

# Clinical and Financial Outcomes Analysis with Existing Hospital Patient Records

R. Bharat Rao, Sathyakama Sandilya Computer Aided Diagnosis Siemens Medical Solutions, Malvern, PA 19355, USA (610) 219-8454	Radu Stefan Niculescu Dept of Computer Science Carnegie Mellon University 5000 Forbes Avenue Pittsburgh, PA 15213, USA (412) 268-3613	Colin Germond, MD Head, Treatment Planning Cancer Care Ontario Sudbury, ON P3E5J1, Canada (705) 522-6237	Harsha Rao, MD, FRCP Professor of Medicine, Univ. of Pittsburgh Medical Center 3601 Fifth Avenue, Pittsburgh, PA 15213, USA (412) 383-7758
---	--	---	---

## ABSTRACT

Existing patient records are a valuable resource for automated outcomes analysis and knowledge discovery. However, key clinical data in these records is typically recorded in unstructured form as free text and images, and most structured clinical information is poorly organized. Time-consuming interpretation and analysis is required to convert these records into structured clinical data. Thus, only a tiny fraction of this resource is utilized. We present REMIND, a Bayesian Framework for Reliable Extraction and Meaningful Inference from Nonstructured Data. REMIND integrates and blends the structured and unstructured clinical data in patient records to automatically create high-quality structured clinical data. This structuring allows existing patient records to be mined for quality assurance, regulatory compliance, and to relate financial and clinical factors. We demonstrate REMIND on two medical applications: (a) Extract “recurrence”, the key outcome for measuring treatment effectiveness, for colon cancer patients (ii) Extract key diagnoses and complications for acute myocardial infarction (heart attack) patients, and demonstrate the impact of these clinical factors on financial outcomes.

## Categories and Subject Descriptors

H.2.8 [Database Management]: Database Applications - Data Mining.

## Keywords

Data Mining, Temporal Reasoning, Bayes Nets, HMMs.

## 1. INTRODUCTION

Hospitals collect vast amounts of electronic information during their daily operations, as they treat and bill patients. These records are a valuable source for outcomes analysis, knowledge discovery, and evaluating the quality of patient care. However, they are underutilized because of the difficulty of analyzing the *clinical* information in existing hospital patient records. This is illustrated

in the example below.

Early detection of breast cancer is one of the leading issues in healthcare. The standards of care demand that physicians inform women who are at risk for breast cancer (typically those over 40), that they need to undergo an annual screening mammogram. This applies to all women seen at the hospital (e.g., a doctor’s visit for high fever). Physician compliance with this guideline is an important measure of hospital quality.

It is quite simple to query the database to gather all hospital records of women over age 40, and then to check how many of them have had a mammogram. This fraction, however, is rarely meaningful for measuring compliance. For instance, a woman may have already had a mammogram at another hospital, or might reject the physician’s advice. For others, a different examination may be medically indicated. This information is typically present in *unstructured* form, namely, as free text in transcribed doctors’ dictations. The common way to measure compliance with this guideline is for a medically qualified person to read these patient records, and determine if the physician has informed the patient about a mammogram, and has documented why it was not done.

Given that this very basic information cannot be determined, it is not surprising that more complex questions about patient care cannot be answered. For instance, it is difficult to tell (without reading the radiologist’s report) which mammograms have abnormal findings, and of these patients, which have been correctly followed-up (with further imaging exams or biopsy), and if positive, whether the patient was correctly treated for cancer.

The fundamental problem addressed by this research is the lack of structured clinical data to support outcomes and quality analysis. Section 2 describes the “data gap” in hospital patient records, its impact on KDD, and presents an abstraction of this problem into inference from multiple data sources. In Section 3 we describe REMIND (for Reliable Extraction and Meaningful Inference from Non-structured Data), a Bayesian reasoning framework that *combines all available data in a principled fashion over time*. REMIND uses easily available domain knowledge to combine information available from multiple sources at different times, to overcome the inherently unreliable data in patient records. We demonstrate REMIND on two different medical applications.

Section 4 provides results on many experiments for inferring the value of *recurrence*, a complex outcomes variable for cancer patients. First we demonstrate the value of incorporating temporal inference into classification: REMIND’s performance is superior to that of many classification and text mining methods. We also show that REMIND improves performance by incorporating information from multiple data sources (structured and

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

SIGKDD '03, August 24-27, 2003, Washington, DC, USA.  
Copyright 2003 ACM 1-58113-737-0/03/0008...\$5.00.

unstructured) into the inference process, and that it is possible to incorporate the results of mining algorithms into REMIND to learn domain knowledge parameters automatically. In Section 5, we present some preliminary results for using REMIND on patients admitted to the ICU with a diagnosis of acute myocardial infarction (AMI or heart attack). By using REMIND for the extraction, the hospital can obtain a more accurate picture of the impact of clinical conditions and complications on financial outcomes. We briefly review the rich body of related work in Section 6. Section 7 summarizes our approach and conclusions.

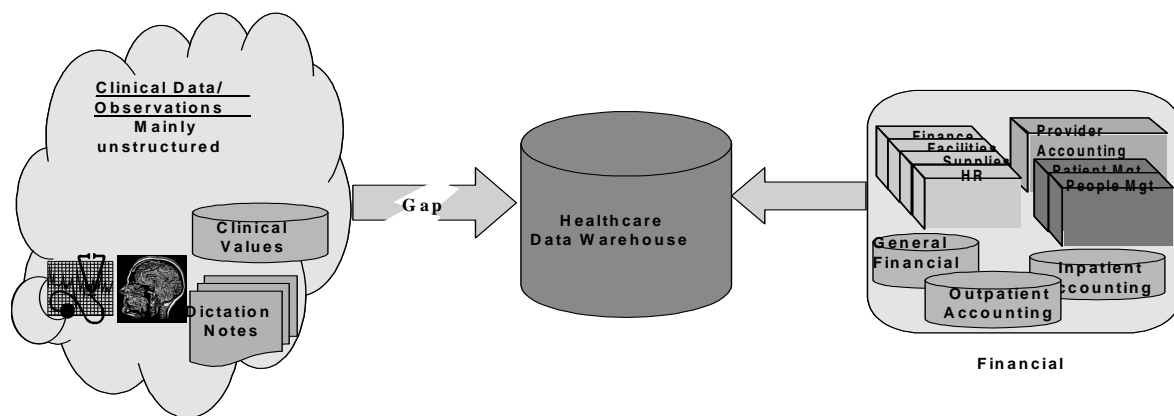


Figure 1: The "Data Gap" is existing hospital patient records

## 2. MOTIVATION

The breast cancer scenario shows that hospitals do collect some basic patient information in structured form, for example, in billing, insurance, lab, and pharmacy databases. However, most of the key clinical information is stored in unstructured form: as free text in admission slips, progress notes, radiology reports, cardiologist findings, etc., as images in various imaging studies (X-Ray, MR, CT, etc.), and possibly even as waveforms. This data requires medical interpretation and analysis before it can be used.

