Paula Farrás 254614
Sergi García 251425

Indexing and Evaluation:

Project Part 2 IRWA

1. Introduction

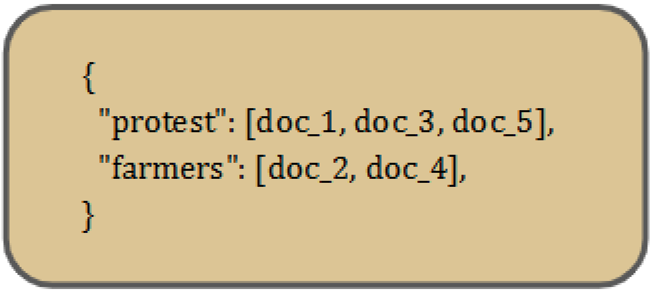
In this second part of the project, we aimed to develop an inverted index to allow efficient tweet retrieval, design and apply some test queries to evaluate their outcome, and rank the retrieved documents using the TF-IDF algorithm. We also evaluated the performance of the search engine using various evaluation metrics and visualized the tweet embeddings into the 2D space. This phase builds on the preprocessing in Part 1, which makes it easier to quickly query the dataset and judge the relevance of the search results.

2. Methodology

2.1 Inverted Index Construction

The inverted index was built to keep track of all term occurrences in the dataset, hence allowing for efficient retrieval while answering a query. The structure of the index is a dictionary, wherein each key is a term and its value is a list of document IDs containing the term.

- The **structure** used is:



```
{
  "protest": [doc_1, doc_3, doc_5],
  "farmers": [doc_2, doc_4],
}
```

- The index was designed to handle conjunctive (AND) queries, where only documents containing all query terms are retrieved.

ASSUMPTIONS	DECISIONS
It was assumed that requiring all query terms to appear in each result would enhance relevance since tweets are usually short and focused on some specific topics. Also, the dictionary structure was assumed to be memory-efficient; sufficient for the dataset size.	We opted for a dictionary-based inverted index because of its efficiency in lookups when processing large datasets. Conjunctive queries are used to improve relevance by ensuring that all terms in the query have to be present in the retrieved documents, given the short nature of tweets.

2.2 Test Queries

We had come up with five custom queries with which to evaluate the performance of the given search engine. The queries were chosen based on high-frequency terms and a relevant topic, namely,

```
"farmers protest",
"indian government",
"support farmers",
"police action",
"human rights"
```

These queries have been designed to represent what typical search intentions might be in the context of our dataset and hence help in performing a relevance and ranking performance test. It could reasonably be expected that high-frequency terms cover the predominant topics of the dataset, thus helping them be used as test queries.

ASSUMPTIONS	DECISIONS
It could reasonably be expected that high-frequency terms cover the predominant topics of the dataset, thus	With terms with both high frequency and relevance, we modeled realistic queries from users.

helping them be used as test queries. We assumed that the tweets that use these terms would provide a representative spread of the relevant ground for testing.	To assess topic coverage, we selected various themes, from broader terms like "protest" to more specific phrases like "support farmers."

2.3 TF-IDF Ranking

Documents are ranked by relevance using the TF-IDF, which rank terms according to the weight assigned to each based on its importance in a document. The higher the TF-IDF score of a document, the more relevant the document is, owing to the fact that the relevant terms appeared less frequently across the corpus and in more documents.

ASSUMPTIONS	DECISIONS
It is believed, as we have stated, that in regard to unique terms in such short texts, TF-IDF would be rather well suited regarding the ranking of tweets. Even so, we assumed that sufficient semantic meaning could express the ranked result in this case to not require any complex embeddings.	Since TF-IDF captures the distinguishing terms occurring in any tweets, the relevant documents would be identified with greater precision. By using TF-IDF to rank documents, we ensured that each query displays its most relevant returns first.

3. Evaluation

Our evaluation approach has two components that consist of baseline queries given in the project instructions and custom test queries. We measured the performance of the search engine using the following metrics:

- **Precision@K** : Measures the percentage of relevant documents among the top K results.
- **Recall@K** : Defines the proportion of the relevant documents that appear within the top K results.

- **Average Precision@K** : Calculation of precision at each relevant document up to K averaged.
- **F1-Score@K** : Scores balance P@K with R@K and provide a unique score for each.
- **Mean Average Precision (MAP)**: Average over several queries of average precision scores at rank K. Computes an overall estimate of each query's precision.
- **Mean Reciprocal Rank (MRR)**: Measures the rank position of the first relevant document and thus greatly favors relevant documents that appear at an early point.
- **Normalized Discounted Cumulative Gain (NDCG)**: Estimates relevance and rank, giving priority to highly relevant documents appearing early.

ASSUMPTIONS	DECISIONS
We assumed that defining relevance as containing all the query terms would simplify the evaluation process and accurately infer user intent. Queries are independent so that constant metrics like MAP and MRR could be used.	We selected our measures to look at precision for accuracy, providing a range of measures for different facets of performance. Baseline queries are used to provide assurance of the consistency in the expected outcomes, while the custom ones ensured the testing of practical relevance.

4. Visualization of Tweet Embeddings

In order to capture the distribution of the tweet dataset, we have visualized the embeddings using T-SNE, whereby T-SNE applies to TF-IDF vectors as well. T-SNE, by performing dimensionality reduction, maps similar tweets into a two-dimensional plot allowing clusters to emerge. This visualization serves to highlight thematic groupings within the dataset, where clusters center on themes such as "farmers protest" or "government criticism."

ASSUMPTIONS	DECISIONS
For us, the T-SNE was meant to find some clustering that may intuitively reflect themes or topics within the given dataset.	We chose T-SNE rather than PCA because it is better at preserving the locality structures in the data, thereby revealing the clusters.

We accepted that the TF-IDF vectors would manage to encapsulate sufficient semantic details specific enough that clustering actually makes sense-to some lesser extent compared to the neural embeddings.	TF-IDF vectors were selected for computation by balancing simplicity and insight into the tweet content.

5. Conclusion

In summary, Part 2 of this project efficiently indexed the tweet dataset, ranked search results using TF-IDF, and evaluated performance through various metrics. The T-SNE visualization allowed us, further, to reveal organic clusters of topical importance in the data. Future work may include testing with several other ranking models or considering adding sentiment analysis for further comprehension of the content of tweets.