An Analysis of Indexing and Querying Strategies on Technologically Assisted Review Tasks

Alexandros Ioannidis
University of Strathclyde
Glasgow, UK

Author 2
institution
city, state

Author 3
institution
city, country

ABSTRACT

This paper is a preliminary experimentation study using the CLEF collection for evaluating the effectiveness of different indexing methodologies of documents and query parsing techniques. Furthermore, it is an attempt to advance and share the efforts of observing the characteristics and helpfulness of various methodologies for indexing PubMed documents and for different topic parsing techniques to produce queries. For this purpose, our research included experimentation with different document indexing methodologies, by utilising existing tools, such as the Lucene4IR[1] (L4IR) information retrieval (IR) system, the Technology Assisted Reviews (TAR) for Empirical Medicine tool[2] for parsing topics of the CLEF collection and the TREC evaluation tool[3] to appraise system's performance.

CCS CONCEPTS

• **Information systems → Evaluation of retrieval results**;

KEYWORDS

PubMed, Medical Information Retrieval (IR), Indexing Schemata, Measurement, Performance, Query Parser, CLEF Collection, BM25

ACM Reference Format:
Alexandros Ioannidis, Author 2, and Author 3. 1997. An Analysis of Indexing and Querying Strategies on Technologically Assisted Review Tasks. In *Proceedings of ACM Woodstock conference (WOODSTOCK'97),* Jennifer B. Sartor, Theo D'Hondt, and Wolfgang De Meuter (Eds.). ACM, New York, NY, USA, Article 4, 5 pages. https://doi.org/10.475/123_4

1 INTRODUCTION

The recently entrenched CLEF 2017 eHealth Task 2 required from the participants to rank a set of PubMed abstracts (A) given the results retrieved from the previous Task 1. The track had 2 objectives. Create an effective ranking of the documents, such that all of the relevant abstracts are recovered as soon as possible and determine a subgroup of (A), which contains as many of the relevant abstracts for the least effort (Kanoulas et al., 2017) [4]. The participants were given the associated qrels judgments file, which follows the TREC format Topic number, Iteration, PubMed identification number, Relevancy, a binary code of 0 (not relevant) and 1 (relevant). The order of documents in the qrels file does not suggest degree of relevance. Documents not occurring in the qrels file were not judged by the human assessor and are presumed to be irrelevant in the evaluations. A more detailed description of the task can be found here[4]. It is important to note some of the different approaches in the submissions of the participants. For example Alharbi et al. [1] utilised the review title and Boolean query to order the abstracts retrieved by the query, by applying standard similarity measures. According to the authors [1] the title and terms extracted from the Boolean query contributed the most useful information for this task. Their methodology made use of 3 topic parts, the text of the title, the terms and Mesh items parsed from the Boolean query. The Mesh items were preprocessed with the same approach applied on the Boolean query. Moreover, preprocessing was added to the PubMed abstracts and data extracted from the topics. The text was tokenised, changed to lower case, stop words were taken out and the rest of the tokens were stemmed. Lastly, the data extracted from the topic and every abstract was converted into TF-IDF weighted vectors [1] to calculate similarity among the topic and every abstract using the cosine metric for the pair of vectors. Ecnu et al. [3] used a customised Learning-to-Rank (L2R) model and the word2vec to represent queries and documents and compute their similarities by cosine distance. Their L2R model consists of 3 points, query expansion, feature extraction and model training. In the query expansion stage, they improved retrieval precision by expanding queries. In the feature extraction stage, they extracted features of each query document pair. When a document was retrieved under a query, it was connected with a weighting score and rank. Finally, in the learning phase of the L2R model, the relevance of a query-document pair was assessed with the random forest classifier. Norman et al. used a system that builds on logistic regression [5], and implemented strategies to handle class imbalance and perform relevance feedback. Additionally,they tested 2 classifiers, logistic regression with stochastic gradient descent learning on the entire data and standard logistic regression trained using conventional methods on a subgroup of the data with added preprocessing to enhance the yield. Nunzio et al. [6] concentrated their work on discovering the optimal combination of certain hyper-parameters by utilising the training data available and a force brute strategy. This strategy created different query features of the identical information need given a minimum amount of relevance feedback. Furthermore, they decided to use a plain BM25 retrieval model and acquire the relevance feedback for the first abstract in the ranking list for each topic. Afterwards, they asked 2 different people to build 2 different queries according to the value of the feedback. The 2 queries

[1]https://github.com/lucene4ir
[2]https://github.com/CLEF-TAR/tar
[3]http://trec.nist.gov/trec_eval

