Evaluation of a Machine Learning Method to Rank PubMed Central Articles For Clinical Relevancy: NCH at TREC 2016 Clinical Decision Support Track

*Wei Chen[1], *Soheil Moosavinasab[1], Anna Zemke[2], Ariana Prinzbach[2], Steve Rust[1], Yungui Huang[1], Simon Lin[1]
[1]Department of Research Information Solutions and Innovation, The Research Institute at Nationwide Children's Hospital
[2]College of Medicine, The Ohio State University
* these two authors contributed equally

Abstract

The goal of the TREC 2016 Clinical Decision Support track is to retrieve and rank PubMed Central (PMC) articles that are relevant to potential tests, treatments or diagnoses of a patient case narrative. Our objective was to develop a machine learning method to rank PMC articles by taking advantage of the previous years' gold standard TREC competition results. The classifier we trained on 2014 data achieved high accuracy when tested with 2015 data (P10=0.59 and infNDCG=0.67) compared with the Elasticsearch method (P10=0.19 and infNDCG=0.22). However, when we applied the same classifier approach with both the 2014 and 2015 data sets combined, and then tested this method against the 2016 cases, the results did not improve over the Elasticsearch method. We concluded that although the machine learning approach was found effective on predicting previous years' results, it was not as effective for 2016 data, most likely due to the change in the topic structures.

Introduction

PubMed articles contain a wealth of medical evidence that is highly valuable to physicians for finding key medical knowledge for decision-making such as information on the diagnosis, tests and treatment of a disease. A computerized decision support system could improve the efficiency of evidence-based decision-making by automatically scanning millions of articles and providing a more pertinent retrieval of information. The Clinical Decision Support (CDS) track of Text REtrieval Conference (TREC) challenges participants to retrieve and rank PubMed Central (PMC) articles that are potentially relevant to given medical record topics [1].

For this 2016 CDS task, we designed a machine learning approach that has not been previously tested within the TREC CDS track. This new method was compared against the Elasticsearch (ES) method, which we utilized as a baseline from data available across three consecutive years, in order to determine its efficacy. We hypothesized that a supervised learning approach could take advantage of existing implicit knowledge of judges from their decisions in determining the relevancy of articles in previous years, and if successful, this learning approach could be applied to predict new cases. In this paper, we report our design of this learning system and discuss the experiment results when using different systems as well as the different data from previous years.

Data

Topical Data

2014-2016 topic data were used along with the gold standard data for 2014 and 2015. The gold standard data included relevancy scores for each article by each topic assigned by TREC CDS judges. The score had a value from 0 to 2 with 0 indicating irrelevant, 1 potentially relevant, and 2 definitely relevant. Given the fact that majority of the articles judged in previous years were graded as irrelevant (overall >90% for all topics), we combined articles judged potentially relevant and definitely relevant into one group and assigned the value of 1 to these articles. In terms of the versions of topics, for the 2014 and 2015 topics we only used case summaries as this was the only data provided, while for the 2016 topics we used case summaries for automated runs, and then the newly included patient notes for the manual run and the test of the baseline method (Elasticsearch method) only.

PubMed Central Data

About 1.25 million PubMed Central (PMC) articles provided by TREC were separated into title, keywords, abstract and body sections for Elasticsearch indexing.

Unified Medical Language System (UMLS)

The NLM UMLS was used as a knowledge component of our system for extracting key medical terminologies from the TREC topic text. UMLS was used to identify medical concepts in the topic text and relate them to semantic categories (i.e. disease, symptoms, findings, etc.) and alternative names (i.e. synonyms, preferred names, etc.). Previous studies have found UMLS to be very effective in NER (Named Entity Recognition) tasks on medical documents [2].

Methods

To build our system, we first used UMLS to extract medical terms from each topic. We then designed two components of our ranking system: the Elasticsearch (ES) component and the learning component. Both components have ranking capabilities to work independently or collaboratively. The focus of this work is on evaluating the learning method against the ES method, and a comparison to previous and this year training as well as other groups for this year's TREC CDS challenge. We also reported the results of our manual approach.

