Testing the effectiveness of using a thesaurus on information retrieval

In this project, we set out to test the effectiveness of using a thesaurus to improve the search results of a simple search engine. To do so a testbed was created, the thesaurus function was implemented and a simple experiment was conducted to compared the performance of the system before and after the thesaurus. The main finding is that the thesaurus increases recall and decrease MAP, NDCG.

1. Implementation details

1.1 Testbed creation

The testbed was created using documents from various from the web [1]–[30]. Three different themes dominate the collection. Food, computer science fields of research and postgraduates scholarships. The three queries are related to these themes. Thus, the three chosen queries are:

1. **traditional Ivorian meals**
2. **postgraduate bursary**
3. **specialisations in computer science**

The length of the queries was kept relatively short on purpose since the simple search system uses an OR rule to get as accurate results as possible. Although the queries were saved in a text files, it was deemed more user friendly to enter the query as command line parameter.

1.2 Thesaurus Functionality

The implementation of thesaurus function was as simple as described below. It was implemented using the python library called thesaurus. This library was chosen mainly because of the simplicity of its interface.

Two approaches were considered. The implementation at indexing level, at query or both. At indexing, this will involve indexing the initial term and its synonyms. There are two apparent drawbacks to this approach. First, the index will easily become very large. Secondly, the synonyms returned by the thesaurus libraries are not very accurate and do not consider any contextual meaning for single word input. Although, the recall may increase, this might lead to a significant decrease in precision.

The query level implementation seems more natural.

Thus, given a query, the synonyms of each term in the query are retrieved and appended to the list of initial terms. The initial terms and their synonyms are used to query the indexed documents/files.

The code snippet below shows the additional code added to the given query.py file. The thesaurus library used sometimes returns input word as synonym. If this occurs, it is removed from the synonyms to avoid duplicates.

```
if parameters.thesaurus:
    synonyms = []
    for term in query_words:
        if term != "":
            word = Word(term)
            synonyms += word.synonyms()
        if term in synonyms:
```

```
            synonyms.remove(term)
    query_words += synonyms
```

This approach does not require any increase in storage required for the index. Besides, it limits the number of irrelevant documents that would have been retrieved if all the synonyms were indexed as discussed in the first approach. Given these reason, the thesaurus functionality was only implemented at query level. Similar to other functionalities, the activation of this functionality is controlled by a Boolean variable defined in the parameters.py file as thesaurus. A True value corresponds to ON and False to OFF.

1.3 Relevance metrics calculations

The relevance metrics calculations were done at the end of the query.py file. All the metrics were calculated at ten (10) because simple search only returns the first ten results.

1.3.1 Recall and Precision

The recall and precision are defined as:

$$Recall = \frac{Number\ of\ relevant\ documents\ returned}{Total\ number\ of\ relevant\ document}$$

$$Precision\ = \frac{Number\ of\ relevant\ documents\ returned}{Total\ number\ of\ documents\ returned}$$

The *total number of documents returned* is the minimum value between 10 and the number of documents matched.
Given a query – query.1, query.2, or query.3, getting the *number of relevant documents returned* and the *total number of relevant documents* requires reading from the corresponding relevant judgement file among relevance.1, relevance.2, or relevance.3.
To achieve this, three Boolean variables were defined and initialised to False in the parameter.py file namely query1, query2, and query3. Depending on the query, the corresponding Boolean variable should be set to True.
The relevance judgements of the documents are then loaded into a list in order. To access a particular document's index, we just need to index the list with the document's identifier minus one.
Thus, *number of relevant documents returned* is obtained by counting the number of documents returned whose relevance judgment is greater than 0. Similarly, the *total number of relevant documents* is obtained by counting the number of 1s and 2s in the list of relevant documents.

1.3.2 Mean Average Precision: MAP

To compute the MAP, different precisions were calculated, as described in the above subsection, at different values: from 1 through N. Where N is the minimum between 10 and the total number of documents matched. For a given query, the average of these precisions gives the average precision (AP). The MAP can be obtained for all the three queries by finding the mean value of the three APs. This last step was done manually.