[4]://sites.google.com/site/clefehealth2017/task-2

WOODSTOCK'97, July 1997, El Paso, Texas USA

ACM ISBN 123-4567-24-567/08/06…$15.00
https://doi.org/10.475/123_4

are combined with the original one in different ways. Finally, they designed alternative strategies that use the following parameters, number of documents to assess in batches or iteratively, percent of documents to assess, maximum number of documents to assess per iteration, number of terms to add at each feedback iteration for the cost-effective evaluation and the minimum precision the system can reach before terminating the search. Singh et al. utilised Lucene's inverted index to index the retrieved articles retrieved from PubMed query performed during a systematic review [7]. The query is processed for term boosting, fuzzy search and used for scoring documents according to TF-IDF similarity. Relevance feedback is used to update the query and become more pragmatic. Essentially, different people have implemented different strategies and they all used different methods, so it is hard to determine what indexing strategy or querying parsing technique is best to perform for the initial round of retrieval.

2 RESEARCH QUESTIONS

Therefore, the following research questions, are raised:

RQ1 How do different indexing schemata affect performance?
RQ2 How does the initial query extracted from the CLEF Topic affects performance?

3 METHOD

3.1 Data and Materials

The data that was used is the CLEF collection[5]. The complete CLEF collection consists of 198,365 XML documents, it contains 50 topics, a total of 266,967 abstracts and the number of relevant documents is . Furthermore, each PubMed article set has a basic structure, that contains a PubMed article. The PubMed article consists of the Medline citation fields and the PubMed data and metadata fields. The Medline citation includes fields such as the ID of the PubMed document, date of creation and completion, ISO abbreviations, title and abstract of the article, the list of author(s), the MeSH heading list, the list of chemicals other information of the Medline Journal. The PubMed data contain information such as historical dates, the publication status and more information. Each topic of the CLEF collection contains the topic-ID, the title of the review and the Boolean query manually drafted and constructed by Cochrane experts and the collection of PubMed document identifiers (PMID) retrieved by executing the query in MEDLINE databases. For all of our indexes we used the standard tokeniser in L4IR, and a variety of token filters. The token filters used are the following: stopping, porter stemmer, generation of word parts, pattern replacement, lowercase and word delimiter with the key parameter of this token filter set to 'preserveOriginal'. Additionally, we utilised the L4IR (Azzopardi et al., 2017) [2], which consists a collaborative effort of researchers to extend the Lucene library and produce a group of statistical evaluation tools for common IR methods such as indexing and retrieval for different types of test collections. The L4IR, includes among other functions, 3 main applications: the IndexerApp, the RetrievalApp and the ExampleStatsApp. The IndexerApp, allows to index multiple different TREC collections, such as the Aquaint Collection, but also the PubMed collection. The RetrievalApp, is a batch retrieval application, which contains various retrieval algorithms, for example the BM25, the PL2 and more. Both the IndexerApp and the RetrievalApp applications can be configured appropriately by editing their parameters files 'index_params.xml' and 'retrieval_params.xml'. Finally, the ExampleStatsApp, is an application, which shows an example of accessing different metrics, such as term posting list, term positions, and more regarding the terms, the documents and the corpus. For the purposes of this study, we used the baseline retrieval algorithm used at CLEF 2017, BM25, to rank the document sets. Moreover, we utilised the CLEF TAR tool, for extracting parts from the topics of the CLEF collection, such as the ID, title and query of the topic and more. Additionally, we used Jupyter-Notebooks to automate the process of running experiments, but also to facilitate more testing in the future, in a larger experimentation setup (e.g. the incorporation of a greater number of indexing methodologies and query parsing techniques).

3.1.1 Measures. The measurements we used for the purpose of this research are Mean Average Precision (MAP) (Table 1), Precision after 10, 20 and 30 documents (Table 2) and Interpolated Recall - Precision Averages (IRPA) at 0.10, 0.20 and 0.30 recall (Table 3).

3.1.2 Indexing Schemata. In our research we used 5 different indexing schemata, which are outlined below. Each indexing schema consists of a unique combination of XML fields.

1 The 1^{st} indexing schema indexed the Title, Abstract and PMID and was named 'TAP'.
2 The 2^{nd} indexing schema included the fields of the 1^{st} indexing schema along with the Author(s), the Journal title and the Year and was named '1+AJY'.
3 The 3^{rd} schema combined the XML fields of the 2^{nd} indexing schema with the Mesh Heading List (MHL) and was named '2+MHL'.
4 The 4^{th} indexing schema included the 2^{nd} indexing schema and the MedlineTA (MTA) field and was named '2+MTA'.
5 The 5^{th} indexing schema combined the 2^{nd} indexing schema with the Mesh HL and the MedlineTA and was named '2+MHLMTA'.