### 2.1 "Data Gap" in Existing Patient Records

Figure 1 illustrates the difficulty of getting high-quality structured clinical data in a usable form. Given this "Data Gap", there are two possible routes for measuring quality and outcomes. (i) If analysis must be supported by the power of numbers (i.e., over many patient records), only the simplest of measures can be computed, based upon already-collected patient data. As we shall see in Section 5, even this rudimentary analysis can be unreliable. (ii) If the analysis needs highly accurate or complex data, medically qualified personnel must manually extract this from patient charts. Obviously, this is feasible only for a small number of patients. For instance, hospitals must manually review only 75 patient charts monthly to comply with reporting requirements for JCAHO [11], the regulatory body for US hospitals.

*At present there is no solution that allows for accuracy in the context of large numbers of patient records by combining both structured and unstructured data.*

At first glance, it may seem difficult to believe that electronic clinical repositories (illustrated by the cloud in Figure 1) are so poorly organized to be unusable. However, for a physician, free text dictations are the most efficient way to document patient history and the complexities of an individual case. Further, other

physicians can understand these notes when treating the patient. Thus, the "Data Gap" does not hinder the practice of medicine.

### 2.2 How is this Relevant to KDD?

From a knowledge discovery point of view, it is frustrating that existing patient data cannot be mined. Despite ongoing efforts {SNOMED}, there are no uniformly accepted standards that are universally used. The gold standard for the medical analysis and discovery is based on *prospective* data that is carefully collected in clinical trials. Any *retrospective* analysis of hospital records requires manual review of charts, or time-consuming data

collection efforts to capture structured data. The reality is that the merest fraction of existing patient data is used for KDD activities.

REMIND aims to "fill this data gap" by automatically creating high quality structured clinical data from existing patient records. In some sense, this can be viewed as "*data discovery*" rather than knowledge discovery. However, once the structured data is created, it leads to new knowledge, often by a simple inspection of the structured data, or via first-order correlations. We provide two examples of simple knowledge discovery below.

(1) In previous work [25][26], we analyzed treatment of colon cancer patients in the first 6-9 months (that is when proper application of chemotherapy guidelines can improve survival by 20 percentage points [8]). By extracting process and guideline information we discovered a key bottleneck in the treatment of patients. The hospital has since changed the way these patients are treated; which is estimated to save up to 5 lives every year.

(2) Analyzing the original structured data used in our experiments in Section 5, may lead to the mistaken impression that diabetes (as a complication) has little impact on heart attack outcomes. (This is due to errors in coding diagnoses and complications.) After REMIND has automatically extracted the clinical data, a simple inspection reveals that diabetic AMI's have much higher costs and lengths-of-stay than non-diabetic AMI's. Based on this the hospital can begin specialized programs to treat diabetic AMI's.

The RODS [29] (Real-time Outbreak and Disease Surveillance) system was commissioned at the last Winter Olympics in an effort to detect a bio-terrorism event by analyzing emergency room patient records. Extracting structured information from the existing patient records can further help in this detection task.

Further once the structured clinical data has been extracted from the hospital records, that data itself can be mined. For instance, a

large database of cancer patients for whom the outcome is correctly extracted can be mined to determine the risk factors and the impact of treatment variations (in that hospital) on outcome.

To summarize, we believe that the “Data Gap” in Figure 1, is the key bottleneck that prevents mining of hospital patient records. The REMIND framework uses easily available domain knowledge to combine information from all available data to automatically create high-quality structured clinical databases.

### 2.3 Problem Abstraction

Data about real-world systems, particularly those with a high cost of failure, are often collected in multiple “*data sources*.” The collected data may be stored in a structured format, such as tables in a relational database (i.e., in a *structured* data source), or as free text or images (i.e., in an *unstructured* data source). Often, critical process information is stored only in an unstructured format; and occasionally may not be recorded reliably by any source. The performance task is to fill the “Data Gap”, i.e., infer the values of key process variables as they change over time from the available data, using easily available domain knowledge.

We need to consider several issues. Each source may begin collecting data at a different time, and may record data at arbitrary points in time, not at evenly spaced time intervals. Data sources may have differing degrees of reliability. The structured data sources are typically missing critical information (hence the need for combining information from all sources in the first place), and may also contain errors. The information in unstructured sources, such as free text, may be reliable when read (and understood) by a human expert. However, information extracted by a computer from the same source (e.g., via natural language processing of the free text) may be unreliable; typically due to uncertainties in the extraction, and sometimes due to errors/imprecision in the original free text. Furthermore, some information may be unavailable, because it is never recorded in electronic form. And, finally, only approximate domain knowledge may be (easily) available.

## 3. DESCRIPTION OF APPROACH

This problem of inference from multiple data sources may be tackled in several different ways. One approach is to improve methods of data collection. Another is to develop better methods for extracting information from unstructured data. A third is to use very accurate domain knowledge (either from a domain expert, or learning it from labeled data). REMIND is different, and yet complementary to the above, as better data collection / extraction, and improved domain knowledge will only improve our results.

Our approach to inference with this data is to model the data as arising from a generative process, and combine prior knowledge about this process with observations for a specific patient using Bayesian techniques. Patient data is collected in a hospital at arbitrary points in time, and these sampling instants vary from patient to patient. Hence, we model the processes of progression of patients’ diseases and the collection of this data as continuous time processes that may be sampled at arbitrary instants. We consider a model wherein a patient has a state (for the disease of interest), and observations about the state and related variables are stored in and may be collected from various data repositories.

### 3.1 Problem Definition

Let  $\mathbf{S}$  be a continuous time random process taking values in  $\Sigma$  that represents the state of the system; note that  $\mathbf{S}$  may be a

combination of multiple variables. Let  $\mathbf{T}=\{t_1, t_2, \dots, t_n\}$ , where  $t_i < t_{i+1}$ , be the  $n$  “times of interest when  $\mathbf{S}$  has to be inferred. Let  $\mathbf{S}_i$  refer to the sample of  $\mathbf{S}$  at time  $t_i \in \mathbf{T}$ . Note that  $\mathbf{T}$  and  $n$  can vary for different realizations of the process.

Let  $\mathbf{V}$  be the set of variables that depend upon  $\mathbf{S}$ .

Let  $\mathbf{O}$  be set of all (probabilistic) observations for all variables,  $v \in \mathbf{V}$ . Let  $\mathbf{O}_i$  be the set of all observations “assigned” to  $t_i \in \mathbf{T}$ ; i.e., all observations about variables,  $v \in \mathbf{V}$  that are relevant for this time-step  $t_i$ . Similarly, let  $\mathbf{O}_i(v)$  be the set of observations for variable  $v$  “assigned” to  $t_i$ .

