Detecting Adverse Drug Events in Discharge Summaries Using Variations on the Simple Bayes Model

Shyam Visweswaran, MD, MS[1,2], Paul Hanbury[1], Melissa Saul[1], Gregory F. Cooper[1,2], MD, PhD
[1]Center for Biomedical Informatics, [2]Intelligent Systems Program
University of Pittsburgh, Pittsburgh, Pennsylvania

ABSTRACT

Detection and prevention of adverse events and, in particular, adverse drug events (ADEs), is an important problem in health care today. We describe the implementation and evaluation of four variations on the simple Bayes model for identifying ADE-related discharge summaries. Our results show that these probabilistic techniques achieve an ROC curve area of up to 0.77 in correctly determining which patient cases should be assigned an ADE-related ICD-9-CM code. These results suggest a potential for these techniques to contribute to the development of an automated system that helps identify ADEs, as a step toward further understanding and preventing them.

INTRODUCTION

A crucial factor in improving the quality of healthcare is the detection and prevention of adverse drug events (ADEs) [1]. Adverse drug events include non-preventable adverse reactions that occur with appropriate use and dose of medications as well as preventable incidents arising from errors in prescribing, dispensing or administering drugs. Not only are ADEs an important cause of morbidity and mortality in hospitalized and ambulatory patients, they also incur significant expense to the healthcare system. It would be useful to have effective and inexpensive tools for routine identification of ADEs, as a step toward characterizing and preventing them.

Hospitals detect and report regularly only a small fraction of adverse events since they rely mainly on voluntary reporting to identify them. While inexpensive, this approach systematically underestimates the incidence of ADEs. Manual chart review, typically used by researchers, is an effective method for detecting ADEs; however, it is too expensive for routine monitoring in hospitals.

With increasing use of electronic hospital systems that capture patient-related data in electronic form, computerized detection is starting to be employed for identifying ADEs. This method employs computer algorithms to detect signals in data that suggest adverse events. Common sources of coded data in hospital systems include administrative coding of diagnoses and procedures, clinical laboratory results and pharmacy data. Computerized rules and queries have been applied to these structured data to identify diagnoses that could reflect an ADE, abnormal laboratory results, elevated drug levels and medications considered to be antidotes [1]. However, such coded data represent only a small part of the clinical information associated with a patient. Increasingly, narrative clinical reports such as admission notes, progress notes, nursing notes, discharge summaries and test reports like those from radiology and pathology, are available in electronic form. These are a rich source of clinical information for identifying ADEs although their minimally structured, free-text form poses challenges for development of effective algorithms.

Various methods have been applied to the detection of adverse events in free-text medical reports. Simple searching of relevant keywords and phrases can uncover many adverse events but has low specificity [2]. In related work on classification, natural language processing, employing pattern matching and rule-based techniques, has been successfully applied to radiology reports and shown to be as accurate as human coders [3]. Automated techniques for classification of free text documents have been widely studied in the machine learning community. Methods such as decision trees, neural networks, probabilistic networks, support vector machines and nearest-neighbor algorithms have been applied to text categorization [4]. These methods learn models from a training set of labeled cases that are then applied to new unlabeled cases to classify them. Feature selection is often applied to improve the accuracy of classifiers and numerous feature selection methods have been developed.

The *simple Bayes* classifier, sometimes called Naïve-Bayes, is a probabilistic model that makes the assumption that features of a case are conditionally independent of each other given its classification label. The construction of a simple Bayes model suitable for classification is straightforward: (1) Select features from data that are judged to be relevant, and (2) Calculate the model parameters (i.e., the conditional probabilities of the features and the marginal probability of the class variable).

Several methods can be used for the selection of appropriate features in step 1. One approach is to add features, one at a time, to the model and evaluate its performance with a metric like classification accuracy or the area under the Receiver Operating Characteristic (ROC) curve. Only those features that improve the existing model are included in the final model. An alternative to selecting a set of good features is to perform model averaging over all possible simple Bayes structures. For *N* features under consideration, there are 2^N possible Bayes structures over which averaging has to be carried out. Remarkably, for the simple Bayes model, tractable exact model averaging can be performed in the same time and space complexity required to construct a single traditional model. The averaged model is represented by a single simple Bayes structure with the parameters adjusted, such that, the resulting predictions are equivalent to those computed by full model averaging. Empirically, the model-averaged classifier has been shown to outperform the simple model on a number of datasets [5].

