Introduction

In today's digital age, the vast amount of information available on the internet presents both an opportunity and a challenge for users seeking specific information. Traditional Information Retrieval (IR) systems rely heavily on keyword-based searches, which often fail to capture the nuanced meaning and context behind user queries. As a result, search results may not fully align with user intentions, leading to a suboptimal search experience.

To address this issue, this report presents the design and implementation of a Query Reformulation System aimed at improving the effectiveness of search queries within an IR system by leveraging semantic closeness. Semantic closeness refers to the degree of relatedness between concepts based on their meaning, rather than just their lexical similarity. By understanding and utilizing the semantic relationships between words, the proposed system aims to reformulate user queries to enhance the relevance and accuracy of search results.

Objectives

The primary objectives of this project are as follows:

1. **Semantic Analysis**: Implementing techniques to analyze the semantic content and context of user queries.
2. **Query Reformulation**: Reformulating user queries by incorporating semantically related terms to broaden and refine search scope.
3. **Text Graph Construction**: Building a semantic text graph that represents the relationships between words based on their semantic closeness.
4. **Efficient Document Retrieval**: Utilizing appropriate data structures to ensure swift and accurate retrieval of relevant documents.

This system will employ word embeddings to understand semantic similarities and construct a semantic graph to aid in query expansion. The ultimate goal is to bridge the gap between user intent and search outcomes, enhancing the user experience in information retrieval.

Modules

1. Semantic Analysis

Semantic analysis is the first step in understanding the user's query. It involves breaking down the query into meaningful components and analyzing the relationships between these components. This module includes:

- **Tokenization**: Splitting the query into individual words or tokens.
- **Stop Words Removal**: Eliminating common stop words (e.g., "the", "is", "at") that do not contribute to the query's semantic meaning.
- **Lowercasing**: Converting all tokens to lowercase to ensure uniformity and facilitate comparison.
- **Part-of-Speech Tagging**: Identifying the grammatical parts of speech (e.g., nouns, verbs, adjectives) to understand the structure and context of the query.

2. Query Reformulation

Once the semantic analysis is complete, the next step is to reformulate the query to enhance its relevance. This module involves:

- **Word Embeddings**: Utilizing pre-trained word embeddings (e.g., GloVe, Word2Vec) to understand the semantic similarity between words.
- **Semantic Text Graph Construction**: Building a graph where nodes represent words and edges represent the semantic closeness between words.
- **Query Expansion**: Identifying and adding semantically related terms to the original query to improve the breadth and relevance of search results.

3. Text Graph Construction

The semantic text graph is a crucial component of the system, representing the relationships between words based on their semantic closeness. This module includes:

- **Graph Building**: Using word embeddings to calculate the cosine similarity between word vectors and create edges between semantically related words.
- **Graph Optimization**: Ensuring the graph is efficient and manageable by limiting the number of connections (edges) for each word to the top N most similar words.
- **Graph Storage and Retrieval**: Efficiently storing the graph structure in a way that allows for quick access and updates.

4. Document Indexing

Efficient document retrieval requires an effective indexing system. This module involves:

- **Inverted Index Construction**: Building an inverted index where each word points to the list of documents in which it appears.
- **Document Preprocessing**: Tokenizing and processing each document to extract relevant terms and store them in the index.
- **Stop Words Removal**: Removing stop words from the documents to focus on meaningful terms.
- **Index Optimization**: Implementing techniques to optimize the index for fast search and retrieval operations.

5. Document Retrieval

The final module is responsible for retrieving and ranking the relevant documents based on the reformulated query. This module includes:

- **Query Matching**: Matching the expanded query against the indexed documents to find potential matches.
- **Relevance Scoring**: Calculating a relevance score for each document based on factors such as term frequency, inverse document frequency, and semantic similarity.
- **Result Ranking**: Sorting the matched documents based on their relevance scores to present the most relevant results to the user.
- **Partial Match Handling**: Implementing a fallback mechanism to handle cases where the reformulated query does not yield sufficient results, by allowing partial matches on the original query.

