Distribution Agreement

In presenting this thesis or dissertation as a partial fulfillment of the requirements for an advanced degree from Emory University, I hereby grant to Emory University and its agents the non-exclusive license to archive, make accessible, and display my thesis or dissertation in whole or in part in all forms of media, now or hereafter known, including display on the world wide web. I understand that I may select some access restrictions as part of the online submission of this thesis or dissertation. I retain all ownership rights to the copyright of the thesis or dissertation. I also retain the right to use in future works (such as articles or books) all or part of this thesis or dissertation.

Signature:

_________________________   ________________.

Shuai Zheng   Date

Online Learning Based Clinical Information Extraction and Classification

By

Shuai Zheng
Doctor of Philosophy

Computer Science and Informatics

James Lu, Ph.D.
Advisor

Fusheng Wang, Ph.D.
Co-Advisor

Jinho Choi, Ph.D.
Committee Member

Raymund Dantes, M.D.
Committee Member

Accepted:

Lisa A. Tedesco, Ph.D.
Dean of the James T. Laney School of Graduate Studies

Date

Online Learning Based Clinical Information Extraction and Classification

By

Shuai Zheng

M.S. Wake Forest University, 2010

Advisor: James Lu, Ph.D.

Advisor: Fusheng Wang, Ph.D.

A abstract of
A dissertation submitted to the Faculty of the
James T. Laney School of Graduate Studies of Emory University
in partial fulfillment of the requirements for the degree of
Doctor of Philosophy
in Computer Science and Informatics
2015

Abstract

Online Learning Based Clinical Information Extraction and Classification

By Shuai Zheng

To enable the research use of clinical reports, pertinent data needs to be extracted from narrated medical reports. Traditional automated based approaches are brittle and do not have the ability to take user interaction as feedbacks for improving the extraction algorithm in real time.

In this dissertation, we present an interactive, online machine learning based system, IDEAL-X, that addresses some key shortcomings of existing systems. IDEAL-X provides a standard interface that can be used for simple data extraction and data entry. It is unique, however, in its ability to transparently analyze and quickly learn, from users' interactions with a small number of reports, the desired values for the data fields. Additional user feedback (through acceptance or edits on system generated values) incrementally refines the decision model in real-time, which further reduces the user's burden in processing subsequent reports. Extensive experiments in multiple use cases show that the system achieves high accuracy on data extraction with minimal effort from users. The system also accepts predefined domain knowledge, in the form of controlled vocabulary, to improve the efficiency and accuracy of data extraction. The system contains components for standardizing and querying extracted values. Moreover, an online learning based classification module can be used to support clinical decision making. We report successful applications of IDEAL-X to extract data in Emory Cardiology, Pathology, and the Centers for Disease Control and Prevention.

Online Learning Based Clinical Information Extraction and Classification

By

Shuai Zheng

M.S. Wake Forest University, 2010

Advisor: James Lu, Ph.D.
Advisor: Fusheng Wang, Ph.D.

A dissertation submitted to the Faculty of the
James T. Laney School of Graduate Studies of Emory University
in partial fulfillment of the requirements for the degree of
Doctor of Philosophy
in Computer Science and Informatics
2015

Acknowledgments

I am thankful for all the teachers, collaborators, staffs and classmates who have helped me unreservedly. They have made my Ph.D. study enjoyable and my days at Emory memorable. I am extremely grateful to my advisor, Dr. James Lu, who leads and mentors with creativity, dignity and kindness. I sincerely appreciate my co-advisor Dr. Fusheng Wang for his kind support and help throughout my study and research. Thanks to Dr. Jinho Choi for his invaluable advices and insightful instructions. And special thanks to Dr. Raymund Dantes for guiding me with expertise and precious career advice. Without them, this dissertation and would not have been possible.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

1.1.1 Clinical Data

While tremendous efforts have been made to enable structured reporting for electronic medical records system, a large amount of medical data remain in free-form narrative text reports. Moreover, most existing medical report systems are based on natural language narrations written in free-form text. While some high-level structures exist --- for example, patient records may contain sections on "Medications on Admission", "Hospital Course" and "Condition on Discharge" --- the narrative style of each section is often highly informal and personal. For instance, each of the following patient record snippets describes similar "heart rate" information of patients during physical examination (See Figure 1.1). As a consequence, useful research data from individual patients are usually obscured and distributed across reports of multiple types with heterogeneous structures and vocabularies.

1. *"...... Blood pressure was 152/63 , heart rate 67 with occasional premature ventricular contractions, respirations 15 ......"*
2. *"......Her pulse was regular at 82 beats per minute......"*
3. *"......The blood pressure was 115/73 and heart rate was 93......"*
4. *"......122/66 , 96.8 , 81 , 21 for vital signs......"*

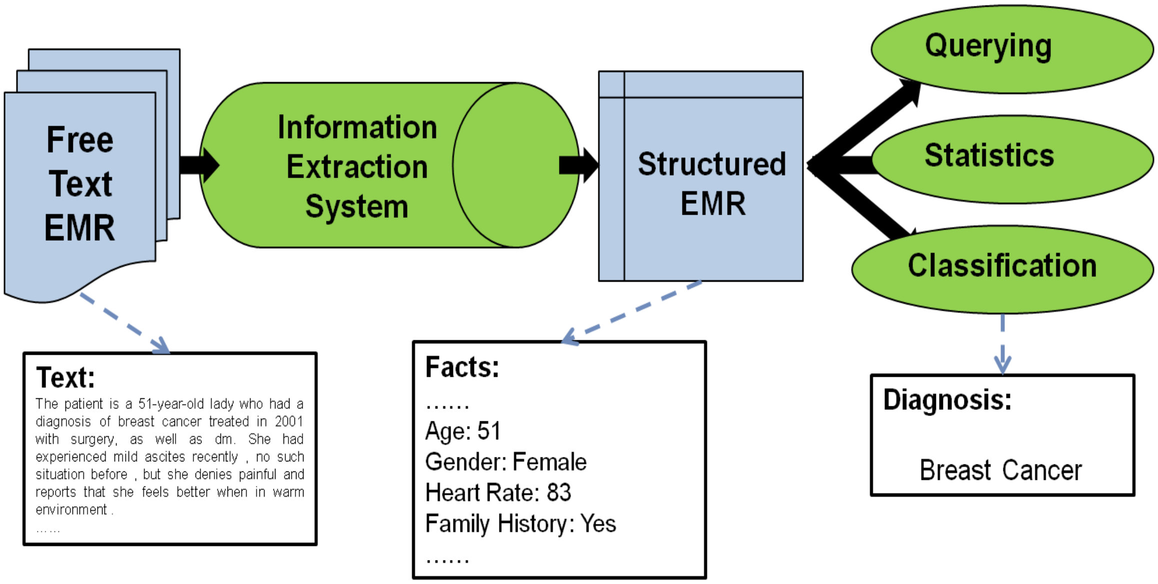
Figure 1.1. Snippets of clinical free text narrations

1.1.2 Structured Reporting

Compared to traditional free-form text reporting system, "structured reporting" [1] offers significant promise for both human consumption and machine processing. Report standards, controlled vocabularies and terminologies have proliferated in medical domains to standardize the creation of medical reports. DICOM structured reporting standardizes reporting in radiology images [2]. Structured Reporting for Anatomic Pathology [3] is under development by IHE for standardized structured pathology reporting. Such structured reporting often depends on pre-defined templates or vocabularies. Examples include the College of American Pathologist's (CAP) cancer protocols and checklists [4], which provide detailed checklists for options in pathology reports. The Cancer Biomedical Information Grid (caBIG) [5]'s Cancer Data Standards Registry and Repository (caDSR) [6] offers APIs and tools to define common data elements (CDEs). HL7 Clinical Document Architecture [7] defines standardized document structure and semantics to share electronic health information. These efforts to standardize medical reporting formats and vocabularies give rise to the possibility for automated searching, browsing, and mining of medical data.

1.1.3 Clinical Data Processing

Despite the obvious benefits of structured reports, majority of EMR systems still allows for (thus encourages) narrative text narration. To facilitate large-scale research and clinical use of data embedded in narrative text, pertinent data needs to be identified first. Figure 1.2 shows a typical pipeline for processing medical reports: valuable facts are extracted from free text EMR either manually or with automated information extraction system. Diagnosis is then realized through human effort or software tools, is conducted based on extracted structured data.



The patient is a 51-year-old lady who had a diagnosis of breast cancer treated in 2001 with surgery, as well as dm. She had experienced mild ascites recently , no such situation before , but she denies painful and reports that she feels better when in warm environment .
……

……
Age: 51
Gender: Female
Heart Rate: 83
Family History: Yes
……

Breast Cancer

Figure 1.2 Information extraction and classification pipeline for medical reports

Next, we discuss traditional approaches and their limitations.

1.1.4 Traditional Approaches

For large datasets, manual extraction and classification are tedious, time-consuming and error-prone. Automated tools, on the other hand, are typically tuned for particular domains, and precisely annotated training datasets need to be developed and learned to establish decision models through which subsequent reports can be processed. Once trained, the decision models of extraction and classification are difficult or impossible to modify and improve. Therefore, these approaches lack the capability of taking user feedbacks, to adapt and improve the extraction or classification algorithm in real time. In addition, automatically extracted data suffers from inaccuracies as a result of natural limits on statistical machine learning techniques.

For healthcare research that requires completely accurate data, the above discussion points to one constant: human involvement is necessary, whether to perform the actual extraction or for post-extraction verification. In this thesis, we consider an online machine learning approach.

1.1.5 Online Learning Based System

Our goal is to provide a generic information extraction and classification framework that is adaptable to diverse clinical reports, enables a dynamic interaction between the human and machine, and produces highly accurate results with minimal human effort. We have developed a system, IDEAL-X (Information and Data Extraction using Adaptive Online

Learning), to support adaptive information extraction from diverse clinical reports with heterogeneous structures and vocabularies. A **demo video** can be found in YouTube [8].

IDEAL-X takes an online machine learning approach that integrates machine learning with interactive human intervention, and combines it with customizable vocabularies. The data extraction and classification engines can automatically predict answers to annotate or classify reports, gradually learn from human feedback, and iteratively improve its accuracy. The online learning algorithms [9-11] make predictions one report at a time, and utilize human feedback to update its prediction model immediately. Compared to traditional batch training based algorithm, which requires training with large volumes of carefully annotated data before deploying a system, online learning based algorithms can avoid costly expense of preparing training data and also render the possibility of updating the deployed system in a dynamically changing data environment.

The online machine learning based approach is enabled through an intuitive interface and workflow. The system provides an interactive interface for users to view and annotate reports, and to verify answers generated by the data extraction or classification engine. When a user makes manual annotations or revises system-generated answers on a report, IDEAL-X learns important linguistic features and patterns of the values to be extracted in subsequent reports. The process is repeated, and the knowledge accrued by the system helps it to improve the extraction or classification accuracy. In turn, this reduces the user's effort over time.

Besides online machine learning, IDEAL-X allows for customizable controlled vocabularies to support data extraction from clinical reports, where a vocabulary enumerates the possible values that can be extracted for a given attribute. (The X in IDEAL-X represents the controlled vocabulary plug-in.) The use of online machine learning and controlled vocabularies is not mutually exclusive; they are complementary, which provide the user with a variety of modes for working with IDEAL-X.

1.2 Research Contribution

Important contributions of this work are as follows.

i) We develop various online learning algorithms to support information extraction and classification. The algorithms form the foundation for IDEAL-X, and experiments reveal the contextual challenges of applying online learning in clinical environments.

ii) We present a workflow and software interface to facilitate a human-machine collaboration aimed at balancing accuracy and efficiency. User may also input domain specific knowledge to improve the system's learning efficiency and prediction accuracy.

iii) We investigate existing clinical knowledge bases, design interface and template for controlled vocabulary. We demonstrate the important role that controlled vocabulary plays in information extraction and data standardization, and present solutions to assist in building domain-specific vocabulary.

iv) We demonstrate the feasibility of IDEAL-X in three real-world uses cases. Working with Emory Clinical Cardiovascular Research Institute, we extract information from heterogeneous clinical reports to support Biobank building, which addresses a variety of research questions in cardiovascular disease. Working with Emory Pathology, we extract diagnosis and genetic information from pathology reports to support patient group identification. Working with the Centers for Disease Control and Prevention, we improve VTE surveillance with information extracted from radiology reports.

1.3 Potential Usages

Compared to free-form text, strictly formatted data is not only more feasible for browsing, but also suitable for further data processing such as querying, statistical analysis, and reasoning to support decision making. For many informatics systems, extracting information from unstructured text is an indispensable procedure. Examples include the following.

I) Clinical reporting and billing system

Most existing medical report systems are based on natural language narrations written or dictated in free-form text. The wide demand for extracting research data from text reports in the medical/health-care field makes IDEAL-X particular relevant. The system can extract information from narrative or dictation text to generated structured forms, which may be further normalized into standardized code or template with medical standards such as ICD-9 or UMLS [12]. Standardized clinical data is extremely feasible for billing purpose and other administration usage in a hospital.

II) Patient Identification and Surveillance

By combining extracted data, such as diagnosis with demographics information, physicians can index patients and identify patient group of specific research interest. When further integrated with temporal and geographic background information, surveillance system can be implemented to detect and monitor outbreak and pandemic phenomenon of certain disease.

III) Clinical decision support system (CDSS)

In an increasingly data-driven world, a preliminary procedure of clinical decision support system [13-15] is to identify and annotate valuable attributes from free-form text [16, 17]. Strictly formatted data, which integrates information from various clinical reports, provides a comprehensive view of the patient's information. With this view, diagnosis could be determined either by physician or automated clinical decision support system (CDSS) based on data mining and statistics technologies, such as clustering and classification.

IV) Medical records de-identification system

De-identifying medical reports and patient records [18-21] can facilitate health information sharing. To free-form text, the first step of concealing patient's personal information is to locate and annotate sensitive attributes [22], especially the 18 data elements designated by HIPPA. This information extraction framework can also be adopted to support de-identification, so as to make clinical data feasible for sharing to advance clinical research.

V) Annotation system

Annotating adequate text to create a training corpus is essential to most natural language processing oriented project. Some annotation tools provides machine-assisted features [23, 24], however, these assistances are still very limited and not intelligent enough. Having a system that can semi-automatically assist in finding and highlighting potentially relevant pieces of information can speed up the annotation work, therefore, facilitates the development of other NLP technologies and systems.

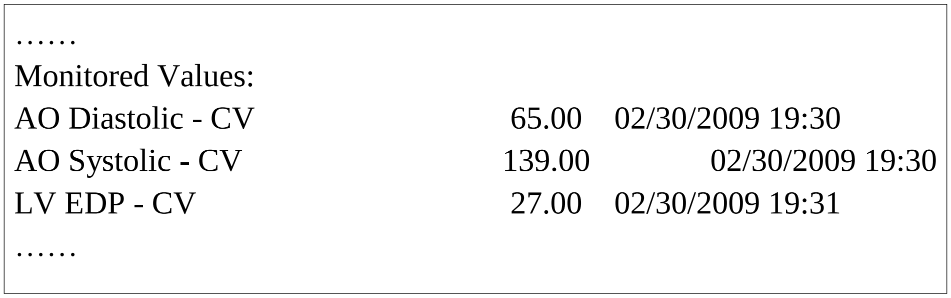
Chapter 2

Background and Related Work

2.1 Background

A patient’s electronic medical record includes a variety of medical reports. Data in these reports provides critical information that can be used to improve clinical diagnosis and support biomedical research. For example, the Emory University Cardiovascular Biobank [25] collects records of patients with potential or confirmed coronary artery diseases undergoing cardiac catheterization, and aims to combine extracted data elements from multiple reports to identity patients for research. Report types include history and physical report, discharge summary, outpatient clinic note, outpatient clinic letter, coronary angiogram report, cardiac catheterization procedure report, echocardiogram report, inpatient report, and discharge medication lists.

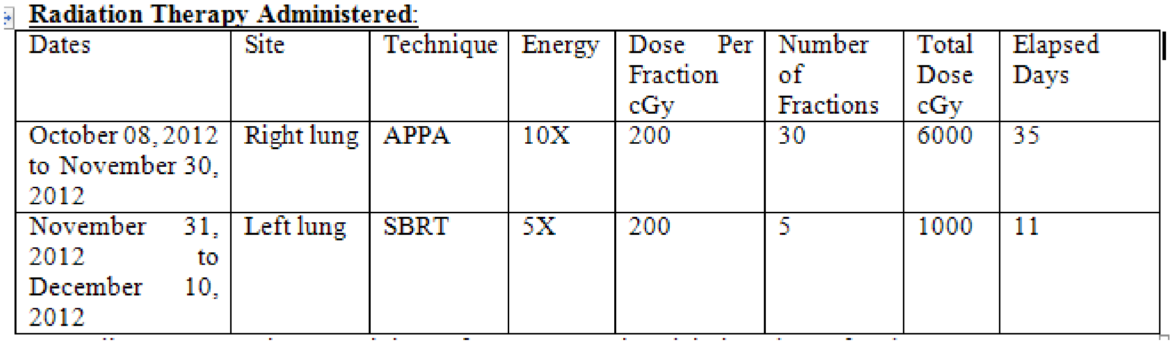
In increasing level of difficulty for data extraction, a clinical report is usually a mixture of semi-structured data, tabular based text, template based narration, and complex narration.



```
……
Monitored Values:
AO Diastolic - CV                    65.00    02/30/2009 19:30
AO Systolic - CV                    139.00             02/30/2009 19:30
LV EDP - CV                          27.00    02/30/2009 19:31
……
```

Figure 2.1 A snippet of semi-structured report

Semi-structured data, as demonstrated in Figure 2.1, describes the value of data elements in the form of Attribute/Value pair. This format is highly readable and may be generated from a database directly. However, processing such data still requires an extraction tool as the original structure may not be preserved.



Radiation Therapy Administered:

Dates	Site	Technique	Energy	Dose Per Fraction cGy	Number of Fractions	Total Dose cGy	Elapsed Days
October 08, 2012 to November 30, 2012	Right lung	APPA	10X	200	30	6000	35
November 31, 2012 to December 10, 2012	Left lung	SBRT	5X	200	5	1000	11

Figure 2.2 A snippet of a report with tabular based text

Tabular based narration, as demonstrated in Figure 2.2, presents information in the form of tables. Similar to semi-structured data, it renders great human readability. But for the computer to interpret it properly is still a challenge.

