Jaume Camps Romaguera - NIA 252021
Arol Garcia Rodríguez - NIA 252718
Albert Garrell Golobardes - NIA 254635

IRWA 2024 Part 2: Indexing and Evaluation

1. Introduction

In Part 2 of our IRWA project, we extended our search engine framework, focusing on core indexing and evaluation functions that improve our ability to retrieve and rank tweets effectively. Leveraging the pre-processed data from Part 1, we implemented an inverted index and a hybrid ranking system. This ranking system uses a TF-IDF score for tweet content and a BERT-based semantic similarity score for hashtags to capture topic relevance. By integrating these two metrics, we aim to provide nuanced, accurate search results that emphasize both tweet content and hashtag relevance. Our repository contains both code and documentation for this segment, tagged for IRWA 2024 Part 2.

2. Indexing

2.1 Data Preparation

We saved and uploaded two JSON files containing the tweets for the output results, and the processed tweets that will be used for the evaluation.

- ➔ '*original_tweets.json*': stores the raw, unprocessed data, retaining tweet text, date, hashtags, likes, retweets, and URLs.
- ➔ '*processed_tweets.json*': stores tokenized and processed tweet content, essential for building the inverted index.

By maintaining both original and processed files, we retain full access to unaltered tweet data while still benefiting from optimized processed data for evaluation and indexing.

2.2 Full Inverted Index Construction

In this section, we detail the construction of a full inverted index focused on tweet content. This index serves as the primary data structure for our search engine, allowing efficient retrieval of tweets based on individual terms within the text. Each term in the tweet content (e.g., words within the tweet text) maps to a list of tweet IDs where that term appears, along with its frequency in each tweet.

To build this index, we first tokenize each tweet into individual terms, excluding stop words and applying lowercasing for normalization (done in the first part). Each term is

then indexed, creating entries in the inverted index with pointers to tweet IDs and corresponding frequency counts. This structure enables quick access to all tweets containing a specific term, facilitating efficient scoring and ranking based on term relevance.

The inverted index construction enables efficient query processing, particularly for text-based searches, by organizing tweets in a way that optimally supports the retrieval and ranking stages of our search engine pipeline.

The full inverted index structure follows this format:

```
{
   Term_id_1: [document_1,[posId0..]], [document_2, [posId3..]]....
   Term_id_2: [document_4,[posId0,posId5..]], [document_10, [posId4..]]....,
   ...
}
```

2.3 Query Processing and Ranking

For ranking, we implemented a hybrid approach that considers both TF-IDF scores and semantic similarity between queries and hashtags.

- ➔ **TF-IDF Score Calculation:** TF-IDF scores capture how frequently query terms appear in the tweet text and how unique these terms are across the dataset. This is effective for determining the relevance of tweet content given a query.
- ➔ **BERT Similarity for Hashtag Relevance** In addition to TF-IDF, we use BERT (Bidirectional Encoder Representations from Transformers) to compute the similarity between query terms and hashtags, focusing on semantic relevance. This approach evaluates the relationship between the query and tweet hashtags in a more context-sensitive way. By eliminating stemming in the query and hashtags, we preserve their semantic integrity, which helps BERT capture context more effectively. For example, multi-word hashtags or queries are treated as complete entities rather than separated tokens, enabling more accurate similarity measurements.

Our final ranking combines both TF-IDF and BERT-based hashtag similarity scores, with alpha and beta parameters controlling the weight assigned to each. This weighted ranking ensures that tweets containing relevant hashtags are prioritized, while still valuing direct content relevance. The inclusion of both content-based (TF-IDF) and context-based (BERT similarity) scoring offers a more comprehensive ranking mechanism that accounts for various types of tweet relevance.

We applied stemming to the main tweet content to unify word forms (e.g., “running” and “run”), which helps reduce noise. However, to preserve the contextual meaning of hashtags, we excluded stemming for hashtags and treated the query in its original form when evaluating hashtag relevance. This decision aligns hashtags and queries, allowing BERT to compare them as complete phrases, which enhances the semantic matching quality.

3. Evaluation

3.1 Ground Truth for Test Queries

To create a benchmark for evaluating the search engine, we selected 150 tweets across five test queries, assigning relevance labels (relevant or non-relevant) to each tweet manually. This binary ground truth helps ensure consistent assessment of search results.

