Atigeo at TREC 2012 Medical Records Track: ICD-9 Code Description Injection to Enhance Electronic Medical Record Search Accuracy

Bryan Tinsley, Alex Thomas, Joseph F. McCarthy, Mike Lazarus
Atigeo, LLC
800 Bellevue Way NE, Suite 600
Bellevue, WA 98004 USA
http://atigeo.com
mike.lazarus@atigeo.com

Abstract

The TREC 2012 Medical Records Track task involves the identification of electronic medical records (EMRs) that are relevant to a set of search topics. Atigeo has a Computer-Aided Coding (CAC) product that analyzes electronic medical records (EMRs) and recommends ICD-9 codes that represent the diagnoses and procedures described in those medical records. We have developed a suite of natural language processing (NLP) components that are useful for both tasks. Our TREC 2012 experiments focused on the ICD-9 admission and diagnosis codes included in more than 90% of the TREC EMRs: we used our comprehensive ICD-9 database to insert one of three variants of the text descriptions associated with each code found in each EMR. We describe the variations of ICD-9 code descriptions we inserted, the NLP components used for processing all the reports and topics, and report on the results of our experiments.

1 Introduction

The 2012 Text REtrieval Conference (TREC 2012) Medical Records Track was designed to promote the research and development of free-text search engines that can find electronic medical records (EMRs or *reports*) that are relevant to specified queries (*topics*). In 2012, the competitive evaluation involved the provision of 100,866 reports aggregated into 17,265 distinct patient encounters (*visits*), a set of 35 topics used for the track in 2011, the official relevance judgments for those topics, and a new set of 50 unseen topics that was used to evaluate performance across several metrics in 2012.

Atigeo participated in the TREC 2012 Medical Records Track in order to test and refine the natural language processing (NLP) components we have developed to support our xPatterns Computer-Aided Coding (CAC) product, which provides capabilities for medical concept extraction and medical coding inference. Although the task of the Medical Records Track – finding aggregated EMRs that are relevant to a specified topic – differs in some significant ways from the CAC task – classifying EMRs based on International Classification of Diseases (ICD) codebooks – we found that NLP components developed for our CAC tool enhanced our TREC components and that the NLP components developed for TREC helped enhance our CAC tool.

One of the most important components in our xPatterns CAC tool is a comprehensive database of ICD-9 codes and descriptions. In reviewing the EMRs, search topics and results from the TREC 2011 Medical Records Track, we discovered cases where the information contained in the descriptions associated with ICD-9 codes contained in the EMRs provide valuable information in determining the relevance of those EMRs to a search topic. We decided to experiment with three variations of inserting ICD-9 code descriptions into the EMRs, to determine how they would affect the relevance of results.

We describe the NLP components used for TREC 2012, and provide more details about the motivation for and use of ICD-9 code descriptions, in the next section. Our scores for all four primary metrics used for TREC 2012 - inferred average precision, inferred normalized discounted cumulative gain, R-precision and Precision @ 10 – were above the average median scores across all topics for all participants. We report those scores, along with some analysis of the variations in our results, in Section 3. We conclude with some thoughts about potential enhancements to the NLP components used for TREC.

2 Atigeo's Experimental Search System for TREC

We developed a modular search system for electronic medical records whose key component is a Natural Language Pre-Processor (NLP-P). The primary purpose of the NLP-P is to reduce the lexical complexity, ambiguity and obscurity in the free text found in both the corpus and search queries. We iteratively tested the system on the 2011 Medical Track search topics, using the results scored against the official judgments file to steer the development, inclusion and arrangement of each component considered for the NLP-P. The NLP-P was then configured into two similar pipelines; the Corpus Natural Language Pre-Processor (CNLP) and the Query Natural Language Pre-Processor (QNLP) illustrated in Figure 1.

