Integrating Medical Ontology and Pseudo Relevance Feedback for Medical Document Retrieval

Andia Ghoddousi

A Thesis Submitted To the Faculty of Graduate Studies

In Partial Fulfillment of the Requirements for the Degree Of

Master of Science

Graduate Program in Computer Science and Engineering

York University

Toronto, Ontario

April 2016

Abstract:

The purpose of this thesis is to undertake and improve the accuracy of locating the relevant documents from a large amount of Electronic Medical Data (EMD). The unique goal of this research is to propose a new idea for using medical ontology to find an easy and more reliable approach for patients to have a better understanding of their diseases and also help doctors to find and further improve the possible methods of diagnosis and treatments. The empirical studies were based on the dataset provided by CLEF focused on health care data. In this research, I have used Information Retrieval to find and obtain relevant information within the large amount of data sets provided by CLEF. I then used ranking functionality on the Terrier platform to calculate and evaluate the matching documents in the collection of data sets. BM25 was used as the base normalization method to retrieve the results and Pseudo Relevance Feedback weighting model to retrieve the information regarding patients' health history and medical records in order to find more accurate results. I then used Unified Medical Language System to develop indexing of the queries while searching on the Internet and looking for health related documents. UMLS software was actually used to link the computer system with the health and biomedical terms and vocabularies into classify tools; it works as a dictionary for the patients by translating the medical terms. Later I would like to work on using medical ontology to create a relationship between the documents regarding the medical data and my retrieved results.

Acknowledgements:

I want to thank all the effort and feedback of the program committee. I also thank all the participants in this study. This research was supported by the Natural Sciences and Engineering Research Council of Canada (NSERC).

Table of Content:

List of Figures:

1. *Introduction:*

In the recent years, health care institutes establish user-friendly environments using new technologies for patients to be more involved and to have a better understanding about their health information. They can also have access to their Electronic Medical Records, which will file the information instantly and securely. EMR are the structured collection of health information about the patients that are stored electronically in a digital format. These records can be shared with the different health institutes across the city or country which can also be available for the doctors. EMR are the logical collection of electronic records of patients' health information such as clinical reports, scans and images, summaries of their personal health and medical issues and the required information for diagnosis and treatments [3] [6]. This demographic system is designed to carefully capture the state of the patient's health history and improve the treatment conditions. Electronic Health Records on the other hand are a range of data consisting of medical history, laboratory tests, personal health, and billing information. They go beyond the data collection phase in health institutes and include a broader history of the patients' health [6]. EHR improves the capability to diagnose diseases and find a better treatment. It consists of different types of structured data such as drugs and the dosages, and also unstructured data such as health descriptions, which consists of the reasoning behind the prescription of the drugs.

EMR are widely used these days because they reduce the use of paper-based records and can track the data and information and also monitor patients' health and treatments due time, which can improve and develop the overall quality of the health care system. This clinical data is collected in a digital format and sent to the health centers; hence it is safe, up to date and accurate [6]. EHR of the patients' acts as a repository of the information about their health status and diagnosis in a computer based format.

Today Information Technology is capable of transforming the way health care is represented and also documented. Unlike traditional clinics, databases in modern clinics and health care institutes can repeatedly capture organized data about the medications, diagnosis, laboratory results, imaging, scans and other aspects of health care. Clinical data that describes and represents the general health conditions and the required treatments for the patients has become a very popular research area these days.

Information Retrieval (IR) is a way to gather relevant and applicable resources, which can be based on metadata or full text indexing. It is usually used for retrieving unstructured data such as enormous groups of electronic text and data. Google is one of the most popular search engines used in IR services, provides easy and reliable access to the most recent and up-to-date information and is becoming one of the main forms of accessing information and unstructured data which has no clear specification or definition and is overtaking the old-styled ways of searching. Web search is by far one of the most popular methods of obtaining information in IR, which can provide a search over too many documents and resources online.

This method is now being used in many universities and libraries around the world. The way Information Retrieval process works is, a user enters a query in the system, which can be a document, image, audio, video etc.; and then several data objects are used in the entered query, which can have similar or different degrees of relevancy. IR system will then score and rank the objects; the top ranked ones are then presented to the users. On the other hand, we first need to search and collect the proper data set. Dataset is a collection of documents, web pages and web sites consisting of the topics related to our research area [5]. After the data is selected, the information will be processed, and then a model will be built using different algorithms and techniques for the collected data set. Following this step, we need to index the data in order to find the related documents based on the similarity of the keywords in the database [1] [2] [3]. When searching through the database, a search term is submitted and then the system will check the query terms and keywords to find information. Query is symbols and circumstances that help us find the proper information regarding the research topic. Information Retrieval system finds the information that is relevant to the users' query and search through the collection of either structured or un-structured data. IR software will then classify the entered data with the existing information in the database and return the results. When searching in the databases, it is usually hard to find the exact term that we are looking for. After the term is found, we need to develop our search to find relevant documents and sometimes need to look into different databases and link the contents [4]. IR system ranks the documents in response to a document and then the occurrence of the query is scored to find relevancy and is sorted in decreasing order. Evaluation is a measurement that combines relevant topics using TREC_eval (Text Revival Evaluation Conference) evaluation tool which was formed by Donna Harman and colleagues for large number of test collections. Working with these new data sets proved that the earlier weighting models and ranking strategies were not suitable for different collection types [5]. The purpose of TREC is to support the research in Information Retrieval and help to provide a method for evaluating the text retrieval in enormous procedures.

There are two main features for measuring the performance of the IR system:

1. Efficiency measures in terms of time (response time) from sending a query by the user and getting the results, and space for storing the index and the data structures.

2. Effectiveness measures the relevancy of the document to the given query. After the relevancies of the context of the documents are ranked the novelty of these contexts are considered.

The evaluation of Information Retrieval System performance is an important factor in improving the techniques and maintaining the effectiveness and measuring how successfully the IR model can reach its goal. System will assign weight for the retrieval documents and provide ranking. More relevant documents are ranked in advance of the documents that are less relevant. It is important for the system to return the results fast, accurate and reliable to the users. It is also useful to compare and rank the results obtained from the different types of retrieval techniques to gain the best results possible.

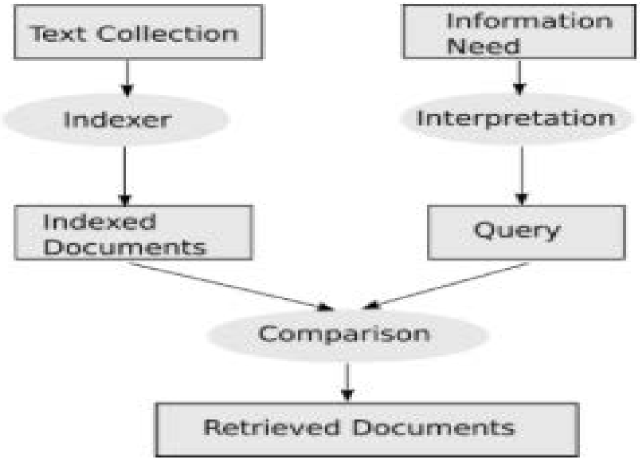
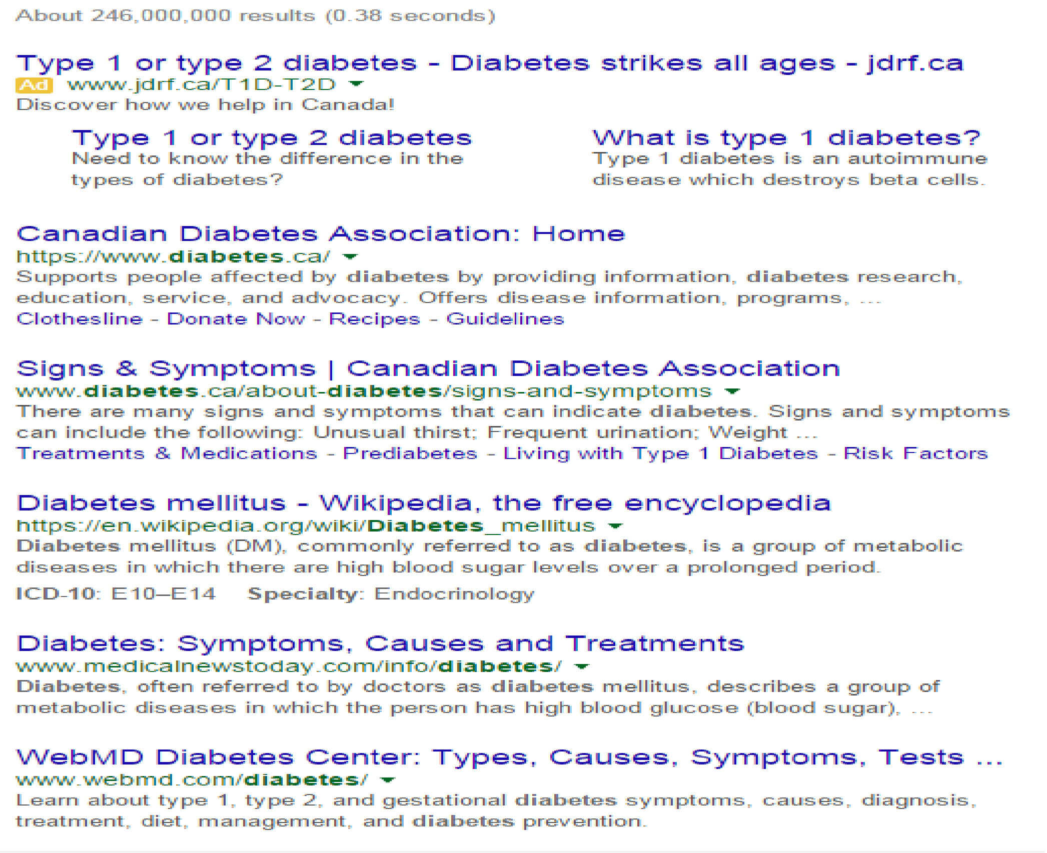


Figure 1 : Document Retrieval

[67] Christopher D. Manning, Prabhakar Raghavan, Hinrich Schütze – An Introduction to Information Retrieval, 2009.

Examples of Information Retrieval:

We use Google to search for the work “diabetes, as clearly shown in the picture 246000000 results pops up in less than 1 second.



About 246,000,000 results (0.38 seconds)

Type 1 or type 2 diabetes - Diabetes strikes all ages - jdrf.ca
Ad www.jdrf.ca/T1D-T2D
Discover how we help in Canada!

Type 1 or type 2 diabetes
Need to know the difference in the types of diabetes?

What is type 1 diabetes?
Type 1 diabetes is an autoimmune disease which destroys beta cells.

Canadian Diabetes Association: Home
https://www.**diabetes**.ca/
Supports people affected by **diabetes** by providing information, **diabetes** research, education, service, and advocacy. Offers disease information, programs, ...
Clothesline - Donate Now - Recipes - Guidelines

Signs & Symptoms | Canadian Diabetes Association
www.**diabetes**.ca/about-**diabetes**/signs-and-symptoms
There are many signs and symptoms that can indicate **diabetes**. Signs and symptoms can include the following: Unusual thirst; Frequent urination; Weight ...
Treatments & Medications - Prediabetes - Living with Type 1 Diabetes - Risk Factors

Diabetes mellitus - Wikipedia, the free encyclopedia
https://en.wikipedia.org/wiki/**Diabetes**_mellitus
Diabetes mellitus (DM), commonly referred to as **diabetes**, is a group of metabolic diseases in which there are high blood sugar levels over a prolonged period.
ICD-10: E10–E14 **Specialty**: Endocrinology

Diabetes: Symptoms, Causes and Treatments
www.medicalnewstoday.com/info/**diabetes**/
Diabetes, often referred to by doctors as **diabetes** mellitus, describes a group of metabolic diseases in which the person has high blood glucose (blood sugar), ...

WebMD Diabetes Center: Types, Causes, Symptoms, Tests ...
www.webmd.com/**diabetes**/
Learn about type 1, type 2, and gestational **diabetes** symptoms, causes, diagnosis, treatment, diet, management, and **diabetes** prevention.

Figure 1.1 : Example of Searching for Diabetes

Document Retrieval is the process of finding matched text records for a user query and ranking the matched results based on their relevancy. There are two methods of indexing in DR, Form-Based that addresses the particular unstructured text and compares it to the matching word of the search query, and Content-Based that is the process of finding connection between queries and documents. Content-Based DR usually uses invert index algorithm for retrieving relevant documents. Document Retrieval is the concept of providing users with the relevant documents in a secure and fast manner [58].

Natural Language Processing (NLP) started in 1950s and is a field in Artificial Intelligence, computational linguistic and computer science which is related to the interaction between computers and human language. Modern NLP algorithms are based on statistical machine learning which inputs the enormous amount of data focused on statistical models and then makes the probabilistic decision based on the weight of the data and converts the text information into an understandable natural human language [39]. NLP uses different techniques such as stemming (which is the process of mapping the searched words to some form), chunking (which is the process of separating the words into two or more phrases), word sense disambiguation (which is the process of distinguishing the correct word), part-of-speech tagging (which is the process of assigning syntactic class to each word in the actual text to resolve the uncertainty) and so on [59].

Information Extraction however is the process of extracting novel and special data in a collection of documents in a specific domain. Information Extraction converts the elements of unstructured texts in the document into a structured data and also identifies the relationship among the data in the database. IR returns documents but IE returns and extracts specific type of information and details. These days IE is useful for extracting biomedical and clinical text;

Biomedical texts are specific information that are written in the books, medical article, abstracts and so on, but clinical tests are the information and facts that describe patients' health, medical history, diagnosis diseased and so on. One approach in IE is pattern matching which achieves basic patterns over the diversity of structures but it lacks the ability of generalizing restrictions and the ability of reaching to the new domain [38].

Information Extraction is used mainly in NLP system to extract precise and accurate information from the text or document and creates records in the database. While the information comes from too many different sources, ontology should consider all references in the text by figuring out if the references in different resources relate to the same object in the research domain.

Ontologies deliver formal and clear terms of conceptualizations which is why they are really important these days in IE process and semantic Web applications.

Ontology for General Medical Science (OGMS) is ontology of the medical and clinical entities such as diseases, diagnostics and patients' health care but OGMS only covers the human ontology [50].

Natural Language Processing distinguishes the difference between Information Retrieval and Information Extraction; IR retrieves the relevant subject from a collection of documents and presents them to the users. The majority of the IR systems search for the keywords. PubMed search engine is using IR techniques, which is a free search engine to access MEDLINE (Medical Literature Analysis and Retrieval System Online) database regarding biomedical information [57]whereas IE (Information Extraction) is the process of retrieving the relevant medical information and facts from the documents, then structuring these documents and adding semantics to them [48]. IE system can be used in health care to keep records of patients' health problems, diagnoses, test results, symptoms and the required treatments in their files. These systems can help the health institutes to keep track of all the useful information.

The amount of medical and clinical data these days are used based on the amount of information extracted. Natural Language Processing is used for translating and converting this vast amount of information and text into more user friendly and human language texts.

Ontology-Based Information Extraction is now considered as a subfield of IE in a way that the result of the IE is characterized within ontology. Ontology is usually specified for a particular domain and since IE extracts data from a specific domain, combining these two methods can be useful.

Bioinformatics, on the other hand, is the science of managing, analyzing and extracting the information. Along with Information Retrieval and understanding the genome, DNA and protein sequencing since 1980, Bioinformatics uses computers to gather molecular biology and analyze DNA and Protein structure using mathematical algorithms. Research in bioinformatics has been changed by the new data. The biomedical data can vary from simple to complex. Database is used to maintain and obtain the information. Gene on the other hand is a protein coding area, which makes a small part of the genome. Evaluation is a measurement that combines relevant topics using TREC_eval evaluation tool.

Databases retrieve and analyze the information in different steps: First they retrieve the sequences by features and annotations or by patterns. Then, they compare those sequences. In Biomedical Information Retrieval, database translates DNA sequences to protein sequences. After this step is done, database understands the protein structure using statistical methods. Then the database should identify the pattern in order to see the relationship between the sequences. The last step is to provide the molecular graphics in a classified structure [5] [6].

1.1 Terrier:

Terrier is an open source search engine for collecting, indexing and querying the large scale of documents, and retrieving the results. The program was developed in Java in the University of Glasgow, Computer Science department. It runs on both Windows and UNIX. Index/ and results/ are collected in var/. In order to index, Terrier parses the collection of documents and then develops the tokens and creates compacted index structures.

Indexing Architecture:

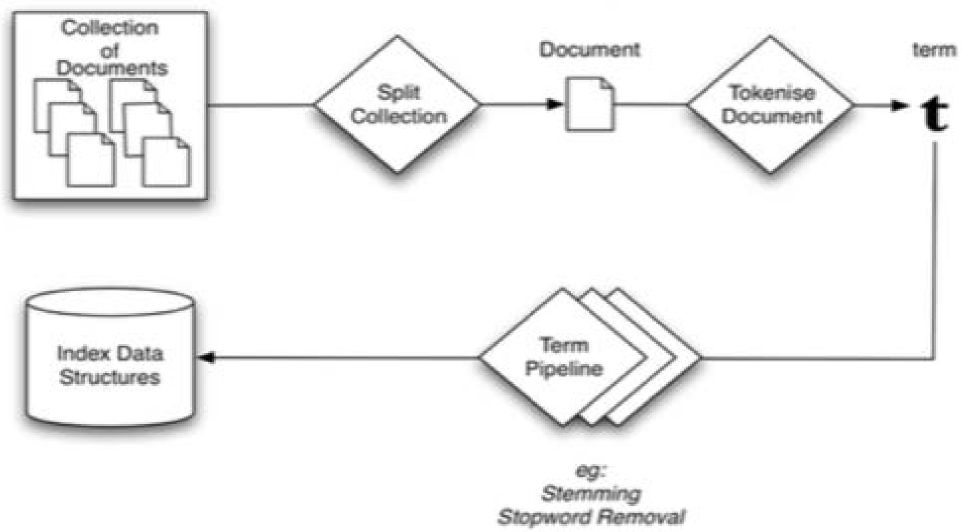


Figure 1.1.2 : Example of how Terrier Works

[7] http://trec.nist.gov

Then Terrier pipelines the tokens by passing the tokens through the term pipeline in two ways. One is transforming the term while the other dropping the term. Indexing uses Lexical or Direct inverting to index. Lexical indexing stores the collection of vocabularies and the matching documents while Direct Indexing stores the identifier of the terms that are located in each document and consists of the information about the document [7].

Inverted Indexing is indexing data structure for full text search of the words when a document is added to the database. Inverted Index is the central element of indexing algorithm and the main purpose of using Inverted Indexing is to improve the speed of the query. In order to build an Inverted Index, the system will first needs to collect the documents to be indexed, and then it tokenizes the text in order to change each document into a list of tokens. Later, the lists of normalized tokens are formed and the terms are indexed to create Inverted Index.

Lexical Indexing is the process of converting the query terms into words or tokens. It is the first step of automatic indexing and query processing and produces the index terms and tokens that are parsed into an internal demonstration for comparison with the other indexes.

Terrier consists of many DivergenceFromRandomess weighting models along with statistical retrieval models such as BM25 and also Language Model. It provides real-time indexing, flexible retrieval, efficient and effective ranking strategies.

Terrier Platform is flexible and useful in text retrievals on TREC standard and CLEF test collation. Terrier can read and index tagged, TREC formats and Web collections. It consists of different weighting and statistical retrieval models. It has also been used for ad-hoc and cross-language retrievals. Terrier platform has successfully been used in Web and TREC 2002-2004 and considerably achieved better results against the median of the runs submitted by the participants in CLEF. It also provides and supports languages other than English [7].

2. Research Problems:

Information Retrieval system is designed to ideally evaluate the exact relevance of each document in a collection and create a query. However in a retrieval process, it is sometimes difficult to calculate the relevancy of the document. In this research, there were multiple query terms and therefore various options to find the relevancy. The document collection and the amount of given data provided by CLEF was really large and the goal was to search through the entire collection and get the most relevant and matching document in the shortest period of time and thus keep the system response time for processing this high amount of data as minimal as possible. The size of the data collection as well as the format of the data set was not readable by Terrier Platform, the IR system initially planned to use; hence the first step was to change the format of each collection in order to be able to index the quires. This solution resulted in another minor problem which was a need to use too many queries for indexing. Finally, I ranked the retrieval results and looked for the best possible answer among the documents.

Finding the relevancy in the documents was the first and most important factor in this research. My main motivation to start working on such topic in health care and biomedical environment was to retrieve and find the best possible results to help the doctors improve their diagnosis tools and find better treatments for their patients as well as help the patients have a better understanding of their health issues and diseases.

In general, this research addresses the following questions:

- Why Information Retrieval is important in medical field?
- What is the use of EMR and EHR?
- What is medical ontology?
- Why did I use PRF as my preferred weighting model for ranking and UMLS for translating the medical terms?
- How can UMLS help both patients and doctors?

2.1 Contributions:

I first used BM25 as the base weighting model to rank and evaluate the indexed queries in order to obtain the relevancy, and then I used Pseudo Relevance Feedback (PRF) field based weighting model. Later, I compared the results obtained from the different runs and ranked them using probabilistic modeling such as Vector space and Unified Modeling Language System. My main idea was to find a unique way to help the patients understand the biomedical science by using a simple dictionary and also be able to make a link between medical terms and everyday vocabularies in order to have a better understanding of their disease and possible ways of treatment. I proposed using medical ontology to retrieve the medical data as my approach and then combined it with using UMLS (Unified Medical Language System) for indexing and looking for the matching documents and queries regarding patients and their diseases. UMLS is a tool and software that is used to understand the health care and the medical terms and vocabularies more precise and accurate across different computer systems. The queries were then mapped to the relevant medical concept in the ontology so that the meaning of each medical term could be determined. Each medical term consists of statistical information of the particular disease in the patient's records. In order to define the degree of the relevancy of each query term and rank the results That being said, I used UMLS which can act as a repository that provides a structured database and software to further understand the ontology of biomedical concepts and improves the electronic medical records as well as obtains the health data and records [16]. UMLS translates the diagnosed diseases into a natural language that is easy to read and comprehend for the patients. UMLS has more than million terms for medical concepts whereas biomedical vocabularies consist of the relations between these terms. These two can be related internally or can also be linked to the external knowledge resources and databases to obtain up-to-date information.

This method provided automatic local analysis for obtaining improved retrieval performance as well as finding the most relevant documents.

3. Probabilistic Model:

3.1 BM25

BM25 is the classical probabilistic retrieval model and stands for “Best Matching”. It was first developed in 1970s and has been known as one of the most effective ranking functions used in Information Retrieval for many years. The advantage of using BM25 over other models is its ability to perform fast indexing and obtain reasonable results. It is the base weighting model and is used for ranking the matching documents based on their likelihood of relevancy in the searched query terms. BM25 is also mentioned as “Okapi BM25” in some references. BM25 can be divided into two different ranking modifications:

- BM25F in which research documents and queries are collected from various different fields and formats containing headers, main body, footnotes, texts and etc.
- BM25+, on the other hand, is the addition to BM25 that was established for ranking and scoring the long documents and finding the relevancy between the query terms. BM25+ scores the documents using the formula below:

$$\mathrm{Score}(D,Q) = \sum_{i=1}^{n} IDF(q_i)\left[\frac{f(q_i,D).(k_1+1)}{f(q_i,D)+k_1.\left(1-b+b.\frac{|D|}{avgdl}\right)} + \delta\right] \qquad [23]$$

Where D is a document and Q is a query then $f(q_i, D)$ is term frequency in the document.

$|D|$ is the length of the document , and $avgdl$ is the average document length in the text collection.

k_1 and b are free parameters.

$IDF(q_i)$ is the IDF (inverse document frequency) weight of the query term q_i.

In this research, I used BM25 normalization model as my base line. In BM25, the weight of each term is assigned by taking into account the query term frequency in the documents [9]. A document's weight for a query is given by the sum of its weight for each term in the query:

$$BM25(D) = \sum_{i=1}^{|Q|} W(q_i, D)$$

[10]

i=1 where w is the term weight, and |Q| is the length of the query Q [10].

Terrier provides two different implementations of BM25,

1. Standard BM25 implementation
2. BM25-DFR [8].

3.2 DivergenceFromRandomness (DFR):

DivergenceFromRandomness or DFR is also one of the first models of Information Retrieval. In this model, the weights of the terms are calculated by measuring the divergence among the distribution made by a random process and the actual distribution itself. DFR first selects the basic randomness model and after applying the first normalization, it tries to normalize term frequency [8]. The term weight is contrarily related to the probability of the term frequency in the document “d” obtained by model “M” of randomness [4].

$$Weight(t|d)\alpha - logProb_{M}(t \in d|Collection)$$

[4]

In other words, the term weights are measured by calculating the divergence between a term allocation obtained by a random process and the actual term distribution [8]. DivergenceFromRandomness model is similar to the language model and is based on the idea that if the frequency of the document term in the collection is more divergent, the more information is carried by the word in the document.

DFR is based on two randomness models:

1. Binomial Model: is the existence of a single term in a document, it is the probability of having a failure or a success; it is sometimes called Bernoulli trial.

$$Prob_1(tf) = Prob_1 = B(N, F, tf) = \binom{F}{tf} p^{tf} q^{F-tf}$$

$$with\, p = \frac{1}{N}\; and\; q = \frac{N-1}{N}$$ [67]

N is the number of documents in a collection

F is the total occurrence of a term

tf is the number of occurrences of a term in a document

2. Bose-Einstein Model: is the random distribution of a word in documents. This model is the probability of achieving and enhancing the statistical relationship between the terms by calculating the possible occurrence of the relevant document.

$$-logProb_G(t \in d|Collection) = -\log\left(\left(\frac{1}{1+\lambda}\right).\left(\frac{\lambda}{1+\lambda}\right)^{tf}\right)$$ [69]

TF is the term-frequency of the term t in the Collection

tf is the term-frequency of the term t in the document d

N is the number of documents in the Collection

p is 1/N and q=1-p

4. Ontology based approach using Pseudo Relevance Feedback for ranking and UMLS for Translating:

In computer science, ontology is referred to the demonstration of entities, types, ideas, events and properties that exists for a specific domain of terms and is used to create a relationship between the sets and documents that are used and calculated. It is also a description of the concepts and the relations between them that is mostly used in Web applications because it can provide a shared knowledge about the terms that are used in the real world [31].

Ontology basically deals with the nature and the association of the certainty which nowadays has been combined with the computer science research area.

According to Aristotle ontology is the science of being and he defined 10 categories for his theory, which later Franz Brentano summarized Aristotle's categories in to more organized structure. This idea was later challenged by Emmanuel Kant; he classified his theory in to 4 main categories such as Quantity, Quality, Relation and Modality and divided each 4 in to 3 more subcategories [73].

In recent years medical ontologies are commonly being used in Information Retrieval and Biomedical Informatics. The main purpose of ontology is to reformulate the queries in order to improve the quality of the obtained results and therefore, it has been moved from the artificial intelligence laboratories to the desktops of the specialists to provide operational configuration for supporting diagnoses based on the data. It is a way to provide more common and user-friendly terms in order to enable the communication between the patients and the medical doctors. It can also be used as query for the information resources [35].

Our approach in this thesis was to develop and improve a unique retrieval tool which I am explaining in my thesis that is going to be beneficial for doctors to find patients with similar diseases and look for better treatments. Medical Ontology is a knowledgeable specification and definition of a term and concept. It can be used to filter out un-necessary and un-related information based on the resources in a research domain. It has been a very useful area in Information Retrieval and Biomedical Informatics [24]. However, searching for biomedical information in an enormous collection of medical data is not an easy task and that is why we require tools and resources to make the Information Retrieval process faster and more accurate. We realized that, by using medical ontology, we were able to develop the queries and improve the results in a timely manner.

4.1 Different Types of Ontology:

- *Top-Level Ontology*:

Are the most general ontologies which defines the top level in the ontology and other ontologies are connected either directly or indirectly.

- *Domain Ontology:*

Is used to define a specific domain, such as medicine, politics, and clinical trials. Domain Ontology is usually attached to the top-level ontology.

- *Task Ontology:*

Is used to describe the top-level Ontology for a specific activity.

- *Domain-Task Ontology:*

Is used to describe the domain-level ontologies on the domain-specific activities.

- *Method Ontology:*

Is usually used to define a relevant concept and the relations between them.

- *Application Ontology:*

I usually used to achieve the information in a particular application [73].

5.1 Relevance Feedback:

Relevance Feedback is a technique in Information Retrieval which collects the results that are initially returned from the query and uses that information to figure out if those results are relevant to initiate implementation of a new query [26]. This technique is used to improve the users' initial query and facilitate a better retrieval process. First, the user forms a query and then the system returns the preliminary set of results and helps the user to mark the returned documents as relevant or non-relevant. It also gives feedback through which the system can process a better and more accurate ranking based on the given feedback to improve the retrieval results [27].

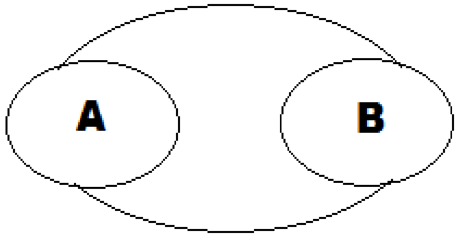


Figure 5.1.1 : Relevance Feedback Between and A and B

Feedback exists between A and B and affects both parts.

First we have to find the top document which can be reached by using inverted index algorithm, and then we have to find the highest terms in the top documents.

There are three different types of Relevance Feedback:

1. Explicit Feedback: which is gained from the evaluators of the relevance documents retrieved for a query. The degrees of relevancy in the documents are referred to as numbers and/or letters.
2. Implicit Feedback: This is based on the users' search performance on the documents they select and also the duration of the search.
3. Pseudo Feedback: It is also referred to as blind relevance feedback and is used for automatic local analysis for retrieving improved results [26].

Relevance Feedback can be very effective in the case that users don't have very good information about the data collection therefore RF can track and calculate users' needs to the best of their knowledge. RF can be extremely useful for Image Search by helping the users to see the returned results and easily understand the relevant and non-relevant images and documents. The way it works is that RF searches over the images and then the users can view the initial query of the results and locate the relevant results [27]. Using RF enables us to add the query terms and modify their weight and provides the users with the relevant documents in the initial set of results. These calculations are later used to improve the ranking process and give a higher rank to the more relevant documents. That is the reason why RF can significantly improve the effectiveness of the Information Retrieval process even though sometimes it is expensive to use [26].

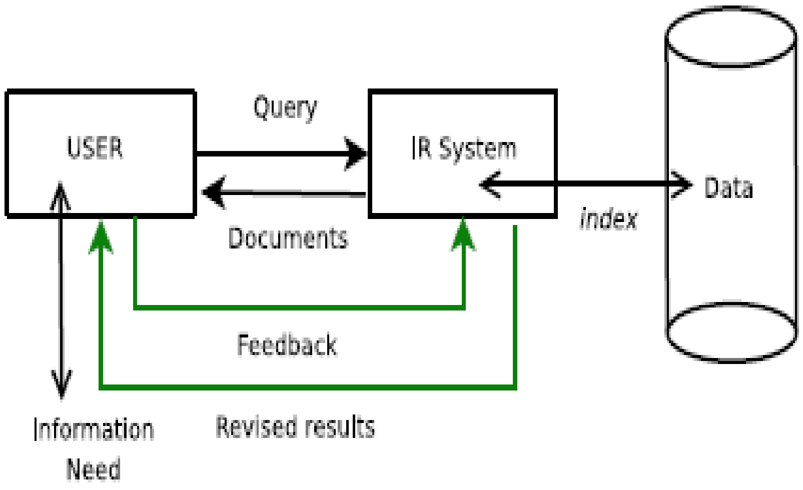


Figure 5.1.2 : IR System's Approch when user enters a query

[28] Sprachwissenschaft, International Studies in Computational Linguistics, Winter 2007.

In this example user enters a search query, the Information Retrieval system looks in the database to find the required results, relevant documents are then returned to the user. The feedback will then be sent to the IR system whether more results are needed or not.

We should take into consideration that RF does not work when there is a misspelling in the query or when we require cross-language retrieval. Another short coming is when the vocabulary used in a term is vague or unclear. However RF is more appropriate for Information Retrieval on the Web.

Example of Relevance Feedback:

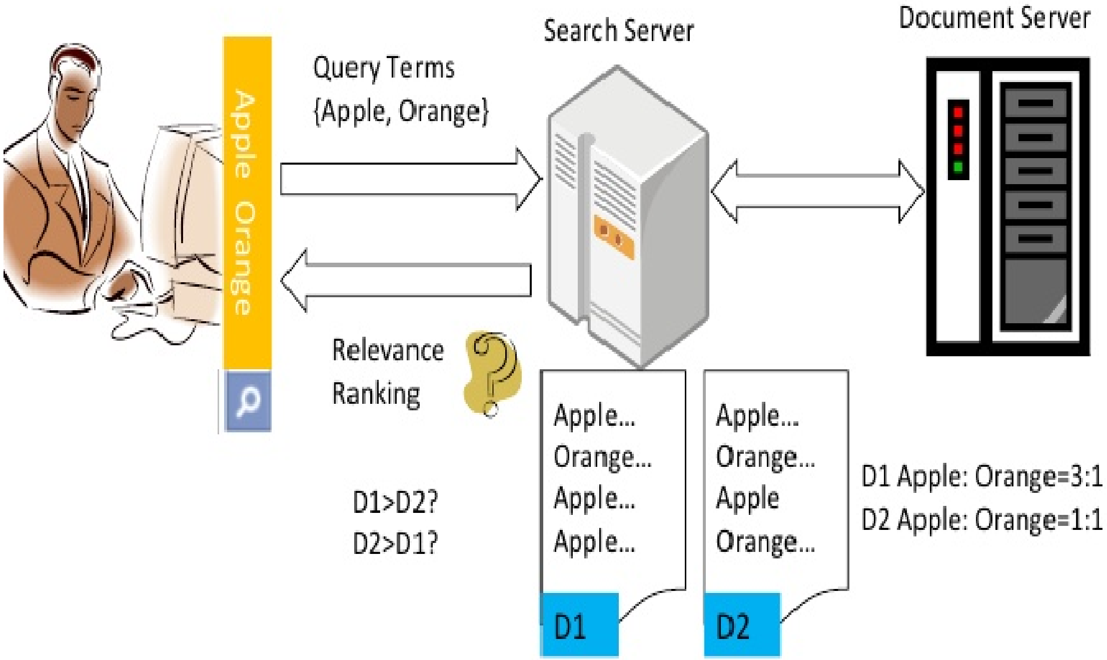


Figure 5.1.3 : Searching for matched documents.

