

Medical-Miner at TREC 2011 Medical Records Track

¹J.M. Córdoba, ¹M.J. Maña, ¹N.P. Cruz, ¹J. Mata, ²F. Aparicio,
²M. Buenaga, ³D. Glez-Peña, ³F. Fdez-Riverola

¹Universidad de Huelva

²Universidad Europea de Madrid

³Universidad de Vigo

Abstract

This paper presents work done at Medical Minner Project on the TREC-2011 Medical Records Track. The paper proposes four models for medical information retrieval based on Lucene index approach. Our retrieval engine used an Lucen Index scheme with traditional stopping and stemming, enhanced with entity recognition software on query terms. Our aim in this first competition is to set a broader project that involves the develop of a configurable Apache Lucene-based framework that allows the rapid development of medical search facilities. Results around the track median have been achieved. In this exploratory track, we think that these results are a good beginning and encourage us for future developments.

1 Introduction

Clinical databases have accumulated large quantities of information about patients and their medical conditions. Relationships and patterns within this data could provide new medical knowledge. Unfortunately, few methodologies have been developed and applied to discover this hidden knowledge. In the TREC framework, Medical Records track encourages to research on providing content-based access to the free-text fields of electronic medical records. Related with this same research area, the Medical Minner initiative¹ works for provide new useful tools in this environment, enabling the achievement of a double benefit: first, to provide specific and relevant mechanisms to a professional environment (biomedical), and secondly, obtaining innovative techniques and significant advances in research in the area of information retrieval, particularly on this domain, and based on a significant integration effort. In this context, this paper presents the models developed for our first participation in TREC, particularly in the Medical Records track.

¹<http://www.medicalminer.org>

This paper is organised as follows. In Section 2 we describe some of the ways have been used to represent the documents fields in reports processing, and describe the way the index used in our later experiments is built automatically from document collection. In Section 3 we cover the overall architecture of a indexing application and where Lucene fits. The document retrieval models described in Section 4. This section, also covers the further development and elaboration of the models. Data and results tables are given in Section 5 with key results summarised. Finally, in Section 6, some conclusions are provided.

2 Document Collection Preprocessing

Assuming the system has access to a large document collection as a knowledge resource for answering medical topics, this collection may need to be processed before querying, in order to transform it into a form which is appropriate for topic answering. Thus, in order to effective TREC medical visits retrieval, some issues need to be solved. First, reports need to be grouped into visits. Next, XML Reports must be adapted to be queried in a rapid and accurate way. Additionally, another type of additional processing can be performed. In our case, we used the ICD codes for better retrieval.

2.1 Visit Processing

As the Medical Records track will use the visit as the answer unit then is there need for additional processing in order to group the reports into visits. The Medical Records Track repository contains a simple ASCII table called the Report-to-Visit Mapping Key that specifies which reports belong to the same visit. This table is sorted per visit ID. Once the table is ordered, the visits reports can be processed sequentially. As a precaution, checking procedures have been implemented to control null visits, number of reports per visit, total visits, etc

2.2 XML Parsing

We need to quickly search a collection of XML documents, and, to do this, we need to create an index of terms keeping track of the context in which these terms appear. Our solution uses Apache Lucene and the Java SAX library in order to create an index of Lucene Document objects for the lowest level of granularity to search. In order to have a testing framework as complete as possible, all reports fields have been processed.

2.3 ICD9 Mining

The International Statistical Classification of Diseases and Related Health Problems (most commonly known by the abbreviation ICD) is a medical classification that provides codes to classify diseases and a wide variety of signs, symptoms,

abnormal findings, complaints, social circumstances, and external causes of injury or disease. The Collection reports includes fields with discharge diagnosis in ICD9 codes form. In our approach, codes are processed to include a textual description to gather additional information. Thus, when a ICD9 code is detected in report processing by SAX the code description is searched in a database and automatically added to the index.

3 Indexing

Model main component is a search engine based on Apache Lucene. Lucene is a powerful Java library that lets you easily add document retrieval to any application. In recent years Lucene has become exceptionally popular and is now the most widely used information retrieval library. What we end up with after running Lucene is a directory named index, which contains files used by Lucene to associate terms with documents. To accomplish this, a Lucene index was created with a specific analyzer model-dependent. An Analyzer takes a series of terms or tokens and creates the terms to be indexed. A unique kind of Lucene index has been used for all developed models, or in other words, all models share the same Lucene index.