BACKGROUND

The Identifying Patient Sets (IPS) system is currently used at the University of Pittsburgh to bcate electronic medical records of interest for clinical research [6]. Given a superset of patient records, it helps the researcher build a simple Bayes model to locate records that are of interest. A preprocessor indexes the entire set of records by the occurrence of all Unified Medical Language System (UMLS) terms and all single words. The user initially labels a few examples of records of interest in IPS, based on which the system derives and lists those terms that distinguish the records of interest from the rest. From this suggested list of terms, the user selects those that are clinically meaningful and constructs a simple Bayes model that IPS applies to the still unreviewed records in order to rank them according to the probability they are of interest. The user can then selectively review those records that are most likely to be of interest.

Recently, we developed and implemented a new IPS module called the Automatic Model Creator (AMC) that facilitates the creation of models in IPS. Users currently utilize AMC to automatically construct models that can be applied to locate records of interest. Given minimal sensitivity and specificity levels acceptable to the user, AMC searches the space of possible models in a greedy fashion and finds those that meet the specified levels.

IPS and AMC have been developed as a general-purpose medical record retrieval system. Here, we focus on one possible application: the identification of ADE-related discharge summaries. We compare the performance of simple Bayes (SB), model-averaged simple Bayes (MASB), simple Bayes with double feature selection (DFS) and AMC. The inputs to these algorithms are a list of terms, where a *term* is a word, a phrase or a UMLS concept. A *feature* refers either to a single term (in SB, MASB, DFS models) or to a disjunction of terms (in AMC models). Only those terms that are ranked highly by IPS are included in the input list; thus, all algorithms start with a limited set of terms. SB is the traditional model that incorporates all terms in the input list as features. MASB performs inference over the power set of simple Bayes models for the set of terms. DFS further selects a subset of terms from the list to construct a simple Bayes model. Each of these algorithms is described in more detail in the next section.

METHODS

Our study is based on 32,702 admissions to a large urban teaching hospital during a 1-year period from July 2001 through June 2002. Of these, 936 were labeled with ICD-9-CM codes E930 - E949 (E-codes) for ADEs. Discharge summaries for 912 of these 936 admissions, plus an additional 1095 admissions chosen at random from the same 1-year period, were extracted as narrative reports stored in one of the hospital's information systems. Multiple discharge summaries for a single admission were merged to generate 877 records indicative of ADEs and 1014 records not indicative of ADEs. To maintain confidentiality, the discharge summaries used in this study were de-identified in accordance with the HIPAA (Health Insurance Portability and Accountability Act) guidelines. De-identification was done using a program called De-ID that replaces identifiable text with specific de-identification tags [6].

For the following experiments, we split the records into a training set and a test set such that each record had a 30% chance of being randomly assigned to the test set. This resulted in 1328 records being assigned to the training set and 563 to the test set, with both sets containing a similar proportion of ADE records. All models were trained using only the training set and evaluated on the test set.

Preprocessing and indexing. The IPS system's preprocessor identifies and indexes all single words and UMLS terms up to 4 words in length that are present in the records. A negation detection algorithm incorporated into the preprocessor identifies and tags pertinent clinical findings and diseases that are negated [7]. IPS ranks the association of the indexed terms with ADE-labeled records by their *likelihood ratio* (*lr*), defined as:

$$lr(term) = \frac{P(term \mid ADE)}{P(term \mid not\ ADE)},$$

where the probabilities are estimated from frequency counts using a Bayesian prior that effectively "smoothes" the estimates.

For all experiments, we excluded terms with *lr* of less than 1.5 as we anticipate they would have little discriminative power. This resulted in an input list with 100 terms that corresponded to the top 100 terms in the ranked list generated by IPS. We preprocessed the records in two ways: 1) We used the existing IPS system's preprocessor to index UMLS terms and single words occurring in 3 or more records (*single-word indexing*), and 2) We modified the preprocessor to index UMLS terms and terms containing 3 words or fewer (i.e., all strings of exactly one, two or three words) occurring in 3 or more records (*multi-word indexing*). The two methods preprocessed UMLS terms identically and differed only in the indexing of non-UMLS terms.