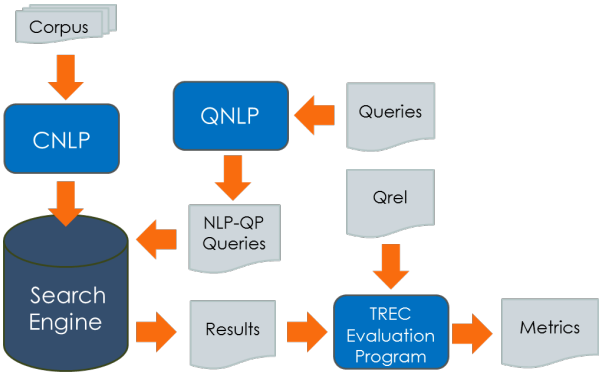

Figure 1. Atigeo Search System Process Flow

The CNLP was configured to extract and process the free text portions of the TREC EMRs prior to their being indexed by a search engine. The CNLP components are designed for use with any search engine; we used Indri for our official TREC 2012 runs. The QNLP used a subset of these components for processing the topics. The CNLP and QNLP pipeline and text processing at each stage is illustrated in Figure 2.

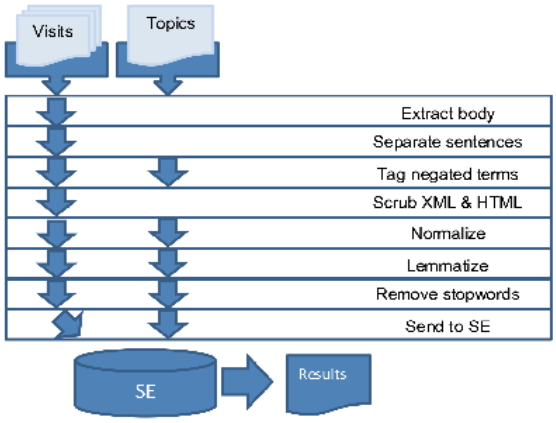

Figure 2. NLP-P pipeline for visits & topics

2.1 Natural Language Processing Components & Resources

One of the ongoing challenges in developing systems that effectively process natural language and retrieve relevant information in any domain is to strike an appropriate balance between comprehensiveness and accuracy. Throughout our development effort, we found that many of the large dictionaries and other resources traditionally used in the medical informatics community, such as the Unified Medical Language System (UMLS)[1] and its Systematized Nomenclature of Medicine--Clinical Terms (SNOMED-CT)[2], and the more general purpose English lexical database, WordNet[3], helped increased the coverage of medical concepts, but often at the expense of increased ambiguity (and, thus, decreased accuracy). Thus, we used such large-scale resources sparingly.

The components we used for the NLP-P represent a mix of off-the-shelf open source NLP tools, domain- or task-specific extensions we developed for such tools, and some special components we created at Atigeo for enabling more effective processing of EMRs

[1] http://www.nlm.nih.gov/research/umls/
[2] http://www.nlm.nih.gov/research/umls/Snomed/
[3] http://wordnet.princeton.edu/

ICD-9 Code Description Injector
In some configurations, the CNLP inserted descriptions of ICD-9 codes found in the *admit_diagnosis* or *discharge_diagnosis* fields of the TREC EMRs into our representation of those documents. More details on the variants of will be described in the Section 2.3.

Text extractor
The text extractor parsed the XML and extracted the contents of the following fields from each report contained in a visit file:

- *subtype*
- *type*
- *chief_complaint*
- *report_text*

The *report_text* field included line-break characters presumably inserted for the purpose of wrapping lines longer than 72 characters. These line-break characters were replaced with blank spaces to better facilitate identifying sentence boundaries. We also searched for the de-identified ages, and proximal patterns indicating race and/or gender (e.g., "The patient is a **AGE[in 20s]-year-old African-American male…"), and inserted special markers for these elements in our document representations.

Sentence Separator
We used the Apache OpenNLP Sentence Detector[4] to separate sentences into separate lines

Negation Tagger
We implemented a negation tagger based on the NegEx tagger [Chapman, *et al*., 2001] using an "n0" prefix to mark negated terms, as described by Limsopatham, *et al.*, [2011]. We extended the 237 negation patterns included with the standard NegEx distribution to cover 25 negation contexts found in the TREC data set that were not well represented in the default dictionary.

XML/HTML Scrubber
We removed all the XML tags surrounding the fields that were extracted. Approximately 5% of the TREC visits include reports with one or more HTML tags in their *report_text* field, (e.g., "<start header>" or "<end footer>"). These appear to be artifacts of software used to generate the report files and do not appear to add any informational value to the reports, so they were removed (or scrubbed).