The coronary territory consisting of the mid/distal left anterior descending coronary artery and diagonal branches contained luminal irregularities. There is 90 % stenosis in the mid left anterior descending coronary artery.
……

Figure 2.3 A snippet of a report with template based narration

Template based narration, as demonstrated in Figure 2.3, is a very common report form. The narrative style including its sentence patterns and vocabulary is consistent, direct, and resembles the use of the same template across different records. To extract information from this type of text (e.g., “mid left anterior descending artery”), specific linguistics based rules or constrains need to be applied. This may require non-trivial NLP expertise, especially when the number of attributes to be extracted is large.

……
The patient is a healthy 51-year-old lady who had a diagnosis of breast cancer treated in 2001 with surgery. She had experienced mild ascites recently, but she denies painful and reports that she feels better when in warm environment.
……

Figure 2.4. A snippet of complex narration report

Complex narration is essentially free-form text. Compared to template based narration, it often contains discourse that is informal and personal, and the narrative patterns are diverse. It is the most difficult narration style to interpret and process by NLP algorithms. Certain types of information, such as diseases and medications, can be extracted with high accuracy with the aid of a controlled vocabulary.

IDEAL-X implements a solution for each of the above report format.

2.2 Important Problems of Clinical NLP

NLP issues that play an important role in medical information extraction are summarized as follows.

I) Negation Detection

In medical reports, the meaning of concept may be reversed by negative terms for example, "the patient has no history of diabetes", the term "diabetes" should not be extracted as a history of a disease.

II)Uncertainty Detection

Similar to negation terms, uncertain terms also alter the meaning of a sentence, for example, “the patient is planned to take radiation therapy” means the “radiation therapy” hasn’t been conducted yet.

III) Timex Detection

As temporal information [26-28] plays an important role in medical reports, it would be critical to extract temporal information so as to identify important medical events.

IV) Standardization

In medical domain, the value to be extracted or processed usually have considerable variances and abbreviation. For instance, "DM" is the abbreviate of diabetes, and "Pindolol", is a hyponym of beta-blocker.

2.3 Related Work

2.3.1 Clinical Information Extraction

A number of research efforts have been conducted in the field of medical information extraction. cTAKES (clinical Text Analysis and Knowledge Extraction System) [29] is an open-source NLP system designed for extracting information from clinical text. It offers various NLP tools trained especially for clinical fields. Most algorithms or systems focus on a particular application domain such as pathology reports [30, 31]or biomedical text [32, 33]. caTIES [34] (Cancer Text Information Extraction System) is a cancer text information extraction system specialized in tissue annotations. MedEx [35] extracts medications and related information such as dosage and duration. ONYX [36] adapts semantically annotated grammar rules to analyze sentence level text. MedLEE [37] (Medical Language Extraction and Encoding system) is a clinical information extraction system that offers the feature of mapping information to controlled vocabularies.

These automated tools have noticeable shortcomings.

1) Both rule and machine learning based approach are domain-specific and not easily adaptable to new reports. In addition, once implemented, the decision model remains static and is difficult to improve.

2) In machine learning based approaches, a precisely annotated training dataset is required. Such a training set can be difficult and expensive to obtain.

3) In rule based approaches, rules need to be manually engineered by clinical domain experts and linguistic professional.

4) Automated systems cannot produce completely accurate results due to either incomplete rule coverage or the nature of statistical machine learning, especially in noisy settings.

2.3.2 Clinical Decision Support

Clinical Decision Support is one of the most important applications for information extracted from clinical reports. Machine learning algorithms, especially classification algorithms, have been widely used in clinical domain to support medical decision making [38]. Medical decision support system was used for heart disease diagnosis [39], neuromuscular disorders detection[40], fetal well-being [41] and cancer [42, 43]. Apart from disease or disorder diagnosis, it also has been applied to support clinical procedures in hospital, for example examining the performance of physician and patient outcome [44], and providing support for antibiotic prescribing [45].

Similar to traditional information extraction approaches, most existing clinical decision support systems and algorithms lack the ability to accommodate streaming data --- typical in a clinical environment, and are thus not easily adaptable.

2.3.3 Online Machine Learning

Online machine learning provides customizations for different working environments and supports incremental improvement. Amilcare [46] is an adaptive information extraction system used for Semantic Web annotation. Its algorithm, $(LP)_2$[47], generalizes and induces symbolic rules. DUALIST [48] allows users to select system populated rules for feature annotation to support text classification, word sense disambiguation and information extraction. Another related research area is interactive annotation, which attempts to ease the annotation process by incorporating machine learning techniques. MIST [49]'s classifier automatically learns to support de-identification. RapTAT [50] learns document phrases to accelerate annotation.

Different from online machine learning, which processes instances in sequence, active learning [51, 52] is semi-supervised, with the ability to retrieve labels for most informative data points by inquiring users actively. Example applications in healthcare informatics include word sense disambiguation [53] and phenotyping [54].

2.3.4 Contextual Feature Extraction

Many researches emphasize the value of special contextual features within text. Detecting negation expression and the affected scope [55-60] is very important to clinical report since it may overturn the meaning of a statement. Temporal feature provides critical information for medical reasoning and decision [61-67]. It represents medical encounters with a timeline. Some other researches value the importance of medical events, which are modeled based on medicine procedure, time and negative information [65, 68-71].

Chapter 3

IDEAL-X: The User Perspective

3.1 Goals

We aim at a system that supports convenient and intelligent data extraction from different types of reports using knowledge learned from human interaction during ordinary manual data extraction. Towards that end, the development of IDEAL-X adheres to the following design goals:

Ease of use: The system should have a low learning curve for new users. Interactions between a user and the system should follow (or minimally deviate from) the conventional process for manual extraction/classification and the input of domain knowledge should be easy to interpret and construct.

Domain agnostic: The system should be easily adaptable to different types of reports and clinical environments. Thus, the system should employ designs and algorithms that are problem neutral.

High accuracy: The system should ensure high accuracy to meet the rigorous requirement of clinical research or healthcare quality improvement. Besides the online learning based method, the system should allow domain-specific knowledge such as customized vocabularies, when available, to be incorporated into the decision model to amplify the system's performance.

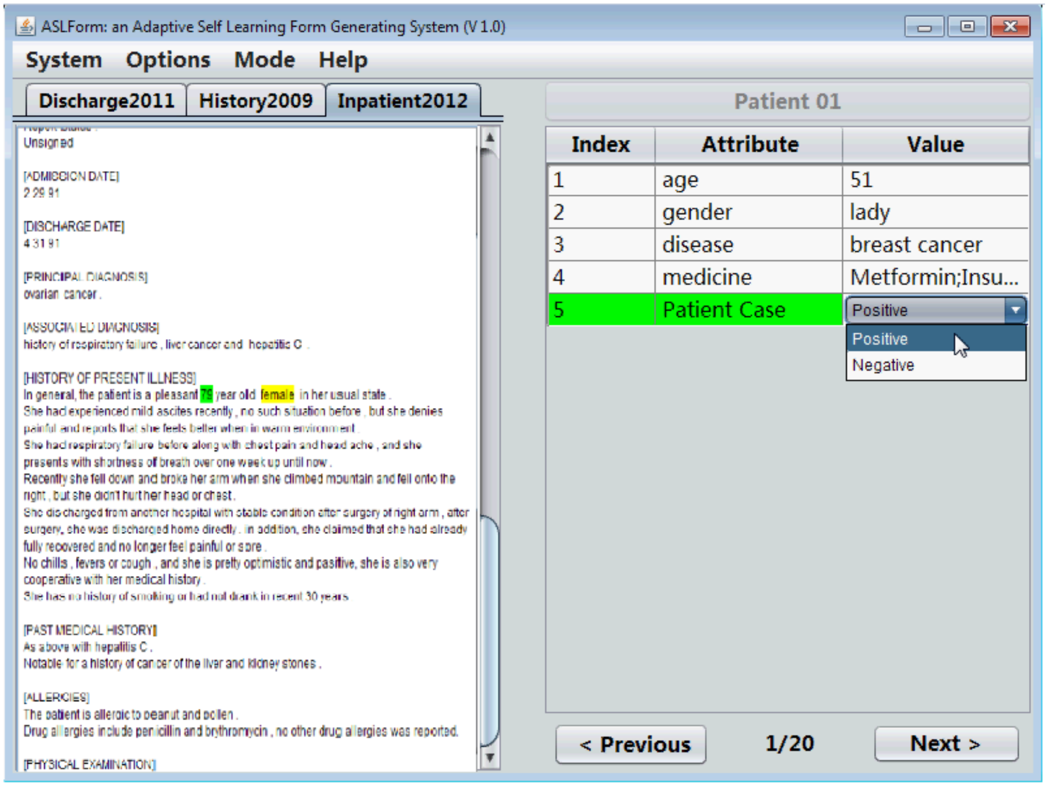
High efficiency: The system should maintain a consistent, crisp response time to each document, regardless of the number of documents in the input collection or have been processed.

In sum, all these goals raise constraints and challenges to system development and algorithm design.

3.2 Human-Computer Interaction

Human-Computer interaction (HCI), as a field, aims to optimize the interface between human and computer [72]. To online learning based systems that rely on user feedbacks for improving predictions, the quality of the interface for collecting user feedback most directly affects the user experience.

We design interface and operations with the following features:



Unsigned

[ADMISSION DATE]
2 29 91

[DISCHARGE DATE]
4 31 91

[PRINCIPAL DIAGNOSIS]
ovarian cancer .

[ASSOCIATED DIAGNOSIS]
history of respiratory failure , liver cancer and hepatitis C .

[HISTORY OF PRESENT ILLNESS]
In general, the patient is a pleasant 79 year old female in her usual state .
She had experienced mild ascites recently , no such situation before , but she denies painful and reports that she feels better when in warm environment .
She had respiratory failure before along with chest pain and head ache , and she presents with shortness of breath over one week up until now .
Recently she fell down and broke her arm when she climbed mountain and fell onto the right , but she didn't hurt her head or chest .
She discharged from another hospital with stable condition after surgery of right arm , after surgery, she was discharged home directly . in addition, she claimed that she had already fully recovered and no longer feel painful or sore .
No chills , fevers or cough , and she is pretty optimistic and positive, she is also very cooperative with her medical history .
She has no history of smoking or had not drank in recent 30 years .

[PAST MEDICAL HISTORY]
As above with hepatitis C .
Notable for a history of cancer of the liver and kidney stones .

[ALLERGIES]
The patient is allergic to peanut and pollen .
Drug allergies include penicillin and erythromycin , no other drug allergies was reported.

[PHYSICAL EXAMINATION]

Index	Attribute	Value
1	age	51
2	gender	lady
3	disease	breast cancer
4	medicine	Metformin;Insu...
5	Patient Case	Positive Positive Negative

Figure 3.1 An example screenshot of IDEAL-X's interface

1) The system learns feedback from users' regular annotation interactions transparently and incrementally.

2) From the knowledge gained by observing the user, the system generates normalized answers to populate forms in real-time

3) No special configuration or training sets are required. Initial training data or configuration may be provided, however, as an option.

3.2.1 Interface Design

IDEAL-X provides a graphical user interface (GUI) for data extraction/classification and human interaction (Figure 3.1). The main window of IDEAL-X is split into two panels that sit side by side: a text input panel and an output panel. The text input panel displays the reports of current subject being processed, where multiple reports of the subject (when available) are arranged through tabs. The output panel displays extracted name-value ("Attribute" and "Value") pairs of data elements of interest, and the configured classifiers will be embodied as additional attributes in output table. When selected, locations of values of the form data elements will be highlighted in the text input panel. The "Previous" and "Next" buttons at the bottom on the right-hand side allow users to navigate through the input document collection. In extraction, the user may review and update any prefilled values by mouse-highlighting the correct value in the text, followed by clicking the data field. In classification, the user may choose available class label by clicking the combo box of classification attribute. The interface is simple and intuitive, and the underlying text processing and learning process is transparent to users.

3.2.2 Basic Operations

I) Select a value

To select a value from the input text, the user highlights the text to be extracted by left-clicking and dragging the mouse over the text region. The highlighted text is then dropped into the value field of the corresponding attribute in the output table with a single left click over the field. Color highlights are used to match each field index with the physical text in the input text. When multiple values are inserted into the field, these values are listed in the output Table in the same order as their appearances in the input text.

II) Unselect a value

To unselect a text, whether user selected or system generated, the user may single left click the highlighted text in the input text. The system will remove the value from "Value" field as well as unhighlight the text in the input text.

III) Clear all values in a field

To clear all the values in a field, the user may left single click all the highlighted texts one by one, as described above, or right single click the row of the appropriate form element in the output table. The latter clears the "Value" field in the output table, as well as unhighlights all corresponding texts in the input text.

IV) Navigation

A single left click of the "Next" button loads the next text document into the "Text input" panel, and populates the "Output Table" with system generated values.

A single left click of the "Previous" button loads the previous (already processed) text document in the collection into the "Text input" panel, and displays all selected values in the "Output Table".

3.2.3 Interactions

The system provides a wizard for constructing the metadata of the output form. The user builds the form by specifying a list of data elements and their constraints. An example is the data element “Heart Rate”, which is constrained to be a numerical value between 0 and 200. Other constraints include sections of the report that may contain the values.

However, except for the names of the data elements, specifying constraints are optional, as these can be learned by the system.

Once data element, which represents value to be extracted, is defined, IDEAL-X interacts with users in two approaches: through interactive annotation process and domain knowledge input process. Based on these two ways, the decision model for IDEAL-X may be established through online learning, the input of domain knowledge, or by a combination of the two.

I) Interactive annotation process

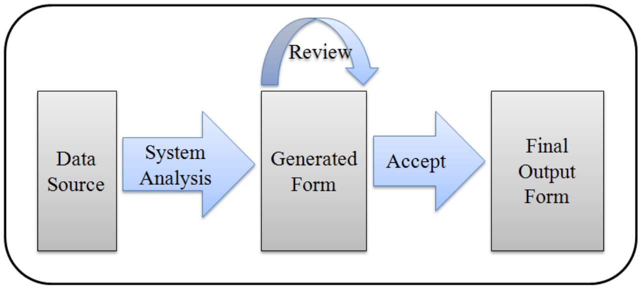


Figure 3.2. The workflow of interactive annotation process

In the online learning mode (See Figure 3.2) of information extraction, the user either manually extracts information from each document, or inspects and correct, if necessary, any prefilled values by the system in the “Output Form”. As the user moves through the document collection, the system learns features of correct and incorrect answers by comparing system extracted values and manually revised ones. Through this process, the decision model improves, and the amount of information that the system is able to correctly prefill grows over time. As one might expect, the "Output Form" for the first few documents may be empty.

When perform classification, user specifies the following information to construct a classification attribute: input attributes, labels of output classes, and machine learning model of classification. While processing each instance, user specifies the class label for each classification attribute. When new instance gets loaded, the system populates labels for classification attributes automatically. User then verify and revise system generated class label to further improve the accuracy of classification.

Once the decision model accrues an acceptable level of accuracy, the user has the option to turn off manual review and to allow the system to complete the extraction/classification for the remaining documents in batch mode. A demo video can be found in YouTube [8].

II) Domain knowledge input process

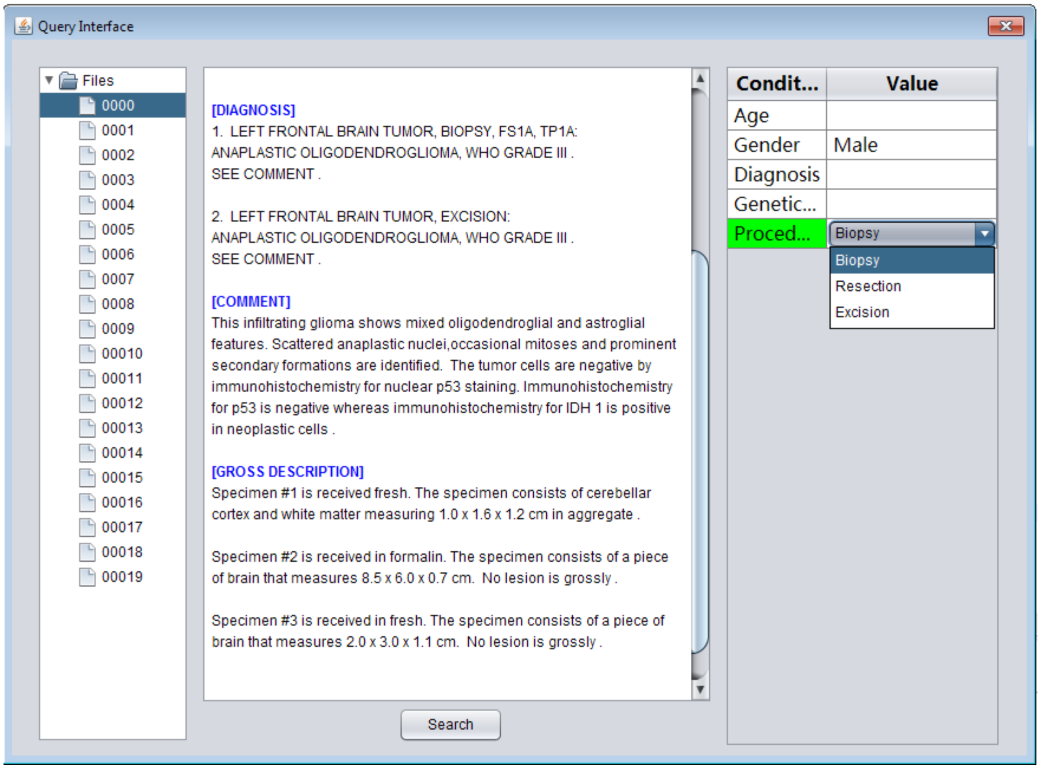
The system also provides a graphical user interface (see section Knowledge Loading Component for detail) for the user to customize domain knowledge, such as an controlled vocabulary, which may contain terminology of attributes and structural properties of documents.

The terminology includes lists of values and their normalization mappings. For example, Disease terminology includes “Diabetes Mellitus" with variations “DM” and “Diabetes”. It also defines inductions. For example, taking “Insulin” or “Metformin” indicates having Diabetes Mellitus. Structural properties provide positive and negative contextual information for giving terms. For example, to extract medications taken by patients, the “Allergies” section is a negative context and medicine names in the section will be skipped. Controlled vocabularies can be a powerful tool to support data extraction: it can be used to locate sentences and chunks of possible values, and to perform normalization for extracted values, discussed in Chapter 5.

The domain knowledge is incorporated directly into the decision model that may be further enhanced by online learning.

3.2.4 Query Interface and Engine

The system provides a built-in query interface (see figure 3.3) that allows the user to search for patients or reports based on user-specified conditions. The extracted data is organized and indexed with reversed index to facilitate querying.



