Identification of Patients with Congestive Heart Failure using a binary classifier: a case study.

Serguei V. Pakhomov
Division of Medical Informatics Research
Mayo Foundation
pakhomov@mayo.edu

James Buntrock
Division of Medical Informatics Research
Mayo Foundation
buntrock@mayo.edu

Christopher G. Chute
Division of Medical Informatics Research
Mayo Foundation
chute@mayo.edu

Abstract

This paper addresses a very specific problem that happens to be common in health science research. We present a machine learning based method for identifying patients diagnosed with congestive heart failure and other related conditions by automatically classifying clinical notes. This method relies on a Perceptron neural network classifier trained on comparable amounts of positive and negative samples of clinical notes previously categorized by human experts. The documents are represented as feature vectors where features are a mix of single words and concept mappings to MeSH and HICDA ontologies. The method is designed and implemented to support a particular epidemiological study but has broader implications for clinical research. In this paper, we describe the method and present experimental classification results based on classification accuracy and positive predictive value.

1 Introduction

Epidemiological research frequently has to deal with collecting a comprehensive set of human subjects that are deemed relevant for a particular study. For example, the research focused on patients with congestive heart failure needs to identify all possible candidates for the study so that the candidates could be asked to participate. One of the requirements of a study like that is the completeness of the subject pool. In many cases, such as disease incidence or prevalence studies, it is not acceptable for the investigator to miss any of the candidates. The identification of the candidates relies on a large number of sources some of which do not exist in an electronic format, but it may start with the clinical notes dictated by the treating physician.

Another aspect of candidate identification is prospective patient recruitment. Prospective recruitment is based on inclusion or exclusion criteria and is of great interest to physicians for enabling just-in-time treatment, clinic trial enrollment, or research study options for patients. At Mayo Clinic most clinical documents are transcribed within 24 hours of patient consultation. This electronic narration serves as resource for enabling prospective recruitment based on criteria present in clinical document.

Probably the most basic approach to identification of candidates for recruitment is to develop a set of terms whose presence in the note may be indicative of the diagnoses of interest. This term set can be used as a filtering mechanism by either searching on an indexed collection of clinical notes or simply by doing term spotting if the size of the collection would allow it. For example, in case of congestive heart failure, one could define the following set of search terms: “CHF”, “heart failure”, “cardiomyopathy”, “volume overload”, “fluid overload”, “pulmonary edema”, etc. The number of possible variants is virtually unlimited, which is the inherent problem with this approach. It would be hard to guarantee the completeness of this set to begin with, which is further complicated by morphological and spelling variants. This problem is serious because it affects the recall, which is especially important in epidemiological studies.

Another problem is that such term spotting or indexing approach would have to be intelligent enough to identify the search terms in negated and other contexts that would render documents containing these terms irrelevant. A note containing “no evidence of heart failure” should not be retrieved, for example. Identifying negation reliably and, more importantly, its scope is far from trivial and is in fact a notoriously difficult problem in Linguistics [1]. This problem is slightly less serious than the completeness problem since it only affects precision which is less important in the given context than recall.

In order to be able to correctly identify whether a given patient note contains evidence that the patient is relevant to a congestive heart failure study, one has to “understand” the note. Currently, there are no systems capable of human-like “understanding” of natural language; however, there are methods that allow at least partial solutions to the language understanding problem once the problem is constrained in very specific ways. One such constraint is to treat language understanding as a classification problem and to use available machine learning approaches to automatic classification to solve the problem. Clearly, this is a very limited view of language understanding but we hypothesize that it is sufficient for the purposes referred to in this paper.

2 Previous work

The classification problems that have been investigated in the past are just as varied as the machine learning algorithms that have been used to solve these problems. Linear Least Squares Fit [2], Support Vector Machines, Decision trees, Bayesean learning [3], symbolic rule induction [4], maximum entropy [5], expert networks [6] are just a few that have been applied to classifying e-mail, Web pages, newswire articles, medical reports among other documents.

