Social Search on r/advice

Lorenzo Tripepi - 513176
Pietro Scapini - 513351
Vincenzo Perazzoli - 512994

8 July 2025

1 Goal

The objective of this project is to develop a search engine for social search within Reddit. The project aims to:

- Define the search domain
- Identify and collect appropriate data to support social search
- Identify the main dimensions of relevance
- Evaluate effectiveness in terms of relevance and usefulness for users

2 Datasets Construction

We extracted the most popular posts from the **r/advice** subreddit, selecting only those with at least 16 comments. For each post, we chose 16 relevant comments with at least 10 and no more than 150 upvotes, also including metadata. Finally, we constructed the following datasets:

Posts

We collected posts that contained at least 16 comments. Each post is represented by the columns: **post_id**, **post_text** (title of the post) and **post_description** (full text of the post).

Comments

We extracted 16 relevant first-order comments with at least 10 and no more than 150 upvotes. Each comment contains: **comment_id**, **text**, **upvotes**, **replies** (number of direct responses a comment has received), **timestamp** (date of creation), **karma_post** (total score that the original post has earned), **karma_comments** (total score that the comment section of that post has earned)

Qrels

The `qrels` DataFrame contains 200 queries, each associated with 10 comments labeled with relevance levels from 1 to 4. We constructed it as follows:

1. We calculated the Wilson score for every comment associated with each query in the dataset. This score balances high upvote ratios with the total number of votes.

2. We assigned a relevance level (1–4) to each comment based on its Wilson score within the same `post_id` group. This was done by computing quantiles (Q1, Q2, Q3) for each group and assigning relevance levels as follows:
   - Score $\leq$ Q1: low relevance
   - Q1 $<$ Score $\leq$ Q2: moderate relevance
   - Q2 $<$ Score $\leq$ Q3: high relevance
   - Score $>$ Q3: top relevance
3. For each query, the selection is based on upvote count within each level:
   - Level 4 $\rightarrow$ top 2 comments
   - Level 3 $\rightarrow$ top 3 comments
   - Level 2 $\rightarrow$ top 3 comments
   - Level 1 $\rightarrow$ top 2 comments

3 Structure of project

Reddit Dataset Construction

- Creation of posts and comments dataset
- Loading posts and comments dataset
- Creation of qrels dataset

Exploratory Data Analysis

- Distribution of posts length (in words) for *Title* and *Description*
- Distribution of comments lengths (in words)
- Top word frequency in posts (excluding stop words) for *Title* and *Description*
- Top word frequency in comments (excluding stop words)
- Correlation Matrix: Comment Features and Relevance Level

Retrieval Models: BM25 and Neural reranking

- Baseline Retrieval (BM25)
- Neural reranking

Functions for Evaluation Metrics

- Precision and Recall
- MAP
- nDCG

Evaluation Metrics

- **BM25**
  - Precision and Recall,
  - MAP,
  - nDCG
- **Neural Reranking**
  - Precision and Recall,
  - MAP,
  - nDCG

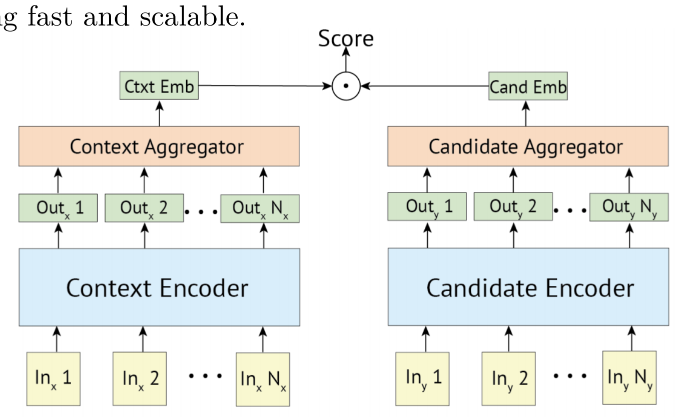
4 Retrieval Models

BM25

BM25 (Best Matching 25) is a ranking algorithm used in information retrieval to evaluate the relevance of a document with respect to a query. It belongs to the family of probabilistic models and is based on the frequency of terms in both the documents and the query.
BM25 penalizes overly frequent terms, preventing common words from dominating the score, and adjusts for document length, so that longer texts are not unfairly favored simply because they contain more words. It is widely used in search engines and information retrieval systems as an effective and fast baseline, thanks to its simplicity and strong performance

Neural Reranking (Bi-Encoder)

A Bi-Encoder is a model that turns a query and a document into separate vectors , then compares them to see how similar they are, in our case using cosine similarity. The query and document are encoded independently, so document vectors can be precomputed and stored. This makes searching fast and scalable.



(a) Bi-encoder

5 Evaluation Metrics

Precision and Recall

Precision: Measures the proportion of retrieved documents that are actually relevant. It evaluates how accurate the system is in returning only relevant results.

Recall: Measures the proportion of relevant documents that have been successfully retrieved. It reflects the system's ability to find all relevant results.

- Precision drops as *k* increases: we retrieve more items, including possibly non-relevant ones.
- Recall increases with *k*: we cover more of the relevant items as we allow more results.
- At around $k = 5$, precision and recall intersect, meaning we hit a balance point between relevance density and coverage.

Mean Average Precision (MAP)

Represents the average of the precision values at the ranks where relevant documents are found, averaged across all queries. It considers both relevance and ranking.
The plot shows that MAP decreases steadily as k increases. This is expected because:

- Average precision rewards relevant items appearing early in the ranking.
- As we increase the cutoff k, we include more documents that may not be relevant or are ranked lower.
- This dilutes the contribution of early hits and lowers the mean score.

Normalized Discounted Cumulative Gain (nDCG)

Assesses the quality of the ranking by giving higher importance to relevant documents appearing at the top. It is normalized to allow fair comparison across different queries.
The plot shows how nDCG changes as we increase the cutoff *k*. The trend is decreasing, which is expected:

- nDCG rewards highly relevant items appearing early in the ranking.
- At $k = 1$, we observe the highest score, indicating that many queries have a highly relevant item ranked first.
- As k increases, the model still retrieves relevant results, but some of them appear in lower positions — reducing their impact on the total score due to logarithmic discounting.

6 Results

BM25 Scores

Precision	Recall	MAP	nDCG
0.15	0.15	0.63	0.31

Neural Reranking Scores

Precision	Recall	MAP	nDCG
0.18	0.18	0.62	0.36

Precision and Recall: The neural reranker shows a slight improvement over BM25, suggesting it retrieves marginally more relevant items in the top-k.
nDCG: The bi-encoder achieves a small improvement in nDCG, indicating a better ranking of relevant items at higher positions.
MAP: Interestingly, BM25 slightly outperforms the neural method on MAP. This may happen because MAP weighs early hits evenly across rank positions and BM25, being keyword-based, might occasionally match high-relevance terms more directly.