[74] www.slideshare.net

In this example user is searching for an article to introduce both “Apple” and “Orange” there for his input query is {Apple, Orange}. In this example system finds two matched documents. Apple appears 3 times in document1 and Orange appears once, however in document2 both Apple and orange occur equally.

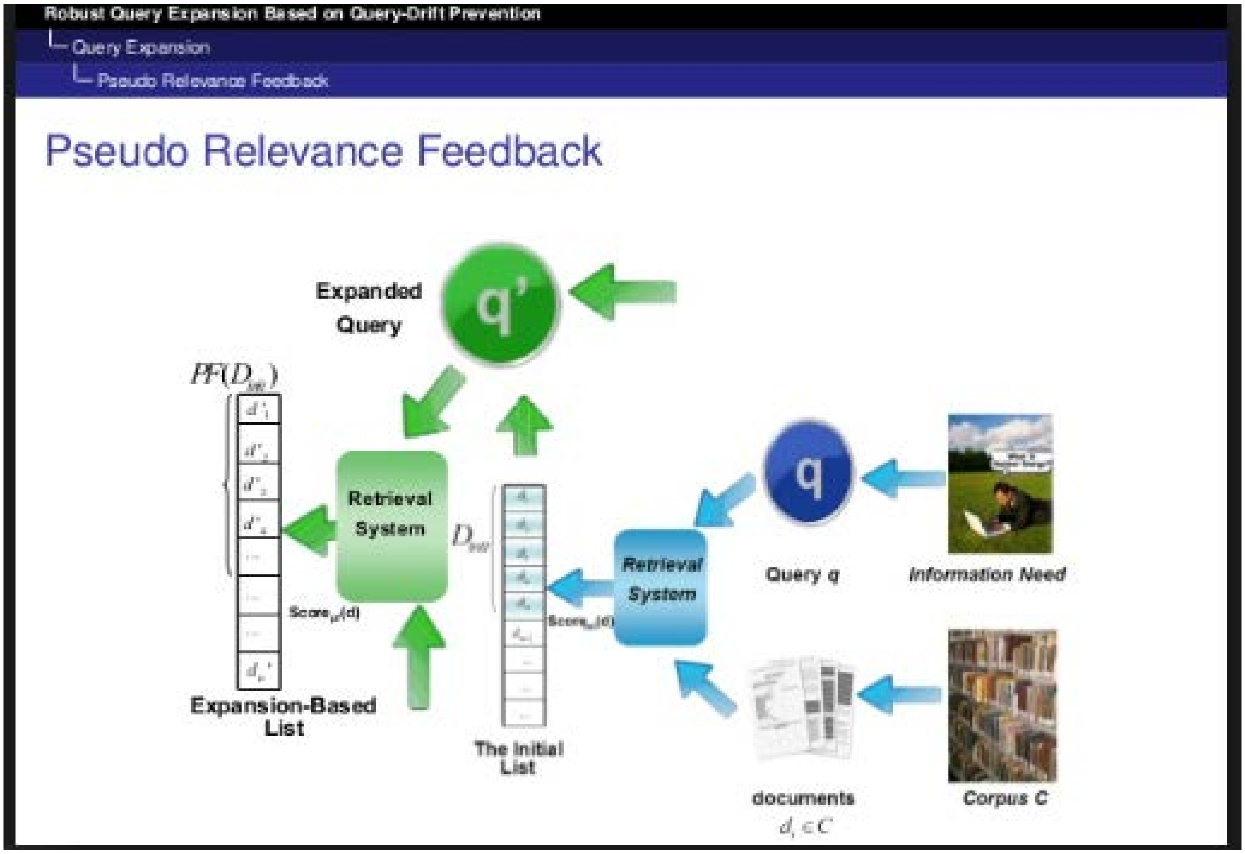
5.2 Pseudo Relevance Feedback:

Pseudo Relevance Feedback is used for automatic local analysis for retrieving more accurate results. The quality of the results is based on the top ranked documents. This method is used to modify the original query provided by the user and then add the required terms. The system returns the retrieved results and selects the relevant documents. At this point, system calculates a better term based on those collected results. Then the query is prolonged and the retrieval results are displayed.

PRF is a query expansion technique that works based on the assumption that the top ranked documents, which have been originally retrieved, are relevant. We can identify the high scored documents that are more relevant to the initial retrieval.

The advantage of this method is that if some relevant documents get lost during the query expansion, it can be retrieved and improve the performance of the Information Retrieval process especially in TREC and ad-hoc tasks [25].

Example of Pseudo Relevance Feedback:



Robust Query Expansion Based on Query-Drift Prevention

└ Query Expansion

└ Pseudo Relevance Feedback

Pseudo Relevance Feedback

Figure 5.2.1 : Query Expansion

[75] www.slideshare.net

User selects a query that is built from the query logs or thesaurus, and then the words are extracted according to the initial query word. The words that occurred multiple times belong to the same query filed.

PRF is based on three areas:

1. Documents, which are being calculated based on their relevancy.
2. Result extraction, which is the list of terms and queries.
3. Reorganizing the query term obtained from the previous results.

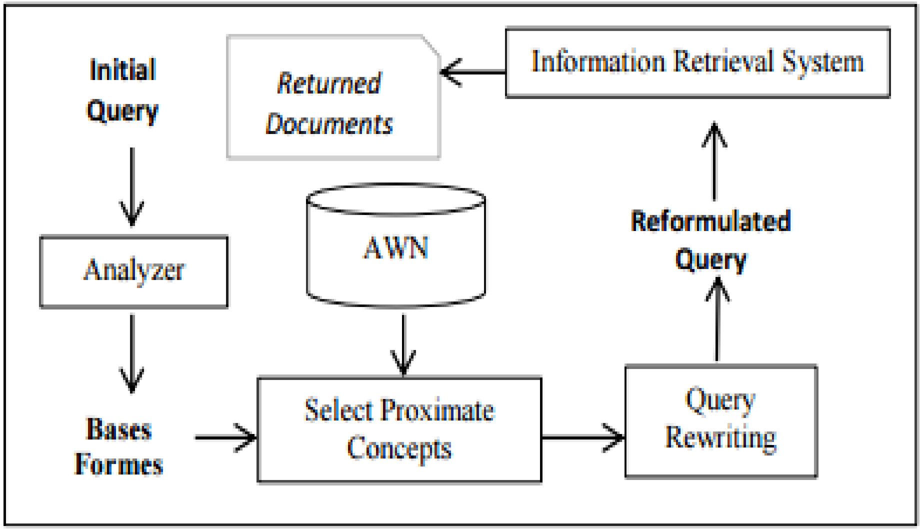


Figure 5.2.2 : How Pseudo Relevance Feedback works.

[25] Mohammed El Amine Abderrahim, Concept Based vs. Pseudo Relevance Feedback Performance Evaluation for Information Retrieval System.

PRF or Pseudo Relevance Feedback is a technique to improve the accuracy of the retrieved results. PRF works based on the assumption that the documents with the top ranked score results which are obtained from the initial retrieval process and relevant to the query terms help improve the performance. It is used to do normal retrieval in order to find the initial set of the most relevant documents and improve the performance especially in TREC and ad-hoc retrieval tasks [12]. In PRF, scattering of the documents may have an impact on the results. In this method the associated terms are extracted from the top document set and return the response based on the original query. Then the terms are added to the initial query and run again to retrieve a new set of documents to be returned to the users. As we see, PRF is based on the assumption of expanding the query by using the terms associated to the query term. These terms are built from the local document set [25]. During the query run time, all queries look for the related terms in the applicable knowledge structure and modify the original query that was entered by the user in order to develop the queries. Some techniques are used in here such as adding proper terms obtained from the original documents list which has been retrieved [25].

My preferred method in this research was to use Pseudo Relevance Feedback in retrieval process in order to achieve better results and be able to give feedback on the relevancy of the documents. Taking this approach, I first obtained the queries from the indexes which I had converted to a proper format earlier using a java program, and then used Information Retrieval application (Terrier platform in this research) to index and obtain the results to determine the relevant and non-relevant documents [9] [13]. I was looking for the documents that were as similar as possible to the original query and return the results.

Using query expansion in PRF is a technique or a methodology that adds the new terms into the query to reformulate the initial query in order to obtain the better result and enhance the overall performance of the Information Retrieval System. It is based on the assumption that the first retrievals are more relevant to the initial query. Therefore the terms that are closer to the query term are more likely to be relevant to the query topic.

5.3 Rocchio Algorithm:

Rocchio algorithm is used for implementing relevance feedback, which was introduced in 1970s and was industrialized using Vector Space Model. The theory behind Rocchio Algorithm is based on the consideration that users have a general knowledge of distinguishing which documents are relevant and which are non-relevant, and also negative weights are usually not calculated. Rocchio is a classic algorithm used to implement the relevance feedback by improving the query illustration. It is also a way to join the Relevance Feedback information to the Vector Space model. All the documents use a retrieval model. Each document is represented as a weighted term and each of these queries are then developed by taking the linear combination of initial query term vector and the feedback documents vector [19]. A vector of the term weights characterizes the documents. Then the distance between the query points and the documents is ranked according to their possible relevancy. Vector model represents index terms these terms are usually single words or keywords. Vectors are also used to compare documents in queries based on their similarities [15].

When documents are about to be ranked for a query, an ideal query should rank the relevant and non-relevant documents. Rocchio algorithm calls this query an optimal query vector.

Using this algorithm changes search queries into relevant and non-relevant documents [10].

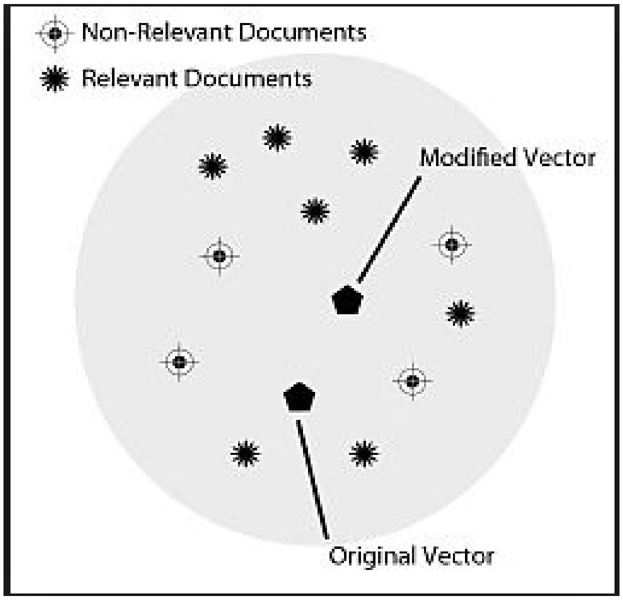


Figure 5.3.1 : the Spread of Relevance and Non-Relevance Documents

[12] J. Rocchio. Relevance Feedback in Information Retrieval, pages 313–323. 1971.

In this example optimal query vector is used to point out the relevant and non-relevant documents which are spread unevenly.

The formula for Rocchio Relevance is calculated as follow:

$$\overrightarrow{Q_m} = \left(a.\overrightarrow{Q_0}\right) + \left(b.\frac{1}{|D_r|}.\sum_{\overrightarrow{D_j} \in D_r} \overrightarrow{D_j}\right) - \left(c.\frac{1}{|D_{nr}|}.\sum_{\overrightarrow{D_k} \in D_{nr}} \overrightarrow{D_k}\right)$$

[12]

“a” is the original query weight, “b” is the weight of the related documents and “c” is the weight of the non-related documents [12].

D_r and D_{nr} are not vectors. Q_m is modified query vector, Q_0 is the original query vector D_j is the vector to the related document and D_k is a vector for non-related document.

5.4 Unified Medical Language System (UMLS):

UMLS is a set of files and software that links health care information, biomedical terms, drug names and disease codes into different computer systems in a way that the meaning of biomedicine and health is understandable by the computer system [16]. UMLS consists of databases and software tools that provide a mapping structure between biomedical vocabularies and terminologies to be used by system developers in biomedical informatics. UMLS is designed and maintained by US national Library of medicine. It is used to enhance and improve applications such as dictionaries and language translators [16].

I used UMLS as a repository to clarify the collection of complex and enormous volume of medical data provided by CLEF into a data structured queries and also used these queries to match with the topic and search for the relevancy. Each medical term contains statistical information of the particular disease in the patient’s records. When all queries are indexed into a structured data, then PRF weighting model is used to find the degree of relevancy of each term in respect to the topic to compare the scores. All the topics are related to the patient’s health record and the treatments used.

UMLS has three different knowledge resources known as Metathesaurus, Semantic Network and

Specialist Lexicon and Lexical Tool [16].

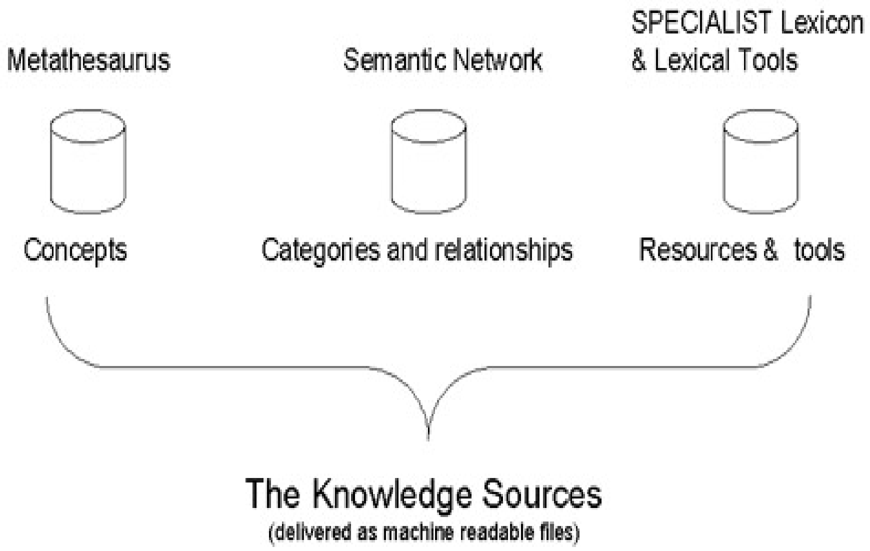

Metathesaurus
Semantic Network
SPECIALIST Lexicon & Lexical Tools
Concepts
Categories and relationships
Resources & tools
The Knowledge Sources
(delivered as machine readable files)

Figure 5.4.1: Three Different Knowledge Resources Of UMLS

[16] U.S. National Library of Medicine, 8600 Rockville Pike, Bethesda, MD 20894, National Institutes of Health, Health & Human Services

1. Metathesaurus: This method is used to retrieve codes, vocabularies and identical terms.

2. Semantic Network: This is used to view the definition, relationship and the structure of the documents.

3. Specialist Lexicon and Lexical Tools: This tool is for processing natural language.

Metathesaurus:

Metathesaurus is a very great and multi-lingual vocabulary database that consists of the health related information. It is built up on the electronic codes used in patients' health records. It is organized by the combination of the vocabularies and the relationship between the attributes. The way it works is that Lexical tools process the codes and terms and then collect the identical terms into a concept. Then Semantic Network categorizes these concepts by semantic types, combines them and finds the relationship between the attributes and the vocabularies. It outputs the data in a proper format [17]. Metathesaurus reflects the meaning of the concept, when two different vocabularies use the same term for different ideas. Metathesaurus indicates which meaning is referring to which vocabulary and stores the retrieved information in a common format.

Metathesaurus is the major concept of UMLS which works as a repository that holds biomedical concepts and models. Semantic Network is used to categorize Metathesaurus theories whereas Lexical Tools is used for creating and producing the meaning for the biomedical terms. These knowledge resources usually get updated every three month [32]. UMLS knowledge is structured based on the concept. Identical terms are clustered together to structure a concept and link them based on their relationship. Then each concept is later classified by means of semantic types. Since biomedical terms are wide and the related information contains too many vocabularies, external references and resources are also used in order to function as a cross-reference between the medical terminologies and the external resources in the database [32]. Searching can be based on the concept of the term or the Concept Unique Identifier (CUI) or a code. Search can later be limited based on the source and the search type by looking for the exact match or the normalization string of the word and finding the definition along with the relationship between the concepts.

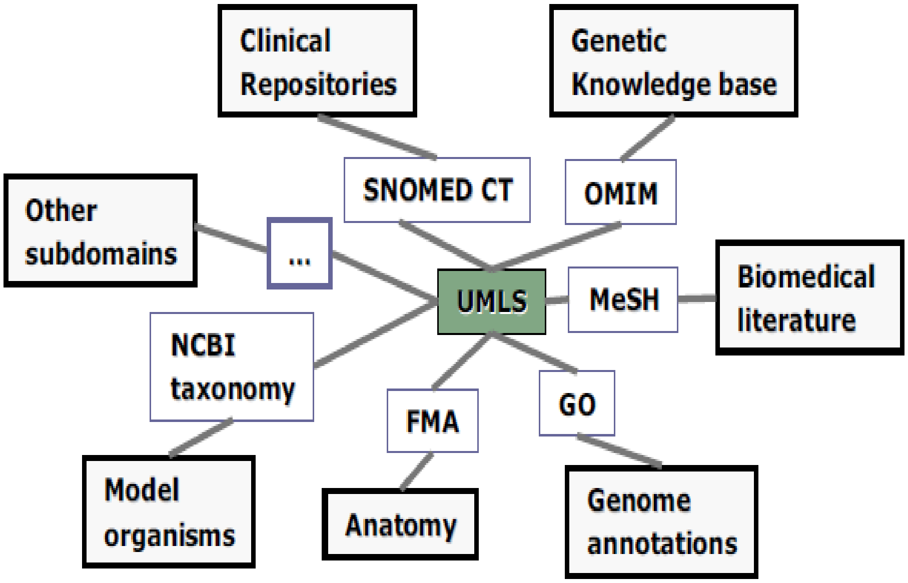


Figure 5.4.2 : UMLS Tools

*[34] Russ B Altman, et al .Text mining for biology - the way forward: opinions from leading scientists**,** Genome Biology 2008,* ***9****(Suppl 2)****:****S7*

UMLS tools are usually programs or web-based services that help users to search and retrieve UMLS data. These tools are:

1. UMLS Terminology Services (UTS)
2. MetamorphoSys
3. Sample Load Script and Data Model
4. RRF Subset Browser

UTS (UMLS Terminology Services) is useful for requesting Metathesaurus license and creating UTS account as well as searching through the Internet and present Metathesaurus, Semantic Network and SNOMED CT. It can also help to download UMLS files such as UMLS knowledge resources, RxNorm updates, SNOMED CT, and the list of problems. Users can also query and retrieve data remotely using UTS. In order to access MetamorphoSys, users should download and save the files that have been released by year, letter code, position, and version. Sample Load Scripts are available online and the files only require some modification to fit the database and the user's need. Rich Release Format (RRF) Subset Browser is a way for searching through local subsets and a means to enable reviewing the raw data for a definite UMLS concept by searching for the CUI, the string and the code of the certain terminology and term [34].

Metamap :

Metamap is a tool for recognizing UMLS concepts and is a configurable program enveloped in National Library of Medicine by Dr. Alan Aronson in Lister Hill National Center for Biomedical Communications. It maps the biomedical text to UMLS metathesaurus concepts for indexing and retrieval. Metamap uses natural language processing (NLP) knowledge intensive approach and linguistic techniques for indexing the biomedical literature in order to identify and classify the medical informatics. It can also be useful for Information Extraction, Classification of the text, Data-mining, natural language analysis of biomedical and clinical txt. It was originally established to improve the retrieval of MEDLINE and clinical reports [36].

This tool is now being used all over the world. It is one of the basic tools of National Library of Medicine medical text indexer that is used for indexing of biomedical terms in NLM. MTI (Medical Information Training) is the main recommendation of indexing and was based on MeSH (Medical Subject Headings) vocabulary, which has later been expanded by NLM. MTI delivers references for the new citations every week to index and process the files. It was also introduced as the first line indexer (MTIFL) in some journals later in 2013.

Indexing life Cycle introduced by NLM is as follows:

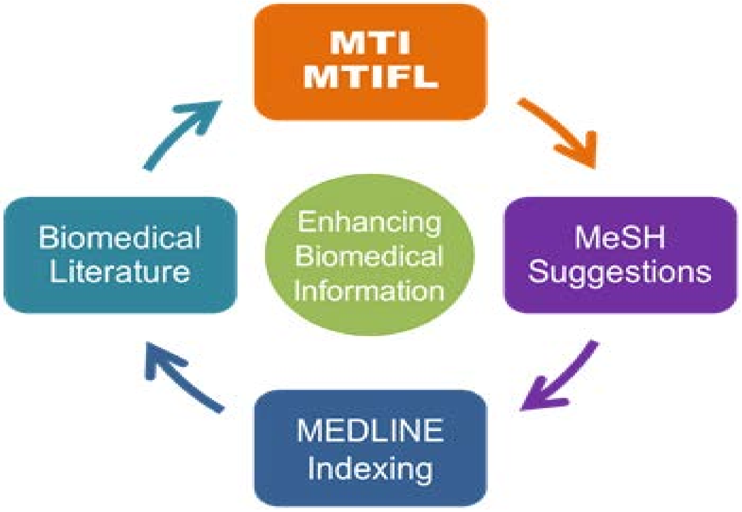


Figure 5.4.3 : Indexing Life Cycle

[36] http://metamap.nlm.nih.gov/

Biomedical terms are first being managed by MTI/MTIFL and deliver a set of MeSH vocabularies which adds topics such as MeSH heading, descriptions, supplementary concept records, publication type, and databank repositories to the MEDLINE indexer. MEDLINE then indexes the medical terms and provides details about the query topic and improves the understanding of the document.

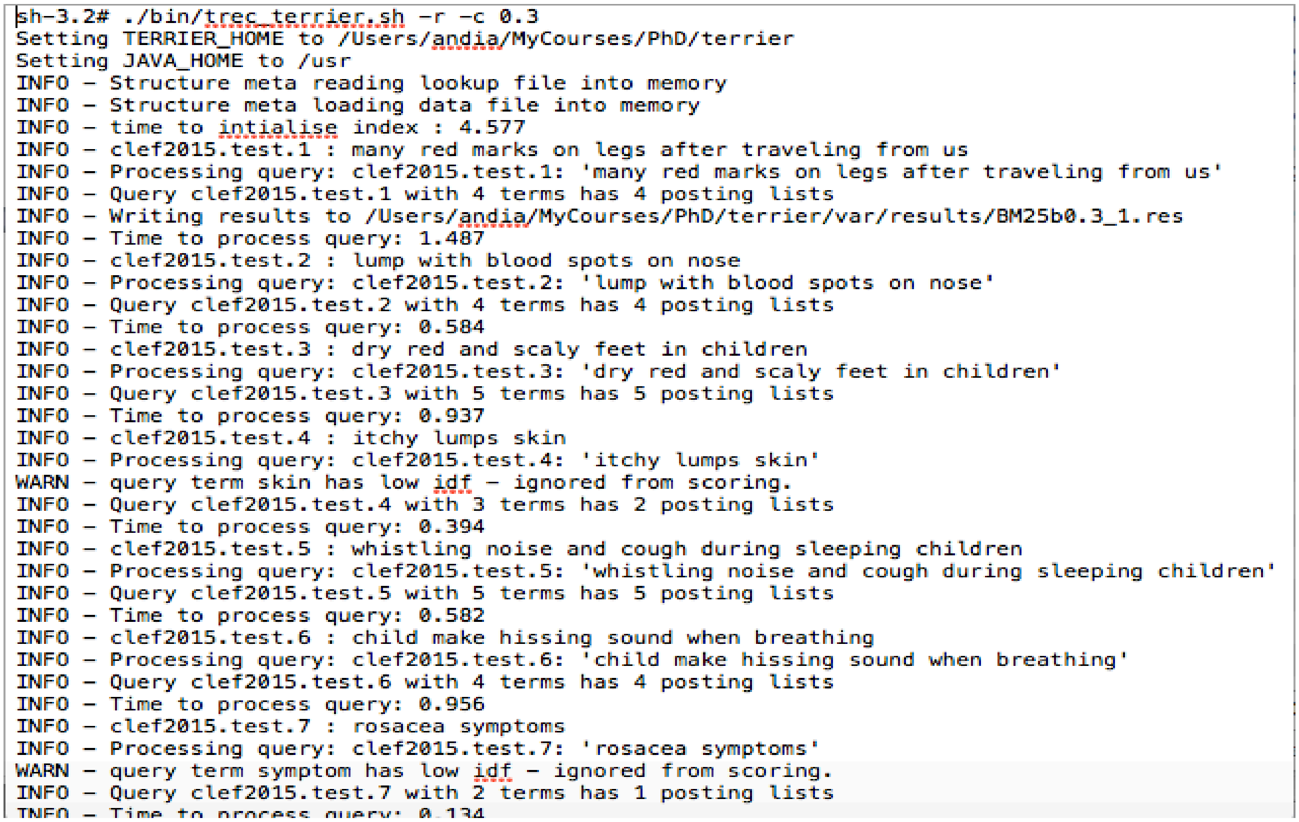
5.5 Optimal Feedback:

This following formula maximizes the likeness to the relevant documents and minimizes the likeness to the non-relevant documents:

$$\overrightarrow{q_{opt}} = \frac{1}{C_r}\sum_{\vec{d}\in C_r}\vec{d} - \frac{1}{N-C_r}\sum_{\vec{d}\notin C_r}\vec{d}$$ *[10]*

N is the total number of documents.

d is the document and q_{opt} is the optimal query.



```
sh-3.2# ./bin/trec_terrier.sh -r -c 0.3
Setting TERRIER_HOME to /Users/andia/MyCourses/PhD/terrier
Setting JAVA_HOME to /usr
INFO - Structure meta reading lookup file into memory
INFO - Structure meta loading data file into memory
INFO - time to intialise index : 4.577
INFO - clef2015.test.1 : many red marks on legs after traveling from us
INFO - Processing query: clef2015.test.1: 'many red marks on legs after traveling from us'
INFO - Query clef2015.test.1 with 4 terms has 4 posting lists
INFO - Writing results to /Users/andia/MyCourses/PhD/terrier/var/results/BM25b0.3_1.res
INFO - Time to process query: 1.487
INFO - clef2015.test.2 : lump with blood spots on nose
INFO - Processing query: clef2015.test.2: 'lump with blood spots on nose'
INFO - Query clef2015.test.2 with 4 terms has 4 posting lists
INFO - Time to process query: 0.584
INFO - clef2015.test.3 : dry red and scaly feet in children
INFO - Processing query: clef2015.test.3: 'dry red and scaly feet in children'
INFO - Query clef2015.test.3 with 5 terms has 5 posting lists
INFO - Time to process query: 0.937
INFO - clef2015.test.4 : itchy lumps skin
INFO - Processing query: clef2015.test.4: 'itchy lumps skin'
WARN - query term skin has low idf - ignored from scoring.
INFO - Query clef2015.test.4 with 3 terms has 2 posting lists
INFO - Time to process query: 0.394
INFO - clef2015.test.5 : whistling noise and cough during sleeping children
INFO - Processing query: clef2015.test.5: 'whistling noise and cough during sleeping children'
INFO - Query clef2015.test.5 with 5 terms has 5 posting lists
INFO - Time to process query: 0.582
INFO - clef2015.test.6 : child make hissing sound when breathing
INFO - Processing query: clef2015.test.6: 'child make hissing sound when breathing'
INFO - Query clef2015.test.6 with 4 terms has 4 posting lists
INFO - Time to process query: 0.956
INFO - clef2015.test.7 : rosacea symptoms
INFO - Processing query: clef2015.test.7: 'rosacea symptoms'
WARN - query term symptom has low idf - ignored from scoring.
INFO - Query clef2015.test.7 with 2 terms has 1 posting lists
INFO - Time to process query: 0.134
```

6. Ranking Methods for Textual Documents:

6.1 Exact Matching (Boolean):

This technique is good for ranking the textual documents and Boolean query retrieval consisting of one or more words. Boolean queries are considered to be accurate and can give more control to the users. Each query term identifies a set of documents which contains of the terms; Query terms and documents are sets of words and use processes such as “or (∨)”, “and (∧)”, and “not (¬)”. The outcomes are either True/False or exact-match. Boolean model is easy to understand, explain and implement all its terms and features are imported equally and the non-relevant documents can be eliminated from the search [47].

In Boolean retrieval, users are able to use large text queries by just typing one or more words and then the system can determine which document is more relevant to the query. IR system implements extended Boolean retrieval by adding additional operator that is called term proximity. Term proximity determines that two terms in a query occur adjacent to each other in the document this is measured by the number of dominant words.

Boolean model is precise which provides the user with more control over the retrieval results and also more effective ranking strategies [47].

Boolean query processing works as below:

1. Locate Brutus in the dictionary

2. Retrieve the position

3. Locate Calpurnia in the dictionary

4. Retrieve the position

5. Intersect the two positions

In the Boolean Model a typical strategy is to use a conjunctive reading of the different aspects in a query and disjunctive reading aspects of terms, for example in a query of [a $\wedge$ b $\vee$ c] should be written as: [(a $\wedge$ b) $\vee$ c] or [a $\wedge$ (b $\vee$ c)].

Algorithm for Intersection:

```
$INTERSET(P_1, P_2)$
answer $\leftarrow \langle \ \rangle$
While $p_1 \neq NIL$ and $p_2 \neq NIL$
Do if $docID(p_1) = docID(p_2)$
Then $ADD(answer, docID(p_1))$
$p_1 \leftarrow next(p_1)$
$p_2 \leftarrow next(p_2)$
Else if $docID(p_1) < docID(p_2)$
Then $p_1 \leftarrow next(p_1)$
Else $p_2 \leftarrow next(p_2)$
Return answer
```

[47] Christopher D. Manning, Prabhakar Raghavan, Hinrich Schütze – An Introduction to Information Retrieval, 2009.

6.2 Vector Space Model:

A Vector Space Model is a vector of the term weights and is an algebraic model characterizes the text documents; then the distance between the query points and the documents are ranked according to their possible relevance. Vector model represents index terms; terms are usually single words or keywords. Vectors are used to compare documents with queries, based on their similarities [15]. Documents are symbolized as vectors. This model compares the documents with the queries and retrieves and ranks these documents based on the particular query. Ranking the similarity between the documents can be calculated by comparing the cosine of the angles of the documents and the original query:

$$\cos\theta = \frac{d_2 \cdot q}{\|d_2\| \|q\|}$$

[30]

“ $\mathbf{d_2} \cdot \mathbf{q}$ ” is the intersection of the document. Documents are ranked by decreasing the value of the cosine, and the higher weight has more impact on cosine.

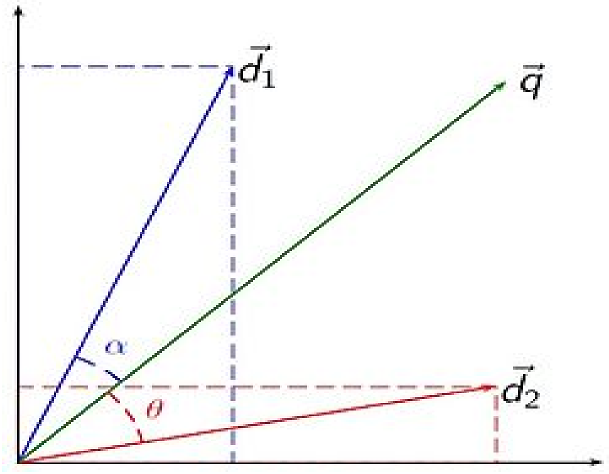


Figure 6.2.1 : The Similarity between the Documents

[30] G. Salton, A. Wong, and C. S. Yang (1975), "A Vector Space Model for Automatic Indexing,"Communications of the ACM, vol. 18, nr. 11, pages 613–620.

If the cosine equals to zero it means that the documents have no similarity with the query term. Vector space calculates a continuous degree of similarity between the document and the query and rank the documents based on their relevance, it also calculates the partial matching between the documents and query. The similarity between two documents can be calculated using a function of the angle between their vectors in the term vector space depending on the weights of the terms. However it is not suitable for the documents with long and different vocabulary. Vector Space Model is based on the assumption that meaning of the documents can be the result of the document's basic term. It can also help the users to search through the documents that are more similar or have better use than other documents. This model puts terms and a query in a document term space and calculates the similarities between the terms and the queries that improve the ranking measurements of the results based on the relevance and it calculates the weight of the ranked retrieval results based on their importance.

6.3 Probabilistic Model:

Maron and Kuns first proposed this model of indexing in 1960s and then in 1976 it has been expanded by Robertson and Sparck Jones. This model is based on the assumption that the relevant documents appeared in the first retrieval process have more weight than if they don't appear in the relevant documents. In other words if we have some relevant and non-relevant documents we can easily estimate the probability of appearing a relevant term in the document. Probabilistic retrieval model ranks the documents in reducing order of the probability of relevance. It uses probability theory to model the uncertainty in the retrieval process. The evaluation of the relevance probability is based on the occurrence of the term in the query [44]. The probability of the term is calculated as follows:

$$P(t|\bar{R}) = log\,\frac{N-n+0.5}{n+0.5}$$ *[44]*

R is complete unknown.

And the probability of the term t is calculated from the Log of (number of documents N minus n+1/2 divided by n+1/2).

6.4 Language Model:

This model is a statistical probability distribution over a sequence of documents in a collection; these documents are ranked based on the probability of the word that most likely appears in a relevant document and use those words in the query. The idea behind language model is that each document is considered as a language sample and then evaluates the probability of producing individual terms in the documents. Language Model consists of Unigram, Bigram and N-gram models. Unigram model is based on the assumption that the word occurs independently of each other and the results are obtained based on the probability of the individual word. However in the Bigram model the probability of the new word is based on the probability of the occurrence of the previous word and therefor for the N-gram the probability of the new word is based on the probability of the occurrence on the N previous words [45].

Unigram: $P(W_{1},n) = P(W_1)P(W_2)\dots P(W_n)$

Bigram: $P(W_{1},n) = P(W_1)P(W_2|W_1)\dots P(W_n|W_{n-1})$

Trigram: $P(W_{1},n) = P(W_1)P(W_2|W_1)P\left(W_3|W_{1,2}\right)\dots P(W_n|W_{n-2,n-1})$

[45] Fei Song, W.Bruce Croft, A General Language Model for Information Retrieval.

