Data Mining: Default Programming Project

Students:

Zimbru Florin Grigore

Pop Maria Alexandra

Preparation of Terms for Indexing

In our code, the process of preparing terms for indexing Wikipedia content primarily focuses on the basics of text processing and indexing using the Whoosh library. Here's how our code addresses these aspects:

- *Indexing Process*: Our code sets up a Whoosh index schema, which is a crucial step for defining the structure of the indexed data, including fields like titles and content from Wikipedia pages. This schema is defined at the beginning of our code, indicating the fields that will be indexed.
- *Wikipedia-Specific Content Handling*: While our code does not explicitly detail strategies for dealing with Wikipedia-specific issues like disambiguation pages and redirects, such handling is essential for ensuring the quality of our indexed content. Normally, this would involve preprocessing steps to identify and exclude or properly process these types of pages before indexing.

Retrieval Component Implemenation:

Our retrieval system is designed to process Jeopardy clues (and optionally, categories) to find the most relevant Wikipedia page. Here's a detailed breakdown:

- *Query Construction*: Our code constructs queries from the Jeopardy clues by using the entire clue text as input for the search. This approach ensures that all aspects of the clue are considered in the search process, although it does not explicitly filter or select a subset of words for querying. The inclusion of the entire clue is a straightforward method that relies on the search engine's built-in capabilities to match query terms with indexed content.
- *Algorithm for Selecting Subset of Words*: Although our current implementation does not apply a specific algorithm for selecting a subset of words from the clue, this could be an area for improvement. Optimizing query construction by identifying key terms within the clue could enhance the system's ability to retrieve the most relevant Wikipedia page more efficiently.
- *Use of Category Information*: Our code optionally incorporates the category of the Jeopardy question into the search query. This inclusion can help refine the search results by adding context that may narrow down the potential Wikipedia pages to those most relevant to the category in question.

Implementation of P@1 Measurement in Our Code

To implement the measurement of P@1 in our code, we follow these steps:

- Collect Test Data: We compile a set of Jeopardy questions (clues) along with the correct answers (the titles of the corresponding Wikipedia pages). This dataset serves as our test data for evaluating the system.
- Run Queries: For each question in our test dataset, we run a query through our Jeopardy system to retrieve the top-ranked Wikipedia page title.
- Calculate P@1: For each query, we compare the top result to the correct answer. If the top result matches the correct answer, the query scores a 1; otherwise, it scores a 0. The P@1 score for our system is then calculated as the average of these scores across all queries in the test dataset.

  Code Implementation: While the specific lines of code are not provided, this calculation would typically be implemented in a function that iterates over the test dataset, performs the queries, and tallies the number of correct top results. The P@1 score is then computed as the sum of correct scores divided by the total number of queries.

Reporting Performance Using P@1

After calculating the P@1 score for our system, we report this metric as a clear indicator of our system's performance. A higher P@1 score indicates a higher accuracy of the system in correctly identifying the first result, which is our primary goal. The performance report would include the P@1 score along with an analysis of the results, highlighting the system's strengths and areas for improvement.

Based on the specified performance, if our Jeopardy question-answering system found correct answers to 70-75% of the given questions, this indicates a relatively high level of accuracy, especially for a system that may rely on straightforward text matching and retrieval techniques without extensive natural language processing capabilities. Here's how this performance metric fits into our error analysis:

Performance Overview

- *Correctly Answered Questions*: Our system correctly answered between 70-75% of the questions. This high success rate suggests that our indexing and retrieval mechanisms are effective for a significant portion of the Jeopardy clues, particularly those that have direct keyword matches with Wikipedia content.
- *Incorrectly Answered Questions*: Conversely, 25-30% of the questions were answered incorrectly. This portion represents the challenges our system faces, likely due to factors such as question complexity, ambiguity, or limitations in our current retrieval strategy.

Analysis Based on Performance

- **Strengths**:
    - The system's strengths lie in its ability to effectively match clues with Wikipedia content for a majority of the questions. This success rate implies that for many Jeopardy clues, the essential keywords or phrases are sufficiently represented in the Wikipedia page titles or content, allowing for accurate retrieval.
    - The inclusion of question categories in the retrieval process may also contribute to this performance, helping to narrow down the search results to more relevant pages.
- ***Areas for Improvement***:
    - The incorrect answers highlight areas where our system could be improved. For example, enhancing the understanding of question context, disambiguation, and the use of advanced NLP techniques could address the gap in performance.
    - Specifically, addressing the 25-30% of questions that were answered incorrectly would require a deeper analysis of the types of errors (e.g., misinterpretation of the clue, failure to retrieve the correct page due to indexing issues, or limitations in handling complex inferences).
    -

A simple system can successfully answer correct Jeopardy questions primarily due to the direct matching of unique keywords found within the questions to the structured and comprehensive content of Wikipedia. The straightforward nature of many Jeopardy clues, coupled with Wikipedia's detailed articles, allows a basic keyword search algorithm to effectively identify relevant pages. This success is further supported by the structured indexing of Wikipedia, which facilitates quick and accurate retrieval of information that directly corresponds to the clues provided.

However, the system faces challenges with incorrectly answered questions, largely because of ambiguity in the clues, the complexity of certain questions that require advanced reasoning beyond keyword matching, and the lack of context which makes it difficult to disambiguate and accurately match the query with the correct Wikipedia content. Additionally, simple systems struggle with the nuanced understanding of natural language, relying on the explicit presence of keywords without grasping the subtleties of human language or the comprehensive context, leading to inaccuracies in retrieving the correct answers for more complex or nuanced Jeopardy questions.