6. Evaluation and Optimization

To ensure the effectiveness of the Query Reformulation System, continuous evaluation and optimization are necessary. This module involves:

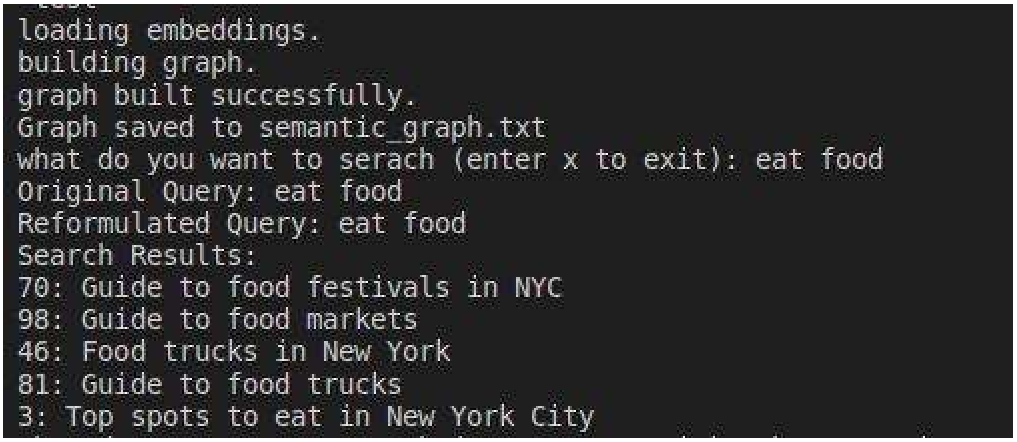
- **Performance Metrics**: Defining metrics such as precision, recall, and F1 score to evaluate the system's performance.
- **User Feedback**: Collecting user feedback to identify areas for improvement and refine the query reformulation process.
- **System Tuning**: Adjusting parameters and algorithms based on evaluation results and user feedback to enhance system performance.

By integrating these modules, the Query Reformulation System aims to significantly improve the effectiveness of search queries in an IR system, leading to more relevant and accurate search results. This comprehensive approach not only addresses the limitations of traditional keyword-based search methods but also enhances the overall user experience in information retrieval contexts.

OUPUT SAMPLES:

OUTPUT 1:

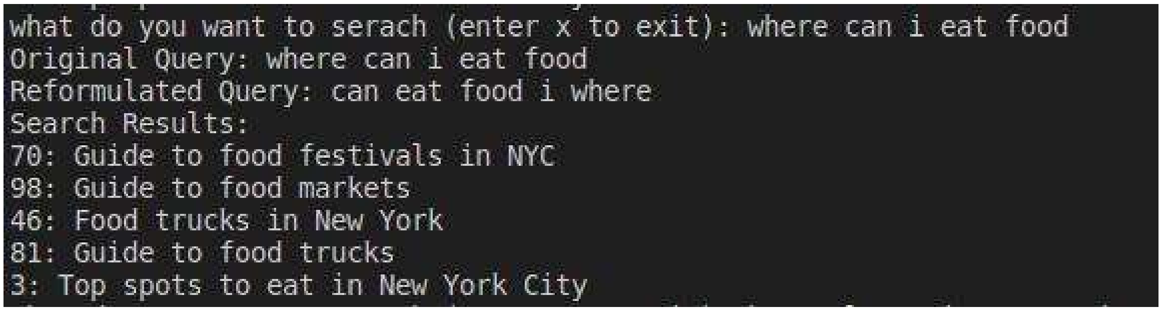
The system has successfully built the semantic graph and saved it as `semantic_graph.txt`. The original query "eat food" was entered, and the reformulated query remains "eat food", indicating no changes were made by the system for this specific query. The search results returned by the system are:



```
loading embeddings.
building graph.
graph built successfully.
Graph saved to semantic_graph.txt
what do you want to serach (enter x to exit): eat food
Original Query: eat food
Reformulated Query: eat food
Search Results:
70: Guide to food festivals in NYC
98: Guide to food markets
46: Food trucks in New York
81: Guide to food trucks
3: Top spots to eat in New York City
```

OUTPUT 2:

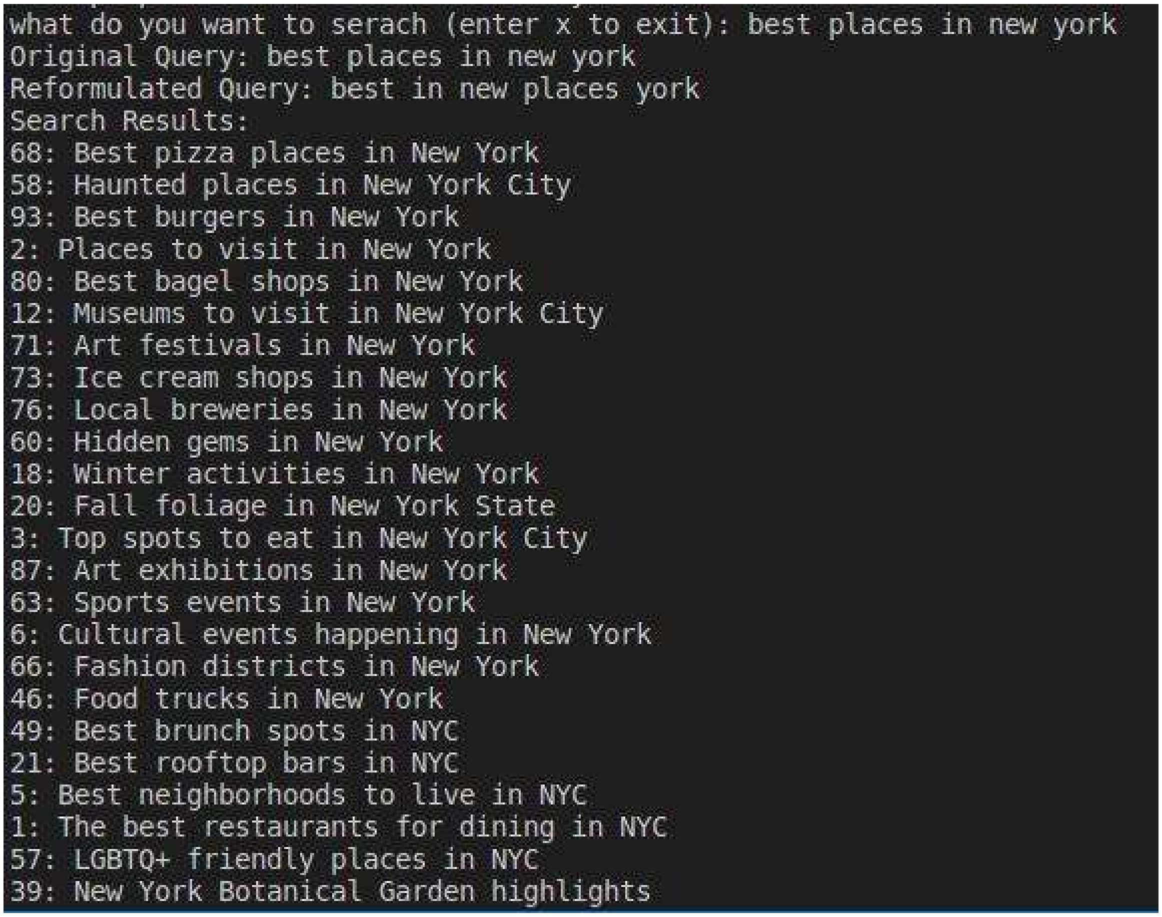
The output for the query "where can I eat food". The original query was "where can I eat food", and the reformulated query is "can eat food i where". This indicates a reordering of the words, likely due to the semantic analysis. The search results returned are:



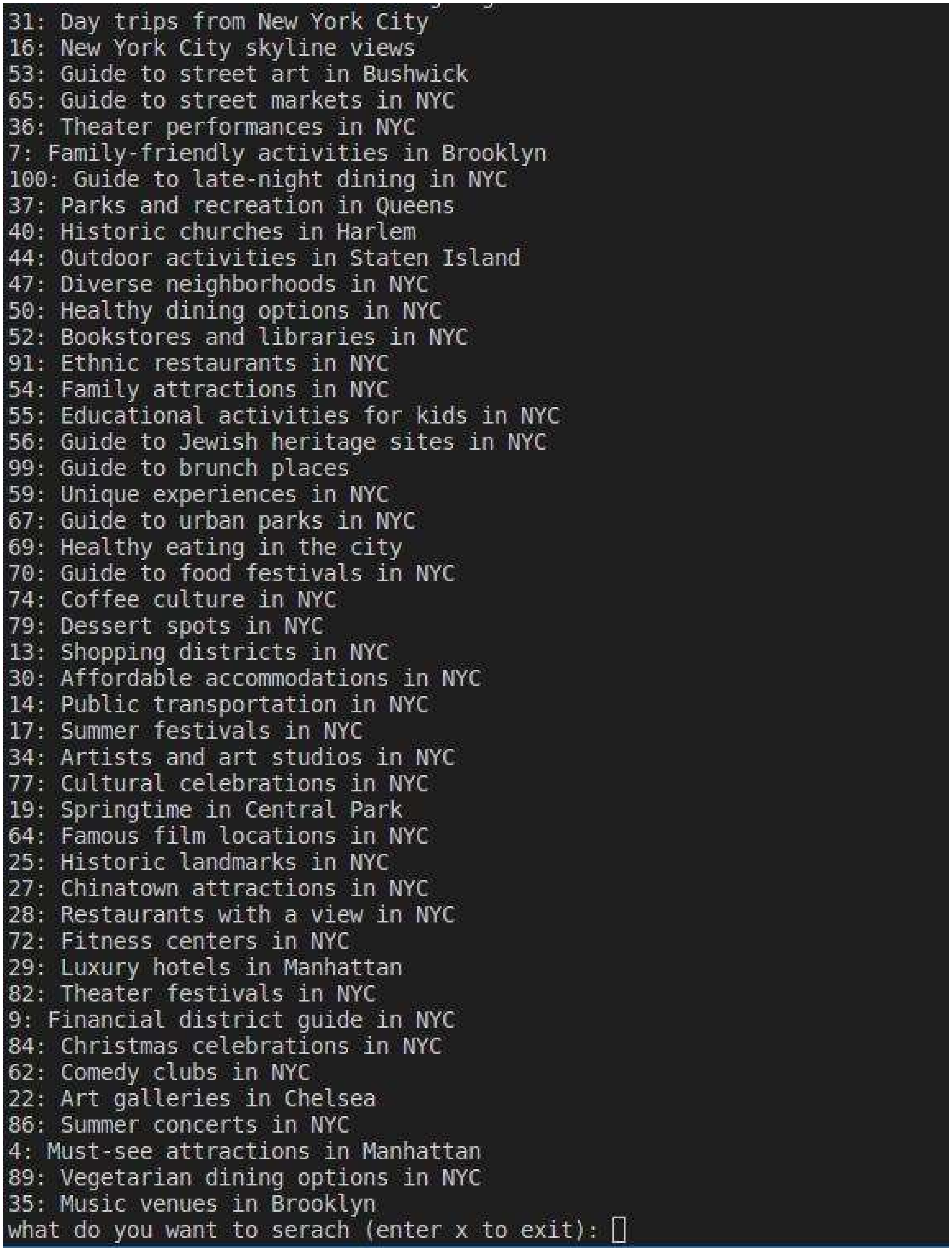
```
what do you want to serach (enter x to exit): where can i eat food
Original Query: where can i eat food
Reformulated Query: can eat food i where
Search Results:
70: Guide to food festivals in NYC
98: Guide to food markets
46: Food trucks in New York
81: Guide to food trucks
3: Top spots to eat in New York City
```