Aronow et al. [7] have investigated a problem very similar to the one described in this papers. They developed an ad hoc classifier based on a variation of relevance feedback technique for mammogram reports where the reports were classified into three “bins”: relevant, irrelevant and unsure. One of the features of the text processing system they used had to do with the ability to detect and take into account negated elements of the reports.

Wilcox et al. [8] have experimented with a number of classification algorithms for identifying clinical conditions such as congestive heart failure, chronic obstructive pulmonary disease, etc. in raidograph reports. They found that using an NLP system such as MedLEE (Medical Language Extraction and Encoding System) and domain knowledge sources such as UMLS® [9] for feature extraction can significantly improve classification accuracy over the baseline where single words are used to represent training samples.

Jain and Friedman [10] have demonstrated the feasibility of using MedLEE for classifying mammogram reports. Unlike Wilcox [8], this work does not use an automatic classifier, instead, it uses the NLP system to identify findings that are considered suspicious for breast cancer.

3 NaiveBayes vs. Perceptron

We experimented with two widely used machine learning algorithms, Perceptron and Naïve Bayes, in order to train models capable of distinguishing between clinical notes that contain sufficient evidence of the patient having the diagnosis of congestive heart failure (positive examples) from notes that do not contain such evidence (negative examples). The choice of the problem was dictated by a specific grant aimed at studying patients with congestive heart failure.

The choice of the algorithms was largely dictated by efficiency considerations. Both Perceptron and Naïve Bayes belong to a family of linear classifiers which tend to be computationally more manageable on large feature sets like the one we are addressing than other algorithms. Damerau et al. [11] show on the Reuters corpus that sparse feature implementations of linear algorithms are capable of handling large feature sets. We used a sparse feature implementation of these two algorithms available in the SNoW (Sparse Networks of Winnows) Version 2.1.2 package [12]. Perceptron and Naïve Bayes classifiers.

Perceptron is a simple iterative learning algorithm that represents in its simplest form a two-layer (input/output) neural network where each node in the input layer is connected to each node in the output layer. A detailed description can be found in [13] and [14]. There are several well known limitations of this algorithm. The most significant is that the simple Perceptron is unable to learn non-linearly separable problems. In order for this algorithm to work, one should be able to draw a hyperplane in the training data feature space that will linearly separate positive examples from negative. With large multidimensional feature spaces, it is hard to know a priori whether the space is linearly separable; however, a good indication of that can be gleaned from the classification accuracy testing on several folds of training/testing data. If the accuracy results show large fluctuations between folds, then that would be a good indication that the space is not linearly separable. On the other hand if the standard deviation on such a cross-validation task is relatively small, then one could be reasonably certain that Perceptron is a usable technique for the problem.

The other less serious limitation is that there is a chance that the algorithm will falsely conclude convergence in a local minimum on the error function curve without reaching the global minimum, which could also account for low or inconsistent accuracy results. This limitation is less serious because it can be controlled to some extent with the learning rate parameter, which sets the amount by which the weights are adjusted each time Perceptron makes a classification error during training [14].

Naïve Bayes does not have the limitations of Perceptron, but does have limitations of its own. The Bayes decision rule chooses the class that maximizes the conditional probability of the class given the context in which it occurs:

$$(1) \qquad C` = \text{argmax} \quad P(C)\prod_{j=1}^{n} P(V_j \mid C)$$

Here, C` is the chosen category, C is the set of all categories and V_j is the context. Naïve Bayes decision algorithm makes a simplifying assumption that the words in V_j are independent of each other. A particular implementation of the Naïve Bayes decision rule based on the independence assumption to text categorization and word sense disambiguation problems is also known as "bag of words" approach [13]. This approach does not attempt to take into account any sort of possible dependency between the individual words in any given context, in fact it assumes that the word "heart" and the word "failure", for example, occur completely independently of each other. Theoretically, such assumption makes Naïve Bayes classifiers very unappealing for text categorization problems, but in practice it has been shown to perform well on a much greater range of domains than the theory would support.