Let  $seq$  be a random variable in  $\Sigma^n$ ; i.e., each realization of  $seq$  is a specific (legal) sequence  $\langle \mathbf{S}_1, \mathbf{S}_2, \dots, \mathbf{S}_n \rangle$ .

In the case when we are interested only in the value of a variable at a point in time (e.g., in the AMI example, we simply wish to know if the patient really had an AMI), our goal is to estimate:

$$V_{MAP} = \arg \max_v P[V | O]$$

When we wish to track the patient’s progress over time (e.g., for colon cancer recurrence), our goal is to estimate the most likely state sequence,  $seq_{MAP}$ , the maximum a-posteriori estimate of  $seq$  given  $\mathbf{O}$ .

$$seq_{MAP} = \arg \max_{seq} P[seq | O]$$

### 3.2 Overview of Approach

We view  $\mathbf{S}$  as a continuous time Markov process from which we observe non-uniform samples. Our implementation of REMIND assumes that  $\mathbf{S}$  is a stationary Markov process, whereas variables,  $v \in \mathbf{V}$  that depend on  $\mathbf{S}$  have conditional distributions (on the parent variable) that are non-stationary. However, our framework can be extended to handle even non-stationary Markov processes.

REMIND’s 3-step process that estimates the distribution of the variable of interest  $V_{MAP}$  (or  $seq_{MAP}$ ) is summarized below. Our goal is to extract and combine information from all data sources.

- (1) **Extraction** step: information about the variables,  $v \in \mathbf{V}$ , is extracted from the data sources, and converted into a uniform representation, called *probabilistic observations*. These play the same role as likelihood findings in standard Bayesian reasoning. Every observation  $o \in \mathbf{O}$  is assumed to be potentially incorrect.
- (2) **Combination** step: each observation is assigned to one time of interest,  $t_i \in \mathbf{T}$ . Then each state,  $\mathbf{S}_i$  is estimated from  $\mathbf{O}_i$ .
- (3) **Inference** step: the inferences are propagated across time.

### 3.3 Extraction: Observations from data

In this step we *produce probabilistic observations*,  $o_i$ , from data sources. Each  $o_i$  is drawn entirely from a single piece of information in a data source (e.g., from a phrase in a sentence, or a row in a database), and hence is assumed to be inherently undependable (either due to errors in the data or in the extraction process). An observation  $o_i$  is of the form  $\langle \text{NAME}, \text{DATE}, \text{DIST} \rangle$  where NAME is an observed variable  $v \in \mathbf{V}$ , DATE is the date of the observation, and DIST defines a distribution over all possible values that can be taken by NAME given the observation. REMIND currently does extraction from 2 types of data sources.

### 3.3.1 Structured Data Source: Relational Databases

REMIND communicates with all databases using SQL. REMIND is implemented in Java, and its built-in interface to relational databases (JDBC) allows uniform access to data in different DBMS. The results of executing a query on the database, for instance, retrieving a lab test result, can be expressed as another probabilistic observation; the DIST can encapsulate any combination of the inherent error associated with the test (some tests need to be repeated to determine their accuracy), the uncertainty about threshold value (if reported as a qualitative result), or the likelihood of an error in the database. Note that a null result of executing a query can also generate an element, for instance, about a procedure not being performed on a patient. Here the DIST can capture the likelihood that the procedure was performed at another institution.

Notice that, in the above examples, uncertainty arises because the observation is drawn just once from a single data source. Repeated observations of the same variable, for example, the same test repeated a week later, or confirmation in the doctors' notes about a procedure, can change DIST, the *a posteriori* distribution for any variable.

### 3.3.2 Unstructured Data Source: Free Text

The REMIND interface partitions a document into sections based upon knowledge about the structure of the document. Observations, such as document date, can be extracted from the structured sections (e.g., headers or footers). Each unstructured section is converted into a stream of tokens, via a standard lexical parser (JFlex, a Java implementation of FLEX). The tokens are then gathered into sentences (terminated by EOS token).

*Phrase-spotting Rules:* We extract information from the token stream via phrase spotting, a simple and easy-to-implement method from computational linguistics. Phrase spotting is about as simple as it sounds! A phrase-spotting rule is applied within a single sentence. A rule encodes the knowledge that when a pattern is found in the document, then the extraction system should create a new observation (which is then used by the reasoning engine). The pattern in a rule is specified in a rudimentary phrase description language that allows a user to define patterns of co-occurrence of words in a sentence. It can also encode compound rules wherein one part of the pattern to be matched is the pattern of another rule. Moreover, it allows for encoding synonyms so that only one rule has to be written for all the synonyms of a particular word. When a rule is matched, then an observation is created for the specified variable with the specified distribution e.g., a rule whose pattern is "evidence" & "recurrence" and whose action is "Recurrent (true, 0.8, false, 0.2)" generates an observation stating that the value of the variable "Recurrent" is "true" with probability 0.8 and "false" with probability 0.2.

These observations generated from the data sources are meant to encode the *a posteriori* distribution of a variable given the section of the data source that they are extracted from, and are subsequently converted into likelihood findings for computation in the Bayesian Network.

## 3.4 Combination + Inference

The primary focus of our interest is estimating what happened to the system across the duration of interest. Hence, a natural abstraction of the problem is to look for the best estimate of the

sequence of system states across time, and the maximum *a posteriori* (MAP) estimate is one that maximizes the probability of picking the correct sequence. Hence, given the observations that we have extracted, we would like to estimate the *a posteriori* probability of each legal state sequence and pick the best. This can be done in two steps, the first of which is combination of observations at a fixed point in time and the propagation of these inferences across time.

We use a Markov Model to estimate the evolution of the patient's state. As the observations about patients are spaced non-uniformly across time, the standard discrete-time Markov approximations are not necessarily justifiable. In order to overcome this shortcoming, we model the process of evolution of the patient state as a continuous-time Markov process from which we get to observe non-uniform samples. More specifically, the parameters we need to model are the dwell time in each state and the transition rates from each state to every other. In our current implementation, we consider the state to be a stationary Markov process whereas the other variables that depend on it can have conditional distributions that are non-stationary. Our framework, however, can be modified to handle even the case of non-stationary state processes.

Each piece of information that is extracted in the previous step is in the form of an *a posteriori* probability of a variable given the small context that it is extracted from. We can thus have multiple such assertions from different parts of the same source and from different sources at any given instant in time. All the assertions about a variable at a given point in time are combined into one assertion in a straightforward manner by using Bayes' theorem (under the assumption that the observations are independent given the variable) as follows:

$$\Pr[v | O_1^i(v), \dots, O_k^i(v)] \propto \Pr[v] \cdot \prod_{j=1}^k \Pr[O_j^i(v) | v] \propto \frac{\prod_{i=1}^k \Pr[v | O_i^i(v)]}{\Pr[v]^{k-1}}$$

We model the relationships between the set of all variables of interest using a Bayesian Network, which is used to infer the posterior distributions of all the variables at a given point in time given all the information at that time. For inference across time, we may now use a standard dynamic programming based approach (e.g. the Viterbi algorithm, see [24] for details).

