Prototype 1

Github link: https://github.com/nihilistsumo/Mongoose

Runnable jar in jelly server:
/home/sk1105/sumanta/mongoose/mongoose-v[version number].jar

Sample project.properties file in jelly server:
/home/sk1105/sumanta/mongoose/project.properties

Passage run files in jelly server:
/home/sk1105/sumanta/mongoose/output/passage-runs/

1. **Describe each method that will be included:**

i) Hierarchical Agglomerative clustering based on similarity metrics - Based on similarity scores between pre-processed terms of each pair of paragraphs, we will run hierarchical agglomerative clustering on paragraphs on the set of retrieved paragraphs for each page queries. When the clustering result is good enough (based on Adjusted RAND, F1 and bcubed measure), we will map the clusters to the top-level sections and calculate trec-eval measures.

ii) Page rank on graph representation of TREC CAR where paragraphs are nodes and entities that are mentioned in two paragraphs are edges. As the graph walk algorithm gives us a ranking of nodes, we can evaluate the ranking quality of using the section/page-level entity qrels of benchmarkY1train.

iii) Heading weights - Indexing full text of each paragraph, BM25 query combinations as 1) page name, 2) lowest heading and 3) interior headings be used as queries. Combinations will be evaluated using MRR, MAP and Rprec.

iv) Clustering paragraphs based on their similarity - In this method, we use the paragraph text as a query and retrieve the relevant paragraphs. The current implementation uses the Lucene BM25 for ranking. The retrieved score acts as the similarity measure between the paragraphs, which would further be used in clustering.

2. Describe who will be responsible for which method:

i) Hierarchical Agglomerative clustering based on similarity metrics - Sumanta Kashyapi (sk1105)

ii) Page rank on CAR Hypertext graph-Shubham Chatterjee (sc1242)

iii) Heading weights with BM25 combination of query variants, Shortest path - Ajesh Vijayaragavan (avv1004)

iv) Clustering paragraphs based their similarity - Tarun Prasad Ganesa Pandian (tg1052)

3. Which results have been obtained so far?

i) Hierarchical Agglomerative clustering based on similarity metrics - These numbers are obtained with benchmarkY1-train dataset on top-level sections and using true page-paragraph map as input. We also have the retrieved page-paragraph version running in the server. For benchmarkY1-test dataset we have provided the runfile for top-level sections using the learned weigh vector from the training dataset.

Method : Mongoose	MAP	Rprec	recip_rank
Train top level with true article-paragraphs	0.0893	0.0920	0.2253

ii) Work in progress

iii)

Method : BM25	MAP	Rprec	recip_rank
Page name	0.0069	0.0046	0.0100
Lowest heading	0.0024	0.0016	0.0042

iv) Work in progress

4. Which risks or difficulties do you expect?

i) Calculating similarity metrics of all the term pair of each paragraph pair is computation intensive.

ii) Representing the CAR graph as a matrix. We might use an iterative approach for calculating the page rank but not sure about the time complexity. Representing the graph as a matrix would require huge space and we are not sure how to implement

matrix operations on the sparse matrix. We could represent the graph using a hashmap where each key is the node and the value is a list of nodes it links to.

iii) Priority queue operational error while implementing shortest path breadth first traversal.

iv) The process is expensive in terms of computations as we compare all paragraph pairs.