Topic Processing Using UMLS

We used UMLS to extract key medical terminology from the summary and note text of each topic. The goal of UMLS processing is to find medical terms in the input text that are key for clinical decision making such as disease, symptoms, findings, etc. Information such as gender, age and ethnicity can be useful for diagnosing certain diseases. However, in other cases, such information is not particularly specific to the diagnosis, test or treatment of a disease and therefore were excluded from the UMLS parsing.

The most recent UMLS MetaMap Java API was used to first detect UMLS terms and their CUIs (Concept Unique Identifiers). For example, the case summary "a 78 year old male presents with frequent stools and melena." will be parsed into two concepts "frequent stools (C0848342)" and "melena (C0025222)". The numbers in the parenthesis are CUIs of each term, which can be used to uniquely identify a concept. Based on the CUI, we expanded the concept to include some of its variations, such as its English synonyms or preferred names. We did not include non-English variations, or English variations that were not synonyms or preferred names, or whose usage was suppressed.

UMLS is very effective in finding optimal phrase boundaries. In the above example, although "stools" is also a concept but because it is part of another meaningful phrase "frequent stools", the extraction of the longer phrase "frequent stools" triumphed over that of the shorter phrase "stools". We think this decision made by UMLS is intuitively appropriate because longer phrases usually tend to be more precise than single terms (i.e. "frequent stools" vs. "stools") and the correct boundary cutting can be critically important for effective clinical decision-making.

The results of UMLS processing include two lists of terms for each version of the topic: UMLS terms and UMLS expansion terms. All terms were converted to lower cases and all duplicates were removed. UMLS terms were used in building the Elasticsearch system to index PubMed articles while both UMLS terms and UMLS expansion terms were used in building the learning system.

Different from previous years, the 2016 topic data included the medical history of a patient and conditions that could not be decided as either current or previous. To differentiate present conditions from previous, we implemented a regular expression parser based on a previous study [3], to classify extracted medical terms in three categories: present, previous and unclassified.

Supervised Learning

The availability of the gold standard data from the two previous years, makes it possible to build a supervised learning system to classify topic-specific articles. In the gold standard data, we combined a score of 1 and 2 to be 1 and made it a single pot of relevant set, that is, 1 indicating relevant and 0 irrelevant. From UMLS, we obtained two lists of keywords in a specific topic text: UMLS terms and UMLS expansion terms.

The extracted UMLS terms cannot be used as features directly either as a binary or as a count variable because the list is ever expanding given the content of a new topic. Therefore, we implemented a vectorization process to convert continuous keyword occurrences to a Weighted Keyword Count (WKC) for both UMLS terms and UMLS expansion terms. Based on WKC, we developed two features for training: WKC_{UMLS}, the feature accounting for the frequency of UMLS term occurrences in the article and $WKC_{UMLSExpansion}$, the feature accounting for the frequency of UMLS expansion term occurrences in the article. The two WKC features was calculated as follows:

$$WKC_{UMLS} = \frac{the\ total\ number\ of\ unique\ UMLS\ term\ matches\ in\ the\ article}{total\ number\ of\ UMLS\ terms\ in\ the\ topic}$$

$$WKC_{UMLSExpansion} = \frac{the\ total\ number\ of\ unique\ UMLS\ expansion\ term\ matches\ \ in\ the\ article}{total\ number\ of\ UMLS\ expansion\ terms\ in\ the\ topic}$$

The two features were calculated for each article by each topic in the 2014 and 2015 gold standard dataset. Due to the extreme unbalancing of the data in the gold standard set, we matched the total number of irrelevant entries to the total number of relevant entries to mandate a 50:50 class split for machine learning. This resulted in about 16,000 annotated machine learning instances for binary classification.

Additionally, we added a binary topic type feature indicating whether the topic type keyword is present in the article or not. The topic keywords were also derived using UMLS based on three primitive types: diagnosis, test and treatment. The training process aimed to find both the best features to use and the best model to classify instances.