3.2 Evaluation Metrics

We assessed retrieval performance using multiple evaluation metrics that address different aspects of ranking quality. These metrics include:

- ➔ Precision@K (P@K): Measures the proportion of relevant tweets within the top K results.
- ➔ Recall@K (R@K): Measures the proportion of relevant documents retrieved out of the total number of relevant documents in the dataset.
- ➔ Average Precision@K (P@K): Computes the average precision across the top K results, balancing relevance with ranking position.
- ➔ F1-Score@K: The harmonic mean of precision and recall, offering a balanced assessment of relevance and completeness.
- ➔ Mean Average Precision (MAP): Calculates the mean AP across all queries, providing an overall measure of precision.
- ➔ Mean Reciprocal Rank (MRR): Considers the rank position of the first relevant document, helping gauge how quickly users find relevant information.
- ➔ Normalized Discounted Cumulative Gain (NDCG): Evaluates the ranking position of relevant tweets while applying a logarithmic discount, with more weight on earlier positions. This metric reflects how well the search engine surfaces relevant content in the highest-ranked positions. Then it’s normalized with the best possible ranking score.

Together, these metrics give a comprehensive view of our search engine’s retrieval capabilities, from immediate relevance in top-ranked tweets to the distribution of relevance across the ranked list.

3.3 Results for the Baseline Queries

After testing with the predefined queries, we observed the following trends:

- ➔ Tweets with hashtags more closely aligned with the query terms, as measured by BERT similarity, consistently ranked higher.
- ➔ The TF-IDF approach effectively ranked tweets based on content relevance, but integrating BERT similarity allowed us to surface tweets that might have used different terms to describe the same concepts.
- ➔ Our use of combined metrics, particularly the tuning of alpha and beta parameters, demonstrated flexibility in adjusting ranking based on the specific nature of queries.

These results support our dual-metric approach, illustrating that combining TF-IDF and BERT similarity yields a nuanced understanding of tweet relevance, enhancing retrieval quality.

Query 1: "People's Rights"

- ➔ Precision@10: 0.8 shows that 80% of the top 10 documents retrieved were relevant.
- ➔ Recall@10: 0.53 indicates that only about half of all relevant documents were retrieved within the top 10 results.
- ➔ Average Precision@10: 0.91 is high, suggesting that the relevant documents are well-ranked in the top 10, enhancing user satisfaction.
- ➔ F1-score: 0.89, showing a strong balance between precision and recall.
- ➔ NDCG@10: 0.85, reflecting a reasonable ranking order where relevant documents are prioritized.

The results here suggest that while relevant documents are captured, some relevant ones remain lower in ranking. Incorporating BERT similarity likely helped position related concepts higher.

Query 2: "Indian Government"

- ➔ Precision@10: 1.0 demonstrates perfect precision, with all top 10 documents relevant.
- ➔ Recall@10: 0.67 indicates that two-thirds of the relevant documents are retrieved within the top 10.

➔ Average Precision@10, F1-score, and NDCG@10: All at 1.0, signifying perfect rankings in relevance order.

This query benefited from precise term matching, and the high NDCG suggests an optimal document order, likely from strong TF-IDF alignment with clear keyword matches.

Mean Average Precision (MAP) @10: 0.95 reflects generally high relevance for baseline queries across queries.

Mean Reciprocal Rank (MRR) @10: 1.0 implies that the first relevant document appeared at the top position in all queries.

3.4 Results for Our Queries

Query 1: "Are farmers being respected in India?"

➔ Precision, Average Precision, F1, NDCG @10: All at 1.0 indicate that relevant documents dominate the top positions.
➔ Recall@10: 0.67 implies that some relevant documents fell below the top 10.

Query 2: "Is people supporting Farmers?"

➔ Precision@10: 1.0, meaning all retrieved documents are relevant.
➔ Recall@10: 0.91, showing better coverage of relevant documents within the top 10 than other queries.

High F1 and NDCG scores at 1.0 suggest a ranking that effectively prioritizes relevance, with BERT similarity potentially capturing related phrases effectively.

Query 3: "Fight of farmers in India"

➔ Similar to the previous queries, Precision, F1, NDCG, Average Precision@10 all reached 1.0.
➔ Recall@10: 0.67 indicates missed relevant documents beyond the top 10.

The strong performance across precision and ranking metrics shows the strength of the BERT component in matching nuanced phrases and concepts for this query.

Query 4: "Impact of the Protests in India"

➔ Every metric achieved a perfect score of 1.0, showcasing optimal retrieval and ranking.

This query's precision and coverage suggest that the TF-IDF and BERT hybrid model managed both explicit term matching and semantic similarity exceptionally well.