OUTPUT 3:

The output for the query "best places in new york". The original query was "best places in new york", and the reformulated query is "best in new places york", reflecting a reordering of the terms. The search results are extensive, with 100 entries listed. Here are the first few results:



```
what do you want to serach (enter x to exit): best places in new york
Original Query: best places in new york
Reformulated Query: best in new places york
Search Results:
68: Best pizza places in New York
58: Haunted places in New York City
93: Best burgers in New York
2: Places to visit in New York
80: Best bagel shops in New York
12: Museums to visit in New York City
71: Art festivals in New York
73: Ice cream shops in New York
76: Local breweries in New York
60: Hidden gems in New York
18: Winter activities in New York
20: Fall foliage in New York State
3: Top spots to eat in New York City
87: Art exhibitions in New York
63: Sports events in New York
6: Cultural events happening in New York
66: Fashion districts in New York
46: Food trucks in New York
49: Best brunch spots in NYC
21: Best rooftop bars in NYC
5: Best neighborhoods to live in NYC
1: The best restaurants for dining in NYC
57: LGBTQ+ friendly places in NYC
39: New York Botanical Garden highlights
```



```
31: Day trips from New York City
16: New York City skyline views
53: Guide to street art in Bushwick
65: Guide to street markets in NYC
36: Theater performances in NYC
7: Family-friendly activities in Brooklyn
100: Guide to late-night dining in NYC
37: Parks and recreation in Queens
40: Historic churches in Harlem
44: Outdoor activities in Staten Island
47: Diverse neighborhoods in NYC
50: Healthy dining options in NYC
52: Bookstores and libraries in NYC
91: Ethnic restaurants in NYC
54: Family attractions in NYC
55: Educational activities for kids in NYC
56: Guide to Jewish heritage sites in NYC
99: Guide to brunch places
59: Unique experiences in NYC
67: Guide to urban parks in NYC
69: Healthy eating in the city
70: Guide to food festivals in NYC
74: Coffee culture in NYC
79: Dessert spots in NYC
13: Shopping districts in NYC
30: Affordable accommodations in NYC
14: Public transportation in NYC
17: Summer festivals in NYC
34: Artists and art studios in NYC
77: Cultural celebrations in NYC
19: Springtime in Central Park
64: Famous film locations in NYC
25: Historic landmarks in NYC
27: Chinatown attractions in NYC
28: Restaurants with a view in NYC
72: Fitness centers in NYC
29: Luxury hotels in Manhattan
82: Theater festivals in NYC
9: Financial district guide in NYC
84: Christmas celebrations in NYC
62: Comedy clubs in NYC
22: Art galleries in Chelsea
86: Summer concerts in NYC
4: Must-see attractions in Manhattan
89: Vegetarian dining options in NYC
35: Music venues in Brooklyn
what do you want to serach (enter x to exit):
```