

A Generic Framework: From Clinical Notes to Electronic Medical Records

Hyoil Han¹, Yoori Choi², Yoo Myung Choi¹, Xiaohua Zhou¹, and Ari D. Brooks³

College of Information Science and Technology, Drexel University¹

College of Engineering, Drexel University²

Department of Surgery, College of Medicine, Drexel University³

3141 Chestnut Street, Philadelphia, PA 19104, USA

{[hyoil.han](mailto:hyoil.han@drexel.edu), [ycrc22](mailto:ycrc22@drexel.edu), [ymc22](mailto:ymc22@drexel.edu), [Xiaohua.Zhou](mailto:Xiaohua.Zhou@drexel.edu)}@drexel.edu, Ari.Brooks@DrexelMed.edu

Abstract

Electronic Medical Records are important to manage health data and save lives to improve the quality of service in hospitals. Clinical medical records contain a wealth of information, largely in free-text form. This paper proposes a generic framework to semi-automatically extract and mine data from clinical note, automatically learn patterns for each physician's clinical notes, and automatically populate EMR databases for multi users. In this paper, we also develop a web-based system with a relational database to automatically store data from MEDical Information Extraction (MedIE) system that extracts and mines a variety of patient information with breast complaints from semi-structured clinical records.

1. Introduction

Patient medical records contain a wealth of information that can prove invaluable for the conduct of clinical research. Clinical records are largely maintained in free-text form. Thus, a reliable and efficient method to extract structured information for future data mining from free-text using information extraction techniques may greatly benefit research endeavors. Manual processing of patient information into a database is subject to errors and may not always maintain objectivity, as well as being prohibitively expensive. These are some reasons why electronic medical records have yet to be widely adopted in medical practice.

This paper proposes a generic framework to extract and mine data from clinical notes prepared by many medical doctors. This work is in the line of our previous work [1, 2]. In the previous work, we extracted and mined data from clinical notes, but did not populate an EMR database automatically. The previous work is related to extracting and mining data

from clinical notes prepared by a single medical doctor. In this paper, we implement a system that automatically populates an EMR database from data extracted and mined from clinical notes prepared by a single medical doctor.

We explain the development of extracting information, discovering knowledge, and building a database to store Electronic Medical Records (EMR) that accurately identify desirable information from transcribed consultation notes. A total of 125 separate initial consultation notes were processed and mined by our system. In our previous work [1], results were then compared to a medical student's independent manual processing of the same 125 consultation notes.

The remainder of this paper is organized as follows. Section 2 describes related work. Section 3 presents information extraction from clinical notes for a single physician. Section 4 explains the database and user interface design and population. In section 5 we present a generic framework to build an EMR database for multi-users (i.e., multiple physicians). A short conclusion ends the article.

2. Related Work

A lot of research related to extracting data from web and text is accomplished using wrappers, which basically provide an interface for extracting information from Web pages [3] [4] [5] [6]. A Wrapper finds out the structure of underlying data for each data source. Then the Wrappers are used to extract instances from data sources. Therefore how to build the wrapper is the main point of view in this research.

Ashish and Knoblock in [4] creates wrappers for structured Web pages from individual data sources. They use several HTML tags to find out section headings of Web pages. They also find out the

hierarchical structure of Web pages using simple heuristic rules, such as font size and indentation. The drawback of their approach is that if Web pages change their format, their approach does not work or they should provide a different set of HTML tags to identify section headings. Their approach is too brittle because it depends on specific HTML tags too much and uses simple rules that fit only specific Web page formats. Kushmerick et al. in [5] introduces wrapper induction, which automatically creates wrappers by generalizing from examples. To label examples, they use an "oracle", which consists of a recognizer. Actually the recognizer contains terms for the attributes and can be used across domains, because they contain domain-independent terms. For example, a recognizer has company names. They use their own algorithms to learn the structure of input instances and create wrappers, so that the wrapper can extract data from a new similar type of input pages. This work mainly extracts information from tabular forms. Their work can handle both text pages and html pages.

The wrapper approach has problems when the underlying structure changes. Especially our work is to build a system for extracting and mining data from clinical notes created by multiple physicians. Therefore the underlying structures of clinical notes are various. With this reason, the wrapper approach is not appropriate for our work. On the other hand, the NoDoSE system in [3] is an interactive tool to determine the structure of documents semi-automatically and requires user-defined schemas in advance and user interaction to select an interesting region for data extraction based on the given schema. After a user selects an interesting small region, NoDoSE mines the selected region and infers the grammar for the structure of the region. Then NoDoSE is loaded with the same type of files, identifies similar regions automatically and extracts data from plain text files using the inferred grammar. Our work was inspired from the NoDoSE approach.

