Automated classification of medical-billing data

R. Crandall[1], K.J. Lynagh[2], T. Mehoke[1] and N. Pepper[1,2,*]

[1]Center for Advanced Computation, Reed College, Portland, OR
[2]Qmedtrix Systems Inc., Portland OR
[*]Contact author: peppern@reed.edu

Automated classification of medical-billing data

Abstract. When building a data pipeline to process medical claims there are many instances where automated classification schemes could be used to improve speed and efficiency. Medical bills can be classified by the statutory environment which determines appropriate adjudication of payment disputes. We refer to this classification result as the *adjudication type* of a bill. This classification can be used to determine appropriate payment for medical services.

Using a set of 182,811 medical bills, we develop a procedure to quickly and accurately determine the correct adjudication type. A simple naïve Bayes classifier based on training set class occurrences gives 92.8% accuracy, which can be remarkably improved by instead presenting these probabilities to an artificial neural network, yielding 96.8 ± 0.5 % accuracy.

1 Introduction

In this paper we discuss a medical bill classification methodology. Our exposition focuses on the bill adjudication type, which is but one of many possible classifications that may be desired. This paper starts with a discussion of the motivation for creating this machine learning component of a larger information processing pipeline, followed by a description of the problem and data fields used for prediction. The paper then continues on to an exposition of our bayesian and neural net solutions. Finally we discuss our actual training and testing data and present results and conclusions.

A bill's adjudication type is an important factor in determining proper compensation for medical services. In some domains such as workers' compensation, there are legally mandated fee schedules that dictate the price of medical procedures. These fee schedules vary by state and facility type; a procedure performed by an outpatient hospital may have a different fee schedule rate than the same procedure provided at an ambulatory surgical center. We categorize bills into seven basic adjudication types (table 1).

The reader may be wondering why such a classification scheme is needed when the tax identification number (tin), and thus the corporate identity of the service provider, is already known. While the tin does help narrow down the possible adjudication type it is by no means conclusive. For example a hospital with one tin is likely to have inpatient, outpatient and emergency room services all billed under the same tin and housed at the same street address. To make matters more confusing, many hospitals have free standing surgery centers (asc) near the actual hospital building. While the asc may be a different building and thus subject to a different set of billing rules, it will likely still be considered part of the same company and therefore bill under the same tin. In extreme cases, there are large companies which provide many types of medical services across a disparate geographic area and yet all bill as one company under one tin.

1.1 Motivation for automation

Traditionally, in medical bill review human auditors will determine the adjudication type of a bill manually and then apply the appropriate rules or fee schedules. There are two problems with this approach which an automated solution can help solve.

First, humans are much slower than an optimized machine learning engine. While the engine described in this paper can classify around 14,000 bills per second a human takes anywhere from 10 seconds to 45 seconds to classify a single normal bill. Not only does the algorithm classify bills many orders of magnitude faster than a human, it also saves a large amount of worker time which translates to huge savings and increased efficiency. Additionally, if one were to build a fully automated system for applying fee schedules and rules to medical bills this would eliminate the human bottleneck entirely.

Second, the algorithm can be carefully tuned to reduce errors. Humans tend to make errors when trying to do many detail-oriented tasks rapidly.

While many bills may be classified using an extensive rules-based automated system, such a methodology is inferior to a machine learning approach. Rules-based systems cannot reliably handle edge cases and are labor intensive to modify or update. The genesis of this project was a desire to solve a software engineering problem: accurately route bills through a fraud detection and cost containment pipeline. Initial attempts at solving this problem using manually coded rule sets created error rates that were unacceptable. This lead to much of the classification being redone manually. The machine learning approach presented here was developed for two purposes. First the system allows for improvement of classification accuracy of manual rule sets and manual auditors by examining the output of the machine learning algorithm. Second this system can be integrated as a part of this larger information system.

2 Analog and digital data fields

We have elected to split relevant medical data into two types. First, *analog* (equivalently, numerical) data are those data whose magnitude has meaning. Such data include bill cost (in dollars), duration (in days), and entropy (in bits); that is, an analog datum carries with it a notion of physical units. On the other hand *digital* (equivalently, categorical) data are represented by strings indicating a medical procedure (such as with icd9 codes). These data have no magnitude or natural distance metric.

