixLatentSemanticIndexing.py`

- Takes inputs from the given corpus and makes a matrix of the vocabulary size into the number of documents.
- Then performs the dimensionality reduction on this data to reduce it to a 100-dimension matrix using *singular value decomposition*.
- This is then used to make a tf-idf matrix which can be used for ranking and retrieving in the future. These weights are saved as a file and used in the next section.

`RankingLatentSemanticIndexing.py`

- Takes in the csv that the above-mentioned file generates, and takes in a query, the same operations performed above to preprocess the data are done on the query as well
- Now the dimension of the query is reduced, and we use cosine similarity to check which of the documents is the closest to the given input, and this is ranked highest in the ranking schema.