Ann K. Irvine. Natural Language Processing and Temporal Information Extraction in Emergency Department Triage Notes. A Master's Paper for the M.S. in I.S degree. April, 2008. 56 pages. Advisor: Stephanie W. Haas

Electronic patient records, including the Emergency Department (ED) Triage Note (TN), provide a rich source of textual information. Processing clinical texts to create important pieces of structured information will be useful to clinicians treating patients, clinicians in training, and researchers and practitioners in biosurveillance. This work applies natural language processing (NLP) and information extraction (IE) techniques to the TN genre of text. In particular, it presents the Triage Note Temporal Information Extraction System (TN-TIES), which combines a shallow parser, machine learned classifiers, and hand-written rules to identify, extract, and interpret temporal information in TNs in preparation for the automatic creation of a timeline of events leading up to a patient's visit to the ED. The success of TN-TIES suggests that NLP and IE techniques are appropriate for the genre and that the automatic production of a timeline of TN events is a realistic application.

Headings:

Medical informatics – Technological innovations

Medical records – Data processing

Natural Language Processing

Semantics – Data processing

Temporal event analysis

Text processing (Computer Science)

NATURAL LANGUAGE PROCESSING AND TEMPORAL INFORMATION EXTRACTION IN EMERGENCY DEPARTMENT TRIAGE NOTES

by
Ann K. Irvine

A Master's paper submitted to the faculty
of the School of Information and Library Science
of the University of North Carolina at Chapel Hill
in partial fulfillment of the requirements
for the degree of Master of Science in
Information Science.

Chapel Hill, North Carolina

April 2008

Approved by

Stephanie W. Haas

I. Introduction

When patients arrive at the Emergency Department (ED), triage nurses must evaluate and record their status extremely quickly. At the triage stage of the ED visit, three fields of information are documented as part of the patient's record: a brief Chief Complaint (CC), a timestamp, and the Triage Note (TN), which is a large free-text field. The descriptions in the TN are usually in unstructured natural language and a few phrases long. More and more, hospitals are maintaining these elements as part of electronic patient records, which are saved in databases for some time. These unique pieces of textual information are ripe for information extraction and text mining analyses. One such application is in the automatic generation of a timeline of events leading up to a patient's visit as described in a triage note. Such a timeline may allow doctors to more efficiently and accurately understand a patient's condition. This application involves extracting events, symptoms, and temporal expressions from the triage note and ordering them all in a correct and useful way.

A. Objectives

The major goal of this research is to build a system that can accurately identify all clinically significant events (symptoms or incidents) in a TN and, by extracting and interpreting temporal information and incorporating domain knowledge and assumptions,

produce an accurate timeline of events. This type of language processing can unlock the valuable data that exists within medical records (Hripcsak, Friedman, Alderson, DuMouchel, Johnson, & Clayton, 1995). The work that I present focuses on identifying, extracting, and interpreting temporal information in TN text. The results do not complete the project but make substantial contributions to its long-term goals. Using a combination of shallow natural language processing, supervised machine learning algorithms, and domain knowledge, I work towards achieving an operational degree of accuracy in extracting, classifying, and interpreting many of the temporal expressions (TEs) found in the domain of TN text.

In addition to building a practical application, my work aims to describe the specific challenges and opportunities that the TN domain of text presents. The idea that standard medical records may offer significant research opportunities has reached general audiences outside of the biomedical informatics community in recent years (de Lusignan & van Weel, 2006). In order to harness the potential that exists in the availability of electronic medical records, the natural language processing and bioinformatics research communities must be familiar with the content and language use within such documents.

B. Importance

The larger project, of which my current work is a part, will eventually benefit health care professionals and the patients that they serve. Suominen et al. have noted that "language technology may help nurses give more efficient care for their patients through the possibility to understand faster and more efficiently the content of narratives" (2007,

S297). Furthermore, public health officials and researchers are interested in monitoring the status of patients entering EDs and their reasons for the visits. Automatically processing and normalizing the free-text of TNs will make the data more accessible and useful. Finally, using the system, ED clinicians in training may gain additional understanding of how to generate or use TN text to assist them in their treatment of patients.

This work may also benefit other researchers working on natural language processing projects related to routine medical records. Much published NLP work, particularly in the biomedical field, is proprietary or does not include sufficient processing details, so my thorough discussion will be useful for beginning researchers and students in the field. Finally, this work provides an important attempt at understanding and processing text written in the genre of TN language, which is distinct from standard English. In what follows, I provide a detailed documentation of my methods and results, as well as obstacles and opportunities.

II. Background

A. Triage Note Text

When patients arrive at the ED, the triage nurse on duty records information about their status. The information that nurses record is often a combination of the patient's words and the nurse's interpretation of the patient's report. Nurses also often report their own observations (e.g., *appears anxious in triage*). The triage nurses complete both the brief

Chief Complaint (CC) field and the larger Triage Note (TN) field. The two free-text fields become part of the patient record, along with a timestamp indicating when the record was created. In this work, we use data that has been recorded electronically, though some hospitals still have paper-based systems.

Portions of two example triage records are shown in Figure 1. The information recorded in the CC field is very brief, often just 1 or 2 words describing a patient's primary reason for coming to the ED. In my current work, I focus on the TN field of the triage report. The TN includes a (usually) longer description of a patient's circumstances leading up to the ED visit. The descriptions typically include symptoms (e.g., *cough and cold* or *vomiting*), previous visits or treatments (e.g., *Seen here last night for same*), and related incidents (e.g., *bike landed on right leg*). We call all of the symptoms, previous treatments, and incidents noted in the TN 'events' (Haas, Travers, Waller, & Kramer-Duffield, 2007). We focus on a TN-specific definition of event because these items are especially important to clinicians and, thus, important to display on the timeline. Hospitals vary somewhat in the information that nurses record in the TN. The TNs from some of the hospitals in our dataset include a large amount of patients' personal and family medical histories. TNs from most hospitals whose records are included in our data, however, are 2-4 phrases long. The average TN length in our sample of records (described below) is 31 words.

2006-11-22 00:54:00 COUGH
Pt Has had cough and cold for last 1 1/2 weeks. Seen here last night for same. Pt coughing and vomiting, can't sleep. Pt has clear bil BS. No fever

2007-04-17 15:48:00 WRECKED MOTORCYCLE
MCA approx 30 minutes ago-bike landed on right leg. Now with painful ambulation-increased pain right knee

Figure 1: Example Triage Records

Despite the variety of events found in TN, the language that triage nurses use is consistent and of a particular genre. Travers and Haas (2003) explore the language used in the CC, which has a specific subject matter and purpose. In the CC, nurses use semi-standard abbreviations and shorthand notations. Much of the language of TNs is similar to that of CCs, particularly references to incidents and symptoms, which are often noun phrases. Additionally, TNs are fairly consistent in the verb phrases and grammatical structures that they employ. Commas are generally used to list events while, like standard prose English, periods and semicolons denote more significant topical shifts. Symptoms are often listed following the verbs "states," "reports," and "presents." Few other verbs are used and full phrases (those that end in a period or semicolon) are rarely well-formed complete sentences. Prepositional phrases that include temporal information usually directly follow the relevant incident or symptom (e.g., *cough and cold for last 1 1/2 weeks*). Although some incidents and symptoms reported in TNs are described by such explicit temporal information, many are not (e.g., *No fever.*). Within the TN genre of text, incidents and symptoms not described with explicit temporal information as well as symptoms that began in the past (e.g., *headaches started yesterday*) can be assumed to be true at the time the TN is recorded. The example triage records in Figure 1 illustrate these properties.

Clinicians rely upon the information in the TN as they begin to treat patients. It is important that they receive accurate reports of the patient's status. Additionally, the TN helps clinicians focus their attention and begin effective treatment as quickly as possible. The TN is saved as part of a patient's record and is sometimes referred to in the discharge summary, which is composed when the patient leaves the hospital.

B. Temporal Information

In this research 'temporal information' includes any reference to when an event occurred, either on a standard calendar or timeline (e.g., *8:00 a.m. on 11/20/06*) or relative to other events (e.g., *She went for a walk after dinner*). Information about when an event occurred on a standard timeline may be very specific (e.g., *8:00 a.m. on 11/20/06*) or may be only a broad indication of a temporal zone on the timeline. For example, *she'll feel better after tomorrow* indicates a temporal zone that spans the infinite future beginning with when *tomorrow* ends. Temporal information may be expressed explicitly (e.g., *on Saturday*, *last night*, or *June 25*), but it is also often indicated by verbal morphology, including tense (e.g., present tense *she runs* vs. past tense *she ran*) and aspect markers (e.g., imperfective *she was running* vs. perfective *she ran*). Additionally, temporal references may be absolute (*June 25*) or relative to the time that they are spoken or written (e.g., *last night*). Temporal information provides a way to order events chronologically. Sometimes domain knowledge also provides a way to order events chronologically. For example, in the statement "She made a sandwich and ate it," domain knowledge (in this case, world knowledge) allows us to order the *made a sandwich* event before the *ate it* event. Both

temporal information and domain information sometimes yield event ordering interpretations, but they are distinct types of information. Some domain assumptions do provide a general framework for interpretation of TEs in this work, however, the focus is on temporal information.

C. Temporal Formalisms

Allen (1984) and Allen and Hayes (1989) develop a theoretical approach to formally representing temporal logic about events and actions. In particular, Allen and Hayes (1989) explore the distinction between points and intervals in time. They propose that intervals of time are decomposable into subperiods, while points (of zero duration) and moments (of nonzero duration) in time are not. Hobbs and Pan (2004) describe instances, like Allen and Hayes' points, as having equal start and end times. Like Allen and Hayes, Hobbs and Pan define intervals as having a start time that is unequal and prior to its end time and, thus, a nonzero duration. The system for interpreting temporal expressions in TN text, described below, is based upon such notions. Because of their inherent characteristics related to durational assumptions, we have chosen to treat some events (e.g., *MVC* [motor vehicle crash] and *fell down stairs*) as instances (points) and some (e.g. *cough and cold* and *back pain*) as intervals.