[DIAGNOSIS]
1. LEFT FRONTAL BRAIN TUMOR, BIOPSY, FS1A, TP1A:
ANAPLASTIC OLIGODENDROGLIOMA, WHO GRADE III .
SEE COMMENT .

2. LEFT FRONTAL BRAIN TUMOR, EXCISION:
ANAPLASTIC OLIGODENDROGLIOMA, WHO GRADE III .
SEE COMMENT .

[COMMENT]
This infiltrating glioma shows mixed oligodendroglial and astroglial features. Scattered anaplastic nuclei,occasional mitoses and prominent secondary formations are identified. The tumor cells are negative by immunohistochemistry for nuclear p53 staining. Immunohistochemistry for p53 is negative whereas immunohistochemistry for IDH 1 is positive in neoplastic cells .

[GROSS DESCRIPTION]
Specimen #1 is received fresh. The specimen consists of cerebellar cortex and white matter measuring 1.0 x 1.6 x 1.2 cm in aggregate .

Specimen #2 is received in formalin. The specimen consists of a piece of brain that measures 8.5 x 6.0 x 0.7 cm. No lesion is grossly .

Specimen #3 is received in fresh. The specimen consists of a piece of brain that measures 2.0 x 3.0 x 1.1 cm. No lesion is grossly .

Condit...	Value
Age	
Gender	Male
Diagnosis	
Genetic...	
Proced...	Biopsy

Figure 3.3: Query interface

The interface is split into three main panels. The right panel shows the search condition. For each attribute, the user may specify a value from the list of available values that the system has collected during extraction. The “Search” button finds all reports that match all of the search criteria, and displays the results as a directory tree in the left panel. Selecting a node in this tree loads the content of the corresponding report into the text area of the second panel.

Chapter 4

Online Learning Based Clinical Information Extraction

4.1 Online Machine Learning

4.1.1 Online Learning Overview

Traditional machine learning algorithms take a two-stage approach: batch training based on an annotated training dataset, followed by batch prediction on new datasets based on the model generated from stage one (see Figure 4.1 (a)). In contrast, online machine learning algorithms [10, 11] take an iterative approach (Figure 4.1 (b)). It learns one data points at a time, and based on external feedback, adapts its prediction model for subsequent data points.

Online learning matches the data collection model of heathcare organizations, where new patient data is received on a daily basis. In IDEAL-X, when the predication model achieves a satisfactory accuracy, the system may be switched to run in batch mode.

In both batch and online learning, the objective is to learn some function:

$$F: X \text{->} Y$$

where X is the input dataset and Y is the set of predicted outcomes (also called labels). Given a sequence of input data $(x_1, y_1), (x_2, y_2), \ldots\ldots (x_n, y_n)$, an online learning algorithm generate successive approximations to F that best capture the data points that have already been processed. Each approximation is calculated only on previous approximation and current data point. Previous data points need not be stored, which guarantees constant memory usage.

The goal of online learning is to minimize the cumulative gap between predicted value and the real answers. This could be represented with a loss function:

$$£ = \Sigma E(f(x_{i)} - y_i)$$

where $f(x_i)$ is the predicted value of iteration i, and y_i is the real answer. Function E estimate the different between these two values.

Online machine learning not only significantly reduces human's effort for annotation, since user's role evolves from annotator to reviewer over time, it also provides the mechanism for collecting feedback from human-machine interaction to continuously improved the system's model.

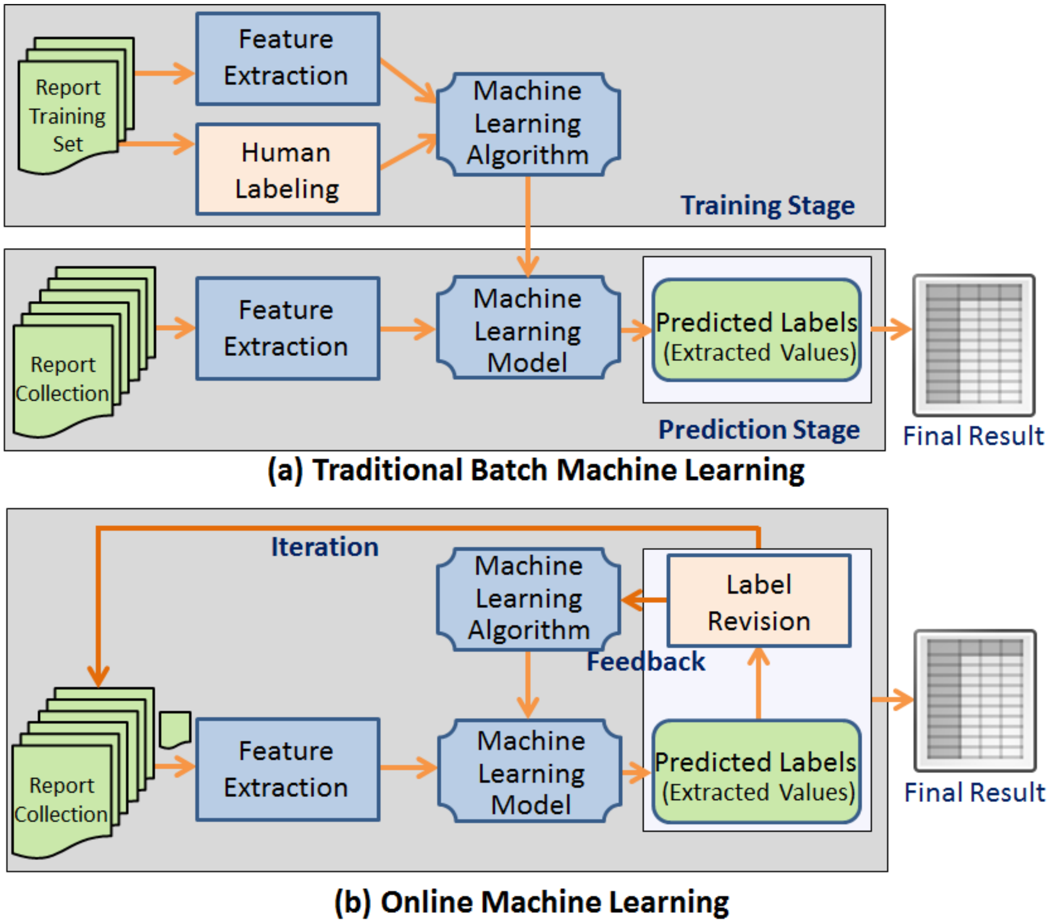


Figure 4.1. Online machine learning versus batch learning. (a) Batch machine learning workflow; (b) Online machine learning workflow

4.1.2 Online Learning Based Information Extraction and Classification

Online learning is integrated into the information extraction workflow of IDEAL-X as follows.

i) Upon loading each document, the system attempts to fill the output form automatically according to its internal extraction/classification model.

ii) The system updates its model automatically based on user feedbacks during the extraction/classification process.

iii) Optionally, the user may provide domain specific knowledge to further support data extraction, standardization and classification. Pre-training with human annotated data is not required for these steps.

Automatic population of the output form is performed with the following steps. To extract information, regions of the input document where target values may appear are detected by a combination of locations in the text and co-occurring words. Then, candidate values are extracted either with machine learning based model or a dictionary. Lastly, constraints such as regular expressions and ranges of numerical values are applied to narrow the candidate set. Candidates that receive confidence scores above the threshold are used to fill the output form, which could be later transformed into a standardized format based on controlled vocabularies and further integrated to render a single structured view.

To classify instances, the information extracted from free text is first integrated with structured data retrieved from a database or data warehouse. The integrated structured data are then analyzed by an online learning algorithm. Finally, the system proposes class labels with associated confidence scores to populate the classification data elements in the output table (See details in Chapter 7).

With online learning, the information extraction and classification models are updated gradually as the collection of processed documents grow. As a user reviews system-generated values, unrevised values and manually updated values are treated as correct answers. Related information are exploited to update the decision model. the system learns the features and contexts of pertinent values and class labels, and is able to propose and generate values for data elements (both information extraction elements and classification elements) automatically after processing a few reports.

4.2 System Design Overview

The design of IDEAL-X consists of two main parts: the frontend graphical user interface (GUI), and the backend data extraction engine. The GUI provides the interface for annotation, feedback, navigation, and customization (Chapter 3). The information extraction system contains of the following major components: preprocessing, domain knowledge loading, data extraction engine, and online learning. Figure 4.2 shows how the data flows among the components (in gray).

The preprocessing component converts input texts and output forms into internal data structures used by the data extraction engine. The domain knowledge component imports domain knowledge, in the form of controlled vocabulary, to support information extraction, standardization and classification. The data extraction engine extracts values from input texts or predicts label for classes to fill the output forms. The online learning component utilizes user inputs, in the form of edits on generated values, to update the decision models of the answer generating component. We describe the modules of each component.

The interface and workflow conform to traditional annotation systems: a user browses an input document from the input document collection and fills out an output form. Upon loading each document, the system attempts to fill the output form automatically with its data extraction engine. The user can review and revise incorrect answers. The system then updates its data extraction model automatically based on user feedbacks. Optionally, the user may provide a customized controlled vocabulary to further support data extraction and answer normalization. Pre-training with manually annotated data is not required, as the prediction model behind the data extraction engine can be established incrementally through online learning, customizing controlled vocabularies, or a combination of the two.

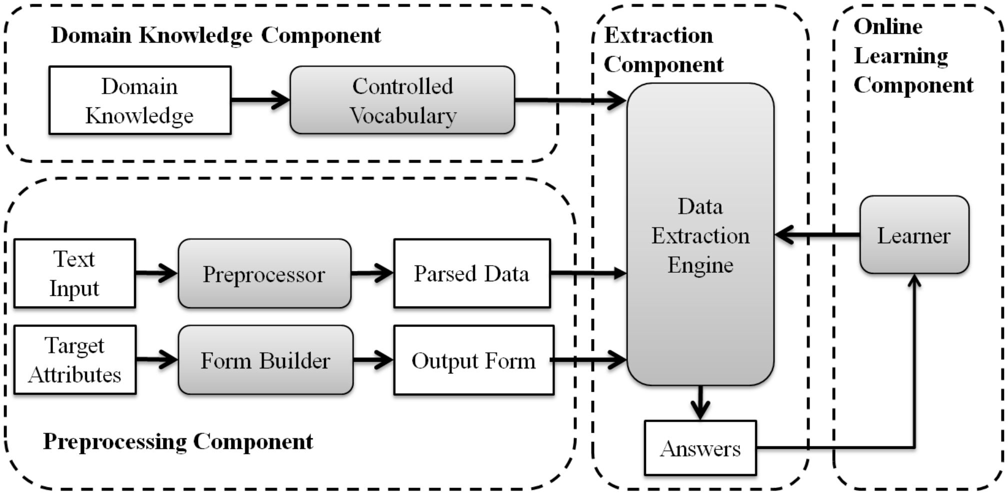


Figure 4.2. System components and dataflow

The system can operate in two modes: 1) interactive: through online learning, the system predicts values to be extracted for each report, and the user verifies or corrects the predicted values; and 2) batch: batch predicting for all unprocessed documents once the accrued accuracy is sufficient for users. While interactive mode uses online machine learning to build the learning model incrementally, batch mode runs the same as the prediction phase of batch machine learning.

4.3 Document Preprocessing

When a report is loaded, the text is first parsed into an in-memory hierarchical tree consisting of four layers: section, paragraph, sentence and token. Apache OpenNLP [73] is used to support parsing. Linguistic features such as part of speech (POS) and data type are also analyzed.

A reverse index of tokens is created to support efficient keywords based search. The index is used to find locations (e.g., sections, paragraphs, sentences and phrases) of a token, as well as its properties such as part of speech and data type. For example, given the token "DM", the system can quickly identify the section (e.g., "History") and the containing sentences. Such token search is frequently performed in answer prediction, and the in-memory index structures enable high efficiency for such operations.

In this step, unique contextual information such as timex, and special data structure such as tabular data, are also identified and analyzed. To timex extraction, the system employs pre-defined regular expressions (See Table 4.1 for examples) to mark temporal data in text:

Timex Example	**Java Regular Expression**
12/03/2012	"\\d{1,2}/\\d{1,2}/\\d{4}"
1-12-98	"\\d{1,2]-\\d{1,2}-\\d{2}"
12012010	"\\d{8}"

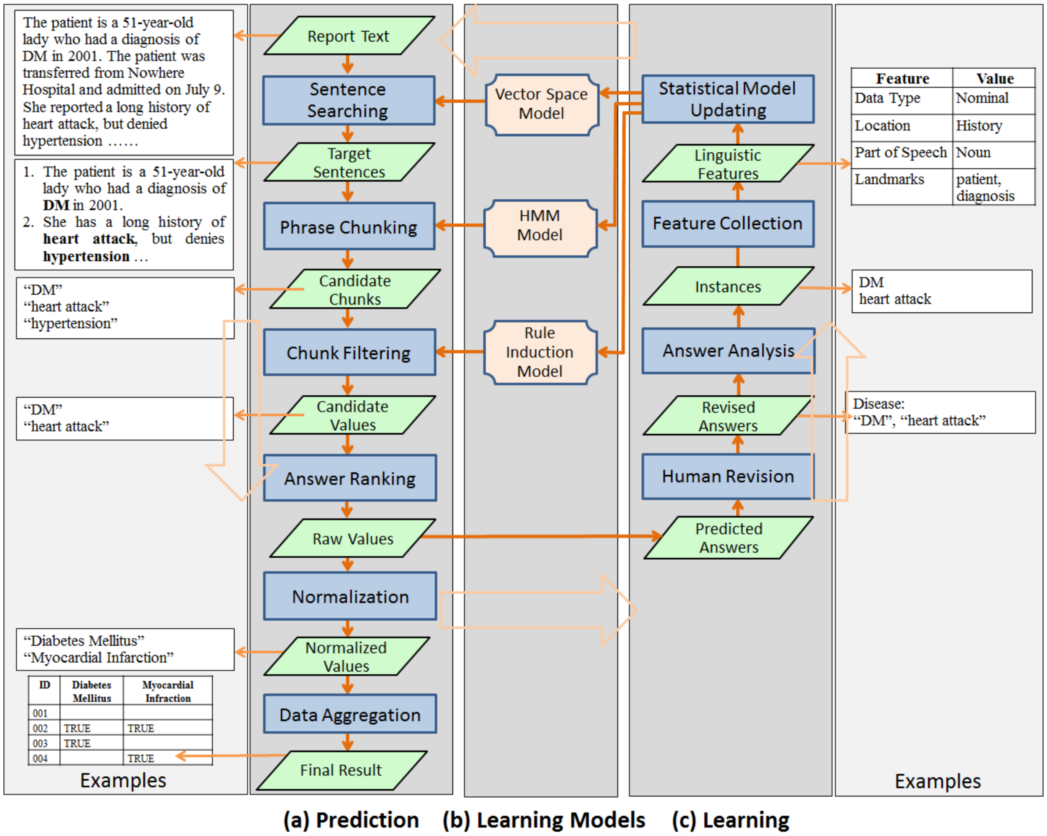
Table 4.1 Timex regular expression examples

To tabular information, the system parses table in input text and comprehends underlying relations between value and metadata.

4.4 The Data Extraction Engine

While the user interacts with IDEAL-X interface, the data extraction engine works transparently in the background. The engine has three major components: answer prediction, learning, and the learning model that the online learning process continuously updates (Figure 4.3).

The system combines statistical and machine learning based approaches with controlled vocabularies for effective data extraction. We present details of the algorithms for prediction and learning below. A key design criteria is the *O(1)* complexity to ensure the overall efficiency of the system.



The patient is a 51-year-old lady who had a diagnosis of DM in 2001. The patient was transferred from Nowhere Hospital and admitted on July 9. She reported a long history of heart attack, but denied hypertension ……

1. The patient is a 51-year-old lady who had a diagnosis of **DM** in 2001.
2. She has a long history of **heart attack**, but denies **hypertension** …

"DM"
"heart attack"
"hypertension"

"DM"
"heart attack"

"Diabetes Mellitus"
"Myocardial Infarction"

ID	Diabetes Mellitus	Myocardial Infraction
001		
002	TRUE	TRUE
003	TRUE	
004		TRUE

Feature	Value
Data Type	Nominal
Location	History
Part of Speech	Noun
Landmarks	patient, diagnosis

DM
heart attack

Disease:
"DM", "heart attack"

Figure 4.3. Overview of system workflow

4.4.1 Answer Prediction

Predicting the value of each data element involves the following steps: 1*) Identifying target sentences* that are likely to contain the answer; 2) *Identifying candidate chunks* in the sentences; 3) *Filtering the chunks* to generate candidate values; 4) *Ranking candidate values* to generate (raw) values; 5) *Normalizing values*; and 6) *Aggregating values* from multiple reports. The workflow is shown in Figure 4.3 (a).

I) Identifying target sentences

Through online learning, the system accrues keywords from past answers (answer keywords) along with co-occurring words in the corresponding sentences (contextual words). For example, given answer keywords “diabetes” and “hypertension” in the sentence “The patient reports history of diabetes and hypertension”, contextual words are ”patient”, “report” and “history”. Such answer keywords and contextual words combined with customized vocabularies can be utilized to identify sentences that are likely to contain answers with the following methods.

a) Similarity based search using the vector space model

Given a collection of contextual words and their frequencies, the system computes the similarity against sentences in the document [74]. Sentences with high similarities are selected. For example, most sentences about “disease” contain "diagnosis" and "history". In particular, the algorithm builds space vectors and ranks candidate sentences with the following five steps:

1) Accumulated contextual keywords are represented by a query q using space vector. The query vector consists of terms whose frequency f are above acceptance threshold α. w_i is the weight of a individual term.:

$$q = (w_1, w_2, w_3, \ldots\ldots, w_n)$$

2) Calculating weight of individual term for a query space vector takes the following steps:

Frequency f is defined as the count of term c_i divided by numbers of accumulated instances n:

$$f_i = c_i / n$$

For terms with frequency f above acceptance threshold α, basic weight b_i is computed with the following function, where *max(c)* is the maximum count among individual terms. *log* curves and smoothes the number of count:

$$b_i = log(c_i) / log\ (max(c))$$

Employing ideas similar to term frequency-inverse document frequency (tf-idf), we promote terms that have high concurrence with the value to be extracted. The high concurrence score o_i is calculated with the following formula:

$$o_i = log\ (\ ci / (g_{i\,/}\,n) * \ \beta)$$

In the expression, g_i is the total count of term i in all processed documents, gi / n represents the average appearance of term i. Since c_i is the number of times term i co-occurs with the answer, the term with the stronger correlation with the answer will receive a higher o_i score, indicating its higher distinguishing power. For example, $ci / (g_{i\,/}$

n) equals 1 means whenever the term *i* appear, the sentence that it belongs to always contains the value to be extracted. In this case, term *i* serves as a most reliable contextual word for identifying the target sentence. β is the parameter used for adjusting the scale of promotion. Finally, the term weight w_i is calculated with $b_i * o_i$.

