An Error Detecting and Tagging Framework for Reducing Data Entry Errors in Electronic Medical Records (EMR) System

Yuan Ling, Yuan An
College of Computing and Informatics, Drexel University
Philadelphia, USA
{yl638, ya45}@drexel.edu

Mengwen Liu, Xiaohua Hu
College of Computing and Informatics, Drexel University
Philadelphia, USA
{ml943, xh29}@drexel.edu

Abstract—we develop an error detecting and tagging framework for reducing data entry errors in Electronic Medical Records (EMR) systems. We propose a taxonomy of data errors with three levels: Incorrect Format and Missing error, Out of Range error, and Inconsistent error. We aim to address the challenging problem of detecting erroneous input values that look statistically normal but are abnormal in medical sense. Detecting such an error needs to take patient medical history and population data into consideration. In particular, we propose a probabilistic method based on the assumption that the input value for a field depends on the historical records of this field, and is affected by other fields through dependency relationships. We evaluate our methods using the data collected from an EMR System. The results show that the method is promising for automatic data entry error detection.

Keywords—Data Entry Errors; Electronic Medical Records (EMR); Error Detecting; Error Tagging

I. INTRODUCTION

Currently, large-scale data analysis is conducted in almost every area. Advanced analytical process and data mining techniques have been used to discover valuable information and knowledge from large-scale data. However, most data mining techniques assume that data has already been cleaned [1]. This is not always true in real situations. Data preparation process and data quality issues are critical to reliable data mining results, because data with errors will compromise the credibility of the data mining results. The data lifecycle of an application typically consists of the following steps: data collection, transformation, storage, auditing, cleaning, and analyzing to making decisions [2]. Data errors can be reduced in each step of the data lifecycle. Since data collection is the first step of the lifecycle, reducing data entry errors in the data collection stage is the earliest opportunity to enable data quality.

Error and outlier detection have been studied in various application domains, such as credit card fraud detection[3], network intrusion detection[4], and data cleaning in data warehouse[5]. In general, errors can be classified into three types: point errors, contextual errors, and collective errors [6].

In the healthcare domain, data errors in electronic medical databases have been studied recently [7, 8]. Generally, Electronic Medical Records (EMR) data errors have been classified as incomplete, inaccurate, or inconsistent [9]. More specifically, based on EMR visit notes, the previous study [10] classifies the data entry errors into Inconsistent Information, Miscellaneous Errors, Missing Sections, Incomplete Information, and Incorrect Information [10]. For our case, we conducted a comprehensive literature review. As a result, we summarized different levels of data errors in a more detailed manner. Three levels of data entry errors are proposed as shown in Table I. Example errors are discussed in the following paragraphs.

<table>
<thead>
<tr>
<th>LEVELS OF DATA ENTRY ERRORS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Level 1: Incorrect Format and Missing</td>
</tr>
<tr>
<td>L1-1: Incorrect Format Data</td>
</tr>
<tr>
<td>L1-2: Missing Data</td>
</tr>
<tr>
<td>Level 2: Out of Range</td>
</tr>
<tr>
<td>L2-1: Out of normal range</td>
</tr>
<tr>
<td>L2-2: Out of population trends</td>
</tr>
<tr>
<td>L2-3: Out of personal trend</td>
</tr>
<tr>
<td>Level 3: Inconsistent</td>
</tr>
<tr>
<td>L3-1: Personal Inconsistent</td>
</tr>
<tr>
<td>L3-2: Population Inconsistent</td>
</tr>
</tbody>
</table>

L1-1 errors can be easily controlled with prior knowledge about the data format. Constraints can be set for each field. For example, only numbers are allowed to be entered into blood pressure field. L2-1 error also can be easily controlled as long as there are knowledge about the normal range of each field. For example, height values can be constrained in the range of 0 to 3 meters. Detecting L1-2 errors, however, is a difficult research task, but we can learn to detect missing data according to clinical guidelines and caregiving practices. Once the missing values are detected, we can set constraints to this particular field as we did for L1-1 and L2-1. For L2-2 errors, using statistics for detecting erroneous values for a single attribute is well-studied and robust results can be obtained. For example, L2-2 errors can be calculated using a univariate outlier detection technique called Hampel X84 [11]. In this paper, we do not focus on solving the L1-1 to L2-2 error problems.