D. TimeML

Pustejovsky et al. (2003) describe the TimeML specification language in detail. Developed several years ago, TimeML extends Timex2, which is a markup system for indicating the temporal relationships described in natural language. Timex2 had provided

a way to represent relationships between events, and TimeML extends that system to allow for separately representing events themselves, as well as the relationships between them. TimeML allows for temporally anchoring events, ordering events, and including relative temporal information. Appendix A shows part of a TN and its TimeML representation. My present work aims to extract and interpret temporal information and produce output that is both human and machine readable. Extending the current system's machine readable output to be compliant with the TimeML standard may be done easily. The TimeML representation is much more detailed than what I currently need, however, so we use a simpler form.

E. Information Extraction

Researchers have been attempting information extraction (IE) tasks for several decades, and applications in the biomedical domain have been a part of this work since it began (Cowie & Lehhert, 1996). Attempts to extract temporal information, specifically, from text have been encouraged by the Automatic Content Extraction (ACE) program (Ahn, 2006), and, like general IE, much of that work has been within the domain of biomedical texts (Adlassnig, Combi, Das, Keravnou, & Pozzi, 2006).

Cowie and Lehnert (1996) provide an excellent overview of the motivations, developments, and significance of information extraction research. There is a huge amount of text in electronic form in the world today. IE is one area of research that explores using computers to deal with large amounts of textual information. Essentially, IE systems transform, digest, filter, and analyze text and output small, meaningful pieces

of information for humans to ponder, or structured information (e.g., in a database or XML format) that can serve as input to additional applications. Cowie and Lehnert explain why applications of IE are exciting and meaningful and why NLP research may want to pay more attention to its possibilities. Their examples include an IE application that can help financial analysts monitor industrial production of semiconductor devices, as it is documented in newspapers and journal articles. They also discuss the example situation of an IE system tracking U.S. forestry companies' profits compared with European forestry companies' profits. Importantly, they note that the tasks and gold standards in IE are often well-defined. In contrast to NLP applications like summarization and machine translation, IE system evaluation is relatively objective and straightforward, which makes it an attractive area of research for many.

Additionally, Cowie and Lehnert (1996) describe several techniques that tend to be productive in IE. These include partial parsing (chunking) in place of full syntactic parsing and the use of shallow knowledge bases in place of detailed knowledge engineering. In fact, a lot of IE research points to the success that can often be achieved using a relatively simple methodology. For example, Kraus, Blake, & West (in press) have managed to extract drug dosages from physician notes using a single heuristic while achieving precision and recall results of 96.7% and 79.7%, respectively.

F. Temporal Information Extraction

Identifying temporal information has been singled out as an important task by many researchers (Ahn, 2006; Fissaha & de Rijke, 2005; Verhagen et al., 2005). It is often

coupled with event detection and ordering, which is frequently the end goal of a given project. Johansson et al. (2005) and Verhagen et al. (2005) are two good examples of systems that have been built to annotate text for temporal information. Both use a combination of machine learned and hand-written rule sets.

Johansson et al. (2005) work with a corpus of Swedish newspaper texts that describe traffic accidents. They use a series of linguistically informed preprocessing techniques including part-of-speech (POS) tagging, shallow parsing (including noun phrase chunking and clause segmentation), and complex word recognition. The preprocessing output informs a further series of modules, including entity detection, coreference disambiguation, and, finally, event detection. Their approach to ordering events is a hybrid system that combines feature-based machine learned decision trees with hand-written rules. Although they do not describe their hand-written rules, their machine learning system makes use of the following features: temporal signals between phrases (e.g., prepositions like *before* and *during*), verbal morphology (including tense and aspect), and pairwise features that indicate the distance between events as they appear in the text, measured in counts of tokens.

Verhagen et al.'s system is also built for newspaper text. They combine several modules, the first of which (the GUTime tagger) annotates absolute, relative, and durational temporal expressions and modifiers. They define an absolute time (e.g., *October 10, 2004*) as having a meaning not dependent on the spoken or written context and a relative time (e.g., *Monday* or *last year*) as dependent on a reference time, typically the time a

document (i.e., the newspaper article) was published. The entire pipeline of modules outputs news text fully annotated for events, temporal expressions, and links between them. The GUTime tagger extends the work presented in Mani & Wilson (2000), which developed a rule-based IE system for detecting temporal information. The rules are both hand-written and machine-learned. In contrast to the Johansson et al. work on Swedish text, Verhagen et al. only use POS tagging and chunking preprocessing techniques on their English text. Although Verhagen et al. do not provide any details on the results of performance evaluations, Johansson et al. report a F-measure performance of .62 (for extracting temporal relations related to correctly detected events).

Also working within the domain of news texts, several research groups led by David Ahn have dealt with temporal information in two parts. They first identify temporal information (the detection task) and then order events relative to one another on a timeline (the normalization, or interpretation, task) (Ahn, 2006; Ahn, Fissaha, & de Rijke, 2005a; Ahn, Fissaha, & de Rijke, 2005b; Fissaha & de Rijke, 2005). For both tasks, the group tends to emphasize data-driven machine learning techniques and they assert that such methods, when used in combination with rule-based systems, can outperform approaches that are strictly rule-based (Ahn et al., 2005b). However they do make it clear that careful feature selection in any data-driven methodology is critical (Ahn, 2006; Fissaha & de Rijke, 2005).

In summary, temporal information extraction systems vary somewhat in the type of temporal information on which they focus. Generally, the systems extract bits of temporal

information to use as features in chronologically ordering events at a target level of granularity. Systems for ordering events based on extracted temporal features tend to combine data-driven, machine learned models and hand-written rules. Systems usually use a diverse set of features, and most research suggests that feature selection influences system performance more than most other system design choices.

G. Medical Information Extraction

As mentioned above, many useful IE applications have focused on the biomedical field. Many of those applications have worked to identify important information in clinical texts. Suominen et al. (2007) systematically explore the risks and benefits of applying natural language processing (NLP) techniques to electronic patient records created by nurses and conclude that such work has a great potential to improve patient care. In particular, the group considers how narratives written by nurses in intensive care units could inform NLP-based clinical decision support systems, especially alert systems. They conclude that compromising patient privacy is the main risk involved in such tasks, but the group predicts a net improvement in the medical service available to patients.

At New York University, Sager was doing IE work on hospital patient discharge summaries even before 1970 (Cowie & Lehnert, 1996). More recent research on extracting temporal information from medical text has also focused discharge summaries (Bramsen, Deshpande, Lee & Barzily, 2006; Hripcsak et al., 2005; Zhou & Hripcsak, 2007). Although researchers have written about the potential that lies in applying temporal information extraction methodologies to important medical texts, news text has

received more practical attention and more applications have been built to process the edited, narrative style of English. Zhou & Hripcsak (2007) discuss several of the particular challenges involved in dealing with medical text, including the following: “diverse temporal expressions,” “medical grammar,” “complex temporal relations in the medical domain,” “implicit information,” varying degrees of temporal granularity within a single text, "temporal fuzziness and uncertainty," the “semi-interval problem” (having information on the start or end time of an event but not both), and the presence of temporally contradictory information about events (p. 194-195). Zhou’s research group, which has worked with discharge summaries, has developed an annotation scheme and formal system architecture, but none of it has been implemented yet. (Zhou & Hripcsak, 2007; Zhou, Melton, Parsons, & Hripcsak, 2006; Hripcsak et al., 2005). Their annotation scheme divides temporal expressions in clinical texts into the following major categories: Date and time, Relative date and time, Duration, Key events, Other events, Fuzzy time, and Recurring time. Using evidence from a corpus of hospital discharge summaries, they argue that their classification scheme is natural and placing temporal expressions within it would assist in interpretation and ordering tasks (Zhou, Melton, Parsons, & Hripcsak, 2006). The classification scheme that my work uses is based upon this system.

Two systems that have been developed to extract temporal information from medical text are described in Gaizauskas, Harkema, Hepple & Setzer (2006) and Bramsen et al. (2006). Gaizauskas et al. (2006) have created a baseline system that extracts temporal relations from a corpus of clinical narratives related to procedures such as x-rays, ultrasounds, etc. and. They emphasize that targeting relevant temporal information for

extraction is a reasonable and practical strategy, asserting that exhaustive temporal information does not need to be detected within their domain. They use a combination of the GUTime tagger (also used in Verhagen et al., 2005), a gazetteer of important clinical investigation event terms, and a large amount of manual annotation to create system input that is perfectly tagged with relevant time and event annotations. Their system is a pipeline of modules that uses a parser, a discourse interpreter, and an annotation writer to link event tags to their proper temporal tags. This processing architecture is similar to the overall TN project architecture (shown in Figure 5 and discussed below). With perfect manually generated event and temporal annotation input, their system achieves 73.83% precision and 58.70% recall in identifying the links between the tags. The significant manual preprocessing work that their system requires illustrates the overt need for a temporal information tagger that is as successful within the medical domain as the GUTime tagger is in the news text domain. The work that I present here describes the development of such a tool.

Bramsen et al. (2006) describe an attempt at actually detecting temporal information in discharge summaries. They use a corpus-based, supervised machine learning technique that classifies segmented phrases as either containing a 'temporal shift' or not. They define temporal shifts as "abrupt changes in temporal focus" (p. 81) such as the shift from a discussion of a patient's condition in triage to a discussion of his medical history. Their notion of a temporal shift is quite granular, but they suggest that their approach is useful in situations of sparse temporal information and that their output may be used as input into system that can produce finer-grained temporal interpretations. Their preprocessing

techniques separate the text into phrases and then unigrams, bigrams, and trigrams of words as well as corresponding POS tags are chosen as features extracted from the beginning and the end of each phrase (at the boundaries). They also use topical information in some features with the thought that shifts in topic may indicate shifts in temporal references. Their system achieves an 83% F-measure for identifying temporal shifts.

At the University of Pittsburgh, Chapman and others are also working to extract key information from medical discharge summaries. They have used the CC field to predict the ICD-9 patient diagnosis as reported in the discharge summary (Chapman, Dowling, & Wagner, 2005) and to classify patients into syndromic categories for biosurveillance purposes (Chapman & Dowling, 2007). Importantly, Chapman's group has also developed a system, ConText, for determining the scope of words or phrases that qualify items reported in the TN (Chapman, Chu, & Dowling, 2007). They refer to such qualifying words or phrases as contextual features, which include negatives and indications of a historical or hypothetical frame of reference. That is, the scope of some references, including some temporal information, may span more than one event. For example, in *history of bronchitis and sinus congestion,* the scope of 'history' includes both events. It will be important to resolve the scope of such words or phrases in ordering all TN events on a timeline. The system architecture of the current project (shown in Figure 5, and discussed below) incorporates the need for a tool similar to ConText.