6.5 Set-Oriented Ranking Model:

This model is based on the assumption that query is ranked against the hieratical set of related documents. The desirable results are the combination of relevance and diversity of top-N tasks. The way this model works is to mix the diversity model with the traditional model and build a set-oriented model. By using this model we can obtain the top-N results directly, which can be both relevant and diversified. Experiments show that this model works better than the traditional models in terms of diversity and also has better performance in finding relevant documents [41].

In the set retrievals however system divides the amount of data into two subsets of documents based on the ones that are relevant to the search query and the ones that are not, but ranked retrievals ranks the documents based on their relevancy.

Information Retrieval system combines the set retrievals and ranked retrievals in a way that first define all the matching documents and then rank them based on their relevancy; this idea is used by most of the search engines such as Google.

7. Method:

The First task of this thesis research was to convert the format of the datasets provided by CLEF, which were not readable by the Terrier platform. I used Eclipse as my preferred software in order to write a Java program that can change the format of the CLEF datasets. These datasets consist of a collection of a million documents provided through the Khoresmoi project and contain information and web pages that cover a wide range of medical and health care related topics [8]. The only issue with changing the format of the datasets was that it created too many collections of data for indexing and querying by Terrier that I was able to solve. After the first part of my task was successfully completed I was able to use Terrier to index the queries. Index is used to store and map the contents of the data structure into a preferred location in the database. Each index query represents a set of documents by weighted keywords. These indexes were later used to compare and calculate the scores between the queries [8]. My first approach was to use medical ontology to find an easy and reliable method for patients and help them to have a better understanding of their diseases, also help doctors to discover better possible methods of treatments. Terrier was my preferred platform for indexing and querying the collected documents, I used BM25 as my baseline to score probabilistic model retrieval and the scaling strategy I used was to range my results from 0 to1, and also retrieved the optimal result. I then used Pseudo relevance feedback weighting model to improve the retrieval results; I then obtained the results that are originally returned from the query and determined if the information was relevant or non-relevant based on the initial query [13]. The relevant documents were clustered together. I was able to improve the performance by using PRF as my re-ranking strategy. I was also able to reformulate the dataset given by CLEF and created queries using Unified Medical Language System which acts as a dictionary for the medical ontology and indexed these new queries and submitted up to 10 runs in TREC style. The runs included the top1000 documents returned for each set. I obtained results from the top documents and returned the results with the relevance assessments to the original query and expand these queries to find the related terms, and then I compared each of the runs against the query topic.

Relevance Feedback (RF) however adds the query terms and adjusts the weight of each query term of relevant and non-relevant documents and ranks the lists, so the relevant documents get higher ranks. The best query is the one that has the most similarity to the relevant document. Relevance feedback is used to improve the efficiency of Information Retrieval [14]. Relevance feedback created long revised queries and is sometimes expensive to process, that is why I used PRF and ranked my results and my idea was to use the medical ontology with my obtained results in order to get better performance and then combined the use of PRF with UMLS for retrieving better results and understanding the meaning of the biomedical terms.

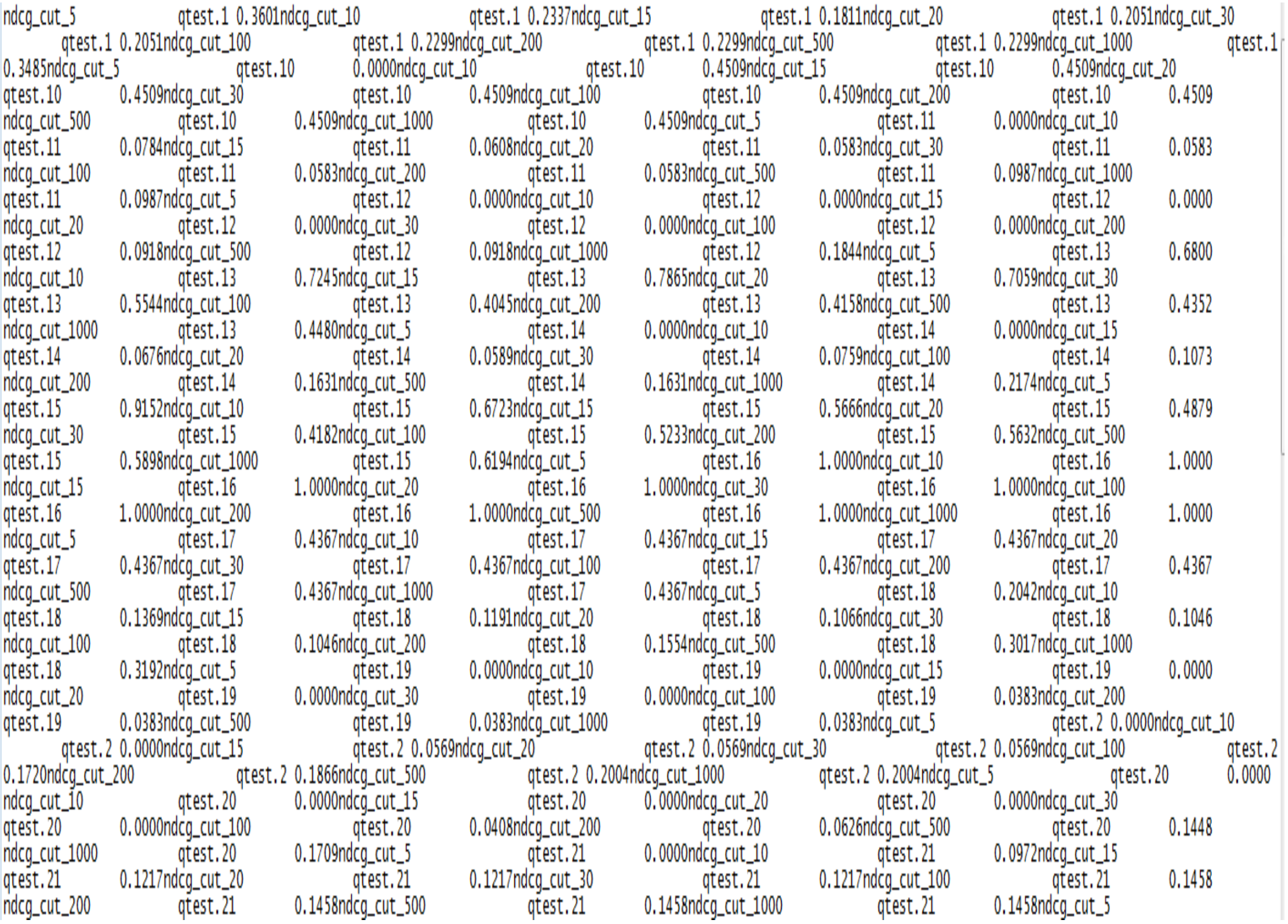
I have submitted my work to CLEF eHealth 2015 and my paper has been successfully approved for publication in CLEF 2015 work group; I have also been invited to attend the September 2015 in Toulouse- France to present my work.

8. Environmental Settings and Evaluation Metric:

Evaluation focus on P@5, P@10, NDCG@5, NDCG@10. Also other IR evaluation measuring models were used to evaluate the submitted runs [20]. Dataset in my research was provided through the Khoresmoi project and was a collection of health care records prepared and given by CLEF. I used Terrier platform for indexing and querying the collected documents, BM25 as my baseline scoring for probabilistic model retrieval ranging from 0 to1, PRF was used for finding and improving the relevancy in the documents, and UMLS was also used as dictionary to translate the medical terms. Evaluation was done by CLEF team using their golden standard method, the collection contained of 1000 documents and was around 8 GB.

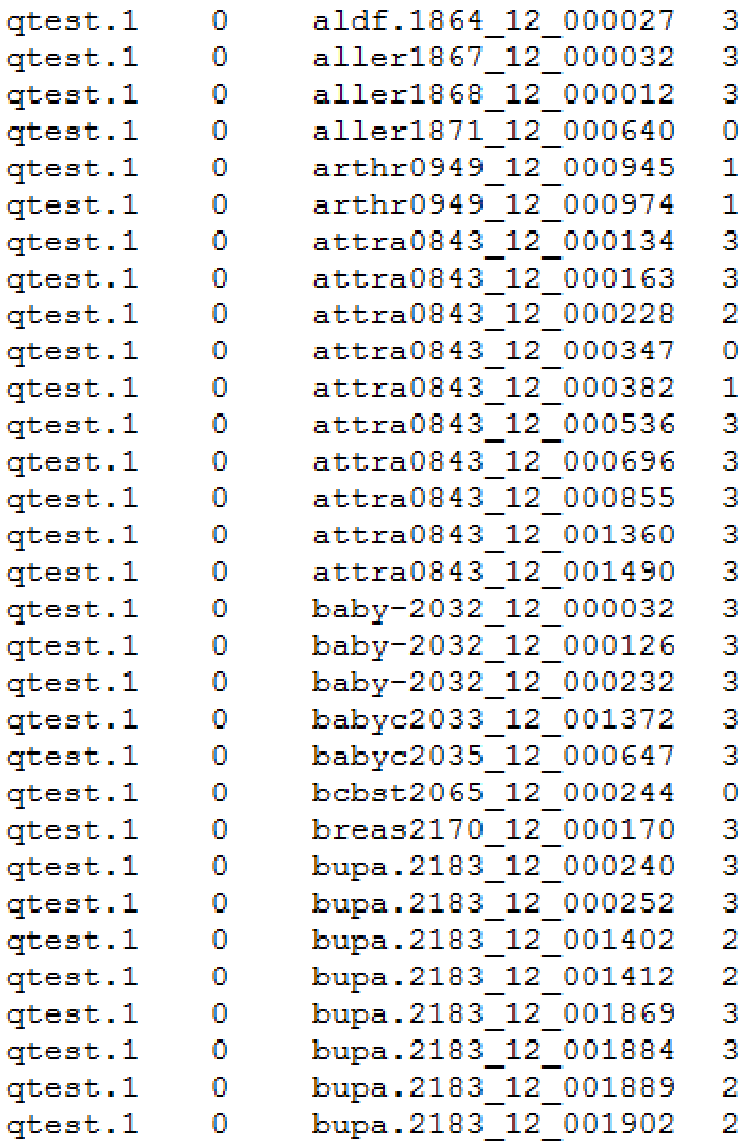
and the results are as follows.

Examples of the Evaluation:



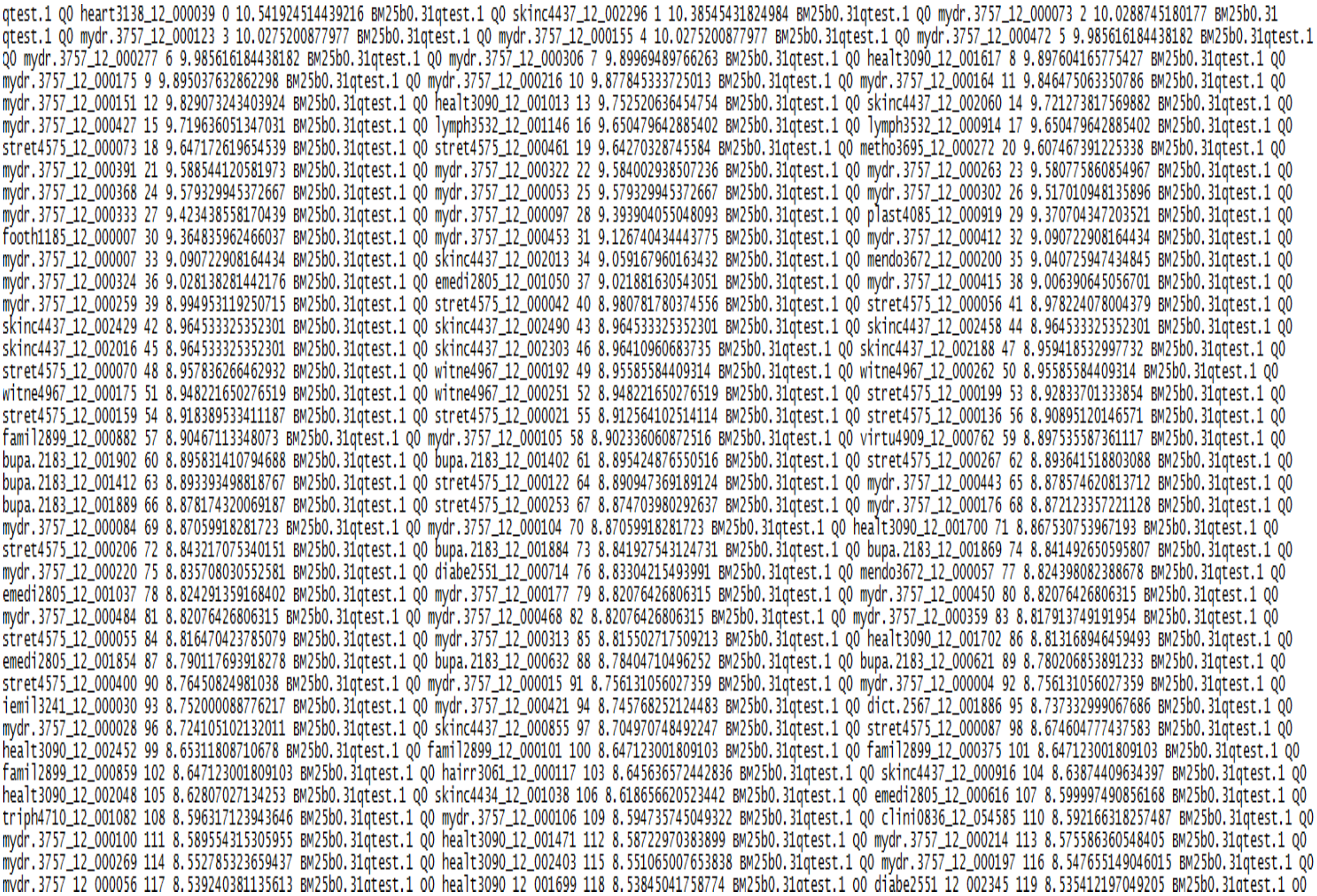
```
ndcg_cut_5              qtest.1 0.3601ndcg_cut_10               qtest.1 0.2337ndcg_cut_15               qtest.1 0.1811ndcg_cut_20               qtest.1 0.2051ndcg_cut_30
        qtest.1 0.2051ndcg_cut_100              qtest.1 0.2299ndcg_cut_200              qtest.1 0.2299ndcg_cut_500              qtest.1 0.2299ndcg_cut_1000             qtest.1
0.3485ndcg_cut_5                qtest.10        0.0000ndcg_cut_10               qtest.10        0.4509ndcg_cut_15               qtest.10        0.4509ndcg_cut_20
qtest.10        0.4509ndcg_cut_30               qtest.10        0.4509ndcg_cut_100              qtest.10        0.4509ndcg_cut_200              qtest.10        0.4509
ndcg_cut_500            qtest.10        0.4509ndcg_cut_1000             qtest.10        0.4509ndcg_cut_5                qtest.11        0.0000ndcg_cut_10
qtest.11        0.0784ndcg_cut_15               qtest.11        0.0608ndcg_cut_20               qtest.11        0.0583ndcg_cut_30               qtest.11        0.0583
ndcg_cut_100            qtest.11        0.0583ndcg_cut_200              qtest.11        0.0583ndcg_cut_500              qtest.11        0.0987ndcg_cut_1000
qtest.11        0.0987ndcg_cut_5                qtest.12        0.0000ndcg_cut_10               qtest.12        0.0000ndcg_cut_15               qtest.12        0.0000
ndcg_cut_20             qtest.12        0.0000ndcg_cut_30               qtest.12        0.0000ndcg_cut_100              qtest.12        0.0000ndcg_cut_200
qtest.12        0.0918ndcg_cut_500              qtest.12        0.0918ndcg_cut_1000             qtest.12        0.1844ndcg_cut_5                qtest.13        0.6800
ndcg_cut_10             qtest.13        0.7245ndcg_cut_15               qtest.13        0.7865ndcg_cut_20               qtest.13        0.7059ndcg_cut_30
qtest.13        0.5544ndcg_cut_100              qtest.13        0.4045ndcg_cut_200              qtest.13        0.4158ndcg_cut_500              qtest.13        0.4352
ndcg_cut_1000           qtest.13        0.4480ndcg_cut_5                qtest.14        0.0000ndcg_cut_10               qtest.14        0.0000ndcg_cut_15
qtest.14        0.0676ndcg_cut_20               qtest.14        0.0589ndcg_cut_30               qtest.14        0.0759ndcg_cut_100              qtest.14        0.1073
ndcg_cut_200            qtest.14        0.1631ndcg_cut_500              qtest.14        0.1631ndcg_cut_1000             qtest.14        0.2174ndcg_cut_5
qtest.15        0.9152ndcg_cut_10               qtest.15        0.6723ndcg_cut_15               qtest.15        0.5666ndcg_cut_20               qtest.15        0.4879
ndcg_cut_30             qtest.15        0.4182ndcg_cut_100              qtest.15        0.5233ndcg_cut_200              qtest.15        0.5632ndcg_cut_500
qtest.15        0.5898ndcg_cut_1000             qtest.15        0.6194ndcg_cut_5                qtest.16        1.0000ndcg_cut_10               qtest.16        1.0000
ndcg_cut_15             qtest.16        1.0000ndcg_cut_20               qtest.16        1.0000ndcg_cut_30               qtest.16        1.0000ndcg_cut_100
qtest.16        1.0000ndcg_cut_200              qtest.16        1.0000ndcg_cut_500              qtest.16        1.0000ndcg_cut_1000             qtest.16        1.0000
ndcg_cut_5              qtest.17        0.4367ndcg_cut_10               qtest.17        0.4367ndcg_cut_15               qtest.17        0.4367ndcg_cut_20
qtest.17        0.4367ndcg_cut_30               qtest.17        0.4367ndcg_cut_100              qtest.17        0.4367ndcg_cut_200              qtest.17        0.4367
ndcg_cut_500            qtest.17        0.4367ndcg_cut_1000             qtest.17        0.4367ndcg_cut_5                qtest.18        0.2042ndcg_cut_10
qtest.18        0.1369ndcg_cut_15               qtest.18        0.1191ndcg_cut_20               qtest.18        0.1066ndcg_cut_30               qtest.18        0.1046
ndcg_cut_100            qtest.18        0.1046ndcg_cut_200              qtest.18        0.1554ndcg_cut_500              qtest.18        0.3017ndcg_cut_1000
qtest.18        0.3192ndcg_cut_5                qtest.19        0.0000ndcg_cut_10               qtest.19        0.0000ndcg_cut_15               qtest.19        0.0000
ndcg_cut_20             qtest.19        0.0000ndcg_cut_30               qtest.19        0.0000ndcg_cut_100              qtest.19        0.0383ndcg_cut_200
qtest.19        0.0383ndcg_cut_500              qtest.19        0.0383ndcg_cut_1000             qtest.19        0.0383ndcg_cut_5                qtest.2 0.0000ndcg_cut_10
        qtest.2 0.0000ndcg_cut_15               qtest.2 0.0569ndcg_cut_20               qtest.2 0.0569ndcg_cut_30               qtest.2 0.0569ndcg_cut_100              qtest.2
0.1720ndcg_cut_200              qtest.2 0.1866ndcg_cut_500              qtest.2 0.2004ndcg_cut_1000             qtest.2 0.2004ndcg_cut_5                qtest.20        0.0000
ndcg_cut_10             qtest.20        0.0000ndcg_cut_15               qtest.20        0.0000ndcg_cut_20               qtest.20        0.0000ndcg_cut_30
qtest.20        0.0000ndcg_cut_100              qtest.20        0.0408ndcg_cut_200              qtest.20        0.0626ndcg_cut_500              qtest.20        0.1448
ndcg_cut_1000           qtest.20        0.1709ndcg_cut_5                qtest.21        0.0000ndcg_cut_10               qtest.21        0.0972ndcg_cut_15
qtest.21        0.1217ndcg_cut_20               qtest.21        0.1217ndcg_cut_30               qtest.21        0.1217ndcg_cut_100              qtest.21        0.1458
ndcg_cut_200            qtest.21        0.1458ndcg_cut_500              qtest.21        0.1458ndcg_cut_1000             qtest.21        0.1458ndcg_cut_5
```

Examples of the Query:



```
qtest.1    0    aldf.1864_12_000027   3
qtest.1    0    aller1867_12_000032   3
qtest.1    0    aller1868_12_000012   3
qtest.1    0    aller1871_12_000640   0
qtest.1    0    arthr0949_12_000945   1
qtest.1    0    arthr0949_12_000974   1
qtest.1    0    attra0843_12_000134   3
qtest.1    0    attra0843_12_000163   3
qtest.1    0    attra0843_12_000228   2
qtest.1    0    attra0843_12_000347   0
qtest.1    0    attra0843_12_000382   1
qtest.1    0    attra0843_12_000536   3
qtest.1    0    attra0843_12_000696   3
qtest.1    0    attra0843_12_000855   3
qtest.1    0    attra0843_12_001360   3
qtest.1    0    attra0843_12_001490   3
qtest.1    0    baby-2032_12_000032   3
qtest.1    0    baby-2032_12_000126   3
qtest.1    0    baby-2032_12_000232   3
qtest.1    0    babyc2033_12_001372   3
qtest.1    0    babyc2035_12_000647   3
qtest.1    0    bcbst2065_12_000244   0
qtest.1    0    breas2170_12_000170   3
qtest.1    0    bupa.2183_12_000240   3
qtest.1    0    bupa.2183_12_000252   3
qtest.1    0    bupa.2183_12_001402   2
qtest.1    0    bupa.2183_12_001412   2
qtest.1    0    bupa.2183_12_001869   3
qtest.1    0    bupa.2183_12_001884   3
qtest.1    0    bupa.2183_12_001889   2
qtest.1    0    bupa.2183_12_001902   2
```

Examples of the Run:



```
qtest.1 Q0 heart3138_12_000039 0 10.541924514439216 BM25b0.31qtest.1 Q0 skinc4437_12_002296 1 10.38545431824984 BM25b0.31qtest.1 Q0 mydr.3757_12_000073 2 10.0288745180177 BM25b0.31
qtest.1 Q0 mydr.3757_12_000123 3 10.0275200877977 BM25b0.31qtest.1 Q0 mydr.3757_12_000155 4 10.0275200877977 BM25b0.31qtest.1 Q0 mydr.3757_12_000472 5 9.985616184438182 BM25b0.31qtest.1
Q0 mydr.3757_12_000277 6 9.985616184438182 BM25b0.31qtest.1 Q0 mydr.3757_12_000306 7 9.89969489766263 BM25b0.31qtest.1 Q0 healt3090_12_001617 8 9.897604165775427 BM25b0.31qtest.1 Q0
mydr.3757_12_000175 9 9.895037632862298 BM25b0.31qtest.1 Q0 mydr.3757_12_000216 10 9.877845333725013 BM25b0.31qtest.1 Q0 mydr.3757_12_000164 11 9.846475063350786 BM25b0.31qtest.1 Q0
mydr.3757_12_000151 12 9.829073243403924 BM25b0.31qtest.1 Q0 healt3090_12_001013 13 9.752520636454754 BM25b0.31qtest.1 Q0 skinc4437_12_002060 14 9.721273817569882 BM25b0.31qtest.1 Q0
mydr.3757_12_000427 15 9.719636051347031 BM25b0.31qtest.1 Q0 lymph3532_12_001146 16 9.650479642885402 BM25b0.31qtest.1 Q0 lymph3532_12_000914 17 9.650479642885402 BM25b0.31qtest.1 Q0
stret4575_12_000073 18 9.647172619654539 BM25b0.31qtest.1 Q0 stret4575_12_000461 19 9.64270328745584 BM25b0.31qtest.1 Q0 metho3695_12_000272 20 9.607467391225338 BM25b0.31qtest.1 Q0
mydr.3757_12_000391 21 9.588544120581973 BM25b0.31qtest.1 Q0 mydr.3757_12_000322 22 9.584002938507236 BM25b0.31qtest.1 Q0 mydr.3757_12_000263 23 9.580775860854967 BM25b0.31qtest.1 Q0
mydr.3757_12_000368 24 9.579329945372667 BM25b0.31qtest.1 Q0 mydr.3757_12_000053 25 9.579329945372667 BM25b0.31qtest.1 Q0 mydr.3757_12_000302 26 9.517010948135896 BM25b0.31qtest.1 Q0
mydr.3757_12_000333 27 9.423438558170439 BM25b0.31qtest.1 Q0 mydr.3757_12_000097 28 9.393904055048093 BM25b0.31qtest.1 Q0 plast4085_12_000919 29 9.370704347203521 BM25b0.31qtest.1 Q0
footh1185_12_000007 30 9.364835962466037 BM25b0.31qtest.1 Q0 mydr.3757_12_000453 31 9.126740434443775 BM25b0.31qtest.1 Q0 mydr.3757_12_000412 32 9.090722908164434 BM25b0.31qtest.1 Q0
mydr.3757_12_000007 33 9.090722908164434 BM25b0.31qtest.1 Q0 skinc4437_12_002013 34 9.059167960163432 BM25b0.31qtest.1 Q0 mendo3672_12_000200 35 9.040725947434845 BM25b0.31qtest.1 Q0
mydr.3757_12_000324 36 9.028138281442176 BM25b0.31qtest.1 Q0 emedi2805_12_001050 37 9.021881630543051 BM25b0.31qtest.1 Q0 mydr.3757_12_000415 38 9.006390645056701 BM25b0.31qtest.1 Q0
mydr.3757_12_000259 39 8.994953119250715 BM25b0.31qtest.1 Q0 stret4575_12_000042 40 8.980781780374556 BM25b0.31qtest.1 Q0 stret4575_12_000056 41 8.978224078004379 BM25b0.31qtest.1 Q0
skinc4437_12_002429 42 8.964533325352301 BM25b0.31qtest.1 Q0 skinc4437_12_002490 43 8.964533325352301 BM25b0.31qtest.1 Q0 skinc4437_12_002458 44 8.964533325352301 BM25b0.31qtest.1 Q0
skinc4437_12_002016 45 8.964533325352301 BM25b0.31qtest.1 Q0 skinc4437_12_002303 46 8.96410960683735 BM25b0.31qtest.1 Q0 skinc4437_12_002188 47 8.959418532997732 BM25b0.31qtest.1 Q0
stret4575_12_000070 48 8.957836266462932 BM25b0.31qtest.1 Q0 witne4967_12_000192 49 8.9558558440931 4 BM25b0.31qtest.1 Q0 witne4967_12_000262 50 8.9558558440931 4 BM25b0.31qtest.1 Q0
witne4967_12_000175 51 8.948221650276519 BM25b0.31qtest.1 Q0 witne4967_12_000251 52 8.948221650276519 BM25b0.31qtest.1 Q0 stret4575_12_000199 53 8.92833701333854 BM25b0.31qtest.1 Q0
stret4575_12_000159 54 8.918389533411187 BM25b0.31qtest.1 Q0 stret4575_12_000021 55 8.912564102514114 BM25b0.31qtest.1 Q0 stret4575_12_000136 56 8.90895120146571 BM25b0.31qtest.1 Q0
famil2899_12_000882 57 8.90467113348073 BM25b0.31qtest.1 Q0 mydr.3757_12_000105 58 8.902336060872516 BM25b0.31qtest.1 Q0 virtu4909_12_000762 59 8.897535587361117 BM25b0.31qtest.1 Q0
bupa.2183_12_001902 60 8.895831410794688 BM25b0.31qtest.1 Q0 bupa.2183_12_001402 61 8.895424876550516 BM25b0.31qtest.1 Q0 stret4575_12_000267 62 8.893641518803088 BM25b0.31qtest.1 Q0
bupa.2183_12_001412 63 8.893393498818767 BM25b0.31qtest.1 Q0 stret4575_12_000122 64 8.890947369189124 BM25b0.31qtest.1 Q0 mydr.3757_12_000443 65 8.878574620813712 BM25b0.31qtest.1 Q0
bupa.2183_12_001889 66 8.878174320069187 BM25b0.31qtest.1 Q0 stret4575_12_000253 67 8.874703980292637 BM25b0.31qtest.1 Q0 mydr.3757_12_000176 68 8.872123357221128 BM25b0.31qtest.1 Q0
mydr.3757_12_000084 69 8.87059918281723 BM25b0.31qtest.1 Q0 mydr.3757_12_000104 70 8.87059918281723 BM25b0.31qtest.1 Q0 healt3090_12_001700 71 8.867530753967193 BM25b0.31qtest.1 Q0
stret4575_12_000206 72 8.843217075340151 BM25b0.31qtest.1 Q0 bupa.2183_12_001884 73 8.841927543124731 BM25b0.31qtest.1 Q0 bupa.2183_12_001869 74 8.841492650595807 BM25b0.31qtest.1 Q0
mydr.3757_12_000220 75 8.835708030552581 BM25b0.31qtest.1 Q0 diabe2551_12_000714 76 8.83304215493991 BM25b0.31qtest.1 Q0 mendo3672_12_000057 77 8.824398082388678 BM25b0.31qtest.1 Q0
emedi2805_12_001037 78 8.824291359168402 BM25b0.31qtest.1 Q0 mydr.3757_12_000177 79 8.82076426806315 BM25b0.31qtest.1 Q0 mydr.3757_12_000450 80 8.82076426806315 BM25b0.31qtest.1 Q0
mydr.3757_12_000484 81 8.82076426806315 BM25b0.31qtest.1 Q0 mydr.3757_12_000468 82 8.82076426806315 BM25b0.31qtest.1 Q0 mydr.3757_12_000359 83 8.817913749191954 BM25b0.31qtest.1 Q0
stret4575_12_000055 84 8.816470423785079 BM25b0.31qtest.1 Q0 mydr.3757_12_000313 85 8.815502717509213 BM25b0.31qtest.1 Q0 healt3090_12_001702 86 8.813168946459493 BM25b0.31qtest.1 Q0
emedi2805_12_001854 87 8.790117693918278 BM25b0.31qtest.1 Q0 bupa.2183_12_000632 88 8.78404710496252 BM25b0.31qtest.1 Q0 bupa.2183_12_000621 89 8.780206853891233 BM25b0.31qtest.1 Q0
stret4575_12_000400 90 8.76450824981038 BM25b0.31qtest.1 Q0 mydr.3757_12_000015 91 8.756131056027359 BM25b0.31qtest.1 Q0 mydr.3757_12_000004 92 8.756131056027359 BM25b0.31qtest.1 Q0
iemil3241_12_000030 93 8.752000088776217 BM25b0.31qtest.1 Q0 mydr.3757_12_000421 94 8.745768252124483 BM25b0.31qtest.1 Q0 dict.2567_12_001886 95 8.737332999067686 BM25b0.31qtest.1 Q0
mydr.3757_12_000028 96 8.724105102132011 BM25b0.31qtest.1 Q0 skinc4437_12_000855 97 8.704970748492247 BM25b0.31qtest.1 Q0 stret4575_12_000087 98 8.674604777437583 BM25b0.31qtest.1 Q0
healt3090_12_002452 99 8.65311808710678 BM25b0.31qtest.1 Q0 famil2899_12_000101 100 8.647123001809103 BM25b0.31qtest.1 Q0 famil2899_12_000375 101 8.647123001809103 BM25b0.31qtest.1 Q0
famil2899_12_000859 102 8.647123001809103 BM25b0.31qtest.1 Q0 hairr3061_12_000117 103 8.645636572442836 BM25b0.31qtest.1 Q0 skinc4437_12_000916 104 8.63874409634397 BM25b0.31qtest.1 Q0
healt3090_12_002048 105 8.62807027134253 BM25b0.31qtest.1 Q0 skinc4434_12_001038 106 8.618656620523442 BM25b0.31qtest.1 Q0 emedi2805_12_000616 107 8.599997490856168 BM25b0.31qtest.1 Q0
triph4710_12_001082 108 8.596317123943646 BM25b0.31qtest.1 Q0 mydr.3757_12_000106 109 8.594735745049322 BM25b0.31qtest.1 Q0 clini0836_12_054585 110 8.592166318257487 BM25b0.31qtest.1 Q0
mydr.3757_12_000100 111 8.589554315305955 BM25b0.31qtest.1 Q0 healt3090_12_001471 112 8.58722970383899 BM25b0.31qtest.1 Q0 mydr.3757_12_000214 113 8.575586360548405 BM25b0.31qtest.1 Q0
mydr.3757_12_000269 114 8.552785323659437 BM25b0.31qtest.1 Q0 healt3090_12_002403 115 8.551065007653838 BM25b0.31qtest.1 Q0 mydr.3757_12_000197 116 8.547655149046015 BM25b0.31qtest.1 Q0
mydr.3757_12_000056 117 8.539240381135613 BM25b0.31qtest.1 Q0 healt3090_12_001699 118 8.53845041758774 BM25b0.31qtest.1 Q0 diabe2551_12_002345 119 8.535412197049205 BM25b0.31qtest.1 Q0
```

BootStrap:

```
/*!
 * Bootstrap v2.3.2
 *
 * Copyright 2012 Twitter, Inc
 * Licensed under the Apache License v2.0
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Designed and built with all the love in the world @twitter by @mdo and @fat.
 */

.clearfix {
  *zoom: 1;
}

.clearfix:before,
.clearfix:after {
  display: table;
  line-height: 0;
  content: "";
}
```

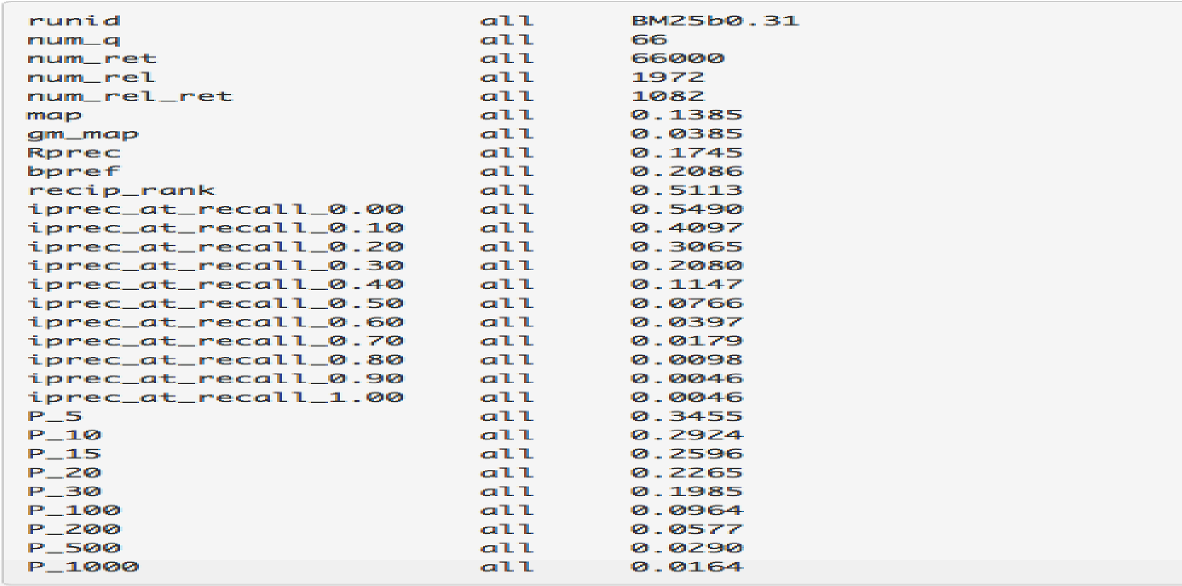
9. Preliminary Results and Comparison Chart:

This Year CLEFeHealth 2015 built result pools from the submissions. According to CLEF evaluation standard my Run2 and run3 had the highest priority among my 10 runs. The primary measurement used was P@5 and the secondary measurement used was normalized cumulative gain at rank 10 [20].