The challenge lies in the detection of values that look statistically normal but are abnormal in medical sense like L2-3, L3-1, and L3-2 errors. For example, the weight value of a patient increases to 230lbs from 180lbs in three months. The weight value of 230lbs is in the normal population distribution...
range, but for this particular patient, it is suspicious if he/she has a healthy living style and does not have major medical problems. It is categorized as a L2-3 error. The following illustrates other examples. A record showing a male patient has the status of pregnancy would be considered as a L3-1 error. An obese and diabetic patient with BMI (Body mass index) larger than 35 is likely to have a cardiovascular disease manifested as hypertension and dyslipidemia. If the patient’s blood pressure values are significantly low, but still in the normal range, the blood pressure value would be inconsistent with the BMI value from the population perspective. This signals a L3-2 type of error.

In this paper, we propose an error detecting and tagging mechanism in real time to address these challenges based on historical data and dependency relationships. The entire framework consists of three main components:

1) Find a similar patient group. Error detection is based on historical datasets. For a single patient, the historical datasets should be composed from patients who have equivalent medical situations, clinical conditions, and basic demographic information. Because it is difficult to find a group of patients with exactly the same medical conditions, we detect errors based on a dataset from patients with similar conditions.

2) Learn a probabilistic model. Based on the dataset from a group of similar patients, we learn parameters for our proposed model. The probabilistic model is used to calculate the input value probability for each field.

3) Return a tag. Error tagging is based on results from the probabilistic model. If an input value probability for a field is relatively high, then the value has a higher chance to be a correct value and a “Normal” tag will be returned. Whereas if the probability is relatively low, the tag is “Suspicious”. We use a threshold to decide which tag should be returned.

Our experimental results show that our error detecting and tagging method is promising. The predictive accuracy of the probabilistic model is between 70% and 85% using our tagging method.

Our experimental results show that our error detecting and tagging framework is presented in Section II. The rest of the paper is organized as follows: Section II describes related work. Section III presents our error detecting and tagging framework. Section IV introduces our probabilistic model and error detecting method. Section V presents our evaluation process and experimental results. Section VI presents our conclusions and future research directions.

II. RELATED WORK

Statistical methods have been used to analyze and detect data errors at data cleaning stage [12-14]. However, we need some methods to reduce data entry error in the data collection stage[15].

There are different ways to detect errors and control data quality, including classification based neural networks, Bayesian networks[16], support vector machine, rule-based methods, unsupervised clustering, and instance-based methods. The Usher system is developed to detect errors on form entry fields by using Bayesian network and a graphical model with explicit error modeling[17]. The system includes a probabilistic error model based on a Bayesian network for estimating contextualized error likelihood for each field on a form. The method uses Bayesian network to learn dependency relationships among fields, but it does not take historical records for a particular field into consideration. In this paper, we detect errors based on dependency relationships among fields as well as historical data for a particular field. In addition, we classify patients based on some basic information to cluster them into similar groups before we apply error detection. As our method is probabilistic in nature, selecting the right dataset for estimating the probabilities is critical. For example, it is unreasonable to predict the input value for a person whose age is 50 using a dataset for people whose ages are around 10.

Using tagging mechanism has been shown to decrease certain types of errors [18]. Data entry interfaces can be designed to help improving data quality. We propose a system that classifies the input values of each field as “Suspicious” or “Normal”, and provides this information as feedback to the user. If the “Suspicious” tag occurs, then it reminds the user to check the input value. Consequently, it would help reducing data entry errors.

III. ERROR DETECTING AND TAGGING FRAMEWORK

A. Error Detecting and Tagging Framework

Our error detecting and tagging framework is presented in Fig. 1. A data entry interface contains multiple fields. After a user inputs a value for a field, a tag (“Normal” or “Suspicious”) will be returned.
Here we use a simple example to illustrate the historical data format and the error detecting task. Fig. 2 shows a set of records for 3 different patients. There are five fields for each record: Date (Observation Date), BP_S (Systolic Blood Pressure value), BP_D (Diastolic Blood Pressure value), BMI, and Weight. Our error detecting task is to tell whether the patient with ID = 3 has a normal or suspicious input value for BP_S, BP_D, BMI and Weight at the observation date: 1/13/2010. The feedback is generated based on the former 8 records from the patient with ID = 3, and also the other 11 records from patients with ID = 1 and ID = 2, assuming these three patients have similar conditions.

