Association for Information Systems
AIS Electronic Library (AISeL)

AMCIS 2006 Proceedings

Americas Conference on Information Systems (AMCIS)

December 2006

Effect of Dirty Data on Free Text Discharge Diagnoses used for Automated ICD-9-CM Coding

Eitel J.M. Lauría
Marist College- Poughkeepsie

Alan March
Universidad del Salvador

Effect of Dirty Data on Free Text Discharge Diagnoses used for Automated ICD-9-CM Coding

Eitel J.M. Lauría
School of Computer Science and Mathematics,
Marist College, Poughkeepsie, NY , USA
Eitel.Lauria@marist.edu

Alan D. March
Schools of Medicine and of Business
Administration, Universidad del Salvador,
Buenos Aires, Argentina
amarch@conceptum.com.ar

ABSTRACT

We discuss data quality issues that emerge when applying text mining classification methods for automated ICD-9-CM coding. In particular our work investigates the extent to which errors in input text data propagate to the classification model. Text classification techniques based on two Bayesian machine learning algorithms (naive Bayes and shrinkage) were applied to a set of free-text outcome diagnoses, which were previously coded using the Spanish Edition of the International Classification of Diseases – Clinical Modification (ICD-9-CM). A measure of predictive accuracy was calculated for each of the text mining algorithms under analysis. Subsequently, the quality of the sample data was incrementally deteriorated by simulating typographical errors in the text. The predictive accuracy was recomputed for each of the dirty samples for comparison purposes. Our results suggest a low impact of errors on the performance of automatic coding by ICD-9-CM.

Keywords

Text mining, machine learning, data quality, medical coding, ICD-9-CM.

INTRODUCTION

In order to benefit the most from Information Technology (IT), medical scenarios require that the large amounts of information which are produced during physician-patient encounters, diagnostic testing and therapeutic procedures be made readily available to computer systems. Researchers have resorted to the manual coding of information contained in medical documents, using a wide variety of approaches. One of the most widespread of such systems is the International Classification of Diseases (ICD) family of classifications, and its adaptation the US produced Clinical Modification (ICD-9-CM), based on World Health Organization's 9th edition.

Coding has often been criticized as a poor way of both organizing and representing medical information. Whereas the latter is true to a certain extent, the former deserves closer analysis. As has been said, coding as it is practiced at present is a manner of classifying and indexing information. As is true to all classification schemes, coding indeed introduces biases in the clustering, organization and presentation of data. In the present state of affairs, and as has already been mentioned above, the better part of "raw" medical information still remains poorly structured or simply expressed in narrative form. Codes represent a secondary abstraction of information otherwise more fully expressed through other means, namely free text.

One of the main problems with medical coding is that it is a time-consuming and expensive process requiring specially trained human resources (Friedman, 2004). Classification schemes such as ICD-9-CM appear as deceptively simple code lists but in reality are complex, rule-based systems for the assignation of one or more codes to well defined units of information, as for example discharge diagnoses or procedures done on patients. Bibliography is ripe with examples of the lower precision and correctness of coding when it is done by untrained personnel. Clinical documents, including discharge summaries, X-ray or pathology reports, problem lists, and other semi-structured "blocks" of clinical information are recorded as free text, and are prone to typographical errors and semantic misinterpretations of ambiguous terms and phrases. Consequently, improving machine readability of available free text information remains the centerpiece of the problem (Lussier, 2004).

Our purpose in this paper is to analyze the predictive power and robustness of Bayesian text classification models used for automatic ICD9-CM coding, under circumstances in which data quality is at issue. Beginning with a dataset of free text discharge diagnosis previously coded by human coders, we simulate common grammatical errors in order to produce a series of "dirty" datasets, where the proportion of errors increased from 10% to 80% of the total word count.

The following section provides an introduction to Bayesian text classification and the two algorithms under consideration (i.e. naive Bayes and shrinkage). The next section describes the experimental setup and reports the results. The paper ends with our conclusions, including future research pointers[1].

BAYESIAN TEXT CLASSIFICATION

The automated classification of free text documents is a classic statistical machine learning problem: a statistical model is created using an algorithm and a training set of free text samples, each of them labeled with a given document class value; the trained model is the tested using a collection of labeled samples to verify the accuracy of the classification method. A variety of statistical machine learning techniques have been proposed for text classification (see Yang, 1999). In this work we concentrate on Bayesian classifiers, specifically naive Bayes and shrinkage-based naive Bayes (McAllum et al, 1999).