9.1 Evaluation with standard TREC_eval metric for Run2 and Run3:

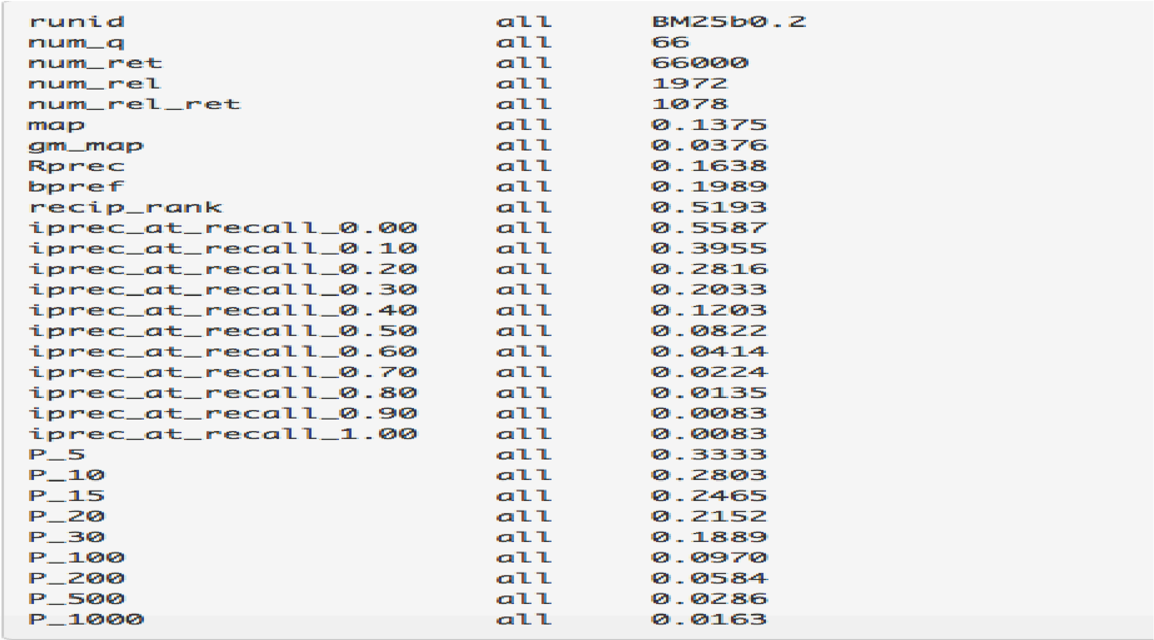
./trec_eval -c -M1000 qrels.clef2015.test.bin.txt runName

YorkU_EN_Run.2.dat



```
runid                    all     BM25b0.31
num_q                    all     66
num_ret                  all     66000
num_rel                  all     1972
num_rel_ret              all     1082
map                      all     0.1385
gm_map                   all     0.0385
Rprec                    all     0.1745
bpref                    all     0.2086
recip_rank               all     0.5113
iprec_at_recall_0.00     all     0.5490
iprec_at_recall_0.10     all     0.4097
iprec_at_recall_0.20     all     0.3065
iprec_at_recall_0.30     all     0.2080
iprec_at_recall_0.40     all     0.1147
iprec_at_recall_0.50     all     0.0766
iprec_at_recall_0.60     all     0.0397
iprec_at_recall_0.70     all     0.0179
iprec_at_recall_0.80     all     0.0098
iprec_at_recall_0.90     all     0.0046
iprec_at_recall_1.00     all     0.0046
P_5                      all     0.3455
P_10                     all     0.2924
P_15                     all     0.2596
P_20                     all     0.2265
P_30                     all     0.1985
P_100                    all     0.0964
P_200                    all     0.0577
P_500                    all     0.0290
P_1000                   all     0.0164
```

YorkU_EN_Run.3.dat



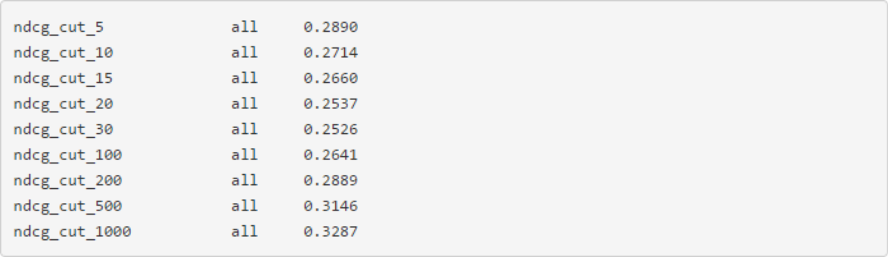
```
runid                    all     BM25b0.2
num_q                    all     66
num_ret                  all     66000
num_rel                  all     1972
num_rel_ret              all     1078
map                      all     0.1375
gm_map                   all     0.0376
Rprec                    all     0.1638
bpref                    all     0.1989
recip_rank               all     0.5193
iprec_at_recall_0.00     all     0.5587
iprec_at_recall_0.10     all     0.3955
iprec_at_recall_0.20     all     0.2816
iprec_at_recall_0.30     all     0.2033
iprec_at_recall_0.40     all     0.1203
iprec_at_recall_0.50     all     0.0822
iprec_at_recall_0.60     all     0.0414
iprec_at_recall_0.70     all     0.0224
iprec_at_recall_0.80     all     0.0135
iprec_at_recall_0.90     all     0.0083
iprec_at_recall_1.00     all     0.0083
P_5                      all     0.3333
P_10                     all     0.2803
P_15                     all     0.2465
P_20                     all     0.2152
P_30                     all     0.1889
P_100                    all     0.0970
P_200                    all     0.0584
P_500                    all     0.0286
P_1000                   all     0.0163
```

<u>*See Appendix A for all the runs*</u>:

9.2 Evaluation with nDCG:

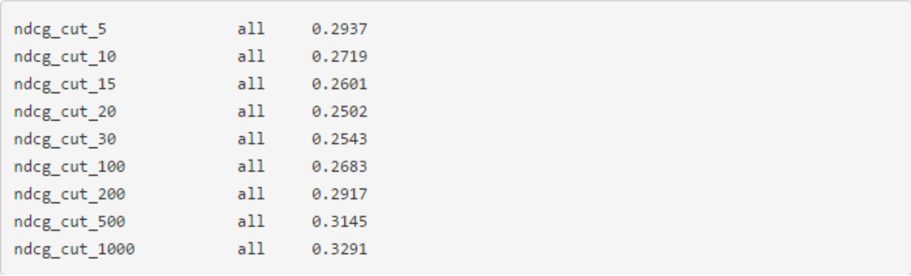
```
./trec_eval -c -M1000 -m ndcg_cut qrels.clef2015.test.graded.txt runName
```

YorkU_EN_Run.2.dat



```
ndcg_cut_5              all     0.2890
ndcg_cut_10             all     0.2714
ndcg_cut_15             all     0.2660
ndcg_cut_20             all     0.2537
ndcg_cut_30             all     0.2526
ndcg_cut_100            all     0.2641
ndcg_cut_200            all     0.2889
ndcg_cut_500            all     0.3146
ndcg_cut_1000           all     0.3287
```

YorkU_EN_Run.3.dat



```
ndcg_cut_5              all     0.2937
ndcg_cut_10             all     0.2719
ndcg_cut_15             all     0.2601
ndcg_cut_20             all     0.2502
ndcg_cut_30             all     0.2543
ndcg_cut_100            all     0.2683
ndcg_cut_200            all     0.2917
ndcg_cut_500            all     0.3145
ndcg_cut_1000           all     0.3291
```

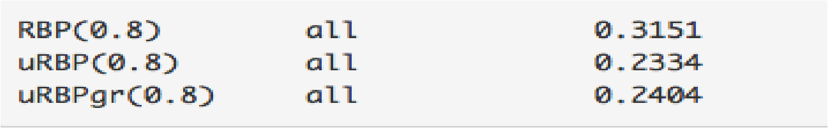
<u>*Please See Appendix B:*</u>

9.3 Reliability Biased-Evaluation:

This year's evaluation was based on understanding the information along with the relevance assessment and the reliability of the assessors that has been provided by the judges. These results have been obtained with the binary relevance assessment and graded reliability assessments. Documents with a readability score of 0 or 1 are believed not to be readable and documents with readability score of 2 or 3 are thought to be readable.

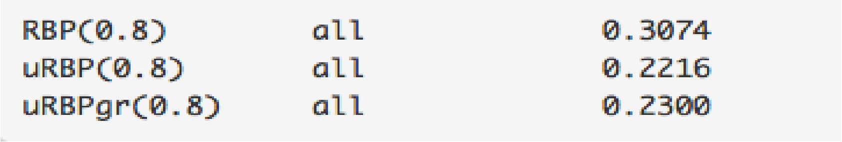
```
java -jar /tools/ubire.0.1.jar --qrels-file=qrels/qrels.clef2015.test.bin.txt --qread-file=qrels/qread.clef2015.test.graded.txt --readability --rbp-p=0.8 --ranking-file=runName
```

YorkU_EN_Run.2.dat



```
RBP(0.8)          all               0.3151
uRBP(0.8)         all               0.2334
uRBPgr(0.8)       all               0.2404
```

YorkU_EN_Run.3.dat



```
RBP(0.8)          all               0.3074
uRBP(0.8)         all               0.2216
uRBPgr(0.8)       all               0.2300
```

<u>*See Appendix C:*</u>

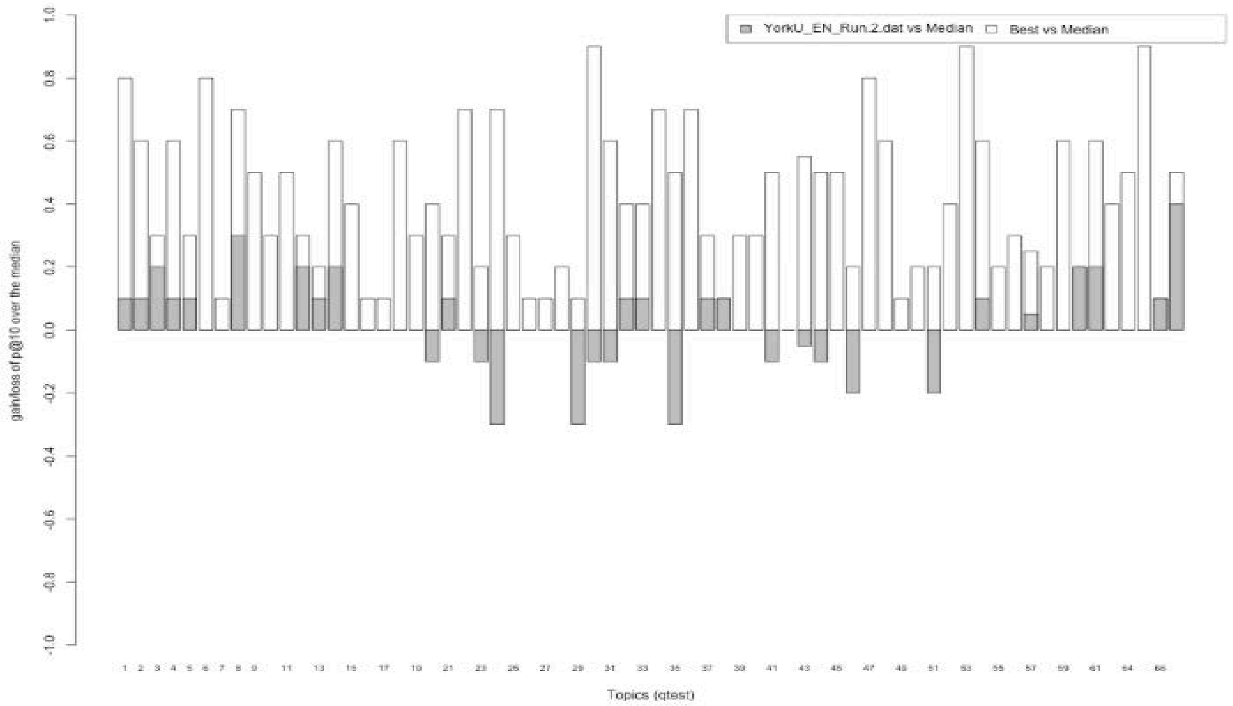
9.4 POLT P@10:

This plot compares each of the runs against median across each has been submitted to CLEF for each query topic where: [20]

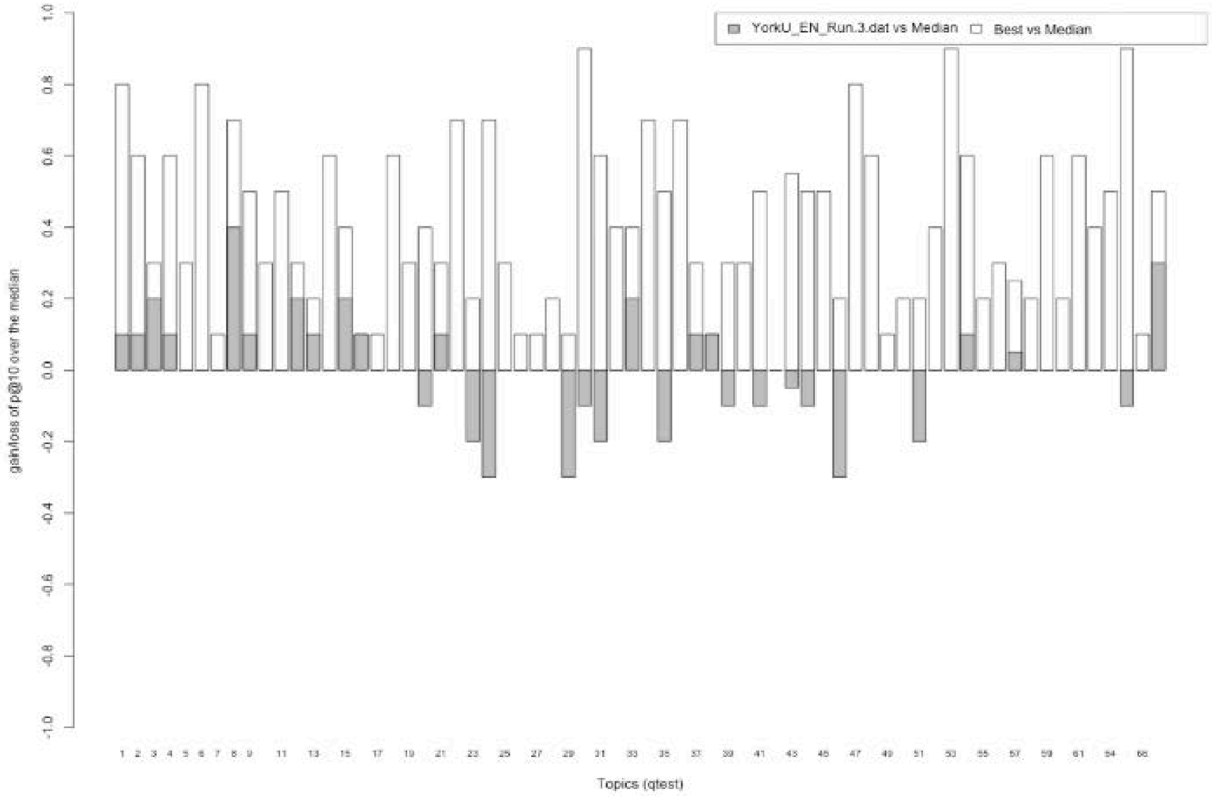
grey bars: height(q) = your_p@10(q) - median_p@10(q)

white bars: height(q) = best_p@10(q) - median_p@10(q)

YorkU_EN_Run.2.dat



YorkU_EN_Run.3.dat



Please see Appendix D:

10. Summary and Conclusion and Future Work:

In this research I first had to convert the format of the dataset using a java program in Eclipse, as the format of the data set provided by CLEF was not readable by Terrier platform, after the data was converted Terrier could index it, I used the base normalization method, which is BM25 to retrieve information from the dataset prepared by CLEF; Then I used Pseudo Relevance Feedback weighting model to estimate the better results for improving the overall performance of the Information Retrieval process, I was able to show that this normalization model ranks the documents based on their relevance in an effective and timely manner. The terms that are closer to the query term are more relevant to the topic. I used Unified Medical Language System (UMLS) to translate the medical query terms into the user friendly vocabularies to help the patients have a better understanding about their diseases and their health conditions and also help the doctors to deliver a better diagnosis and determine which treatment is suitable for their patients. The evaluation method was based on TREC standards represented by CLEF. The result pools were created based on the submissions and according to the CLEF evaluation standard, my Run2 and Run3 among all my 10 runs had the highest rate of relevancy. All the10 runs were also compared against the median through all the submissions to CLEF and then the chart for this comparison was plotted; the bars characterized the gain or loss of the system.

In future and towards my PhD research I would like to work on extracting information about Electronic Medical Records (EMR) and Electronic Health Records (HER) and look into how Information Retrieval can be an important and improving asset to find the best treatment and care and also reduce the medical errors to help patients have a better understanding about their health and the possible methods of treatments in a fast and accurate routine by the help of Medical Language Processing. I would also like to look more into the concept of Speech

Recognition applications that are being used in these days to collect the clinical and text data that is convenient and easy to use also improve the quality and reduce the possible errors. This method is a way to automatically translate words into texts in such a way that the doctor speaks directly using a microphone and the words enter and analyze into the system.

My goal is to use Information Extraction technique which is a sub-domain of Natural Language Processing in order to obtain the knowledge from the available medical data, I would like to first gather the relevant text data and documents and then extract the exact type of information. My focus is going to be on biomedical data extraction in EMR/EHR and record the health information along with the imaging, scans, laboratory results, diagnosis, and doctor’s notes; and also use UMLS tools as a dictionary to link the health information and medical terms and vocabularies between patients and doctors which help the patients to have a better understanding of their health situation.

Bibliography:

[1] Hsu, Hui-Huang. *Advanced Data Mining Technologies In Bioinformatics*. Hershey PA: Idea Group Pub., 2006. Print.

[2] Lesk, Arthur M. "The Unreasonable Effectiveness of Mathematics in Molecular Biology." *The Mathematical Intelligencer* 22.2 (2000): 28-37. Web.

[3] Kasperowicz, D. and Huang, X. J. "Semantic Matching Models for Medical Information Retrieval: A Case Study", Proceedings of the 2012 Advances in Health Informatics Conference (AHIC'12) (2012): 25-27. Web.

[4] Pecina, Pavel, Ondřej Dušek, Lorraine Goeuriot, Jan Hajič, Jaroslava Hlaváčová, Gareth J.f. Jones, Liadh Kelly, Johannes Leveling, David Mareček, Michal Novák, Martin Popel, Rudolf Rosa, Aleš Tamchyna, and Zdeňka Urešová. "Adaptation of Machine Translation for Multilingual Information Retrieval in the Medical Domain." *Artificial Intelligence in Medicine* 61.3 (2014): 165-85. Web.

[5] Géry, Mathias, and Christine Largeron. "BM25t: A BM25 Extension for Focused Information Retrieval." *Knowledge and Information Systems* 32.1 (2011): 217-41. Web.

[6] Gunter, Tracy D., and Nicolas P. Terry. "The Emergence of National Electronic Health Record Architectures in the United States and Australia: Models, Costs, and Questions." *Journal of Medical Internet Research* 7.1 (2005): n. pag. Web.

[7] "Text REtrieval Conference (TREC) Home Page." *Text REtrieval Conference (TREC) Home Page*. N.p., n.d. Web. 21 Feb. 2016.

<http://trec.nist.gov/>.

[8] Voorhees, E., and D. K. Harman. *TREC: Experiment and Evaluation in Information Retrieval*. Cambridge, MA: MIT, 2005. Print.

[9] Bendersky, Michael, Donald Metzler, and W. Bruce Croft. "Learning Concept Importance Using a Weighted Dependence Model."*Proceedings of the Third ACM International Conference on Web Search and Data Mining - WSDM '10* (2010): n. pag. Web.

[10] Sun, Heli, Jianbin Huang, and Boqin Feng. "QoRank: A Query-dependent Ranking Model Using LSE-based Weighted Multiple Hyperplanes Aggregation for Information Retrieval." *International Journal of Intelligent Systems.* 26.1 (2010): 73-97. Web.

[11] Baumgarten, Christoph. "A Probabilistic Model for Distributed Information Retrieval." *ACM SIGIR Forum* 31.SI (1997): 258-66. Web.

[12] Jalali, Vahid, and Mohammad Reza Matash Borujerdi. "Information Retrieval with Concept-based Pseudo-relevance Feedback in MEDLINE." *Knowledge and Information Systems* 29.1 (2010): 237-48. Web.

[13] Salton, G. "Some Research Problems in Automatic Information Retrieval." *ACM SIGIR Forum* 17.4 (1983): 252. Web.

[14] Efron, Miles. "Query Expansion and Dimensionality Reduction: Notions of Optimality in Rocchio Relevance Feedback and Latent Semantic Indexing." *Information Processing & Management* 44.1 (2008): 163-80. Web.

[15] Salton, Gerard, and Michael J. McGill. *Introduction to Modern Information Retrieval*. New York: McGraw-Hill, 1983. Print.

[16] Manson, Spero M. "Extending the Boundaries, Bridging the Gaps: Crafting Mental Health: Culture, Race, and Ethnicity, a Supplement to the Surgeon General's Report on Mental Health." *Culture, Medicine and Psychiatry* 27.4 (2003): 395-408. Web.

[17] "National Library of Medicine - National Institutes of Health." *U.S National Library of Medicine*. n.d. Web. 21 Feb. 2016.

<http://Nlm.nih.gov/>.

[18] Jordan, Michael I., Sara A. Solla, and Michael J. Kearns. *Advances in Neural Information Processing Systems 10*. Cambridge, MA: MIT, 1998. Print.

[19] "CEUR-WS.Org - CEUR Workshop Proceedings". Web. 2016.

<//http://Ceur-ws.org/>

[20] "Log In To Easychair For IEEE-TCDL-DC-2016". Web. 2016.

<//http:// Easychair.org/>

[21] Ballesteros, Lisa, and W. Bruce Croft. "Phrasal Translation and Query Expansion Techniques for Cross-language Information Retrieval."*ACM SIGIR Forum* 31.SI (1997): 84-91. Web.

[22] Kelly, Liadh, Lorraine Goeuriot, Hanna Suominen, Tobias Schreck, Gondy Leroy, Danielle L. Mowery, Sumithra Velupillai, Wendy W. Chapman, David Martinez, Guido Zuccon, and João

Palotti. "Overview of the ShARe/CLEF EHealth Evaluation Lab 2014. *Multilinguality, Multimodality, and Interaction* (2014): 172-91. Web.

[23] Robertson, S.E., S. Walker, and M.M. Hancock-Beaulieu. "Large Test Collection Experiments on an Operational, Interactive System: Okapi at TREC." *Information Processing & Management* 31.3 (1995): 345-60. Web.

[24] Besbes, Ghada, and Hajer Baazaoui-Zghal. "Modular Ontologies and CBR-based Hybrid System for Web Information Retrieval." *Multimedia Tools and Applications* 74.18 (2014): 8053-077. Web.

[25] Yoo, Sooyoung, and Jinwook Choi. "Evaluation of Term Ranking Algorithms for Pseudo-Relevance Feedback in MEDLINE Retrieval." *Healthc Inform Res Healthcare Informatics Research*17.2 (2011): 120. Web.

[26] Buettcher, Stefan. "Information Retrieval: Implementing and Evaluating Search Engines20115 Information Retrieval. Cambridge, MA: MIT Press 2011.40.9/10 (2011): 1555. Web.

[27] Wang, Xu-Yang. "Query Expansion Based on User Relevance Feedback and Ontology." *Journal of Computer Applications* 28.11 (2009): 2958-960. Web.

[28] Martín Vide, Carlos. *Mathematical And Computational Analysis Of Natural Language*. Amsterdam: John Benjamins Pub. Co., 1998. Print.

[29] Manning, Christopher D, Prabhakar Raghavan, and Hinrich Schütze. *Introduction To Information Retrieval*. New York: Cambridge University Press, 2008. Print.

[30] Salton, G., A. Wong, and C. S. Yang. "A Vector Space Model for Automatic Indexing." *Communications of the ACM Commun.* 18.11 (1975): 613-20. Web.

[31] Moreno, Edward David. "Platforms and Applications in Hardware Security: Trends and Challenges." *International Journal of Security and Its Applications* 7.5 (2013): 289-304. Web.

[32] Trotman, Andrew. "Learning to Rank." *Information Retrieval* 8.3 (2005): 359-81. Web.

[33] ”The Rich Release Format (RRF) Subset Browser" Web. 2016.

<http://Nlm.nih.gov/>

[34] Rudd, Murray A. "Scientists’ Opinions on the Global Status and Management of Biological Diversity." *Conservation Biology* 25.6 (2011): 1165-175. Web.

[35] Nalchigar, Soroosh, S.M.R. Nasserzadeh, and Babak Akhgar. "Simulating Strategic Information Systems Planning Process Using Fuzzy Cognitive Map". *International Journal of Business Information Systems* 8.3 (2011): 286.

[36] "Metamap - A Tool For Recognizing UMLS Concepts In Text". Web. 2016.

<http:// Metamap.nlm.nih.gov>

[37] Song, Weihua, and XianWei Wu. "Application of Relevance Feedback Based On Rocchio Theory For Medical Image Retrieval". *Advanced Science Letters* 10.1 (2012): 295-298.

[38] Song, Weihua, and Xianwei Wu. "Application of Relevance Feedback Based on Rocchio Theory for Medical Image Retrieval." *Advanced Science Letters* 10.1 (2012): 295-98. Web.

[39] He, Yan, and Shu Jin Wang. "The Application and Study on Intelligent Real-Time Machine Translation Technology."*Applied Mechanics and Materials* 687-691 (2014): 1695-699. Web.

[40] Peters, C. *Advances in Cross-language Information Retrieval: Third Workshop of the Cross-Language Evaluation Forum, CLEF 2002: Revised Papers*. Berlin: Springer, 2003. Print.

[41] Ye, Zheng, Jimmy Xiangji Huang, and Hongfei Lin. "Finding a Good Query-related Topic for Boosting Pseudo-relevance Feedback." *Journal of the American Society for Information Science and Technology* 62.4 (2011): 748-60. Web.

[42] Robertson, S.E., C.l. Thompson, M.J. Macaskill, and J.D. Bovey. "Weighting, Ranking and Relevance Feedback in a Front--end System." *Journal of Information Science* 12.1-2 (1986): 71-75. Web.

[43] Shivade, Chaitanya, Pranav Malewadkar, Eric Fosler-Lussier, and Albert M. Lai. "Comparison of UMLS Terminologies to Identify Risk of Heart Disease Using Clinical Notes." *Journal of Biomedical Informatics* 58 (2015): n. pag. Web.

[44] Needham, Christopher Donald. *Organizing Knowledge in Libraries: An Introduction to Information Retrieval*. London: Deutsch, 1971. Print.

[45] Bommel, Patrick Van. *Information Modeling for Internet Applications*. Hershey PA: Idea Group Pub., 2003. Print.

[46] Xu, Yangsheng, and Ming Ge. "Hidden Markov Model-based Process Monitoring System." *Journal of Intelligent Manufacturing* 15.3 (2004): 337-50. Web.

[47] Rowley, J. E. *Organising Knowledge: An Introduction to Information Retrieval*. Aldershot, Hants, England: Gower Pub., 1987. Print.

[48] Stojanovic, Nenad. *Ontology-based Information Retrieval: Methods and Tools for Cooperative Query Answering*. S.l.: S.n., 2005. Print.

[49] Wimalasuriya, D. C., and Dejing Dou. "Ontology-based Information Extraction: An Introduction and a Survey of Current Approaches."*Journal of Information Science* 36.3 (2010): 306-23. Web.

[50] Sondhi, P., J. Sun, C. Zhai, R. Sorrentino, and M. S. Kohn. "Leveraging Medical Thesauri and Physician Feedback for Improving Medical Literature Retrieval for Case Queries." *Journal of the American Medical Informatics Association* 19.5 (2012): 851-58. Web.

[51] Zghal, Hajer Baazaoui, and Antonio Moreno. "A System for Information Retrieval in a Medical Digital Library Based on Modular Ontologies and Query Reformulation." *Multimedia Tools and Applications* 72.3 (2013): 2393-412. Web.

[52] O'Shaughnessy, Douglas. "Invited Paper: Automatic Speech Recognition: History, Methods and Challenges." *Pattern Recognition* 41.10 (2008): 2965-979. Web.

[53] Coden, Anni R., Eric W. Brown, and Savitha Srinivasan. *Information Retrieval Techniques for Speech Applications*. Berlin: Springer, 2002. Print.

[54] Hu, Qinmin, and Jimmy Xiangji Huang. "Passage Extraction and Result Combination for Genomics Information Retrieval." *Journal of Intelligent Information Systems* 34.3 (2009): 249-74. Web.

[55] Lee, C., C. Grasso, and M. F. Sharlow. "Multiple Sequence Alignment Using Partial Order Graphs." *Bioinformatics* 18.3 (2002): 452-64. Web.

[56] Hersh, William. "Relevance and Retrieval Evaluation: Perspectives from Medicine." *Journal of the American Society for Information Science.* 45.3 (1994): 201-06. Web.

[57] Gall, Carole, and Frances A. Brahmi. "Retrieval Comparison of EndNote to Search MEDLINE (Ovid and PubMed) versus Searching Them Directly." *Medical Reference Services Quarterly* 23.3 (2004): 25-32. Web.

[58] Minguet, Fernando, Teresa M. Salgado, Lucienne Van Den Boogerd, and Fernando Fernandez-Llimos. "Quality of Pharmacy-specific Medical Subject Headings (MeSH) Assignment in Pharmacy Journals Indexed in MEDLINE." *Research in Social and Administrative Pharmacy* 11.5 (2015): 686-95. Web.

[59] Callan, Jamie. "Distributed Information Retrieval." *The Information Retrieval Series Advances in Information Retrieval* (2003): 127-50. Web.

[60] Braschler, Martin, and Bärbel Ripplinger. "Stemming and Decompounding for German Text Retrieval." *Advances in Information Retrieval* (2003): 177-92. Web.

[61] “Abstracts of Articles in the Information Retrieval Area Selected by Gerard Salton." *ACM SIGIR Forum* 21.1-2 (1986): 39-50. Web.

[62] Prince, Violaine, and Mathieu Roche. *Information Retrieval in Biomedicine: Natural Language Processing for Knowledge Integration*. Hershey: Medical Information Science Reference, 2009. Print.

[63] Zhou, Lina, and Dongsong Zhang. "NLPIR: A Theoretical Framework for Applying Natural Language Processing to Information Retrieval." *Journal of the American Society for Information Science and Technology* 54.2 (2003): 115-23. Web.

[64] Peters, C., Martin Braschler, and Paul Clough. *Multilingual Information Retrieval: From Research to Practice*. Heidelberg: Springer, 2012. Print.

[65 Koopman, Bevan. "Towards Semantic Search and Inference in Electronic Medical Records: An Approach Using Concept Based Information Retrieval." *Australasian Medical Journal* 5.9 (2012): 482-88. Web.

[66] "Roi Blanco's Academic Home Page." *Roi Blanco's Academic Home Page*. Web. 2016.

< http://www.dc.fi.udc.es/~roi/>

[67] Newsam, Shawn, Sitaram Bhagavathy, Charles Kenney, B.s. Manjunath, and Leila Fonseca. "Object-based Representations of Spatial Images." *Acta Astronautica* 48.5-12 (2001): 567-77. Web.

[68] Michelangeli, Alessandro. "Role of Scaling Limits in the Rigorous Analysis of Bose-Einstein Condensation." *Journal of Mathematical Physics* 48.10 (2007): 102102. Web.

[69] Rubi, J. M. "Book Review: Bose–Einstein Condensation. Lev Pitaevskii and Sandro Stringari, Oxford University Press, Oxford, 2003."*Journal of Statistical Physics* 115.5/6 (2004): 1763-764. Web.

[70] Klampanos, Iraklis A. "Manning Christopher, Prabhakar Raghavan, Hinrich Schütze: Introduction to Information Retrieval." *Information Retrieval* 12.5 (2009): 609-12. Web.

[71] Yin, Xiaoshi, Jimmy Xiangji Huang, Zhoujun Li, and Xiaofeng Zhou. "A Survival Modeling Approach to Biomedical Search Result Diversification Using Wikipedia." *IEEE Transactions on Knowledge and Data Engineering* 25.6 (2013): 1201-212. Web.

[72] Hersh, William R. "Report on the TREC 2004 Genomics Track." *ACM SIGIR Forum* 39.1 (2005): 21. Web.