Based on the data, we will learn dependency relationships among BP_S, BP_D, BMI and Weight. These relationships are captured in a probabilistic graphical model.

**B. Error Detecting Process**

Based on our error detecting and tagging framework, we propose the error detecting process as shown in Fig. 3. A patient is modeled as a set of data fields describing the patient’s demographic information and clinical conditions. A form is used to collect a set of specific types of information during an encounter, for example, blood pressure, weight, etc. For our method, we divide the entire patient information into two categories: independent fields which are considered as fixed during the error detection process and dependent field which will be collected and flagged as “normal” or “suspicious”.

As shown in Fig. 3, our error detecting process consists of the following five steps. First, patient records are categorized into groups based on information about the basic independent fields from the current patient and other historical records using a classification model. Second, we find a similar patient group for the current patient. Third, we build a probabilistic model based on clinical fields and historical records from a group of similar patients. Fourth, we set a threshold for the probability of input value. Finally, based on results from the probabilistic model and the threshold setting, we return a “Normal” or “Suspicious” tag to the current user.

**IV. DATA ENTRY ERROR DETECTING METHOD**

**A. Data Entry Error Detecting Method**

Given a data entry interface \( I \), let \( F = \{ F_1, ..., F_j, ..., F_l \} \) be a set of input fields on the interface \( I \). For example, in Fig. 4, the field for entering the “Weight” value can be represented as \( F_{\text{Weight}} \).

For a single field \( F_i \), we can use a sequence of values \( v = \{ v_1, ..., v_{i-1}, v_i \} \) to represent all historical records for a patient, where \( v_i \) is the current value entered by a user for the field \( F_i \). \{ \bar{v}_1, ..., \bar{v}_{i-1} \} is the historical record sequence for the field \( F_i \) and \( v_{i-1} \) is the value entered for the field \( F_i \) at the \((i-1)\)th time.

We assume that a patient’s current input value for a field depends on historical input records of this field. Abnormal changes of input values comparing with historical records need to get attention. In probability theory, such a sequentially dependent process can be described by a Markov model. According to \( 1^{\text{st}} \) order Markov assumption, the current value for the field \( F_i \) at \( i^{\text{th}} \) time depends only on the value for the same field at \((i-1)^{\text{th}} \) time. Therefore, for a giving input value \( v_i \), we can use the value \( v_{i-1} \) to predict the probability of the value \( v_i \) for the field \( F_i \) at the current time. The probability can be represented as \( q_a(v_i | v_{i-1}) \). Similarly, we can have a probability according to \( 2^{\text{nd}} \) order Markov assumption as \( q_a(v_i | v_{i-1}, v_{i-2}) \). But for a new patient, there are no historical records for value of field \( F_i \). So we can assume that the current value \( v_i \) for the field \( F_i \) at \( i^{\text{th}} \) time does not depend on previous historical data from the patient. Therefore the predicted probability can be represented as \( q_a(v_i) \).

Our goal is to predict the probability of an input value \( v_i \) for the field \( F_i \), which can be represented as \( q_a(v_i) \). Using \( 1^{\text{st}} \) order Markov assumption, \( 2^{\text{nd}} \) order Markov assumption, and a prior probability, we can calculate \( q_a(v_i) \) as in Equation (1):

\[
q_a(v_i | all \_conditions) = \\
\alpha_1 \times q_a(v_i | v_{i-1}, all \_conditions) \\
+ \alpha_2 \times q_a(v_i | v_{i-2}, all \_conditions) \\
+ \alpha_3 \times q_a(v_i | all \_conditions)
\]
“all _conditions” means that the probability is calculated based on a dataset only from patients that have equal demographic and clinical conditions. In the following equations, we will omit “all _conditions” in each term assuming that they have been taken into consideration. The three parameters \( \alpha_1 \), \( \alpha_2 \), and \( \alpha_3 \) are used to balance the three probabilities \([24, \text{ and } \alpha_1 + \alpha_2 + \alpha_3 = 1] \). Given a set of records, we can estimate each part of probability according to Equation (2), Equation (3), and Equation (4), respectively.