Consider a data sample $D = \{d_1, d_2, ..d_N\}$ where each instance $d_n \in D$ is represented by M attributes $X_1, X_2, .., X_M$, and corresponds to a class value $\{c_i\}$, i=1, |C|. Bayesian learners classify instances $d_n \in D$ by computing the posterior probability of each class $P(c_i \mid D) \propto P(D \mid c_i) \cdot P(c_i)$, and assigning the class value that holds the maximum a posteriori (MAP) probability value. The priors $P(c_i)$ can be estimated by computing frequency counts on the sample data set. A naive Bayes classifier simplifies the problem of computing the likelihood $P(D \mid c_i)$ by assuming conditional independence among attributes of the for sample *D*, and thus calculating $P(D \mid c_i)$ as $\prod_{j=1}^{M} \theta_{ij}$, where $\theta_{ij} = P(x_j \mid c_i)$ is the conditional probability of each attribute value $X_j = x_j$ given the class c_i. Each θ_{ij} can be estimated as the relative frequency of training samples belonging to class c_i that carry attribute value $X_j = x_j$. Using a Dirichlet prior probability distribution with parameters $D_i(\alpha_{ij1}, ..., \alpha_{ijr_i})$ to regularize the sample in cases where there are very few or no instances containing pairs $(x_j; c_i)$, the probability estimates are $\hat{\theta}_{ij} = \left(N_{ij} + \alpha_{ij}\right) \Big/ \left(\sum_j N_{ij} + \sum_j \alpha_{ij}\right)$. The values N_{ij} are the number of sample instances for which attribute X_j takes value x_j, and class value is c_i. Parameters α_{ij} can be seen as counts of fictitious cases. Assuming that, for each class c_i, counts are uniformly distributed over the M attributes, $\alpha_{ij} = 1$ and $\sum_j \alpha_{ij} = M$, which results in

$$\hat{\theta}_{ij} = \left(N_{ij} + 1\right) \Big/ \left(\sum_j N_{ij} + M\right) \tag{1}$$

As described by Mitchell (1997), Naive Bayes can be applied to text classification by adding the assumptions that each word in a sample document is an attribute of the sample instance and that its probability of occurrence is independent of its position in the sample document. If vocabulary V denotes the set of all distinct words occurring in all sample documents, and w_k identifies the k_{th} word in vocabulary V, then $P(X_j = w_k \mid c_i) = P(w_k \mid c_i) = \theta_{ik}$. The estimate $\hat{\theta}_{ik}$ is calculated using the following procedure: (i) organize the document sample into a set of concatenated documents $\{\Delta_i\}$, each of which belong to class value $\{c_i\}$, i=1, |C|; (ii) compute the estimate $\hat{\theta}_{ik}$ as $\left(N_{ik} + 1\right) \Big/ \left(\sum_k N_{ik} + |V|\right)$, where N_{ik} is the number of times a word w_k appears in concatenated document Δ_i, and |V| is the size of the vocabulary. Any new document formed by a set of words, is subsequently classified according to the MAP rule:

$$c_{MAP} = \arg\max_{c_i \in C} \left[P(c_i) \times \prod_{q=1}^{|\#\text{ of positions in document}|} P(w_q \mid c_i) \right] \tag{2}$$

[1] Due to space limitations and the fact that this is ongoing research, we settle for just overviews at this time. We apologize if we have inadvertently left out any material desired by prospective readers. However, we will provide complete coverage of these topics at the conference.

Although these assumptions introduce a major simplification, in practice Naive Bayes has proven to perform well when compared with more sophisticated algorithms. See Mitchell (1997), Rish (2001) for more details.

Class Hierarchies and Shrinkage

For classification problems in which the number of classes is large, the estimates $\hat{\theta}_{ik}$ are much less reliable, which in turn affect the accuracy of the naive Bayes classifier. But if the group of classes is organized hierarchically, as in the case of ICD9-CM, the hierarchical structure can be used to compute better probability estimates. Several authors have proposed Bayesian approaches to hierarchical text classification, including Koller & Sahami (1997), and McCallum et al (1999). In this paper we focus on McCallum et al 's shrinkage algorithm.