Now, we refer to an analog or digital datum as a *feature* of a bill. On the simplifying assumption of statistical independence, the probability of a bill with a feature set $f = \{f_1, f_2, f_3, \ldots\}$ having adjudication type A is taken as

$$p(A \mid f) = \prod_i p(A \mid f_i) \;=\; p(A \mid f_1)\, p(A \mid f_2)\, p(A \mid f_3) \cdots \tag{1}$$

One novel aspect of our approach is that all conditional probabilities are inferred from *log-histograms* (equivalently, histograms with exponentially growing bins). Such a transformation effectively reduces the wide dynamic range of some analog data. Cost, for instance, ranges from a few dollars to more than 10^5 dollars.

Another idea—one that tends to stabilize the performance of a classifier—is to "soften". For example, say that a digital datum such as code `123ABC` occurs only *once* during training, and say it occurs for an iph bill. We do not want to conclude that the `123ABC` code occurs only on iph bills in post-training classification. To avoid this kind of spurious statistical inference, we estimate the conditional probabilities as

$$p(A_i \mid f) = \frac{\alpha_i + \#(A_i \mid f)}{\sum_j \big(\alpha_j + \#\big(A_j \mid f\big)\big)}, \tag{2}$$

where α_i is a positive constant and $\#(A_i \mid f)$ is the count of bills with adjudication type A_i having feature f. (In general, the bill type $i \in [0, T-1]$, i.e. we assume T adjudication types.) The degenerate case $\alpha_i = 0$ is the usual probabilistic estimate, but we want to "soften" for the cases such as the singleton

iph bill mentioned. We typically use $\forall i : \alpha_i = 1$, although any set of positive real α_i will prevent singular division. This probability transformation is equivalent to inserting a virtual entry of some certain size into every histogram, and also works effectively for analog-data histograms that have accidentally empty or near-empty bins.

A third probability transformation option applies when using a neural networks; the option *not* to normalize a length-T probability list. Because neural networks can adjust weights as a form of normalization, we can alternatively define

$$p(A_i \mid f) \;=\; \gamma \cdot \#(A_i \mid f)\,, \tag{3}$$

where γ is a positive constant chosen merely to keep these "unconstrained probabilities" within reasonable range. One possible advantage of this loosening of probability constraints is that, for some features, the histogram counts for all $i \in [0, T-1]$ might be so small as to be insignificant. By not normalizing the T vertical histogram sections, we may well reject some system noise in this way. It is a tribute to neural-network methodology that one may use this normalization, which amounts to using *a priori* adjudication frequencies, or not. Of course, the right option may be obvious after cross-validation (training/testing to assess both options).

3 Naïve Bayes

Naïve Bayes is a well known, simple, and fast classification method [4, 2] that treats bill features as statistically independent from one another. Treating item features independently is a strong assumption, but it typically holds in this domain; the codes on a bill for an organ transplant, say, are almost entirely disjoint from the codes on a bill for a broken bone. In the Bayesian approach, a bill is classified according to the most probable adjudication type A_i that *maximizes* likelihood over all relevant feature sets, said likelihood being

$$\prod_f p(A_i \mid f)\,. \tag{4}$$

We estimate the conditional probabilities $p(A_i \mid f_1)$ from histograms on a training set of the data (see Analog and digital data fields). Because the naïve Bayes classifier chooses between relative probabilities, the inclusion of features shared by many of the classes (a blood draw code, say) is effectively a rescaling, and thus does not affect performance. Finally, note that this procedure can incorporate a variety of features for each bill; the number of terms in equation (1) depends on the total number of codes within individual bills.

The naïve Bayes classification is simple to compute, and performs well (see table 2). The solid performance on tin alone is no surprise; 81% of the tins in the training set correspond to a single adjudication type (11% two, 8% three or more). Thus, classifying an unknown bill as having the most common adjudication type from that provider is a good rule of thumb. However, large hospitals

and some hospital networks use a common tin for differently adjudicated medical services, and a good classification scheme must take into account more data. Incorporating bill procedures improves classification performance; combining all the codes gives the naïve Bayesian classifier an accuracy of 92.83%. Note that, even with the tin omitted (classifying on hcpcs, hrc, svc, icd), the accuracy is 86.28%.