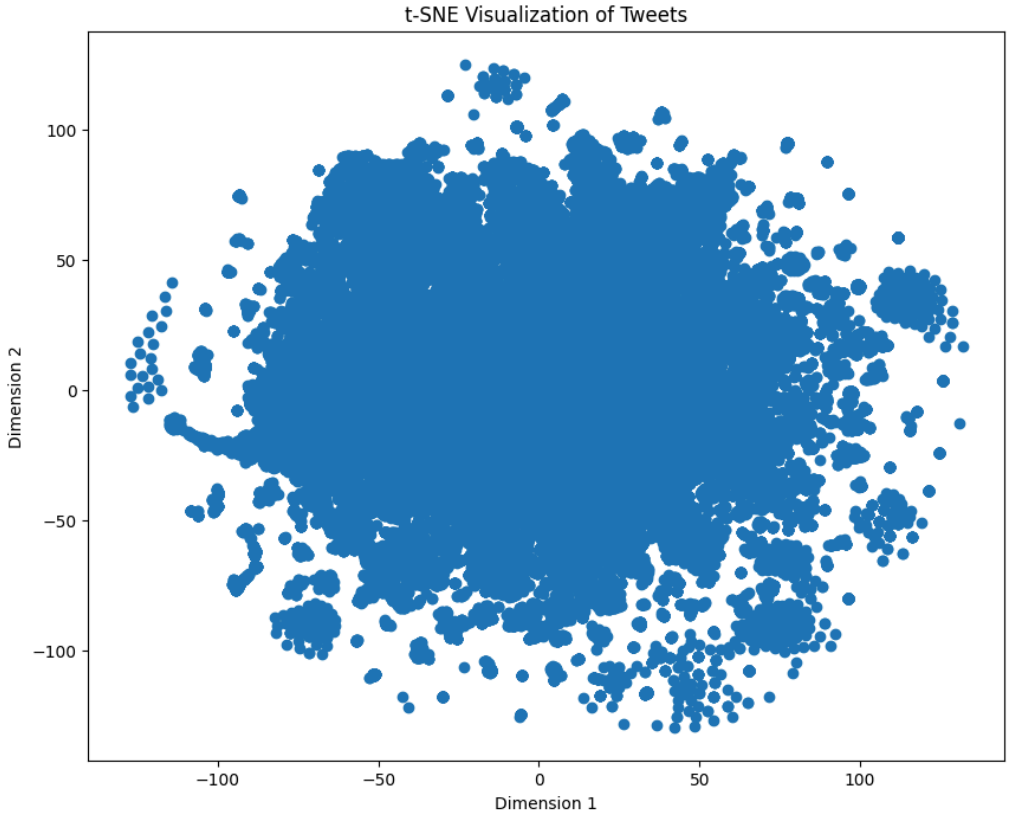
Query 5: "Protest against Indian Government"

- Precision, F1, NDCG, Average Precision@10: All at 1.0, showing consistent top-ranking relevance.
- Recall@10: 0.67 again highlights that some relevant documents appear beyond the top 10.
- Mean Average Precision (MAP) @10: 1.0 across custom queries reflects exceptional ranking relevance, with relevant documents effectively prioritized in the top positions.
- Mean Reciprocal Rank (MRR) @10: 1.0, indicating that the first relevant document is always highly ranked.

The overall high scores demonstrate that the hybrid TF-IDF and BERT-based ranking approach captures both exact term relevance and nuanced, context-driven similarities. Custom queries performed exceptionally well, showing that the BERT integration allowed the engine to recognize contextual meaning across variations in wording. The baseline performed well, though slightly lower on recall and precision, suggesting that queries without semantic nuances benefit from more refined keyword matching.

3.5 Visualization with T-SNE

To inspect clustering behavior in our ranked results, we employed T-SNE (t-distributed Stochastic Neighbor Embedding), which projects high-dimensional tweet vectors into a 2D space. We represent each tweet as the average vector of its words (weighted by TF-IDF) and apply T-SNE to observe clustering patterns. This visualization offers insights into tweet similarity based on their content, allowing us to evaluate how well our model groups related tweets and identify outliers. Clear, cohesive clusters indicate effective indexing and vectorization, while outliers may point to terms or concepts requiring further tuning.



4. Conclusion

Our work on Part 2 of this project demonstrates the potential of combining content-based and context-sensitive ranking metrics in a search engine for tweets. By leveraging TF-IDF for tweet content and BERT similarity for hashtag relevance, we capture both explicit term matching and semantic relevance. The dual-index structure further enables efficient, targeted retrieval across tweet text and hashtags. Through a robust evaluation process, we validated the effectiveness of our ranking approach, confirming that our model can surface relevant content with flexibility and precision.

For additional implementation details and results, please refer to our GitHub Repository and our release tagged for IRWA-2024-part-2.