Elasticsearch Indexing

UMLS terms extracted from the topic text were used to index the title, keywords, abstract and body sections of a PMC article using Elasticsearch (ES). Among different algorithms provided by ES, we selected BM25 (parameters k1=3 and b=0.75) as our ranking algorithm due to slightly better performance we observed than using other algorithms. We also implemented a logical OR query in ES. To compensate the effect of the variations in the length of the queries generated from notes or summaries, we used a “minimum percentage match” criterion for search queries. We require an article to be a match for a note query if at least 15% of the keywords matched in that document. This percentage increases to 20% for summary note type because summaries are shorter.

For the 2016 task, we boosted present keywords more than unclassified and previous keywords. According to our domain experts, for all three topics: diagnoses, tests, and treatment, the data of the current condition of the patient is more useful in selecting relevant articles than data on the previous condition. The difference in importance of current versus previous conditions is most exaggerated in treatment and less so in diagnoses and tests. Therefore, we designed two boosting strategies. For treatment topics, we boosted present condition 3 times, unclassified condition twice and previous condition once. For diagnosis and test topics, we boosted present condition three times, unclassified condition twice and previous condition also twice. We applied this boosting only on summary notes because full notes are too “messy“ to extract present and previous conditions effectively.

Other information retrieval techniques we tested (such as stemming, proximity matching, boosting phrases, and prefix matching) did not contribute to better results; therefore we did not apply these techniques to the final system.

Manual Run

We invited 3 physicians to participate in our experiment. The search keywords were manually provided by our domain experts and we utilized Google to automate the retrieval of PMCIDs within the PubMed domain. Each physician was assigned 10 topics

and provided a list of 2 to 4 key-phrases based on the full note of the topic. The key-phrases did not have to be part of the note, and could be derived from the person's own knowledge after reading the medical case. The key-phrases were then entered into google, and the PubMed articles on the first search page were reviewed to determine if these articles were relevant to the medical case and would assist in establishing the diagnosis. If the majority of articles were deemed pertinent, the domain experts and physician would confirm the utilized key-phrases as finalized. However, if the search returned articles that were not closely relevant to the diagnosis or medical note, the key-phrases were adjusted until the majority of articles on the first search page provided useful information. When the final list of key-phrases was compiled, the computer created a query and automatically retrieved the top 1000 available PMC-IDs for each topic by searching Google and restricting the results from site:ncbi.nlm.nih.gov/pmc.

Results

In machine learning, both features WKC_{UMLS} and $WKC_{UMLSExpansion}$ were shown to have high discriminatory power based on the information gain and principle component feature ranking methods. However, the topic type feature was found to be not discriminatory enough to be included the model. Therefore, we had only two WKC features selected for machine learning. We also calculated the average of WKC_{UMLS} and $WKC_{UMLSExpansion}$ using 2014 and 2015 data combined and test the statistical significance of the mean using the paired t-test (Table 1).

Table 1. Weighted Keyword Count features

Variable	Relevance=0 (N=67410)	Relevance=1 (N=8346)	p-value
Average WKC_{UMLS}	0.14	0.27	<0.01
Average $WKC_{UMLSExpansion}$	0.17	0.38	<0.01

We first implemented logistic regression classifier using 2014 data (about 7000 instances with 50:50 split) for training and 2015 data for testing (about 10000 instances with 50:50 split). The overall AUC of the classifier on 2015 test data is 0.79 (TP=0.72, FP=0.28, F1=0.71). According to the standard TREC evaluation metrics, the overall infNDCG was 0.67 and overall P10 was 0.59 on the 2015 test data.

To predict 2016 results, we then combined the gold standard data from two previous years 2014 and 2015 to train a new logistic regression classifier. ES was used to generate the top 1000 results for each topic. Our new classifier was then used to rerank the ES results. Both the ES results and the classifier reranking results were part of our

TREC 2016 submission. Table 2 shows the results for 2016. All reported TREC performance measures are the averages of all topics.

