Medical Text Representations for Inductive Learning

Adam Wilcox, Ph.D., George Hripcsak, M.D.
Department of Medical Informatics, Columbia University, New York, NY

Inductive learning algorithms have been proposed as methods for classifying medical text reports. Many of these proposed techniques differ in the way the text is represented for use by the learning algorithms. Slight differences can occur between representations that may be chosen arbitrarily, but such differences can significantly affect classification algorithm performance. We examined 8 different data representation techniques used for medical text, and evaluated their use with standard machine learning algorithms. We measured the loss of classification-relevant information due to each representation. Representations that captured status information explicitly resulted in significantly better performance. Algorithm performance was dependent on subtle differences in data representation.

INTRODUCTION

Electronic clinical information is typically stored either as structured, coded data or as full-text reports. Medical text reports represent a significant source of clinical data, especially data that are not available in coded electronic form. A recent study distinguishing planned and unplanned readmissions found that coded information alone was not sufficient for classifying admissions, and that information in text reports significantly improved the classification.[1] To be useful to automated systems, the information stored as narrative text must be represented in a way that can be used effectively. This information includes not only data directly retrieved from the text, but complex conclusions drawn from it.

One method of converting information extracted from narrative text is by classification. Documents can be categorized as to whether or not they are members of a specific class, such as diagnosis. Each document can then be assigned the appropriate class label, representing an interpretation of the content of the report. For example, reports can be classified as to whether or not they indicate specific clinical conditions; e.g., congestive heart failure or acute bacterial pneumonia. The standardized representation of the document by conditions indicated can then be used for such purposes as alerts, outcome studies, automated guidelines, and research queries.

A number of researchers have studied the use of document classification for use with automated decision support systems. Some approaches have classified based on queries for specific free-text phrases within documents.[2] Others have used varying degrees of natural language processing (NLP), and then queried the processed text.[3-5] Both of these approaches have typically required manually written rules to interpret the document queries.

Writing classification rules is typically a difficult and time-intensive task.[6] As a result, other researchers have investigated the use of inductive learning algorithms to automatically generate classifiers for medical documents.[6-13] These generated classifiers have all used either free text or natural language processor output. However, the actual representation of the document used by the inductive algorithms has varied greatly among the different researchers.

In general inductive learning research, it is known that the performance of a generated classifier depends greatly on the actual representation of the objects to be classified.[14] Still, the effect of different representations for medical documents has not been sufficiently addressed, and is largely unknown. This paper analyzes the different medical text representations that have been used by generated classifiers, and evaluates their use by inductive learning algorithms. Details of the studies using different representations are described here.

Studies of inductive text classification

Three studies represented the raw text of the documents. Hersh et al.[9] used logistic regression to assign procedure codes in a trauma registry, based on the text of the dictated report from the initial ER physician. The representation used was a vector of the frequency of each word in the dictation, after removing stop words. Yang and Chute[8] introduced a learning method called Expert Network, and used it to predict ICD-9-CM codes for surgical reports. The reports were also represented as a vector of words, though the words were assigned a weight rather than a raw frequency. Larkey and Croft[10] tested three different types of learning algorithms to assign ICD9 codes to discharge summaries. They used a similar representation to Yang and Chute, with a vector of word weights for each document.

Two studies used a limited form of natural language processing to identify concepts rather than individual words in the documents. Lehnert et al.[13] used a decision tree algorithm to identify patient encounter notes that occurred in response to asthma exacerbations. Only those words and phrases that occurred in a dictionary of relevant terms were used, and then mapped to specific codes representing the concepts. The representation of the document was

1067—5027/00/$5.00 © 2000 AMIA, Inc.

thus a vector based on the presence or absence of the codes detected. Aronow et al.[7] also mapped words and phrases to concepts. However, they included a specific method to detect negation of those concepts. For each concept, a separate concept representing it in negated form was included. For example, the concept "pneumonia" would be converted to two concepts, "pneumonia" and "no_pneumonia" for use in the vector representation. They used this representation and relevance feedback to classify mammography reports for the presence and absence of cancer.