3) Each sentence *t* in a document is treated as a independent bag of words, represented by the vector:

$$d_t = (w_1, w_2, w_3, \ldots\ldots, w_{j,t})$$

where w_i is the weight of each term, or simply the number of occurrences.

4) The score of a sentence, *St*, is computed as follows.

$$St = (q \bullet d_{t)}\ /log(|dt|)$$

The expression $q \cdot d_t$ represents the dot product ($X \cdot Y = \sum Xi\ Yi$) of query vector *q* and sentence vector d_t. This product is normalized by the length of sentence |*dt*| to dampen the effect of long sentences.

5) Sentences with score *St* above *Max(St)* * *γ* are selected as candidates for further processing. *γ* is a parameter in the range [0, 1] that determines the acceptance threshold based on *Max(St),* the maximum score among all the sentences.

The past contextual keywords and their frequency weights are represented and maintained through a learning model discussed later in “Learning” section.

b) Answer keyword matching search

The answer keywords, combined with relevant user customized vocabularies, are also used to identify target sentences with keyword matching. For example, to extract diseases, if a sentence contains the disease name “myocardial infarction” defined in the vocabulary, the sentence is selected as a target.

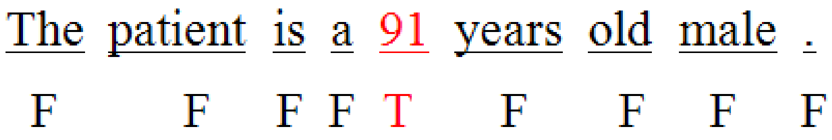
In both approaches, sections to be searched or skipped are also considered in order to narrow the scope of searching.

II) Identifying candidate chunks

After target sentences are selected, the system identifies potential phrases in the sentences using two methods: Hidden Markov model (HMM) [75] and keyword based search.

a) Hidden Markov Model Based Chunking

The HMM is prevalent in pattern recognition and sequence data analysis. It represents target words and contextual words in a sentence with different states, and marks values to be extracted based on probability distributions learned from previously collected values and their sentence. Once an HMM is trained, it can be used for evaluation, which estimates the probability score of a labeled sequence, or for decoding, which aims to optimize labeling for a sequence of input data (observations).



The	patient	is	a	91	years	old	male	.
F	F	F	F	T	F	F	F	F

Figure 4.4 A simple HMM decoding example

For information extraction, usually decoding is applied: tokens are treated as the input sequence of observations, and the algorithm tries to mark this sequence of tokens with a corresponding sequence of states (in other word labels). For example, to extract age information, the following sentence (see Figure 4.4) could be labeled with state T (the state of target value) and state F (the state of irrelevant value).

A Hidden Markov model has three major components:

A set ***S*** of state that represents different labels.

A probabilistic transition function ***a*** that represents, for each pair of states *(i,j)*, the odds of transitions from *i to j*.

Emission probabilities ***b*** which represent the odd that a state *n* output a token *o*

Inspired by a novel adjustable HMM framework [76], we designed a topology for HMM, which emphasizes contextual information of close neighborhood. We present details of this HMM model as follows:

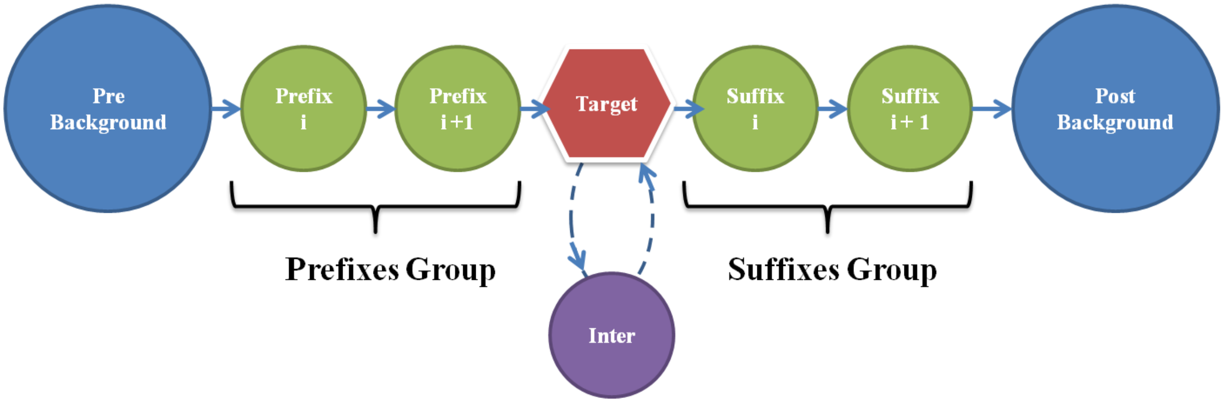


Figure 4.5 Adaptive HMM structure

This HMM graph (Figure 4.5) consists of the follow types of state:

1) Target State: This state represents the token to be extracted.

2) Prefix States: These states represent tokens appear right before target value, window size of which is adjustable.

3) Suffix States: These states represent tokens appear right after target value, window size of which is adjustable.

4) Pre Background and Post Background States: These two states represent irrelevant contextual tokens that are not in the scope of the prefix or suffix window. Pre background state transits to the very first prefix state, and the last suffix state transits to the post background state. Both of these states transit to themselves when background tokens are consecutive.

5) Adaptive Structure (Optional): During the learning phrase, the system learns the feature of values to be extracted and adjusts the structure of HMM correspondingly. When the system learns that the same sentence may have multiple values to be extracted, the inter state (purple state), which represents conjunctions in between values to be extracted, is added to the topology. Its corresponding transaction edges with the target state are also added.

With this adaptable structure, the HMM can operate with the simplest structure. Given a model, and its transition probabilities and emission probabilities, information extraction could be performed with regular HMM decoding algorithms.

In IDEAL-X, we implemented the regular Viterbi algorithm [77]. The time complexity of this algorithm is constraint by the length of input data sequence and the number of states in the HMM. However, regarding processed instances, the decoding time is a constant *O(1)*. The HMM is thus amenable to the efficiency requirements of our system. Each time Viterbi decoding algorithm processes a candidate sentence, which is treated as the input observation sequence, the target chunks that receive high confidence score are selected as candidate chunks for further processing.

b) Keyword based Chunking

The keyword based search finds candidate chunks with the longest match using keywords collected from past answers and the controlled vocabulary.

III) Filtering chunks.

The vector space model narrows the scope of search, and HMM utilizes contextual information to support information extraction. There are additional inherent properties of the tokens that we use for the final extraction, described below. These are implemented in the form of If-Then rules. Whether a rule is applied or not is determined by the learning process based on historical statistics (see next chapter for details).

a) Part of speech (POS)

This filters a phrase by its POS tag in the sentence. Simple example is the system may filter out all the phrases that are not noun when extracting disease names.

b) String pattern

This looks for chunks that match special string patterns, including special characters and capitalizations. For example, criterion may be applied to only approve token that contains character “@” when extract email address.

c) Value domain

This eliminates numerical or enumerated values that fall outside a specified range of values. For example, to hear rate, candidate numbers bigger than 200 could be rejected.

d) Date Type

This eliminates values that are not in the approved categories. For example, the system may only accept integer values for age.

e) Disambiguation Contextual Words

This aims to filter out ambiguous term which is irrelevant to the objective concept. For example: to extract disease information in the sentence “the patient takes cancer screen”, the term “cancer” should not be extracted as disease of the patient since “cancer screen” is a medical procedure, not a disease. In this case, “screen” would be used as landmark to filter out the term “cancer”. The learning process discovers this sort of contextual words, and uses them for disambiguation.

In addition, negation and uncertainty detection are major challenges of clinical information extraction. In IDEAL-X, we use predefined rules to detect negation and uncertain contextual information to rule out negated or uncertain terms.

f) Negation

This removes phrases governed by words that reverses the meaning of the answer. For example, if a candidate chunk “cancer” is extracted from a sentence “the patient has no history of cancer”, “cancer” would not be included.

g) Certainty

Similar to negation filter, this detects and filters uncertain event or situation. For example, a candidate chunk “radiation therapy” for treatment from a sentence “the patient is planned to take radiation therapy” should not be included.

Clearly, when applicable, these rules improve the precision of the extraction, but they do not improve recall. The If-Then rules take time *O(1)* to execute.

IV) Ranking Candidate Values

The combined scores of the selected sentences and chunks are ranked. For a single-valued data element (e.g., heart beat), the candidate value with the highest confidence score is selected. For a multi-valued data element (e.g., medication), values with confidence scores above a threshold are selected. Based on this, each candidate value is either accepted or rejected.

V) Normalizing Values

This step normalizes extracted values through transformation, generalization and induction given by the controlled vocabulary. For example, “DM” is transformed to “Diabetes Mellitus”. “Pindolol” is generalized to its hypernym “beta blocker”. The appearance of medication term “Metformin” (a drug for treating type 2 diabetes) in the text can infer the disease “Diabetes Mellitus”. (See Chapter 5 for detail).

VI) Aggregating Results

Data extracted from multiple reports of a patient will be aggregated into a single table. The aggregation process may normalize values and remove duplicates. For example, “lisinopril” and “captopril” are extracted from discharge summary and inpatient report respectively, and they can be normalized as “ACE inhibitor”. If the same data element is extracted from multiple reports, deduplication is performed. The final output is a table that can be easily exported to other applications such as Excel or a database.

4.4.2 Learning

As described in Chapter 3, system predicted values automatically populate the output table, and the user advances to the next report with or without revision to these values. In both cases, the internal learning and prediction models of IDEAL-X are updated. This is the essence of online learning.

For each instance, IDEAL-X collects and analyzes the following features: 1) Position: location of the answer in the text hierarchy; 2) Landmark: co-occurring contextual keywords in a sentence; 3) POS: parts of speech tag; 4) Value: the tokens of the answer; 5) String patterns: literal features such as capitalization, initial and special punctuation.

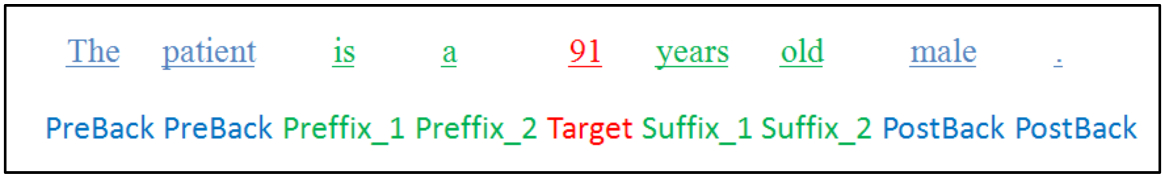
Updating involves updating the three statistical models described above: the space-vector model, HMM, and the rule induction model.

a) Updating Space Vector Model

This model uses “Landmark” features of positive instances. After removing stop words, such as “a”, “and”, “the”, and etc, based on a predefined list, the system updates count of each co-occurring contextual words, which will be used for calculating weights of the query space vector when perform prediction [74].

b) Updating HMM

For positive training instances, HMM lists all words in a sentence as a sequence, in which an extracted value is marked as target value state and other words are recognized as irrelevant contextual states. For example, given neighborhood window 2, the follow sentence is first marked with different states labels.



The	patient	is	a	91	years	old	male	.
PreBack	PreBack	Preffix_1	Preffix_2	Target	Suffix_1	Suffix_2	PostBack	PostBack

Figure 4.6 HMM training example

Based on this labeled sequence, the count of transition between states, and observation (in this case, token) from a state are updated, with which the state transition probabilities and emission probabilities can be recalculated [75]:

c) Updating rule induction model

Most rules are applied to data element, which affects all values to be extracted. These filtering rules are induced based on the coverage percentage [78]. Features such as POS, data types, string patterns of positive instances are analyzed and their respective coverage percentages are modified. Once the support level of a rule reaches a predefined threshold, the rule is triggered for filtering. For example: if the system detects that more than 95% of extracted values are noun, the rule “only accept noun” could be triggered.

Rules for disambiguation are applied to individual values of a data element. For example, to data element “disease”, the disambiguation term “scan” is only applied to the value “cancer”. To learn these disambiguation terms, the system analyze both positive instances and negative instances: the word has only appeared in rejected instances are captured as disambiguation term, and related rejection rule is created for the corresponding term of given data element.

V) Summary

In interactive mode, the above four steps repeat for each report, where the learning models are continuously updated and improved.

For example, to extract age information from a sentence “The patient is a 25 years old gentleman”, the system learns features from a positive instance value “25”. Contextual words “patient”, “years”, “old” and “gentleman” are captured as landmark and their frequencies are updated. The sequence of words in the sentence is also ordered to update the HMM model. As the value “25” is recognized as an integer, the coverage of rule for filtering non-integer candidates will increase correspondingly. These updated models will be applied to prediction for the remaining unprocessed reports.

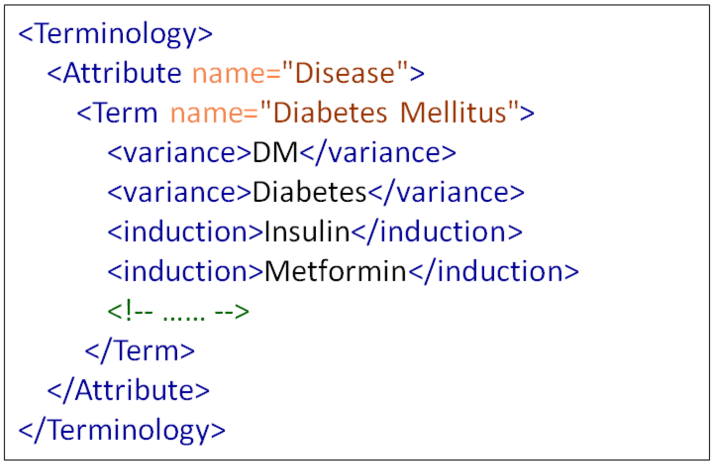
Chapter 5

Controlled Vocabulary Supported Clinical Information Extraction

As a complement to interactive learning, the system allows direct inputs of knowledge – the so called controlled vocabulary, to help with information extraction and standardization. In this chapter, we describe details of the controlled vocabulary in IDEAL-X, its implementation, as well as two approaches for customizing a controlled vocabulary.

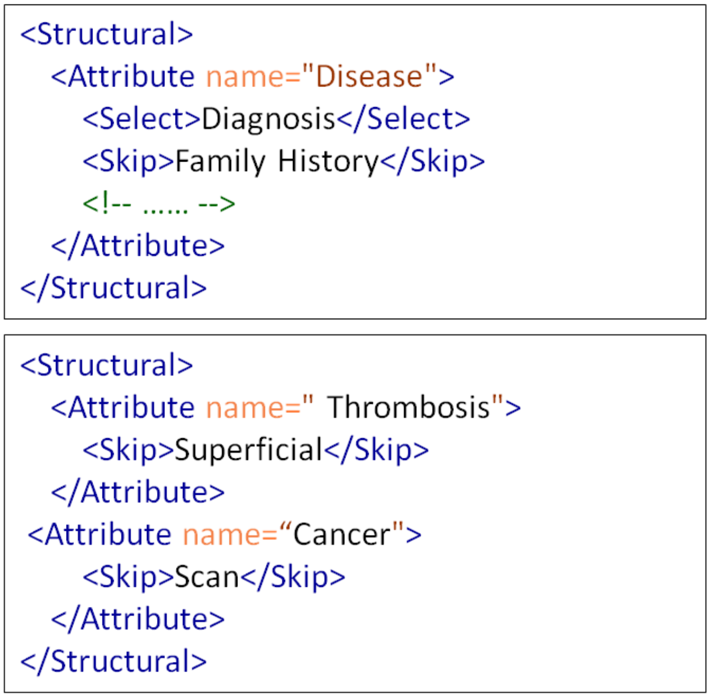
5.1 Construction of Controlled Vocabularies

The current implementation is limited to controlled vocabularies in clinical domains, and can be constructed by physicians and clinical researchers without expertise in natural language processing or programming. The customized knowledge may be stored in XML files or databases.



```
<Terminology>
  <Attribute name="Disease">
    <Term name="Diabetes Mellitus">
      <variance>DM</variance>
      <variance>Diabetes</variance>
      <induction>Insulin</induction>
      <induction>Metformin</induction>
      <!-- …… -->
     </Term>
  </Attribute>
</Terminology>
```

Figure 5.1: Examples of terminology



```
<Structural>
  <Attribute name="Disease">
    <Select>Diagnosis</Select>
    <Skip>Family History</Skip>
    <!-- …… -->
  </Attribute>
</Structural>
```

```
<Structural>
  <Attribute name=" Thrombosis">
    <Skip>Superficial</Skip>
  </Attribute>
 <Attribute name="Cancer">
    <Skip>Scan</Skip>
  </Attribute>
</Structural>
```

Figure 5.2 Examples of structural constraints

The knowledge in the controlled vocabulary can be categorized into two types: 1) Terminology: This includes terms and corresponding standardization mapping. Figure 5.1 shows an example. Variances such as “DM” and “Insulin” indicate ways in which diabetes may appear in reports. The name “Diabetes Mellitus” provides the standard terminology for the variations. 2) Structural Constraints: This helps to indicate sections to include or ignore in the extraction process. Constraints may also contain disambiguation terms, which could further improve the precision of extraction. Figure 5.2 is a simple example that “superficial” is a negative indicator for extracting deep vein “Thrombosis”.

Note that both forms of knowledge in the controlled vocabulary may be learned through online learning. But if such knowledge can be obtained, making it directly available to IDEAL-X can certainly help to facilitate the learning process.

5.2 Supporting Information Extraction

Controlled vocabularies help to improve the answer prediction process in the following ways.

In sentence searching, terminology can be used for identifying target sentences through keyword matching, and section information of structural constraints can be used to narrow the scope of searching.

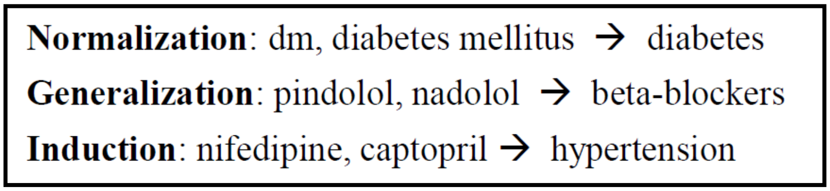
In phrase chunking, terminology can be applied to identify candidate chunks based on the longest match. This complements machine learning based phrase chunking.