Because we model the state process as being Markov, we have the following equation that connects the *a posteriori* probability of a sequence of samples of the state process given all the observations to the temporally local *a posteriori* probability of the state given all observations at each time instant.

$$\begin{aligned} \Pr[seq | Obs] &\propto \Pr[S_0] \cdot \prod_{i=2}^n \Pr[S_i | S_{i-1}] \cdot \prod_{i=1}^n \Pr[Obs_i | S_i] \\ &\propto \prod_{i=2}^n \frac{\Pr[S_i | S_{i-1}]}{\Pr[S_i]} \cdot \prod_{i=1}^n \Pr[S_i | Obs_i] \end{aligned}$$

## 4. EXPERIMENTS: COLON CANCER

The experiments described in this section demonstrate various aspects of REMIND. First, by comparing REMIND with traditional classification and text mining algorithms we show that incorporating temporal constraints can improve classification

performance. Further, we show that REMIND combines structured and unstructured data, and that REMIND's performance improves as additional data sources are available. Finally, we show that data mining algorithms can be "plugged-into" REMIND, to automatically learn the domain knowledge used by REMIND. We begin by describing the medical application and the data.

## 4.1 Data Sources and Domain Knowledge

### 4.1.1 Background: Colon Cancer

Colon cancer patients are grouped into 4 stages, such that each group is homogeneous in respect of survival [1][27]. Our study focuses on Stage III cancer patients: roughly 50% of Stage III patients survive. (Most Stage I and II patients survive, and virtually all Stage IV patients do not.) The key outcomes variable for colon cancer is not death from cancer, but in fact, *recurrence of cancer*; i.e., the cancer is cured and then returns. Because Stage III recurrence is almost invariably fatal, the key measure of performance is *time to recurrence*; i.e., in how many patients did the cancer return, and for those, how long were they disease free.

### 4.1.2 Data Sources at Cancer Care Ontario

We study 344 Stage III colon cancer patients at CCO, a cancer care hospital in Sudbury, Canada. The source of *structured* patient data is OPIS, an oncology patient relational database [4]. OPIS contains data about patient demographics, staging, diagnosis date, and administration of drugs. However, OPIS contains no reliable information about recurrence, the most important outcome. The principal source of *unstructured patient data* is doctors' dictations. CCO began storing data in OPIS from 1988, and began transcribing dictations in 1994.

Our data is based on a snapshot of both data sources taken in January 2001. We restrict our study to patients for whom at least one dictation has been stored. The earliest and latest a patient in our study began treatment was 1985 and November 2000 respectively. Patients averaged 10.8 dictations, with a maximum of 53 dictations. 18 patients had just 1 dictation, and 54 patients had 3 or fewer dictations. (As indicated earlier,  $T$  and  $n$  vary for different "realizations of the process", i.e., patients.)

### 4.1.3 REMIND Domain Knowledge

Domain knowledge (DK) for REMIND is fairly simple:

**Identify state S.** Here  $S$  is a single variable, "disease-state." Since the only transition of interest is from 0 (cured) to 1 (recurrent), we define the legal sequences as  $0^*$  (disease-free) or  $0^*1^*$  (recurrent). Our goal is to document the patient state as diagnosed by the doctor, i.e., to infer when the doctor diagnosed the patient as recurrent. (Note the distinction between State and Stage.)

**Identify V, and the data sources from which they may be extracted.** The variables drawn from doctors' dictations are "recurrence" and "CEA" (a test result documented in the notes, and used as a marker for recurrence). OPIS's DB-Chemo table provides information about chemotherapy treatment and its intent. OPIS's DB-Recur table provides information about recurrence in cases where the hospital recorded this information.

**Identify extraction DK.** 17 simple phrase spotting rules are used to extract information from the text, in addition to 2 compound rules to detect negation and imprecision. SQL queries detect the presence of chemotherapy and/or recurrence in OPIS.

**Define times of interest, T** and assign observations to each time. Each visit to the doctor is a time of interest (the resolution is in days). Any observations between visits are assigned to the next visit. Recurrence diagnoses typically occur during visits (or are documented only at a patient visit, which amounts to the same).

**Identify local dependencies.** A simple Bayesian network between  $S$  and  $V$  is used to infer the posterior distributions of  $S_t$  from  $O_t$ . We model 5 variables of interest for every patient visit, namely assertions about recurrence, the patient's CEA (if the test was performed), patient chemotherapy and if given, the intent of the chemotherapy, and the disease state. The conditional distributions in this network were assigned in consultation with domain experts. The network is replicated for each time instant of interest with the state at a  $t$  being the "cause" of the state at  $t+1$ .

**Determine dwell times and transition probabilities.** The medical literature provides survival curves for colon cancer patients [8]. We approximated these by an exponentially decaying curve (we could easily have used the actual curve and interpolated, but this was most convenient to implement).

## 4.2 Experimental Setup

### 4.2.1 Performance Measures

REMIND's output is a sequence of the form  $0^*$  or  $0^*1^*$  (where "0" corresponds to not recurrent, and "1" to recurrent). We assess performance in two ways:

- Patient classification: for accuracy of classification of final state: whether or not the patient recurred ("final state=1")?
- Patient visit classification: classification accuracy for each state. So, if a patient with 7 visits is declared recurrent on the 4<sup>th</sup> visit, the first 3 visits are labeled "0" and the last 4, "1".

A related measure to [B] is Sequence prediction: if the patient recurred, how accurately did we estimate the date of recurrence?

### 4.2.2 Experimental setup for REMIND

The performance of REMIND is assessed with respect to "ground truth" – the classification made by a colon cancer specialist after reviewing the entire patient record for each patient (a process that took the doctor about 3 months). 40% of these patients (138) were randomly chosen to be in the "training set." We reviewed the notes on half the training set to arrive at 17 generic text extraction rules, and then ran REMIND on the other half of the training set to adjust potential gross errors in the conditional probabilities in the model (both in consultation with a domain expert). However, we refrained from any fine-tuning of the rules. For example, all phrase rules asserted observations with the same probability.

After making all such adjustments, we ran the system over the remaining 60% (206) patients, using different combinations of data sources and domain knowledge. Execution time was not an issue: for all 206 of the test patients (i.e., executing all 3 steps for every test set patient), REMIND took less than 60 seconds total on a Pentium III 800 MHz laptop (most of the 60 seconds was spent in disk access time to read the dictations).