In summary, systems for extracting event and temporal information from medical texts are less advanced than systems for processing standard, edited English. Zhou, et al. (2006) present a carefully developed annotation scheme, but they have not implemented a system that uses it. Gaizauskas et al. (2006) describe a system heavily dependent on manual annotations, and Bramsen et al. (2006) describe a system targeted at a very coarse grained level of temporal information extraction. Chapman, Chu, & Dowling (2007) present ConText, a tool for identifying the scope of a variety of frames of reference, including temporal references, which could be useful in improving the performance of end-to-end extraction systems.

H. Summary of Background

The task of automatically ordering events described in natural language is important, and a substantial portion of the work required to do so consists of extracting and interpreting temporal information. Temporal interpretation involves normalizing natural language references to time with respect to an absolute timeline. Although temporal information is critical in such projects, its expression and meaning are often complex. Using temporal formalisms may assist in representing and interpreting temporal information. Currently, most temporal information extraction systems are written for standard edited English, and they tend to use a variety of textual features in hybrid systems that combine hand-written and data-driven, machine-learned rules.

III. Research Motivation and Hypothesis

The need for a system to automatically produce a structured representation of the timing of events reported in ED TNs has motivated this work. The major research question asks whether or not typical information extraction techniques are appropriate for approaching the unique genre of TN text. Many IE systems have been successful, but it is not yet clear how their methods will apply to text in patient records, including TNs. This work focuses on developing a system for extracting temporal information in the domain of TN text. The research methods combine existing techniques with domain knowledge of the genre of TN text. I hypothesize that a system employing a combination of shallow natural language processing, a supervised machine learning algorithm, and hand-written processing rules informed by domain-specific knowledge can achieve an operational degree of accuracy in extracting, classifying, and interpreting temporal expressions in this narrow domain of text.

IV. Data

A. Sample

The North Carolina Disease Event Tracking and Epidemiologic Collection Tool (NC-DETECT: http://www.ncdetect.org) is a public health surveillance tool that monitors data from North Carolina EDs, the Carolinas Poison Center, and the Pre-hospital Medical Information System. The system is maintained at the University of North Carolina at Chapel Hill and is used for a wide variety of public health objectives. With permission

from the North Carolina Division of Public Health, NC-DETECT provided 598 patient records of visits to North Carolina EDs to our research team, which is composed of one faculty member and two graduate students along with two domain expert advisors. My current work is based upon those records. At the time the sample was collected, 94 (84% of the 112 North Carolina EDs) hospitals provided patient records to NC-DETECT daily. The sample of records (henceforth the 598-sample) was chosen randomly from records contributed by hospitals that include TN in their data feed during a two day period in November of 2006. Each de-identified patient record includes a timestamp, a CC, and a TN. This research was approved by the University of North Carolina at Chapel Hill Institutional Review Board. Figure 1 shows two example TNs. The average length of a TN in the 598-sample was 31 words.

B. Gold Standard

Temporal Expressions (TEs)

The research team and I developed a gold standard annotation scheme for identifying and tagging temporal expressions (TEs) in TN text. Before developing a classification scheme, we determined that we wanted individual TEs to be phrase-like and include both explicit temporal references (e.g., *yesterday*, *since last night*, or *hx of [history of]*) and the event that is syntactically the most closely related to it (e.g., *fell yesterday*, *cough since last night*, or *hx of seizures*). We have defined events as a class of expressions that includes both symptoms (e.g., *cough*, *back pain*) and incidents (e.g., *mva* [motor vehicle accident], *fell*). Within the TE, we decided to include both explicit temporal references and the event in order to maintain the direct association between the two. Explicit

temporal references frequently describe more than one incident or symptom (e.g., *sob* [short of breath] *and tearful this morning*). In our system architecture (see below), the initial extraction step identifies the TE and its immediately contiguous event (*tearful this morning*). Subsequent steps will expand the scope of the TE over lists and conjunctions to link other events (*sob…this morning*).

We developed a detailed standard for identifying TE boundaries. In addition to the two primary elements included in the TE (the explicit temporal reference and the syntactically most closely related event), we have included details (noun or verb phrase modifiers) about the temporal reference and event within the TE as they are reported in the TN. We felt that presenting too much detail about an event on the timeline was less risky than presenting too little detail. Additionally, the system architecture will allow us to prune the initially coarse-grained TE later, before it is presented on the timeline. For example, we currently tag *tripped over football today at school* as a single TE. The prepositional phrases *over football* and *at school* are included in the current gold standard. We may maintain this information in the final timeline, but later parsing and pruning may also allow us to only present the most relevant information, *tripped today* or *tripped over football today*. In Figure 2 the TEs in two example TNs are underlined.

2006-11-22 00:54:00 COUGH Pt Has had cough and cold for last 1 1/2 weeks. Seen here last night for same. Pt coughing and vomiting, can't sleep. Pt has clear bil BS. No fever	2007-04-17 15:48:00 WRECKED MOTORCYCLE MCA approx 30 minutes ago-bike landed on right leg. Now with painful ambulation-increased pain right knee

Figure 2: Example Triage Notes, with TEs underlined

TE Classification

After defining the TE and describing appropriate boundaries, we developed a way to classify the TEs for the purpose of interpretation. Our conceptual classification scheme is based upon the work done by Zhou et al. (2006) on discharge summaries. Zhou's group identified seven mutually exclusive major categories and 24 subcategories for classifying TEs. In considering the domain of our text and moving towards interpretation, we have renamed one category and modified some of the subcategories. Additionally, unlike Zhou's group, we have used non-exclusive classes. That is, we have double and triple annotated some TEs according to the type(s) of temporal information that they contain. Intuitively, some TEs belong to more than one class. For example, *cough since yesterday* indicates a duration as well as a relative date reference, and *fell down ladder yesterday 3 pm* indicates a time of day as well as a relative date reference. Using non-exclusive classes also assisted us in developing a method for interpreting the TEs (see below). Table 1 shows the temporal classes and subclasses and the distribution of tags among the 598-sample. Table 2 shows an example TE for each of the seven major classes.

Major Class: Subclass	Tags	Percent of Tags
Date and time:	**151**	**15%**
Date	10	
Named day or month	40	
Others	2	
Part of day	5	
Time of day	82	
Timestamp	12	
Duration:	**207**	**20%**
Others	8	
Event duration	131	
Symptom since	68	
Fuzzy Time:	**77**	**7%**
Future	2	
Non-specific time	1	
Past	71	
Present	2	
This time/that time	1	
Key Event:	**202**	**19%**
Arrival	27	
Current visit	115	
Operation	14	
Other key events	6	
Previous visit	40	
Other events:	**28**	**3%**
NOS	28	
Relative date and time:	**362**	**35%**
(A period of time) Ago/later	90	
In/within (a period of time)	9	
Past/next (time unit)	6	
Yesterday, today, tomorrow	257	
Recurring time:	**14**	**1%**
NOS	14	
Total:	**1041**	**100%**

Table 1: Frequency of TE class and subclass tags in 598-sample

Major TE Class	**Example TE**
Date and Time	*Discharged from hospital on 11-17-06 for same*
Duration	*fever of 103 for 3 days*
Fuzzy Time	*history of constipation*
Key Event	*Pain level now: 8/10*
Other Event	*took two Vicodin after accident*
Relative Date and Time	*involved in an MVC 1 week ago*
Recurring Time	*dizzy one day every month*

Table 2: Example TEs, by class

We manually annotated the TEs in the 598-sample and performed several iterations of modifications to the gold standard classes and annotation rules. In the final round of annotation, two coders found a total of 1041 TE tags. Both coders tagged 945 (90.8%) of the final set of tags, and the coders agreed upon 100% of those TE tags that they both annotated. One-hundred percent of the tag discrepancies were for TEs with multiple tags. The discrepancies were a result of simple human errors in applying the categorization rules rather than disputed classifications, and the coders resolved them in consultation. We found an average of 1.5 TEs and 1.74 TE tags per TN. In the 598-sample, 13.9% of TNs have more than one tag, yielding an average of 1.16 tags per TE. Figure 3 shows the distribution of TEs over the 598-sample. Most (64%) TNs have one or two TEs; 21% of the TNs in the 598-sample have no TEs.

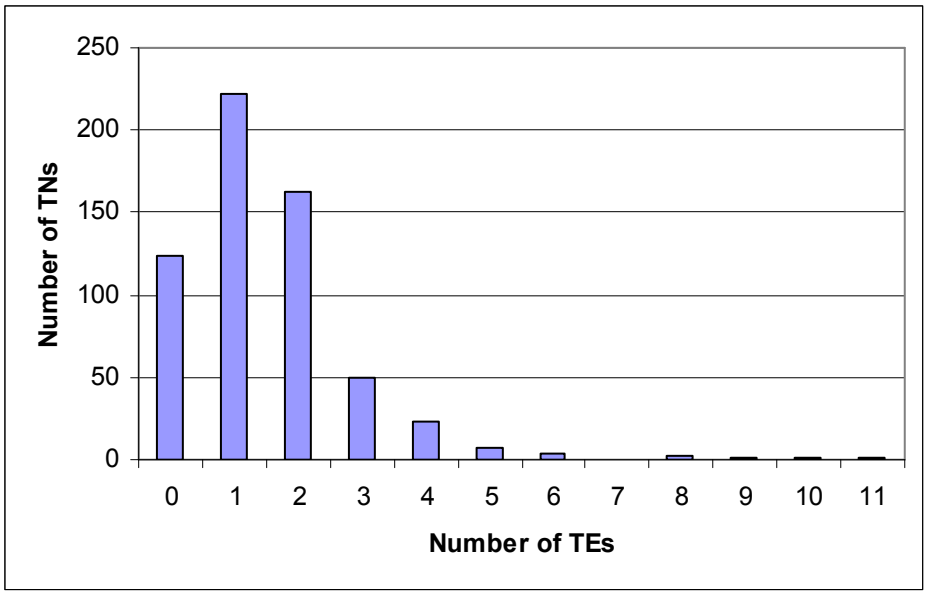

Figure 3: Distribution of TEs over TNs in 598-sample

TE Interpretation

As part of our goal of generating a complete timeline of events leading up to a patient's visit to the ED based on text in the TN, we have mapped individual TEs to a timeline. We carefully read and manually interpreted many of the TEs in the 598-sample and then derived more formal interpretation rules. Some TEs are quite easy to interpret. For example, absolute, or explicit temporal references (Han, Gates, & Levin, 2006), such as a date (e.g., *9/20/2006*), are easily interpreted unambiguously. Other TEs, particularly those that include relative and deictic temporal references (e.g., *last week*, *yesterday*), are more challenging because their resolution depends on the time at which they were uttered. TEs that include time of day references (e.g., *this morning*) are particularly challenging because a relatively precise zone consisting of a period of just a few hours must be resolved in addition to the reference date.