Parameter estimation for SB. The parameters for SB include the probabilities for the document class and conditional probabilities for all terms being considered. These are computed as follows. Let *C* represent a patient record and *c* denote the two possible values that it can take -- *ADE* and *not ADE*. Let F_i represent a term and *f* denote the two possible values that it can take -- *present* or *absent*. Suppose *F* is a set of *n* terms, each of which is known to be either *present* or *absent*. The probability of *c* given *F* is computed according to Bayes rule:

$$P(C=c\,|\,F)=\frac{P(C=c)\prod_{i=1}^{n}P(F_i=f\,|\,C=c)}{\sum_{C}\left\{P(C)\prod_{i=1}^{n}P(F_i=f\,|\,C)\right\}},$$

where *n* is the number of terms in the model and the sum in the denominator is over all possible values of *C*. We estimate the probabilities $P(C = c)$ and $P(F_i = f\,|\,C = c)$ for the 4 possible combination values of *f* and *c*. $P(C = c)$ is estimated as $(freq(C = c) + 1) / (N + 2)$ where $freq(C = c)$ is the number of records that belong to class *c* and *N* is the total number of records in the training set. Similarly, $P(F_i = f \,|\, C = c)$ is estimated as $(freq(F_i = f, C = c) + 1) / (freq(C = c) + 2)$ where $freq(F_i = f, C = c)$ is the number of times that *f* and *c* occur together in the training set. These ratios produce estimates that are less extreme than maximum likelihood estimates based on the ratios $freq(F_i = f, C = c\,) / freq(C = c)$ and $freq(C = c) / N$.

Parameter estimation for MASB. The model-averaged structure is a modified, simple Bayes network over the list of terms. The desired parameters for this network are computed according to the equations derived in [5].

Feature selection for DFS. We used a greedy algorithm that begins with a model with no terms and attempts to add terms to it, one at a time. For every term considered for inclusion, a probability is computed for each record in the training set with the model parameterized from the remaining records using the method described for SB. The area under the ROC curve (AUC) obtained from these probabilities is the metric used for scoring the model. If the AUC improves, the term is included in the model and the algorithm considers the next term from the list; or else, the term being considered is rejected, the algorithm terminates and the existing model is returned.

AMC. The features of the models created by AMC differ from those generated by the above algorithms, in that, each feature is not constrained to be a single term but is allowed to be a disjunction of terms. For example, an AMC model may have 2 features where the first feature has the terms *neutropenia* and *thrombocytopenia* combined disjunctively and the second feature has only the term *toxicity*. This model indicates that a record has a high probability of being labeled for an ADE if the term "neutropenia" OR the term "thrombocytopenia" occurs AND the term "toxicity" also occurs. AMC does feature selection and for each term considered for inclusion in a model, a probability is computed for each record in the training set with the model parameterized from the remaining records using the method described for SB.

To generate a set of models, AMC assesses the minimum sensitivity and specificity levels that are acceptable to the user. AMC then searches within each of several ranges of sensitivities (e.g., 0 to 0.05, 0.05 to 0.10, …, 0.95 to 1.0), trying to maximize the specificity within each range. Similarly, it searches within each of several different ranges of specificities trying to maximize the sensitivity within each range.

Figure 1 shows the pseudo-code for locating a model in a given sensitivity range; the specificity code is analogous. We used AMC with the minimum sensitivity and specificity set to 0 and the range interval of both parameters set to 0.05.

Evaluation. For each of the four algorithms -- SB, MASB, DFS and AMC -- we constructed two models (in case of AMC, two sets of models) using the two methods of indexing. We evaluated the models on the test set and plotted ROC curves for each one.

```
initialize current model to empty
for each term t in the terms list do:
    create new models by adding t either
        as a disjunct to one of the existing features of the
        leading model or as a new feature of that model.
    for each new model do:
        compute parameters as in the SB algorithm.
        consider the new model as the leading model if it
        has a  sensitivity in current range of focus and
        either has a higher specificity or a higher
        sensitivity + specificity than any previously
        considered model.
    loop
loop
```

Figure 1. Pseudo-code used by AMC to locate a model in a given sensitivity range.