### 4.2.3 Adjustments to ground truth

REMIND initially produced 7 classification errors on the training set. However, on further review, we found that 2 of these were not classification errors, but in fact misclassifications by our specialist (i.e., our specialist had overlooked a couple of factors in patient history while determining ground truth – not uncommon when a physician has to wade through a large patient record to get an

accurate summary of the patient history). The results presented herein are after correcting the ground truth for these mistakes. We have not yet done such an analysis for the Test Set.

#### 4.2.4 Train set – test set splits

We also ran experiments on the data with classification algorithms from *Naïve Bayes* (NB)[20], *k-Nearest Neighbor* (*k*-NN)[20], and *Support Vector Machines* (SVM)[3], and also with *RAINBOW*, a *bag-of-words text-mining* system[19].

REMIND’s use of a “*training set*” is different from the way any of these systems would use a training set. REMIND currently does no learning or model training. Instead the “*training set*” is used to verify the phrase spotting rules and deal with any gross errors in the conditional probabilities – a largely manual process (except, see Section 4.6). Whereas, for any of the above automated systems we can easily run multiple trials, e.g., a 10-fold cross-validation, this is not feasible for REMIND. The domain knowledge (DK) has been developed on the training set (40% of the patients), and cannot be “*forgotten*” and re-developed for multiple runs (at least, not without considerable effort). To ensure like comparisons, all algorithms are trained and tested on the exact same train/test splits used for REMIND.

### 4.3 Experiments: Compare Classifiers

This first set of experiments compares REMIND with various mining algorithms. Whereas REMIND can perform inference directly from text documents and a database, most classifiers need the data to be in a standard attribute vector representation. For this first set of experiments we restrict ourselves just to unstructured data: all labeling is done based solely on the doctors’ dictations.

There are (at least) two potential ways to label patient visits and patients from documents. One is to use the entire document as input to the classifier (as we do with the bag-of-words text miner). Another alternative is to use natural language processing methods to extract features from each document, and label the visit (or patient) based on these features. As traditional classification algorithms accept only fixed length feature vectors, we converted the observations from text that are used by REMIND into a vector that encodes how many times each pattern of interest occurred in the document. The classifiers are trained on the 1506 visits for the 138 patients in the training set. Table 1 shows classification results with false positives, false negatives and error rates for the 2181 patient visits in the test set. 480 of the 2181 visits are positive (recurrent).

As expected in data of this complexity, 1-nearest neighbor performs poorly, but stabilizes quickly for  $k \geq 3$ . On the other hand, we were initially pleasantly surprised by RAINBOW’s performance. Doctors typically state diagnoses as negations (“there is NO evidence of metastatic cancer”), and bag-of-words loses this information (either removing negations as stop words or losing word position data). On inspection we found that many supplementary words, (e.g., “palliative”) including many not in our phrase spotting rules were used for document discrimination. (RAINBOW is marginally better with stop words, and those are the results reported.)

Table 1 shows that REMIND’s performance is superior to the other systems. Note that the observations provided to REMIND after phrase spotting have the same information as the 68-feature vector provided to all the classification algorithms (except RAINBOW). The additional information for REMIND is the

Bayesian network, the temporal constraints imposed by the legal state sequences, and the dwell time and transitional probabilities derived from the survival curves in medical literature.

**Table 1. Visit classification for 2181 test set patient visits**

Algorithm	FP	FN	Errors	Error %
Naïve Bayes	50	231	281	12.9
SVM	32	273	305	14.0
1-NN	661	131	792	36.3
3-NN	67	284	351	16.1
5-NN	54	281	335	15.4
7-NN	33	294	327	15.0
9-NN	27	307	334	15.3
RAINBOW	126	166	292	13.4
REMIND	102	40	142	<b>6.5</b>

We also used these algorithms to classify the patient as recurrent or not (i.e., final state=1?) based upon just the text data. For the classification algorithms we tried voting schemes based upon all visits, or just the final visit – the best performance was with a single 68-feature vector for each patient, produced by adding the 68-feature vectors for each visit. Similarly, for RAINBOW, we present the performance with one large document per patient, produced by concatenating all the documents for that patient. REMIND’s results for classifying visits came from the same run that produced the visit classification. The state label of the final state was REMIND’s classification for each patient. As before we trained on the 138 patients in the train set. Table 2 shows the classification results on the 206 test set patients, of whom 58 are recurrent.

**Table 2. Patient classification for 206 test set patients**

Algorithm	FP	FN	Errors	Error %
Naïve Bayes	19	3	22	10.7
SVM	6	17	23	11.2
3-NN	2	26	28	15.6
BOW	23	18	41	19.9
REMIND	5	10	15	7.3

Here we present only the best results for *k*-nearest neighbor (as in Table 1, performance is fairly similar for values of *k*, except  $k=1$ ). RAINBOW’s performance degrades in comparison to visit classification. This reason for this may be that most recurrent patients typically have initial visits where they are not recurrent. Thus, when we concatenate all dictations the set of words that help identify non-recurrent visits may be less useful in identifying non-recurrent patients. (Also there are only 138 training patients vs. 1506 training visits.) As in Table 1, the 2<sup>nd</sup>-best performance comes from the Naïve Bayes classifier built upon the phrase-spotting feature vector used by REMIND.

### 4.4 Experiments with different data sources

One of the motivating factors for REMIND is that critical information is often recorded redundantly in patient records. For

**Table 3. Visit Classification with different data sources**

	Text	DB-Meds	DB-Recur	FP	FN	Errors	Err %
REMIND	X			99	48	147	6.7
	X	X		102	40	142	6.5
	X	X	X	51	41	92	4.2

example, any of these observations can influence probability of concluding recurrence: statements in the dictations about recurrence, pain, or side-effects; structured clinical data about chemotherapy drugs (might be for another cancer) or procedures (colonoscopy or surgery), billing information about referral to specialist, etc. Table 3 details the results for inferring the patient visit classification, from different data sources.

Table 3:Row 1 is the same as the REMIND entry in Table 1; only the text documents are used as patient data. Table 3:Row 2 shows that adding in chemotherapy information from the Medication database reduces the number of errors by 5. Finally adding information from OPIS’s own table for tracking recurrence gets the best performance on the test data: almost 96% accuracy.

**Table 4. Patient Classification with different data sources**

	Text	DB-Meds	DB-Recur	FP	FN	Errors	Err %
REMIND	X			5	10	15	7.3
	X	X		5	8	13	6.3
	X	X	X	6	4	10	4.9

Table 4 shows that REMIND’s performance for classifying patient visits also improves as additional data sources are provided.