We decided to address interpretation of the RDT class first, in part because RDT is the most common TE tag in the 598-sample. Additionally, for reasons discussed in the Methods section below, the system is likely to interpret RDT references before other classes. Initially, we used our own intuitions to formally represent the meanings of many TEs. For example, we estimated that the phrase "last night" when spoken at 3:00 pm refers to the period between 5:00 pm on the previous day and 5:00 am on the current day. Some TEs are easier for humans to interpret than others. For example, the phrase "on Friday," uttered on a Sunday evening likely refers to the day two days previously. However, preceding text may have created a non-recent past (what Zhou et al. call "narrative reference" and Han, Gates & Lavie call a “temporal focus”) frame of reference ("Vacation last summer was great. We got there on Wednesday and then on Friday…").TEs spoken in the triage situation are often easier to interpret than in typical narrative prose because the current time and very recent past are especially salient. Returning to the phrase "on Friday," we expect that the chance that the preceding text in the TN contains information creating a non-recent past frame of reference is much smaller than in standard narrative prose. This relatively strong constraint is based on our analysis of the content of TNs, as well as our understanding of the common circumstances in which patients come to the ED. Most events recorded in the TN occurred in the recent past.

In addition to exploring our own intuitions of temporal references and careful reading of many TNs, we also gathered a quantitative description of the distribution of the hourly

usage of many key explicit temporal reference words and phrases. For example, we all have general intuitions about the period of time referred to by the expression “yesterday.” The concept generally indicates about a 24-hour interval that started on the date previous to the current one sometime in the middle of the night or around the time that we woke in the morning and lasted until the middle of the following night or around the time we went to sleep the previous night. We used the relative usage patterns in the 598-sample to determine more specific temporal boundaries for such concepts. We observed that the common temporal reference word “yesterday” dips in usage at around 6 a.m. The word is used with much more frequency between 9 a.m and 6 p.m. Thus, we were able to determine that the boundaries for the temporal zone should be 6 a.m. of the previous date and lasting until 6 a.m. of the current date. Many interpretations are sensitive to the time at which they are used. For example, this “yesterday” interpretation works if the term is used between 6 a.m. and midnight of a given day. However, if it is used between midnight and 6 a.m. it likely refers to a temporal zone that began at 6 a.m. on the date two before the current one. Figure 4 illustrates this distinction. Returning to the phrase "last night," we concluded that the interpretation beginning and end points should correspond to 6:00 p.m. and 6:00 a.m., respectively. Similar observations and analyses were used to determine the most likely interpretations for other RDT TEs.

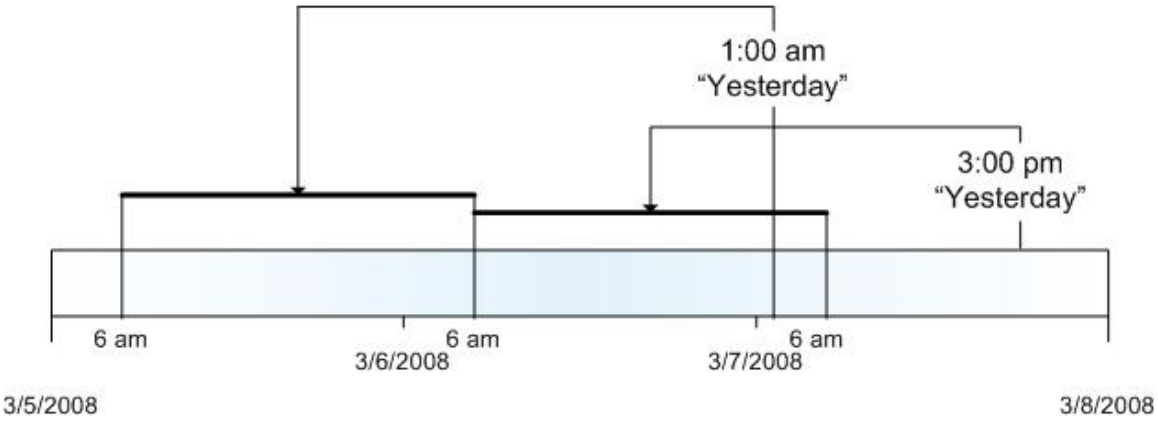

Figure 4: Illustration of "yesterday" interpretation guideline

We have created general descriptive guidelines for how to interpret many TEs classified as Relative Date and Time (RDT) expressions. In addition, we manually interpreted many RDT TEs in the 598-sample to use as a gold standard interpretation. Although we have created specific guidelines for interpretations, and especially the related conceptual temporal boundaries, it would be easy to alter the guidelines in light of additional information from domain experts or empirical evidence.

C. Data Summary

In summary, the three primary members of our research team (one faculty member and two graduate students) have worked with 598 sample TN records to create a gold standard for identifying and classifying all TEs and interpreting some classes of TEs. The 598-sample includes 895 TEs and 1039 TE tags. In general, the work that follows focuses on creating a way to process TN text automatically and output information that matches the gold standard. As the work continues in the future, details of the gold standard,

especially the specifics of the interpretation output, may be easily revised if needed to fit the intuitions of experts in other disciplines who could use such temporal information.

V. Methods

Figure 5 presents the complete system architecture for processing TNs and generating a timeline of events leading up to a patient's visit to the ED. The research described here focuses on the section of the system highlighted with a dotted line. The Triage Note Temporal Information Extraction System, or TN-TIES, is a system that, given a raw triage note, identifies, extracts, classifies, and interprets the temporal expressions within it. Throughout the system development process, I used 80% of the 598-sample (712 TEs) as training data and 20% (183 TEs) as testing data.

A. Chunker

I used the Python computing language as well as some of the modules included in the Natural Language Toolkit (NLTK: http://nltk.sourceforge.net/) to do a shallow parse (chunking) of the triage notes. The TN-TIES chunking algorithm that I developed is linguistically very shallow and based on observations of TN characteristics. The regularity of TN text makes such a rule-based approach appropriate. I created the chunker by iteratively adding and revising rules and by examining a small part of the training data closely. My goal was to output phrase-like lines of text, specifically chunks that correspond to the gold standard TE boundaries. The chunker splits the text of the triage notes on some punctuation markers, the conjunction word "and", and some

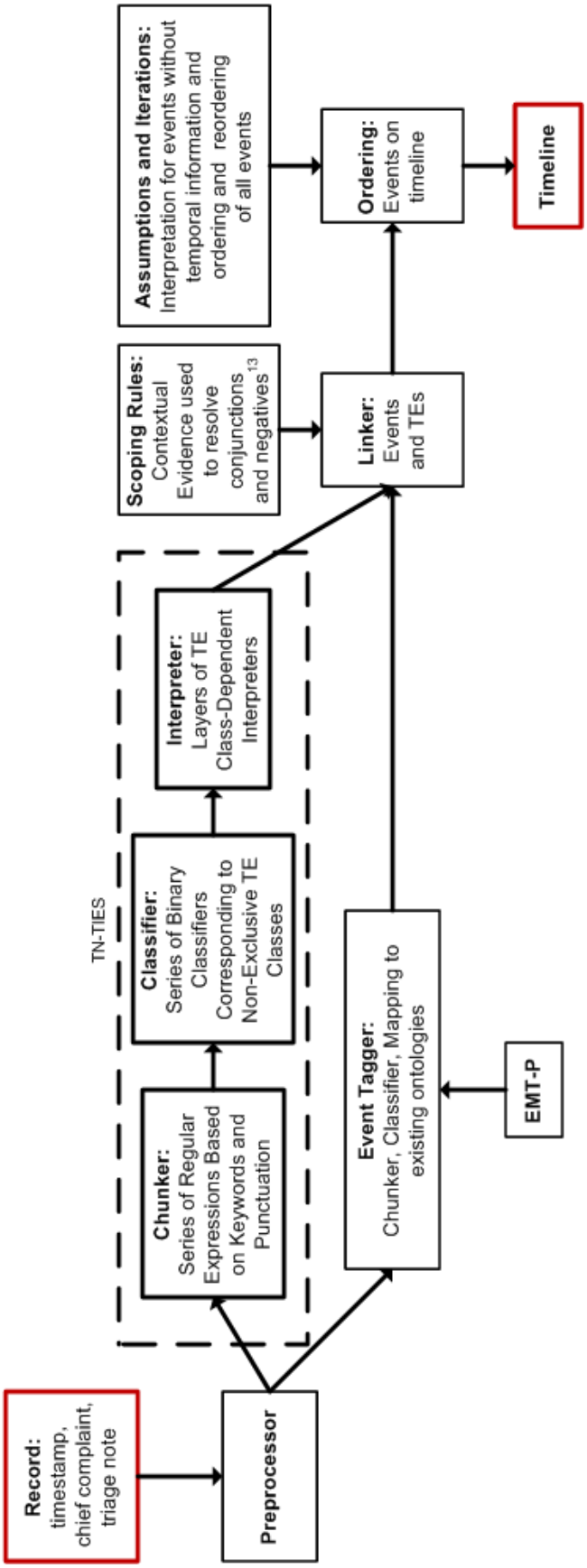

Figure 5: System architecture, highlighting TN-TIES

communication verbs commonly found in the triage notes, including the following words and their morphological variants: "presents", "says", and "reports." In developing the system, I tried including some rules to chunk the text based on part-of-speech (POS) tags assigned by the NLTK bigram tagger and trained on the Brown Corpus. For example, I tested rules that split the text after all verbs, rather than the list of communication verbs. However, I observed a decrease in chunker performance with such POS tag-based rules, so the final version does not include them. Figure 6 displays examples of the TN-TIES chunker output for two TNs. The gold standard TEs are underlined.