One line of research related to ours is Named Entity Recognition (NER) in free-text. Though most NER methods cannot handle medical terms directly, their concepts, such as pattern matching, can be borrowed. General Architecture for Text Engineering (GATE) [7] uses patterns written in regular expressions to implement all its components such as tokenization and named entity recognition. It also provides a Java Annotated Pattern Engine (JAPE) [8], by which users can extend the NER component to identify entities of interest. However, because medical terms are full of synonyms and morphologic variants, ontology is necessary to achieve high extraction accuracy. A research project, "Acquiring Medical and Biological

Information from Text" (AMBIT) [9], led by a research group at the University of Sheffield, aims to build such a large database of medical terminology for information extraction from clinical records. In this particular project, we adopt Unified Medical Language System (UMLS) (<http://www.nlm.nih.gov/research/umls/>) as the domain ontology to identify concepts.

Pattern-based template filling is a common technique for information extraction. AutoSlog [10], PALKA [11], CRYSTAL [12] and WHISK [13] can automatically induce linguistic patterns from training examples. However, supervised pattern learning is very expensive. Instead, we use an unsupervised approach, which makes use of the parsing results of link grammar parser [14], to extract a good portion of knowledge in the project.

3. Text Mining for Clinical Notes

This section presents information extraction from clinical notes for a single physician. First we explain our methodologies to extract and mine clinical notes from a single physician. It consists of three steps: term identification, term association, and term classification.

First, our term identification is based on the publicly available ontology, Unified Medical Language System (UMLS) (<http://www.nlm.nih.gov/research/umls/>). Medical term identification essentially belongs to the task of named entity recognition. However, medical terms are full of synonyms and morphologic variants. It is necessary to adopt an ontology (or dictionary) for high accuracy extraction of medical terms from clinical records. Medical terms are often multi-word phrases; therefore, it is not efficient to search all combinations of sequential words in the sentence through ontology. Instead, we use part of speech tagger [7] and ordered patterns to obtain a list of term candidates and then see if the candidate terms exist in the ontology. The approach is illustrated by the example of extracting past medical history. Ontology-based method for medical term extraction achieves high precision and recall. But it still fails to retrieve a portion of terms of interest simply due to the ontology incompleteness. We relieve the problem by guessing some terms based on the idea that elements in a parallel sentence structure should play the same role. An ontology-based approach for medical term identification is much better than general named entity recognition approaches in terms of precision and recall. However, it requires intensive searching, though we adopt ordered part of speech patterns to minimize the number of term candidates.

Second, The concept association is comprised of two steps. The first is identification of all concepts including diseases, symptoms, human body parts, persons, numbers, dates, etc. The second is the association of concepts. Extraction of persons, dates, and numbers is not difficult. In fact, most natural language processing (NLP) development tools, such as GATE [7], provide the module of named entity recognition modules, which annotate above-mentioned concepts in text with extremely high precision and recall. The concept association is achieved in two ways: pattern-based approach and graph-based approach. The major advantage of the pattern-based approach lies in its simplicity. However, it is difficult to exhaust all patterns since the expression of natural language is so flexible. The graph-based approach utilizes the graph of the Linkage diagram obtained from Link Grammar parser [14] (<http://www.link.cs.cmu.edu/link/>). Link Grammar is an original sentence parser, producing not only a constituent tree as most parsers yield, but also a linkage diagram that consists of links between two words.

Last, term classification is accomplished by utilizing decision tree. Text classification is another type of information extraction tasks in our project. For high accuracy, an analytic NLP approach is recommended by most of the literature. Usually pattern-based semantic analysis would be performed to classify cases. However, the analytic approach highly demands large amounts of domain knowledge and is consequently difficult to generalize. Conversely, a machine learning technique does not depend on domain knowledge and the approach can easily be generalized. In this project, we employed an ID3-based decision tree [15] for categorical fields. According to information theory, Information Gain (Mutual Information) of the predictor and dependent variable is a good measure of the predictor's discriminating ability. Thus, the ID3 decision tree is supposed to use less features than other decision tree algorithms.

The performance (i.e., precision and recall) of information extraction using concept association is 86-100% [1]. The performance of term classification is 89-93.7% [1]. The precision of medical term extraction is 88-96.7% and its recall is 89-96.7% [1].