### 4.5 Improvements in Recording Outcomes

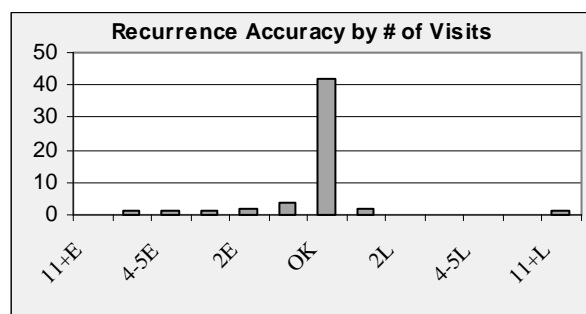
The final rows in Tables 3 and 4 represent the best results for REMIND (or indeed any algorithm) on the 206 test set patients. OPIS’s own DB-Recur table is supposed to track patient outcome. Table 5 shows how REMIND’s classification is significantly superior to the structured data recorded in OPIS.

**Table 5. Outcomes from REMIND and OPIS**

	Pred Recur rent	False +ve	Pred Non Recur	False -ve	False +ve rate	False -ve rate
Truth	58	0	148	0	0	0
REMIND	60	6	146	4	4%	7%
DB-Recur	37	1	169	22	<1%	35%

Table 5:Row 1 shows the results of 3 months of painstaking chart review by a cancer specialist. A glance at Table 5:Row 3 confirms the need for this effort – OPIS’s own DB-Recur table, which is supposed to track this crucial outcome, is filled in correctly for just 62% of the recurrent patients (37/58). If REMIND’s classification were used (thus saving the manual chart review), the number of total errors in OPIS would reduce from 23 to 10.

We note here that at least 4 of our classification errors were unavoidable. In these cases, the specialist’s declaration was based



**Figure 2: Timing Error histogram on 206 test patients**

on paper records (doctors’ handwritten notes from patient visits before 1994, which is when CCO began transcribing dictations).

We also assess REMIND’s accuracy for sequence prediction. Table 5:Row 2 shows that REMIND correctly classifies 54/58 recurrent patients. Figure 2 displays histograms that provide the distribution of errors in the time of recurrence for patients that were actually recurrent and declared recurrent by REMIND. The X-axis is the number of visits that our prediction missed by (“3E” is “3 visits early”, and “2L” is “2 visits late”). For 42 recurrent patients (out of 54) REMIND predicts the exact time of recurrence. Including correct predictions for 142/146 non-recurrent patients, REMIND predicts 89.3% of the patient sequences with complete accuracy.

### 4.6 Integrating mining into REMIND

For the REMIND results shown above, all the domain knowledge (DK) has been manually derived. Notably, REMIND’s extraction from unstructured text is based upon phrase spotting domain knowledge that is provided by physicians (and verified on the training set). However, RAINBOW’s better-than-expected performance in classifying visits (see Table 1) provides an opportunity to automatically mine the dictations in the training set to learn the phrase extraction DK used by REMIND.

In these experiments, instead of using phrase spotting to generate probabilistic observations for REMIND, we simply use the RAINBOW’s classification of the patient visit as the only probabilistic observation for that visit. Table 6 shows preliminary results of plugging RAINBOW into the REMIND framework for the 206 test patients.

**Table 6. Text mining in REMIND for 206 test set patients**

Algorithm	Data Used	FP	FN	Errors	Error %
BOW	Text only	23	18	41	19.9
BOW + REMIND	Text only	5	17	22	10.7
	Text + DB	5	9	14	6.8
REMIND	Text + DB	6	4	10	4.9

Table 6: Row 1 is identical to the RAINBOW results in Table 2 (i.e., using just bag of words to classify patients). Row (2) shows that using REMIND’s temporal reasoning over these observations significantly reduces the errors by a factor of almost 2. (This is consistent with the observations in Tables 1 and 2, which show that incorporating temporal constraints significantly improves classification. Now using REMIND we can also incorporate the

structured clinical data into the classification. Table 6:Row (3) shows that adding this information further reduces errors by about one-third. Although our preliminary results are not quite as good as the best REMIND results (replicated in Table 6:Row 4, from Table 4), this is an exciting area of research that is worthy of further exploration. Table 7 shows similar results for classifying patient visits.

**Table 7. Text Mining in REMIND for 2181 patient visits**

Algorithm	Data Used	FP	FN	Errors	Error %
BOW	Text only	126	166	292	13.4
BOW + REMIND	Text only	67	122	189	8.7
	Text + DB	61	55	116	5.3
REMIND	Text + DB	51	41	92	4.2

## 5. EXPERIMENTS: AMI

In this section we present preliminary results on an AMI (Acute Myocardial Infarction) cohort. AMI (or heart attack) is one of the leading causes of sudden death in the Western world, and is the focus of many quality initiatives. Though we have 1000 patient records available, these results are reported on just 52 patients (the total number for whom the ground truth is known).

In these set of experiments, we compare REMIND’s accuracy in generating clinical diagnoses against the diagnoses assigned by the coders in the hospital’s billing department. We also show that by using REMIND’s code, (at least for this small sample) the hospital can get a much better idea of the impact of clinical factors on financial outcomes.

### 5.1 Data Sources and Domain Knowledge

#### 5.1.1 Data sources at UPMC

The patient cohort is selected randomly from over 1000 patients admitted to the Intensive Care Unit (ICU) of the University of Pittsburgh Medical Center (UPMC) in 2001, whose principal billing diagnosis was 410.xx: this corresponds to a clinical diagnosis of acute myocardial infarction (AMI). This billing diagnosis is assigned by coders in the hospital billing department after manually reviewing the patient chart (after discharge) to bill the patient’s insurance company. (These are coded using an internationally accepted standard for diagnoses, the ICD-9 codes.)

The Medical Archival System (MARS) at UPMC is the source of all patient data. *Structured* patient data from MARS includes patient demographic information (age, sex), length of stay in the ICU, all ICD-9 diagnoses codes, procedure codes, lab and pharmacy data. MARS also has *unstructured* doctors’ dictations including the discharge slip, progress notes from all doctors’ visits in the ICU, and cardiologists’ reports based on EKG waveforms.

#### 5.1.2 Ground Truth

A physician manually reviewed the 52 patient charts to answer 3 specific questions:

1. Diagnosis: Did the patient really have an AMI? It was determined that 9 of 52 patients (all of whom were coded as having had an AMI) did not have an AMI.
2. Co-morbidities: Did the patient have any related condition (e.g., diabetes) which could impact his treatment / outcome? 17 of the 52 patients were diabetic.