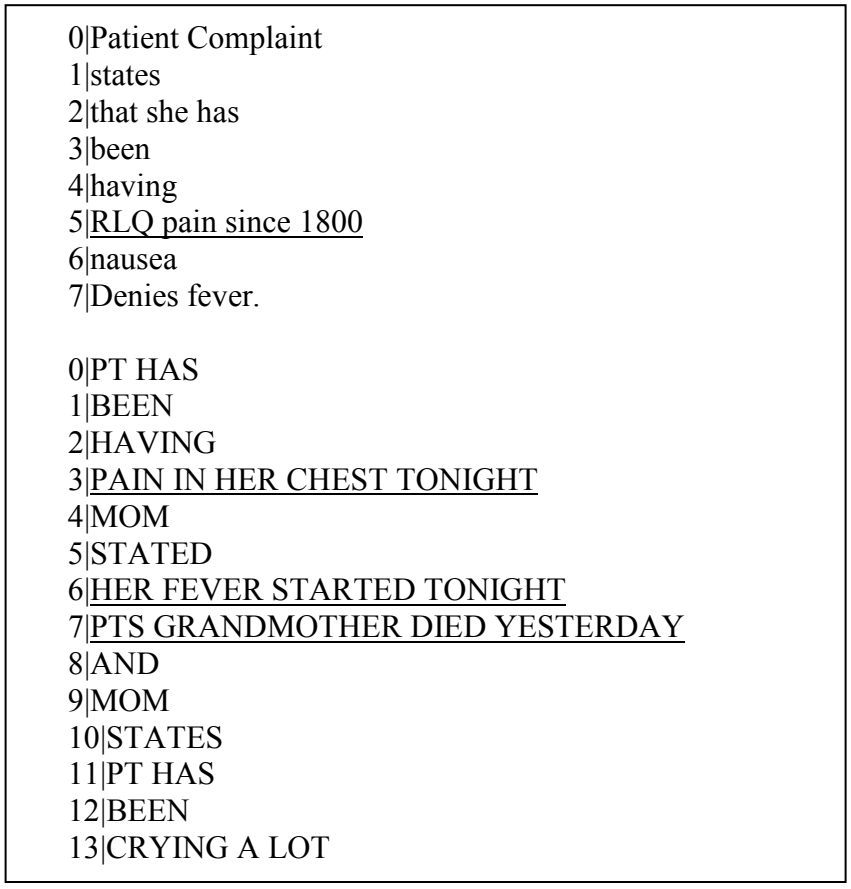

Figure 6: Example TN-TIES Chunker output, with gold standard TEs underlined

B. Classifier

I used the Oracle SQL Developer and Data Miner (http://www.oracle.com/technology/products/bi/odm/odminer.html) to develop and test a variety of algorithms for classifying the TN chunks. I tokenized the 598-sample TN text on whitespace and uploaded the terms into a database. I also developed gold standard binary classification lists for each of the five major TE classes (that is, a list of all chunks and an indication of whether or not each belonged to a given class). Not enough training data was available to develop classifiers for the infrequent Other Event and the Recurring Time classes.

I used a combination of manual and automatic methods for compiling a feature list to use in training the five binary classifiers. I first explored the relative term frequencies of words appearing in individual TE classes (e.g., the Duration class) compared with the entire document collection. However, I found a very large amount of overlap among the five TE classes. So, I used a single feature list to train all five of the binary TE classifiers.

Table 3 displays the 48 terms in this initial feature list. Nearly all of the features are rooted in a single term that appeared very frequently in the target classes and less frequently in the non-target classes. I should note that I chose not to employ tf-idf weights to choose features because idf calculations added almost no new information to simple term frequency counts. The average length of the chunks was 2.86 words and, thus, terms very rarely appeared more than once in a single chunk; frequency counts for

terms were very close to their document frequency rates. Term frequency counts alone, however, did prove to be quite informative.

(+)	onset
0/10	pain
about	past
after	pm
ago	present
am	prior
approx	pta
around	regex: 4 digits
arrival	same
at	seen
began	since
days|day	started
earlier	this
few	time
here	today
hours|hour	tonight
hx|history	triage
last	week|weeks|wk|wks
level	when
minutes|minute|min	worse
months|month	years|year
morning|mornings	yesterday|yest
night|nights	monday|mon||tuesday|tues|wednesday|wed| thursday|thurs|thur|thu|friday|fri|saturday| sat|sunday|sun
now	
occurred	

Table 3: Initial feature list

Using simple term frequency counts, I manually extracted those terms that appeared very frequently in the target classes and not frequently in the non-target class, indicating that they provided consistent positive evidence for the target classes. I did not include words that provided negative evidence for the target temporal classes because, simply, it appeared that such terms did not exist. That is, there were no high frequency words in the non-target class that were not also high frequency words in the target classes. Because the

gold standard TEs include both explicit time references and closely associated events, nearly any word has the potential to appear in a TE. Almost everything in a triage note can be classified as an event (incident or symptom), and, thus, almost all words that appear in triage notes have the potential to be included in a TE. So, in summary, I chose to hand-pick only those features that appeared to provide strong positive evidence for the target temporal classes.

Having extracted those words that appeared to provide strong positive evidence for chunks in the target classes, I manually expanded each feature. For example, the word ‘yesterday’ appeared very frequently among chunks in the RDT and Duration target classes and, in this case, never among non-TE chunks. With my own domain expertise based on analysis of the records, I knew that nurses often use the abbreviation “yest” for “yesterday.” So, I expanded this feature to include occurrences of either “yesterday” or “yest.” Essentially, I created a "yesterday" class, which contains features that correspond to the strings or expressions used by triage nurses to indicate the concept. I created several similar classes, including a “week” class, which has the members “weeks”, “wk”, and “wks.” This approach has several advantages. First, it allows me to add additional expressions to classes if found (e.g., local or regional variants). Second, in the future the system may link to a thesaurus or ontology to expand the classifier to other types of documents that use different expressions for the same concepts.

In addition to combining morphological variants of expressions into single conceptual classes for use as individual features, I also combined groups of lexical items according

to their semantics and distributions of use. For example, the word “monday” was used very frequently among the triage notes in my training data. I expanded this feature to include not only “mon”, but also other days of the week and their abbreviations. The names of different days of the week are equivalent in their distributions of use, context, and type of information conveyed. I anticipated that such feature combinations would increase the classifier's performance. I also included a feature for any term consisting of exactly 4 digits (e.g., *1400*). Again, in inspecting the training data, it is clear that nurses nearly always intend to indicate time of the day when using this format. Four digits rarely refer to non-temporal concepts.

I used the Oracle Data Miner software to build 10 binary classifiers. All of the classifiers are trained on the 80% training dataset, the gold standard TE annotations, and the 48 features listed in Table 3. I built a Decision Tree and Naïve Bayes classifier for each of the five most prevalent temporal classes. Additionally, I explored the use of a Support Vector Machine classifier, but its performance was extremely poor because of the small number of gold standard training examples for each TE class.

I made several iterations of observing the classification results and revising the feature set. I paid particular attention to the classifiers that identified RDT TEs, because these would be used in the automatic interpretation phase. I observed incremental increases in performance, especially as I expanded the coverage of morphological variants of individual features. As mentioned, this list of features is included in Table 3. After developing the classifiers and analyzing their performance, I implemented those

classifiers that had reached an operational degree of accuracy (the RDT and Duration classifiers) and were ready for inclusion in the system. I used the Python computing language to create a seamless series of processing modules. After implementing the classifiers, I performed several more iterations of feature engineering.

C. Interpreter

The TN-TIES interpreter is a layered processing module that interprets each TE classification tag individually. That is, for the TEs that have multiple tags, our basic strategy is to interpret the TE starting with the largest granularity first to establish the "temporal zone", (e.g., interpret "last Monday" before interpreting "3 p.m."), and then narrow the estimated relevant point or time interval. Two examples of our systematic, layered approach to interpretation are shown in Figure 7.

(a) *seizure 3pm yesterday*
(1) chunk classified both as a RDT and as a Date and Time (DT) TE
(2) *yesterday*, as an RDT trigger word, is interpreted relative to the ED visit timestamp and has an interval of roughly 24 hours
(3) *3pm*, as a DT trigger word, is interpreted within the bounds of the zone found in (2) – the 3pm within the 'yesterday' zone

(b) *migraine since Friday*
(1) chunk classified as both a DT and a Duration TE
(2) *Friday*, as a DT trigger word, is interpreted as the most recent Friday relative to the timestamp
(3) *since* triggers the duration quality, indicating that the migraine began on the Friday identified and has continued to the time indicated by the timestamp

Figure 7: Examples of layered, systematic approach to interpretation

We developed this general framework for interpretation as well as a description for interpreting TEs within major classes and those containing specific temporal concepts.

Using the framework and descriptions, we formalized our representations of interpretation. A start point and an end point denote the temporal zone during which an event occurred. The temporal zone is separate from the notion of how long an event lasted. A TE may explicitly indicate an event's duration and, separately, the temporal zone. For example, the temporal zone start and end points for the TE *vomiting for two days last week* represent the concept *last week*. Our framework for non-exclusive TE classification allows us to interpret the general temporal zone first and then, separately, represent the TE's duration in an additional feature (e.g., *for two days*). In some cases, durational information may also inform the temporal zone start and end points. For example, in the TE *vomiting since yesterday*, our model would determine the temporal zone of *yesterday* first, and then, when the durational quality indicated by *since* is interpreted in a separate layer of processing, the temporal zone end point would extend to the TN timestamp (the present time). In this case, the feature representing the TE's duration would indicate that the event lasted for entire temporal zone interval.

Characteristics of duration are often implicit in the type of event. For example, the event in the TE *headache yesterday* may be interpreted to have lasted all day. In contrast, the event in the phrase *seizure yesterday* likely only lasted for a few moments at some point during the day. As explained, we have chosen to interpret temporal zones and temporal durations separately. So, in developing the TE temporal zone interpretation rules, we did not take into account the type of event described in the TE. Returning to the example, the TE interpretation rules interpret *yesterday* systematically and without regard for the other information in the TE. We will be able to resolve such differences and add and revise

durational qualities as part of the process of tagging events, which is shown in the system architecture (Figure 5).