Normalizer
In normalizing the visit files, we converted all letters to lowercase, removed punctuation symbols, special characters, and extra space characters.

Lemmatizer
Early experiments with the Porter stemmer [Porter, 1980] and default Snowball stemmer [Porter, 2001] revealed examples of ambiguity we believed would have a significantly negative impact on performance. For example, "liver" and "live" were both found to have the same stem, "liv". Other experiments with Morphadorner[5], an open source lemmatizer supported by Northwestern University, appeared to produce better results but still introduced instances of ambiguity, e.g., reducing "liver" to "live." Thus, we created our own context-sensitive lemmatizer designed to reduce the inflectional forms of words to their respective lemmas while minimizing the introduction of additional ambiguity. For the TREC corpus, this more conservative lemmatization approach reduced the number of distinct tokens by 10%, from 120,000 to 107,000. Results of our early experiment in comparing the performance of the search system on the TREC 2011 topics using a Porter stemmer vs. our context-sensitive lemmatizer are shown in Table 1.

	Stemming vs. Lemmatization	
	Porter stemmer	*Lemmatizer*
bpref	45.53	46.61
R-prec	34.56	35.72
P@10	50.00	51.18

Table 1: Stemmer vs. Lemmatizer on TREC 2011 topics.

Stopword Remover
Our stopword list of 202 prepositions, conjunctions, determiners, negation terms and pronouns reduced the number of tokens in the TREC corpus by approximately 33% (from 37 million down to 12 million).

[4] http://opennlp.apache.org/

[5] http://morphadorner.northwestern.edu/]

2.2 ICD-9 Code Description Injection

One of the core xPatterns CAC components is a comprehensive database of the 9th and 10th revisions of the International Classification of Diseases (ICD-9[6] and ICD-10[7]), which are composed of 17,000 and 155,000 short alpha-numeric codes and associated text descriptions, respectively. Over 90% of the 100,866 TREC reports include ICD-9 codes in either the *admit_diagnosis* or *discharge_diagnosis* fields.

In reviewing the TREC 2011 Medical Records Track topics, we noticed that there are some topics that include substrings that closely match ICD-9 code descriptions. For example, all 85 visits that were judged relevant to topic 127 ("Patients admitted with morbid obesity and secondary diseases of diabetes and or hypertension") include ICD-9 code 278.01 ("Morbid obesity") in their *discharge_diagnosis* field (though only 15 of these visits include code 278.01 in their *admit_diagnosis* field).

Further analysis suggests that some relevance judgments must have been based, in part, on the ICD-9 codes. A dramatic example of this can be seen in topic 125, "Patients co-infected with Hepatitis C and HIV", for which 14 visits were judged relevant. Of these visits, only one contains the string "HIV", but 9 include the ICD-9 code 042 ("Human immunodeficiency virus [HIV] disease") and the other 5 include the ICD-9 code V08 ("Asymptomatic human immunodeficiency virus [HIV] infection status").

From our experience with ICD-9 codes in developing our CAC tool, we knew that some code descriptions are rather minimal (e.g., "Other", "Group A" or "Unspecified site"), with additional context provided in the descriptions associated with parent and/or grandparent codes. The following ICD-9 codes and descriptions offer an illustration:

- 786: Symptoms involving respiratory system and other chest symptoms
- 786.5: Chest pain
- 786.59: Other

We therefore decided to experiment with three variations on automatically inserting variants of the descriptions associated with any ICD-9 codes found in these reports, and to use one run with no code descriptions as a control. Thus, the four runs we submitted to TREC 2012 represent the following four variations on ICD-9 code description insertions:

- **None** (no code descriptions inserted)
- **Minimal**: only the descriptions associated with the codes actually found in the reports. For a report with code 786.59, we would insert "Other".
- **Moderate**: the minimal description plus the description associated with a parent code, if applicable. For a report with code 786.59, we would insert "Other" and "Chest pain".
- **Maximum**: the moderate descriptions, plus the description associated with the grandparent code, if applicable. For a report with code 786.59, we would insert "Other", "Chest pain" and "Symptoms involving respiratory system and other chest symptoms".