Documents and fields are Lucene’s fundamental units of indexing and searching. A document is Lucene’s atomic unit of indexing and searching. It is a container that holds one or more fields, which in turn contain the “real” content. Each field has a name to identify it, a text or binary value, and a series of detailed options that describe what Lucene should do with the field value when you add the document to the index. To index our collection sources, we must first translate it into Lucene’s documents and fields. The indexing module takes from preprocessed collection all visit reports into a single Lucene document. Lucene allows duplicate fields to be added to a Document. This can make updating documents easy because it allows to add fields of various reports into a single Lucene document, with a mixing of all visit reports in a single document Lucene. At the end, we have the visit reports sharing the same Lucene document and the fields of the various reports sharing the same Lucene field.

4 Retrieval approaches

This section presents the different models developed for evaluation. Among the models developed four have been selected for submission: Baseline², Medical Face³, BWRTDDD⁴ and BWRTDDD2⁵. From the four topics samples review, we consider using some alternative models to the final runs. Although discard some models, we believe interesting to include a review of these unused models that can have interesting for scientific community.

²UHU1BL TREC run ID.

³UHU2MFB TREC run ID.

⁴UHU3BWRTDDD TREC run ID.

⁵UHU4BWRTDDD2

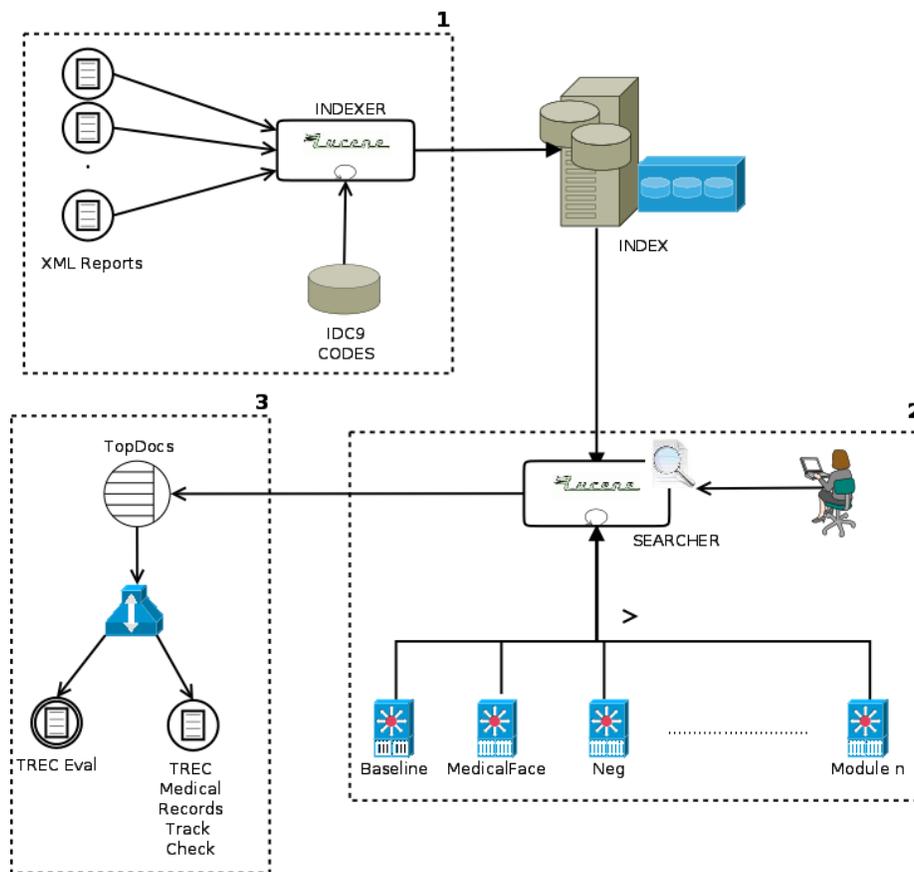


Figure 1: Prototype architecture.