As mentioned in the description of the gold standard interpretation guidelines, we manually developed rules for resolving TEs in the RDT class before those in other classes. RDT tags were the most frequent among TEs in the 598-sample. Additionally, RDTs tend to indicate temporal zones of a higher granularity than those with which they co-occur. For example, Date and Time tags often co-occur with RDT tags and usually narrow the temporal zone in focus. For example, in. *headache since 3pm yesterday*; *3pm* narrows the focus of the temporal zone indicated by *yesterday*. It is likely that RDT references will be resolved in earlier interpretation layers than other tags, so we have built those guidelines first, and I have implemented them in TN-TIES. Given a piece of text that forms a single chunk within a TN that is identified as a RDT TE, TN-TIES outputs the temporal zone during which the event occurred. The output is relative to the visit timestamp and consists of two endpoints. The output is both human and machine readable. As mentioned above, the output is not currently compliant with the TimeML standards, but if such a need arises in the future, it will be easy to achieve.

Although TN-TIES currently only implements the RDT interpretations, I have developed its structure carefully so that it will be easily scalable to interpret the other classes and combinations of classes. TEs in the Date and Time class having a single tag or also belonging to the RDT class will require no data structures beyond the existing two temporal endpoints for interpretation. TEs in the Duration class will require an additional

data value indicating the length of the interval during which an event occurred. In the case that a TE belongs to the Duration class in addition to a RDT or Date and Time tag, the two endpoints will be important values but must be supplemented with the data value indicating an event's duration. TEs in the Fuzzy Time and Key Event classes typically have a less fine-grained interpretation than those in the RDT, Date and Time, and Duration classes and interpreting them will be simply a matter of recognizing subclasses. For example, interpreting *hx of seizures* will only require that the system identifies the TE as in the Fuzzy Time Past subclass. Similarly, interpreting *pain now 8/10* will only require that the system identifies the TE as in the Key Event Current Visit subclass. In addition to Duration, Recurring Time may be the only class that requires a data structure for interpretation that is not already implemented in TN-TIES. However, because the existing training data has so few examples of such TEs, there is currently no plan for how the system will interpret or present such TEs.

In summary, the TN-TIES interpreter currently handles RDT TEs, which are the most frequent. The data structure developed to handle RDT interpretations will be used to represent some additional classes and can be used as a starting point for interpreting other classes and combinations of classes.

VI. Results

The parts of TN-TIES that I have fully implemented performed well. The chunker splits approximately 91% of TEs appropriately. The RDT and the Duration classifiers both

achieve greater than 90% precision and recall. Finally, the interpreter outputs accurate human and machine readable interpretations of many of the RDT TEs. Results for each component of TN-TIES are described below.

A. Chunker

Table 4 shows the chunker's performance in terms of its accuracy in providing output chunks that correspond to the gold standard TEs in the testing set. Good output chunks are those that include both the explicit time reference and the event syntactically associated with it. Some of the good chunks include extra words and information (particularly prepositional phrases attached to the noun or verb phrase in the TE). However, as mentioned above, the overall project architecture (Figure 5) includes pruning steps that will make such superfluous information easy to deal with later. I developed the chunker to err on the side of including too much information in single chunks and tried to minimize 'bad splits,' like the 8 referenced in the second line of the table. The chunker split 4.5% of the gold standard temporal chunks in the testing data into two pieces. Additionally, 8 gold standard temporal chunks were included in 4 "doubled" output chunks.

	Number (gold standard TE chunks)	Percent (gold standard TE chunks)
Good chunks	162	91%
Time reference and event split in two chunks	8	4.5%
Doubled: 2 temporal chunks included in a single output chunk	8	4.5%
Total chunks in testing set	178	

Table 4: Chunking accuracy on testing dataset

Adding rules based on POS tags did not improve the chunker's performance. Part of the reason for the lack of increased performance may be the somewhat high rate or error in applying a standard POS-tagger, which is trained on standard narrative English prose, to TN text. Another reason is that the gold standard TE chunks are not consistently linguistic constituents of any type. That is, the TE chunks tend to be semantically rather than syntactically determined.

Overall, the TN-TIES chunker performs quite well, accurately identifying the boundaries of 91% of TE chunks in the testing set. However, this is by no means the overall system's upper limit in performance. Assuming well-implemented scoping rules in the system's later processing stages (see Figure 5), we can assume that the 'bad splits' referenced in Table 4 can be resolved. For example, an example of a bad split is the TE *headache on and off, since Tuesday* where *headache on and off* and *since Tuesday* are split into two chunks as a result of the comma in the middle of the TE. Scoping rules implemented later in the system should be able to increase the scope of *since Tuesday* to include the relevant event. In this case, this resolution will likely occur during processing stages designed to increase the scope of temporal information over lists of events, which are often delimited by commas. Additionally, although more challenging, the system may parse and individually identify the chunks that include two TEs in later processing stages. Such parsing may be prompted by an error checking phase that follows the TN-TIES interpreter.

B. Classifier

The results of the Decision Tree (DT) and Naïve Bayes (NB) classifiers after the first rounds of iterations of feature engineering are shown in Figures 8 and 9, respectively. These results correspond to the classifiers that employ the features listed in Table 3. Precision is defined as the number of chunks that the classifier correctly predicts are in the target class compared to the total number of chunks that the classifier predicts are in the target class. Recall is defined as the number of chunks that the classifier correctly predicts are in the target class compared to the total number of chunks in the target class, as defined by the gold standard. Overall accuracy is defined as the percentage of chunks (both in the target and non-target classes) that the system correctly classifies. There are many more chunks not in the target classes and the system is frequently able to unambiguously identify them as such, so the overall accuracy measure is quite high for all of the classifiers. As mentioned, the SVM classifiers' performances were quite low. Although the RDT SVM classifier achieved a 99% precision and 68% recall, the other SVM classifiers predicted that all chunks were in the target classes, thus maximizing recall while keeping precision values very low. This was likely due to the limited amount of training data available. In general, the DT classifiers outperformed the NB classifiers and the precision values were much higher than recall values. Because of their success and their ease of implementation, I decided to implement the DT classifiers in the TN-TIES system.