For each node in a hierarchy tree of r levels, a maximum likelihood estimate without regularization $\hat{\theta}_{ik}^{(h)} = N_{ik}^{(h)} / \sum_{k} N_{ik}^{(h)}$, $h = 1...r$ is computed using all the document samples that belong to that hierarchy level. The estimates along the path discount each child's data from its parent's before computing the parent's estimate, in order to ensure that the maximum likelihood estimates remain independent. A uniform distribution parameter $\hat{\theta}_{ik}^{(0)} = 1/|V|$ is added, to deal with unreliable estimates at the root level due to the presence of uncommon words. The estimate of each leaf node $\hat{\theta}_{ik}$ is then computed by interpolating ("shrinking") its estimate based on the estimates of its *(r+1)* ancestors $\{\hat{\theta}_{ik}^{(0)}, \hat{\theta}_{ik}^{(2)}, ..., \hat{\theta}_{ik}^{(r)}\}$ in the tree path

$$\hat{\theta}_{ik} = \lambda_i^{(0)} \cdot \hat{\theta}_{ik}^{(0)} + \lambda_i^{(1)} \cdot \hat{\theta}_{ik}^{(1)} + ... + \lambda_i^{(r)} \cdot \hat{\theta}_{ik}^{(r)} \quad (3)$$

The interpolation weights $\lambda_i^{(0)}, \lambda_i^{(1)}, ..., \lambda_i^{(r)}$ among the ancestors of class c_i add to 1. The optimal weights $\lambda_i^{(s)}, s = 1...k$ are computed using a simple variation of the EM algorithm. For details of the algorithm see McCallum et al (1999).

EXPERIMENTAL SETUP

Data Source

A set of 11776 free-text outcome diagnoses occurring in 7380 hospitalizations was obtained. The list of discharge diagnoses was obtained from discharge abstracts in which physicians recorded this information as free text phrases. Two experienced coders assigned corresponding codes using the 1999 Spanish Edition of ICD-9-CM. ICD-9-CM is known as a hierarchical coding system due to the fact that the codes which are effectively assigned to diagnosis may be aggregated into blocks of decreasing level of granularity, thus forming a tree-like structure similar to the Internet newsgroup hierarchies. Figure 1 shows a snapshot of the computerized "ICD-Navigator" used by human coders, which illustrates the hierarchical organization of ICD-9-CM. For this study, original codes were aggregated at the third and fourth level of the hierarchy (which roughly correspond to the Section and 3-digit code levels of ICD-9-CM). The 3-level class hierarchy, contained a total of 408 leaf codes, of which only 172 were effectively used in out dataset (that is, these codes were effectively assigned to patients); the 4-level hierarchy included 2687 leaf codes, of which the data set included 651. Ten percent of the document sample (1178 documents) was randomly selected to be used as the test hold out. The remaining 10598 documents were used to train the text classifiers. Both the training and test samples were preprocessed to eliminate excessively frequent words from the free text diagnoses (e.g. articles and prepositions). The test data set was analyzed to check that both the vocabulary and the classes were well represented in the training data set. We verified 79.7% of the words in the test data set vocabulary were present in the training data set vocabulary; and that for both class hierarchies (3-level and 4-level), 97% of the classes in the test data set were present in the training data set.

Dirty Data Simulation

Two kinds of dirty data were considered: (a) free text diagnoses with typos, misspellings or misleading abbreviations; (b) erroneously coded diagnoses (wrong ICD-9-CM codes) due to coders' lack of experience. We chose in this paper to focus on text errors in diagnoses (option a), following these guidelines:

- We standardized the simulated percentage of damage by considering each word in each sample diagnose as a potential target for perturbation (errors were limited to one per word).

- We progressively deteriorated the document sample introducing typographical errors randomly selected among the list of the most common errors in Spanish (commons transpositions and substitutions of letters). We assumed uniform distribution of these errors in the sample.
- We generated 8 dirty data sets with perturbations of 10%, 20%, 30%, 40%, 50%, 60%, 70% and 80% of the cases.