1.3.3 Normalized Discounted Cumulative Gain: NDCG

NDCG was calculated using the formula below.

$$NDCG = \frac{DCG}{IDCG}$$

DCG was computed as follows. For each returned document its relevant judgment value was divided by $\log_2(i + 1)$, where, i is the rank of the document and added to an accumulator.
The ideal DCG, IDCG was calculated by re-sorting the returned documents according to their relevant judgment values first and recalculating the DCG.
NDCG was obtained by dividing DCG by IDCG.

In order to test the system, all the functionalities except the thesaurus were set to True together with the correct query Boolean – query1, query2, query3. The metrics values were then printed out and recorded as before values. This was done for each query. The after values were obtained by repeating the same experiment with the thesaurus Boolean set to True. The results are presented in the next section.

2. Results and Analysis

Table 1: Simple Search Performance Before and After Thesaurus

	Recall		**Precision**		**AP**		**NDCG**	
	Before	***After***	***Before***	***After***	***Before***	***After***	***Before***	***After***
Query 1	0.80	1.00	0.50	0.50	0.67	0.59	0.92	0.89
Query 2	0.375	1.00	1.00	0.89	1.00	0.99	1.00	1.00
Query 3	0.75	0.88	0.60	0.70	0.89	0.75	0.90	0.88
Mean	0.64	0.96	0.70	0.70	MAP=0.85	MAP=0.78	0.94	0.92