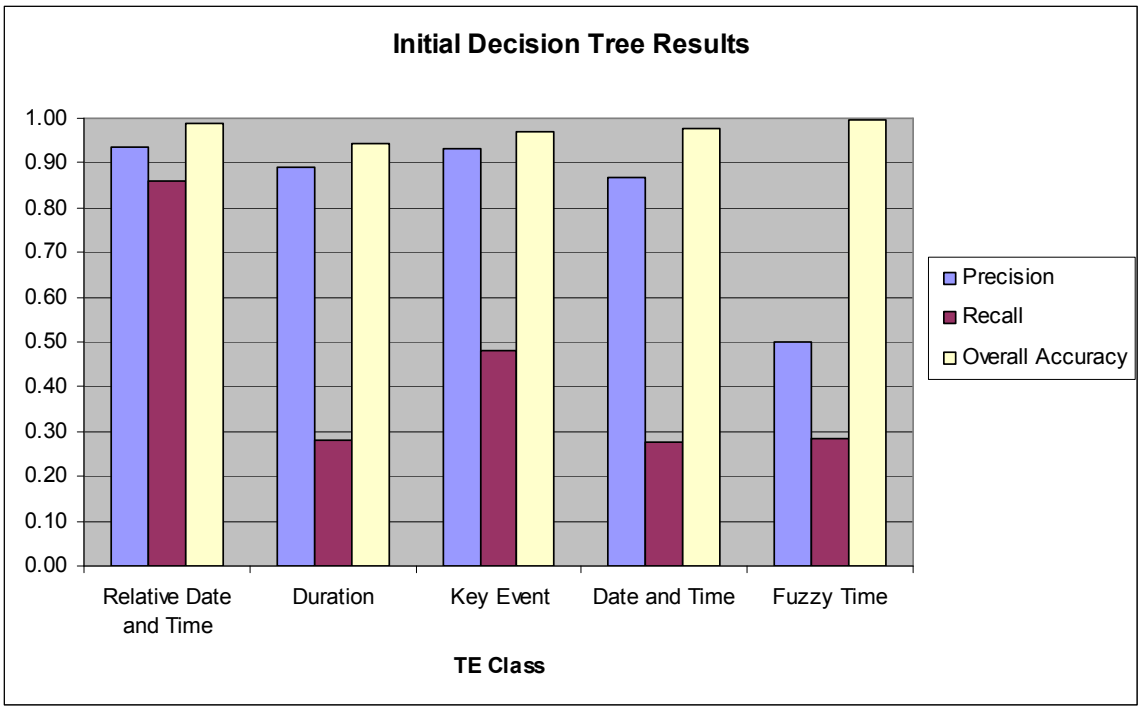

Figure 8: Results of Initial Decision Tree Classifiers

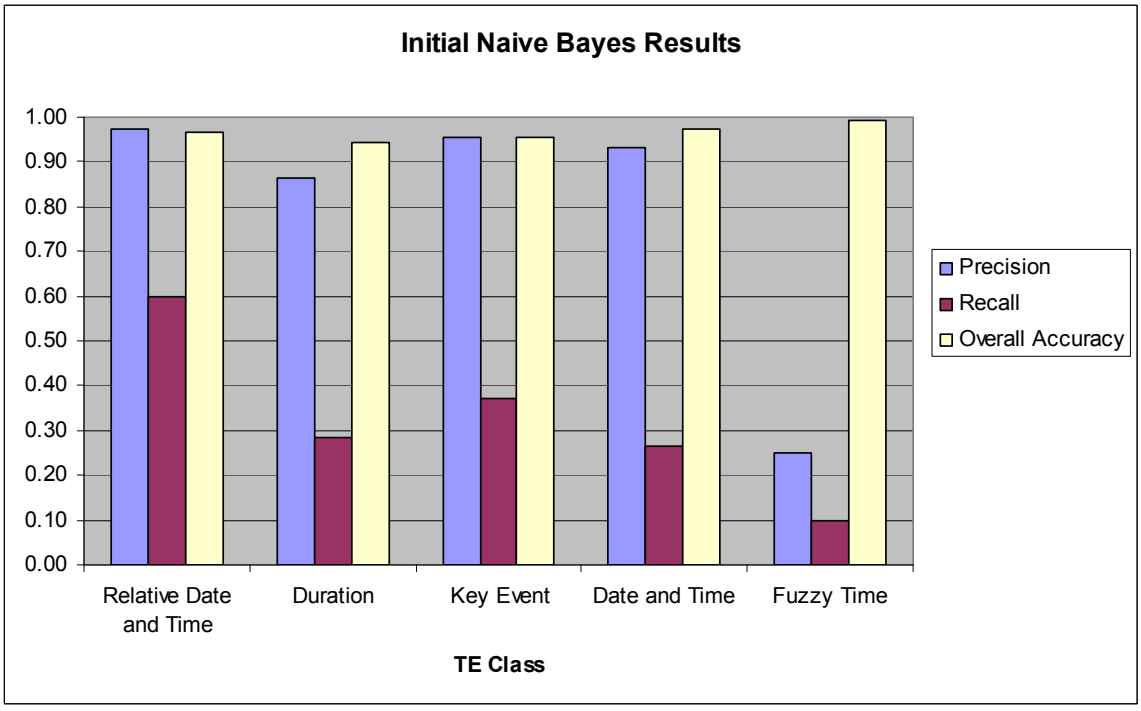

Figure 9: Results of Initial Naïve Bayes Classifiers

The final versions of the implementations of the RDT and Duration DT classifiers are shown in Figures 10 and 11, respectively. Compared with the feature space available to the classifiers (Table 3), both the RDT and Duration classifiers are relatively simple. The final precision and recall values of these DT classifiers on the training data are shown in Figure 12. The precision and recall values for both classifiers are above 90%.

If *ago*:
    RDT
If starts with *yest*:
    RDT
If *today*:
    RDT
If *tonight*:
    RDT
If *last*:
    And if *night* or *pm* or *evening or* #s or *several* or *couple* or *few* or *week* or *hour*:
        RDT
If *this*:
    And if *am* or *pm* or *evening* or *afternoon*
        RDT
    And if starts with *morn* or *week* or *month*
        RDT

*Note that DT includes additional morphological variants and regular expressions to match spelling variations

Figure 10: Decision Tree Classifier for Relative Date and Time TE class

If *since*:
    Duration
If *x* or *for* or *all* or *lasting* or starts with *preg*:
    And if *week* or *month* or *hour* or *minute* or *year* or *second* or *night*
        Duration
    And if *ago:*
        Not Duration

*Note that DT includes additional morphological variants and regular expressions to match spelling variations

Figure 11: Decision Tree Classifier for Duration TE class

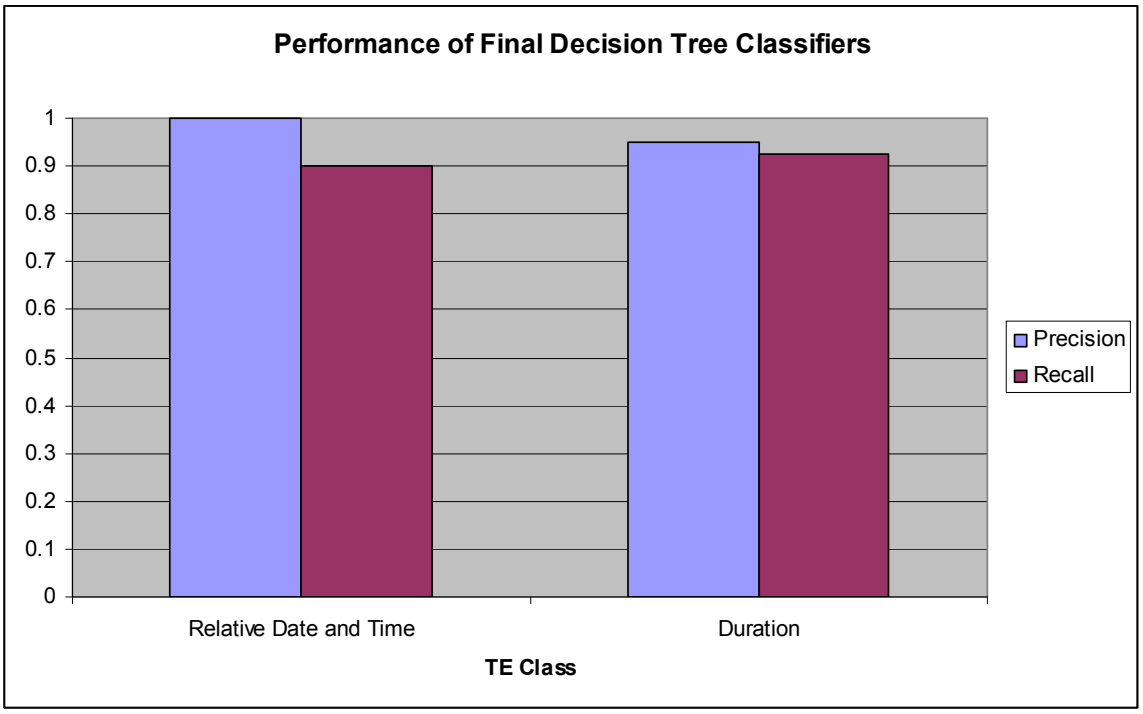

Figure 12: Results of Final Decision Tree Classifiers

The classifiers for the other TE classes are currently not operationally effective. This is a result of both small amounts of training data and their not undergoing a second round of iterative feature engineering. However, our research team does have access to more training data and the success of the RDT and Duration classifiers indicate that operational classifiers for all classes will be possible to achieve.

C. Interpreter

An example of the TN-TIES interpreter output is shown in Figure 13. The term *yesterday* is a trigger word for one of the RDT interpretation rules. The interpretation is based upon the TN timestamp. In this case, the interpreter reports that the seizure occurred sometime between 6:00 a.m. on 11/21/06 and 6:00 a.m. on 11/22/06. The RDT interpreter also

includes rules for phrases that indicate concepts such as “last night”, “this morning”, and “today.” Several trigger words and phrases may refer to a single concept. For example, the trigger phrases "this morning" and "this am" refer to a single concept. These concepts usually, but not always, correspond to the concepts generated as features for the purposes of classification. For example, all of the expressions contained within the "yesterday" classification feature ("yesterday", "yest", etc.) are also indicative of an interpretation concept. However, the days of the week (e.g., Monday, Tue, Fri, Saturday, etc.), which are grouped together for classification purposes, are interpreted independently.

2006-11-22 09:25:00 seizure

Pts mom reports pt had seizure yesterday. Treatment PTA: none.

TN-TIES output:

seizure yesterday:
between 2006-11-21 05:00 and 2006-11-22 05:00

Figure 13: Example TN-TIES Interpreter Output

VII. Discussion

TN-TIES is successful in achieving its processing goals. Additionally, it is an important first step towards a complete interpretation of text in the triage notes genre. Its development has implications for future work in the domain as well as on other text mining and natural language processing projects.

A. Implications

TN-TIES is the first of a series of processing modules that the TN project's system architecture (Figure 5) describes. It is currently able to identify and extract many of the temporal expressions in triage note text as well as interpret relative date and time TEs. Additionally, because it is the first of the series of modules, it has tackled many of the preprocessing and structural output tasks and decisions necessary to build an end-to-end system. TN-TIES has set an agenda for TN processing, defining the system's general look and feel and laying out appropriate approaches for managing TN text.

One of the most important lessons learned from this work is related to TN text processing as well as NLP in general and comes from the success of relatively simple, semi-supervised methods. I found that a chunker dependent on simple keywords and punctuation achieves a very high degree of accuracy for the task that we defined. Additionally, the relatively simple Decision Tree classifiers outperformed the more complex Naïve Bayes classifiers. These findings are consistent with previous information extraction work that demonstrates the success of simple methods (Cowie & Lehhert, 1996; Kraus, et al., in press). After implementing the simple classifiers in the system, my feature engineering attempts based on intuition and regular expression pattern matching greatly increased measures of accuracy. Throughout the system development, my close inspections of the training data and frequent uses of human intuition were valuable. These findings are also consistent with previous research. As discussed in the Background, hybrid approaches that combine manually constructed and machine learned rule sets have been effective in a variety of information extraction systems (Johansson, et al., 2005;

Verhagen, et al., 2005). In summary, typical information extraction techniques have worked well in processing TN text. In particular, a combination of simple hand written and machine learned approaches has been successful.

B. Future Work

TN-TIES currently does not classify and interpret all TEs. Extending the system to do so will require several efforts. First, the classifier modules for identifying TEs in the remaining five classes need to be improved. Currently, the RDT and the Duration classifiers are the only operational ones. Improving the accuracy, particularly recall, levels of the remaining classifiers will require further feature engineering. My iterative attempts at expanding the features relevant to the RDT and Duration classes have been successful, and those experiences indicate that it is possible to achieve a high classification accuracy for all classes. However, in addition to further feature engineering efforts, additional training data may be necessary to build operational classifiers for the remaining classes. The RDT and the Duration classes were the most common in the 598-sample, which made building corresponding classifiers easier than for the other classes. NC-DETECT has provided additional data, only further manual annotation efforts are necessary.

In addition to extending the classifier module to other classes, the interpreter module needs to be extended to apply to non-RDT TE classes. This requires manually developing interpretation guidelines, which then can be implemented. The guidelines for some

classes may seem trivial (e.g., Date and Time class), but explicitly stating the structure and output of their interpretation will make implementation easier.

Improving the TN-TIES chunker may be made possible by creating a domain specific POS-tagger that is trained on TN text. Such a tool could generate POS tags more accurately than taggers trained on standard edited English, and incorporating POS-based rules into the TN-TIES chunker may then improve its performance. Such a domain specific tool may be useful in the development of pruning and scoping rules as well.