Three studies represented documents based on the output of existing natural language processors. Zingmond and Lenert[6] used CAPIS,[15] developed at Stanford University. CAPIS was used to extract findings or observations from a text report, and assign each finding one of state values: instantiated-positive, instantiated-negative, or not-instantiated. Findings that were mentioned in the report were considered instantiated; the positive and negative qualifiers separated findings that were present conditions of the patient, versus those that were resolved or explicitly stated as not present (i.e., refuted). Instead of identifying all observations in a report, it only identified target findings specified by a user. The representation of a document was thus a vector of the specified findings and their state values. Zingmond and Lenert used CAPIS and a decision tree generator to extract findings from chest radiograph reports for identifying indications of cancer.

Chapman and Haug[12] used the SymText parser[16] developed at LDS Hospital. They used Bayesian networks and decision trees to classify chest x-ray reports indicating pneumonia. SymText identifies observations and characteristics or modifiers describing those observations. Rather than include all modifiers, Chapman and Haug identified the finding state, which had possible values of "present" or "not present". This resulted in a document vector similar to that of Zingmond and Lenert, except that there was no difference between instantiated-negative and not-instantiated.

Wilcox and Hripcsak[11,17] used MedLEE, developed at Columbia University, in previous studies. They classified these reports for indications of six different clinical conditions using various machine learning algorithms. MedLEE outputs observations and modifiers in a hierarchical structure similar to SymText. Rather than only using the observation with its identified state in a document vector, they combined modifiers with their associated observations. The document vectors thus contained concepts representing individual observations, as well as separate concepts representing these observation-modifier combinations. The values of the observations were either "instantiated" or "not instantiated", while the possible values of the modifiers were dependent on the modifier. They also tested a representation that only used concepts based on observations, where presence for negated concepts was inferred from modifiers. This representation was identical to that used independently by Chapman and Haug.

METHODS

We tested various document representations for chest radiograph reports. From the described studies, we developed eight different data representations for evaluation: *text*, *text-bin*, *concepts*, *con-mod*, *no-concepts*, *NLP-mod*, *NLP-ref*, *NLP-neg*.

The first two representations were based on free text alone. We extracted stop words from the same stop list used by Hersh, and then formed a document vector of all the existing words in the documents. The *text* representation weighted each term using *tf*idf* weighting. With other general text classification studies in machine learning, weighting has instead been binary.[18] Therefore, the *text-bin* representation uses vectors of the raw text with values indicating only whether they occur in the document.

Three representations are based on those studies that used limited natural language processing. We processed the reports using MedLEE, and created a vector of all observations that were included. We used a single processor to isolate the effects of the representations from biases of different processors. For each document vector, the observations were assigned a binary coding of whether they were instantiated in the document. This resulted in the *concepts* representation. In the *con-mod* representation, we considered the concepts and the modifiers with their values as separate observations. For example, if the concept "edema" occurred, with the modifier-value "degree = low", it would be represented as two separate concepts, "edema" and "degree=low," each assigned a value of presence or absence on whether or not they were instantiated in the report. All concepts were treated independently; i.e., no distinction would be seen between "degree=low" that was modifying edema or pneumonia. The *no-concepts* representation was a slight variant of the *concepts* representation, that also included for refuted concepts. If the observation "edema" was instantiated but negative, the concept "edema" would not be indicated as present, but the concept "no_edema" would be.

An additional three representations were based on studies using natural language processing. The *NLP-mod* representation paired modifiers with their

Table 1: Representation characteristics.

Representation	Text representation	Vector components	Component values
text	raw text	words	*tf*idf weights*
text-bin	raw text	words	instantiated, not instantiated
concept	NLP output	concepts	instantiated, not instantiated
con-mod	NLP output	concepts, modifiers	instantiated, not instantiated
no-concept	NLP output	concepts, "NO_"+concepts	instantiated, not instantiated
NLP-mod	NLP output	concepts, concepts+modifiers	instantiated, not instantiated, *modifier value*
NLP-ref	NLP output	concepts	instantiated-positive, instantiated-negative, not instantiated
NLP-neg	NLP output	concepts	present, not present

associated observations and included them as concepts (e.g., "edema" and "edema^degree" would be the concepts from the above example). The *NLP-ref* representation only used the actual observations as concepts, but assigned "instantiated-positive," "instantiated-negative," and "not-instantiated" as values. The *NLP-neg* was the same as *NLP-ref*, except that both "instantiated-negative" and "not-instantiated" were converted to "not present." Table 1 shows the differences between the different representations.