3. Process: Of the diabetic AMI’s, how many were treated according to medically accepted standards?

#### 5.1.3 Domain Knowledge

The diagnosis of AMI was established according to the MONICA [30] criteria (International Consensus Conference on Monitoring of Trends and Determinant of Cardiovascular Disease, Augsburg 1995). Variables taken into consideration included the presence or absence of changes typical of AMI in three categories: clinical (symptoms that fit into the category of “cardiac pain”), biochemical (increases in levels of enzymes in blood reflecting myocardial injury) and electrocardiographic (wave forms of abnormal electrical activity diagnostic of myocardial infarction). In each of these categories, changes were classified as abnormal, equivocal and normal, depending on strictly identified criteria. The final diagnosis was arrived at according to the preponderance of the combined evidence from the three categories, as defined by the MONICA criteria. Note that arriving at this diagnosis involves extracting and combining information from structured (lab) and unstructured (patient history, EKG reports) data sources. Similarly, the diagnosis of diabetes mellitus (DM) was established by combining information from lab and pharmacy records, with corroborating evidence from the patient history.

#### 5.1.4 Experimental Setup

We developed and verified phrase spotting rules and the MONICA criterion on 25 patients. Then REMIND was run on the remaining 27 patients. Because there was no significant difference in REMIND’s performance, all experimental results are presented on the entire data set.

## 5.2 Experimental Results

Table 8 compares the hospital billing codes with Ground Truth, and also REMIND with ground truth for diagnosis of AMI and Diabetes Mellitus (DM). Whereas the diagnostic accuracy of the coded information is only 83% for AMI and 90% for DM, results based on REMIND are much closer to Ground Truth (90% and 95%, respectively). Thus, of the 52 patients coded as AMI, only 43 actually fit the MONICA criteria for AMI (Definite, Probable or Possible). In comparison, REMIND correctly identifies 8 of the 9 patients with No AMI. Of the 52 patients, 19 had diabetes, based on the Ground Truth. REMIND makes only one diagnostic error, compared to 5 in the coded information.

**Table 8. Accuracy for AMI & Diabetes for 52 patients**

Diagnosis	Hospital CODES			REMIND		
	FP	FN	Acc	FP	FN	Acc
AMI	0	9	83%	1	2	94%
DM	1	4	90%	0	1	98%

Next we compare the impact of incorrect coding on two key financial outcomes: Length of Stay (LOS) and Charges. LOS derived from coded information in all 52 patients coded, as having an AMI is about 0.5 days less than the Ground Truth. (This is because 9 patients who actually don’t have an AMI, but have been incorrectly coded as having an AMI, are included in computing the Average LOS) Table 9 shows that using the diagnosis extracted by REMIND achieves much greater accuracy, being only 0.1 days off the truth. Similarly, coded information

leads to an underestimation of charges incurred in AMI patients by about \$5000, whereas REMIND is only off by ~ \$1500.

**Table 9. Outcomes on AMI patients**

Outcomes	CODERS	Ground Truth	REMIND
LOS (days)	7.54	7.93	8.05
Charges (\$)	\$89673	\$94688	\$96379

The errors in coded information regarding AMI and DM compound the underestimation of both LOS and Charges in diabetics with AMI. Thus, coded information would lead to the conclusion that LOS for diabetics was 0.6 days less than for non-diabetics, and charges incurred were lower by ~\$26,000. In actual fact (Ground Truth), diabetics stayed an average of ~4.5 days **longer**, and incurred an additional ~\$15,000 in **extra** charges. REMIND was much closer to Ground Truth, correctly identifying that diabetics both stayed longer (by ~5 days), and incurred higher charges (by ~\$21,000) (In addition to the above, REMIND enabled the identification of patients with diabetes who achieved good, moderate, or fair control of their blood sugars during their hospital stay and in the immediate peri-event period of the AMI.)

**Table 10. Impact of diabetes on Financial Outcomes**

Outcomes	Patient-type	CODERS	Truth	REMIND
LOS (days)	Diabetics	7.13	11.00	11.67
	Non-diabetics	7.70	6.60	6.60
Charges (\$)	Diabetics	70,854	105,100	114,887
	Non-diabetics	97,302	90,175	88,976

Table 10 demonstrates the value of REMIND in correctly identifying specific diagnostic categories of patients for outcomes research. They also show the hazards of plotting cost-saving strategies and resource allocations based on electronically coded information. For instance, the results for LOS and Charges based on the coded information would lead to the conclusion that patients with diabetes do not pose a problem with LOS or Charges, and that no specific resource allocations need be made for the care of diabetics with AMI. Ground Truth revealed exactly the opposite. This establishes the utility of REMIND, which paralleled Ground Truth, in correctly identifying and analyzing outcomes in a large cohort.

## 6. REVIEW OF RELATED WORK

Our work draws heavily on earlier work on Bayesian networks and graphical models (see [12] for an overview). Here we briefly list some temporal reasoning methods in medicine (see [6] for an extensive bibliography) that are similar to REMIND in some aspects. Ngo et al [23] describe a temporal probabilistic reasoning method via context-sensitive model construction. Bellazi et al [2] describe a system that uses a Dynamic Bayesian Network to analyze the blood glucose level of a patient over a time interval. Kayaalp et al [15] use structured information to predict probabilities of survival for ICU patients. Other related research [10][14][16] deals with representing temporal data and enforcing temporal integrity.

Clearly, far superior methods to phrase spotting may be used to extract information from free text. Our research focus is the combination of extracted evidence, which is why we used a simple method. Clearly, using any of a standard part-of-speech tagger, better natural language parsing methods, or augmenting the aliases with a lexical reference [7] should improve performance. Taira et al [28] have done research on automatic structuring of radiology reports (which is of direct relevance for the illustrative example in Section 1). REMIND is implemented so that other extraction (and reasoning) methods can be easily plugged into REMIND (see Section 4.6).

REMIND can clearly benefit from using better Natural Language Processing methods [17] than phrase spotting. Of direct relevance is the analysis of doctors' dictations by Chapman [5] which identifies the 7 most common uses of negation in doctors' dictations. Augmenting our aliases with a general lexical reference [7] or a medical language dictionary (SNOMED) should improve performance. Furthermore text-mining research to identify relevant documents[22] may help eliminate irrelevant documents that are mixed in with doctors' dictations.

DISCOTEX [21], like REMIND, extracts information from text, and integrates it via data mining. DISCOTEX focuses on learning rules, whereas REMIND uses domain knowledge for data mining.

## 7. SUMMARY

REMIND exploits the inherent redundancy in medical patient records. Despite the presence of incorrect observations, by performing inference over the entire set of observations and combining them with the domain knowledge (DK), REMIND can reach the correct conclusion.

In our implementation, the parameters of the model were obtained from domain experts. Section 4.6 shows that we can mine the training set to learn the extraction knowledge to replace the phrase spotting DK used by REMIND. It should also be possible to mine the training data set to automatically learn other DK: for instance, the conditional probabilities in the Bayesian network [9], its structure, and the dwell times and transitional probabilities.