The retrieval module completes the visits retrieval process. The subsequent processes of assessment and results extraction have been implemented according to the TREC eval utilities. The overall process can be seen in Figure 1.

4.1 Baseline

This model has been designed to be the simplest approximation to the task. The Baseline model is based on the method of the bag of words. In this model, topics text is represented as an unordered collection of words, disregarding grammar and even word order. Therefore, the model matches the words in the topic with the words contained in the index. The usefulness of the model is twofold: provides a basis results for comparing and, on the other hand, its code serves as a basis for implementing more complex models. In order to maintain the simplicity, the baseline model makes match the topics words with the words contained in the field “report text”.

To develop this model we used a very simple analyzer. The analyzer takes only the topic and provides a set of terms to be searched in the index. Our base analyzer discards stops words with little semantic value, such as "the", "a", "an", "for",.... Cutting down on the number of terms indexed can save time and space in an index, but it can also limit accuracy.

4.2 Medical Face

MedicalMiner initiative brings together three research groups from Huelva, Vigo and Madrid European Universities. MedicalMiner initiative proposes to analyse, experiment and develop new text and data mining techniques in an interrelated way, in intelligent medical information systems. An intelligent information access system based on them will be developed, offering advanced functionalities able to interrelate medical information, mainly information (text and data) from clinical records and scientific documentation, making use of standard resources of the domain (e.g. UMLS, SNOMED, Gene Ontology). An open source platform will be developed integrating all the elements [3]. A main component of this platform is Medical Face⁶. Medical Face is a Biomedical text annotation system using multiple ontologies. In its simplest form, Medical Face takes a input text and makes medical entity recognition over this text.

For our approach, Medical Face has been used to develop an analyzer that extracts relevant entities in the medical domain from the topics. This relevant entities are used to search for relevant documents in the generated index during indexing process. In this model, only the “report text” field is taken account.

4.3 BWRTDDD

The BWRTDDD model is based on the method of the bag of words in a similar way of the baseline model. Unlike the baseline model this model takes into

⁶<http://orion.esp.uem.es:8080/MedicalFace/mt.html>

account the “report text” and the modified discharge diagnosis fields. The modified discharge diagnosis field is a modified version of the field that takes account a textual description of ICD9 codes generated during preprocessing process. In addition, a specific analyzer has been developed for processing the items.

Like our base analyzer, the BWRTDDD analyzer discards common English words with little semantic value, such as "the", "a", "an", "for",..... Additionally, some irrelevant terms are removed from the topic (patient, for example).

4.4 Exploring Age and Gender

In the first exploration of the sample topics, we think that would be interesting to identify genre and age in the topics and reports. It is easy to detect the occurrence of words that help classify subjects by patient gender and/or age (like woman or elderly). While gender can be detected directly in both the topics and reports, age can be estimated from some labels in the topics (young, adult, elderly,...) and reports (**AGE[in XXs]-year-old). With this simple approach we can match those relevant visits to a topic according to age and gender. Unfortunately, this approach has been unsuccessful with the test topics by the small number of examples that show some of these features (only the adult, children, female and women tags have been found by this model).

4.5 Explore Negation

Another way that looked promising was the negation detection. Usually, narrative reports in medical records contain a wealth of information that may augment structured data for managing patient information and predicting trends in diseases. Pertinent negatives are evident in text but are difficult to automatic processing. In our preliminary study of the sample topics, we saw that one of the examples contained a negation.

Our research group have experience in the development of new text mining techniques adapted to bilingual medical domain. Including basic operations of text categorization and information extraction, capable of addressing specific problems of special relevance in the domain, such as the treatment of negation.

In [2] we present a machine learning system that identify the negation and speculation signals in biomedical texts, in particular, in the BioScope corpus.

Using the [2] approach, a prototype to handle negation was developed. This model takes a topic to detect negation presence. If the topic has a negative part then the medical entities in the negative clause are detected. A visit search are performed deleting those visits with presence of these “negative” entities. Unfortunately, no topic with negative part was given, therefore, no model run was sent.

	bpref	R-prec	P@10
Best	0.7763	0.5917	0.9500
Median	0.4219	0.2924	0.4500
Worst	0.0000	0.0000	0.0000
UHU1BL	0.4534	0.3543	0.4000
UHU2MFB	0.4382	0.3103	0.3000
UHU3BWRTPDD	0.3709	0.1657	0.2000
UHU4BWRTPDD2	0.3709	0.1498	0.1000

Table 1: Medical Minner Medical Records results at TREC 2011.