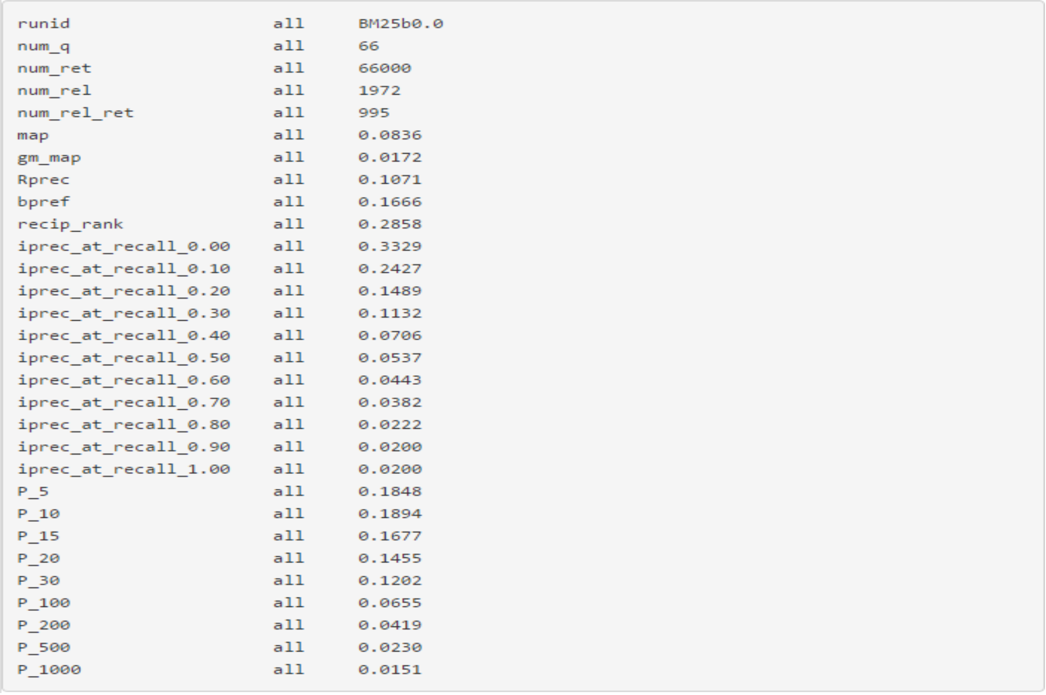
[73] Andreasen, Troels, and Henrik Bulskov. "Summarization by Domain Ontology Navigation." *International Journal of Intelligent Systems.* 28.1 (2012): 72-92. Web.

[74] Budanitsky, Alexander, and Graeme Hirst. "Evaluating WordNet-based Measures of Lexical Semantic Relatedness." *Computational Linguistics* 32.1 (2006): 13-47. Web.

[75] Chen, Peter Pin-Shan. "The Entity-relationship Model---toward a Unified View of Data." *ACM Transactions on Database Systems.* 1.1 (1976): 9-36. Web.

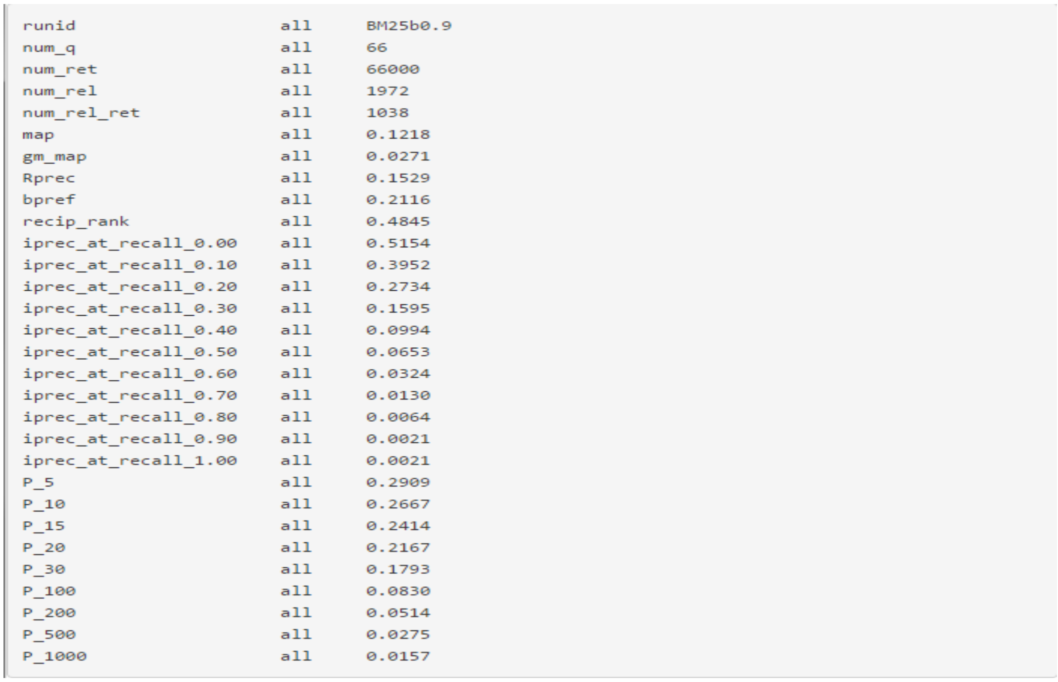
Appendix A:

YorkU_EN_Run.1.dat



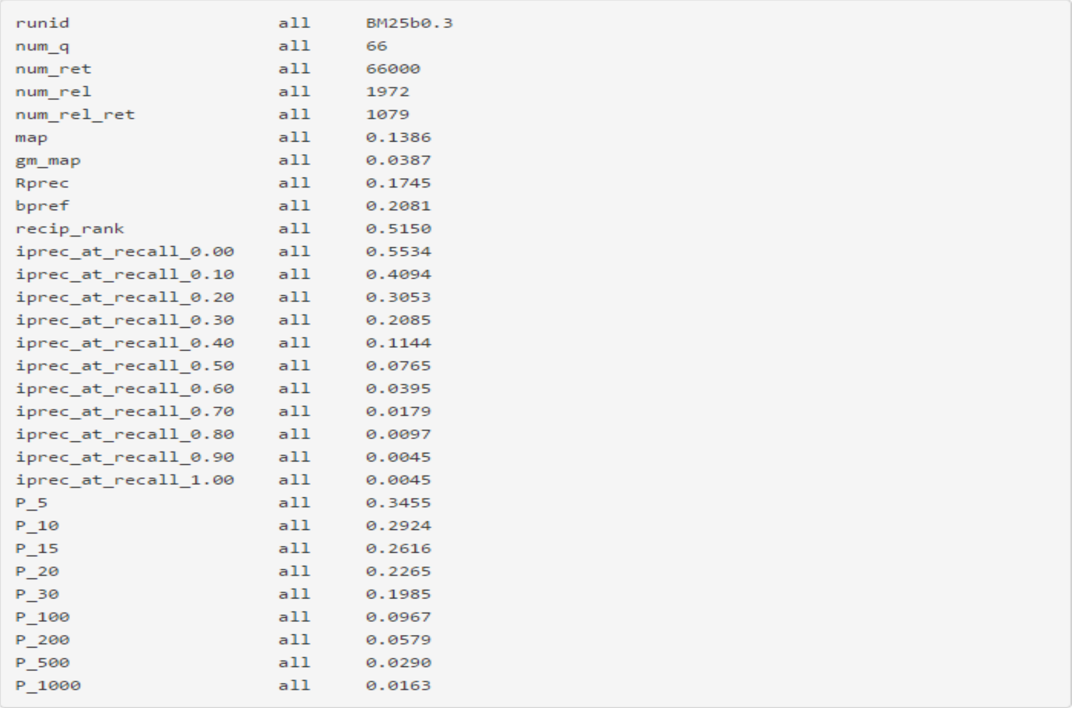
```
runid                    all     BM25b0.0
num_q                    all     66
num_ret                  all     66000
num_rel                  all     1972
num_rel_ret              all     995
map                      all     0.0836
gm_map                   all     0.0172
Rprec                    all     0.1071
bpref                    all     0.1666
recip_rank               all     0.2858
iprec_at_recall_0.00     all     0.3329
iprec_at_recall_0.10     all     0.2427
iprec_at_recall_0.20     all     0.1489
iprec_at_recall_0.30     all     0.1132
iprec_at_recall_0.40     all     0.0706
iprec_at_recall_0.50     all     0.0537
iprec_at_recall_0.60     all     0.0443
iprec_at_recall_0.70     all     0.0382
iprec_at_recall_0.80     all     0.0222
iprec_at_recall_0.90     all     0.0200
iprec_at_recall_1.00     all     0.0200
P_5                      all     0.1848
P_10                     all     0.1894
P_15                     all     0.1677
P_20                     all     0.1455
P_30                     all     0.1202
P_100                    all     0.0655
P_200                    all     0.0419
P_500                    all     0.0230
P_1000                   all     0.0151
```

YorkU_EN_Run.10.dat



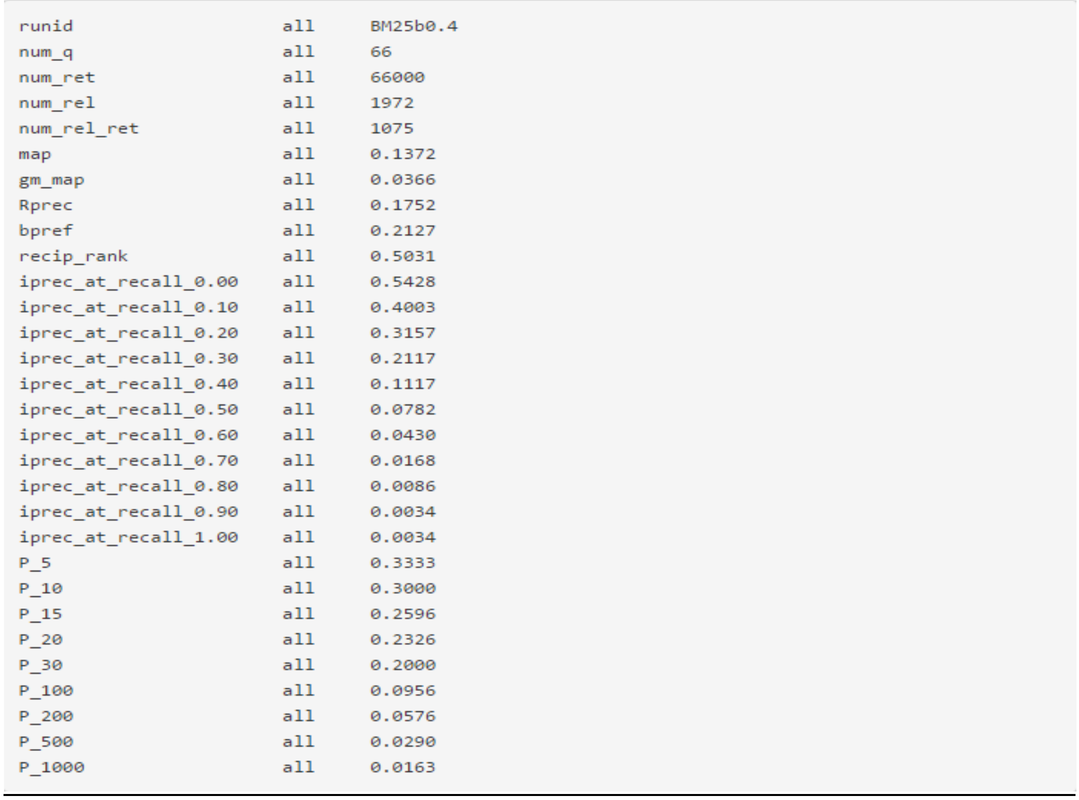
```
runid                    all     BM25b0.9
num_q                    all     66
num_ret                  all     66000
num_rel                  all     1972
num_rel_ret              all     1038
map                      all     0.1218
gm_map                   all     0.0271
Rprec                    all     0.1529
bpref                    all     0.2116
recip_rank               all     0.4845
iprec_at_recall_0.00     all     0.5154
iprec_at_recall_0.10     all     0.3952
iprec_at_recall_0.20     all     0.2734
iprec_at_recall_0.30     all     0.1595
iprec_at_recall_0.40     all     0.0994
iprec_at_recall_0.50     all     0.0653
iprec_at_recall_0.60     all     0.0324
iprec_at_recall_0.70     all     0.0130
iprec_at_recall_0.80     all     0.0064
iprec_at_recall_0.90     all     0.0021
iprec_at_recall_1.00     all     0.0021
P_5                      all     0.2909
P_10                     all     0.2667
P_15                     all     0.2414
P_20                     all     0.2167
P_30                     all     0.1793
P_100                    all     0.0830
P_200                    all     0.0514
P_500                    all     0.0275
P_1000                   all     0.0157
```

YorkU_EN_Run.4.dat



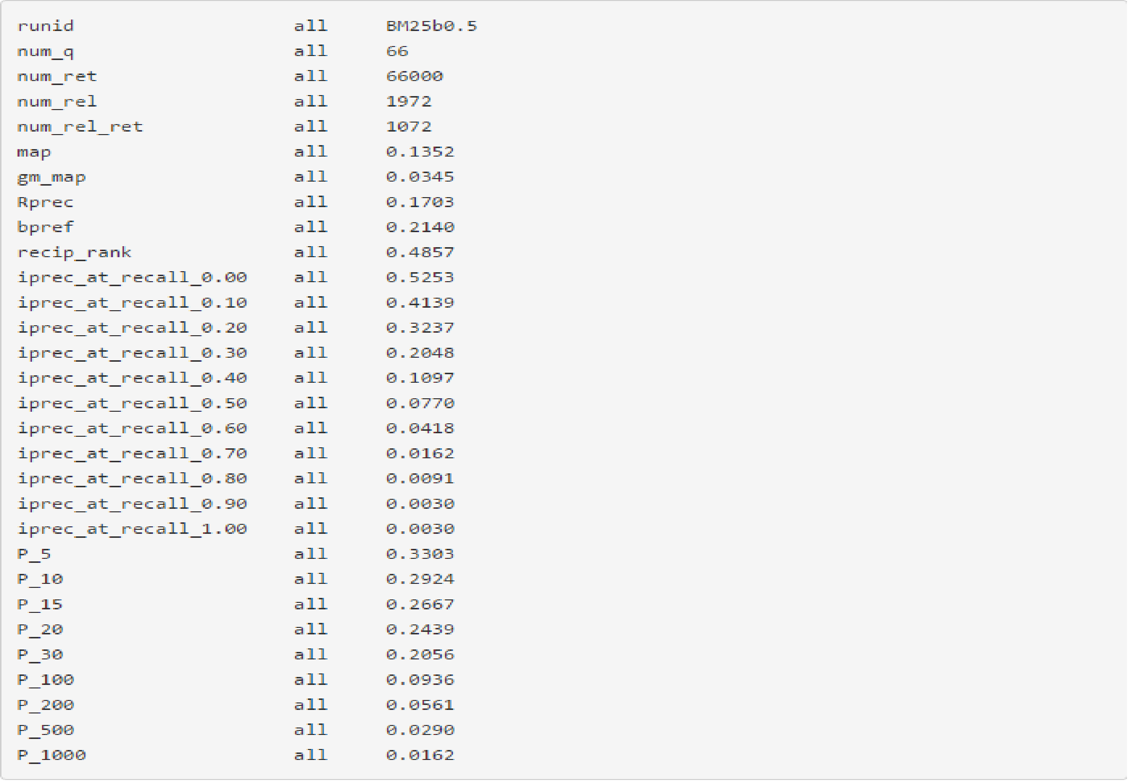
```
runid                   all     BM25b0.3
num_q                   all     66
num_ret                 all     66000
num_rel                 all     1972
num_rel_ret             all     1079
map                     all     0.1386
gm_map                  all     0.0387
Rprec                   all     0.1745
bpref                   all     0.2081
recip_rank              all     0.5150
iprec_at_recall_0.00    all     0.5534
iprec_at_recall_0.10    all     0.4094
iprec_at_recall_0.20    all     0.3053
iprec_at_recall_0.30    all     0.2085
iprec_at_recall_0.40    all     0.1144
iprec_at_recall_0.50    all     0.0765
iprec_at_recall_0.60    all     0.0395
iprec_at_recall_0.70    all     0.0179
iprec_at_recall_0.80    all     0.0097
iprec_at_recall_0.90    all     0.0045
iprec_at_recall_1.00    all     0.0045
P_5                     all     0.3455
P_10                    all     0.2924
P_15                    all     0.2616
P_20                    all     0.2265
P_30                    all     0.1985
P_100                   all     0.0967
P_200                   all     0.0579
P_500                   all     0.0290
P_1000                  all     0.0163
```

YorkU_EN_Run.5.dat



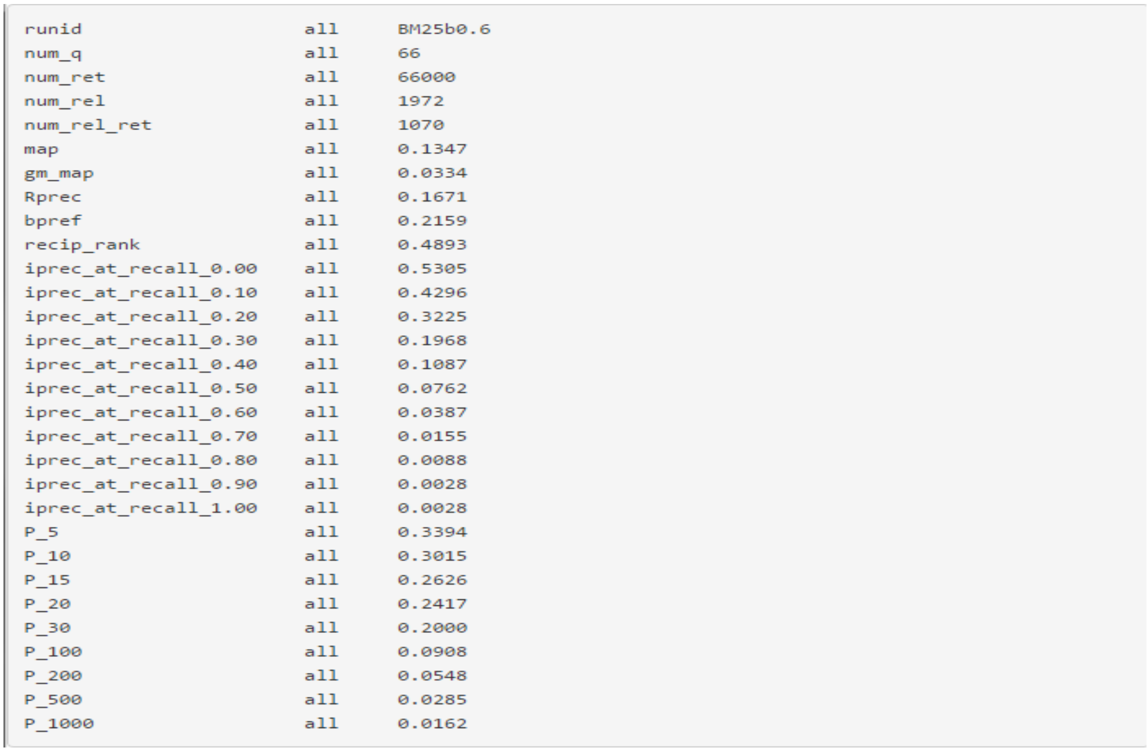
```
runid                   all     BM25b0.4
num_q                   all     66
num_ret                 all     66000
num_rel                 all     1972
num_rel_ret             all     1075
map                     all     0.1372
gm_map                  all     0.0366
Rprec                   all     0.1752
bpref                   all     0.2127
recip_rank              all     0.5031
iprec_at_recall_0.00    all     0.5428
iprec_at_recall_0.10    all     0.4003
iprec_at_recall_0.20    all     0.3157
iprec_at_recall_0.30    all     0.2117
iprec_at_recall_0.40    all     0.1117
iprec_at_recall_0.50    all     0.0782
iprec_at_recall_0.60    all     0.0430
iprec_at_recall_0.70    all     0.0168
iprec_at_recall_0.80    all     0.0086
iprec_at_recall_0.90    all     0.0034
iprec_at_recall_1.00    all     0.0034
P_5                     all     0.3333
P_10                    all     0.3000
P_15                    all     0.2596
P_20                    all     0.2326
P_30                    all     0.2000
P_100                   all     0.0956
P_200                   all     0.0576
P_500                   all     0.0290
P_1000                  all     0.0163
```

YorkU_EN_Run.6.dat



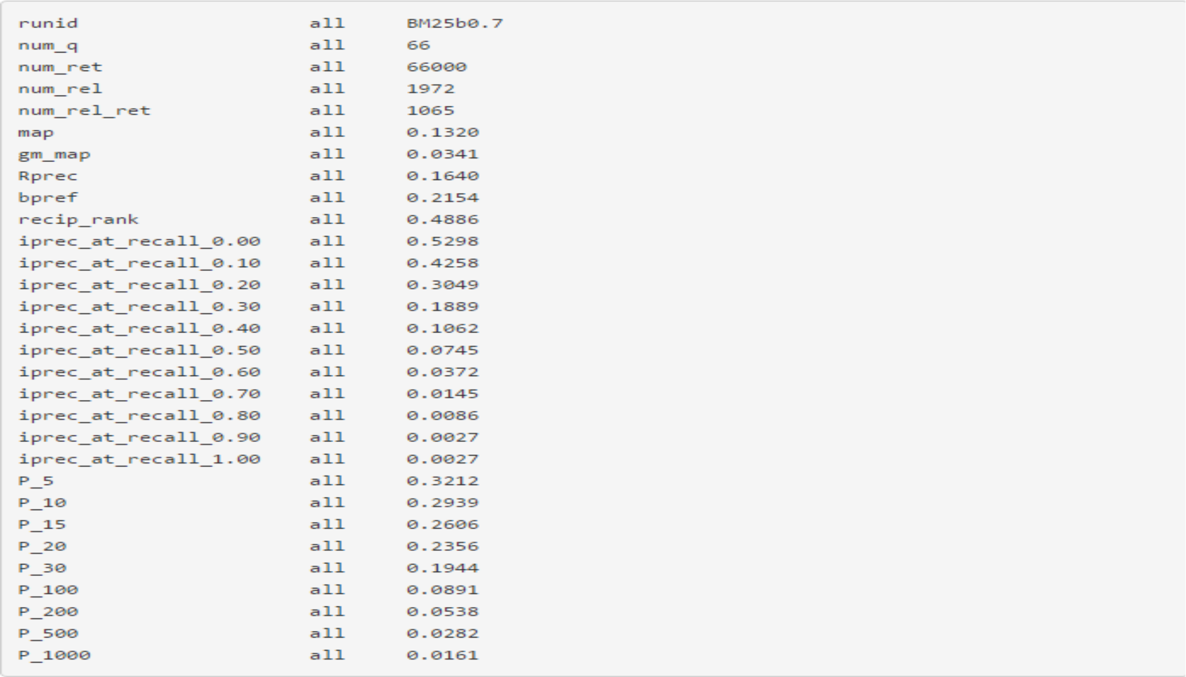
```
runid                   all     BM25b0.5
num_q                   all     66
num_ret                 all     66000
num_rel                 all     1972
num_rel_ret             all     1072
map                     all     0.1352
gm_map                  all     0.0345
Rprec                   all     0.1703
bpref                   all     0.2140
recip_rank              all     0.4857
iprec_at_recall_0.00    all     0.5253
iprec_at_recall_0.10    all     0.4139
iprec_at_recall_0.20    all     0.3237
iprec_at_recall_0.30    all     0.2048
iprec_at_recall_0.40    all     0.1097
iprec_at_recall_0.50    all     0.0770
iprec_at_recall_0.60    all     0.0418
iprec_at_recall_0.70    all     0.0162
iprec_at_recall_0.80    all     0.0091
iprec_at_recall_0.90    all     0.0030
iprec_at_recall_1.00    all     0.0030
P_5                     all     0.3303
P_10                    all     0.2924
P_15                    all     0.2667
P_20                    all     0.2439
P_30                    all     0.2056
P_100                   all     0.0936
P_200                   all     0.0561
P_500                   all     0.0290
P_1000                  all     0.0162
```

YorkU_EN_Run.7.dat



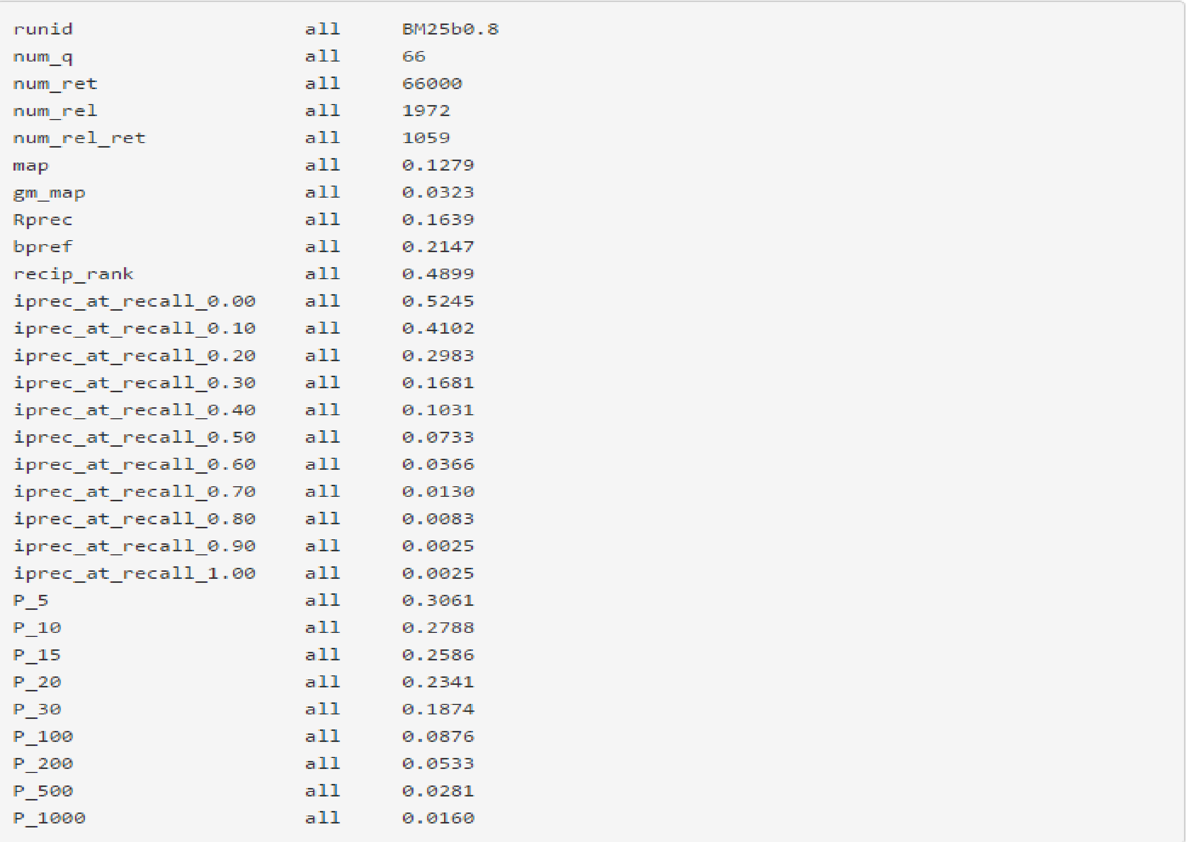
```
runid                   all     BM25b0.6
num_q                   all     66
num_ret                 all     66000
num_rel                 all     1972
num_rel_ret             all     1070
map                     all     0.1347
gm_map                  all     0.0334
Rprec                   all     0.1671
bpref                   all     0.2159
recip_rank              all     0.4893
iprec_at_recall_0.00    all     0.5305
iprec_at_recall_0.10    all     0.4296
iprec_at_recall_0.20    all     0.3225
iprec_at_recall_0.30    all     0.1968
iprec_at_recall_0.40    all     0.1087
iprec_at_recall_0.50    all     0.0762
iprec_at_recall_0.60    all     0.0387
iprec_at_recall_0.70    all     0.0155
iprec_at_recall_0.80    all     0.0088
iprec_at_recall_0.90    all     0.0028
iprec_at_recall_1.00    all     0.0028
P_5                     all     0.3394
P_10                    all     0.3015
P_15                    all     0.2626
P_20                    all     0.2417
P_30                    all     0.2000
P_100                   all     0.0908
P_200                   all     0.0548
P_500                   all     0.0285
P_1000                  all     0.0162
```

YorkU_EN_Run.8.dat



```
runid                    all     BM25b0.7
num_q                    all     66
num_ret                  all     66000
num_rel                  all     1972
num_rel_ret              all     1065
map                      all     0.1320
gm_map                   all     0.0341
Rprec                    all     0.1640
bpref                    all     0.2154
recip_rank               all     0.4886
iprec_at_recall_0.00     all     0.5298
iprec_at_recall_0.10     all     0.4258
iprec_at_recall_0.20     all     0.3049
iprec_at_recall_0.30     all     0.1889
iprec_at_recall_0.40     all     0.1062
iprec_at_recall_0.50     all     0.0745
iprec_at_recall_0.60     all     0.0372
iprec_at_recall_0.70     all     0.0145
iprec_at_recall_0.80     all     0.0086
iprec_at_recall_0.90     all     0.0027
iprec_at_recall_1.00     all     0.0027
P_5                      all     0.3212
P_10                     all     0.2939
P_15                     all     0.2606
P_20                     all     0.2356
P_30                     all     0.1944
P_100                    all     0.0891
P_200                    all     0.0538
P_500                    all     0.0282
P_1000                   all     0.0161
```

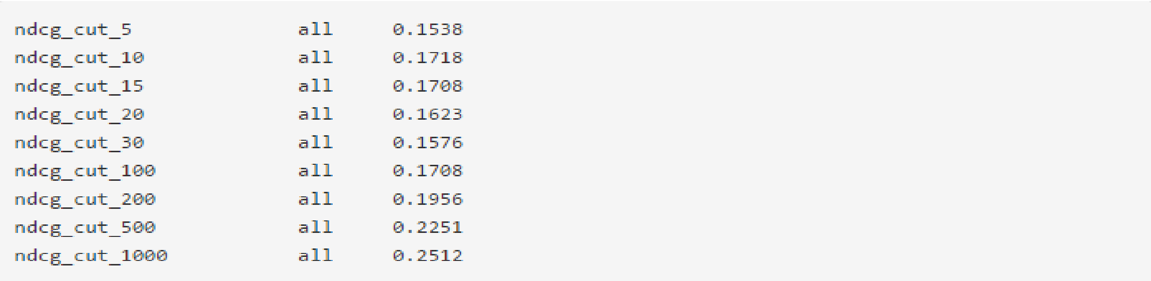
YorkU_EN_Run.9.dat



```
runid                    all     BM25b0.8
num_q                    all     66
num_ret                  all     66000
num_rel                  all     1972
num_rel_ret              all     1059
map                      all     0.1279
gm_map                   all     0.0323
Rprec                    all     0.1639
bpref                    all     0.2147
recip_rank               all     0.4899
iprec_at_recall_0.00     all     0.5245
iprec_at_recall_0.10     all     0.4102
iprec_at_recall_0.20     all     0.2983
iprec_at_recall_0.30     all     0.1681
iprec_at_recall_0.40     all     0.1031
iprec_at_recall_0.50     all     0.0733
iprec_at_recall_0.60     all     0.0366
iprec_at_recall_0.70     all     0.0130
iprec_at_recall_0.80     all     0.0083
iprec_at_recall_0.90     all     0.0025
iprec_at_recall_1.00     all     0.0025
P_5                      all     0.3061
P_10                     all     0.2788
P_15                     all     0.2586
P_20                     all     0.2341
P_30                     all     0.1874
P_100                    all     0.0876
P_200                    all     0.0533
P_500                    all     0.0281
P_1000                   all     0.0160
```

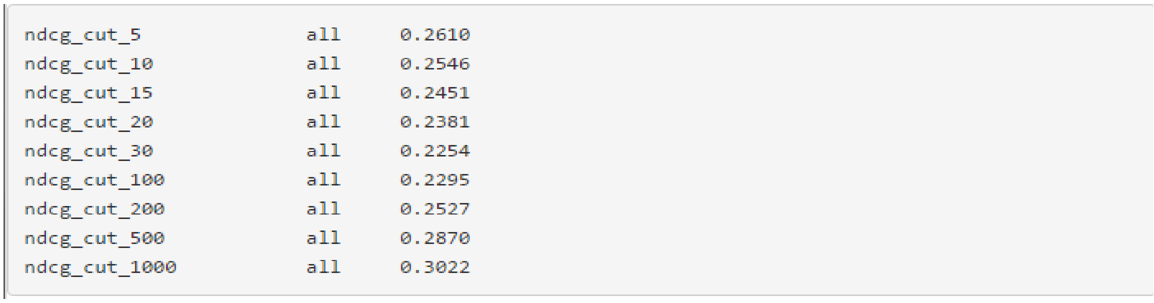
Appendix B:

YorkU_EN_Run.1.dat



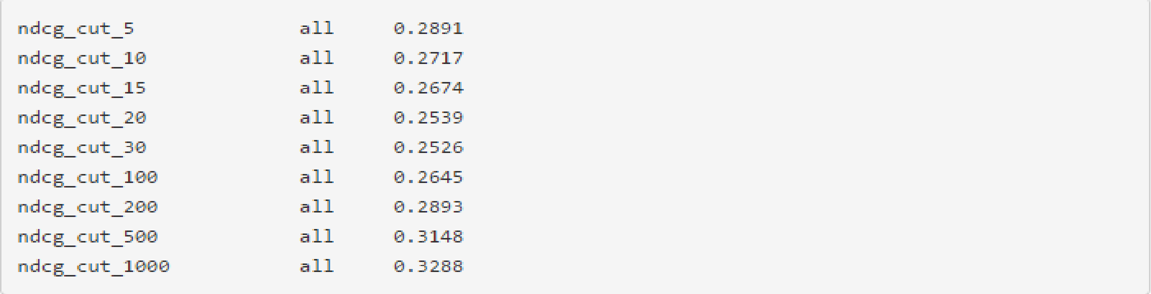
```
ndcg_cut_5              all     0.1538
ndcg_cut_10             all     0.1718
ndcg_cut_15             all     0.1708
ndcg_cut_20             all     0.1623
ndcg_cut_30             all     0.1576
ndcg_cut_100            all     0.1708
ndcg_cut_200            all     0.1956
ndcg_cut_500            all     0.2251
ndcg_cut_1000           all     0.2512
```