To determine whether an instantiated concept should be considered as present or refuted (or absent), we wrote a simple rule to examine the "certainty" and "status" modifiers of each concept. For each of these modifiers, we determined beforehand values that imply an instantiated observation was actually not a present condition, but rather was refuted as a condition in the report.

Information capture

We first evaluated those representations based on NLP output for their ability to capture information relevant to classification. Each representation loses some information that originally represented in the narrative form, as well as information represented in the NLP output. For example, *concepts*, *no-concepts*, *NLP-ref* and *NLP-neg* all lose some modifier information, while *con-mod* loses the association between modifiers and the observations. Most of the representations, including *NLP-mod*, lose information about the frequency of occurring concepts. We wanted to test whether the information lost was important for classification.

The representations were evaluated within the specific task of detecting six clinical conditions from the information contained in the original reports: congestive heart failure, chronic obstructive pulmonary disease, acute bacterial pneumonia, neoplasm, pleural effusion without congestive heart failure, and pneumothorax. We used 200 randomly selected reports that had been classified by physicians for presence and absence of each of the clinical conditions. These reports were used in previous evaluation studies; specifically, they were used in a study that demonstrated the ability of MedLEE to accurately code information from free text reports.[3] In that study, rules written by a physician and knowledge engineer were used to query the MedLEE output for each document and classify each report. The study found that the rules using MedLEE were not significantly different in performance from physicians who classified the narrative text reports.

We modified these rules to query the specific data representations, without changing the logic of the rules. The rules were then used to classify the reports using each representation, and the performance was compared to the original rules. Performance was measured in terms of sensitivity and specificity, from which we calculated the ROC area using the Pollack-Norman estimate A'.[19,20] This value was then averaged over all six conditions to generate a single performance measure for each representation. We used bootstrapping to estimate variances directly from the data.[21]

Inductive learning algorithm performance

We also evaluated the different representations in terms of learning algorithm performance. For each representation, we used machine learning algorithms to build classifiers, and then evaluated the performance of the classifiers. For this evaluation, we used the same 200 chest radiograph reports and 6 clinical conditions. To allow for use of as large a training set as possible, we tested using leave-one-out cross-validation. To prevent overfitting the classifiers to the training data, we reduced features based on the predictive accuracy of each feature.[17] The number of features was limited to one-tenth the size of the training set.[22]

The *text* representation was studied previously using information retrieval methods (relevance feedback) for classification,[11] and the results of that evaluation are included here. The other representations were evaluated with three different machine learning algorithms: MC4, naïve-Bayes and

IB. MC4 is a variant of the C4.5 decision tree algorithm,[23] naïve-Bayes is a common probabilistic algorithm which predicts via assuming conditional independence among attributes,[24] and IB is a *k*-nearest neighbor algorithm.[25] The algorithms were interfaced by the MLC++ machine learning library.[26] We used these algorithms because they are well-known and studied in machine learning, and represent completely different approaches to inductive learning. For each algorithm, we computed an average ROC area for the six clinical conditions.

RESULTS

Table 2: Information capture.

Representation	ROC Area A' (95% CI)
rules	0.94 (0.92 – 0.97)
concepts	0.90 (0.87 – 0.92)
con-mod	0.85 (0.81 – 0.88)
no-concepts	0.94 (0.92 – 0.96)
NLP-mod	0.95 (0.93 – 0.97)
NLP-ref	0.94 (0.92 – 0.96)
NLP-neg	0.94 (0.92 – 0.96)

Table 2 shows the average ROC area for different representations when using expert written rules to classify reports. It also shows the baseline performance of the rules using NLP output. The *concepts* and *con-mod* representations had significantly worse performance than the original rules, while there was no detected loss of information for the other representations.

Table 3 contains the average ROC area for the three different learning algorithms when using different representations, as well as the ROC area from *text* using relevance feedback. Figure 1 shows the performance of each algorithm as well as these averages. The dark lines at the bottom of the figure indicate significance between the averages; i.e., we detected no difference between the representations covered by the same line. The *text* representation was significantly worse than all others except *text-bin*.