The precision and recall for i-th patient are defined respectively as:

$$R_i = \frac{ETrue_i}{TInst_i} \quad P_i = \frac{ETrue_i}{ETotal_i}$$

Precision and recall for all cases, respectively, are defined as below:

$$R = \frac{\sum_i ETrue_i}{\sum_i TInst_i} \quad P = \frac{\sum_i ETrue_i}{\sum_i ETotal_i}$$

Where:

$ETrue_i$: number of extracted true terms in i-th subject.

$ETotal_i$: number of extracted terms in i-th subject.

$TInst_i$: number of total true terms in i-th subject.

4. Database and User Interface Design

This section describes our database design and web-based user interface design for our EMR database. The database is designed with nine tables for the data extracted from the clinical notes. The nine tables include patient, presentation, patient_history, past_surgical_history, drug and alcohol use, vitals, radiology, gynecologic_history, and breast_history with 39 attributes. For the attribute names in each table, National Cancer Institute Terminology Resources: NCI Thesaurus (<http://www.cancer.gov/cancertopics/terminologyresources>). Our database was designed manually, but was populated automatically using Java Database Connectivity (JDBC) techniques. In this way, we can populate EMR database automatically.

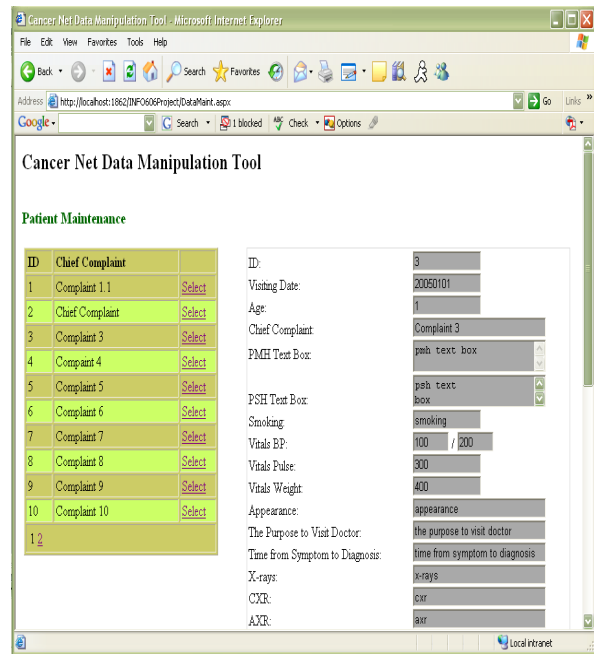


Figure 1. A Web-based user interface for displaying data.

The use interface is designed to help users to update and display EMR data through the web interface. Figure 1 shows part of web-based user interface implemented in our current system.

Also users can input and update data through the web-based user interface. Figure 2 shows the snapshot of data entry using the form. As shown in red, when user tries to enter invalid format, the system will generate an error message to prompt the user to re-enter the valid data. The attributes ID, PRESENTATION_AGE, BLD_PRESS_SYSTOLIC, BLD_PRESS_DIASTOLIC, PULS_RATE, WT_LB_NUM, PREG_AGE_B_FCHLD_CAT, PT_MENARCHE_AGEYR, PT_PREG_CT, and PREG_LIVEBIRTH_CT should be numeric.

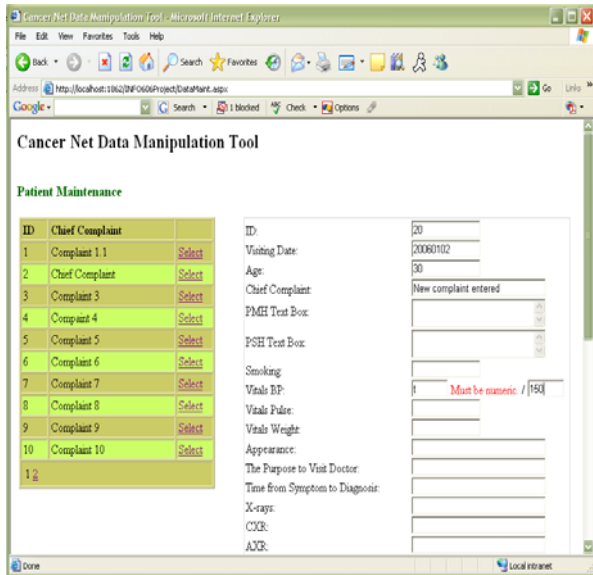


Figure 2. A Web-based user interface for data entry.