3.2 Query Parser

Furthermore, 3 different query parsers were implemented, so that the performance of the IR system L4IR could be tested by using different query variations. The 3 different query parsers that were used are listed below:

A The Title of the topic and was named 'title'.
B The Query of the topic and was named 'query'.
C The combination of the Title and Query of the topic and was named 'title and query'.

4 TOKEN FILTER PIPELINE

4.1 Retrieval Model and Parameters

As mentioned above the IR model used was the BM25 and its free parameters b and k_1 were set to be 0.75 and 1.2 respectively, for this experimentation round. To compare our results, we used the SPSS statistical tool and the one-way ANOVA and one-way repeated measures ANOVA statistical test procedures. We denoted statistical significance at 5%. The results from both statistical tests

[5]http://www.clef-initiative.eu/dataset/test-collection

indicated significance at $p < 0.05$ and the $p-value < 0.00001$. More specifically, the one-way ANOVA and one-way repeated measures ANOVA procedures returned a total standard deviation of 0.0362, the F-ratio values were estimated at 233.5 and 1373.005 respectively and the total mean values were 0.0888 and 0.089 accordingly.

5 RESULTS

5.1 Tables

Table 1: Comparison of MAP between the 5 different indexing schemes and the 3 different query parsers

	TAP	1+AJY	2+MHL	2+MTA	2+MHLMTA
title	0.1211	0.1237	0.135	0.1233	0.136
query	0.039	0.0419	0.048	0.0419	0.0485
title and query	0.0836	0.0954	0.0993	0.0942	0.1013

Table 2: Comparison of precision after certain # documents are retrieved between the 5 different indexing schemes and the 3 different query parsers

	TAP	1+AJY	2+MHL	2+MTA	2+MHLMTA
					Precision after 10 documents retrieved
title	0.258	0.29	0.274	0.29	0.284
query	0.156	0.13	0.138	0.128	0.144
title and query	0.218	0.198	0.22	0.204	0.216
					Precision after 20 documents retrieved
title	0.251	0.271	0.269	0.271	0.266
query	0.128	0.126	0.145	0.132	0.141
title and query	0.189	0.197	0.2	0.197	0.2
					Precision after 30 documents retrieved
title	0.2467	0.2573	0.262	0.2553	0.2593
query	0.112	0.1187	0.132	0.1247	0.1333
title and query	0.178	0.188	0.1927	0.19	0.1927

Table 3: Comparison of IRPA at # recall between the 5 different indexing schemes and the 3 different query parsers

	TAP	1+AJY	2+MHL	2+MTA	2+MHLMTA
					IRPA at 0.10 recall
title	0.2842	0.292	0.3206	0.2866	0.322
query	0.1187	0.1258	0.14	0.1297	0.1415
title and query	0.212	0.2307	0.2445	0.2436	0.2543
					IRPA at 0.20 recall
title	0.2203	0.2125	0.236	0.2101	0.2411
query	0.0643	0.073	0.09	0.0723	0.0862
title and query	0.132	0.1571	0.1631	0.1572	0.1683
					IRPA at 0.30 recall
title	0.1542	0.159	0.172	0.1576	0.1776
query	0.0373	0.0434	0.0584	0.042	0.0621
title and query	0.1083	0.1228	0.1284	0.1167	0.1313

5.2 Figures and Observations

The figures below display the information captured from our experiments in the tables above. The vertical axis exhibits the 3 different query parsing techniques. Because the fifth indexing methodology incorporates the most fields of the XML documents compared to the other 4 indexing methodologies, it was portrayed with a distinctive dashed line in order to draw attention into its performance against the rest. This allows us to draw easier conclusions on whether adding more XML fields into our indexer increases the performance of our IR system.

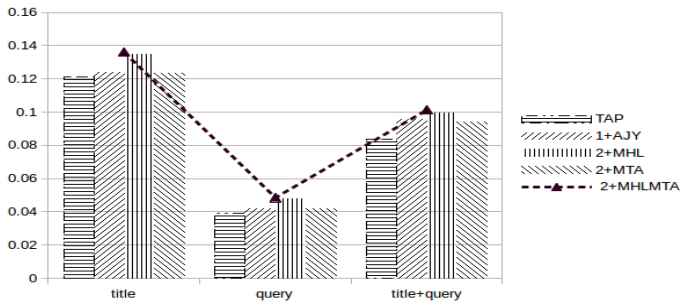

Figure 1: Graphical comparison of the MAP measurement for the 5 different indexing schemes and the 3 different query parsing techniques