The manual process of acquiring DK is not particularly time-consuming. We have investigated the sensitivity of REMIND's performance with respect to model parameters (e.g., by varying the confidences attached to text extraction rules, and by performing rule ablation experiments). REMIND is relatively insensitive to the exact parameter values indicating that these can be acquired relatively easily. We anticipate 2-4 weeks for each new application. This time will further reduce as we gain more experience, and develop tools to help verify and learn new DK.

Furthermore, once the DK is acquired it can be used repeatedly at the same hospital without additional effort. For instance, the UPMC study gives us access to 3000 patient records for AMI alone, for just the last 2 years. Once we build a structured database for 3000 AMI patients, not only can it be used for outcomes and quality analysis at UPMC, it can be further mined to learn patterns about variations in treatment, and their impact.

The real bottleneck for REMIND is not the DK – the architecture is designed so that core REMIND functionality and DK can be transferred to different hospitals. Interfacing to patient records in different hospital IT systems (some are home grown) is arduous.

The two applications test different aspects of REMIND. For colon cancer, our task is to determine when the doctor diagnosed recurrence – not when the patient actually recurred, or could have been diagnosed, but to identify the time the doctor diagnosed recurrence (treatment can only begin after diagnosis). For AMI, our task is to find the correct diagnosis from independent clinical factors, and in fact, to correct any errors in coding the diagnosis.

In conclusion, we have proposed a general framework for performing Bayesian inference on random processes that are sampled at arbitrary time instants by integrating information from structured and unstructured data sources. We have also demonstrated a successful application of REMIND for extracting structured information for stage III colon cancer patients and for AMI patients.

## 8. REFERENCES

- [1] Beahrs, O.H., Henson, D.E., Hutter, R.V.P., Kennedy, B.J. Handbook for Staging of Cancer. Lippincott, Philadelphia, 1993.
- [2] Bellazzi, R., Larizza, C., De Nicolao, G., Riva, A., Stefanelli, M. Mining biomedical time series by combining structural analysis and temporal abstractions. Journal of AMIA (symposium supplement), vol. 5 (1998), 160-164.
- [3] Burges, C.J.C., A tutorial on support vector machines for pattern recognition. Data Mining and Knowledge Discovery, 1998 2(2):955-974.
- [4] Cancer Care Ontario. OPIS, The Oncology Patient Information System, Version 4.3. Toronto, 1999.
- [5] Chapman, W., Bridewell W., Hanbury P., Cooper, G., Buchanan, B.G. Evaluation of Negation Phrases in Narrative Clinical Reports. Proceedings of AMIA, 2001.
- [6] Combi, C., Shahar, Y. Temporal Reasoning and Temporal Data Maintenance in Medicine: Issues and Challenges. Computers in Biology and Medicine 27(5), 1997, 353-368.
- [7] Fellbaum, C., WordNet: An Electronic Lexical Database. MIT Press, May 1998.
- [8] Fleming, I.D., Cooper, J.S., Henson, D.E., Hutter, R.V.P., Kennedy, B.J., Murphy, G.P., O'Sullivan, B., Yarbro, J.W. AJCC Cancer Staging Manual. Lippincott, Philadelphia, 1997.
- [9] Heckerman, D. A tutorial on learning with Bayesian networks. Microsoft Research tech. report, MSR-TR-95-06, 1996.
- [10] Horn, W., Miksch, S., Egghart, G., Popow, C., Paky, F. Effective Data Validation of High Frequency Data: Time-Point, Time-Interval, and Trend-Based Methods. Computers in Biology and Medicine, 1997.
- [11] Joint Commission on Accreditation of Healthcare Organizations (JCAHO) <http://www.jcaho.org>.
- [12] Jensen, F.V. An introduction to Bayesian Networks. UCL Press, 1996.
- [13] Johnson, D.B., Taira, R.K, Zhou, W., Goldin, J.G., Aberle, D.R., Hyperad: Augmenting and visualizing free text radiology reports. RadioGraphics 1998; 18:507-515.
- [14] Kahn, M., Fagan, L., Tu, S. Extensions to the Time-Oriented Database Model to Support Temporal Reasoning in Expert Medical Systems. Methods of Information in Medicine, 1991, 30, 4-14.
- [15] Kayaalp, M., Cooper, G. F., Clermont, G. Predicting ICU Mortality: A Comparison of Stationary and Nonstationary Temporal Models. Proc. AMIA Symposium, 2000, 418-422.
- [16] Larizza, C., Moglia, A., Stefanelli, M. M-HTP: A System for Monitoring Heart Transplant Patients. Artificial Intelligence in Medicine, 1992, 4, 111-126.
- [17] Manning, C.D., Schütze, H. Foundations of Statistical Natural Language Processing, MIT Press, 1999.
- [18] McCallum, A., Freitag, D., Pereira, F. Maximum Entropy Markov Models for Information Extraction and Segmentation. Proceedings of ICML 2000.
- [19] McCallum, A.K., BOW: A toolkit for statistical language modeling, text retrieval, classification and clustering. <http://www.cs.cmu.edu/~mccallum/bow>. 1996.
- [20] Mitchell, T., Machine Learning. McGraw Hill, 1997.
- [21] Nahm, U.Y., Mooney, R.J. A Mutual Beneficial Integration of Data Mining and Information Extraction. Proceedings of AAAI 2000, 627-632.
- [22] Nigam, K., McCallum, A., Thrun, S., Mitchell, T. Learning to Classify Text from Labeled and Unlabeled Documents. Proceedings of AAAI 1998.
- [23] Ngo, L., Haddawy, P., Krieger, R.A., Helwig, J. Efficient Temporal Probabilistic Reasoning via Context-Sensitive Model Construction. Computers in Biology and Medicine, 1997.
- [24] Rabiner, R.L. A Tutorial on Hidden Markov Models and Selected Applications in Speech recognition. Proceedings of the IEEE, 77(2), 257-286.
- [25] Rao, R. B., Towell, G.G., Miller, M., Schmucking, I., Germond, C. Data Mining for Disease Management: Adding value to patient records, Electromedica 2000, 68, 63-67
- [26] Rao, R.B, Sandilya, S, Niculescu R, Germond C., Geol A, Mining Time-Dependent Outcomes from Patient Records, Proceedings of AMIA, 2002.
- [27] Sobin, L.H., Wittekind, C., TNM Classification of malignant tumours, 5<sup>th</sup> ed. Wiley-Liss, New York, 1997.
- [28] Taira, R., Soderland, S., Jakobovits, R, Automatic Structuring of Radiology Free Text Reports. RadioGraphics, 2001 21: 237-245
- [29] Tsui F-C, *et al.* Data, Network, and Application: Technical Description of the Utah RODS Winter Olympic Bio-surveillance System. Proceedings of AMIA, 2002.
- [30] Tunstall-Pedoe H., The World Health Organization MONICA Project (Monitoring Trends and Determinants in Cardiovascular Disease): A major international collaboration, J Clinical Epidemiology 1988, 41, 105-14.