Topic	bpref					
	Best	Median	UHU1BL	UHU2MFB	UHU3BWRTPDD	UHU4BWRTPDD2
101	0.8605	0.7149	0.6030	0.6030	0.7179	0.7179
102	0.7519	0.4102	0.2613	0.5275	0.4098	0.4098
103	0.7083	0.0972	0.2361	0.0000	0.1528	0.1528
104	0.8765	0.5926	0.8025	0.0000	0.5926	0.5926
105	0.9760	0.9504	0.9672	0.9672	0.9630	0.9630
106	0.7510	0.2487	0.1914	0.4069	0.2163	0.2163
107	0.6465	0.3856	0.3667	0.4858	0.3705	0.3705
108	0.4201	0.0828	0.1243	0.1538	0.2071	0.2071
109	0.8810	0.7529	0.8269	0.0299	0.7753	0.8231
110	0.9537	0.8695	0.9022	0.9197	0.7515	0.7515
111	0.8413	0.1701	0.5057	0.0249	0.0567	0.0567
112	0.9878	0.8685	0.8563	0.8050	0.8848	0.8837
113	0.6173	0.3061	0.3622	0.0918	0.1122	0.0918
114	0.8364	0.7005	0.7514	0.6231	0.6036	0.6281
115	0.7292	0.4336	0.5671	0.2909	0.3627	0.4259
116	0.9800	0.6000	0.6600	0.6900	0.0600	0.0600
117	0.9153	0.4029	0.0000	0.4711	0.6426	0.6426
118	0.7925	0.1783	0.2770	0.7925	0.3092	0.2463
119	0.7453	0.4490	0.6673	0.6645	0.4286	0.3431
120	0.8262	0.6174	0.4310	0.7123	0.5700	0.5700
121	0.4431	0.2062	0.0669	0.1300	0.3194	0.3194
122	0.7083	0.4444	0.4757	0.4705	0.4462	0.4462
123	0.7319	0.4389	0.5188	0.2121	0.2718	0.2718
124	0.5833	0.0000	0.0000	0.1667	0.0000	0.0000
125	1.0000	0.0663	0.0000	0.4694	1.0000	1.0000
126	0.7600	0.2000	0.1600	0.2400	0.1600	0.1600
127	0.9837	0.7914	0.8245	0.8562	0.9650	0.9650
128	0.7970	0.4429	0.6167	0.1107	0.3701	0.3701
129	0.5514	0.2496	0.1830	0.3948	0.3713	0.3713
131	0.5758	0.3551	0.3731	0.0703	0.1555	0.1555
132	0.9861	0.9083	0.9456	0.8574	0.9861	0.9861
133	0.2750	0.0475	0.0700	0.0000	0.0525	0.0525
134	0.4732	0.2448	0.3400	0.2889	0.1574	0.1574
135	0.7955	0.5286	0.5260	0.5010	0.7087	0.7087

Table 2: Color data indicating results best than median.