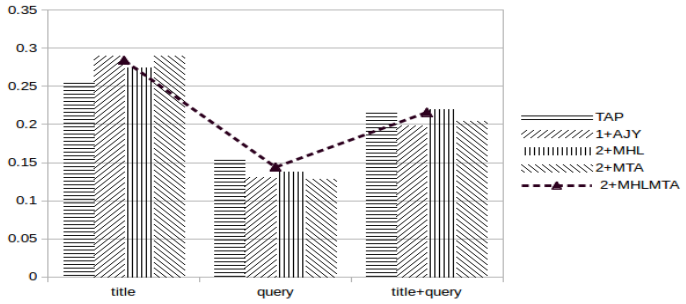

Figure 2: Graphical comparison of the precision after 10 documents retrieved for the 5 different indexing schemes and the 3 different query parsing techniques

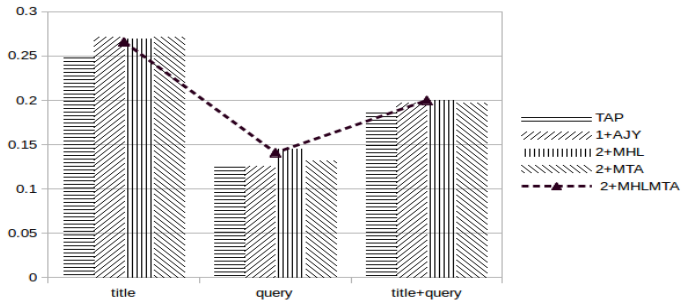

Figure 3: Graphical comparison of the precision after 20 documents retrieved measurement for the 5 different indexing schemes and the 3 different query parsing techniques

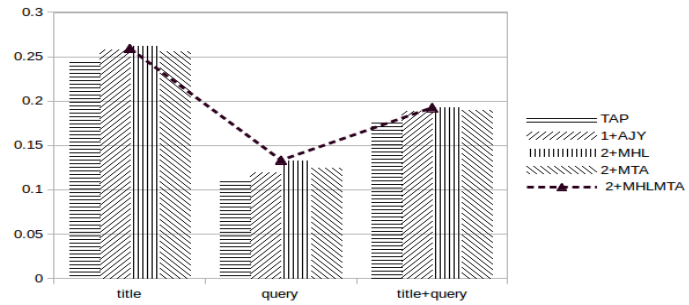

Figure 4: Graphical comparison of the precision after 30 documents retrieved measurement for the 5 different indexing schemes and the 3 different query parsing techniques

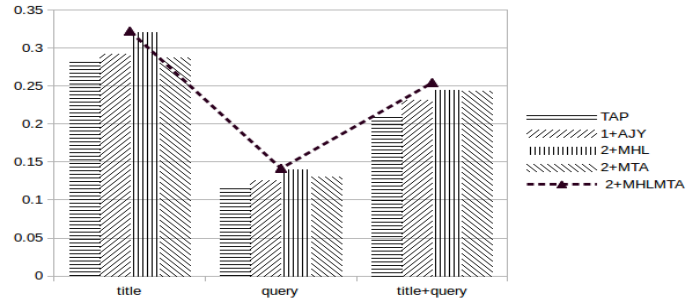

Figure 5: Graphical comparison of the IRPA at 0.10 recall, for the 5 different indexing schemes and the 3 different query parsing techniques

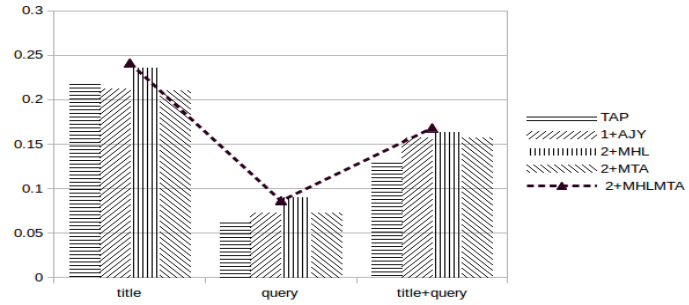

Figure 6: Graphical comparison of the IRPA at 0.20 recall, for the 5 different indexing schemes and the 3 different query parsing techniques