5. A Generic Framework for EMR from Clinical Notes for Multiple Physicians

A generic framework is proposed to generate EMR from clinical notes created by multiple physicians. The generic framework includes modules to semi-automatically extract and mine data from clinical notes, automatically learn patterns for each physician's clinical notes, and automatically populate EMR databases. Figure 3 shows the proposed generic framework. The work to build the generic infrastruc is

an ongoing project. We current implemented part of the framework as explained in sections 3 and 4.

Per physician, clinical notes are used for testing data. Here clinical notes from more than one physician are not mixed to input to annotation of concepts and instances. UMLS is used as an ontology for annotation of concepts. In the annotation of instances, we utilize methodologies such as term identification, graph-based concept association and decision tree techniques in section 3. Once annotation of concepts and instances is completed for a single physician, the system requires that the physician review manually if her/his clinical notes are correctly annotated with concepts and their corresponding values. The reason to introduce user intervention is to increase the accuracy of learned patterns because the patterns will be automatically generated based on the annotations. The accuracy of learned patterns can be degraded if the annotation is not correct. The user can fix any incorrect annotation through graphic user interface (GUI) to improve the procedure of user intervention.

Per physician, patterns are automatically learned through machine learning and natural language processing techniques and stored in a database called RuleBase shown in Figure 3. The RuleBase contains patterns (or rules) for each physician. For the rules of each physician in the RuleBase, her/his clinical notes are used as testing data and can be automatically processed to populate EMRs.

6. Conclusions

In this paper, we presented the automatic population of an EMR database and proposed a generic framework to semi-automatically extract and mine data from clinical notes, automatically learn patterns for each physician's clinical notes, and automatically populate EMR databases for multi-users. Our previous work [1, 2] was only for clinical notes from a single physician. The generic framework to build EMRs for clinical notes from multiple physicians is an ongoing project that is partially implemented as explained in this paper.