4 Neural network

Artificial neural networks are a well-studied family of classifiers [1, 5, 8]. We implemented artificial neural networks with the fann library [7], and trained the network with the conditional probabilities used in the naïve Bayes classifier. See our Appendix for network-architecture parameters.

Unlike the naïve Bayes classifier, which simply multiplies all the conditional probabilities, the neural network is capable of learning nonlinear weighted combinations that improve classification performance. Because neural networks can settle into suboptimal extrema, we perform 10 training runs and average the results. These generalized capabilities resulted in a final classification performance of $96.9 \pm 0.6\%$.

5 Data and methods

Table 1. Possible adjudication types of medical bills. The proportion of bills refers to our entire sample (testing and training sets) of 182811 bills.

Adjudication type	proportion of bills
amb: Ambulance or medical transport	4.86 %
asc: Ambulatory surgery center	20.77 %
dme: Durable medical equipment	11.24 %
er: Emergency room	18.23 %
iph: Inpatient hospital	5.44 %
oph: Outpatient hospital	24.72 %
pro: Professional services	14.73 %

The data used in this procedure consist of medical bills from across the United States in 2000–2009, representing more than a billion dollars of charges. The bills contain both categorical and numeric features. The sets of codes pertaining to individual line items (cpt, ..., 7,527 codes) the patient (diagnostic icd9, ... 6,302 codes), and the provider (11,603 tin) represent more than 25,000 total categorical features.

The bills also contain numeric values: the total bill cost, the duration (length of treatment or overnight stay), and the service cost entropy. This latter quantity

is derived from the line item charges on bills, according to

$$E = \quad -\sum_i \frac{c_i}{c} \log_2 \left(\frac{c_i}{c} \right), \tag{5}$$

where c_i is the line charge for line i and $c = \sum c_i$ is the total bill cost (we take $0 \cdot \log(0) = 0$). Essentially, this quantity E models the distribution of charges; a bill consisting of several same-priced weekly treatments (i.e. a uniform distribution) has a very high entropy E, whereas a hospital stay with charges for meals, medication, and surgery charged together on the same bill (i.e. a very skewed price distribution) gives a relatively low entropy.

The data were split into disjoint training/testing sets of 121,852 and 60,959 bills, respectively. This was done by randomly sampling the data into two buckets of roughly $\frac{1}{3}$ for testing and $\frac{2}{3}$ for training. All results shown reflect the outcome of predicting testing data using models formed with our training data.

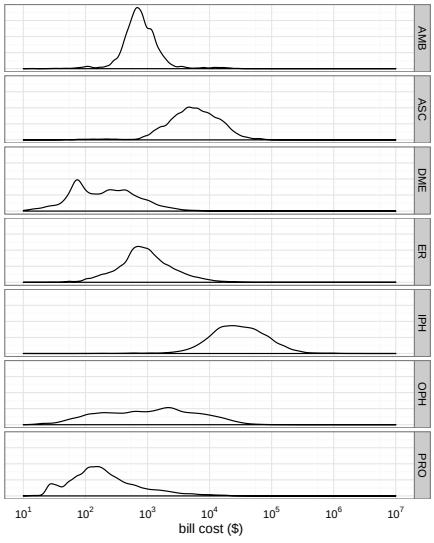

Fig. 1. Bill histograms vs. total bill cost, displayed by adjudication type. It is a key idea that the horizontal axis is nonlinear, to reduce skewness; we use logarithmic abscissa throughout our runs. See our Appendix for density-estimation parameters

6 Results

Table 2. Percentage accuracy of a uniform prior naïve Bayes classifier trained on 121649 bills, tested on 61162 bills for each code type alone.

code	classification accuracy
svc	71.58 %
hrc	55.08 %
ndc	4.83 %
icd	56.49 %
tin	79.03 %
hcpcs	20.26 %

Confusion matrices[6] are used to visualize classification results, this is a standard way to display classification accuracy and error. The true classification of a bill is given across the columns, and the predicted classification is quantified down the rows. A perfect classifier would have nonzero entries only along the diagonal, each corresponding to 100%.

