Case Study: Natural Language Interface for Patients' Electronic Health Records (EHR)

Madhumitha Sivaraj
ms2407

Revanth Korrapolu
rrk69

Abstract

Electronic Health Record (EHR) related problems have proved to be costly for hospitals, stress-inducing for physicians, and life-threatening for patients. Instead of solving problems of the paper-based healthcare system, they create new ones. Numerous studies have shown this is a result of poorly design EHR systems. To solve this, we propose a Natural Language Interface (NLI) that can assist doctors is navigating the maze of patient data. By simply taking in the doctor's question as input, our system will query the patient health database and find relevant information, and then construct an natural language answer as output. Our application is a case study of the real-world efficacy of using modern NLU and NLG techniques to create an EHR NLI.

1 Problem

The health care environment often consists of long working days, stressful speed, time pressure, and emotional intensity. This atmosphere tends to place doctors and other clinicians at a high risk of burnout. Physician burnout is a universal dilemma that is characterized by emotional exhaustion, depersonalization, and a feeling of low personal accomplishment.[1]

One AHRQ-funded project, Minimizing Error, Maximizing Outcome also known as MEMO (Grant HS11955), found that more than half of the physicians reported experiencing time pressures when conducting physical examinations. According to the Agency for Healthcare Research and Quality, nearly a third of physicians felt they needed at least 50 percent more time than was allotted for this patient care function.[2] The MEMO study found that electronic health record (EHR) systems actually contributed to physician burnout.

Among 91% of EHR users, 70% reported health information technology (HIT) related stress, with the highest prevalence in primary care-oriented specialties.[3] Physicians reporting poor or marginal time for documentation had 2.8 times the odds of burnout, compared to those reporting sufficient time, after adjustments.[3] Physicians reporting moderately high or excessive time on EHRs at home had 1.9 times the odds of burnout, compared to those with minimal or no EHR use at home.[3] Those who agreed that EHRs add to their daily frustration had 2.4 times the odds of burnout, compared to those who disagreed.[3]

In a study published last year in the journal Health Affairs, Ratwani and colleagues studied medication errors at three pediatric hospitals from 2012 to 2017. Using eye-tracking technology, Ratwani has demonstrated just how easy it is to make mistakes when performing basic tasks on the nation's two leading EHR systems. Some tasks include picking a drug from long drop-down menus. In roughly 1 out of 1,000 orders, physicians accidentally select the wrong drug and roughly 1 in 5 of those could have resulted in patient harm, the researchers found.[4] In a separate study, Dr. Martin Makary, a surgical oncologist at Johns Hopkins, identified the "poor interface design" as the third-leading cause of death in America.

2 Goal

Needless to say, physician burnout propagates medical errors and diminishes the quality of patient care. Instead of mitigating these problems, poorly-designed EHR's are exacerbating the issues. As seen from the previously mentioned studies, there is overwhelming evidence for the need for an improved solution for healthcare professionals to easily navigate and input electronic health record systems. In order to combat physician burnout and minimize EHR-related errors, we seek to build a natural language interface in which physicians can interact with with EHR's.

Our goal is to create a toy model which provides doctors a more intuitive way of interacting patient data. Our natural language interface consists of two parts: Natural Language Understanding (NLU) and Natural Language Generation (NLG).

2.1 Architecture

The interface consist of three layers as shown below in the figure 1.

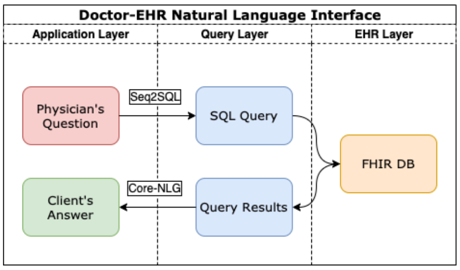


Figure 1: This architecture shows three layers which we will implement to simulate a realistic interaction between a doctor and EHR system.

1. **Application Layer** - defines how the client (doctor) will interact with the interface. This may consist of a web app or chat bot in which the doctor can ask the interface questions about patient data in natural language.
2. **Query layer** - acts an intermediary layer that transposes Physicians' questions into SQL queries (using Seq2SQL) and query results into natural language (using Core-NLG).
3. **EHR layer** - consists of our toy database holding patient data. Fast Healthcare Interoperability Resources (FHIR) specifies a new standard of data formatting for healthcare data. As such, we believe a FHIR-compliant database is the archetype of future EHR data systems.

2.2 Workflow

A typical workflow starts with a physician asking the interface a natural language question. This question is translated into a SQL query. Assuming that the information can be found in the database, the query will then be run against our sample patient database. The FHIR database will contain all patient information including the patient id, age, height, weight, symptoms, diagnosis, notes for check-ins and more. The resulting query response will then be formatted into a sentence using NLG and provided to the client.

2.3 Evaluation

We will test various scenarios against our model and expect our model to handle error gracefully. The four different type of outcomes include:

1. *True Positive* - This situation the model successful interprets and answers the query.
2. *True Negative* - This situation arises when the client asks for information that is not present in the database and model . For example, if a given table or row doesn't exist, then it will be impossible to find such information.
3. *NLU Error* - Error in the translation of a question to SQL query.
4. *NLG Error* - Error in the translation of a query results to an answer.

We hope to reduce the NLU Error and the NLG Error, as they will account for false positives and false negatives.

Note: The prototype will not handle nested questions/queries, therefore, such questions will not be apart of the evaluation.