In chunk filtering, disambiguation terms provides contextual information to rule out unqualified candidate chunks.

Terminology, when explicitly specified by domain experts, is reliable and hence provides important assistance in information extraction. Another pivotal function of controlled vocabulary is standardization, discussed next.

5.3 Supporting Standardization

After values are extracted, the system consults the mapping table to standardize the output in three ways: normalization, generalization or induction. Figure 5.3 shows examples for each standardization scenario. For example, dm, which abbreviates diabetes, can be normalized to diabetes; Pindolol, a beta-blocker, generalizes to its hypernym; Nifedipine is a medication taken by hypertension patient, therefore, the system infers hypertension even if "hypertension" isn't explicitly mentioned in the text.



Normalization: dm, diabetes mellitus → diabetes
Generalization: pindolol, nadolol → beta-blockers
Induction: nifedipine, captopril → hypertension

Figure 5.3 Examples of standardization

In general, rule-based standardization provides a generic and convenient solution for importing domain knowledge to normalize extracted results.

5.4 Generating Vocabulary

Controlled vocabulary is important to both information extraction and standardization. However, building a vocabulary manually could be cumbersome, especially when the size of terminology is large or the structure of reports is not available. To ease the process of vocabulary building, we implemented the following two convenient approaches:

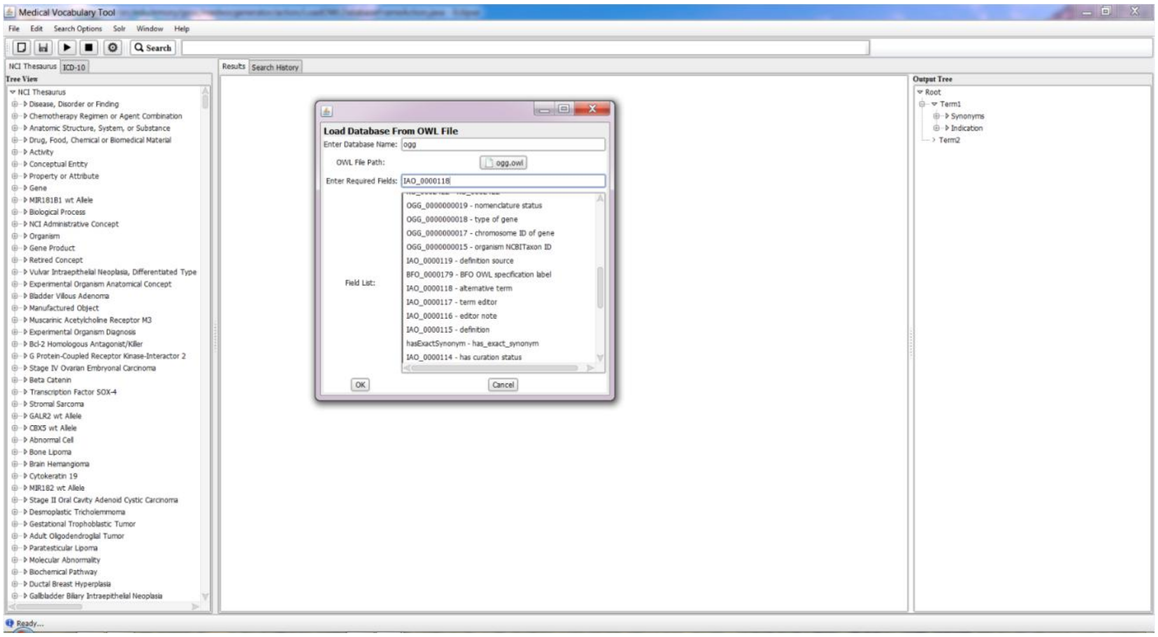
5.4.1 Adaptive Vocabulary

The system incrementally refines the controlled vocabulary during the interactive extraction process. The user can start with a seed vocabulary, that is, terminology obtained from existing domain knowledge resources. Typically, a standard ontology such as the SNOMED Clinical Terms [79], or the NCI Thesaurus [80] is a useful starting point for creating the seed vocabulary.

However, standard vocabulary may be incomplete or miss certain terminologies specific to the local reporting environments. When a mismatch occurs between the vocabulary and an extracted value, IDEAL-X refines the vocabulary adaptively by adding the extracted terms and removing unneeded terms. In this way, the vocabulary converges to a lexicon that is consistent with the extraction task. In addition, the vocabulary may also be exported for reproducibility, other extraction projects, and ontology construction.

5.4.2 Vocabulary Generating Tool

In a related project (Google Summer of Code 2014) [81], we implemented a tool to support vocabulary building based on clinical terminology resources. Clinical ontologies and standards such as ICD-10 code and NCI Thesaurus are reliable resources of clinical lexicon. However, there is a lack of tools to explore and integrate resources for various data formats. We developed a tool, MedVocGenerator, to ease browsing and searching for medical standards and vocabularies. The tool also enables reuse of existing resources to customize a domain specific terminology.



Load Database From OWL File

Enter Database Name: ogg

OWL File Path: ogg.owl

Enter Required Fields: IAO_0000118

Field List:

OGG_0000000019 - nomenclature status
OGG_0000000018 - type of gene
OGG_0000000017 - chromosome ID of gene
OGG_0000000015 - organism NCBITaxon ID
IAO_0000119 - definition source
BFO_0000179 - BFO OWL specification label
IAO_0000118 - alternative term
IAO_0000117 - term editor
IAO_0000116 - editor note
IAO_0000115 - definition
hasExactSynonym - has_exact_synonym
IAO_0000114 - has curation status

Medical Vocabulary Tool
File
Edit
Search Options
Solr
Window
Help
Search
NCI Thesaurus
ICD-10
Tree View
NCI Thesaurus
Disease, Disorder or Finding
Chemotherapy Regimen or Agent Combination
Anatomic Structure, System, or Substance
Drug, Food, Chemical or Biomedical Material
Activity
Conceptual Entity
Property or Attribute
Gene
MIR181B1 wt Allele
Biological Process
NCI Administrative Concept
Organism
Gene Product
Retired Concept
Vulvar Intraepithelial Neoplasia, Differentiated Type
Experimental Organism Anatomical Concept
Bladder Villous Adenoma
Manufactured Object
Muscarinic Acetylcholine Receptor M3
Experimental Organism Diagnosis
Bcl-2 Homologous Antagonist/Killer
G Protein-Coupled Receptor Kinase-Interactor 2
Stage IV Ovarian Embryonal Carcinoma
Beta Catenin
Transcription Factor SOX-4
Stromal Sarcoma
GALR2 wt Allele
CBX5 wt Allele
Abnormal Cell
Bone Lipoma
Brain Hemangioma
Cytokeratin 19
MIR182 wt Allele
Stage II Oral Cavity Adenoid Cystic Carcinoma
Desmoplastic Tricholemmoma
Gestational Trophoblastic Tumor
Adult Oligodendroglial Tumor
Paratesticular Lipoma
Molecular Abnormality
Biochemical Pathway
Ductal Breast Hyperplasia
Gallbladder Biliary Intraepithelial Neoplasia
Results
Search History
OK
Cancel
Output Tree
Root
Term1
Synonyms
Indication
Term2
Ready...

Figure 5.4 Screenshot of MedVocGenerator

The GUI of MedVocGenerator is intuitive and user-friendly. A corpus may be loaded from the menu-bar. The content of the corpus is display as a tree on the left panel. The user may search for terms with the support of auto-complete and spell-check function. Search results are displayed on the center panel, and the right panel shows the vocabulary to be outputted in a tree view.

This interface also supports convenient drag-and-drop operation. The user can drag terms from loaded corpus or text in search results, and drop them into the output vocabulary tree. New nodes may be defined and add new nodes to the output tree. The resulting vocabulary can be outputted as XML/JSON or other formats to support other applications.

5.4.3 Discussion

Both adaptive vocabulary feature of IDEAL-X and vocabulary generating tool MedVocGenerator aim to ease the process of vocabulary construction, and are complementary. Adaptive vocabulary polishes an existing vocabulary automatically and transparently within the normal information extraction workflow. In contrast, the MedVocGenerator tool assists the user in the vocabulary construction as a precursor to the information extraction process.

Chapter 6

Clinical Information Extraction Use Cases

We demonstrate the clinical effectiveness of the system with three use cases. Each case focuses on one or more features and usages of the system. In the first case, the system processes and integrates heterogeneous clinical reports with the support of controlled vocabulary. The second case illustrates the system importing and adapting existing terminology such as ontology. In the third case, we study the applicability and challenge of using IDEAL-X to identify low prevalence diseases. The following evaluation metrics are used.

- Precision: This estimates the correctness of extraction.

$$Precision = TP / (TP + FP)$$

- Recall: This estimates the completeness.

$$Recall = TP / (TP + FN)$$

- F-1: This is the weighted average of precision and recall.

$$F1 = 2 * (Precision * Recall) / (Precision + Recall)$$

For clinical effectiveness focused study, we use sensitivity and specificity, and their confidence interval for comparison.

$$Sensitivity = TP / (TP + FN)$$

$$Specificity = TN / (FP + TN)$$

The use cases provided motivation problems and helped to evolve and improve the design of IDEAL-X over the course of our research. The experimental results not only demonstrate the clinical effectiveness of our approach, they also provide firsthand experience with real world clinical data and in-depth understandings of practical challenges.

6.1 Use Case 1: Large Scale Cohort Identification for Cardiology Research

6.1.1 Background and Motivation

Emory University Cardiovascular Biobank aims to address a variety of research questions in cardiovascular diseases. It is a registry of patients with suspected or confirmed coronary artery disease undergoing cardiac catheterization. The final database will store approximately 12,000 patients' records, and will contain information from eight sources including major Emory Healthcare units. Apart from the data collected with standardized questionnaire, clinical data is collected from up to eight types of reports: Cardiac Catheterization Procedure Report, Echocardiogram Report, History and Physical Report, Discharge Summary, Outpatient Clinic Note, Outpatient Clinic Letter, Coronary Angiogram Report, and Inpatient Report as well as Discharge Medication lists. Data elements extracted from reports and structured records are integrated to provide comprehensive information for patient identification. Manual extraction of the data is infeasible due to the large number of reports.

6.1.2 Experiment Setup

I) Datasets

We use three datasets from 100 patients that are randomly sampled from a collection of about 5,000 patients in the Emory Biobank database. Dataset 1 is a set of semi-structured reports and contains 100 cardiac catheterization procedure reports. Dataset 2 is a set of template-based narration and contains 100 coronary angiographic reports. Dataset 3 is a set of complex narration and contains 315 reports, including history and physical report, discharge summary, outpatient clinic notes, outpatient clinic letter, and inpatient discharge medication report.

II) Ground truth

The test data sets are independently hand-annotated by domain expert annotators from Emory Clinical Cardiovascular Research Institute. An arbitrator – an independent cardiovascular disease researcher reconciles incompatible outputs of the system and the manual annotations to produce the final ground truth.

III) Evaluation Metrics

For validation, precision, recall, and F1 scores are used to estimate the effectiveness of extraction by comparing the system predicted results (before human revision) and the ground truth.

IV) Experiment Settings

We aim to evaluate the effectiveness of the system with respect to using online learning and controlled vocabularies, and to understand their applicability to different report forms. By analyzing the report styles and vocabularies, we discover that online learning is more suited for semi-structured or template based narration reports, and controlled vocabulary guided data extraction would be more effective on complex narration with a finite vocabulary. Thus, we design three experiments:

1) Online learning based data extraction, where controlled vocabularies are not provided, based on Dataset 1 (semi-structured) and Dataset 2 (template based narration);

2) Controlled vocabularies based data extraction, where online learning is not used, based on Dataset 3 (complex narration);

3) Controlled vocabularies guided data extraction combined with online learning, based on Dataset 3.

6.1.3 Performance Evaluation

Experiment 1: Online Machine Learning Based Data Extraction

This experiment is based on Dataset 1 and 2. The system starts in interactive mode with an empty decision model. The defined data elements are summarized in Table 1 and

Table 2 in Appendix. The user processes one report at a time, and each system predicted value (including empty values for the first few reports) before user revision is recorded for calculating precision and recall.

Results are summarized in Table 6.1 and Table 6.2 for the two datasets, respectively. The results are divided into two stages to demonstrate how quickly the system learns: reports 1 to 20, and reports 21 to 100. Within the first 20 reports, IDEAL-X achieves precisions higher than 96% for both datasets. Over the 80 reports in the second stage, we observe notable improvements on recall (from 90% and 74% for the first 20 reports to over 97% for the last 80 reports).

Dataset	**Numbers of Data Elements**	**Number of Values**	**Precision**	**Recall**	**F1**
Report 1-20	19	247	99.5%	90.2%	94.6%
Report 21-100	19	1025	99.9%	98.0%	98.9%
Overall	**19**	**1272**	**99.8%**	**96.5%**	**98.1%**

Table 6.1. Results of data extraction from semi-structured reports (Dataset 1)

Dataset	**Number of Data Elements**	**Number of Values**	**Precision**	**Recall**	**F1**
Report 1-20	16	138	96.2%	74.6%	84.0%
Report 21-100	16	590	97.4%	97.6%	97.5%
Overall	**16**	**728**	**97.2%**	**93.2%**	**95.2%**

Table 6.2 Results of data extraction from template based narration reports (Dataset 2)

Experiment 2: Controlled Vocabularies Guided Data Extraction

In this experiment, online learning is disabled and data extraction is performed in batch using controlled vocabulary. Diseases and medications are extracted from Dataset 3 (values to be extracted are shown in Table 3 in Appendix). Customized controlled vocabularies, including terminology and structural properties, have been created by physicians through analyzing another development report dataset of 100 patients, disjoint from Dataset 3.

The results in Table 6.3 show that controlled vocabularies are highly effective for data extraction over complex narratives. Domain-specific data, for example cardiology related diseases and medications, have limited numbers of possible values (or domain values), and a carefully customized controlled vocabulary can achieve high extraction accuracy.

Type of Data Elements	**Number of Data Elements**	**Number of Ground Truth Values**	**Precision**	**Recall**	**F1**
Diseases	15	418	94.5%	99.0%	96.7%
Medications	10	437	98.6%	99.7%	99.2%
All	**25**	**855**	**96.5%**	**99.4%**	**97.9%**

Table 6.3 Results of controlled vocabularies guided data extraction from complex narration (Dataset 3)

Experiment 3: Controlled Vocabularies Guided Data Extraction Combined with Online Machine Learning

In this experiment, we perform two tests. Test 1 generates the baseline for comparison, and Test 2 demonstrates the effectiveness of combining online machine learning and controlled vocabularies. Dataset 3 is used to extract all diseases and medications.

For Test 1, terminology is used and online machine learning is disabled, so the test is guided by controlled vocabulary without any structural properties. We notice that comprehensive terminology contributes directly to high recall rate, which means that the system seldom misses values to be extracted. However, if structural properties are not included, compared to the result in Experiment 2, the precision is much lower. This highlights the value of positive and negative contexts in an extraction task.

For Test 2, both terminology and online machine learning are used. Online machine learning will support learning structural properties. To show how quickly the system learns, only reports of the first 10 patients are processed with interactive online learning, and all following reports are processed in batch. Results in Table 6.4 show an overall precision of 94.97%, which demonstrates that online learning could quickly learn structural properties.

Test	**Controlled Vocabulary**	**Online learning**	**Precision**	**Recall**	**F1**
1	Terminology Only	N/A	80.9%	99.4%	89.2%
2	**Terminology Only**	**Applied to first 10 patients**	**94.9%**	**99.4%**	**97.1%**

Table 6.4 Results of controlled vocabularies guided data extraction combined with online learning

6.1.4 Discussion

Online learning is highly effective for reports with relatively clear structural patterns, such as semi-structured or template based narration. For complex narration constrained by finite data domain, controlled vocabularies are highly effective for supporting extraction. In addition, structural properties such as section constraints can greatly assist in improving the accuracy of extraction.

In most clinical information such as procedure, diagnosis and medicine, the relevant terminology is a finite set. For example, the possible values for specimen receiving status consists of only "Fresh", "In formalin", etc. For these cases, preparing a comprehensive terminology is feasible. A comprehensive enumeration of structural properties is more challenging, as it requires an understanding of the report structure and contextual pattern that may be specific to the local reporting environment. However, online learning is capable of learning these structural properties. The combination is thus complementary and effective.

6.2 Use Case 2: Support Patient Search on Pathology Reports

6.2.1 Background and Motivation

Synoptic reporting [82-84] has become a powerful tool for providing summarized findings through predefined data element templates such as CAP Cancer Protocols [4]. Meanwhile, standard groups such as IHE are proposing structured reporting standards such as Anatomic Pathology Structured Reports [3] in HL7. While the community is tending towards structured reporting, a vast amount of pathology reports exists in legacy systems in unstructured format, and the standardization effort only captures major data elements, leaving useful research information in free text that is difficult to process and search.

We explore the adaptive vocabulary feature of IDEAL-X, which employs an initial controlled vocabulary that is continuously refined through online learning during the extraction process. We also provide a built-in query interface to support searching patients based on extracted data elements.

6.2.1 Experiment Setup

To test the performance of information extraction, we conduct two experiments. Experiment 1 examines the effectiveness of online learning, and experiment 2 studies the importance of adaptive vocabulary. The driving medical research is brain tumor study, in which pathology reports need to be queried based on demographic data, disease, procedure, etc, in order to locate patients with certain traits. The Human Disease Ontology, Cell Cycle Ontology and NCI Treasure were used as seed vocabulary.

I) Dataset

We randomly selected and annotated 50 anatomic pathology reports manually as the test dataset for this study. These pathology reports were from patients that had been diagnosed with a grade II or grade III infiltrating glioma and had their tumors resected at Emory University Hospitals. Another 50 reports, disjoint from the test set, were used for development.

II) Experiment settings

All experiments began with an empty model, without prior training or predefined constraints. We performed tests on extracting personal information such as age and gender, and most frequently queried medical information such as diagnosis, genetic marker and therapy/procedure (See table 6.5 for details). To support extraction, we employ a seed vocabulary consisting of diagnosis, genetic marker (both gene and protein) and procedure lexicon, which are loaded from the Human Disease Ontology [85], the Cell Cycle Ontology [86] and the NCI Thesaurus [80] Ontology respectively.

Attributes	Seed Vocabulary Sources	Value Amount
Age & Gender	None	100
Diagnosis	Human Disease Ontology	147
Gene & Protein	Cell Cycle Ontology	146
Therapy & Procedure	NCI Treasure	324

Table 6.5. Test cases of data extraction

III) Evaluation Metrics

We compared the system’s output with the manually annotated ground truth with respect to precision, recall and F-1 measure.