The common feature between the two techniques is that both are linear classifiers and are relatively efficient which makes them attractive for learning from large feature sets with lots of training samples.

4 CHF pilot study

As part of preliminary grant work to investigate and evaluate incidence, outcome, and etiology trends of heart failure, a pilot study for prospective recruitment using term spotting techniques was tested. Prospective recruitment was needed for rapid case identification with 24 hours of newly diagnosed heart failure patients.

Within Mayo Clinic approximately 75% of clinical dictations are electronically transcribed on the date of diagnosis allowing them to be processed using natural language techniques. Using the terms "cardiomyopathy, heart failure, congestive heart failure, pulmonary edema, decompensated heart failure, volume overload, and fluid overload" all electronic outpatient, emergency department, and hospital dismissal notes were processed. These results were reviewed by trained nurse abstractors to determine if this technique could provide identification of patients with clinically active heart failure. Using the term spotting technique no cases were omitted as compared to standard human diagnostic coding methods of final diagnosis. This pilot provided a valid basis for using term spotting for prospective recruitment; however, the nurse abstractors reported filtering out a large number of documents that were irrelevant to the query, thus indicating that there was room for improvement especially in precision. These were not quantified at the time. The results derived from the test sets used for the study described in this paper display similar tendencies.

5 Human Expert Agreement

For testing a classifier, it is important to have a test bed that contains positive as well as negative examples that have been annotated by human experts. It is also important to establish some sort of an agreement between annotators. For this study we used a test bed created with a specific focus on the diagnosis regarding the patient described within the medical document for a separate pilot study of agreement between annotators (de Groen et al., p. c.).

One of the topics selected for this test bed creation study included congestive heart failure. For each topic, 90 documents were selected for evaluation. Seventy of the 90 documents were chosen from documents with a high likelihood of containing diagnostic information regarding the topic of inquiry. Specifically, thirty-five documents were randomly selected from a pool of documents based on a coded final diagnosis; thirty-five documents were randomly selected from a pool of documents based on a textual retrieval of lexical surface forms (term spotting). The final twenty documents were randomly selected from the remaining documents, not originally included in the coded or text identified collections. A group of Emeritus physicians acted as the human experts for this annotation task. The experts were instructed to determine whether the information contained in the clinical note could support inclusion of the patient in a clinical/research investigation, if such investigation was centered on patients having - at the time the note was created - the topic of inquiry.

Each document was judged by three physicians on the following scale: (confirmed-probable-indeterminate-probably not-definitely not). For the purposes of our study we collapsed "confirmed" and "probable" categories into one "positive" category. We also collapsed "probably not" and "definitely not" into a "negative" category. The "indeterminate" category happened to include such artifacts as differential diagnosis as well as uncertain judgements and therefore was ignored for our purposes. The agreement on this particular topic happened to be low: only 31% of the instances were agreed upon by all three experts; therefore, we decided to use the agreed upon subset of the notes only for testing our approach. The low level of agreement was partly attributable to the breadth of the topic and, partly, to how the instructions were interpreted by the experts. Despite the low level of agreement, we were able to select a subset of 26 documents where all three annotators agreed. These were the documents where all three annotators assigned either the "positive" or the "negative" category. 7 documents were judged as "positive" and 19 were judged as "negative" by all three experts.

