Assignment 3: Milestone 3 Report

Alexander Fino, Derek Trujillo, Andrew Rich

University of California, Irvine

IN4MATX 141 - Information Retrieval
Professor Lopes
December 8, 2024

10 Good Queries				
Query	**BEFORE**		**AFTER**	
	Relevant Results?	**Acceptable Performance?**	**Relevant Results?**	**Acceptable Performance?**
1. informatics	✓	✓	✓	✓
2. python	✓	✓	✓	✓
3. research opportunities	✓	✓	✓	✓
4. java programming	✓	✓	✓	✓
5. undergraduate admissions	✓	✓	✓	✓
6. esports gaming highlight	✓	✓	✓	✓
7. human computer interaction	✓	✓	✓	✓
8. master of software engineering	✓	✓	✓	✓
9. tech trends in 2019	✓	✓	✓	✓
10. 2017 ics industry showcase	✓	✓	✓	✓

10 Bad Queries				
Query	**BEFORE**		**AFTER**	
	Relevant Results?	**Acceptable Performance?**	**Relevant Results?**	**Acceptable Performance?**
1. machine learning models and machine learning techniques for machine learning applications	✓	✘	✓	✓
2. Sorting algorithms, search algorithms, graph algorithms, greedy algorithms	✓	✘	✓	✓
3. network security, network firewalls, and network encryption	✘	✘	✓	✓
4. the uci ics school of computer science for the department of informatics	✓	✘	✓	✓
5. how to fix a bug in my code	✘	✓	✓	✓
6. how to contact the school for more information	✘	✓	✓	✓
7. random access memory	✘	✓	✘	✓
8. central processing unit	✘	✓	✘	✓
9. uci ics cs	✘	✓	✘	✓
10. informatics labs and centers	✘	✓	✘	✓

Comments Regarding Bad Performing Queries

There were a few patterns that we recognized that the search engine performed poorly on.

The first are queries that contain the same tokens multiple times. This caused the search engine to repeatedly score the token over and over, causing unnecessary performance overhead. This was resolved by not allowing the search engine to score the same token more than once. This led to an increase in performance for queries 1 to 3.

The second are queries that are long, and contain multiple stop words. This caused the search engine to perform poorly because each stop word causes large performance overhead. This was a major issue primarily because their inverted lists are very long and can output irrelevant search results to float to the top, especially if the stop words are contained in important header tags (h1, h2, title, etc.) This problem was resolved by removing all stop words in queries that contain more than 3 words. From our testing, this led to an increase in performance and relevancy for queries 4 to 6.

The third are queries that rely on Positional Indexing or N-Grams. These were identified as producing irrelevant results and deemed unfixable because it is a limitation of our search engine. Positional Indexing and/or N-Grams were not implemented, so we could not increase the relevancy of queries 7 to 8.

Lastly are queries that are likely to rely on hubness and/or authority ranking. This is another limitation of our search engine that was not implemented and was deemed unfixable, similar to the positional indexing/ngrams. As such, we were unable to increase the relevancy of queries 9 to 10.

All changes made to resolve these issues preserved the performance and relevancy of the 10 good queries.

10 Bad Queries Summary		
Query	**Method Used**	**Summary**
1. machine learning models and machine learning techniques for machine learning applications	remove repeated tokens	increased performance
2. Sorting algorithms, search algorithms, graph algorithms, greedy algorithms	remove repeated tokens	increased performance
3. network security, network firewalls, and network encryption	remove repeated tokens	increased relevancy and performance
4. the uci ics school of computer science for the department of informatics	remove stop words if query > 3 words	increased performance
5. how to fix a bug in my code	remove stop words if query > 3 words	increased relevancy
6. how to contact the school for more information	remove stop words if query > 3 words	increased relevancy
7. random access memory	none (need positional indexing or ngrams)	limited by search engine capabilities
8. central processing unit	none (need positional indexing or ngrams)	limited by search engine capabilities
9. uci ics cs	none (need page ranking)	limited by search engine capabilities
10. informatics labs and centers	none (need page ranking)	limited by search engine capabilities