3 Achievability

Code frameworks:

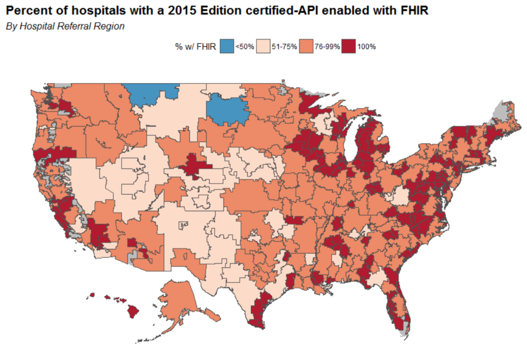
- Python3
- PyTorch
- Stanford Stanza
- Core-NLG
- SQLNet

Computational Resources: Using pre-trained models, only need PC to create training/testing environment.

3.1 Data/Resources

- Where can I get sample EHR data?
  - https://github.com/smart-on-fhir/sample-patients
  - MedNLI
- Data storage in Fhirbase (DB): https://fhirbase.aidbox.app/getting-started
- Seq2SQL dataset: WikiSQL dataset

3.1.1 Why FHIR?


Percent of hospitals with a 2015 Edition certified-API enabled with FHIR
By Hospital Referral Region
% w/ FHIR
<50%
51-75%
76-99%
100%

Source: CHPL; Medicare EHR Incentive Program
Notes: (1) gray areas = HRR with no hospital; (2) The most recent attestations to the Medicare EHR Incentive Program were used to determine EHR installations for all hospitals. These attestations may not reflect the most currently installed technology for all hospitals.

Figure 2: Proliferation of FHIR-compliant EHR systems across Hospitals in US.[5]

The Fast Healthcare Interoperability Resource (FHIR) is a standard for describing data formats and elements (known as ”resources”) and an API for exchanging electronic health records. Examples of resources, include Patient, Medication, Observation, etc. There has been an aggressive push to create this standard such that patient data can be easily exchanged between hospitals, researchers, and other health applications. As seen in figure 2, FHIR protocol has reached an 82% adoption rate nationwide and is the new gold standard for formatting electronic health records.

For more information see: http://hl7.org/fhir/

3.2 NLU: Seq2SQL

In 2017, Salesforce Research team published the novel Seq2SQL model which generates structured queries from natural language using reinforcement learning. By applying policybased reinforcement learning with a query execution environment to WikiSQL, Seq2SQL outperforms a state-of-the-art semantic parser, improving execution accuracy from 35.9% to 59.4% and logical form accuracy from 23.4% to 48.3%.[9] There training set consisted of natural language questions and associated SQL queries from the WikiSQL dataset.

Additional studies in 2018 by Wang et al, have demonstrated a ”Transfer-Learnable Natural Language Interface for Databases”. In essence they have created learning one model that can be used as NLI for any relational database. Their approach also outperforms previous NLI methods on the WikiSQL dataset and the model that is learned can be applied to another benchmark dataset without retraining.[10] Our project hopes to build off these two studies a test the efficacy of a transfer-learnable NLI in the context of EHR data.

3.3 NLG: Core-NLG

After exploring the existing NLG implementations, there are two notable open source libraries, RosaeNLG and Core-NLG. Core-NLG is written in python and seems to be the more convenient choice. However, RosaeNLG provide more sophisticated linguistic features likes anaphora, verbs, words and adjectives agreements. Both provide viable solutions.

4 Related Work

Research in creating more efficient EHR systems have existed for a while. Originally, these systems consisted of medical answering systems are geared towards answering doctors’ medical-related questions, but not questions about a patients medical history. In 2011, AskHERMES was published to allow physicians to enter a question in a natural way with minimal query formulation and allows physicians to efficiently navigate among all the answer sentences to quickly meet their information needs. It acted as a search engine for medical related data. In 2015, MEANS was developed to answer user questions from a collection of documents or a database based on semantic search and query relaxation. Both were limited by poor data sets, consisted of MEDLINE articles.

Recently, there have been more powerful data sources and word embeddings. For example in the paper, ”Transfer Learning in Biomedical Natural Language Processing”, Peng et al used BioELMo and BioBERT, a biomedical versions of embeddings from language model ELMo and BERT. These models are pre-trained on 10 million of the most recent PubMed abstracts. Their research has shown that these pre-trained models have boosted performance and Bio-BERT outperforms existing state-of-the-art models.[11] Another recent paper in 2019, MedNLI, curated a dataset of doctors performing a natural language inference task (NLI), grounded in the medical history of patients. They used the past medical history of a clinical note since they found it to be the most informative and created the most substantial performance gains to downstream tasks.[12]

Our project is unique because we hope to create transfer-learnable version of Seq2SQL, specifically for the medical domain. By default, Seq2SQL uses vanilla GloVe embeddings. We want to experiment with the newer medical embeddings (BioELMo and BioBERT) to see if there are any performance gains. Lastly, we hope to show the practicality of our EHR NLI by building a webapp/chatbot and testing our model on a FHIR-complaint database.

References

[1] https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6262585/
[2] https://www.ahrq.gov/prevention/clinician/ahrq-works/burnout/index.html
[3] https://academic.oup.com/jamia/article/26/2/106/5230918
[4] https://khn.org/news/death-by-a-thousand-clicks/
[5] https://healthit.gov/buzz-blog/interoperability/heat-wave-the-u-s-is-poised-to-catch-fhir-in-2019
[6] https://github.com/xiaojunxu/SQLNet
[7] https://github.com/societe-generale/core-nlg
[8] https://github.com/stanfordnlp/stanza
[9] https://arxiv.org/pdf/1709.00103.pdf
[10] https://arxiv.org/pdf/1809.02649.pdf
[11] https://arxiv.org/pdf/1906.05474.pdf
[12] https://arxiv.org/abs/1808.06752