Table 2. 2016 submission overall results

Run name	NoteMan	NoteES	SumES	SumClsRerank	SumCmbRank
Topic Source	Notes	Notes	Summary	Summary	Summary
Approach	Manual	ES	ES	ES + ML reranking	ES + ML averaged
infNDCG	0.29	0.16	0.22	0.14	0.20
P10	0.46	0.25	0.34	0.22	0.28

In comparing our results with submissions from the other 25 participating organizations, our manual run was the overall top performer among 8 manual runs using the note data, and our SumES automatic method was ranked top 10 overall among 46 automatic runs using the summary data (more specifically ranking 2nd on infAP, 4th on P@10 and 7th on infNDCG among all automatic runs using the summary data). Figure 1 and Figure 2 show the performance comparison of our top manual and automatic runs against the median and best runs from each topic for TREC 2016 CDS. We omitted the worst performance here because the majority of the values were zeros. Topic 31 does not exist and is calculated as the average performance across all topics for each comparison (i.e. NCH, Median and Best).

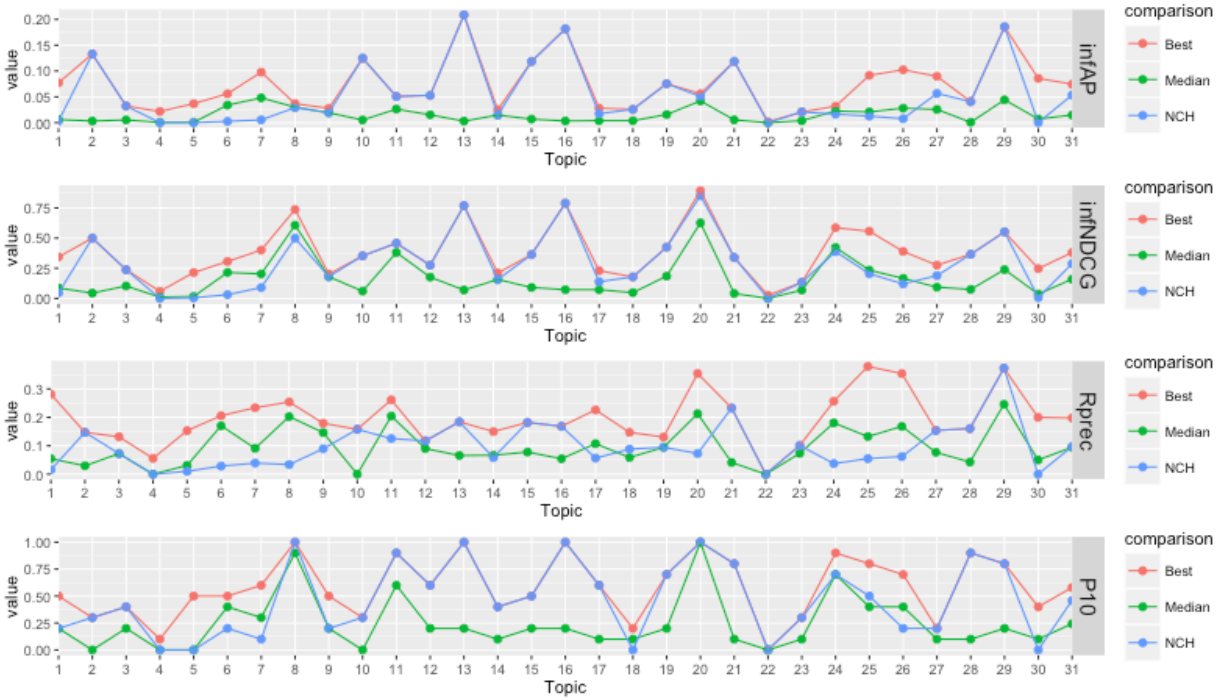

Figure 1. Our manual run (blue) performance compared with the median (green) and best (red) performance from all manual run submissions on each topic (Topic 31 is calculated as the group average).