YorkU_EN_Run.10.dat



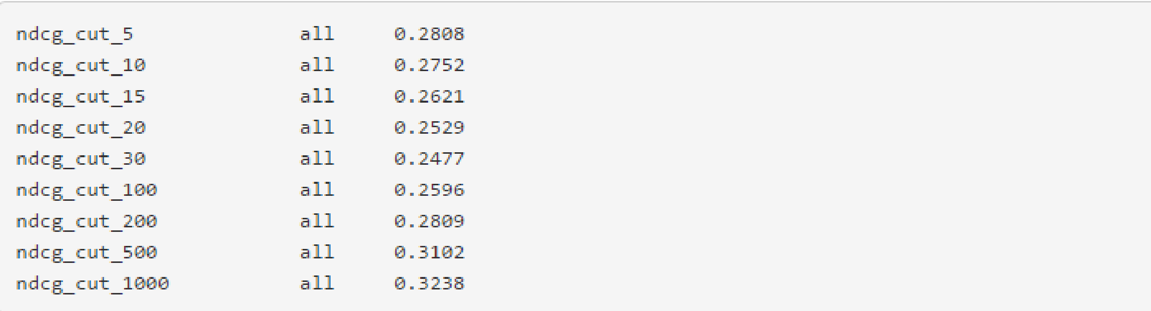
```
ndcg_cut_5              all     0.2610
ndcg_cut_10             all     0.2546
ndcg_cut_15             all     0.2451
ndcg_cut_20             all     0.2381
ndcg_cut_30             all     0.2254
ndcg_cut_100            all     0.2295
ndcg_cut_200            all     0.2527
ndcg_cut_500            all     0.2870
ndcg_cut_1000           all     0.3022
```

YorkU_EN_Run.4.dat



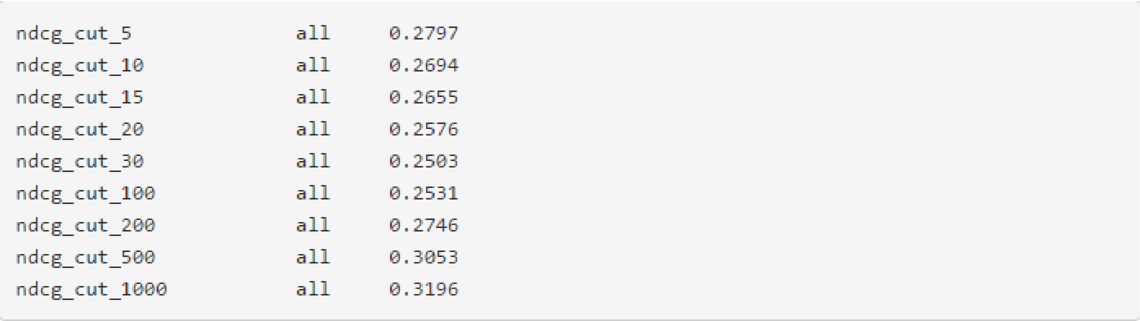
```
ndcg_cut_5              all     0.2891
ndcg_cut_10             all     0.2717
ndcg_cut_15             all     0.2674
ndcg_cut_20             all     0.2539
ndcg_cut_30             all     0.2526
ndcg_cut_100            all     0.2645
ndcg_cut_200            all     0.2893
ndcg_cut_500            all     0.3148
ndcg_cut_1000           all     0.3288
```

YorkU_EN_Run.5.dat



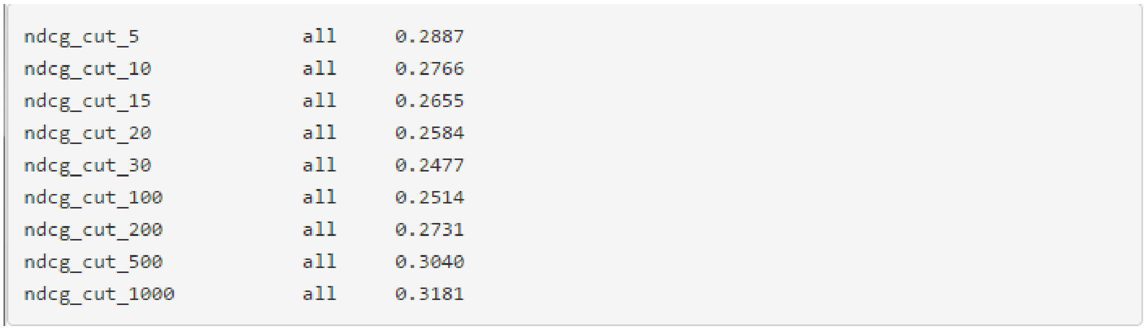
```
ndcg_cut_5              all     0.2808
ndcg_cut_10             all     0.2752
ndcg_cut_15             all     0.2621
ndcg_cut_20             all     0.2529
ndcg_cut_30             all     0.2477
ndcg_cut_100            all     0.2596
ndcg_cut_200            all     0.2809
ndcg_cut_500            all     0.3102
ndcg_cut_1000           all     0.3238
```

YorkU_EN_Run.6.dat



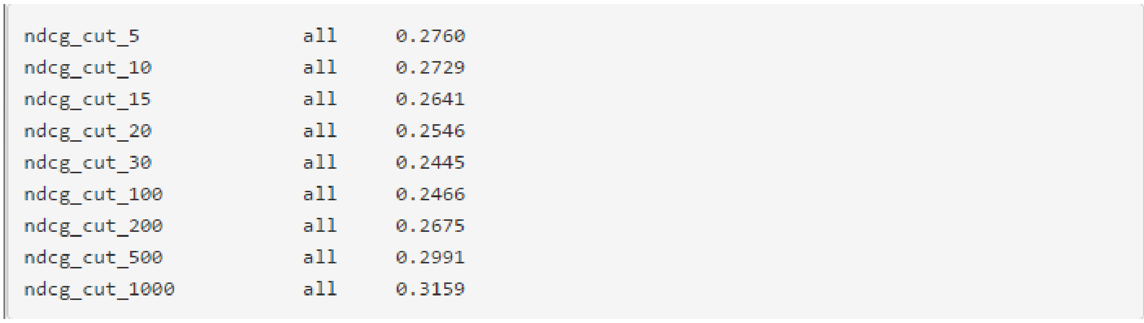
```
ndcg_cut_5               all     0.2797
ndcg_cut_10              all     0.2694
ndcg_cut_15              all     0.2655
ndcg_cut_20              all     0.2576
ndcg_cut_30              all     0.2503
ndcg_cut_100             all     0.2531
ndcg_cut_200             all     0.2746
ndcg_cut_500             all     0.3053
ndcg_cut_1000            all     0.3196
```

YorkU_EN_Run.7.dat



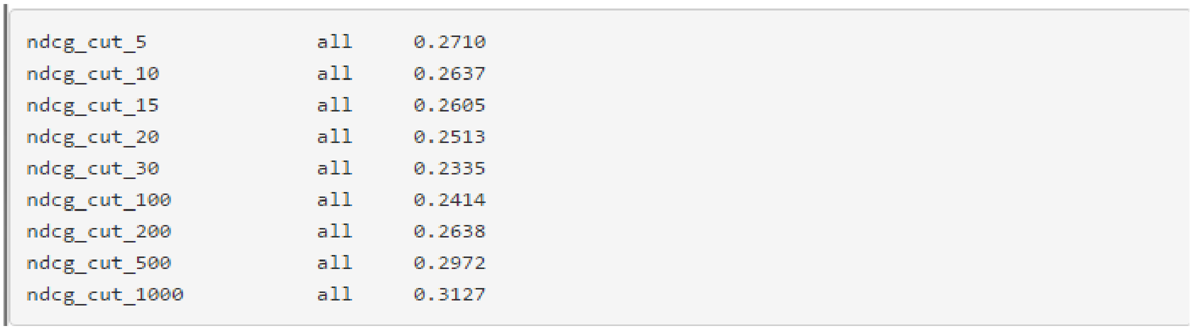
```
ndcg_cut_5               all     0.2887
ndcg_cut_10              all     0.2766
ndcg_cut_15              all     0.2655
ndcg_cut_20              all     0.2584
ndcg_cut_30              all     0.2477
ndcg_cut_100             all     0.2514
ndcg_cut_200             all     0.2731
ndcg_cut_500             all     0.3040
ndcg_cut_1000            all     0.3181
```

YorkU_EN_Run.8.dat



```
ndcg_cut_5               all     0.2760
ndcg_cut_10              all     0.2729
ndcg_cut_15              all     0.2641
ndcg_cut_20              all     0.2546
ndcg_cut_30              all     0.2445
ndcg_cut_100             all     0.2466
ndcg_cut_200             all     0.2675
ndcg_cut_500             all     0.2991
ndcg_cut_1000            all     0.3159
```

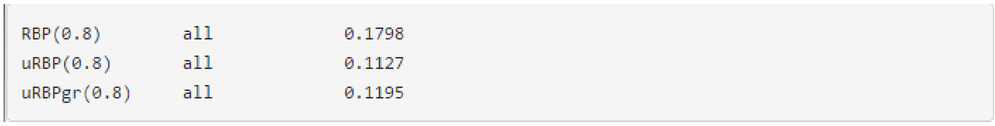
YorkU_EN_Run.9.dat



```
ndcg_cut_5               all     0.2710
ndcg_cut_10              all     0.2637
ndcg_cut_15              all     0.2605
ndcg_cut_20              all     0.2513
ndcg_cut_30              all     0.2335
ndcg_cut_100             all     0.2414
ndcg_cut_200             all     0.2638
ndcg_cut_500             all     0.2972
ndcg_cut_1000            all     0.3127
```

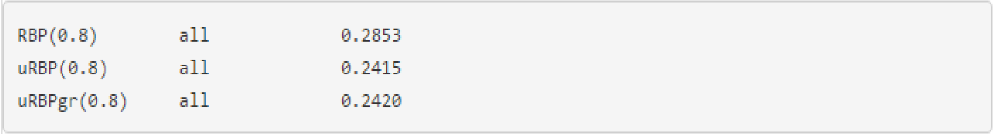
Appendix C:

YorkU_EN_Run.1.dat



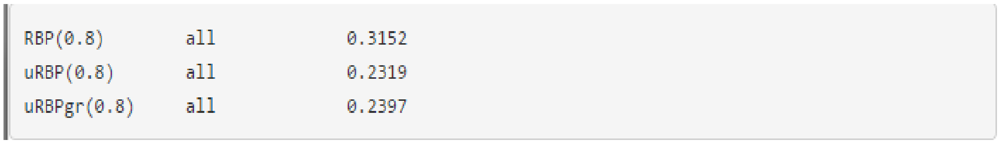
```
RBP(0.8)        all             0.1798
uRBP(0.8)       all             0.1127
uRBPgr(0.8)     all             0.1195
```

YorkU_EN_Run.10.dat



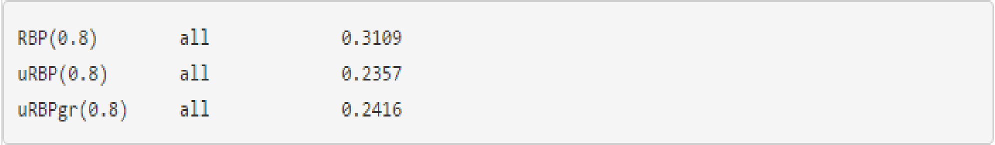
```
RBP(0.8)        all             0.2853
uRBP(0.8)       all             0.2415
uRBPgr(0.8)     all             0.2420
```

YorkU_EN_Run.4.dat



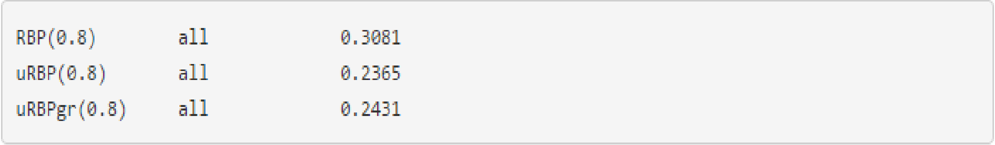
```
RBP(0.8)        all             0.3152
uRBP(0.8)       all             0.2319
uRBPgr(0.8)     all             0.2397
```

YorkU_EN_Run.5.dat



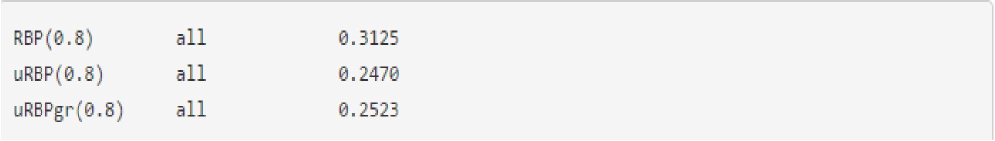
```
RBP(0.8)        all             0.3109
uRBP(0.8)       all             0.2357
uRBPgr(0.8)     all             0.2416
```

YorkU_EN_Run.6.dat



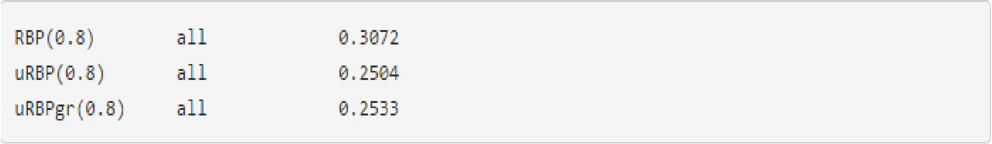
```
RBP(0.8)        all             0.3081
uRBP(0.8)       all             0.2365
uRBPgr(0.8)     all             0.2431
```

YorkU_EN_Run.7.dat



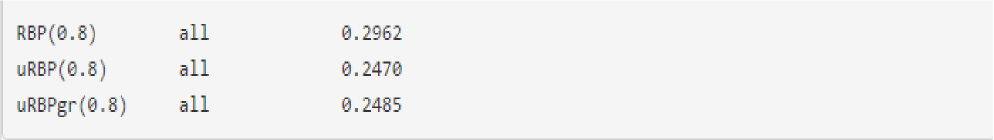
```
RBP(0.8)        all             0.3125
uRBP(0.8)       all             0.2470
uRBPgr(0.8)     all             0.2523
```

YorkU_EN_Run.8.dat



```
RBP(0.8)        all             0.3072
uRBP(0.8)       all             0.2504
uRBPgr(0.8)     all             0.2533
```

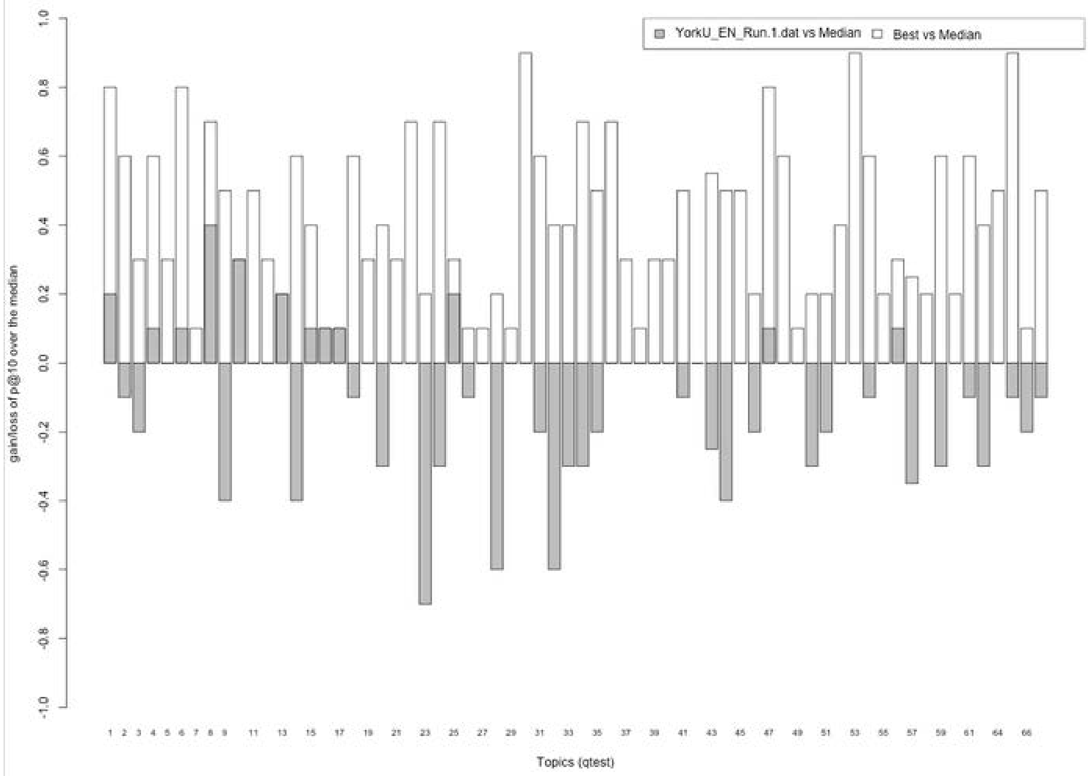
YorkU_EN_Run.9.dat



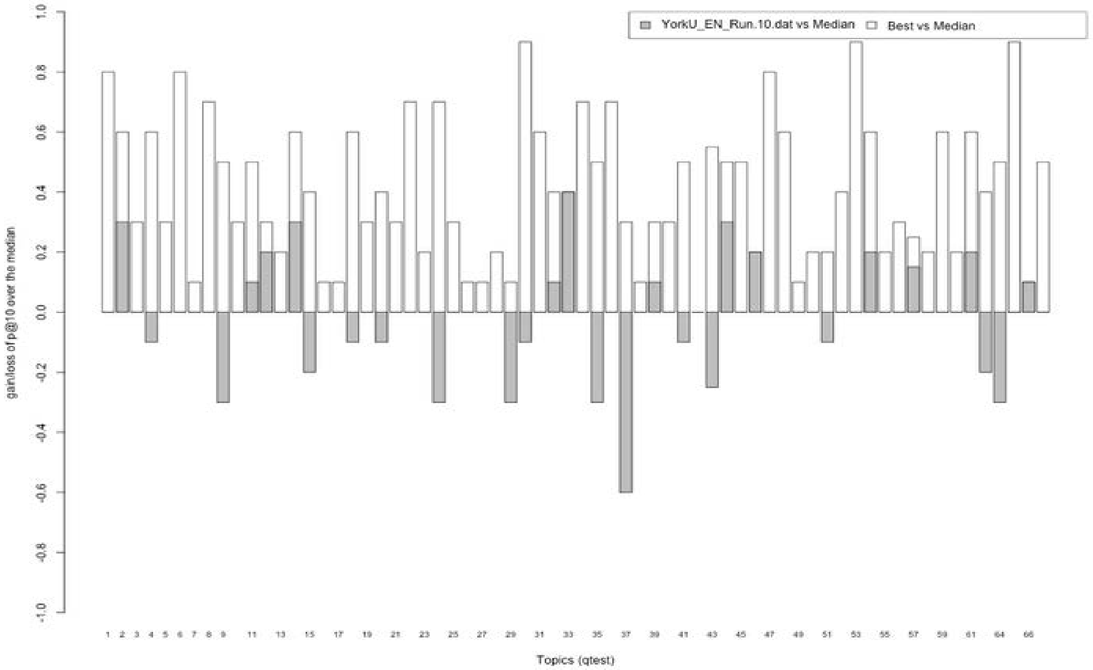
```
RBP(0.8)       all           0.2962
uRBP(0.8)      all           0.2470
uRBPgr(0.8)    all           0.2485
```

Appendix D:

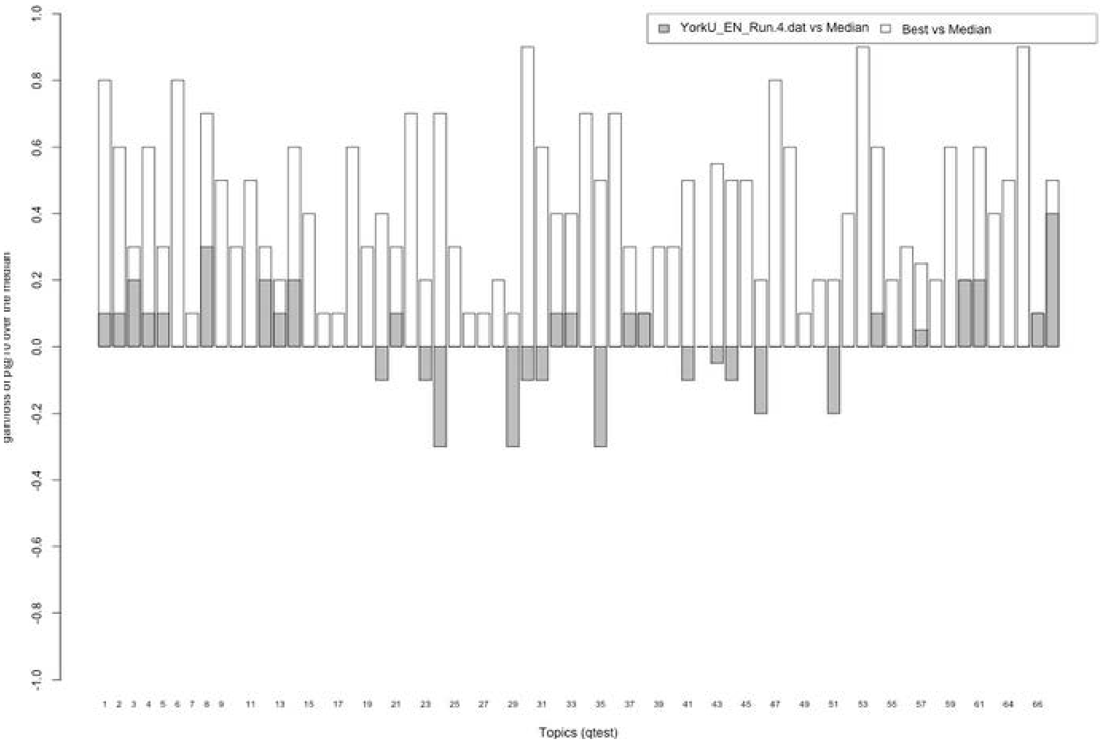
YorkU_EN_Run.1.dat



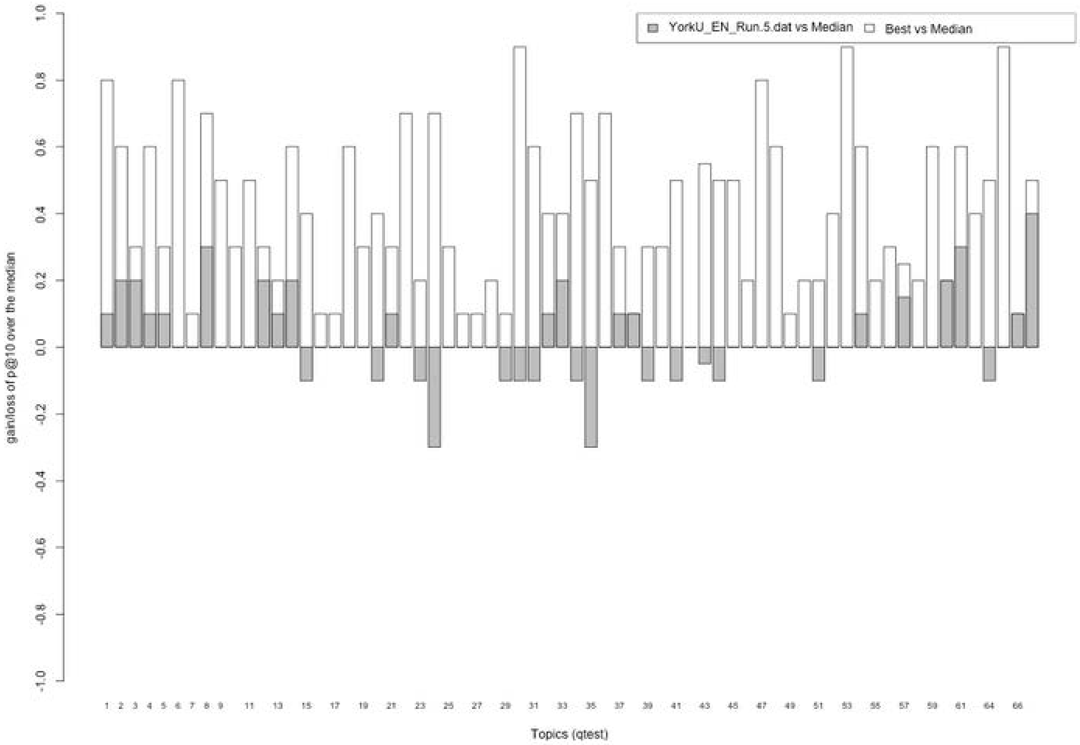
YorkU_EN_Run.10.dat



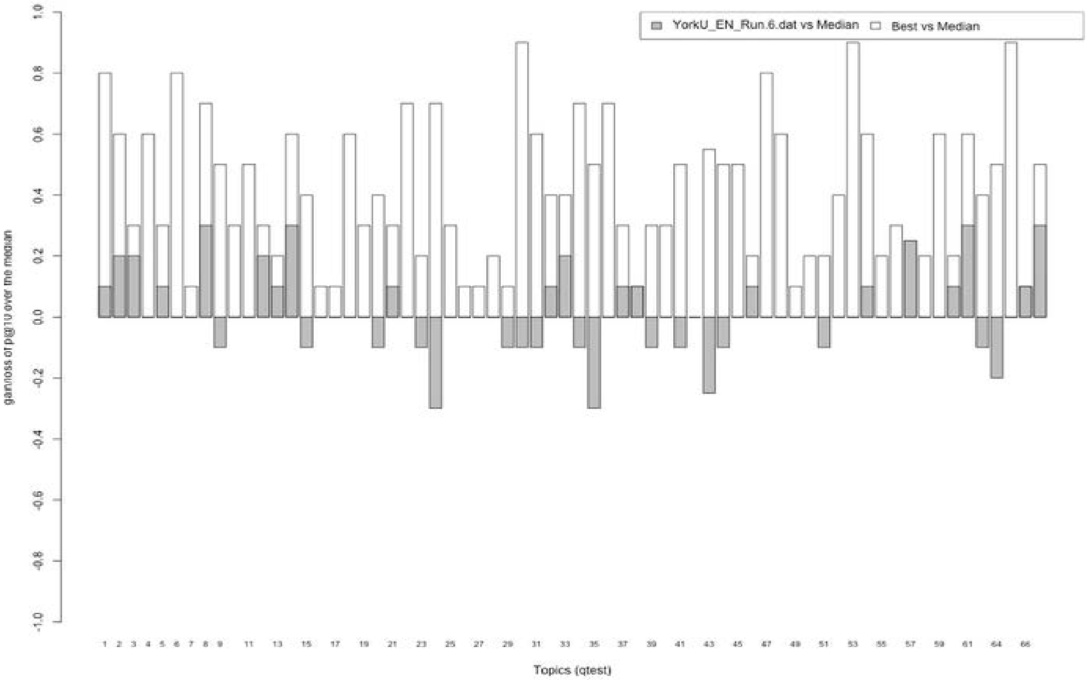
YorkU_EN_Run.4.dat



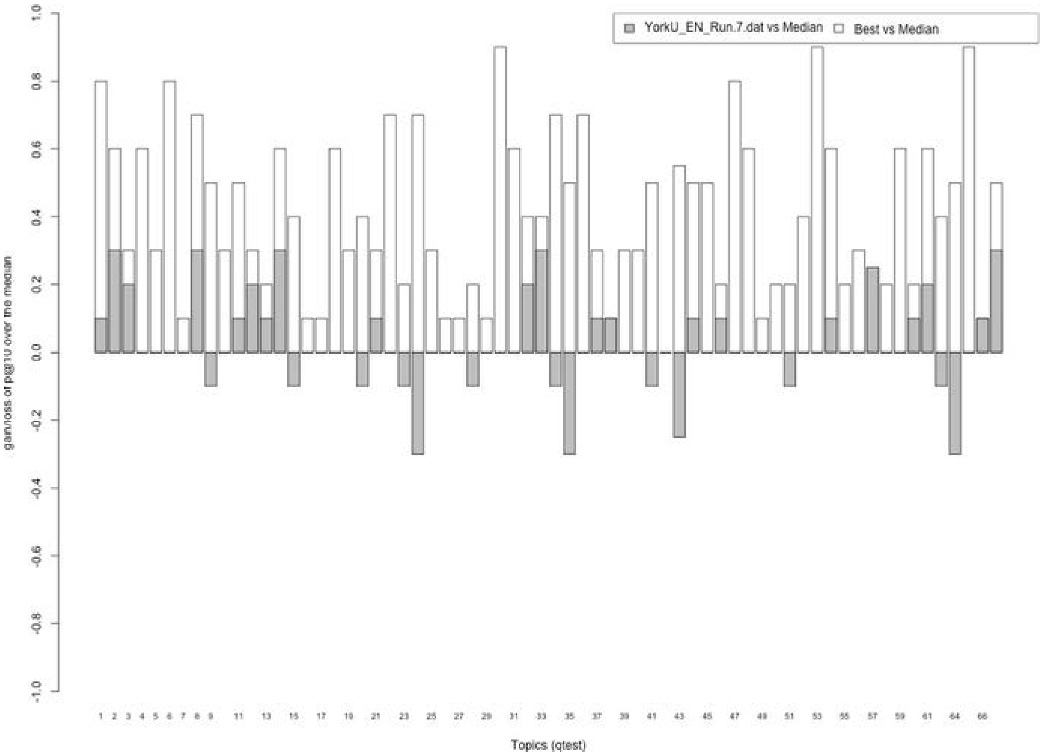
YorkU_EN_Run.5.dat



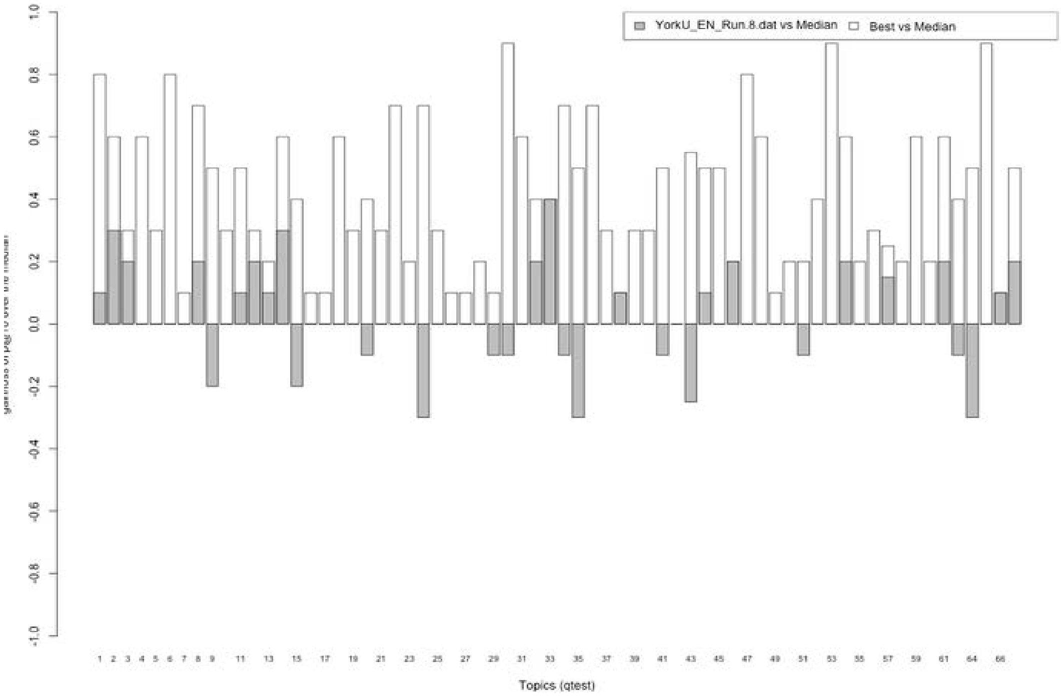
YorkU_EN_Run.6.dat



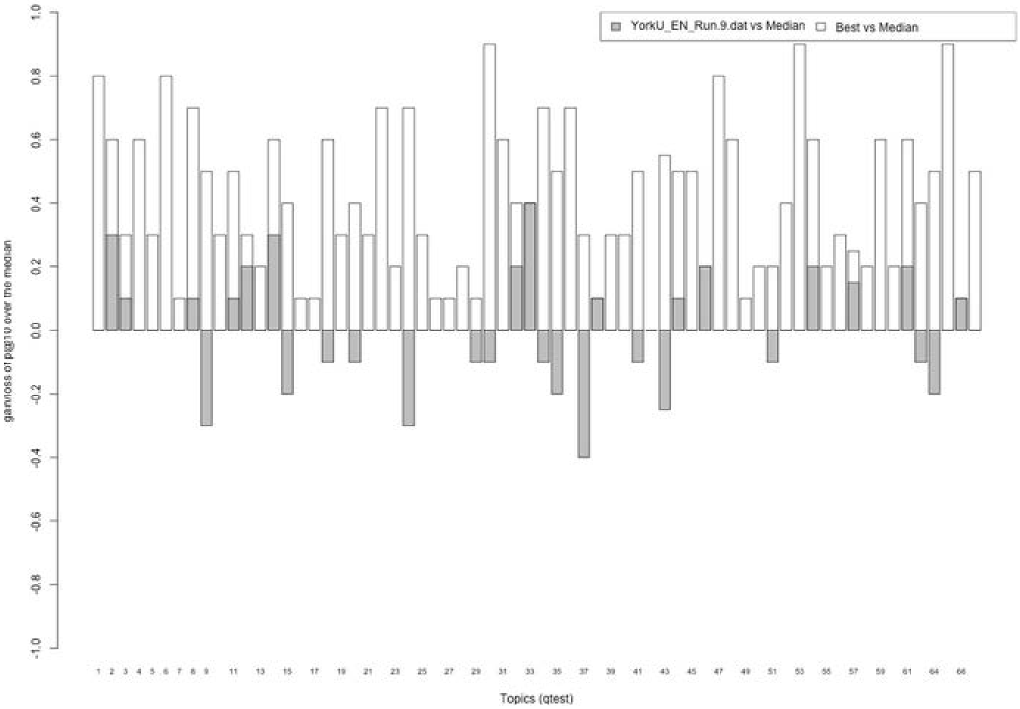
YorkU_EN_Run.7.dat



YorkU_EN_Run.8.dat



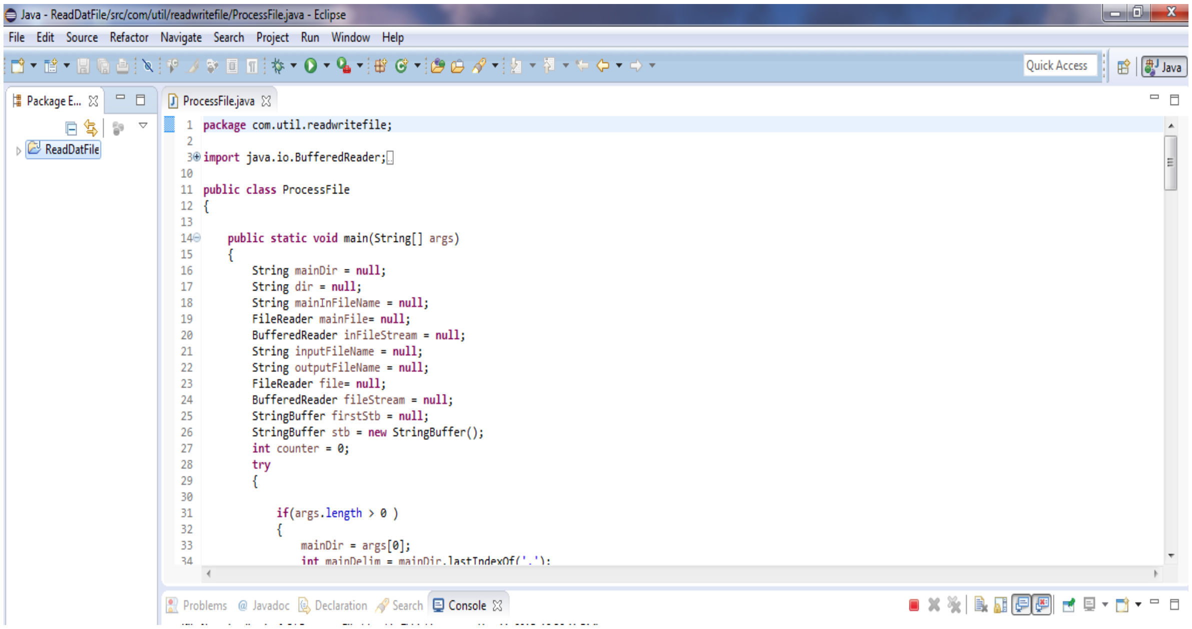
YorkU_EN_Run.9.dat



Appendix E:

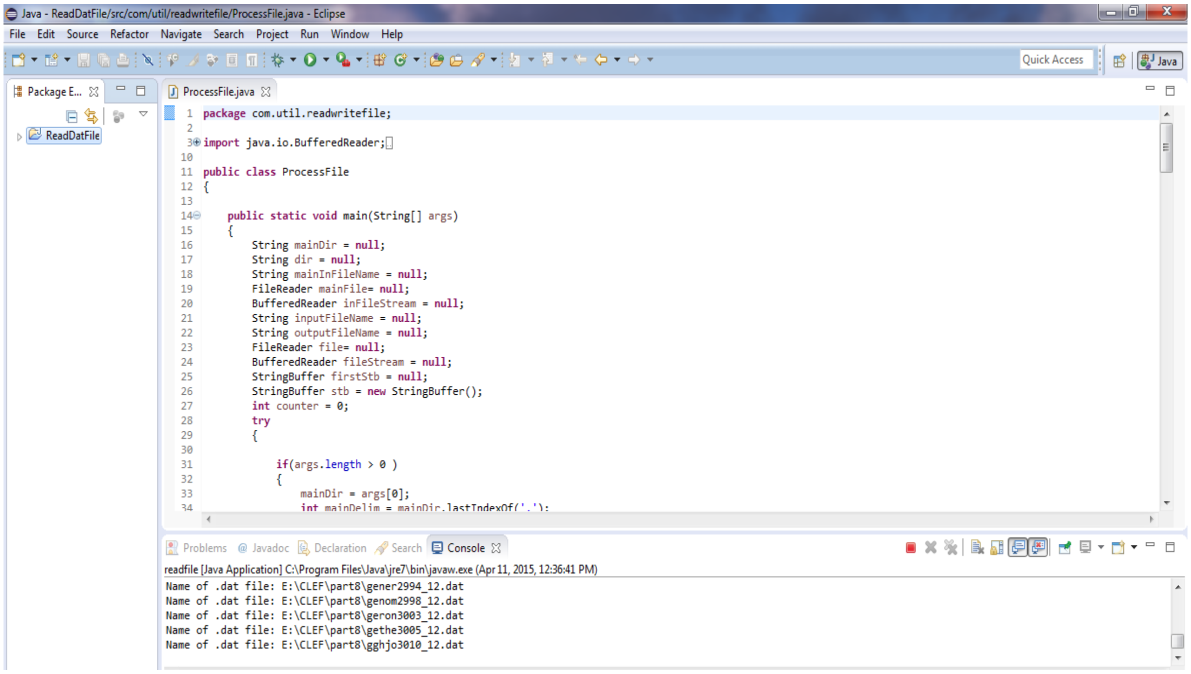
Screenshot of my task:

My first step using Eclipse to convert the data set in a format readable by Terrier:


Java - ReadDatFile/src/com/util/readwritefile/ProcessFile.java - Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help
Quick Access
Java
Package E...
ReadDatFile
ProcessFile.java
Problems @ Javadoc Declaration Search Console

```
package com.util.readwritefile;

import java.io.BufferedReader;

public class ProcessFile
{

    public static void main(String[] args)
    {
        String mainDir = null;
        String dir = null;
        String mainInFileName = null;
        FileReader mainFile= null;
        BufferedReader inFileStream = null;
        String inputFileName = null;
        String outputFileName = null;
        FileReader file= null;
        BufferedReader fileStream = null;
        StringBuffer firstStb = null;
        StringBuffer stb = new StringBuffer();
        int counter = 0;
        try
        {

            if(args.length > 0 )
            {
                mainDir = args[0];
                int mainDelim = mainDir.lastIndexOf('.');
```

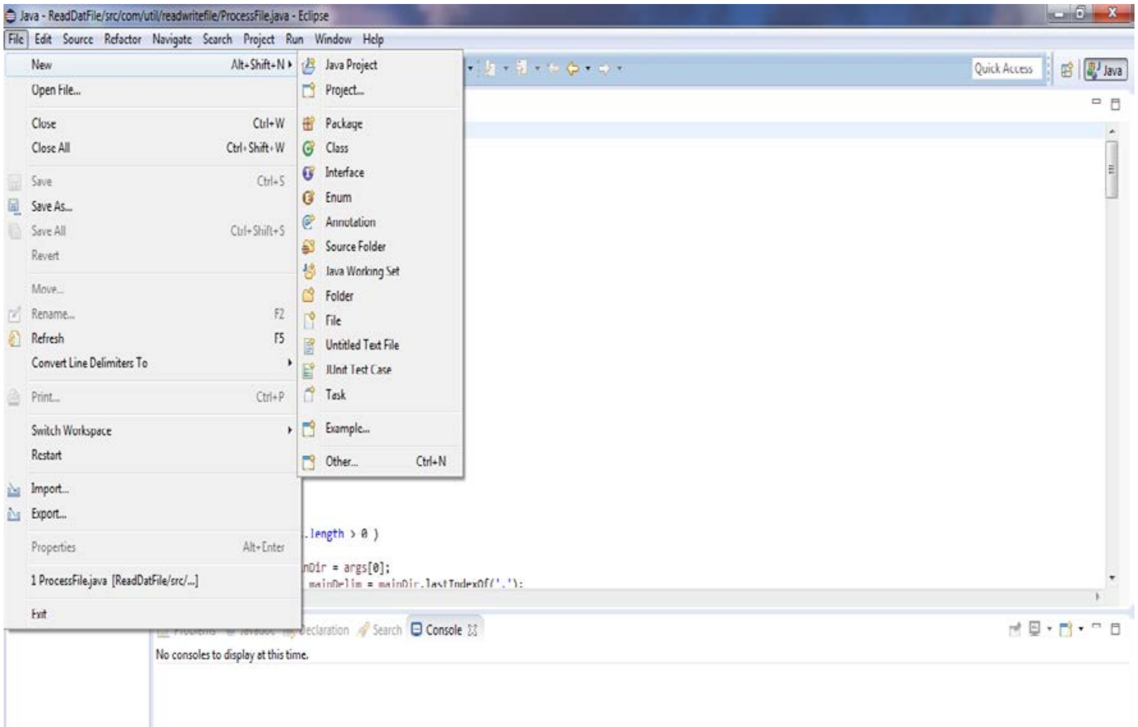
The program starts running and converting the format of the dataset:


Java - ReadDatFile/src/com/util/readwritefile/ProcessFile.java - Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help
Quick Access
Java
Package E...
ReadDatFile
ProcessFile.java
Problems @ Javadoc Declaration Search Console

```
package com.util.readwritefile;

import java.io.BufferedReader;

public class ProcessFile
{

    public static void main(String[] args)
    {
        String mainDir = null;
        String dir = null;
        String mainInFileName = null;
        FileReader mainFile= null;
        BufferedReader inFileStream = null;
        String inputFileName = null;
        String outputFileName = null;
        FileReader file= null;
        BufferedReader fileStream = null;
        StringBuffer firstStb = null;
        StringBuffer stb = new StringBuffer();
        int counter = 0;
        try
        {

            if(args.length > 0 )
            {
                mainDir = args[0];
                int mainDelim = mainDir.lastIndexOf('.');
```

```
readfile [Java Application] C:\Program Files\Java\jre7\bin\javaw.exe (Apr 11, 2015, 12:36:41 PM)
Name of .dat file: E:\CLEF\part8\gener2994_12.dat
Name of .dat file: E:\CLEF\part8\genom2998_12.dat
Name of .dat file: E:\CLEF\part8\geron3003_12.dat
Name of .dat file: E:\CLEF\part8\gethe3005_12.dat
Name of .dat file: E:\CLEF\part8\gghjo3010_12.dat
```

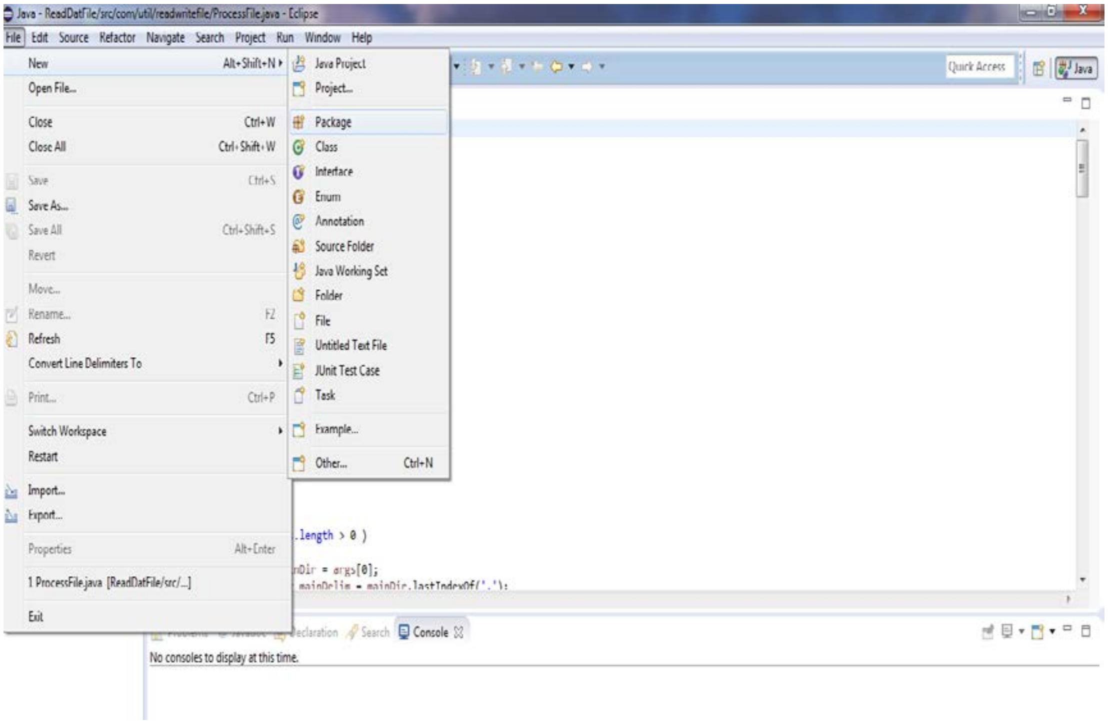
First we open Eclipse and create a new java project:



```
.length > 0 )
nDir = args[0];
 mainDelim = mainDir.lastIndexOf('.');
```

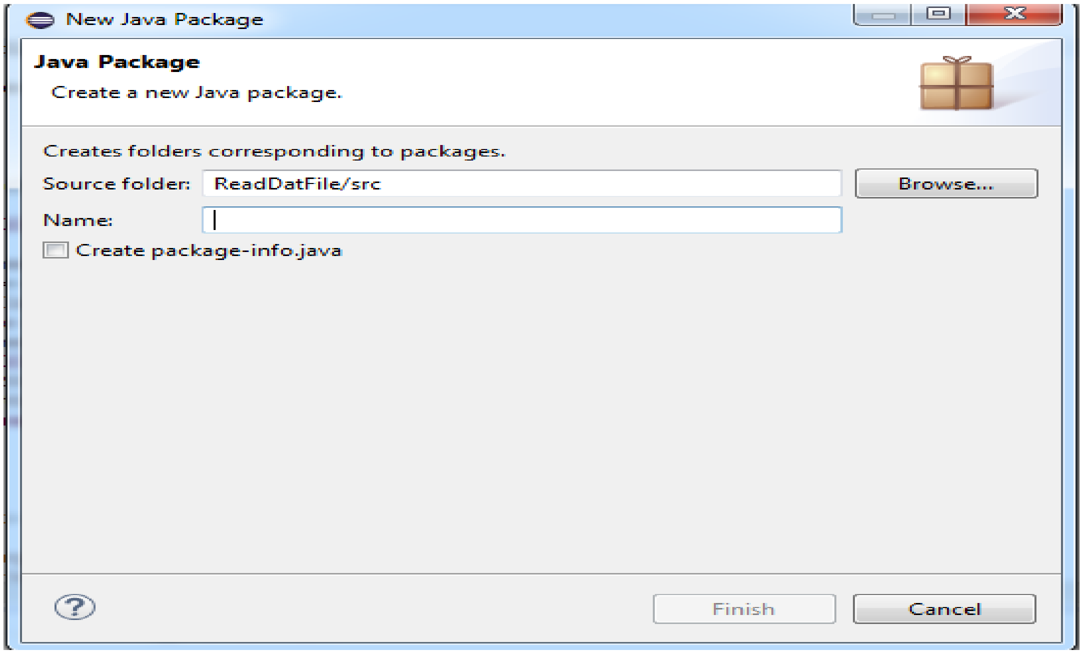
No consoles to display at this time.

Now we create the package:



```
.length > 0 )
nDir = args[0];
 mainDelim = mainDir.lastIndexOf('.');
```

No consoles to display at this time.



New Java Package

Java Package

Create a new Java package.

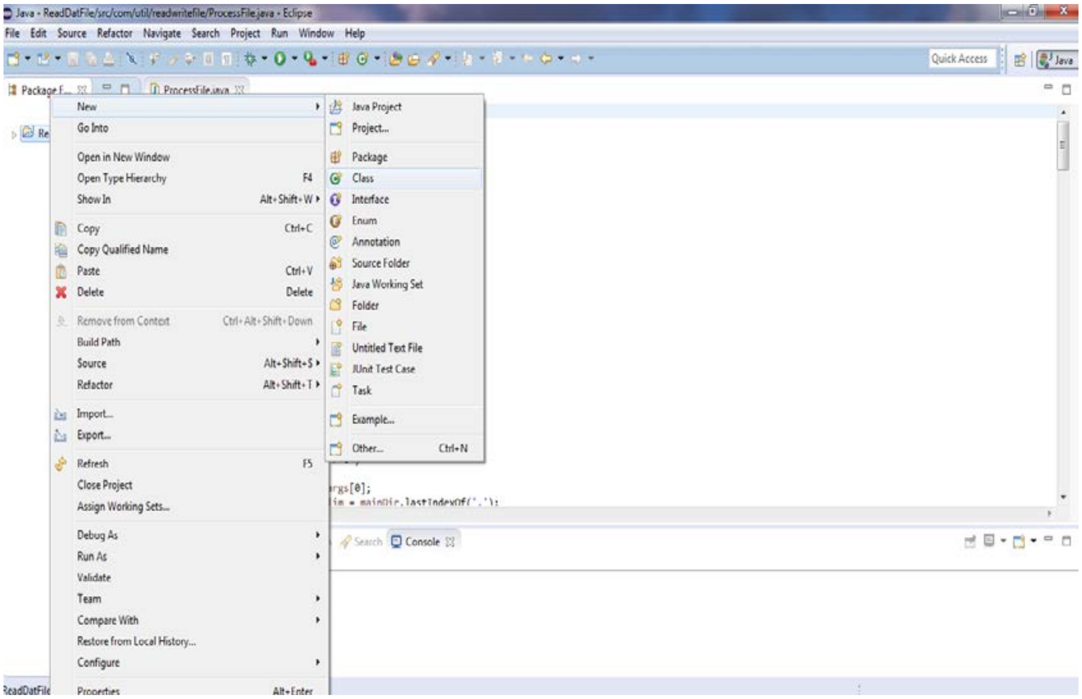
Creates folders corresponding to packages.

Source folder: ReadDatFile/src

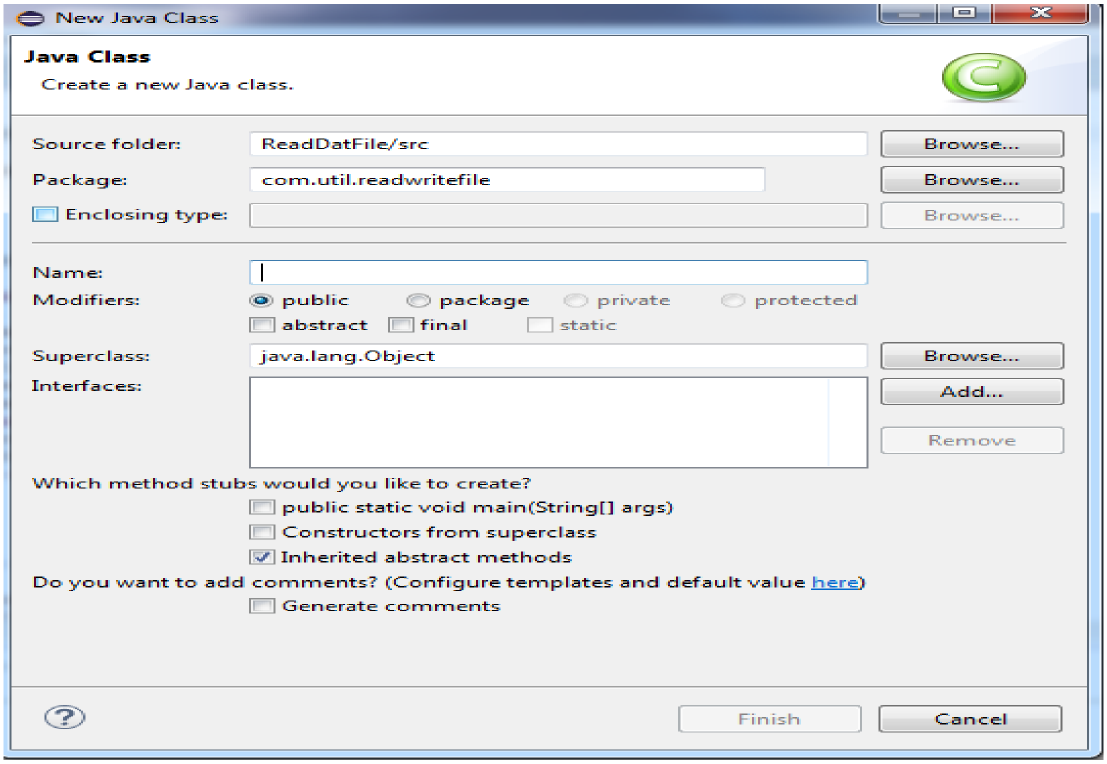
Name:

□ Create package-info.java

After the package is created we specify the class name:



Java - ReadDatFile/src/com/util/readwritefile/ProcessFile.java - Eclipse



Java Class

Create a new Java class.

Source folder: ReadDatFile/src

Package: com.util.readwritefile

□ Enclosing type:

Name:

Modifiers: ● public ○ package ○ private ○ protected

□ abstract □ final □ static

Superclass: java.lang.Object

Interfaces:

Which method stubs would you like to create?

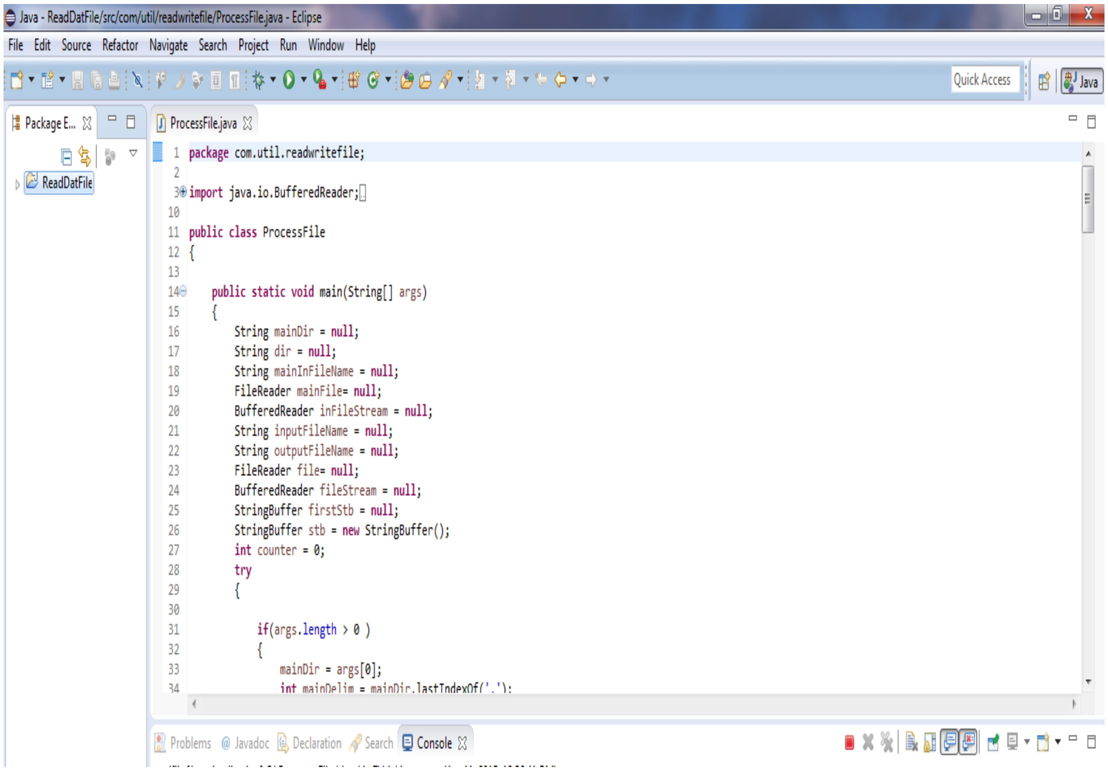
□ public static void main(String[] args)

□ Constructors from superclass

☑ Inherited abstract methods

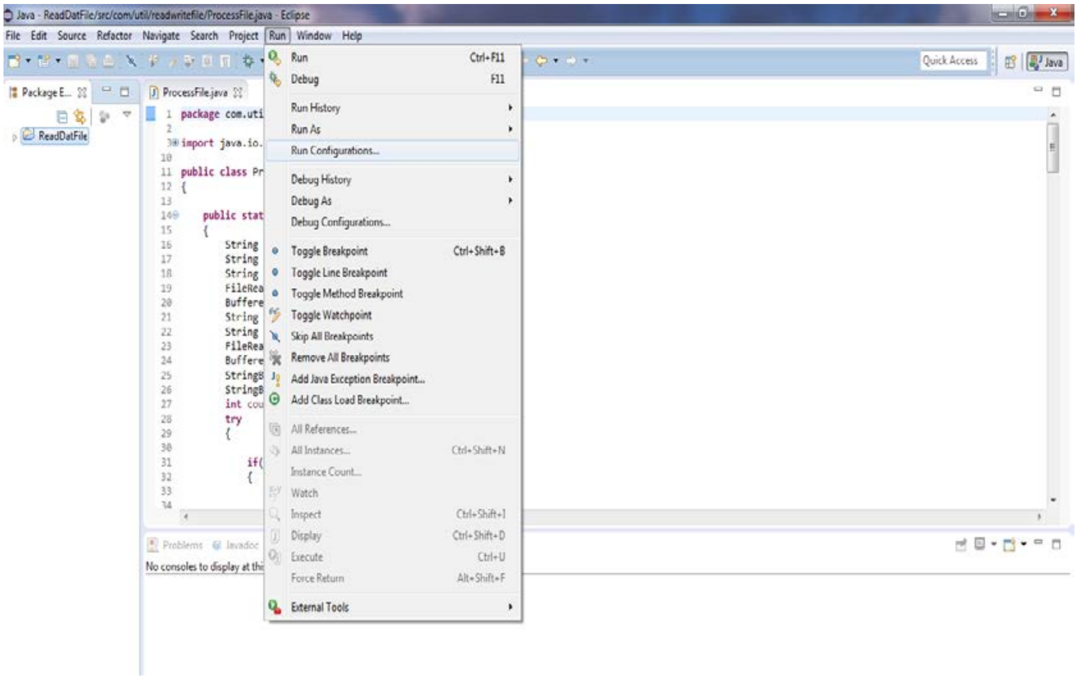
Do you want to add comments? (Configure templates and default value here)

□ Generate comments



```
package com.util.readwritefile;

import java.io.BufferedReader;

public class ProcessFile
{

    public static void main(String[] args)
    {
        String mainDir = null;
        String dir = null;
        String mainInFileName = null;
        FileReader mainFile= null;
        BufferedReader inFileStream = null;
        String inputFileName = null;
        String outputFileName = null;
        FileReader file= null;
        BufferedReader fileStream = null;
        StringBuffer firstStb = null;
        StringBuffer stb = new StringBuffer();
        int counter = 0;
        try
        {

            if(args.length > 0 )
            {
                mainDir = args[0];
                int mainDelim = mainDir.lastIndexOf('.');
```

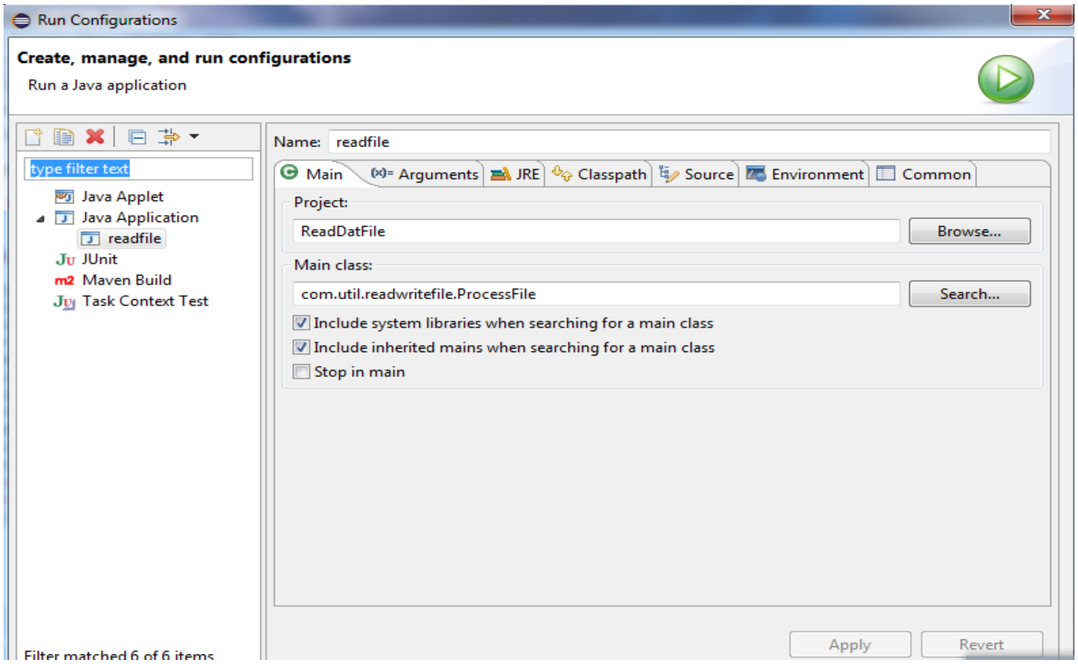
To run the program we hit Run and click on Run Configurations:



Java - ReadDatFile/src/com/util/readwritefile/ProcessFile.java - Eclipse

```
package com.uti

import java.io.

public class Pr
{

    public stat
    {
        String
        String
        String
        FileRea
        Buffere
        String
        String
        FileRea
        Buffere
        StringB
        StringB
        int cou
        try
        {

            if(
            {
```

No consoles to display at thi



Run Configurations

Create, manage, and run configurations

Run a Java application

type filter text

- Java Applet
- Java Application
  - readfile
- JUnit
- Maven Build
- Task Context Test

Filter matched 6 of 6 items

Name: readfile

Main | Arguments | JRE | Classpath | Source | Environment | Common

Project:

ReadDatFile [Browse...]

Main class:

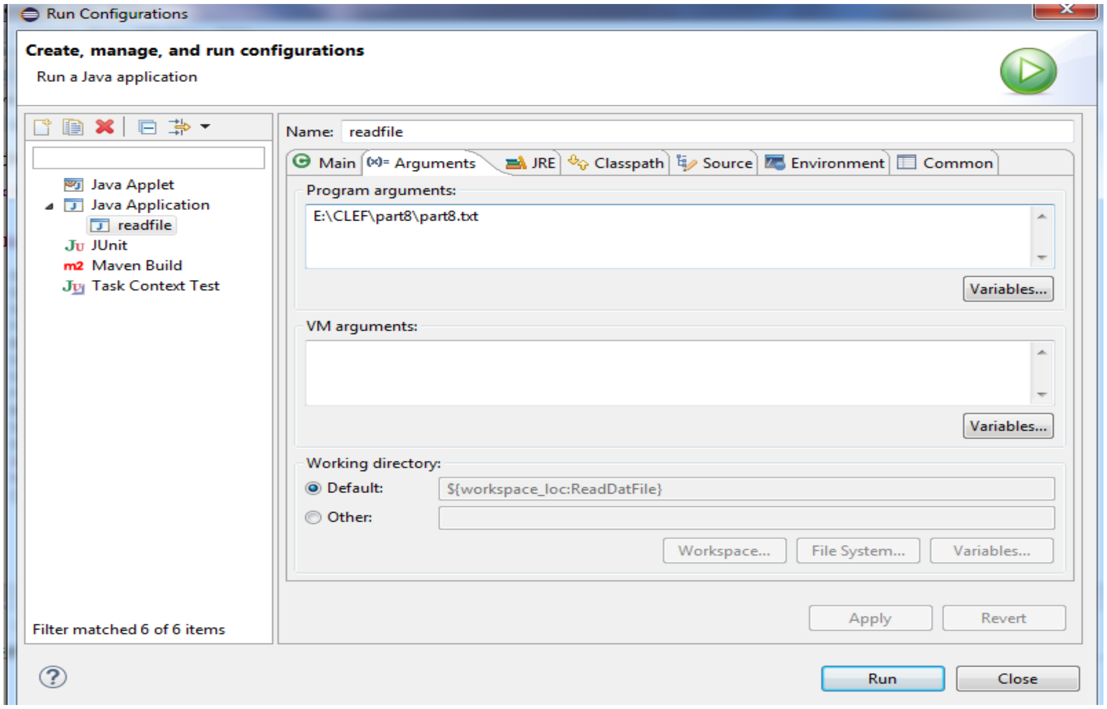
com.util.readwritefile.ProcessFile [Search...]