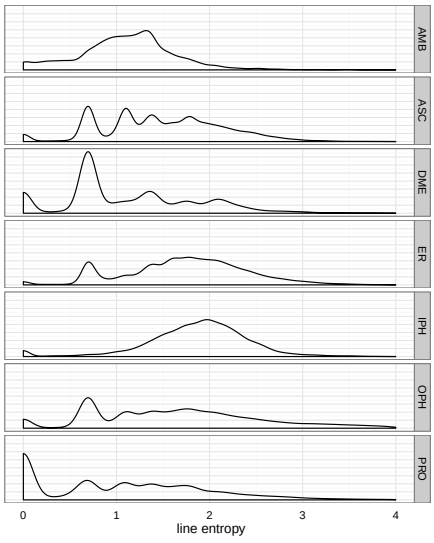

Fig. 2. Bill histograms vs. line cost entropy, displayed by adjudication type. See our Appendix for density-estimation parameters

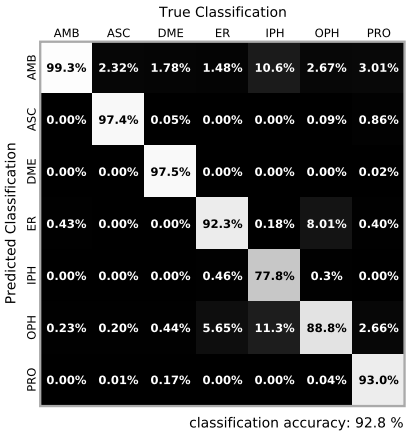

Fig. 3. Confusion matrix from analog-and-digital-data trained Bayes classifier, having 92.8% accuracy.

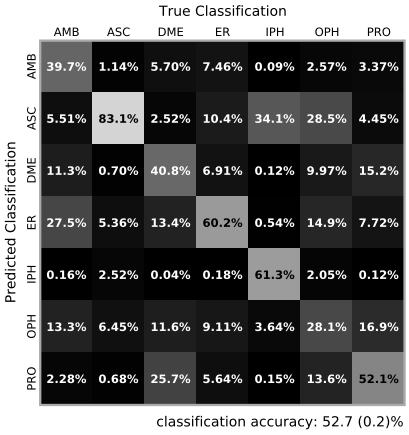

Fig. 4. Confusion matrix from analog-data trained neural network. Average classification accuracy for 10 train/test runs is $52.7 \pm 0.2\%$.

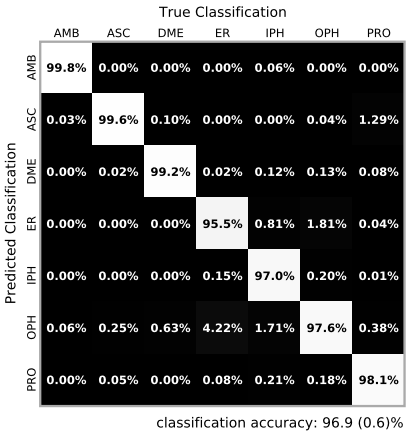

Fig. 5. Confusion matrix from digital-data trained neural network. Average classification accuracy for 10 train/test runs is $96.9 \pm 0.6\%$.

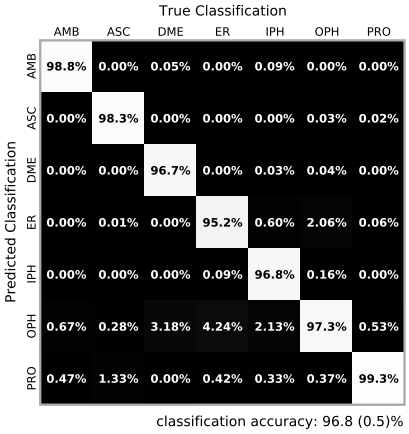

Fig. 6. Confusion matrix from analog-and-digital-data trained neural network. Average classification accuracy for 10 train/test runs is $96.8 \pm 0.5\%$. We take this confusion matrix to be our standard, combining analog/digital data via log-histogram techniques and exploiting the power of neural networks.