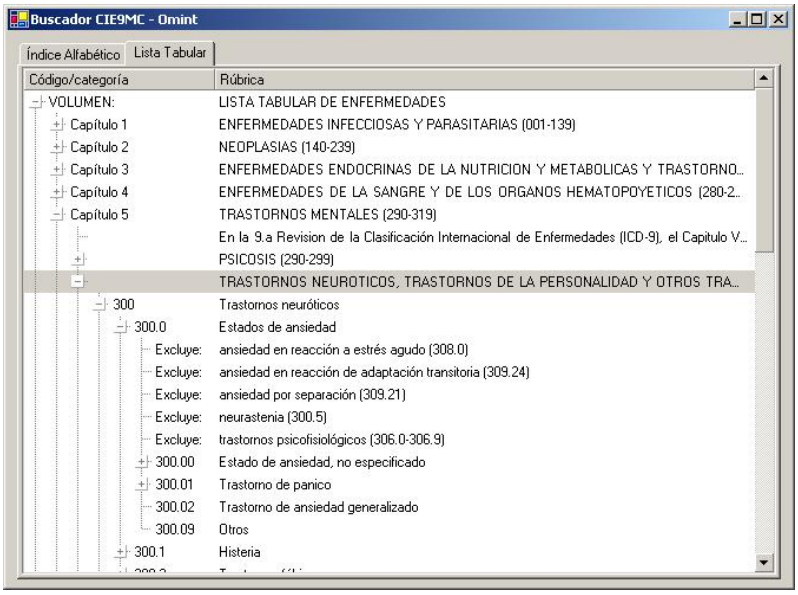

Figure1: Spanish ICD9-CM browser. The successive levels of the hierarchy are easily identified as Volume, Chapter, Section, and 3-, 4- and 5-digit level codes. The authors explored automatic assignation at the Section level, which is usually employed in epidemiological reports.

Model Generation and Performance Evaluation

During the training stage of the experiment, a model was built for every combination of text classifying algorithm (naive Bayes and shrinkage), class hierarchy (3-level and 4-level) and training data set (1 clean data set, 8 simulated dirty data sets), 2 x 2 x 9 = 36 models all in all. The classifiers' performance was measured by calculating their predictive accuracy. Each of the 36 models was tested using the test data holdout and the predictive accuracy was measured as a mean value of the percentage of successful predictions and an error bar given by the standard error (SE).

Results

Preliminary results of the classification analysis are presented in Table 1, and Figure 2. Table 1 shows the assessment of predictive accuracy of the text classifier models, trained with clean and dirty data. The accuracy is provided as a point estimate +/- SE. Figure 2 displays the mean accuracy of the classifiers as a function of the percentage of dirty data.

As expected, the shrinkage algorithm outperforms naive Bayes for all combinations of clean and dirty data set and for both types of class hierarchies (3-level and 4-level).

In the case of the clean data sets, for 3-level hierarchies (172 class classes) naive Bayes' predictive accuracy was 78.41% and Shrinkage reached 85.07%.For 4-level hierarchies (651 classes), naive Bayes' predictions were accurate 67.92% of the times, while Shrinkage made correct predictions in 82.08% of cases.

% of dirty data	3- level hierarchy				4-level hierarchy			
	Naive Bayes		Shrinkage		Naive Bayes		Shrinkage	
	Mean	SE	Mean	SE	Mean	SE	Mean	SE
0%	78.41	1.2	85.07	1.04	67.92	1.36	82.08	1.12
10%	78.24	1.2	84.47	1.06	66.21	1.38	81.66	1.13
20%	77.56	1.22	83.45	1.08	66.3	1.38	80.89	1.15
30%	77.13	1.22	84.13	1.06	65.53	1.38	80.46	1.16
40%	75.68	1.25	84.13	1.06	63.99	1.4	79.61	1.17
50%	75	1.26	82.51	1.11	62.8	1.41	78.84	1.19
60%	74.23	1.27	82.17	1.12	61.43	1.42	78.67	1.19
70%	70.39	1.33	80.8	1.15	57.34	1.44	75.85	1.25
80%	68.17	1.36	79.18	1.18	53.92	1.45	74.32	1.27

Table 1. Predictive Accuracy of Text Classifiers

It is interesting to note that both algorithms prove to be surprisingly robust when subjected to training data with an increasing amount of errors. In the case of the 3-level hierarchy data sets, the shrinkage based models maintained a considerably high level of accuracy which remained practically constant even with training data sets containing 40% of errors and dropping an additional 5% (79.18%) for 80% dirty data sets. For 4-level hierarchies, shrinkage accuracy went from 81.66% to 74.32% as the data was increasingly deteriorated from 10% to 80%.