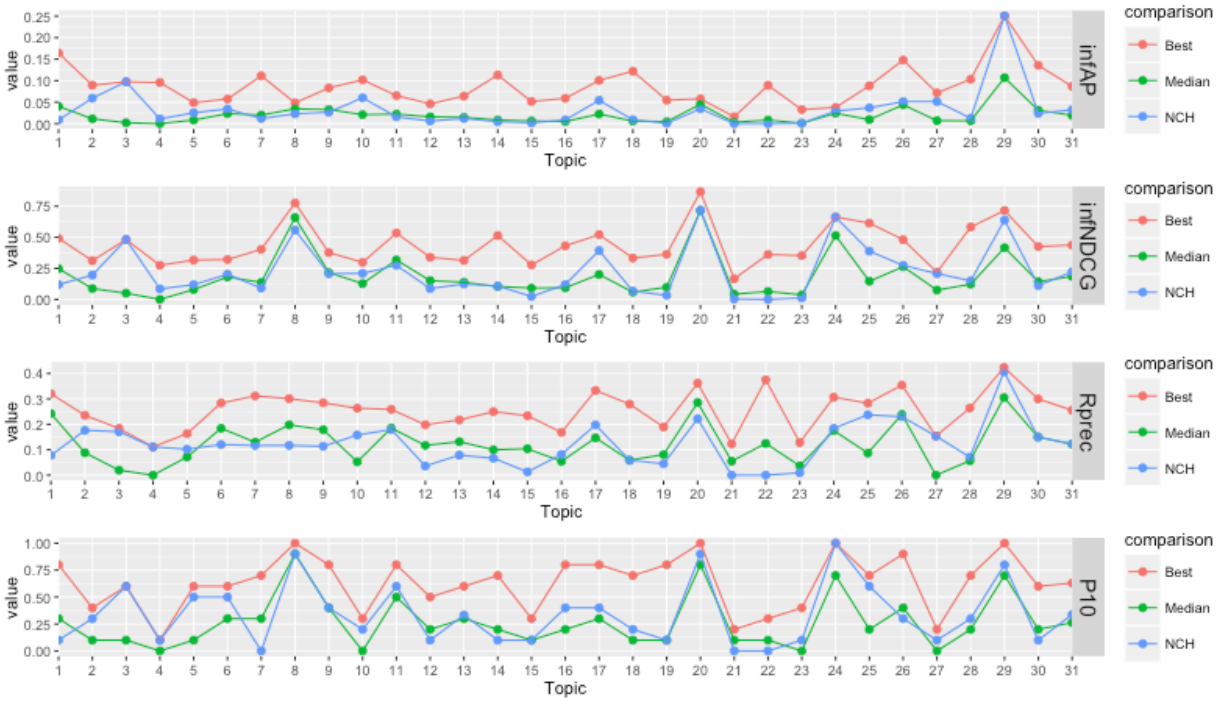

Figure 2. Our best automatic run (SumES, blue) performance compared with the median (green) and best (red) performance from all automatic run submissions on each topic (Topic 31 is calculated as the group average).

Discussion

The performance of the machine learning approach varies in testing against 2015 and 2016 gold standard data. The machine learning classifier trained using 2014 data and evaluated using 2015 data produced more accurate results than the classifier trained using both 2014 and 2015 data combined and tested using 2016 data. Since the classifier was trained using a different structure of data (e.g. 2016 topic data contained a mixture of present, previous and unclassified conditions) than the data provided in the 2014 and 2015 competition, we believe that this method of machine learning could be more effectively utilized in future years when both the training and testing data are utilizing the same level of detail in the topic structure.

Conclusions

In this paper, we experimented with a machine learning approach to classify PubMed articles for the TREC CDS challenge. This approach was very effective with the 2014 gold standard data to predict 2015 gold standard data, but was not effective on re-ranking the ES results for the 2016 task. We concluded that although the use is potentially promising for similar tasks, the machine learning approach was sensitive to the structural change of the input data. Better strategies need to be implemented to generalize this approach to adapt to heterogeneous nature of medical record narratives.

References

1. Roberts K, Simpson MS, Voorhees E, Hersh WR. Overview of the TREC 2015 Clinical Decision Support Track.
2. Park A, Hartzler AL, Huh J, McDonald DW, Pratt W. Automatically Detecting Failures in Natural Language Processing Tools for Online Community Text. Journal of Medical Internet Research 2015;**17**(8)
3. ConText: An algorithm for identifying contextual features from clinical text. Proceedings of the Workshop on BioNLP 2007: Biological, Translational, and Clinical Language Processing; 2007. Association for Computational Linguistics.