6.2.3 Performance Evaluation

Results of experiment 1 are shown in Table 6.6. Age and gender typically appear in report headers with limited contextual variation. For these, the system achieved very high precision and recall. Values related to diagnosis, genetic marker and therapy appear in text with larger structural and narrative variation. With the support of the seed vocabulary, the system achieved F1 scores of 88%, 93% and 97% respectively. To study the effectiveness of learning, for each test case, we divided the 50 reports into two groups: the first 20 reports (as they appear in the directory), and the last 30 reports. The improvements between these two groups were significant, reflecting a high-rate of learning. For the four classes of attributes, F1 scores between the first and second groups increased from 94.7%, 82.1%, 90.0% and 95.3% to 100%, 91.2%, 95.3% and 99.5%, respectively.

Attributes	Subsets	Precision	Recall	F-1
Age & Gender	First 20	100%	90.0%	94.7%
	Last 30	100%	100%	100%
	Overall 50	100%	96.0%	97.9%
Diagnosis	First 20	90.6%	75.0%	82.1%
	Last 30	94.3%	88.2%	91.2%
	Overall 50	93.1%	83.5%	88.0%
Genetic Marker	First 20	90.0%	90.0%	90.0%
	Last 30	94.8%	95.8%	95.3%
	Overall 50	93.1%	93.8%	93.5%
Therapy and Procedure	First 20	97.4%	93.3%	95.3%
	Last 30	100.0%	99.0%	99.5%
	Overall 50	99.0%	96.9%	97.9%

Table 6.6. Test result of experiment 1: study of online learning

Attributes	**Adaptive Vocabulary**	**Precision**	**Recall**	**F-1**
Diagnosis	Off	90.0%	36.9%	52.4%
	On	93.1%	83.5%	88.0%
Genetic Marker	Off	84.1%	86.9%	85.5%
	On	93.1%	93.8%	93.5%
Therapy and Procedure	Off	89.3%	66.9%	76.5%
	On	99.0%	96.9%	97.9%

Table 6.7. Test result of experiment 2: study of adaptive vocabulary

Results of experiment 2 show that the ability to refine the vocabulary significantly improves the extraction accuracy. For diagnosis, genetic marker, therapy and procedure, Table 6.7 shows the difference between using the seed vocabulary with and without refinement. When the system used the seed vocabulary directly without updating, performance of the extraction relies on how closely the vocabulary content aligns with the extraction task. For genetic marker, a comparatively small difference in the F1 score was observed. For diagnosis and procedure, on the other hand, the downloaded ontology subsets contain considerable irrelevant information for pathology. This impacted the precision by 3.1% for diagnosis, and 10.3% for procedure. Moreover, many terms were missing, and it negatively affected the recall by 46% for diagnosis and 30% for procedure. These results show the sometimes large discrepancy between standard ontology and the needs of extraction projects, as well as the benefits of updating vocabulary during extraction.

6.2.4 Discussion

We chose representative attributes for testing the effectiveness of online machine learning and the utility of refining the seed vocabulary. The system showed high accuracy and high learning efficiency. In the most ideal scenario, the user is able to enumerate comprehensive terminology to provide a well-defined dictionary. However, when the terminology set is large, for example genetic marker, building a complete controlled vocabulary a priori may be unattainable. The adaptive vocabulary feature allows the user to take advantage of existing dictionary resources, and to use feedback during the extraction process to closely align the vocabulary with the needs of the extraction task.

As a next step, we will consider a broader set of attributes and enrich the data types the system can support. Values that the system can manage currently are limited to numerical value and nominal value. Extracting temporal information, for example, will improve the utility of the system. In pathology research, medical events such as procedure are time sensitive. Augmenting the output with timelines would contextualize and help to connect the extracted values in important ways.

We will also study the possibility of use of IDEAL-X to structure anatomic pathology reports and synoptic pathology reports. Most existing medical report systems still allow for free-format text and uncontrolled vocabulary. During extraction with IDEAL-X, it may be possible to simultaneously translate the input text according to structured pathology report standards. The resulting text is still easy to read, but will additionally facilitate subsequent analysis and algorithmic processing.

6.3 Use Case 3: Information Extraction Supported Disease Surveillance

6.3.1 Background and Motivation

Venous thromboembolism (VTE), including deep vein thrombosis (DVT) and pulmonary embolism (PE), is associated with significant morbidity and mortality [87]. VTE can be diagnosed by several radiolographic studies, including lower or upper extremity ultrasonography and computerized tomography (CT) of the chest. Federally mandated reporting of VTE defined by the Agency for Healthcare Research and Quality Patient Safety Indicator 12 (AHRQ PSI-12) [88] is based on administrative and billing data, whose accuracy for detecting VTE has yet to be demonstrated. We use IDEAL-X to evaluate its accuracy for identifying VTE diagnosis directly from radiology reports in electronic medical records.

6.3.2 Experiment Setup

Full text of radiology reports, which are complex narration style, and clinical data were extracted from the electronic medical records (Cerner Corp, Kansas City, MO) of 13,248 patients admitted to Emory University Orthopedic and Spine Hospital from 2009-2014. Patient encounters were defined as a hospital admission where both surgery (of the spine, hip, or knee) and a radiology diagnostic study for VTE were performed. A physician manually reviewed each radiology report for diagnosis of a DVT or PE. We use IDEAL-X to analyze the same radiology report under two separate modes: i) controlled vocabulary mode, where the user specifies upfront terminology and contextual information (such as relevant and irrelevant report sections) to be extracted, and ii) online machine learning mode, where all terminology and contextual information is learned incrementally. Performance was analyzed for total radiology reports, and patient encounters (multiple reports per encounter possible).

6.3.3 Performance Evaluation

Among 2083 radiology reports in the testing dataset, IDEAL-X in controlled vocabulary mode correctly identified 176/181 VTE events, achieving a sensitivity of 97.2% (95% Confidence Interval [CI] 93.7-99.1%) and specificity of 99.3% (95% CI 98.9-99.7%) when compared to manual review (Table 6.8). This performance was superior to online machine learning mode, which achieved an overall sensitivity of 92% (95% CI 88.3%-96.1) and 99% specificity (95% CI 98.5%-99.4%), and required approximately 50% of reports to be processed before achieving >95% sensitivity and specificity (Figure 6.1).

Event	**Radiology Report Types**	**Total Reports**	**Positive Reports By Manual Review**	**Positive Reports By IDEAL-X**	**Measure**	**IDEAL-X Performance (95% CI)**
DVT	Ultrasonography of Upper or Lower Extremity	1153	112	109	Sensitivity	97.3% (92.4-99.4%)
					Specificity	99.4% (98.7-99.8%)
PE	CT and MRI of Chest	930	69	67	Sensitivity	97.1% (89.9-99.6%)
					Specificity	99.3% (98.5-99.7%)
Either DVT or PE	All four types above	2083	181	176	Sensitivity	97.2% (93.7-99.1%)
					Specificity	99.3% (98.9-99.7%)

Table 6.8. Performance of IDEAL-X system in controlled vocabulary mode, analyzing total radiology reports

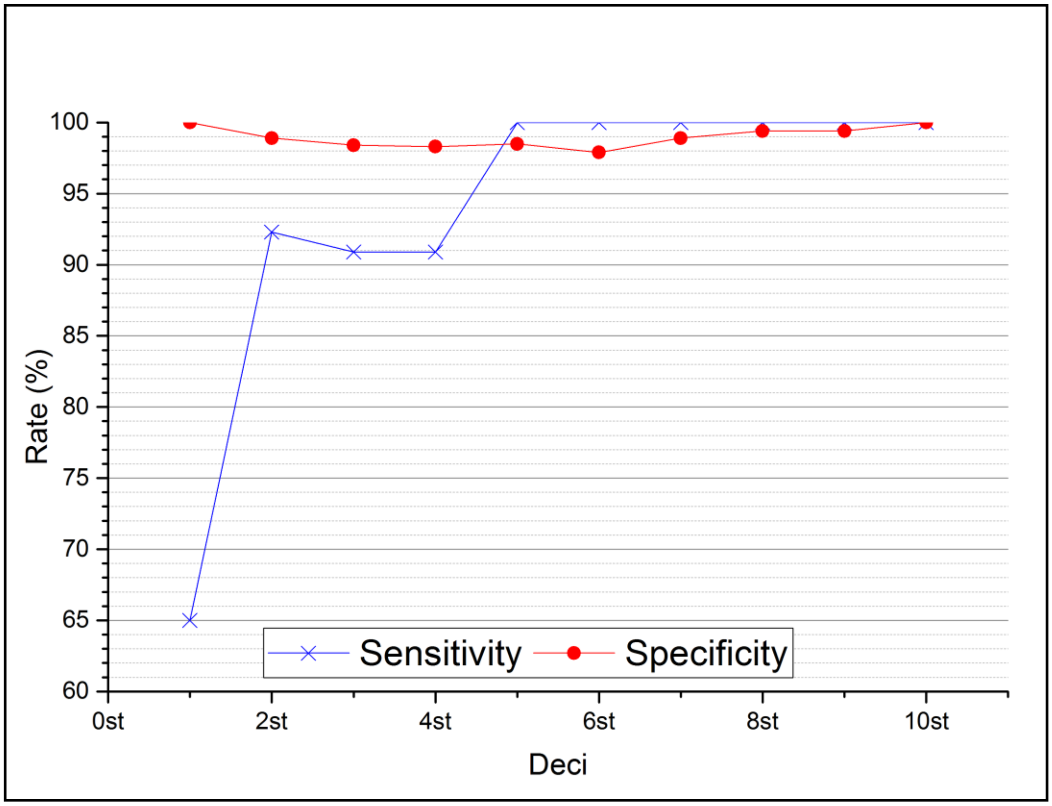


Figure 6.1 Sensitivity and specificity changes over processed records

Among 422 surgical encounters with diagnostic radiographic studies for VTE, IDEAL-X in controlled vocabulary mode correctly identified 41/42 VTE events, achieving a sensitivity of 97.6% (95% CI 87.4-99.6%) and specificity of 99.8% (95% CI 98.7-100.0%) (Table 6.9). The performance surpasses that of AHRQ-PSI 12[88], which has sensitivity of 92.9% (95% CI 80.5-98.4%) and specificity of 92.9% (95% CI 89.8-95.3%), though only the difference in specificity was statistically significant (*p*<0.01).

Event	**Total Patients**	**Events by Manual Review**	**Events by IDEAL-X**	**Events by AHRQ-PSI 12**	**Measure**	**IDEAL-X (95% CI)**	**AHRQ-PSI 12 (95% CI)**	***P* Value**
DVT	422	17	16	13	Sensitivity	94.1% (71.2-99.0%)	76.5% (50.1-93.0%)	0.38
					Specificity	100.0% (99.1-100.0%)	96.1% (93.7-97.7%)	<0.01
PE	422	25	25	25	Sensitivity	100.0% (86.2-100.0%)	100.0% (86.16-100.0%)	1.00
					Specificity	99.8% (98.6-100.0%)	95.7% (93.-97.5%)	<0.01
Either DVT or PE	422	42	41	39	Sensitivity	97.6% (87.4-99.6%)	92.9% (80.5-98.4%)	0.63
					Specificity	99.8% (98.7-100.0%)	92.9% (89.8-95.3%)	<0.01

Table 6.9. Performance of IDEAL-X system in controlled vocabulary mode, compared to Agency for Healthcare Research and Quality Patient Safety Indicator 12, analyzed by patient surgical encounter

6.3.4 Discussion

IDEAL-X is capable of correctly identifying VTE from the free text of radiology reports with very high sensitivity and specificity, surpassing the performance of identification based on AHRQ PSI-12. Clinical quality metrics sourced from clinical records may have increased validity compared to those from administrative data sources.

Customized controlled vocabulary simplifies the deployment process, and is better suited for instances with low positive incidences, including VTE. See Figure 6.1, the system's sensitivity takes more than 400 records to reach 90%. The reason is because when the system constructs its model with online learning mode, it needs enough positive instances to collect features of value to be extracted. When prevalence of a disease is low, it will take more records to accumulate enough positive instances. However, with controlled vocabulary mode, knowledge to be learned is injected by user directly, which alleviated the learning process directly.

IDEAL-X's convenient workflow requires no linguistic expertise from the user, and can be easily adapted to different clinical applications to improve detection and surveillance of medical conditions.

6.4 Summary

The three use cases cover major clinical reports and various narrative styles. As information extraction for simple narration text (semi-structured and template base ones) is straightforward, our discussion here focuses on extraction from complex narration. In general, whether to use interactive annotation or controlled vocabulary depends on several factors: data type, domain of attribute and prevalence of instance.

For numerical data such as age and heart rate, a lexicon is not available. Extraction primarily relies on contextual information, which could be collected during interactive annotation. For nominal data, especially clinical terms which self-express entity they belong to (For example: term "diabetes" refers to a disease, and term "biopsy" refer to a surgical procedure), a controlled vocabulary will be very useful.

Another factor in whether controlled vocabulary is applicable is the domain of attribute. Enumerating terminology manually is simple when the domain is small, for example: race and gender. When a domain is large but still finite, referring to existing knowledge base can ease the effort of vocabulary building. For example, to extract genetic marker, one can start from Cell Cycle Ontology. To nominal data that has infinite domain, such as hospital names, extraction has to be conducted based on contextual information without the support of terminology. This represents the most challenging extraction scenario. And finally, when the prevalence of training instances is low, inputting knowledge directly in the form of controlled vocabulary might be more efficient.

Chapter 7

Online Learning Based Clinical Information Classification

Clinical Decision Support Systems (CDSS) assists healthcare providers in clinical decision making. Most clinical decision support systems employ batching machine learning that do not adapt to changing data environment easily. In this chapter, we study the challenges in clinical classification and extend the online learning functionality of IDEAL-X with integrated classification solutions.

7.1 Motivations and Goals

The impetus for the work comes from providing early stage warning for venous thromboembolism (VTE). Figure 7.1 shows the typical VTE diagnosis scenario of postoperative VTE: First, a patient is admitted to a hospital and undergoes surgery. A blood clot may form in a vein after surgery depending on risk factors like to body habitus, risk of surgery, and other factors. If the blood clot significantly disrupts bloodflow, the patient then develops symptoms. Based on these symptoms, the patient will be recommended for a radiographic test, which may confirm or reject the diagnosis of VTE.

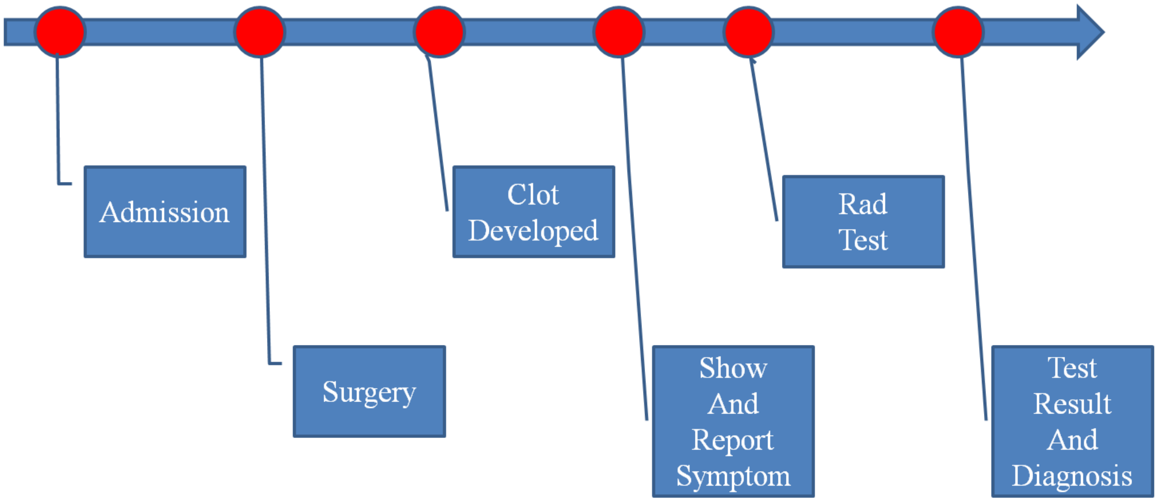


Figure 7.1 VTE development and detection scenario

In the US, there are an estimated 350,000-900,000 VTEs per year, resulting in approximately 100,000 deaths per year [87]. To reduce this disease burden among hospitalized patient, patients who are high risk for VTE could be identified either upon hospital admission or before surgery by CDSS. A CDSS could then prompt clinicians to consider additional measures to prevent the development of VTE.

To a successful CDSS, according to the systematic review [14] of former researches, the following features are critical:

1) The CDSS should operate automatically as an integrated procedure in the regular clinician workflow.

2) The system has to support decision making when patient is hospitalized.

3) The system should be able to recommend clinical care, instead of simple assessment.

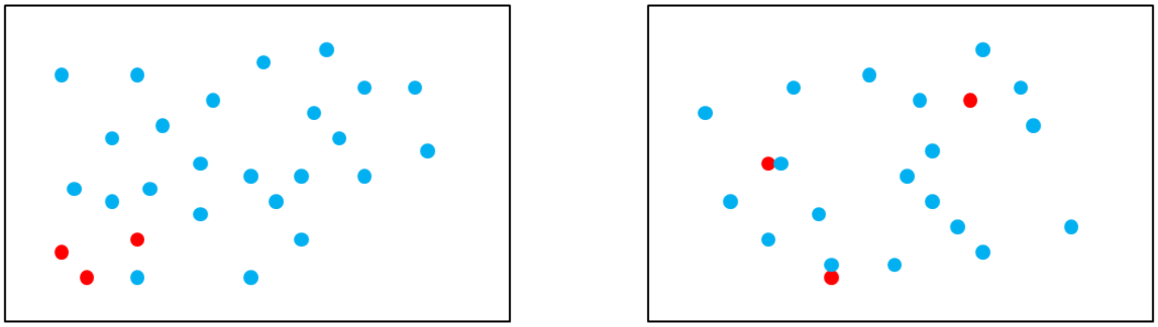
Following these guidelines, we use IDEAL-X to implement a VTE CDSS prototype. It integrates with the current VTE diagnosis workflow to provide an early stage warning for a hospitalized patient. The VTE CDSS serves as a virtual symptom, supplementing common reasons. For prevention, patients would be examined by the CDSS immediately after surgery, and ones with high risk could be recommended for additional medical tests. In this way, the CDSS enhances the clinical workflow without altering its practice.

7.2 Challenges

The following are important challenges to the implementation of a CDSS in a clinical environment.