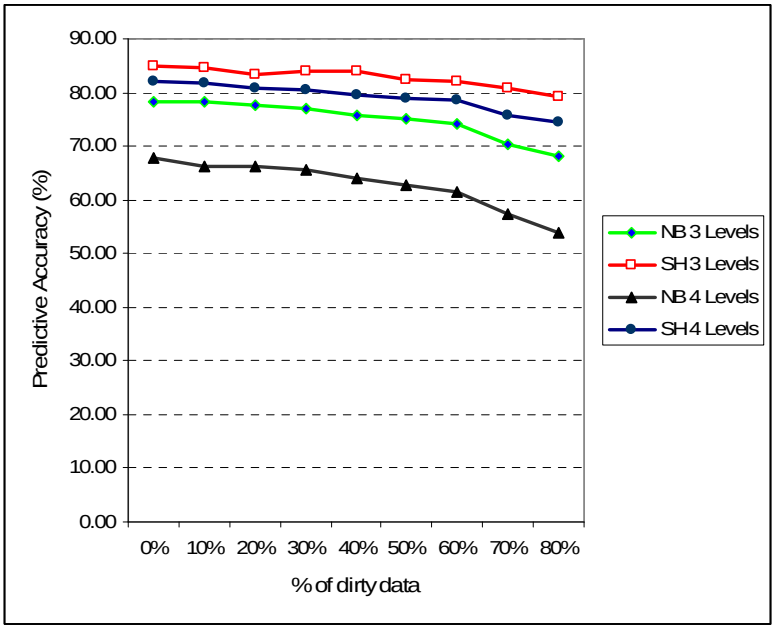

Figure 2. Predictive Accuracy of Text Classifiers

For 3-level hierarchies, Naive Bayes yielded a predictive accuracy of 78.41% with 10% errors , of 75.41% at 40% and of 68.17% at 80%. For 4-level hierarchies, Naive Bayes was 66.21% accurate with 10% errors and 53.92% accurate with 80% errors.

CONCLUSION AND FURTHER RESEARCH

Our present results reinforce our previous conclusions regarding automated coding using statistical language processing methods. The increase in number of cases for both the training and test sets has not altered our previous results with clean data in a significant manner (March et al, 2004). Although our conclusions regarding the influence of "dirty" data are preliminary and require further investigation, a priori, the results suggest that with enough training data and adequate text classification algorithms, the quality of training data is not relevant to guarantee high levels of predictive accuracy.

REFERENCES

1. Friedman C, Shagina L, Lussier Y, et al (2004), Automated encoding of clinical documents based on natural language processing. J Am Med Inform Assoc 11(5):392-402
2. Koller D, Sahami M (1997), Hierarchically classifying documents using very few words. In ICML-97: Proceedings of the Fourteenth International Conference on Machine Learning, pp 170--178.
3. Lussier YA, Friedman C, Shagina L, Eng P (2000), Automating ICD-9-CM encoding using medical language processing: a feasibility study. Proc AMIA Annual Fall Symposium, 1072.
4. March A, Lauría E, Lantos J (2004), Automated ICD9-CM coding employing Bayesian machine learning: a preliminary exploration, Proceedings of SIS2004 (Simposio de Informática y Salud - SADIO), 33rd Conference on Computer Science & Operational Research, Buenos Aires, Argentina.
5. McCallum A, Rosenfeld R, Mitchell T, Ng AY (1998), Improving Text Classification by Shrinkage in a Hierarchy of Classes. In: Proceedings of the Fifteenth International Conference on Machine Learning. Morgan Kaufmann, San Francisco, pp 359-367.
6. Mitchell T (1997), *Machine Learning*, McGraw-Hill
7. Rish I, (2001). An empirical study of the naive Bayes classifier, in IJCAI-01, Workshop on Empirical Methods in AI.
8. Yang Y (1999), An evaluation of statistical approaches to text categorization. Information Retrieval, 1(1-2):69–90