We notice from the MAP measurements that the 5^{th} indexing scheme performs the best in all 3 query parsing scenarios. Respectively, we observe that the 'title' topic-parsing technique A accomplishes the best MAP performance with all 5 indexing scheme scenarios. Furthermore, we notice from the Precision after 10, 20 and 30 documents retrieved, that there is no dominant indexing methodology. Moreover, we notice that the performance of the 2nd and 4th indexing methodology is very similar in almost every query parsing and number of retrieved documents scenario. We also notice that the precision after 10, 20 and 30 documents retrieved is higher when the 'title' parsing technique A is utilised instead of the other 2 initial query extraction techniques. Additionally, we recognise that the measurements of the IRPA at 0.10 and 0.30 recall is always higher when using the 5^{th} indexing scheme. The same can also be said for the IRPA at 0.20 recall, but in this case there is a small exception when utilising the 'query' topic-parsing technique, where we observed that the 5^{th} indexing scheme accomplishes the best score after the 3^{rd} indexing method. Finally, we observed that the IRPA at 0.10, 0.20 and 0.30 recall is always higher when using the 'title' topic-parsing technique.

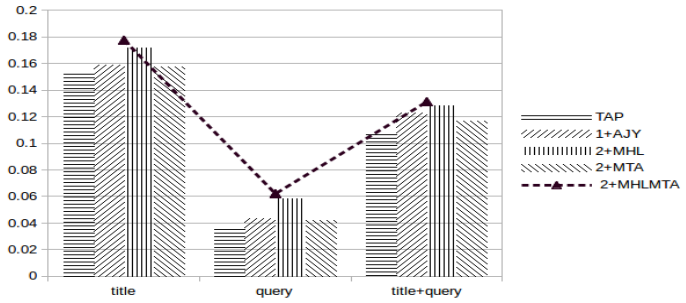

Figure 7: Graphical comparison of the IRPA at 0.30 recall, for the 5 different indexing schemes and the 3 different query parsing techniques

6 CONCLUSION

We focused on discovering how alternative indexing schemata affect performance and how different extraction techniques of the initial query from CLEF topics affect performance. We found out that the IR performance is improved substantially when using the 5^{th} indexing scheme, which enables us to draw the conclusion that adding more fields to the PubMed indexer is actually a positive contributor for the retrieval effectiveness of an IR system such as L4IR. We look forward to increase the number of our experiments using the L4IR system, to index additional fields of the PubMed documents and to find methodologies for indexing and query parsing new empirical test collections. Finally we also plan to test more retrieval models that are available in the L4IR system and make use of Active Learning to enhance the ranking by using feedback from reviewers.

A APPENDIX

We have included a README.md file which contains guide instructions, 2 iPython Notebooks with the code for the automation of our experiments, a Python script for parsing the topics to produce queries, 5 different Java classes one for each indexing methodology and the updated 'IndexerApp' Java class of L4IR.

REFERENCES

[1] Amal Alharbi and Mark Stevenson. [n. d.]. Ranking Abstracts to Identify Relevant Evidence for Systematic Reviews: The University of SheffieldâĂŹs Approach to CLEF eHealth 2017 Task 2. ([n. d.]).
[2] Leif Azzopardi, Yashar Moshfeghi, Martin Halvey, Rami S Alkhawaldeh, Krisztian Balog, Emanuele Di Buccio, Diego Ceccarelli, Juan M Fernández-Luna, Charlie Hull, Jake Mannix, et al. 2017. Lucene4IR: developing information retrieval evaluation resources using Lucene. In *ACM SIGIR Forum*, Vol. 50. ACM, 58–75.
[3] Jiayi Chen, Su Chen, Yang Song, Hongyu Liu, Yueyao Wang, Qinmin Hu, Liang He, and Yan Yang. 2017. Ecnu at 2017 ehealth task 2: Technologically assisted reviews in empirical medicine. *Working Notes of CLEF* (2017), 11–14.
[4] Evangelos Kanoulas, Dan Li, Leif Azzopardi, and Rene Spijker. 2017. CLEF 2017 technologically assisted reviews in empirical medicine overview. In *CEUR Workshop Proceedings*, Vol. 1866. 1–29.
[5] Christopher Norman12, Mariska Leeflang, and Aurélie Névéol. 2017. Limsi@ clef ehealth 2017 task 2: Logistic regression for automatic article ranking. (2017).

[6] GMD Nunzio, Federica Beghini, Federica Vezzani, and Genevieve Henrot. 2017. An interactive twodimensional approach to query aspects rewriting in systematic reviews. ims unipd at clef ehealth task 2. *Working Notes of CLEF* (2017), 11–14.
[7] Jaspreet Singh and Lini Thomas. [n. d.]. IIIT-H at CLEF eHealth 2017 Task 2: Technologically Assisted Reviews in Empirical Medicine. ([n. d.]).