I) Imbalanced (skewed) data

Most diseases and symptoms have low prevalence. To this type of dataset, positive instances typically represent a very small portion of the overall data set. This presents a challenge to most classification techniques. The situation may be somewhat simplified when the positive data points are close together (see the left diagram of figure 7.2). But in general, distribution of the positive data points does not deviate from the rest of the data (the right diagram of figure 7.2), and it makes classification difficult [89, 90].



Simple Case

Complicated Case

Figure 7.2 Conditions of skewed data

Techniques for classifying imbalanced data include the following: over-sample minority classes, under-sample majority classes, and modifying cost or prediction threshold of different classes for given algorithm [91].

II) Heterogeneous data format

Data warehouse and database of a clinical system store EMR in different formats and with varying degrees of structure (as we have seen previously in Chapter 6). Besides extraction, some data transformation may be necessary. For example, to computer algorithm, time span, such as length of hospitalization, will be more meaningful than admission date and discharge date themselves. Therefore, to many classification tasks, data transformation is an indispensable preprocess procedure. In addition, external knowledge could provide important domain specific information to improve data quality. For example, procedure names may be mapped to degrees of risk.

III) Algorithm design

Imbalanced data challenges traditional machine learning algorithm,. In the extreme case, a dataset dominated by negative instances would lead many machine learning algorithms to classify all instances as negative. As we will show, several algorithms can be adapted for online learning.

7.3 System Architecture

The system inputs information from heterogeneous data sources to generate classification result in real time. It consists of three modules, integrating, transformation and classification, as shown in Figure 7.3.

I) Integration Module

The module integrates information from various sources. Based on unique ID, such as MRN and SSN, information extracted from free text could be merged with structured values pulled from database or data warehouse to form a single structured view.

II) Transformation Module

This module provides various converters for data transformation, which allows the user to normalize the original data to formats that are easy to process. User can also use external knowledge to map raw data to target domain, for example, from the body mass index to risk factors of a given disease.

III) Classification Module

This module analyzes transformed data with online learning based classification algorithms, and predicts class labels. A confidence score is also indicated with each label. The user may select a classification algorithm from the algorithm set provided below.

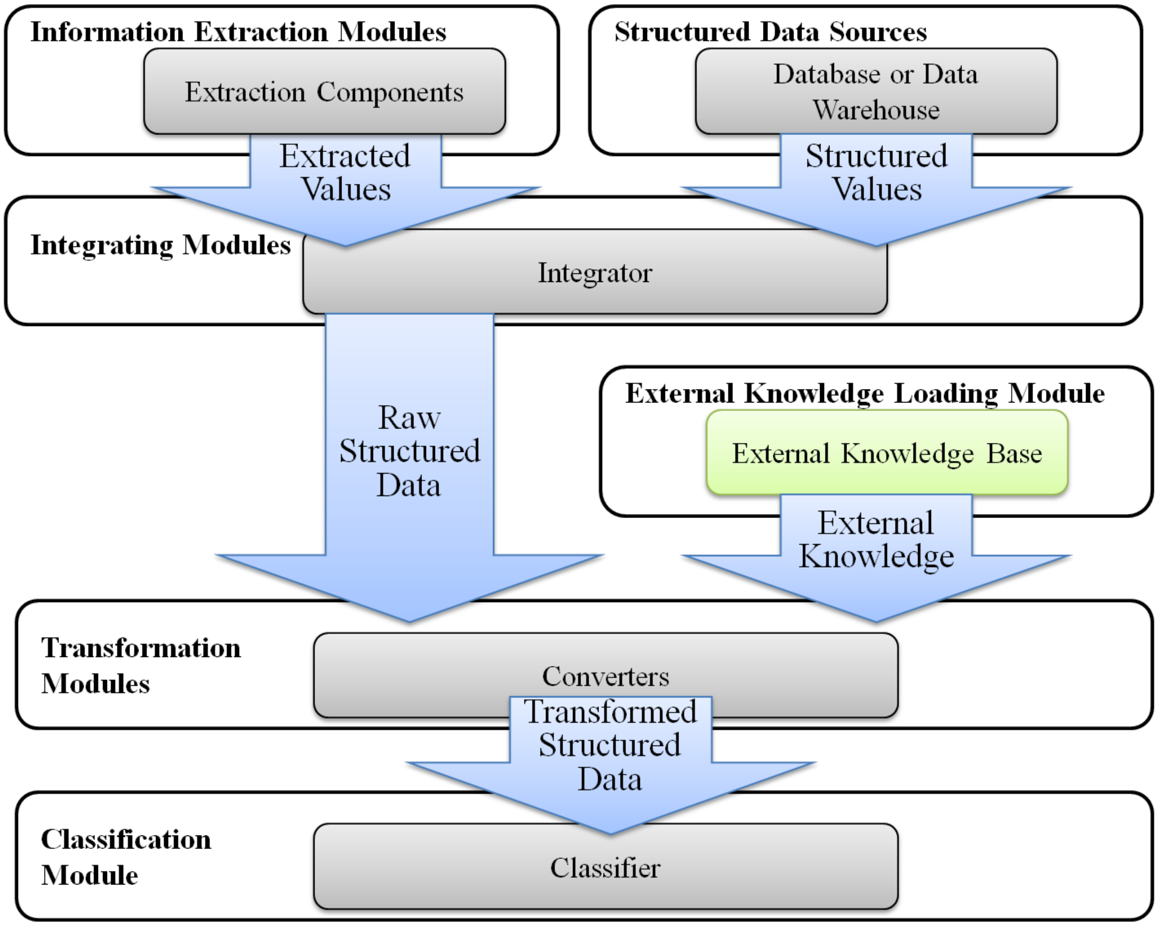


Figure 7.3 Modules of classification

7.4 Algorithms

We have identified several well-known algorithms that can be adapted to work with for online learning in the presence of imbalanced data. An alternative to algorithm modification is through sampling strategies. But sampling is typically performed as a preprocess to classification, and it does not fit the online context.

7.4.1 Na ïve Bayesian

Na ïve Bayesian [92] is well-known for its scalability: both training and testing have time complexity *O(n)*. To adapt the algorithm for online learning, the training step needs to be modified to update the count of classes and attributes after each instance is processed. The probability of each class p (C_k)and the probability of p (x_i | C_k) are updated accordingly (x_i is the value of input vector $X = (x_1, x_2, x_3, \ldots\ldots, x_i)$). These updates can be made in time *O(1)*, hence the overall modified algorithm remains *O(n).*

To predict output, the original prediction function of Na ïve Bayesian can be applied directly in constant time:

$$prediction = arg\ max\ P(C_k)\prod P(x_i \mid C_k)$$

To handle skewed data, especially when costs of misclassification for different classes are not equivalent, bias may be applied to $p\ (C_k)$ of the prediction function so as to prioritize the preference of different classes.

7.4.2 Neural Network

Neural network [93] is naturally adaptive. It makes prediction for one input and use the feedback to update model immediately, so as to make improvement after each epoch. In every epoch, both prediction and updating can be accomplished in real time with *O(1)* time complexity.

Regular backprogation algorithm [94] may be applied directly for training. For instance, weights of a three layers neural network could be updated with the following three steps:

1) The errors of **output layer** is computed using the function

*currentError = output * (1 - output) * (target - output)*

Weight adjustment is computed with

*adjustment = learnignRate * currentInput.get(i) * currentError*

2) Then, error in **hidden layer** is computed with the following function:

*currentError = currentOutput * (1 - currentOutput) * sumOfOutputlayer*

where sumOfOutputLayer is obtained with the following:

*sumOfOutputlayer = Sum(error of output neuron * weight between this hidden neuron and the output neuron)*

Weight adjustment is computed with:

*adjustment = learnignRate * currentInput.get(i) * currentError*

3) At the end, input weights in both hidden and output layer are updated by

UpdatedWeight = OriginalWeight + adjustment

Neural network makes prediction by generating output for each layer in sequence:

1) To input layer:

$$Original\ Sum = \Sigma inputs\ of\ neuron$$

2) To inputs of hidden and output layers, the original sum were applied with an activation function:

$$f(x) = 1 / (1 + e^{-x})$$

With this sigmoid function, values used to feed the output layer or generate final result are normalized to the range (0, 1).Prediction is made based on values of output neurons. By adjusting the learning rate of positive and negative cases during training (in step 1 and 2), the algorithm may weigh the impacts of positive and negative instances differently. In this way, the effects of data imbalance may be cushioned.

7.4.3 k-Nearest Neighbors

kNN [95] uses local information to make prediction and its computation only happens at prediction stage. Training of KNN is straightforward. Each new data point is inserted into the dataset without computation. To make a prediction, an input instance acts as a query on the entire dataset. The goal is to find the *k* closest data points. The most frequent label in the subset is used to label the input instance. To calculate the distance between a query point and other data points, various distance function may be applied: Euclidean, Manhattan, Minkowski, among others.

To identify the nearest data points, the algorithm has to examine all the data points in the corpus. This is time complexity *O(n).* This is a potential bottle for large dataset. Dividing the dataset into blocks and judicious use of indexes would help to improve the efficiency of searching.

To rebalance skewed data, bias could be implemented to weigh instances of different classes differently. A majority vote for the prediction will be based on the weighted sum.

7.5 Results

I) Datasets

We obtained electronic medical record data on patients who were admitted to Emory University Orthopedic and Spine Hospital during 2009 to 2014 and obtained 13,248 encounters. Forty-one positive cases, defined as patients who had hip fracture, hip replacement, knee replacement, or spine surgery and were diagnosed with VTE during the hospitalization, were identified. All the other instances were defined as negative cases. The test data was manually annotated by physician annotators from Emory Hospital.

II) Evaluation Metrics

In clinical decision support, a positive case can be rare but critical. Instead of inspecting the overall accuracy, we focus on positive predictive value (PPV) and true positive rate (Sensitivity). In other words, the precision and recall of positive case are reported. The detail of each metric and its clinical impact in this use case are as follows:

Precision of Positive Case (PPV):

$$PPV = TP / (TP + FP)$$

This indicates that if a case is detected, what is the probability of having VTE.

Recall of Positive Case (TPR):

$$TPR = TP / (TP + FN)$$

This indicates that if a case is detected, what is the percentage of VTE patient we can detect based on prediction.

III) Experiment Setting

We ordered records in the dataset based on admission date to simulate real world data stream. Input attributes used for classification include age, BMI, surgery, pharmacologic prophylaxis medication, mechanical prophylaxis and ICD code of cardiovascular disease. All these input attributes have been transformed to boolean values, 0 or 1, in order to make the implemented algorithms, both numeric and categorical, comparable.

When a radiology test was ordered, the physician specified the symptom or reason for ordering the test. We collect statistics on the reasons as background information based on radiation reports from 2009 to 2014. Table 7.1 shows major reasons (with recall above 3%) for radiation tests and their precision and recall for VTE prediction. This data can be used as the baseline for comparing clinical effectiveness of VTE CDSS.

Symptom	**Precision**	**Recall**
edema	8.9%	17.6%
acute shortness of breath	13.7%	18.0%
pain in limb	7.7%	8.5%
chest pain unspecified	9.0%	5.2%
pulmonary embolism	8.8%	3.3%
vein thrombosis lower leg	13.5%	4.7%

Table 7.1 Common reasons of VTE

IV) Performance Evaluation

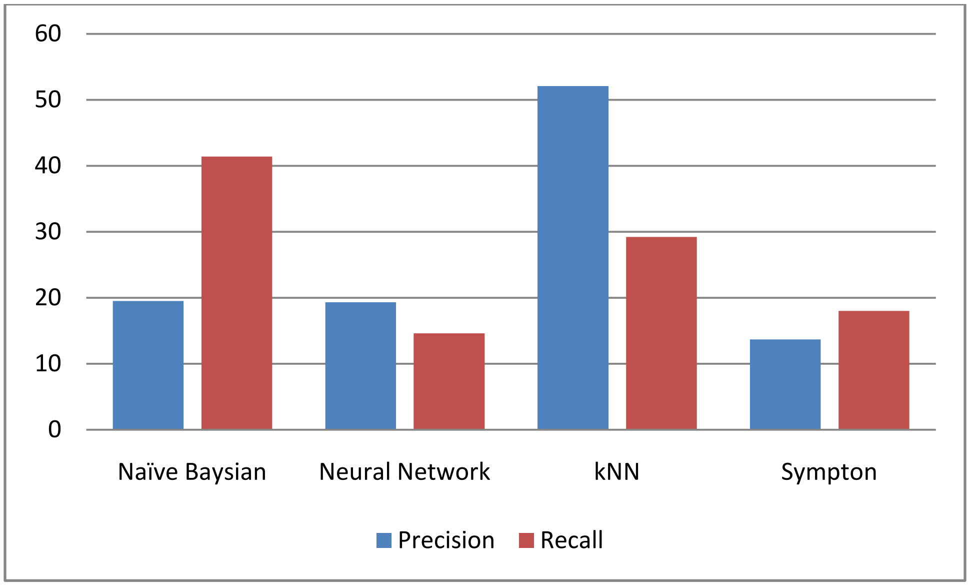


Figure 7.4 Testing result of VTE identification

Figure 7.4 shows the results for the algorithms that we have implemented. The predictions of Naïve Bayesian and kNN are more accurate than the "best" real symptom "acute shortness of breath". In particular, if one only relies on the result of Naïve Bayesian, the CDSS "symptom" can predict VTE with 41.4% probability. If the CDSS predicts "VTE", the chance of VTE is 19.5%. The precision and recall of negative case are above 99% for all algorithms.

Using bias parameters, all the algorithms could be tuned to increase either the precision or recall. For example, to detect more VTE patient, one can adjust parameter to improve recall, though precision will correspondingly decrease. The system also allows the user to select multiple algorithms to construct an ensemble model. This ensemble model uses majority voting to generate the final predication result.

7.6 Conclusion

The built-in online learning classification component further improved the functionality of *IDEAL-X*. Combined with the information extraction module, the system provides a powerful architecture for clinical decision support based on information integrated from heterogeneous sources. Motivated by the VTE early prediction case, we study the special challenges and guidelines of CDSS deployment, and examine available algorithms to provide concrete solution. Experiment results reveal the clinical value this kind of system. We implement algorithms as a toolkit, and allow extension to incorporate new classification algorithms conveniently.

Chapter 8

Conclusion and Future Work

8.1 Conclusion

This dissertation has focused on resolving clinical information extraction and classification tasks using online machine learning based algorithms and human computer interaction. While considerable attention has been given to structured and standardized reporting, most medical reporting systems still allow for (and thus encourage) narrative text descriptions. There is a lack of effective tools to ease the process of information extraction, data transformation and normalization.

IDEAL-X provides a bridge between free-form text reports and structured reports. Its workflow follows the conventional process for manual extraction of information from text, but it noninvasively learns and gradually improves its ability to automatically locate relevant information. The system is powered by an online learning based engine with customizable domain knowledge, and comes with a natural workflow and interface. Using the system requires no linguistic expertise, and the internal algorithms are generic, thus adaptable to diverse clinical reports. In addition, IDEAL-X supports standardizing and integrating extracted data to enhance the utility of the output. A similar online machine learning based solution has also been implemented for classification to facilitate clinical decision support.

8.2 Future Work

A number of extensions are possible to improve the usage and applicability of IDEAL-X and its conceptual framework.

I) Explore a broader set of machine learning algorithms

Machine learning is the most promising technique for identifying candidate text chunks. Besides HMM, we have explored other classifiers such as Naive Bayes classifier and neural networks. A systematic study of different classifiers and their combinations (including Conditional Random Field and Support Vector Machine [96]) for online machine learning based data extraction would be provide invaluable insights. The key challenges in adapting any algorithm in an interactive, online setting are responsiveness and scalability.

II) Automating controlled vocabulary driven standardization

Extracted values might be highly heterogeneous in their raw form, and therefore standardization is a necessary step to increase the utility of the output results. Traditional coding systems [97-101] annotate information with standardized dictionaries. Besides static mapping, one can enhance the mapping at running time with online learning enhanced with semi-automated assistance. Other domain knowledge, such as report standards, controlled vocabulary and ontology, can also be used to enhance medical coding process.

III) Advanced extraction template

Currently the system supports single and multiple value extraction in tabular format, but alternative, more flexible schemas may be useful for representing complex relationships. Examples include semi-structured, hierarchical and network models that can support advance information extraction use cases such as event extraction [69, 102, 103], relation extraction [104, 105] and template extraction [65, 106, 107]. Important challenges include finding suitable user interaction models and machine learning techniques.

IV) Extensible extraction model

To maximize its applicability and portability, IDEAL-X employs generic tactics to support information extraction. However, we also recognize that given a particular task, domain specific information extraction strategies are likely to be more effective. To accommodate potential extraction strategies that take advantage of domain knowledge, an approach would be to allow user-defined plug-ins for tasks such as filtering or searching. Such an open implementation will enable reuse of the system interface and the standardization of the user interaction, and would reduce the cost for the development of similar systems for other research projects.

Bibliography

1. The ASPE Expert Panel on Cancer Reporting Information Technology College of American Pathologists and The Altarum Institute, W., DC. *Electronic Reporting in Pathology: Requirements and Limitations*. 2009; Available from: http://aspe.hhs.gov/sp/reports/2010/erpreqlim/report.shtml.
2. Clunie, D.A., *DICOM structured reporting and cancer clinical trials results.* Cancer informatics, 2007. **4**: p. 33.
3. *Anatomic Pathology Structured Reports*. Available from: http://wiki.ihe.net/index.php?title=Anatomic_Pathology_Structured_Reports.
4. *CAP Cancer Protocols*. Available from: http://www.cap.org/web/home/resources/cancer-reporting-tools/cancer-protocols.
5. *Cancer Text Information Extraction System*. Available from: http://caties.cabig.upmc.edu/.
6. *Cancer Data Standards Registry and Repository*. Available from: http://cbiit.nci.nih.gov/ncip/biomedical-informatics-resources/interoperability-and-semantics/metadata-and-models#caDSR.
7. *Health Level Seven International (HL7)*. Available from: http://www.hl7.org/.
8. *IDEAL-X Demo Video*. [cited 2015 January 20]; Available from: http://youtu.be/Q-DrWi31nv0.
9. Smale, S. and Y. Yao, *Online learning algorithms.* Foundations of Computational Mathematics, 2006. **6**(2): p. 145-170.
10. Shalev-Shwartz, S., *Online learning and online convex optimization.* Foundations and Trends in Machine Learning, 2011. **4**(2): p. 107-194.
11. Shalev-Shwartz, S., *Online learning: Theory, algorithms, and applications.* 2007.
12. *Unified Medical Language System (UMLS)*. Available from: http://www.nlm.nih.gov/research/umls/.
13. Garg, A.X., et al., *Effects of computerized clinical decision support systems on practitioner performance and patient outcomes: a systematic review.* JAMA, 2005. **293**(10): p. 1223-38.
14. Kawamoto, K., et al., *Improving clinical practice using clinical decision support systems: a systematic review of trials to identify features critical to success.* Bmj, 2005. **330**(7494): p. 765.