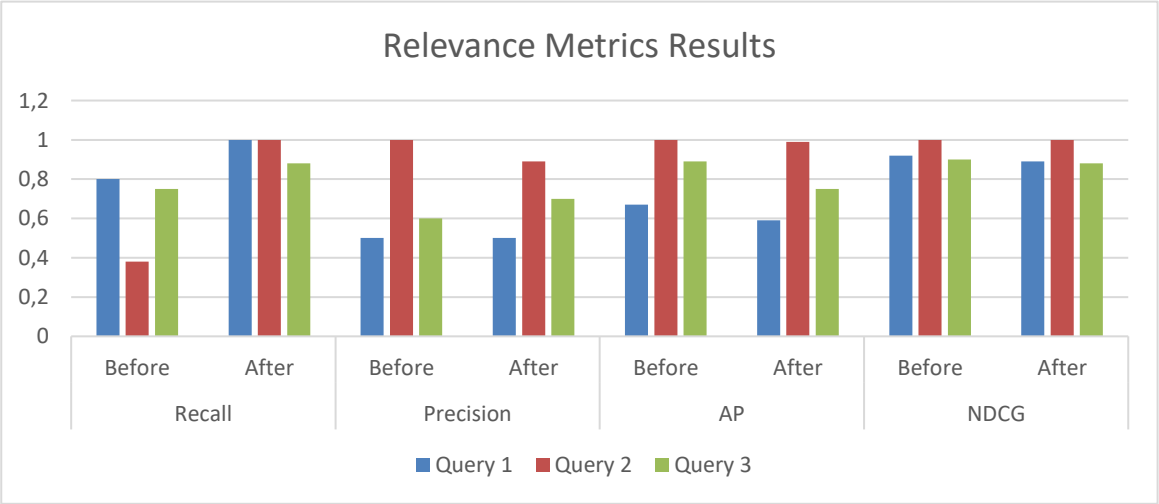

Figure 1: Bar chart of Simple Search Performance Before and After Thesaurus

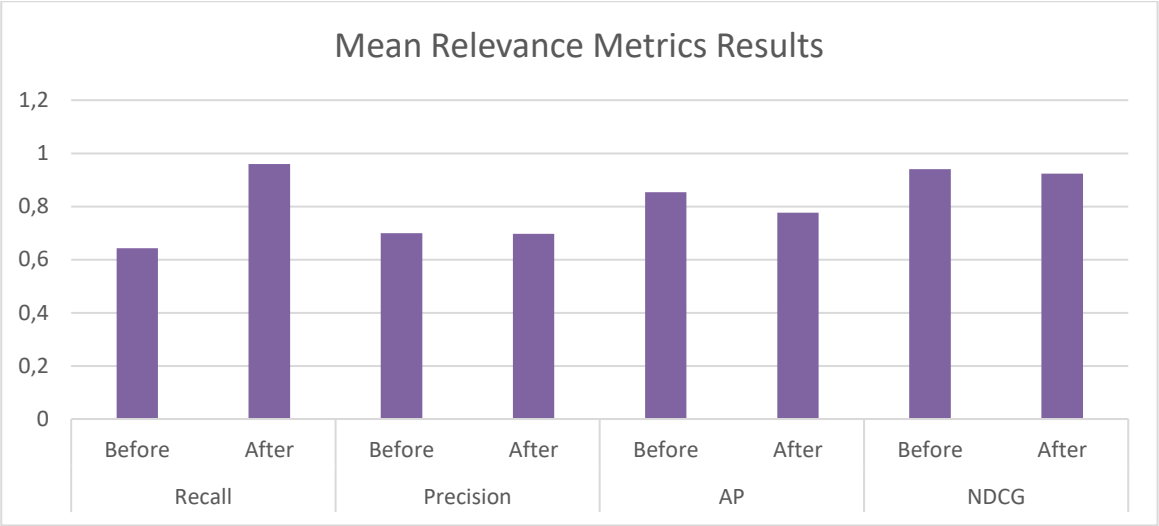

Figure 2: Bar chart of Simple Search Average Performance Before and After Thesaurus

On average, the thesaurus increases the recall from 64% to 96%, decreases the MAP (85% to 78%) and NDCG (94% to 92%) but precision remains fairly the same for the chosen queries. This is as expected except the precision which remained constant. This apparent anomaly can be explained by the relatively small number of relevant documents in the testbed for the given queries.

In conclusion, the effectiveness of the thesaurus depends on the particular application. That is whether recall is more important than other metrics. Given that, NDCG is more robust metric, one could argue that in this experiment, it did not improve the results.

3. References

[1] Wikipedia, "Information retrieval." [Online]. Available: https://en.wikipedia.org/wiki/Information_retrieval. [Accessed: 16-Aug-2018].

[2] Wikipedia, "Machine learning." [Online]. Available: https://en.wikipedia.org/wiki/Machine_learning. [Accessed: 16-Aug-2018].

[3] Wikipedia, "Computer vision." [Online]. Available: https://en.wikipedia.org/wiki/Computer_vision. [Accessed: 16-Aug-2018].

[4] Wikipedia, "Feature extraction." [Online]. Available: https://en.wikipedia.org/wiki/Feature_extraction. [Accessed: 16-Aug-2018].

[5] Wikipedia, "Outline of electrical engineering." [Online]. Available: https://en.wikipedia.org/wiki/Outline_of_electrical_engineering. [Accessed: 16-Aug-2018].

[6] Wikipedia, "West African cuisine." [Online]. Available: https://en.wikipedia.org/wiki/West_African_cuisine. [Accessed: 16-Aug-2018].

[7] Wikipedia, "Confederation of African Football." [Online]. Available: https://en.wikipedia.org/wiki/Confederation_of_African_Football. [Accessed: 16-Aug-2018].

[8] Wikipedia, "West Africa." [Online]. Available: https://en.wikipedia.org/wiki/West_Africa. [Accessed: 16-Aug-2018].

[9] Wikipedia, "Ivorian cuisine." [Online]. Available: https://en.wikipedia.org/wiki/Ivorian_cuisine. [Accessed: 16-Jul-2018].

[10] B. Quotes, "Family Quotes." [Online]. Available: https://www.brainyquote.com/topics/family. [Accessed: 16-Aug-2018].

