Search Results Linkage Analyzer Project (S.R.L.A)
DSAI 103 Data Acquisition
2023 Batch

Made By: Youssef Mohammad - 202300220

Under the Supervision of Prof. Mohammad Maher

Abstract:

What did we make?

The Search Results Linkage Analyzer (S.R.L.A) app retrieves results links with the results relationships and displays them in a dashboard. The app analyzes the results relationships when a user enters a query.

How Does the App works? :

The S.R.L.A. app focuses on identifying the most relevant search results for a user's query and presenting them through visual dashboards.

1. **Data Acquisition and Initial Processing:**

    - The app starts by taking two inputs from the user: their search query and a Serp API key.
    - It then leverages the API key to retrieve the organic search results for the given query.
    - Each retrieved result comes with a snippet, a concise summary of the corresponding webpage.

2. **Extracting Key Information and Assigning Relevance Scores:**

    - The app dives deeper into these snippets by extracting key statements from each one.
    - It employs Term Frequency (TF) analysis to assess the relevance of each statement to the other results snippets statements. Essentially, it calculates how often terms from the statement of each snippet of each result appear within the total statements of total snippets of total results.
    - To further refine relevance, the app considers the position of each result in the search rankings. Results appearing higher on the page are generally considered more relevant, so their TF-based relevance scores are adjusted accordingly.

3. **Identifying Highly Relevant Results:**

    - To establish a baseline for relevance, the app calculates the average relevance score of all retrieved results.
    - It then sets a threshold value that is half of this average.
    - Only results with a relevance score exceeding this threshold are deemed highly relevant to the user's query.
    - Titles of these highly relevant results are compiled into a specific data structure called a "node list."

4. **Network Construction and Centrality Analysis:**

    - Since the focus is on highly relevant results, the app establishes connections (edges) between all the nodes in the node list. This creates a network where each node (a highly relevant result title) is interconnected with every other node.
    - It's important to note that due to the interconnected nature of this network (all nodes are considered highly relevant), a specific network analysis metric called "betweenness centrality" will be zero for every node. Betweenness centrality typically measures the influence a node has in connecting other nodes within a network, but in this case, the complete interconnectedness renders this value irrelevant.
    - Another network analysis metric, "degree centrality," is also calculated, but it might not provide much insight due to the same interconnectedness. Degree centrality simply counts the number of connections (edges) a node has, and in this case, all nodes will have the same high degree value.

5. **Identifying Communities within Relevant Results:**

    - After calculating betweenness centrality, S.R.L.A. goes a step further to uncover potential thematic groupings within the highly relevant results. It achieves this by employing the Girvan-Newman algorithm.
    - In essence, the Girvan Newman algorithm works by iteratively removing the edges with the highest "betweenness centrality" from the network. Betweenness centrality, though zero in this case due to the network structure, conceptually refers to how often a connection (edge) lies on the shortest paths between other nodes.
    - By removing these high-betweenness edges, the algorithm progressively breaks down the network into smaller, more tightly knit clusters. These clusters represent groups of highly relevant results that likely share similar themes or topics.

6. **Visualization with Heatmap and 3D Network Graph:**

   - To visually represent the pre-calculated relevance scores of the highly relevant results, the app creates a heatmap. The X-axis of this heatmap corresponds to the position of each result within the list (remember, only highly relevant results are included), and the Y-axis represents the actual relevance score calculated earlier. By using colour intensity or shading, the heatmap effectively highlights results with higher relevance scores.
   - Finally, the app constructs a 3D network graph to provide a spatial representation of the connections between these highly relevant results. It utilizes a function within the NetworkX library (a Python package for network analysis) called "random layout" to assign random X, Y, and Z coordinates to each node (result title). These coordinates essentially determine the placement of each result within the 3D space, and by plotting them with connections (edges), the app visualizes the network of highly relevant search results.

Exporting for Visualization and Streamlit Dashboard:

- After generating the different visualizations, S.R.L.A. ensures they can be displayed effectively within dashboards. To achieve this:
    - The network graph, 3D network graph, Clustered Community Graph by (Girvan Newman) and heatmap graph are all exported as PNG images. These image files are a common format for displaying visuals on web pages.
    - The app also exports crucial data into JSON files. These files include:
        - Relevancy values for each result.
        - Positions of the results in the search rankings.
        - Degree centrality values.
        - Betweenness centrality values (which will likely be all zeros due to the network structure).
- Finally, all the generated files (PNG images and JSON data) are imported into a Streamlit dashboard. Streamlit is a Python framework for creating web apps. Within the dashboard:
    - The JSON data is converted into Pandas DataFrames, a popular data structure in Python for data analysis.
    - Streamlit then utilizes these DataFrames to display the data (relevancy values, positions, etc.) as bar charts or line charts, allowing for easy comparison of results.
    - The imported PNG images are also incorporated into the dashboard using Streamlit's image display functionalities.
    - To ensure the images render correctly within the dashboard, Streamlit employs a conversion from BGR to RGB colour format.

For Further experimenting You can download the app from the link below:

S.R.L.A EXE.rar

*Note that The file contains all the needed libraries for app functionality.

*Note that not all queries may get accurate results which may disturb app functionality.