15. Kononenko, I., *Machine learning for medical diagnosis: history, state of the art and perspective.* Artificial Intelligence in medicine, 2001. **23**(1): p. 89-109.
16. Demner-Fushman, D., W.W. Chapman, and C.J. McDonald, *What can natural language processing do for clinical decision support?* J Biomed Inform, 2009. **42**(5): p. 760-72.
17. Garvin, J.H., et al., *Automated extraction of ejection fraction for quality measurement using regular expressions in Unstructured Information Management Architecture (UIMA) for heart failure.* J Am Med Inform Assoc, 2012. **19**(5): p. 859-66.
18. Gupta, D., M. Saul, and J. Gilbertson, *Evaluation of a deidentification (De-Id) software engine to share pathology reports and clinical documents for research.* Am J Clin Pathol, 2004. **121**(2): p. 176-86.
19. Beckwith, B.A., et al., *Development and evaluation of an open source software tool for deidentification of pathology reports.* BMC Med Inform Decis Mak, 2006. **6**: p. 12.
20. Neamatullah, I., et al., *Automated de-identification of free-text medical records.* BMC Med Inform Decis Mak, 2008. **8**: p. 32.
21. Gardner, J., et al., *An evaluation of feature sets and sampling techniques for de-identification of medical records*, in *Proceedings of the 1st ACM International Health Informatics Symposium*2010, ACM: Arlington, Virginia, USA. p. 183-190.
22. Deleger, L., et al., *Large-scale evaluation of automated clinical note de-identification and its impact on information extraction.* J Am Med Inform Assoc, 2013. **20**(1): p. 84-94.
23. South, B.R., et al., *A prototype tool set to support machine-assisted annotation*, in *Proceedings of the 2012 Workshop on Biomedical Natural Language Processing*2012, Association for Computational Linguistics: Montreal, Canada. p. 130-139.
24. Ogren, P.V., *Knowtator: a protege plug-in for annotated corpus construction*, in *Proceedings of the 2006 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology: companion volume: demonstrations*2006, Association for Computational Linguistics: New York, New York. p. 273-275.
25. Eapen, D.J., et al., *Aggregate risk score based on markers of inflammation, cell stress, and coagulation is an independent predictor of adverse cardiovascular outcomes.* Journal of the American College of Cardiology, 2013. **62**(4): p. 329-337.
26. Zhou, L. and G. Hripcsak, *Temporal reasoning with medical data—a review with emphasis on medical natural language processing.* Journal of biomedical informatics, 2007. **40**(2): p. 183-202.
27. Sun, W., A. Rumshisky, and O. Uzuner, *Temporal reasoning over clinical text: the state of the art.* Journal of the American Medical Informatics Association, 2013. **20**(5): p. 814-819.
28. Miwa, M., et al., *Event extraction with complex event classification using rich features.* Journal of bioinformatics and computational biology, 2010. **8**(01): p. 131-146.

29. Savova, G.K., et al., *Mayo clinical Text Analysis and Knowledge Extraction System (cTAKES): architecture, component evaluation and applications.* Journal of the American Medical Informatics Association, 2010. **17**(5): p. 507-513.

30. Schlangen, D., M. Stede, and E.P. Bontas. *Feeding owl: Extracting and representing the content of pathology reports*. in *Proceeedings of the Workshop on NLP and XML (NLPXML-2004): RDF/RDFS and OWL in Language Technology*. 2004. Association for Computational Linguistics.

31. Schadow, G. and C.J. McDonald. *Extracting structured information from free text pathology reports*. in *AMIA Annual Symposium Proceedings*. 2003. American Medical Informatics Association.

32. Hobbs, J.R., *Information extraction from biomedical text.* Journal of Biomedical Informatics, 2002. **35**(4): p. 260-264.

33. Tanabe, L. and W.J. Wilbur, *Tagging gene and protein names in biomedical text.* Bioinformatics, 2002. **18**(8): p. 1124-1132.

34. Crowley, R.S., et al., *caTIES: a grid based system for coding and retrieval of surgical pathology reports and tissue specimens in support of translational research.* Journal of the American Medical Informatics Association, 2010. **17**(3): p. 253-264.

35. Xu, H., et al., *MedEx: a medication information extraction system for clinical narratives.* Journal of the American Medical Informatics Association, 2010. **17**(1): p. 19-24.

36. Christensen, L.M., et al. *ONYX: a system for the semantic analysis of clinical text*. in *Proceedings of the Workshop on Current Trends in Biomedical Natural Language Processing*. 2009. Association for Computational Linguistics.

37. Friedman, C., et al., *Automated encoding of clinical documents based on natural language processing.* Journal of the American Medical Informatics Association, 2004. **11**(5): p. 392-402.

38. Harper, P.R., *A review and comparison of classification algorithms for medical decision making.* Health Policy, 2005. **71**(3): p. 315-331.

39. Yan, H., et al., *A multilayer perceptron-based medical decision support system for heart disease diagnosis.* Expert Systems with Applications, 2006. **30**(2): p. 272-281.

40. Subasi, A., *Medical decision support system for diagnosis of neuromuscular disorders using DWT and fuzzy support vector machines.* Computers in biology and medicine, 2012. **42**(8): p. 806-815.

41. Ocak, H., *A medical decision support system based on support vector machines and the genetic algorithm for the evaluation of fetal well-being.* Journal of medical systems, 2013. **37**(2): p. 1-9.

42. Lisboa, P.J. and A.F. Taktak, *The use of artificial neural networks in decision support in cancer: a systematic review.* Neural networks, 2006. **19**(4): p. 408-415.

43. West, D., et al., *Ensemble strategies for a medical diagnostic decision support system: A breast cancer diagnosis application.* European Journal of Operational Research, 2005. **162**(2): p. 532-551.

44. Hunt, D.L., et al., *Effects of computer-based clinical decision support systems on physician performance and patient outcomes: a systematic review.* Jama, 1998. **280**(15): p. 1339-1346.

45. Sintchenko, V., E. Coiera, and G.L. Gilbert, *Decision support systems for antibiotic prescribing.* Current opinion in infectious diseases, 2008. **21**(6): p. 573-579.
46. Ciravegna, F. and Y. Wilks, *Designing adaptive information extraction for the semantic web in amilcare.* Annotation for the semantic web, 2003. **96**.
47. Ciravegna, F. *Adaptive information extraction from text by rule induction and generalisation*. in *International Joint Conference on Artificial Intelligence*. 2001. LAWRENCE ERLBAUM ASSOCIATES LTD.
48. Settles, B. *Closing the loop: Fast, interactive semi-supervised annotation with queries on features and instances*. in *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. 2011. Association for Computational Linguistics.
49. Aberdeen, J., et al., *The MITRE Identification Scrubber Toolkit: design, training, and assessment.* International journal of medical informatics, 2010. **79**(12): p. 849-859.
50. Gobbel, G., et al. *Automated annotation of electronic health records using computer-adaptive learning tools*. in *AMIA Annu Meeting, Washington, DC*. 2011.
51. Settles, B., *Active learning literature survey.* University of Wisconsin, Madison, 2010. **52**: p. 55-66.
52. Olsson, F., *A literature survey of active machine learning in the context of natural language processing.* 2009.
53. Chen, Y., et al., *Applying active learning to supervised word sense disambiguation in MEDLINE.* Journal of the American Medical Informatics Association, 2013: p. amiajnl-2012-001244.
54. Chen, Y., et al., *Applying active learning to high-throughput phenotyping algorithms for electronic health records data.* Journal of the American Medical Informatics Association, 2013.
55. Chapman, W.W., et al., *A simple algorithm for identifying negated findings and diseases in discharge summaries.* J Biomed Inform, 2001. **34**(5): p. 301-10.
56. Chapman, W.W., et al., *Evaluation of negation phrases in narrative clinical reports.* Proc AMIA Symp, 2001: p. 105-9.
57. Elkin, P.L., et al., *A controlled trial of automated classification of negation from clinical notes.* BMC Med Inform Decis Mak, 2005. **5**: p. 13.
58. Huang, Y. and H.J. Lowe, *A novel hybrid approach to automated negation detection in clinical radiology reports.* J Am Med Inform Assoc, 2007. **14**(3): p. 304-11.
59. Mutalik, P.G., A. Deshpande, and P.M. Nadkarni, *Use of general-purpose negation detection to augment concept indexing of medical documents: a quantitative study using the UMLS.* J Am Med Inform Assoc, 2001. **8**(6): p. 598-609.
60. Agarwal, S. and H. Yu, *Biomedical negation scope detection with conditional random fields.* J Am Med Inform Assoc, 2010. **17**(6): p. 696-701.
61. Zhou, L., et al., *A temporal constraint structure for extracting temporal information from clinical narrative.* J Biomed Inform, 2006. **39**(4): p. 424-39.

62. Bellazzi, R., L. Sacchi, and S. Concaro, *Methods and tools for mining multivariate temporal data in clinical and biomedical applications.* Conf Proc IEEE Eng Med Biol Soc, 2009. **2009**: p. 5629-32.
63. Zhou, L., et al., *System architecture for temporal information extraction, representation and reasoning in clinical narrative reports.* AMIA Annu Symp Proc, 2005: p. 869-73.
64. Harkema, H., et al., *ConText: an algorithm for determining negation, experiencer, and temporal status from clinical reports.* J Biomed Inform, 2009. **42**(5): p. 839-51.
65. Sohn, S., et al., *Comprehensive temporal information detection from clinical text: medical events, time, and TLINK identification.* J Am Med Inform Assoc, 2013.
66. Tang, B., et al., *A hybrid system for temporal information extraction from clinical text.* J Am Med Inform Assoc, 2013.
67. Sun, W., A. Rumshisky, and O. Uzuner, *Evaluating temporal relations in clinical text: 2012 i2b2 Challenge.* Journal of the American Medical Informatics Association, 2013.
68. Hazlehurst, B., et al., *MediClass: A system for detecting and classifying encounter-based clinical events in any electronic medical record.* J Am Med Inform Assoc, 2005. **12**(5): p. 517-29.
69. Gold, S., et al., *Extracting structured medication event information from discharge summaries.* AMIA Annu Symp Proc, 2008: p. 237-41.
70. Li, Z., et al., *Lancet: a high precision medication event extraction system for clinical text.* J Am Med Inform Assoc, 2010. **17**(5): p. 563-7.
71. Aramaki E, M.Y., Tonoike M, Ohkuma T, Mashuichi H, and Ohe K. *TEXT2TABLE: Medical Text Summarization System Based on Named Entity Recognition and Modality Identification*. in *HLT-NAACL2003 Workshop on BioNLP*. 2009.
72. Preece, J., et al., *Human-computer interaction*. 1994: Addison-Wesley Longman Ltd.
73. OpenNLP, A.; Available from: http://opennlp.apache.org/.
74. Manning, C.D., P. Raghavan, and H. Schütze, *Introduction to information retrieval*. Vol. 1. 2008: Cambridge university press Cambridge.
75. Elliott, R.J., L. Aggoun, and J.B. Moore, *Hidden Markov Models*. 1994: Springer.
76. Freitag, D. and A. McCallum, *Information extraction with HMM structures learned by stochastic optimization.* AAAI/IAAI, 2000. **2000**: p. 584-589.
77. Viterbi, A.J., *Error bounds for convolutional codes and an asymptotically optimum decoding algorithm.* Information Theory, IEEE Transactions on, 1967. **13**(2): p. 260-269.
78. Fürnkranz, J., *Separate-and-conquer rule learning.* Artificial Intelligence Review, 1999. **13**(1): p. 3-54.
79. *SNOMED Clinical Terms*. Available from: http://www.nlm.nih.gov/research/umls/Snomed/snomed_main.html.
80. *NCI Thesaurus*. Available from: http://ncit.nci.nih.gov/.
81. *Medical Vocabulary Generating Tool*. Available from: https://www.google-melange.com/gsoc/project/details/google/gsoc2014/sid270592/5649050225344512.

82. Srigley, J.R., et al., *Standardized synoptic cancer pathology reporting: A population‐based approach.* Journal of surgical oncology, 2009. **99**(8): p. 517-524.

83. Gill, A.J., et al., *Synoptic reporting improves histopathological assessment of pancreatic resection specimens.* Pathology, 2009. **41**(2): p. 161-167.

84. Leslie, K.O. and J. Rosai. *Standardization of the surgical pathology report: formats, templates, and synoptic reports*. in *Seminars in diagnostic pathology*. 1994.

85. *Human Disease Ontology*. Available from: http://disease-ontology.org/.

86. *Cell Cycle Ontology*. Available from: http://www.cellcycleontology.org/.

87. Rathbun, S., *The surgeon general's call to action to prevent deep vein thrombosis and pulmonary embolism.* Circulation, 2009. **119**(15): p. e480-e482.

88. AHRQ. *Patient Safety Indicators Technical Specifications*. Available from: http://www.qualityindicators.ahrq.gov/modules/PSI_TechSpec.aspx.

89. García, V., R.A. Mollineda, and J.S. Sánchez, *On the k-NN performance in a challenging scenario of imbalance and overlapping.* Pattern Analysis and Applications, 2008. **11**(3-4): p. 269-280.

90. López, V., et al., *An insight into classification with imbalanced data: Empirical results and current trends on using data intrinsic characteristics.* Information Sciences, 2013. **250**: p. 113-141.

91. Ganganwar, V., *An overview of classification algorithms for imbalanced datasets.* International Journal of Emerging Technology and Advanced Engineering, 2012. **2**(4): p. 42-47.

92. Rish, I. *An empirical study of the naive Bayes classifier*. in *IJCAI 2001 workshop on empirical methods in artificial intelligence*. 2001. IBM New York.

93. Hagan, M.T., H.B. Demuth, and M.H. Beale, *Neural network design*. 1996: Pws Pub. Boston.

94. Anthony J. papagelis, D.S.K. *Multi-Layer Perceptron*. Available from: http://www.cse.unsw.edu.au/~cs9417ml/MLP2/index.html.

95. Altman, N.S., *An introduction to kernel and nearest-neighbor nonparametric regression.* The American Statistician, 1992. **46**(3): p. 175-185.

96. Jiang, M., et al., *A study of machine-learning-based approaches to extract clinical entities and their assertions from discharge summaries.* Journal of the American Medical Informatics Association, 2011. **18**(5): p. 601-606.

97. Lussier, Y.A., L. Shagina, and C. Friedman, *Automating SNOMED coding using medical language understanding: a feasibility study.* Proc AMIA Symp, 2001: p. 418-22.

98. Cimino, J.J., *Review paper: coding systems in health care.* Methods Inf Med, 1996. **35**(4-5): p. 273-84.

99. Quan, H., et al., *Coding algorithms for defining comorbidities in ICD-9-CM and ICD-10 administrative data.* Med Care, 2005. **43**(11): p. 1130-9.

100. Crammer, K., et al., *Automatic code assignment to medical text*, in *Proceedings of the Workshop on BioNLP 2007: Biological, Translational, and Clinical Language Processing*2007, Association for Computational Linguistics: Prague, Czech Republic. p. 129-136.

101. Stanfill, M.H., et al., *A systematic literature review of automated clinical coding and classification systems.* J Am Med Inform Assoc, 2010. **17**(6): p. 646-51.

102. Madhyastha, H.V., N. Balakrishnan, and K. Ramakrishnan. *Event information extraction using link grammar*. in *Research Issues in Data Engineering: Multi-lingual Information Management, 2003. RIDE-MLIM 2003. Proceedings. 13th International Workshop on*. 2003. IEEE.

103. Alphonse, E., et al. *Event-based information extraction for the biomedical domain: the Caderige project*. in *Proceedings of the International Joint Workshop on Natural Language Processing in Biomedicine and its Applications*. 2004. Association for Computational Linguistics.

104. Sarawagi, S., *Information Extraction.* Found. Trends databases, 2008. **1**(3): p. 261-377.

105. Cunningham, H., *Information extraction, automatic.* Encyclopedia of Language and Linguistics, 2005: p. 665-677.

106. Vargas-Vera, M., et al., *Template-driven information extraction for populating ontologies.* 2001.

107. Mooney, R.J. and R. Bunescu, *Mining knowledge from text using information extraction.* ACM SIGKDD explorations newsletter, 2005. **7**(1): p. 3-10.

Appendix

Table 1. Attributes for Cardiac Catheterization Procedure Reports (Dataset 1) Test Case

Aortic Diastolic (Ao) Pressure	Aortic Diastolic (Ao) Systolic Pressure	Aortic Diastolic (Ao) Mean Pressure
Left Ventricular End Diastolic Pressure (LVEDP)	Left Ventricular (LV) Systolic	Heparin Amount
Bivalirudin Amount	Abciximab Amount	Fentanyl Amount
Midazolam Amount	Nitroglycerin Amount	Acetylcholine Amount
Heparin Dosage	Bivalirudin Dosage	Abciximab Dosage
Fentanyl Dosage	Midazolam Dosage	Nitroglycerin Dosage
Acetylcholine Dosage		

Table 2. Attributes (Stenosis Values of) for Coronary Angiogram Reports (Dataset 2) Test Case

Stenosis value of:

Left Main Coronary Artery	First Diagonal Branches	Second Diagonal Branches
Proximal Circumflex Coronary Artery	Mid Circumflex Coronary Artery	Distal Circumflex Coronary Artery
Ramus	First Obtuse Marginal Branches	Second Obtuse Marginal Branches
Third Obtuse Marginal Branches	Proximal Right Coronary Artery	Mid Right Coronary Artery
Distal Right Coronary Artery	Proximal Circumflex Coronary Artery	Mid Circumflex Coronary Artery
Distal Circumflex Coronary Artery		

Table 3. Attributes for Complex Narration Data Extraction for Dataset 3 Test Case

a. Diseases:

Diabetes	Coronary Artery Bypass Grafting	Heart Transplant
Stroke	Peripheral Vascular Disease	Coronary Artery Disease
Asthma	Chronic Obstructive Pulmonary Disease	Percutaneous coronary intervention
Myocardial Infarction	Hypertension	Atrial Flutte
Alcohol	Heart Failure	Atrial Fibrillation

b. Medications:

Angiotensin II Receptor Blocker	Thiazides	Warfarin
Aspirin	Thienopyridine	Calcium Channel Blockers
Beta-Blockers	Statin	Loop Diuretics
Angiotensin-Converting-Enzyme Inhibitor		