6 Feature extraction

Arguably, the most important part of training any text document classifier is extracting relevant features from the training data. The resulting data set looks like a set of feature vectors where each vector should represent all the relevant information encoded in the document and as little as possible of the irrelevant information. To capture the relevant information and give it more weight, we used two classification schemes: MeSH (Medical Subjects Headings) [15]and HICDA (Hospital International Classification of Diseases Adaptation) [16]. The MeSH classification is available as part of the UMLS (Unified Medical Language System) compiled and distributed by the National Library of Medicine (NLM) [9]. HICDA is a hierarchical classification with 19 root nodes and 4,334 leaf nodes. Since 1975, it has been loosely expanded to comprise 35,676 rubrics or leaf nodes. It is an adaptations of ICD-8, which is the 8th edition of the International Classification of Diseases. HICDA contains primarily diagnostic statements, whereas MeSH is not limited to diagnostic statements and therefore the two complement each other. It should also be noted that, for mapping the text of clinical notes to these two ontologies, in addition to the text phrases present in HICDA and MeSH, some lexical and syntactic variants found empirically in medical texts were also added. For MeSH, these variants were derived from MEDLINE articles by UMLS developers and for HICDA, the variants came from coded diagnoses. Having these lexical and syntactic variants in conjunction with text lemmatization made the job of mapping relatively easy. Text lemmatization was done using the Lexical Variant Generator's *(lvg[1])* 'norm' function also developed at NLM.

For the purposes of this experiment, we represented each document as a mixed set of features of the following types: (MeSH code mappings, HICDA code mapping, Single word tokens, Demographic data). First, MeSH and HICDA mappings were identified by stemming and lowercasing all words in the notes and finding

[1] umlslex.nlm.nih.gov

their matches in the two ontologies. Next, stop words were deleted from the text that remained unmapped. The remaining words were treated as single word token features. In addition to these lexical features, we used a set of demographic features such as age, gender, service code (the type of specialty provider where the patient was seen (e. g. ‘cardiology’)) and death indicator (whether the patient was alive at the time the note was created). Since age is a continuous feature, we had to discretize it by introducing ranges A-N arbitrarily distributed across 5 year intervals from 0 to over 70 years old. For this experiment, features that occurred less than 2 times were ignored. The extracted feature “vocabulary” consists of 11,118 unique features. Table 1 shows the breakdown of the feature vocabulary by type.

Feature type	**N features**	**Proportion**
MeSH headings	6631	60 %
HICDA categories	2721	24 %
Single words	1635	15 %
Demographic features	131	01 %
Totals	11,118	100 %

Table 1 Breakdown of training features by type.

7 Experimental Setup

Both Naïve Bayes and Perceptron were trained on the same data and tested using a 10-fold cross-validation technique as well as a held-out test set of 26 notes mentioned in section 4.

7.1 Data

Two types of annotated testing/training data were used in this study. The first type (Type I) is the data generated by medical coders for the purpose of conceptual indexing of the clinical notes. The second type (Type II) is the data annotated by Emeritus physicians (experts).

For Type I data, a set of clinical notes for 6 months of year 2001 was collected resulting in a corpus of 1,117,284 notes. Most of these notes contain a set of final diagnoses established by the physician and coded using the HICDA classification by specially trained staff. The coding makes it easy to extract a set of notes whose final diagnoses suggests that the patient has congestive heart failure or a closely related condition or symptom like pulmonary edema. Once this positive set was extracted (2945 notes), the remainder was randomized and a similar set of negative samples was extracted (4675 notes). The total size of the corpus is 7620 notes. Each note was then run through feature extraction and the resulting set was split into 10 train/test folds by randomly selecting 20% of the 7620 notes to set aside for testing for each fold.

Type II data set was split into two subsets: a complete agreement (TypeII-CA) set and a partial agreement set (TypeII-PA). The complete agreement set was created by taking 26 notes that were reliably categorized by the experts with respect to congestive heart failure specifically. These 26 notes represent a set where all three annotators agreed at least to a large extent on the categorization. “A large extent” here means that all three annotators labeled the positive samples as either “confirmed” or “probable” and the negative samples as either “probably not” or “definitely not”. The set contains 7 positive and 19 negative samples. The partial agreement set was created by labeling all samples for which at least one expert made a positive judgement and no experts made a “negative” judgement as “positive” and then labeling all samples for which at least one expert made a negative judgement and no experts made a positive judgements as “negative”. This procedure resulted in reducing the initial set of 90 samples to 74 of which 21 were positive and 53 were negative for congestive heart failure. This partial agreement set is obviously weaker in its reliability but it does provide substantially more data to test on and would enable us to judge, at the very least, the consistency of the automatic classifiers being tested.

