Question Answering on Electronic Medical Records

Preethi Raghavan, PhD[1], Siddharth Patwardhan, PhD[1]
[1]IBM TJ Watson Research Center, Yorktown Heights, NY, USA

Introduction

Electronic medical records (EMRs) document the healthcare provided to the patient using both unstructured clinical notes and structured data. Typically, these records are large, containing hundreds of documents about a single patient. To navigate the vast amounts of information about a patient, current EMR applications available to physicians either attempt to proactively summarize important patient information or they provide a more traditional search service that retrieves notes "relevant" to a given query. However, it may be possible to much more precisely (and automatically) answer very specific questions about the patient's health, by providing short answers to natural language questions – perhaps supported by evidence. For e.g., "Why was the patient prescribed arginine?" or "Were there any complications of the patient's RYGB surgery?" could be answered with very specific phrases or passages from the EMR. Recent advances in automatic question answering (QA) technology may help answer such questions.

While there have been several attempts at question answering in the NLP community (TREC QA challenges[1], IBMs Jeopardy! winning Watson QA[2], Apple's Siri), there have been no question answering efforts in the clinical domain. Patient-specific question answering from an EMR has some significantly different challenges when compared to open-domain QA. These challenges are brought on by the corpus size being limited to the patient's EMR, no redundancy in facts, and longitudinal and domain-specific nature of information centered around a patient. Moreover, the nature of questions is not always factoid. Therefore, deeper analysis of clinical text is required to address problems like temporal reasoning, relation detection, discourse analysis both within and across clinical notes. Recognizing these challenges on noisy clinical sub-language, we propose a baseline implementation of the EMR QA system that provides a starting point for addressing these problems using Watson passage scoring technology[3]. We will extend this initial version of the EMR QA system in our future work.

Methods

The goal of the EMR QA system is to process a natural language question and provide an accurate answer.

Dataset: The questions and corresponding answers in the corpus were primarily generated by our team of annotators (medical students) who are considered as experts for the purposes of annotating clinical text. A physician from our health care partner, Cleveland Clinic, also generated some of the questions. All questions and answers were generated in the context of EMRs, across multiple specialities, provided by Cleveland Clinic. The preliminary dataset of questions and answers comprises 100 question and answer pairs*. The type of questions are restricted to ones that may be answered using all the information available in the EMR.

QA System: This initial version of the EMR QA system implements a simplified version of the UIMA-based Watson QA architecture[3]. However, it also incorporates a few significant changes, such as the EMR Content Manager (Figure 1), to accommodate the longitudinal nature text centered on the patient.

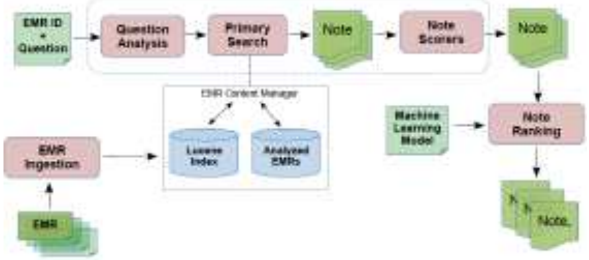

Figure 1. Architecture of the initial version of EMR QA system

The overall flow of information processing in the system, as seen in Figure 1, is as follows: 1. **EMR Ingestion**: This component indexes EMRs for all patients using Lucene. The clinical notes are indexed along with metadata including all sentences and sentence-level features such as sections, note-level features such as the note date and type. 2.

*We plan on releasing an extended dataset of natural language questions for EMR QA

Question Analysis: A question is asked against a specific EMR (corresponding to a single patient). Question analysis extracts and analyzes parts of the question that may be important to other components in the system. This includes medical concepts and relationships expressed between them, temporal keywords and their dependencies, the expected answer type etc. It also generates the search query that is used by the primary search. 3. **Primary Search**: This component searches the Lucene index generated in the EMR Ingestion phase and produces a hit list of clinical notes. The EMR Content Manager maintains access to all notes in the EMR across all components in the system. 4. **Note Scoring:** This initial version of the QA system uses Watson's passage scorers[3] to identify, within clinical notes, direct paraphrases of the fact expressed in the question. The deeper matching problems including temporal analysis, cross-narrative problems and dealing with the semi-structured text in the notes are left to future versions of the system. A set of 50 of passage scorers is now run against all pairs of questions and notes from the hit list. The scorers try to align the question against notes in the hit list, using various structural and semantic similarity metrics, and produce a score. 5. **Note Ranking:** These scores are used as features to train a cost-sensitive logistic regression classifier and identify the notes that answer the question. Another important component in future versions of this system would be candidate answer extraction and scoring. However, we restrict this version of the system to note scoring.

Results

The EMR QA accuracy is measured by taking the top N responses (above threshold) from the classifier, if any of them are the correct answer as determined by our gold-standard annotations, then that question is marked "correctly answered". Accuracy @ N is obtained by dividing the number of correctly answered questions by the total number of questions. We run our EMR QA system using cross-validation with a cost-sensitive logistic regression classifier with parameters tuned on our development set of 100 questions. Over this development set, our system produced an accuracy of 0.36 for N@1, 0.47 for N@5 and 0.48 for N@10.

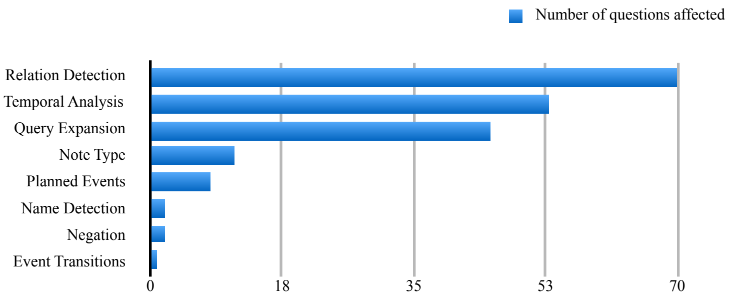

Figure 2. Error Analysis

As seen in Figure 2, error analysis on the questions that we do not answer correctly indicated that the lack of relation detection, i.e. detecting in clinical notes, the medical entities along with relations expressed in the question, affected 70% questions. Further, the ability to resolve temporal constraints in the question and factor it into note scoring would help over 50% of the questions. Performance may also be improved by addressing additional problems such as expanding categorical terms in the question (for e.g., names of medications), modeling the type of clinical notes, identifying medical events that have not yet occurred and are planned for the future, detecting entities like physician names, negated medical events, and the change of state for a medical event (for e.g., medication or procedure started or stopped). We are working on addressing these problems and improving the performance of the EMR QA system.

Conclusion

The contribution of this work is a novel question answering system for automatically answering natural language questions from EMRs. The results are encouraging as this initial version of the EMR QA system answers almost 50% of the questions correctly. The primary issues that need to be addressed for performance improvement include relation detection and temporal analysis. We plan to address the many clinical domain-specific challenges and extend the capabilities of the system for effective and robust EMR question answering.

References

1. Voorhees, Ellen M. "The TREC question answering track." Natural Language Engineering 7, no. 04 (2001): 361-378.
2. Ferrucci, David, Eric Brown, Jennifer Chu-Carroll, James Fan, David Gondek, Aditya A. Kalyanpur, Adam Lally et al. "Building Watson: An overview of the DeepQA project." AI magazine 31, no. 3 (2010): 59-79.
3. Murdock, J. William, James Fan, Adam Lally, Hideki Shima, and B. K. Boguraev. "Textual evidence gathering and analysis." IBM Journal of Research and Development 56, no. 3.4 (2012): 8-1.