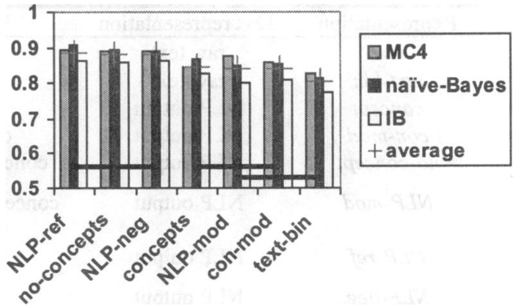
Table 3: Inductive learning algorithm performance.

Representation	Average ROC Area (95% CI)
text	0.79 (0.79 – 0.80)
text-bin	0.81 (0.77 – 0.85)
concepts	0.85 (0.80 – 0.89)
con-mod	0.84 (0.80 – 0.88)
no-concepts	0.88 (0.85 – 0.91)
NLP-mod	0.84 (0.82 – 0.86)
NLP-ref	0.89 (0.86 – 0.91)
NLP-neg	0.88 (0.85 – 0.91)

DISCUSSION

Each representation studied had different advantages and disadvantages. Those that were based on raw text were easily implemented, and did not require natural language processing. However, those that used NLP could detect refuted concepts. Those that did not discern between instantiated and present did not require rules to infer when concepts were actually refuted in a report.

Figure 1: ROC area for different algorithms.



An important result of the study is that most of the representations did not lose information that was significant for the classification task. Converting the hierarchical NLP data to a flattened form for use with learning algorithms can be a complex task, with potential to lose much content, especially at higher levels of the structure (i.e., modifiers). However, the result should not be surprising when considering the dependencies of the study. For one, the performance was evaluated using manually authored rules. It is cognitively difficult to conceptualize hierarchical data and relationships. Consequently, most authored rules are likely to use only one to two levels in a hierarchy. A second dependency is on the knowledge representation of MedLEE itself. A common task in developing a controlled vocabulary for a natural language processing is determining pre-coordination of concepts.[27] Controlled vocabularies often assign a single concept to a combination of a concept and modifier. The determination of pre-coordination is often either by frequency or importance of the combination. The finding "pain" modified by "body location: chest" is often determined important enough to be considered as a separate concept, "chest pain." Therefore, the only attributes that would remain beyond a secondary level are those already determined to be not significant.

The study also found that differences in performance of machine learning algorithms were sensitive to subtle changes in data representations. For example, performance improved when status information for concepts was defined explicitly. *NLP-mod* captured all the information necessary to infer the status of concepts, though it was not defined by the representation. MC4, which creates decision rules for classification, still performed as well as when the status was explicit in the representation (*NLP-ref, NLP-neg, no-concepts*). However, both naïve-Bayes and IB were worse with *NLP-mod*. On the other hand, the naïve-Bayes algorithm performed better relative to MC4 when modifiers were not captured in the representation, and status could not be inferred correctly (*concepts*). Therefore, when comparing learning algorithms, differences between performance may be due more to the data representation rather than how the algorithms match the learning task.

CONCLUSION

Various attempts to use inductive learning for medical text classification have used different representations of the document content. Such differences, while subtle in appearance, can have significant differences both in the information captured from the document, and the performance of different inductive algorithms using the representation.

ACKNOWLEDGMENTS

This work was supported by National Library of Medicine Grants R29 LM05627 "Linking Knowledge-Based Systems to Clinical Databases," 5T15 LM07079 "Research Training Grant," and Pfizer Inc grant "Community Acquired Pneumonia Management." We would also like to thank Dr. Stephen B. Johnson for his suggestions, and Dr. William Hersh for supplying a list of stop words.