In addition to extending the TN-TIES coverage, additional processing tools need to be built to complete the end-to-end interpretation process represented in Figure 5. Many of these tasks are fairly well-defined and the experiences described here will be helpful in approaching them. For example, TN-TIES successfully divides TN text into small, phrase-like chunks. Many of the chunks correspond to what we have defined as TEs, but the non-TE chunks are likely to correspond to other small, meaningful phrasal units. These chunks may, in particular, correspond to individual events (incidents and symptoms). Event tagging, as Figure 5 displays, will be critical to building a complete system. The existing chunking module in combination with the experience of having built such a tool will be valuable in the creation of an event tagger.

In addition to event tagging, Figure 5 calls for scoping, linking, and ordering modules. Chapman, Chu, & Dowling (2007) provide important insights regarding the creation of a scoping tool. Such a tool will provide additional links between events and temporal

references, beyond those that we have maintained in the TE annotations. Once all TEs have been iteratively resolved by use of the TN-TIES interpreter module as well as additional assumptions, ordering events on a timeline should be relatively straightforward.

Once we have developed a system that can generate a complete and accurate timeline of events reported in TN text, we may want to vary the timeline's presentation according to its different uses. In particular, we may want to include more or fewer details about an event depending on the application's context. This may be achieved by using POS tags or additional syntactical parsing steps to vary the boundaries and include or exclude certain types of syntactical constituents of TE and event tags. In linking the TN timeline to other medical records for purposes of biosurveillance, we are likely to want to include fewer details (e.g., including only *tripped today* from the TE *tripped over football at school today*). However, in training ED nurses to understand and use TNs, we are likely to want to include more details. Additionally, if ED clinicians view the timelines electronically, it will be possible to create visualization software that allows them to expand or collapse some information displayed on it. This will help them focus their attention while not denying them easy access to all of the details in the original TN.

As work continues on developing the TN processing system, we will also need additional input from domain experts. In particular, it will be important to understand the amount and type of information that is appropriate to display on the final timeline. Additionally,

we will need to continuously check the accuracy of our assumptions and interpretations of the primary text.

After the complete system is operational or, alternatively, in later stages of its development, it will be important to incorporate a variety of standards and methods of normalization into its output. One of the long term goals of the project is to be able to link information in the TN to information in other pieces of a patient record. We will only be able to achieve this if events and temporal references are standardized to support such automatic linking. Relevant standards will likely include TimeML, and mapping TNs to the controlled vocabularies that are a part of the Unified Medical Language System (UMLS: http://www.nlm.nih.gov/research/umls/) may help us take advantage of those structures that are standardized in other parts of the patient record.

VIII. Conclusion

The availability of electronic patient medical records offers an extraordinary opportunity for researchers to use natural language processing and text mining techniques in order to improve health care, advance biosurveillance methods, and contribute to the education of clinicians. Building applications requires not only access to data but also a fresh approach to processing the text, which is of a unique genre and has not yet been thoroughly examined. Successful projects will combine existing NLP techniques and innovative, domain-specific approaches.

The current work identifies a timeline of events as a key visual representation that may be generated from processing TN text. TN-TIES is a series of modules that extracts, classifies, and interprets temporal expressions within triage notes, which is a first step towards automatically creating a complete timeline. I have used a combination of partial parsing (chunking), machine learning classifiers, and hand-written rules and interpretation guidelines based on domain knowledge in order to build TN-TIES. Each of the three modules (chunker, classifier, interpreter) achieve an operational level of accuracy. The success of TN-TIES suggests that a system that processes TN text and generates a complete timeline of events is feasible.

References

Adlassnig, K. P., Combi, C., Das, A., Keravnou, E. T., & Pozzi, G. (2006). Temporal representation and reasoning in medicine: Research directions and challenges. *Arificial Intelligence in Medicine*, 38(2), 101-113.

Ahn, D., Fissaha Adafre, S., & de Rijke, M. (2005a). Extracting temporal information from open domain text: A comparative exploration. *Journal of Digital Information Management*, 3(1), 14-20.

Ahn, D., Fissaha Adafre, S., & de Rijke, M. (2005b). Towards task-based temporal extraction and recognition. *Proceedings of the Dagstuhl Workshop on Annotating, Extracting, and Reasoning about Time and Events*, Dagstuhl, Germany.

Ahn, D. (2006). The stages of event extraction. *Proceedings of the 2006 ACL Workshop on Annotating and Reasoning about Time and Events*. Sydney, Australia. 1-8.

Allen, J. (1984). Towards a general theory of action and time. *Artificial Intelligence*, 23(2), 123-154.

Allen, J. & Hayes, P. (1989). Moments and points in an interval-based temporal logic. *Computational Intelligence*. 5(3), 225-238.

Bramsen, P., Deshpande, P., Lee, Y., & Barzily, R. (2006). Finding temporal order in discharge summaries. *AMIA Annual Symposium Proceedings*, 81-85.

Cowie, J., & Lehnert, W. (1996). Information extraction. *Communications of the ACM*, 39(1), 80-91.

Chapman, W. Chu, D., & Dowling, J. (2007). ConText: An algorithm for identifying contextual features from clinical text. *Proceedings of the 2007 ACL Workshop on Biological, Translational, and Clinical Language Processing (BioNLP).* Prague. 81-88.

Chapman, W. & Dowling, J. (2007). Can chief complaints identify patients with febrile syndromes? *Advances in Disease Surveillance*, 3(6).

Chapman, W., Dowling, J., & Wagner, M. (2005). Classification of emergency department chief complaints into 7 syndromes: A retrospective analysis of 527,228 patients. *Annals of Emergency Medicine*, 46(5), 445-455.

de Lusignan, S. & van Weel, C. (2006). The use of routinely collected computer data for research in primary care: Opportunities and challenges. *Family Practice*, 23(2), 253-263.

Fissaha Adafre, S. & de Rijke, M. (2005). Feature engineering and post-processing for temporal expression recognition using conditional random fields. *Proceedings of the 2005 ACL Workshop on Feature Engineering for Machine Learning in Natural Language Processing*. Ann Arbor, MI. 9-16.

Gaizauskas, R., Harkema, H., Hepple, M., & Setzer, A. (2006). Task-oriented extraction of temporal information: The case of clinical narratives. *Thirteenth International Symposium on Temporal Representation and Reasoning* (TIME'06), 188-195.

Han, B., Gates, D. & Levin, L. (2006). Understanding temporal expressions in emails. *Proceedings of the Human Language Technology Conference of the North American Chapter of the ACL*, 136-143.

Haas, S.W., Travers, D.A., Waller, A., & Kramer-Duffield, J. (2007) What is an event?: Domain constraints for temporal analysis of chief complaints and triage notes. *Proceedings of the American Society for Information Science and Technology*, 44(1).

Hobbs, J. R. & Pan, F. (2004). An ontology of time for the semantic web. *ACM Transactions on Asian Language Processing (TALIP): Special issue on Temporal Information Processing*, 3(1), 66–85.

Hripcsak, G., Friedman, C., Alderson, P.O., DuMouchel, W., Johnson, S.B., & Clayton, P.D. (1995). Unlocking clinical data from narrative reports: A study of natural language processing. *Annals of Internal Medicine*, 122, 681-688.

Hripcsak, G., Zhou, L., Parsons, S., Das, A., & Johnson, S. (2005). Modeling electronic discharge summaries as a simple temporal constraint satisfaction problem. *Journal of the American Medical Informatics Association*, 12(1), 55-63.

Johansson, R., Berglund, A., Danielsson, M., & Nugues, P. (2005). Automatic text-to-scene conversion in the traffic accident domain. *International Joint Conference on Artificial Intelligence*, 19, 1073-1078.

Kraus, S., Blake, C., & West, S. L. (in press). Information extraction from medical notes. *Proceedings of the 12th World Congress on Health (Medical) Informatics – Building Sustainable Health Systems (MedInfo)*, Brisbane, Australia, 1662-1664.

Mani, I., & Wilson, G. (2000). Robust temporal processing of news. *Proceedings of the 28th Annual Meeting of the Association for Computational Linguistics*, 38(1), 69-76.

Pustejovsky, J., Castaño, J., Ingria, R., Saurí, R., Gaizauskas, R, Setzer, A., & Katz, G. (2003). TimeML: Robust specification of event and temporal expressions in text. *AAAI Spring Symposium on New Directions in Question Answering*. Stanford, CA. 28-34.

Suominen, H., Lehtikunnas, T., Back, B., Karsten, H., Salakoski, T., & Salanterä, S. (2007). Applying language technology to nursing documents: Pros and cons with a focus on ethics. *International Journal of Medical Informatics*, 76(S), S293-S301.

Travers, D.A. & Haas, S.W. (2003). Using nurses' natural language entries to build a concept-oriented terminology for patients' chief complaints in the emergency department. *Journal of Biomedical Informatics*, 36(4-5), 260-270.

Zhou, L., & Hripcsak, G. (2007). Temporal reasoning with medical data—A review with emphasis on medical natural language processing. *Journal of Biomedical Informatics*, 40, 183-202.

Appendix A

Triage note temporal expression represented with TimeML.

HA and fever for last three days

<MAKEINSTANCE eiid=”ei1” eventID=”e1” tense=”PRESENT” aspect=”NONE” pos=”NOUN”/>
<EVENT eid=”e2” class=”SYMPTOM”>
HA
</EVENT>
<MAKEINSTANCE eiid=”ei2” eventID=”e2” tense=”PRESENT” aspect=”NONE” pos=”NOUN”/>
<SLINK eventInstanceID=”ei1” subordinatedEventInstance=”ei2” relType=”EVIDENTIAL” signalID=”s1”/>
and
<EVENT eid=”e3” class=”SYMPTOM”>
fever
</EVENT>
<MAKEINSTANCE eiid=”ei3” eventID=”e3” tense=”PRESENT” aspect=”NONE” pos=”NOUN”/>
<SLINK eventInstanceID=”ei1” subordinatedEventInstance=”ei3” relType=EVIDENTIAL” signalID=”s1”/>
<SIGNAL sid=”s1”>
for
</SIGNAL>
<TIMEX3 tid=”t1” type=”DURATION” value=”P3D” beginPoint=”CALCULATED from timestamp”>
last three days ,
</TIMEX3>
<TLINK eventInstanceID=”e2” relatedToTime=”t1” signalID=”s1” relType=”IS_INCLUDED”/>
<TLINK eventInstanceID=”e3” relatedToTime=”t1” signalID=”s1” relType=”IS_INCLUDED”/>