7.2 Training

The following parameters were used for training the classifiers. Naïve Bayes was used with the default smoothing parameter of 15. For Perceptron, the most optimal combination of parameters was to have the learning rate set at 0.0001 (very small increments in weights), the error threshold was set at 15. The algorithm with these settings was run for 1000 iterations.

7.3 Results

Standard classifier accuracy computation [13] for binary classifiers was used.

$$Acc = 100 * \frac{TP + TN}{TP + TN + FP + FN} \quad (2)$$

Where TP represents the number of times the classifier guessed a correct positive value (true positives), TN is the number of times the classifier correctly guessed a negative value (true negatives), FP is the number of times the classifier predicted a positive value but the correct value was negative (false positives) and the FN (false negatives) is the inverse of FP.

In addition to standard accuracy, positive predictive value was also used. It is defined as:

$$PPV = 100 * \frac{TP}{TP + FP} \quad (3)$$

Where TP+FP constitute all positive samples in the test data set. We are interested in positive predictive value because of the strong preference towards perfect recall in document retrieval for epidemiological studies, even if it comes at the expense of precision. The rule is that it is better to identify irrelevant data that can be discarded upon review than to miss any of the relevant patients.

First, we established a baseline by running a a very simple term spotter that looked for the CHF-related terms mentioned in Section 2 (and their normalized variants) in the collection of normalized[2] documents from the Type II data set. The accuracy of the term spotter is 56% on Type II-CA set and 54% on Type II-PA set. Positive predictive value on Type II-CA set is 85% and on Type II-PA set – 71%. The positive predictive value on Type II-CA set reflects the spotter missing only 1 document out of 7 identified as positive by the experts. The results are summarized in Tables 3 and 4.

Fold	**Naïve Bayes**		**Perceptron**		**Delta**	
	PPV (%)	Acc (%)	PPV (%)	Acc (%)	PPV (%)	Acc (%)
1	89.21	84.06	78.42	88.39	-10.79	4.33
2	88.16	82.41	74.88	85.30	-13.28	2.89
3	89.34	82.74	75.74	86.09	-13.61	3.35
4	90.77	82.02	79.62	87.07	-11.15	5.05
5	90.54	82.07	76.51	86.54	-14.03	4.47
6	89.55	82.74	80.27	87.40	-9.29	4.66
7	88.16	82.41	74.88	85.30	-13.28	2.89
8	88.10	81.16	78.62	86.28	-9.48	5.12
9	89.26	81.69	79.36	86.68	-9.90	4.99
10	88.12	80.45	76.59	85.89	-11.53	5.44
Mean	89.12	82.18	77.49	86.49	-11.63	4.32
Stdev	0.99	0.009	2.01	0.02		

Table 2. Classification test results illustrating the differences between Perceptron and Naïve Bayes.

The results of testing the two classifiers are presented in Table 2. Naïve Bayes algorithm achieves 82.2% accuracy, whereas Perceptron gets 86.5%. The standard deviation on the Perceptron classifier results appears to be relatively small, which leads us to believe that this particular classification problem is linearly separable. The difference of 4.3% happens to be statistically significant as evidenced by a t-test at 0.01

[2] normalization was done with the *lvg* stemmer (umlslex.nlm.nih.gov)

confidence level. The difference in the positive predictive value is also significant, however, is it inversely related to the difference in accuracy. Perceptron models perform on average 11 absolute percentage points worse than Naïve Bayes models.