RESULTS

For both indexing schemes, models generated by SB and MASB included 100 features, one for each input term. The DFS model had 21 and 25 features with single-word and multi-word indexing, respectively. For single-word indexing, AMC created 16 models all having a single feature containing from 4 to 50 terms with an average of 16.6 terms per model. For multi-word indexing, AMC generated models with the number of terms ranging from 5 to 66 with a mean of 24.9 terms per model. An example of an AMC model with a single feature having 20 disjunctively combined terms is shown below. It achieved a sensitivity of 54% and specificity of 86%.

renal failure (9)	prednisone (7)
steroids (8)	appetite (7)
endocrine (8)	bactrim (7)
diflucan (8)	rash (5)
thrombocytopenia (7)	neutropenia (4)

Table 1. Top 10 terms from AMC models with single-word indexing.

steroid use, candida, induced, toxicity, prograf, ana, imuran, bal, neutropenia, rejection, antibody, fk, hepatitis b, diflucan, filter, appetite, orthotopic, individuals, virus, acyclovir

Table 1 lists the top 10 terms that appear in AMC models (generated from UMLS terms and single words) along with their frequency of occurrence. The commonest term is a UMLS phrase, *renal failure*, while the remaining are single-word terms.

Figures 2 and 3 show the ROC plots for the models obtained with single-word and multi-word indexing, respectively. For AMC the ROC is shown as a sequence of circles that indicate distinct models.

The performance of the 4 algorithms on the test dataset is summarized in Table 2. For readability, the area under the ROC curves (AUCs) are reported as percentages so they range from 0 to 100 instead of 0 to 1. The table also gives the p-values for the differences in the mean AUCs between the two indexing schemes for each algorithm. There is no strongly statistically significant difference between the schemes for any algorithm, although interestingly the results are suggestive of single-word indexing performing better.

Table 3 summarizes the pair-wise comparisons of the performance of the algorithms with single-word indexing. The performance of SB, MASB and AMC are statistically similar ($p < 0.05$) and all 3 perform significantly better than DFS.

Algorithm	Single-word indexing	Multi-word indexing	p-value
SB	77.39 ± 2.47	74.27 ± 2.59	0.0676
MASB	77.11 ± 2.48	74.19 ± 2.60	0.0867
DFS	68.07 ± 2.97	64.16 ± 2.82	0.1292
AMC	76.87 ± 4.12	74.59 ± 4.92	0.0742

Table 2. AUCs of the algorithms.

Pair	p-value	Pair	p-value
SB vs MASB	0.1832	MASB vs DFS	0.0089
SB vs DFS	0.0046	MASB vs AMC	0.0932
SB vs AMC	0.1264	DFS vs AMC	0.0165

Table 3. Pair-wise comparisons of AUCs for single-word indexing.

DISCUSSION

Our results suggest that SB, MASB and AMC can form the basis of an automated text classifier for narrative medical records such as discharge summaries related to ADEs. DFS applies additional feature selection to reduce the number of terms in a single model; however, the resulting smaller model decreases classification accuracy. AMC also employs additional feature selection but generates separate models over the range of the ROC curve, enabling it to improve the AUC.

The set of AMC models perform as well as those generated by SB and MASB. In addition, the AMC models are more compact with fewer terms. Models generated by AMC typically had a single feature composed of disjunctions of terms; these terms can be potentially used as keywords for searching free-text medical records. They can also be evaluated for incorporation into rules used to trigger computer-generated signals for chart review.

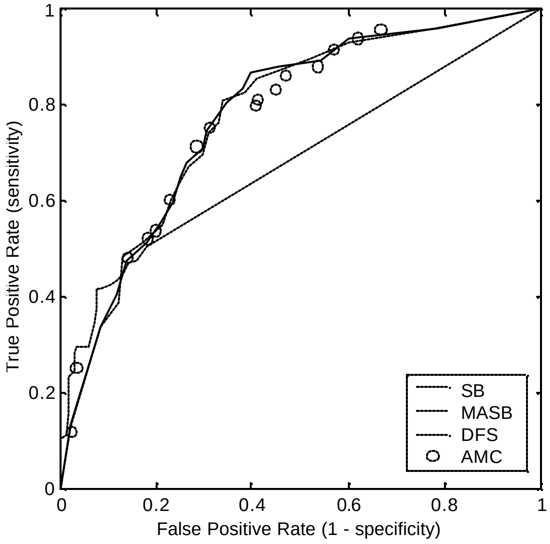

Figure 2. ROC plots for single-word indexing.

