IBM WATSON PROJECT

Data Mining

Birtalan Csaba | Blajan Denisa | Holcan Cosmin | Stroe Teodora | Suteu Andreea

INTRODUCTION - SETUP

- **Developed a console application in Kotlin with Lucene integration. The application offers a menu-driven interface with different key functionalities.**

- **The functionalities present in this project are:**

  - **Create Index:** Build an index for efficient data retrieval.

  - **Custom Query Search:** Allow users to input and search using the created index.

  - **Default Questions:** Run predefined questions as part of the project and System Retrieval Comparison: View a comparison between our system retrieval and re-rank results.

INDEXING AND RETRIEVAL

- **English Analyzer**
    - Tailored specifically for the English language.
    - An extension of the StandardAnalyzer.
    - Trims trailing 's from words to ensure consistency.
    - Facilitates stemming using Porter Stemming Algorithm, reducing words to their base or root form.
    - Omits commonly used words that do not contribute to the core meaning, enhancing focus on relevant terms.

INDEXING AND RETRIEVAL

- **Index Refinement:**
  - Transitioned from 'content' to multi-value fields, the 'content' field contains the majority of a Wikipedia page that will be tokenized, as well as a 'category' field which is also multi-value and not tokenized.
  - Evolved from exclusion to inclusion by adapting the 'title' field into a multi-value field to encapsulate redirecting page titles.
- **Query Strategy:**
  - Utilized both 'content' and 'category' fields, including the clue category.
  - Standardized to '<clue> OR <category>', ensuring broad yet focused search.
  - Streamlined by extracting the substring before '(', and purifying clues by removing special characters for clarity.

MEASURING PERFORMANCE

- **MRR Implementation:**
  - Excellently suited for scenarios without a predefined relevant answer list and for queries yielding multiple answers.
  - Effectively accounts for synonymous titles, treating them as a single content source.
  - Increased hits per query from 10 to 30, capturing a broader spectrum of potentially correct documents.
  - If the correct document is beyond the top 30, we assign a reciprocal rank of 0, reflecting its negligible impact on the MRR.
- **In this way, we obtained a MRR of 0.374**
- **In this analysis, we found that in 30 out of 100 cases, the first result proved to be the correct one.**

ERROR ANALYSIS

  - 30% correct, attributed to Lucene's text-matching and the effectiveness of the EnglishAnalyzer in stemming and stop-word removal.
  - 70% incorrect, necessitating a deeper analysis.
- **Categories of Incorrect Answers:**
  - **Partially Correct:** Correct answer within the first 30 results.
  - **Related but Not Exact:** Answers pertain to the correct subject but miss specific details.
  - **Mismatch with Natural Language Clues:** Clues in natural language form don't match with content in the documents.