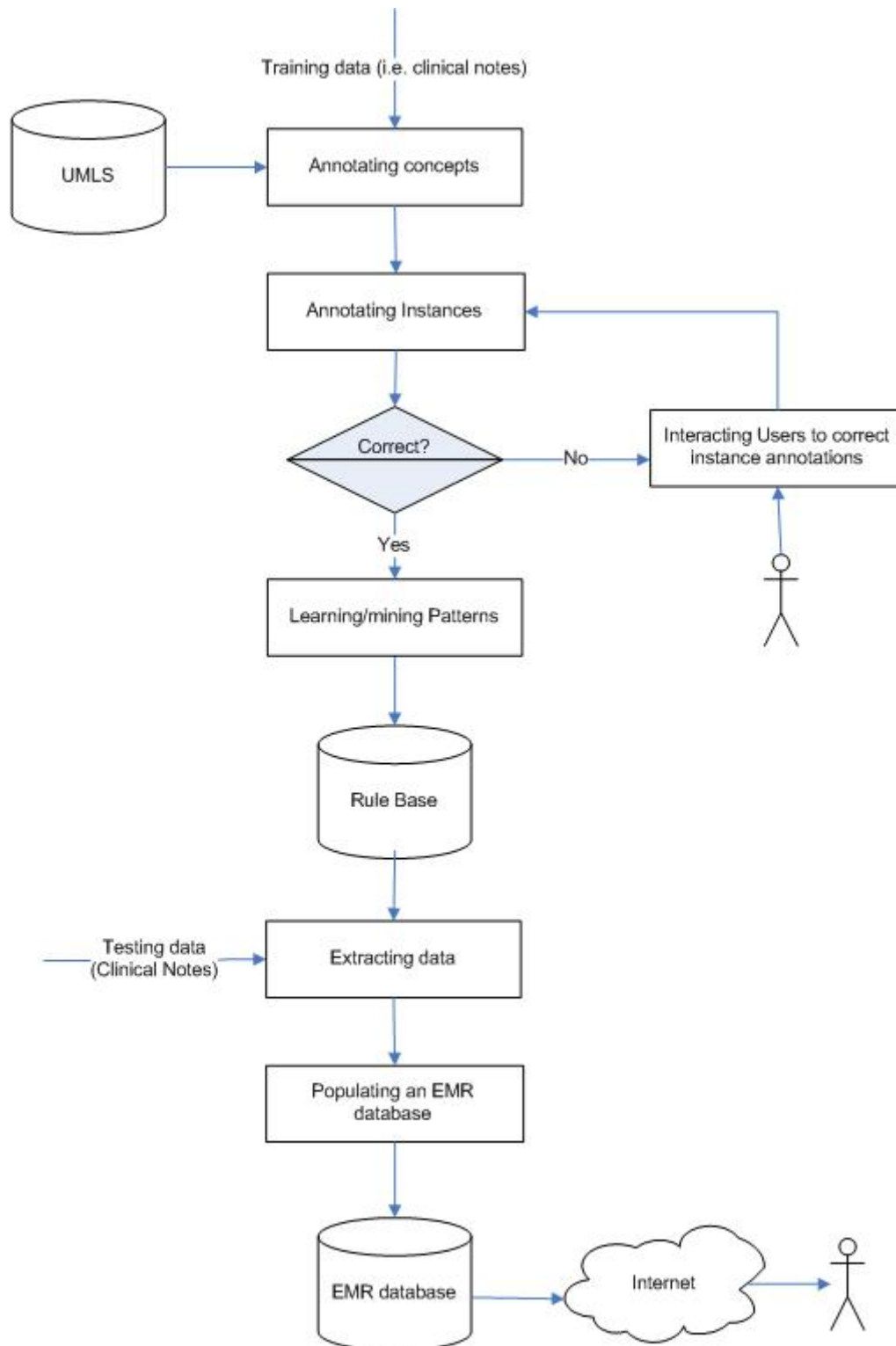


Figure 3. A generic framework to generate EMR from clinical notes prepared by multiple users (i.e., multiple physicians)

References

- [1] X. Zhou, H. Han, I. Chankai, A. A. Prestrud, and A. D. Brooks, "Approaches to Text Mining for Clinical Medical Records," presented at The 21st Annual ACM Symposium on Applied Computing 2006, Technical tracks on Computer Applications in Health Care, Dijon, France, 2006.
- [2] X. Zhou, H. Han, I. Chankai, A. A. Prestrud, and A. D. Brooks, "Converting Semi-structured Clinical Medical Records into Information and Knowledge," presented at International Workshop on Biomedical Data Engineering (BMDE) 2005 in conjunction with the 21st International Conference on Data Engineering (ICDE 2005), Tokyo, Japan, 2005.
- [3] B. Adelberg, "NoDoSE - A Tool for Semi-Automatically Extracting Structured and Semistructured Data from Text Documents," presented at ACM SIGMOD, 1998.
- [4] N. Ashish and C. A. Knoblock, "Semi-automatic Wrapper Generation for Internet Information Sources," presented at Cooperative Information Systems (CoopIS), 1997.
- [5] N. Kushmerick, "Wrapper Induction for Information Extraction," in *Computer Science*. Seattle, WA: University of Washington, 1997.
- [6] I. Muslea, S. Minton, and C. Knoblock, "A Hierarchical Approach to Wrapper Induction," presented at Conference on Autonomous Agents, Seattle, WA., 1999.
- [7] H. Cunningham, D. Maynard, K. Bontcheva, and V. Tablan, "GATE: A Framework and Graphical Development Environment for Robust NLP Tools and Applications," presented at 40th Anniversary Meeting of the Association for Computational Linguistics (ACL'02), 2002.
- [8] H. Cunningham, D. Maynard, and V. Tablan, "JAPE: A Java Annotation Patterns Engine (Second Edition). Technical report CS--00--10, University of Sheffield, Department of Computer Science.," vol. 2005, *Tertiary JAPE: A Java Annotation Patterns Engine*, 2000.
- [9] R. Gaizauskas, M. Hepple, N. Davis, Y. Guo, H. Harkema, A. Roberts, and I. Roberts, "AMBIT: Acquiring Medical and Biological Information from Text," presented at ISMB/ECCB, Poster, 2004.
- [10] E. Riloff, "Automatically Constructing a Dictionary for Information Extraction Tasks," presented at the Eleventh National Conference on Artificial Intelligence, 1993.
- [11] J. T. Kim and D. I. Moldovan, "Acquisition of Linguistic Patterns for Knowledge-Based Information Extraction," *IEEE Transactions on Knowledge and Data Engineering*, vol. 7, 1995.
- [12] S. Soderland, D. Fisher, J. Aseltine, and W. Lehnert, "CRYSTAL: Inducing a Conceptual Dictionary," presented at the Fourteenth International Joint Conference on Artificial Intelligence, 1995.
- [13] S. Soderland, "Learning Information Extraction Rules for Semi-structured and Free Text," *Machine Learning*, 1999.
- [14] D. a. T. D. Sleator, "Parsing English with a Link Grammar," presented at Third International Workshop on Parsing Technologies, 1993.
- [15] J. R. Quinlan, "Induction of Decision Trees," *Machine Learning*, vol. 1, pp. 81-106, 1986.