\[
g_q(v_j | v_{i-1}, v_{i-2}) = \frac{N(v_j, v_{i-1}, v_{i-2})}{N(v_{i-2}, v_{i-1})}
\]

\[
g_q(v_j | v_{i-1}) = \frac{N(v_j, v_{i-1})}{N(v_{i-1})}
\]

\[
g_q(v_j) = \frac{N(v_j)}{N(all)}
\]

For Equation (2) and Equation (3), \( N(v_{i-2}, v_{i-1}) \) is the occurrence number of input value \( v_{i-2} \) for field \( F_i \) at the \((i-2)\)th time followed by the input value \( v_{i-1} \) for field \( F_i \) at the \((i-1)\)th time for a patient. Similarly, \( N(v_{i-2}, v_{i-1}, v_i) \) is the occurrence number of input value in a sequence of \( \{v_{i-2}, v_{i-1}, v_i\} \) for the \((i-2)\), \((i-1)\), and \(i\)th time for a patient.

For Equation (4), \( N(all) \) is the total number of values for field \( F_i \) for historical records from all similar patients. Similar patients means a group of patients who are similar to the current patient. The similarity is measured in terms of their demographic and clinical conditions such as age, gender, disease conditions, medications, treatments, social activities, self-management, etc. \( N(v_i) \) is the occurrence number of \( v_i \) for field \( F_i \) from all historical records based on all similar patients.

The Equation (1) only uses the historical records from a single field. However, there are dependency relationships among different fields on a data entry interface \( I \). The relationships among different fields can be obtained based on prior knowledge from experts or learned from historical data [17]. The relationships can be learned from historical records with a Bayesian network using the Max-Min Hill-Climbing (MMHC) method [19]. We implemented the method using the R packages: bnlearn [20] and gRain [21]. An example of learned relationships among fields is showed in Fig. 5. A Conditional Probability Table (CPT) will also be learned for the Bayesian Network.

For the example shown in Fig. 5, the value \( v_i \) of \( F_i \) depends on the value \( F_w \) and \( F_s \), so we can represent the probability as \( q_{F_i}(v_i | w, u) \). By adding the probability of \( q_{F_i}(v_i | w, u) \) to the Equation (1), we get the following Equation (5):

\[
q(v_i) = \beta \times q_w(v_i) + \gamma \times q_s(v_i | w, u)
\]

where the two parameters \( \beta \) and \( \gamma \), \( \beta + \gamma = 1 \).

We already get the value of \( q_{F_i}(v_i) \) by Equation (1). We can get the value of \( q_{F}(v_i | w, u) \) from CPT of the learned Bayesian Network as Fig. 5. By plugging in the above values, we compute a final probability \( q(v_i | all _conditions) \).

B. Error Tagging Method

We assume that for each field \( F_i \) the set of possible values is finite and discrete. If the values for a field are continuous, we apply appropriate methods for discretizing them. Let \( F = \{v_1, v_2, \ldots\} \) be the set of possible values for a field \( F_i \). The current input value \( v_i \) at the \(i\)th time is a member of the set. For each possible input values in the set \( v_i \), we can calculate probability based on the Equation (5). We get a tag result for current input value \( v_i \) for the field \( F_i \) as Equation (6).

\[
tag(v_i) = \begin{cases} \text{Normal} & q(v_i) \geq \theta \\ \text{Suspicious} & q(v_i) < \theta \end{cases}
\]

For the threshold value \( \theta \) in the Equation (6), there are methods that can be applied to set the threshold. For example, 80/20 rule, Linear Utility Function [22] and Maximum Likelihood Estimation[23]. In this paper, we will use 80/20 rule to dynamically decide the threshold value. In particular, probability locates before 20% highest values means the corresponding value is “Normal”, otherwise it is “Suspicious”.

V. Evaluation

The previous sections described the theoretical derivations of the model and computational steps. In this section, we present an experiment using a set of limited data for evaluating the performance of the model. We aimed to gain some first-hand practical confidence through a pilot study. The limitation of the data is two-fold. First, we only obtained the data for a small number of people, about 1448. Second, we consider a patient model with a small number of fields including age, sex, BP, weight, and BMI. The experiment is by no means thorough and complete because the real model for describing a patient is far more complicated than just several fields. The purpose of the experiment is to demonstrate the effectiveness of the probabilistic model for error detection taking historical data and inter-field dependency relationships into consideration.