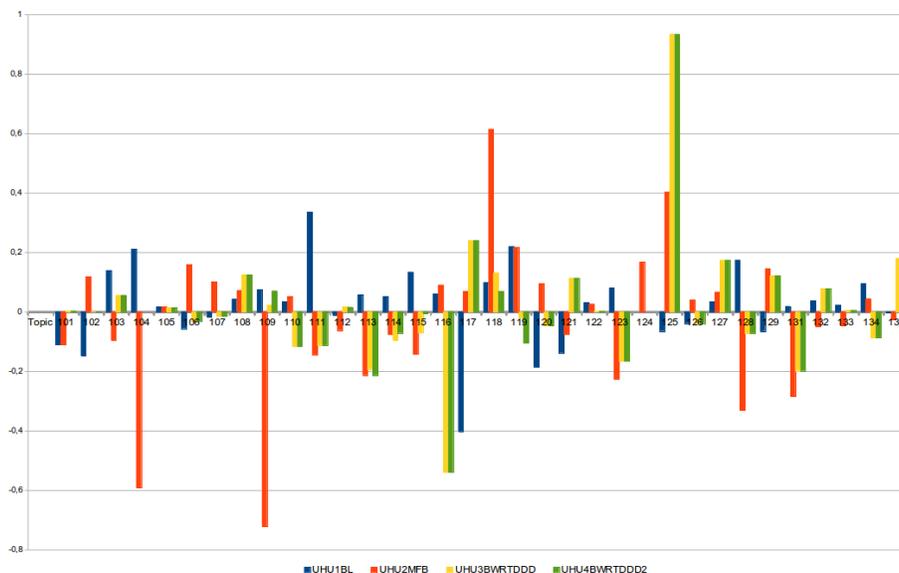


Figure 2: Differences in bpref between the submitted Medical-Miner runs and median results per topic.

5 Results

This section shows the results for our four runs presented to the track (table 1). All our models results are around the evaluation median. In this exploratory track, we think that these results are a good beginning and encourage us for future developments. In despite of initial results, more analysis work needs to be developed. A deeply per topic study must be make. Due to the different topic types, a harder work should be performed to classify the behavior of each model depending on the topic type. About this idea, we have obtained some initial conclusions from the results in some topics. A first improve to the overall system must be made in key concepts detection. At present, our research group are working in a open source platform for medical information retrieval. By now, this platform uses Medline and Freebase as main source of medical concepts detection. Extreme cases (see figure 2), such as topic 104, show that using Medline and Freebase is not enough to detect key medical concepts such as *osteopenia*. In a similar way, our systems have difficult to process expressions like "anti-coagulant medications post-op" (topic 128) or "underwent minimally invasive abdominal surgery" (topic 131). On the other hand, this approximation work fine with usual concepts like Hepatitis C, HIV or coronary stent (topics 118 and 125). The use of new ways of keyword extraction based on medical terminology databases, like UMLS [1] and SNOMED [5], are in our future plans.

About our models, the best results were obtained with the Baseline model. Despite being the best model, there are no major differences with other models.

When inspecting in detail the topics we have seen that an important number of them (14-15 depending on model) have better results with the alternative models (HUH2MFB, UHU3BWRTDDD, UHU4BWRTDDD2), better even track median (see table 2). This performance in certain topics suggests that a better fit in model parameters can give better results if a larger number of sample topics are available.

6 Conclusions

We have presented a novel runs of looking at the problem of medical text retrieval in the Medical Records Track framework. We feel that our models will provide effective retrieval in the order of track median. In our first TREC participation, we feel that track goal has been met reasonably well by the runs proposed.

In the near future, we will make a more detailed study about results. We hope to get more facts about models performance and alternative ways to improve the models.

Apart from the send runs, new ways to improve the medical text retrieval have been proposed. Mainly we have shown our interest in detection of expressions of negation. We think that the detection of negation must be taken into account in the track future.

Our research group has gained experience on development of applied retrieval information systems [4]. We think This first prototype developed to TREC is feasible to help medical domain experts and medical professionals to carry out patient record searches in a more accurate and efficiently way.

References

- [1] O. Bodenreider. The unified medical language system (umls): integrating biomedical terminology. *Nucleic acids research*, 32(suppl 1):D267–D270, 2004.
- [2] N.P. Cruz, M.J. Maña, and J. Mata. Aprendizaje automático versus expresiones regulares en la detección de la negación y la especulación en biomedicina. *Procesamiento de Lenguaje Natural*, 45, 2010.
- [3] M. de Buenaga, F. Fdez-Riverola, M. Maña, E. Puertas, D. Glez-Peña, and J. Mata. Medical-miner: Integración de conocimiento textual explícito en técnicas de minería de datos para la creación de herramientas traslacionales en medicina. *Procesamiento de Lenguaje Natural*, 45, 2010.
- [4] M. Millán, A. Muñoz, M. de la Villa, and M.J. Maña. A biomedical information retrieval system based on clustering for mobile devices. *Procesamiento de Lenguaje Natural*, 45, 2010.

- [5] M.Q. Stearns, C. Price, K.A. Spackman, and A.Y. Wang. Snomed clinical terms: overview of the development process and project status. In *Proceedings of the AMIA Symposium*, page 662. American Medical Informatics Association, 2001.