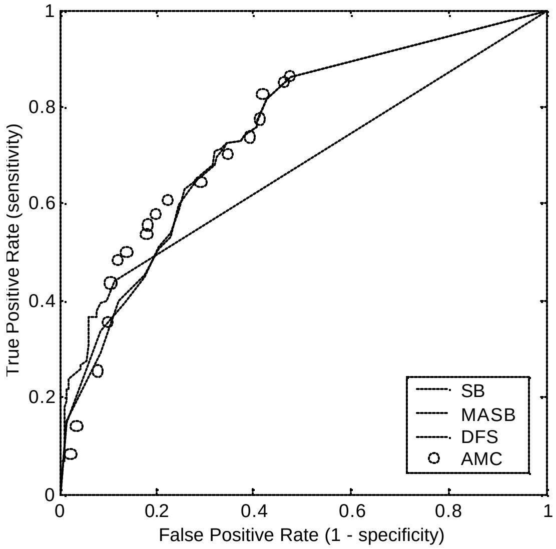

Figure 3. ROC plots for multi-word indexing.

Another finding is that additional indexing of 2-word and 3-word terms did not boost the performance of any of the algorithms we tested. UMLS terms in combination with single words were adequate for building discriminative models for this data set.

Simple Bayes systems are efficient, robust and reasonably easy to implement. We have shown that such systems can learn to correctly classify ICD-9-CM ADE-related codes. The results reported here suggest the following application: First, train models based on a large random sample of discharge summaries that are manually labeled for ADEs by experts. Then, apply those models to signal those patient cases that have high probability for ADEs.

FUTURE RESEARCH

Future research possibilities include application of alternative text categorization algorithms as well as feature selection methods for improving performance. Assessing ADE classification performance of models trained using a large random sample of discharge summaries that have been manually coded for ADEs by experts would be helpful. It would be interesting to investigate the number of true ADE cases found by the models that were not ICD-9 coded as ADEs. Another extension would be to evaluate the algorithms with additional sources of information that are currently available in hospital information systems, such as coded laboratory data and other narrative reports.

ACKNOWLEDGEMENTS

We would like to thank the three anonymous reviewers for their helpful comments. This work was supported in part by the grant T15-LM/DE07059 from the National Library of Medicine (NLM) to the University of Pittsburgh's Biomedical Informatics Training Program and by an IAIMS grant G08 LM06625 from NLM to the University of Pittsburgh.

REFERENCES

1. Bates DW, Evans RS, Murff H, Stetson PD, Pizziferri L, Hripcsak G. Detecting adverse events using information technology. J Am Med hform Assoc 2003 Mar-Apr;10(2):115-28.
2. Honigman B, Lee J, Rothschild J, et al. Using computerized data to identify adverse drug events in outpatients. J Am Med Inform Assoc 2001 May-Jun;8(3):254-66.
3. Hripcsak G, Austin JH, Alderson PO, Friedman C. Use of natural language processing to translate clinical information from a database of 889,921 chest radiographic reports. Radiology 2002 Jul;224(1):157-63.
4. Sebastiani F. Machine learning in automated text categorization. ACM Computing Surveys 2002; 34(1):1-47.
5. Dash D, Cooper GF. Exact model averaging with naïve Bayesian classifiers. In: The Nineteenth International Conference on Machine Learning (ICML 2002); 91-98.
6. Cooper GF, Buchanan BG, Chapman WW, et al. Creating a software tool for the clinical researcher -- the IPS system. In: AMIA Symp 2002; Theater demonstration.
7. Chapman WW, Bridewell W, Hanbury P, Cooper GF, Buchanan BG. Evaluation of negation phrases in narrative clinical reports. In: Proc AMIA Symp 2001; 105-9.