We obtained the data from a health clinic affiliated with Drexel University in Philadelphia, PA, USA. All the data are de-identified for HIPAA (Health Insurance Portability and Accountability Act) compliance. We only obtained records with six fields without any additional clinical information.

A. Data Sets

We selected six fields from databases: Age, Sex, Diastolic Blood Pressure (BP_D) value, Systolic Blood Pressure (BP_S) value, Weight, and Body Mass Index (BMI). We extracted 1448 patients with 12199 records from all these six fields from the EMR system. Because our dataset is limited and patients’ information is insufficient, we cannot guarantee all patients have equal clinical conditions and treatments in this small scale preliminary study. Only Age and Sex are used as basic fields to conduct classification process to find a similar patients group for current patient. Based on the two basic attributes: age and sex. Patients are divided into 14 categories and the numbers of
The average predictive accuracy for all groups is 71%. The predictive power is stable for all these groups, which ranges from 70% to 85%, except for group 1, group 13, and group 14. Their numbers of records are lower than 200, so their predictive accuracy is relatively lower.

We implement Usher’s error model [17] on our dataset. Usher induced the probability of making an error using an error model. The comparative results of Usher and our method is shown in Fig. 8. Usher’s average accuracy is 63.2%, while our method is 71%. Except for group 1 and group 13 in which the numbers of records are too small, our method has better performance than Usher.

Since the number of records and accuracy of group 8 is relative high. We test different sets of parameters on the dataset of group 8. The different parameter sets and corresponding accuracy results are as shown as Table III and Fig. 9. There are 9 different sets of parameters.

As shown in Table III, in Set 1 and Set 2, $\beta = 1$ and $\gamma = 0$, which means that prediction only depends on past historical values of the current single field. In set 3, $\beta = 0$ and $\gamma = 1$, which means that prediction only depends on other fields, but does not depend on historical values of current single field. That is, the result of set 3 is the result of only using Bayesian network. The accuracy results from set 1, 2 and 3 are relatively low. In summary, the accuracy results obtained by the methods purely depending on past historical values of the current single field or on dependency relationships among multiple fields learned from Bayesian network are not as good as the results obtained by combining these two aspects.

In Fig. 9, sets 6, 7, and 8 have relatively higher accuracy. In these four sets, parameters $\beta$ and $\alpha_1$ have higher weight. That means prediction depending on all former 2 stages of historical values of the current single field and dependency relationships among multiple fields improves the accuracy. The set 9 has highest accuracy. But in practice, it is not recommended, because for some small datasets, $\alpha_1$ might equal to 0. The recommended parameter sets are Set 6, 7, and 8.

We also conduct an experiment to compare the performance of our model to that of the Hampel X84[11]. Based on our method, the “Normal” ranges for a patient’s “BP_D” value can dynamically narrow down to [70, 90] as shown in Fig. 10. a), while the Hampel X84 gave “Normal” to a broader range [45.3, 114.7] as shown in Fig. 10. b) for all population. The narrower range indicates that our method is more sensitive to error detection.
VI. DISCUSSION AND FUTURE WORK

This paper proposes a five-step error detecting and tagging framework for reducing data entry errors that pose significant challenges to existing statistic methods. Our probabilistic model for error detecting is based on the assumption that input values for a field can be predicted using historical data and inter-field dependency relationships among fields.

We conducted pilot experiments to evaluate our method. Results show that our probabilistic model has better accuracy than a Bayesian network, it is also better than the method based on a Markov assumption that the input value for a field is based only on historical records of this single field. We also compare our method with Usher’s error model. The result shows our method has a better performance. Moreover, our method has more precise results than the Hampel X84 method.

Our future research work will focus on following aspects:

1) Currently, our tagging method can only return two kinds of tags as “Normal” and “Suspicious”. We would like to extend our model to detect multiple errors as shown in Table I.

2) Our error detecting method only focuses on discrete input value. We plan to improve our method to deal with continuous input values.

3) We plan to conduct a more thorough experimental evaluation by collecting more patient data with more attributes in the future.

ACKNOWLEDGMENT

This work is supported in part by a Drexel Jumpstart grant on Health Informatics and the NSF grant IIP 1160960 for the center for visual and decision informatics (CVDI).

REFERENCES