[11] U. of Pretoria, "Postgraduate scholarships." [Online]. Available: https://www.up.ac.za/fees-and-funding/article/277458/postgraduate-scholarships. [Accessed: 16-Aug-2018].

[12] S. Portal, "321 Scholarships in South Africa." [Online]. Available: https://www.scholarshipportal.com/scholarships/south-africa. [Accessed: 16-Aug-2018].

[13] M. F. S. P.- UCT, "The Mastercard Foundation Scholars Program at UCT." [Online]. Available: http://www.mcfsp.uct.ac.za/. [Accessed: 16-Aug-2018].

[14] UCT, "Entrance scholarships." [Online]. Available: http://www.dsa.uct.ac.za/student-funding-administration/scholarships/entrance. [Accessed: 16-Aug-2018].

[15] T. M. R. Foundation, "Purpose and Vision." [Online]. Available: https://mandelarhodes.org/the-scholarship/purpose-

and-vision/. [Accessed: 16-Aug-2018].
[16] Scholars4dev, "Undergraduate." [Online]. Available: http://www.scholars4dev.com/category/level-of-study/undergraduate-scholarships/. [Accessed: 16-Aug-2018].
[17] E. Out, "21 iconic South African foods – the ultimate guide for visitors." [Online]. Available: http://www.eatout.co.za/article/21-iconic-south-african-foods-ultimate-guide-visitors/. [Accessed: 16-Aug-2018].
[18] A. GAYE, "Ivorian food: the GARBA!" [Online]. Available: http://awalemag.com/ivorian-food-the-garba/. [Accessed: 16-Aug-2018].
[19] F. Rarh, D. Pojee, and S. Zulphekari, "Restaurant Table reservation using time-series prediction," no. Icces, pp. 153–155, 2017.
[20] Wikipedia, "South Africa." [Online]. Available: https://en.wikipedia.org/wiki/South_Africa. [Accessed: 16-Aug-2018].
[21] W. Facts, "The Largest Football (Soccer) Stadiums In The World." [Online]. Available: https://www.worldatlas.com/articles/the-largest-football-soccer-stadiums-in-the-world.html. [Accessed: 16-Aug-2018].
[22] B. Insider, "The 10 most critical problems in the world, according to millennials." [Online]. Available: https://www.businessinsider.com/world-economic-forum-world-biggest-problems-concerning-millennials-2016-8?IR=T. [Accessed: 16-Aug-2018].
[23] The top tens, "Man's Greatest Achievements." [Online]. Available: https://www.thetoptens.com/man-achievements/. [Accessed: 16-Aug-2018].
[24] Forbes, "13 Of 2015's Hottest Topics In Computer Science Research." [Online]. Available: https://www.forbes.com/sites/quora/2015/04/22/13-of-2015s-hottest-topics-in-computer-science-research/#1ccec6621e88. [Accessed: 16-Aug-2018].
[25] E. Z. University, "Areas of Research in Computer Science." [Online]. Available: https://www.inf.ethz.ch/research.html. [Accessed: 16-Aug-2018].
[26] National Research Foundation, "NRF Opportunities." [Online]. Available: http://www.nrf.ac.za/bursaries/opportunities. [Accessed: 16-Aug-2018].
[27] E. B. University, "Database Management Systems (DBMS)." [Online]. Available: https://www2.eecs.berkeley.edu/Research/Areas/DBMS/. [Accessed: 16-Aug-2018].
[28] BERND, "what is a braai." [Online]. Available: https://www2.eecs.berkeley.edu/Research/Areas/DBMS/. [Accessed: 16-Aug-2018].
[29] Wikipedia, "Deep learning." [Online]. Available: https://en.wikipedia.org/wiki/Deep_learning. [Accessed: 18-Aug-2018].
[30] UCT, "Noticeboard UCT Students." [Online]. Available: http://www.students.uct.ac.za/students/fees-funding/postgraduate-degree-funding/noticeboard. [Accessed: 16-Aug-2018].