References

1. Kossovsky MP, Sarasin FP, Bolla F, Gaspoz JM, Borst F. Distinction between planned and unplanned readmissions following discharge from a Department of Internal Medicine. Methods Inf Med 1999;38(2):140-3.
2. de Estrada WD, Murphy S, Barnett GO. Puya: a method of attracting attention to relevant physical findings. Proc AMIA Annu Fall Symp 1997;509-13.
3. Hripcsak G, Friedman C, Alderson PO, DuMouchel W, Johnson SB, Clayton PD. Unlocking clinical data from narrative reports: a study of natural language processing. Ann Intern Med 1995;122(9):681-8.
4. Jain NL, Knirsch CA, Friedman C, Hripcsak G. Identification of suspected tuberculosis patients based on natural language processing of chest radiograph reports. Proc AMIA Annu Fall Symp 1996;542-6.
5. Fiszman M, Chapman WW, Evans SR, Haug PJ. Automatic identification of pneumonia related concepts on chest x-ray reports. Proc AMIA Symp 1999;67-71.
6. Zingmond D, Lenert LA. Monitoring free-text data using medical language processing. Comput Biomed Res 1993;26(5):467-81.
7. Aronow DB, Fangfang F, Croft WB. Ad hoc classification of radiology reports. J Am Med Inform Assoc 1999;6(5):393-411.
8. Yang Y, Chute CG. An application of Expert Network to clinical classification and MEDLINE indexing. Proc Annu Symp Comput Appl Med Care 1994;157-61.
9. Hersh WR, Leen TK, Rehfuss PS, Malveau S. Automatic prediction of trauma registry procedure codes from emergency room dictations. Medinfo 1998;9 Pt 1:665-9.
10. Larkey LS, Croft WB. Combining classifiers in text categorization. Proceedings of the 19th Annual International ACM Special Interest Group in Information Retrieval. 1996; 289-97.
11. Wilcox A, Hripcsak G. Classification algorithms applied to narrative reports. Proc AMIA Symp 1999;(1-2):455-9.
12. Chapman WW, Haug PJ. Comparing expert systems for identifying chest x-ray reports that support pneumonia. Proc AMIA Symp 1999;(1-2):216-20.
13. Lehnert W, Soderland S, Aronow DB, Feng F, Shmueli A. Inductive text classification for medical applications. J Exper Theoret Artif Intell 1995;7:49-80.
14. Lewis DD. Jacobs PS, editors. Text-Based Intelligent Systems. Hillsdale, NJ: Lawrence Erlbaum; 1992; Text representation for intelligent text retrieval: A classification-oriented view. p. 179-97.
15. Lin R, Lenert L, Middleton B, Shiffman S. A free-text processing system to capture physical findings: Canonical Phrase Identification System (CAPIS). Proc Annu Symp Comput Appl Med Care 1991;843-7.
16. Haug PJ, Koehler S, Lau LM, Wang P, Rocha R, Huff SM. Experience with a mixed semantic/syntactic parser. Proc Annu Symp Comput Appl Med Care 1995;284-8.
17. Wilcox A, Hripcsak G. Knowledge discovery and data mining to assist natural language understanding. Proc AMIA Annu Fall Symp 1998;835-9.
18. Moulinier I. A framework for comparing text categorization approaches. 1996; AAAI Spring Symposium on Machine Learning in Information Access.
19. Pollack I, Norman DA. A non-parametric analysis of recognition experiments. Psychonomic Science 1964;1:125-6.
20. Grier JB. Nonparametric indexes for sensitivity and bias: computing formulas. Psychol Bull 1971;75(6):424-9.
21. Efron B; Tibshirani RJ. An Introduction to the Bootstrap. New York: Chapman and Hall; 1993.
22. Peduzzi P, Concato J, Kemper E, Holford TR, Feinstein AR. A simulation study of the number of events per variable in logistic regression analysis. J Clin Epidemiol 1996;49(12):1373-9.
23. Quinlan J. C4.5: Programs for Machine Learning. Redwood City, CA: Morgan Kaufmann; 1993.
24. Langley P, Iba W, Thompson K. An analysis of Bayesian classifiers. Proceedings of the Tenth National Conference on Artificial Intelligence. San Jose, CA: AAAI Press; 1992; p. 223-228.
25. Aha DW. Tolerating noisy, irrelevant and novel attributes in instance-based learning algorithms. International Journal of Man-Machine Studies 1992;36(1):267-87.
26. Kohavi R, Sommerfield D. Data mining using MLC++: a machine learning library in C++. IEEE Computer Society Press; 1996; p. 234-45.
27. Wilcox A, Friedman C, Hripcsak G. Natural language as a tool in the development of a controlled vocabulary. Proc AMIA Annu Fall Symp 1998;1098.