3 Results

Shortly before submitting our 2012 runs, we scored the results our system generated for the 2011 topics based on the 2011 relevance judgments, and saw sizable gains for all variants using ICD-9 code description insertion. Table 2 shows that the relative ranking of all 4 variants was consistent across all 3 primary TREC 2011 metrics: binary preference (B-pref), R-precision (R-prec) and Precision @ 10 (P@10).

	Atigeo			
	None	*Min*	*Mod*	*Max*
B-pref	55.38	61.49	59.62	60.10
R-prec	44.07	46.20	45.14	46.18
P@10	62.94	67.65	65.88	66.48

Table 2. Unofficial results on TREC 2011 topics

Table 3 shows the comparative performance of the different configurations of our system along with the TREC average median score across all TREC 2012 topics. All three variants in which we inserted ICD-9 code descriptions resulted in increased scores for four primary TREC 2012 metrics (inferred average precision, inferred normalized discounted cumulative gain, R-precision and Precision @ 10).

[6] http://www.cdc.gov/nchs/icd/icd9.htm
[7] http://www.cdc.gov/nchs/icd/icd10.htm

	Atigeo				TREC
	None	*Min*	*Mod*	*Max*	*Median*
infAP	19.94	22.42	21.17	21.23	16.89
infNDCG	47.59	52.37	49.80	49.31	42.43
R-prec	33.02	36.21	34.41	34.26	29.61
P@10	50.00	51.91	50.21	50.00	47.02

Table 3: Official scores for TREC 2012

Our results show that the inclusion of the ICD-9 descriptions improved search accuracy for both the 2011 and 2012 search topics. We initially hypothesized that the Maximum condition would yield the best results. We believe the slightly higher scores for the Minimal code description variant – for both 2011 and 2012 – may be due to a lower probability of introducing additional ambiguity through the inclusion of longer, more general descriptions in the Moderate and Maximum variants.

All of our scores for the 2012 topics were lower than for the 2011 topics. We believe this is due to the increased complexity of the topics in 2012, and the nature of working with a known test set vs. a blind test set

4 Conclusion

The TREC 2012 Medical Records Track marks the first time Atigeo has participated in a TREC evaluation. The TREC data set and evaluation framework offered a valuable opportunity for us to test and refine several NLP components. While we do not yet know the specific ranking of our results relative to other participants, we were generally pleased with the results we achieved across the different TREC metrics.

Our main take-away from participating in TREC is the importance of being very judicious in using large-scale, general-purpose NLP resources, and looking for opportunities to filter or constrain the introduction of ambiguity that often accompanies the use of such tools. Given the importance of accurate identification and scoping of negation terms in the analysis of electronic medical records, we plan to continue our context-sensitive extension of negation patterns. We also plan to iteratively refine our lemmatizer to achieve an optimal balance between reducing complexity and introducing ambiguity. Finally, we believe a new algorithm we developed for our CAC tool to identify key sections in EMRs – such as "HISTORY OF PRESENT ILLNESS" and "FAMILY HISTORY" – may also improve performance on the TREC Medical Records Track task.

Acknowledgments

We are grateful to other members of the Atigeo Science and Engineering teams, especially Andrew Simms and Cosmin Ursachi, for their contributions of ideas and code; to Ellen Voorhees and other members of the TREC program committee for managing the dissemination of materials, the collection and evaluation of the results, and organizing the conference at which we were able to share problems and solutions with each other; and to the University of Pittsburgh Medical Center, for making the de-identified electronic medical records available for use by the TREC Medical Records Track participants.

References

Chapman WW, Bridewell W, Hanbury P, Cooper GF, Buchanan BG. (2001) A simple algorithm for identifying negated findings and diseases in discharge summaries. *J Biomed Inform.* 2001;34:301-10.

Limsopatham, N, Macdonald, C, Ounis, I, McDonald, G, Bouamrane, MM. (2011) University of Glasgow at Medical Records Track 2011: Experiments with Terrier. *The Twentieth Text REtrieval Conference (TREC 2011) Proceedings*. NIST Special Publication SP 500-295, National Institute of Standards and Technology.

Porter, MF. (1980); An Algorithm for Suffix Stripping, *Program*, 14(3): 130–137.

Porter, MF. (2001), Snowball: A Language for Stemming Algorithms. Retrieved 15 October 2012. http://snowball.tartarus.org/texts/introduction.html