☑ Include system libraries when searching for a main class

☑ Include inherited mains when searching for a main class

☐ Stop in main

In the Arguments path we specify the path to the txt file and hit Apply then hit Run:



Run Configurations

Create, manage, and run configurations

Run a Java application

Name: readfile

Program arguments:

E:\CLEF\part8\part8.txt

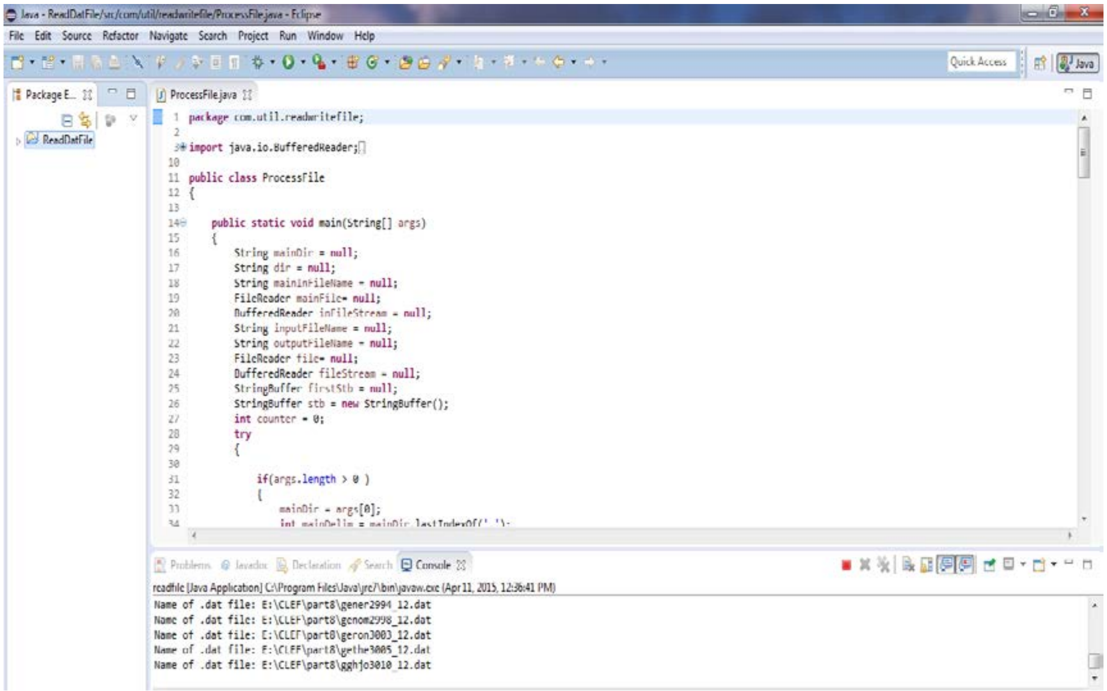
VM arguments:

Working directory:

● Default: ${workspace_loc:ReadDatFile}

○ Other:

Filter matched 6 of 6 items



Java - ReadDatFile/src/com/util/readwritefile/ProcessFile.java - Eclipse

ProcessFile.java

```
package com.util.readwritefile;

import java.io.BufferedReader;

public class ProcessFile
{

    public static void main(String[] args)
    {
        String mainDir = null;
        String dir = null;
        String mainInFileName = null;
        FileReader mainFile= null;
        BufferedReader inFileStream = null;
        String inputFileName = null;
        String outputFileName = null;
        FileReader file= null;
        BufferedReader fileStream = null;
        StringBuffer firstStb = null;
        StringBuffer stb = new StringBuffer();
        int counter = 0;
        try
        {

            if(args.length > 0 )
            {
                mainDir = args[0];
                int mainDelim = mainDir.lastIndexOf('.');
```

Console

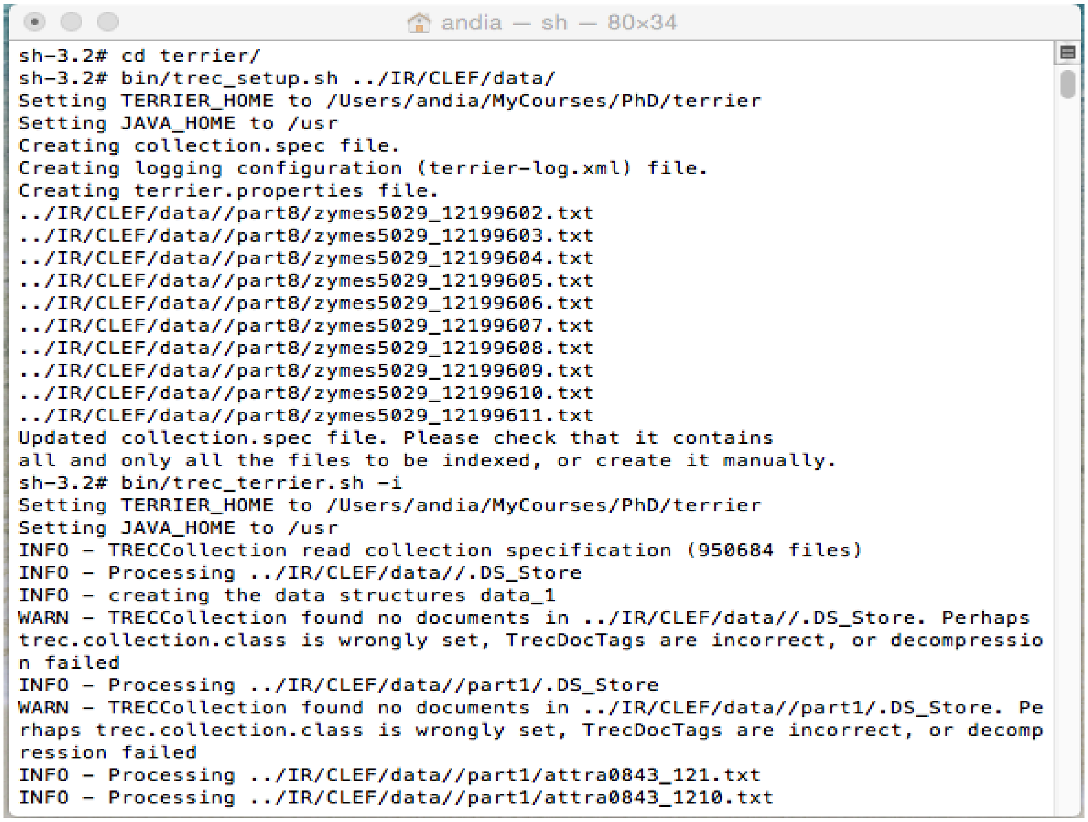
```
readfile [Java Application] C:\Program Files\Java\jre7\bin\javaw.exe (Apr 11, 2015, 12:36:41 PM)
Name of .dat file: E:\CLEF\part8\gener2994_12.dat
Name of .dat file: E:\CLEF\part8\genom2998_12.dat
Name of .dat file: E:\CLEF\part8\geron3003_12.dat
Name of .dat file: E:\CLEF\part8\gethe3005_12.dat
Name of .dat file: E:\CLEF\part8\gghjo3010_12.dat
```

My second step after the files are converted used the following commands to index the datasets:

cd /terrier/

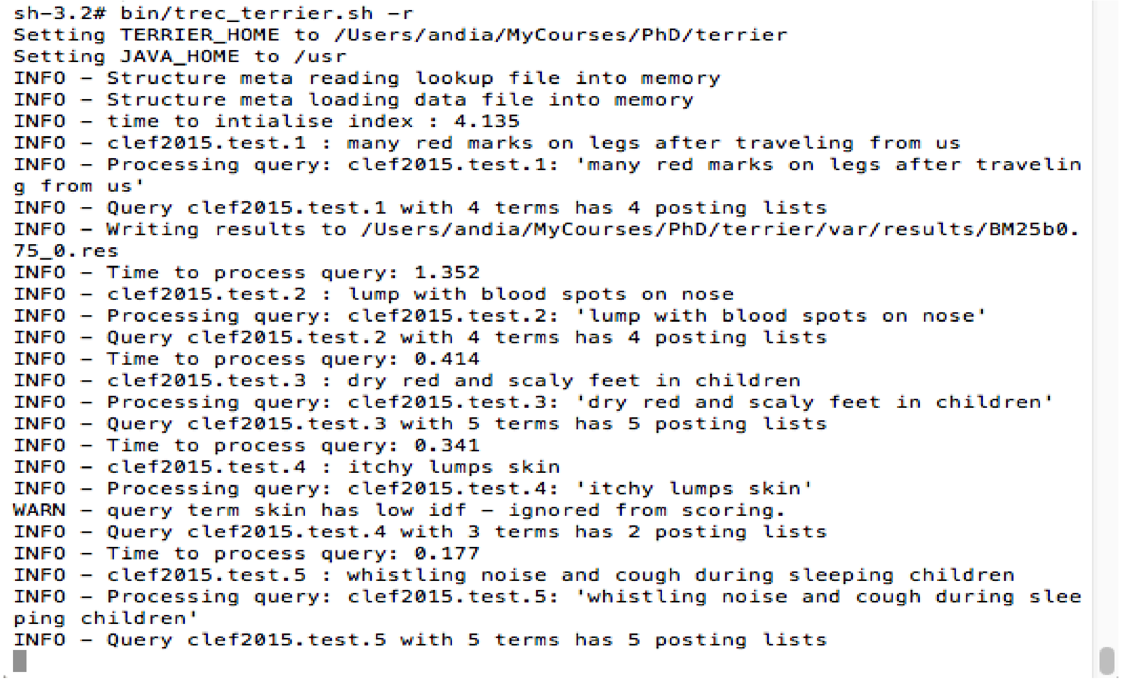
sudo bin/trec_setup.sh ../IR/CLEF/data/

sudo bin/trec_terrier.sh –i

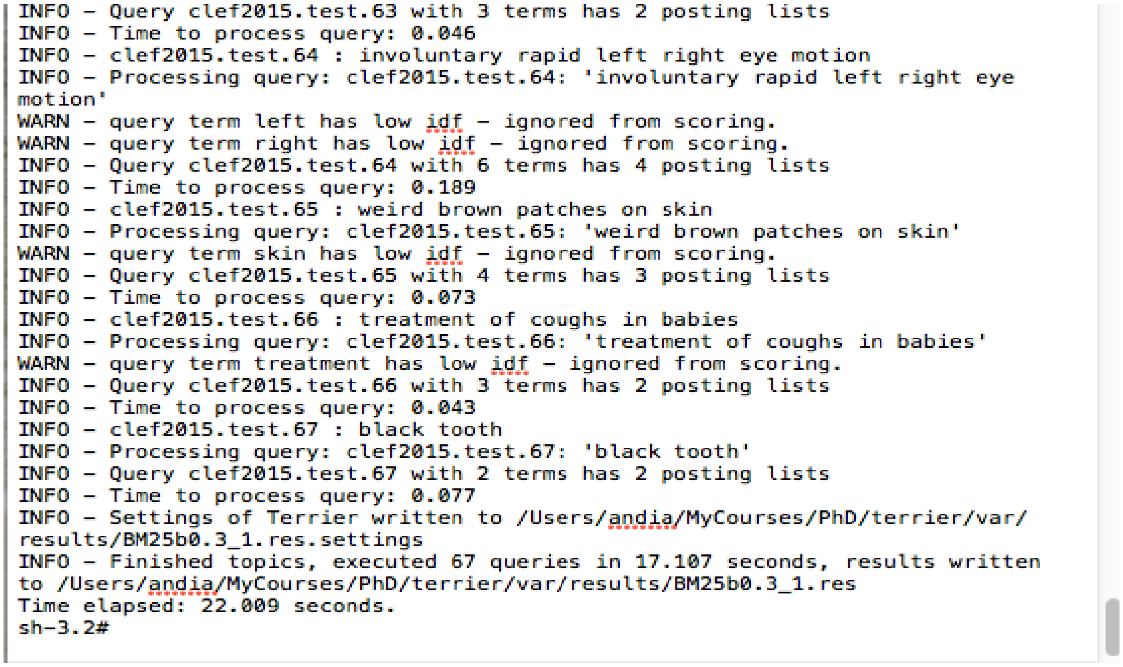


```
sh-3.2# cd terrier/
sh-3.2# bin/trec_setup.sh ../IR/CLEF/data/
Setting TERRIER_HOME to /Users/andia/MyCourses/PhD/terrier
Setting JAVA_HOME to /usr
Creating collection.spec file.
Creating logging configuration (terrier-log.xml) file.
Creating terrier.properties file.
../IR/CLEF/data//part8/zymes5029_12199602.txt
../IR/CLEF/data//part8/zymes5029_12199603.txt
../IR/CLEF/data//part8/zymes5029_12199604.txt
../IR/CLEF/data//part8/zymes5029_12199605.txt
../IR/CLEF/data//part8/zymes5029_12199606.txt
../IR/CLEF/data//part8/zymes5029_12199607.txt
../IR/CLEF/data//part8/zymes5029_12199608.txt
../IR/CLEF/data//part8/zymes5029_12199609.txt
../IR/CLEF/data//part8/zymes5029_12199610.txt
../IR/CLEF/data//part8/zymes5029_12199611.txt
Updated collection.spec file. Please check that it contains
all and only all the files to be indexed, or create it manually.
sh-3.2# bin/trec_terrier.sh -i
Setting TERRIER_HOME to /Users/andia/MyCourses/PhD/terrier
Setting JAVA_HOME to /usr
INFO - TRECCollection read collection specification (950684 files)
INFO - Processing ../IR/CLEF/data//.DS_Store
INFO - creating the data structures data_1
WARN - TRECCollection found no documents in ../IR/CLEF/data//.DS_Store. Perhaps
trec.collection.class is wrongly set, TrecDocTags are incorrect, or decompressio
n failed
INFO - Processing ../IR/CLEF/data//part1/.DS_Store
WARN - TRECCollection found no documents in ../IR/CLEF/data//part1/.DS_Store. Pe
rhaps trec.collection.class is wrongly set, TrecDocTags are incorrect, or decomp
ression failed
INFO - Processing ../IR/CLEF/data//part1/attra0843_121.txt
INFO - Processing ../IR/CLEF/data//part1/attra0843_1210.txt
```

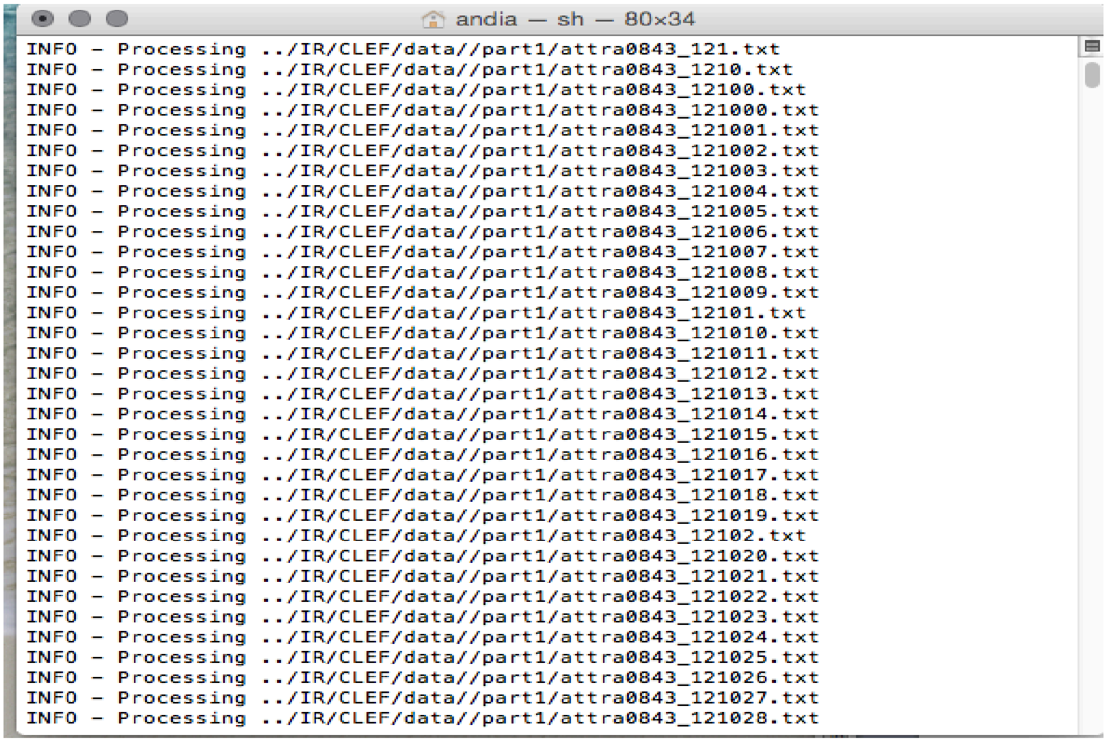
sudo bin/trec_terrier.sh –r



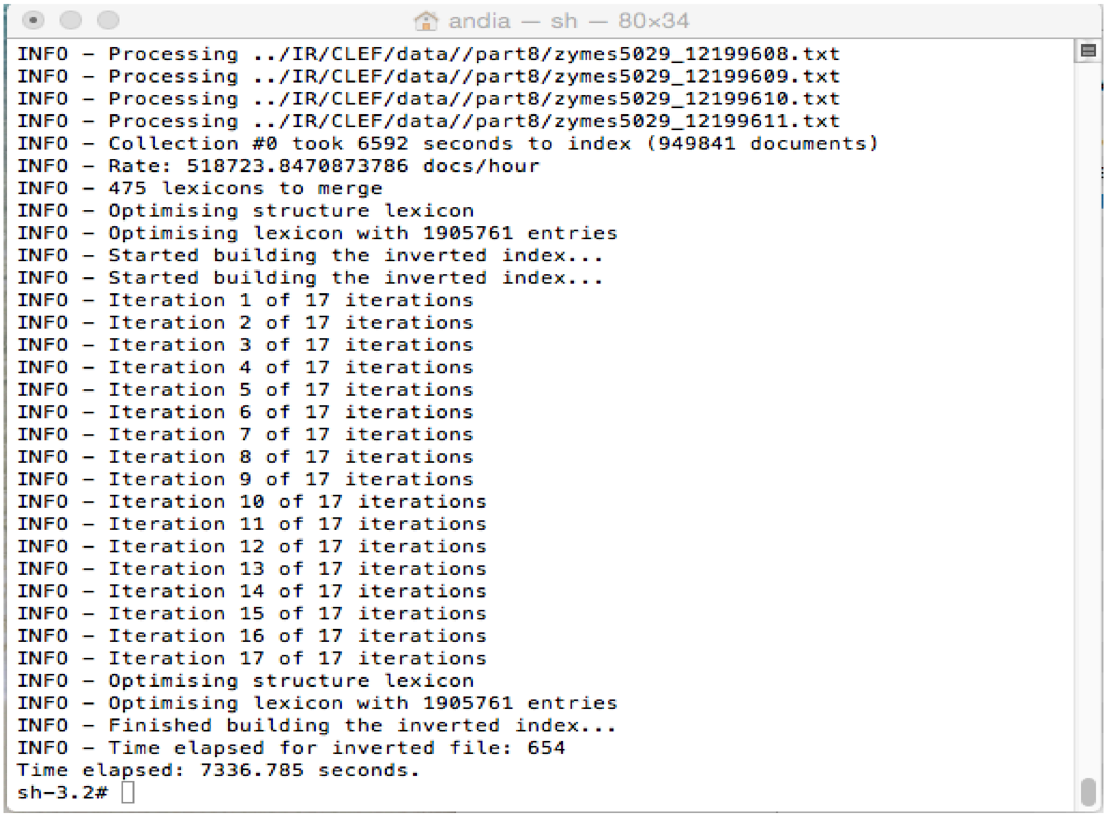
```
sh-3.2# bin/trec_terrier.sh -r
Setting TERRIER_HOME to /Users/andia/MyCourses/PhD/terrier
Setting JAVA_HOME to /usr
INFO - Structure meta reading lookup file into memory
INFO - Structure meta loading data file into memory
INFO - time to intialise index : 4.135
INFO - clef2015.test.1 : many red marks on legs after traveling from us
INFO - Processing query: clef2015.test.1: 'many red marks on legs after travelin
g from us'
INFO - Query clef2015.test.1 with 4 terms has 4 posting lists
INFO - Writing results to /Users/andia/MyCourses/PhD/terrier/var/results/BM25b0.
75_0.res
INFO - Time to process query: 1.352
INFO - clef2015.test.2 : lump with blood spots on nose
INFO - Processing query: clef2015.test.2: 'lump with blood spots on nose'
INFO - Query clef2015.test.2 with 4 terms has 4 posting lists
INFO - Time to process query: 0.414
INFO - clef2015.test.3 : dry red and scaly feet in children
INFO - Processing query: clef2015.test.3: 'dry red and scaly feet in children'
INFO - Query clef2015.test.3 with 5 terms has 5 posting lists
INFO - Time to process query: 0.341
INFO - clef2015.test.4 : itchy lumps skin
INFO - Processing query: clef2015.test.4: 'itchy lumps skin'
WARN - query term skin has low idf - ignored from scoring.
INFO - Query clef2015.test.4 with 3 terms has 2 posting lists
INFO - Time to process query: 0.177
INFO - clef2015.test.5 : whistling noise and cough during sleeping children
INFO - Processing query: clef2015.test.5: 'whistling noise and cough during slee
ping children'
INFO - Query clef2015.test.5 with 5 terms has 5 posting lists
```



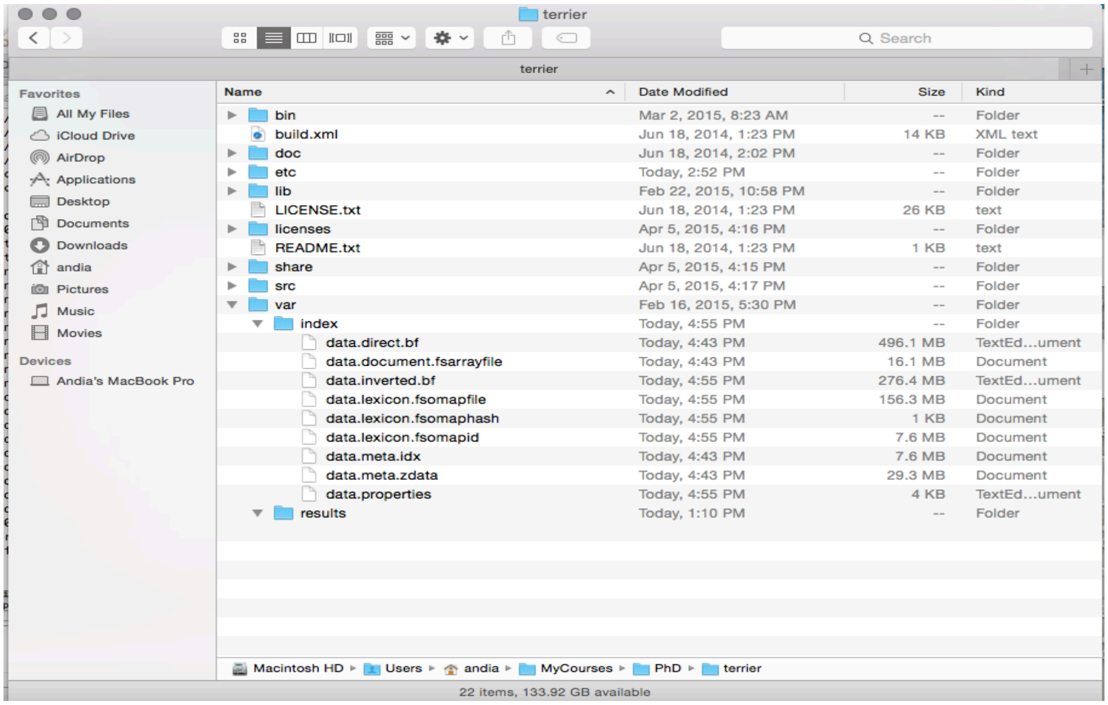
```
INFO - Query clef2015.test.63 with 3 terms has 2 posting lists
INFO - Time to process query: 0.046
INFO - clef2015.test.64 : involuntary rapid left right eye motion
INFO - Processing query: clef2015.test.64: 'involuntary rapid left right eye
motion'
WARN - query term left has low idf - ignored from scoring.
WARN - query term right has low idf - ignored from scoring.
INFO - Query clef2015.test.64 with 6 terms has 4 posting lists
INFO - Time to process query: 0.189
INFO - clef2015.test.65 : weird brown patches on skin
INFO - Processing query: clef2015.test.65: 'weird brown patches on skin'
WARN - query term skin has low idf - ignored from scoring.
INFO - Query clef2015.test.65 with 4 terms has 3 posting lists
INFO - Time to process query: 0.073
INFO - clef2015.test.66 : treatment of coughs in babies
INFO - Processing query: clef2015.test.66: 'treatment of coughs in babies'
WARN - query term treatment has low idf - ignored from scoring.
INFO - Query clef2015.test.66 with 3 terms has 2 posting lists
INFO - Time to process query: 0.043
INFO - clef2015.test.67 : black tooth
INFO - Processing query: clef2015.test.67: 'black tooth'
INFO - Query clef2015.test.67 with 2 terms has 2 posting lists
INFO - Time to process query: 0.077
INFO - Settings of Terrier written to /Users/andia/MyCourses/PhD/terrier/var/
results/BM25b0.3_1.res.settings
INFO - Finished topics, executed 67 queries in 17.107 seconds, results written
to /Users/andia/MyCourses/PhD/terrier/var/results/BM25b0.3_1.res
Time elapsed: 22.009 seconds.
sh-3.2#
```



```
INFO - Processing ../IR/CLEF/data//part1/attra0843_121.txt
INFO - Processing ../IR/CLEF/data//part1/attra0843_1210.txt
INFO - Processing ../IR/CLEF/data//part1/attra0843_12100.txt
INFO - Processing ../IR/CLEF/data//part1/attra0843_121000.txt
INFO - Processing ../IR/CLEF/data//part1/attra0843_121001.txt
INFO - Processing ../IR/CLEF/data//part1/attra0843_121002.txt
INFO - Processing ../IR/CLEF/data//part1/attra0843_121003.txt
INFO - Processing ../IR/CLEF/data//part1/attra0843_121004.txt
INFO - Processing ../IR/CLEF/data//part1/attra0843_121005.txt
INFO - Processing ../IR/CLEF/data//part1/attra0843_121006.txt
INFO - Processing ../IR/CLEF/data//part1/attra0843_121007.txt
INFO - Processing ../IR/CLEF/data//part1/attra0843_121008.txt
INFO - Processing ../IR/CLEF/data//part1/attra0843_121009.txt
INFO - Processing ../IR/CLEF/data//part1/attra0843_12101.txt
INFO - Processing ../IR/CLEF/data//part1/attra0843_121010.txt
INFO - Processing ../IR/CLEF/data//part1/attra0843_121011.txt
INFO - Processing ../IR/CLEF/data//part1/attra0843_121012.txt
INFO - Processing ../IR/CLEF/data//part1/attra0843_121013.txt
INFO - Processing ../IR/CLEF/data//part1/attra0843_121014.txt
INFO - Processing ../IR/CLEF/data//part1/attra0843_121015.txt
INFO - Processing ../IR/CLEF/data//part1/attra0843_121016.txt
INFO - Processing ../IR/CLEF/data//part1/attra0843_121017.txt
INFO - Processing ../IR/CLEF/data//part1/attra0843_121018.txt
INFO - Processing ../IR/CLEF/data//part1/attra0843_121019.txt
INFO - Processing ../IR/CLEF/data//part1/attra0843_12102.txt
INFO - Processing ../IR/CLEF/data//part1/attra0843_121020.txt
INFO - Processing ../IR/CLEF/data//part1/attra0843_121021.txt
INFO - Processing ../IR/CLEF/data//part1/attra0843_121022.txt
INFO - Processing ../IR/CLEF/data//part1/attra0843_121023.txt
INFO - Processing ../IR/CLEF/data//part1/attra0843_121024.txt
INFO - Processing ../IR/CLEF/data//part1/attra0843_121025.txt
INFO - Processing ../IR/CLEF/data//part1/attra0843_121026.txt
INFO - Processing ../IR/CLEF/data//part1/attra0843_121027.txt
INFO - Processing ../IR/CLEF/data//part1/attra0843_121028.txt
```



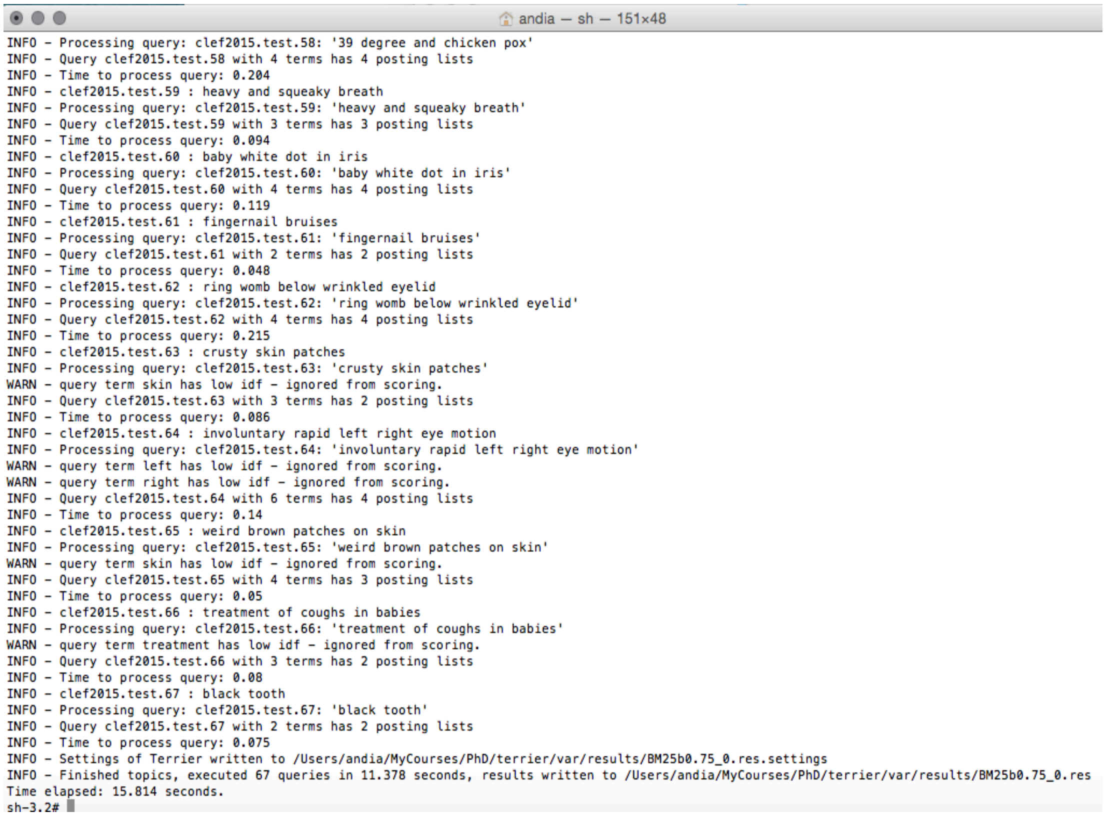
```
INFO - Processing ../IR/CLEF/data//part8/zymes5029_12199608.txt
INFO - Processing ../IR/CLEF/data//part8/zymes5029_12199609.txt
INFO - Processing ../IR/CLEF/data//part8/zymes5029_12199610.txt
INFO - Processing ../IR/CLEF/data//part8/zymes5029_12199611.txt
INFO - Collection #0 took 6592 seconds to index (949841 documents)
INFO - Rate: 518723.8470873786 docs/hour
INFO - 475 lexicons to merge
INFO - Optimising structure lexicon
INFO - Optimising lexicon with 1905761 entries
INFO - Started building the inverted index...
INFO - Started building the inverted index...
INFO - Iteration 1 of 17 iterations
INFO - Iteration 2 of 17 iterations
INFO - Iteration 3 of 17 iterations
INFO - Iteration 4 of 17 iterations
INFO - Iteration 5 of 17 iterations
INFO - Iteration 6 of 17 iterations
INFO - Iteration 7 of 17 iterations
INFO - Iteration 8 of 17 iterations
INFO - Iteration 9 of 17 iterations
INFO - Iteration 10 of 17 iterations
INFO - Iteration 11 of 17 iterations
INFO - Iteration 12 of 17 iterations
INFO - Iteration 13 of 17 iterations
INFO - Iteration 14 of 17 iterations
INFO - Iteration 15 of 17 iterations
INFO - Iteration 16 of 17 iterations
INFO - Iteration 17 of 17 iterations
INFO - Optimising structure lexicon
INFO - Optimising lexicon with 1905761 entries
INFO - Finished building the inverted index...
INFO - Time elapsed for inverted file: 654
Time elapsed: 7336.785 seconds.
sh-3.2# 
```



Name	Date Modified	Size	Kind
▶ bin	Mar 2, 2015, 8:23 AM	--	Folder
build.xml	Jun 18, 2014, 1:23 PM	14 KB	XML text
▶ doc	Jun 18, 2014, 2:02 PM	--	Folder
▶ etc	Today, 2:52 PM	--	Folder
▶ lib	Feb 22, 2015, 10:58 PM	--	Folder
LICENSE.txt	Jun 18, 2014, 1:23 PM	26 KB	text
▶ licenses	Apr 5, 2015, 4:16 PM	--	Folder
README.txt	Jun 18, 2014, 1:23 PM	1 KB	text
▶ share	Apr 5, 2015, 4:15 PM	--	Folder
▶ src	Apr 5, 2015, 4:17 PM	--	Folder
▼ var	Feb 16, 2015, 5:30 PM	--	Folder
▼ index	Today, 4:55 PM	--	Folder
data.direct.bf	Today, 4:43 PM	496.1 MB	TextEd...ument
data.document.fsarrayfile	Today, 4:43 PM	16.1 MB	Document
data.inverted.bf	Today, 4:55 PM	276.4 MB	TextEd...ument
data.lexicon.fsomapfile	Today, 4:55 PM	156.3 MB	Document
data.lexicon.fsomaphash	Today, 4:55 PM	1 KB	Document
data.lexicon.fsomapid	Today, 4:55 PM	7.6 MB	Document
data.meta.idx	Today, 4:43 PM	7.6 MB	Document
data.meta.zdata	Today, 4:43 PM	29.3 MB	Document
data.properties	Today, 4:55 PM	4 KB	TextEd...ument
▼ results	Today, 1:10 PM	--	Folder

Macintosh HD ▸ Users ▸ andia ▸ MyCourses ▸ PhD ▸ terrier

22 items, 133.92 GB available



```
INFO - Processing query: clef2015.test.58: '39 degree and chicken pox'
INFO - Query clef2015.test.58 with 4 terms has 4 posting lists
INFO - Time to process query: 0.204
INFO - clef2015.test.59 : heavy and squeaky breath
INFO - Processing query: clef2015.test.59: 'heavy and squeaky breath'
INFO - Query clef2015.test.59 with 3 terms has 3 posting lists
INFO - Time to process query: 0.094
INFO - clef2015.test.60 : baby white dot in iris
INFO - Processing query: clef2015.test.60: 'baby white dot in iris'
INFO - Query clef2015.test.60 with 4 terms has 4 posting lists
INFO - Time to process query: 0.119
INFO - clef2015.test.61 : fingernail bruises
INFO - Processing query: clef2015.test.61: 'fingernail bruises'
INFO - Query clef2015.test.61 with 2 terms has 2 posting lists
INFO - Time to process query: 0.048
INFO - clef2015.test.62 : ring womb below wrinkled eyelid
INFO - Processing query: clef2015.test.62: 'ring womb below wrinkled eyelid'
INFO - Query clef2015.test.62 with 4 terms has 4 posting lists
INFO - Time to process query: 0.215
INFO - clef2015.test.63 : crusty skin patches
INFO - Processing query: clef2015.test.63: 'crusty skin patches'
WARN - query term skin has low idf - ignored from scoring.
INFO - Query clef2015.test.63 with 3 terms has 2 posting lists
INFO - Time to process query: 0.086
INFO - clef2015.test.64 : involuntary rapid left right eye motion
INFO - Processing query: clef2015.test.64: 'involuntary rapid left right eye motion'
WARN - query term left has low idf - ignored from scoring.
WARN - query term right has low idf - ignored from scoring.
INFO - Query clef2015.test.64 with 6 terms has 4 posting lists
INFO - Time to process query: 0.14
INFO - clef2015.test.65 : weird brown patches on skin
INFO - Processing query: clef2015.test.65: 'weird brown patches on skin'
WARN - query term skin has low idf - ignored from scoring.
INFO - Query clef2015.test.65 with 4 terms has 3 posting lists
INFO - Time to process query: 0.05
INFO - clef2015.test.66 : treatment of coughs in babies
INFO - Processing query: clef2015.test.66: 'treatment of coughs in babies'
WARN - query term treatment has low idf - ignored from scoring.
INFO - Query clef2015.test.66 with 3 terms has 2 posting lists
INFO - Time to process query: 0.08
INFO - clef2015.test.67 : black tooth
INFO - Processing query: clef2015.test.67: 'black tooth'
INFO - Query clef2015.test.67 with 2 terms has 2 posting lists
INFO - Time to process query: 0.075
INFO - Settings of Terrier written to /Users/andia/MyCourses/PhD/terrier/var/results/BM25b0.75_0.res.settings
INFO - Finished topics, executed 67 queries in 11.378 seconds, results written to /Users/andia/MyCourses/PhD/terrier/var/results/BM25b0.75_0.res
Time elapsed: 15.814 seconds.
sh-3.2# 
```