7 Conclusions and future directions

Our method for adjudication type classification can be extended in several ways. Note that both the simple Bayesian model and the neural network return a score for each possible adjudication type, and bills are classified according to the maximum. A more nuanced scheme can be constructed with additional domain knowledge to minimize classification risk. For instance, one could classify bills when one output clearly dominates, but put a bill "on hold" for manual review if its several outputs are close to the maximum.

Also note that, though bills can be classified using the analog features, they do not provide additional discriminative power beyond the categorical features (compare figures). For the purposes of fraud detection and proper payment, this is a desirable situation–we can reliably determine a bill's adjudication type (and hence, proper reimbursement amounts) while ignoring the bill amount entirely!

In conclusion, we have theoretically motivated and technically demonstrated the viability of an automated method for medical bill classification. There are many future research directions motivated by the present work:

- Combining various classifiers (neural networks, support vector machines, decision trees, etc.) using voting schemes.
- Using a separate neural network for post-processing, with, say, *reduced* input count based on the analog levels of the first network; in this way, perhaps false categorization can be further reduced.
- Using multiple classification schemes in a cascaded fashion to improve the granularity and/or accuracy of the classifications.
- Placing the classification scheme in a loop with human oracles determining correct classifications of ambiguous bills to continuously increase accuracy.
- Implement a bagging scheme to increase accuracy and mitigate the effects of possibly having incorrect labels in the training data.
- Determine whether our discovered neural-net "amplification" of first-order Bayesian classification is theoretically—or empirically—equivalent to higher-order Bayesian analysis.

Appendix

Histograms

Figures 1 and 2 were generated using the `ggplot2` graphics library with the default parameters. The histograms used for the analog fields each had 10 exponentially growing bins chosen according to the domain of the training set. Analog data in the testing set outside of this range were counted in the first or last bin, as appropriate.

Neural network

Each feature maps to T (in our runs, $T := 7$ adjudication types) input neurons, giving $3T$ inputs for the analog (numeric) features (bill cost, duration, and entropy) and $5T$ for the digital (categorical) features. Note that, since the input

and output vector dimensions must be fixed for a neural network, we multiplied conditional probabilities for bills with multiple features of the same kind (several icd9 codes, for instance). This gave better performance than simple heuristics like using only the most infrequently occurring code of a given type.

All neural networks were seven hidden neuron feed-forward networks, trained by standard reverse propagation. All neurons used the sigmoidal activation function proposed by Elliott [3]:

$$y(x) = \frac{xs}{2(1 + |xs|)} + 0.5,$$

where the positive constant s is the neuron activation steepness.

Robustness

To show that the neural network architecture and our results are robust, we ran each combination of parameters 10 times, and report the average accuracy and its standard deviation (see figure captions). On each run, bills were split into disjoint training and testing sets (121,852 and 60,959 bills, respectively).

Bibliography

[1] C.M. Bishop. *Neural Networks for Pattern Recognition*. Oxford University Press Inc., 2007.

[2] K. Cios, W. Pedrycz, and R. Swiniarski. *Data mining methods for knowledge and discovery*. Kluwer Academic Publishers, 1998.

[3] David L. Elliott. A better activation function for artificial neural networks. Technical report, Institute for systems research, University of Maryland, 1993.

[4] T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning*. Springer, 2009.

[5] Simon Haykin. *Neural networks and learning machines*. Prentice Hall, third edition, 2008.

[6] R Kohavi. Glossary of terms. *Machine Learning*, 30(December):271–274, 1998.

[7] Steffen Nissen. Implementation of a fast artificial neural network library (fann). Technical report, Department of Computer Science University of Copenhagen (diku), 2003.

[8] Andrea Tettamanzi and Marco Tomassini. *Soft computing: integrating evolutionary, neural, and fuzzy systems*. Springer, 2001.