Table 1 shows results that represent the accuracy of the classifiers on classifying the Type I test data that has been generated by medical coders. Clearly, Type I data is not generated in exactly the same way as Type II. Although Type I data is captured reliably and is highly accurate, Type II data is classified specifically with respect to congestive heart failure only, by expert physicians and, we believe, reflects the nature of the task at hand a little better.

In order to test the classifiers on Type II data, we re-trained them on the full set of 7620 notes of Type I data using the same parameters as were used for the 10-fold cross-validation test. The results of testing the classifiers on Type II-CA data (complete agreement) are presented in Table 3.

Classifier	PPV (%)	Acc (%)
NaiveBayes	100	69.2
Perceptron	85	76.92
TermSpotter	85	56

Table 3. Test results for Type II-CA data (annotated by retired physicians with complete agreement).

These results are consistent with the ones displayed in Table 2 in that Perceptron tends to be more accurate overall but less so in predicting positive samples. Table 4 summarizes the same results for Type II-PA test set and the results appear to be oriented in the same general direction as the ones reported in Table 2 and 3.

Classifier	PPV (%)	Acc (%)
NaiveBayes	95	57
Perceptron	86	65
TermSpotter	71	54

Table 4. Test results for Type II-PA data (annotated by retired physicians with partial agreement).

From a practical standpoint, the results presented here are interesting in that they suggest that the most accurate classifier may not be the most useful for a given task. In our case, if we were to use these classifiers for routing a stream of electronic clinical notes, the gains in precision that would be attained with the more accurate classifier would most likely be wiped out by the losses in recall since recall is more important for our particular task than precision. However, for a different task that may be more focused on precision, obviously, Perceptron would be a better choice.

Finally, both Perceptron and Naïve Bayes performance appears to be superior to the baseline performance of the term spotter. Clearly such comparison is only an indicator because the term spotter is very simple. It is possible that a more sophisticated term spotting algorithm may be able to infer semantic relations between various terms and be able to compensate for misspellings and carry out other functions resulting possibly in better performance. However, even the most sophisticated term spotter will only be as good as the initial list of terms supplied to it. The advantage of automatic classification lies in the fact that classifiers encode the terminological information implicitly which alleviates the need to rely on managing lists of terms and the risk of such lists being incomplete. The disadvantage of automatic classification is that the classifier's performance is heavily data dependent, which raises the need for sufficient amounts of annotated training data and limits this methodology to environments where such data is available.

The error analysis of the misclassified notes shows that a more intelligent feature selection process that takes into account discourse characteristics and semantics of negation in the clinical notes is required. For example, one of the misclassified notes contained "no evidence of CHF" as part of the History of Present Illness (HPI) section. Clearly, the presence of a particular concept in a clinical note is not always relevant. For example, various terms and concepts may appear in the Review of Systems (ROS) section of the note; however, the ROS section is often used as a preset template and may have little to do with the present condition. Same is true for other sections such as Family History, Surgical History, etc. It is not clear at this point which sections are to be included in the feature selection process. The choice will most likely be task specific.

The current study did not use any negation identification, which we think accounted for some of the errors. As one of the future steps, we are planning to implement a negation detector such as the NegExpander used by Aronow et al.[7].

8 Conclusion

In this paper, we have presented a methodology for generating on-demand binary classifiers for filtering clinical patient notes with respect to a particular condition of interest to a clinical investigator. Implementation of this approach is feasible in environments where some quantity of coded clinical notes can be used as training data. We have experimented with HICDA codes; however, other coding schemes may be usable or even more usable as well.

We do not claim that either Naïve Bayes or the Perceptron are the best possible classifiers that could be used for the task of identifying patients with certain conditions. All we show is that either one of these two classifiers is reasonably suitable for the task and has the benefits of computational efficiency and simplicity. The results of the experiments with the classifiers suggest that although Perceptron has higher accuracy than the Naïve Bayes classifier overall, its positive predictive value is significantly lower. The latter result makes it less usable for a practical binary classification task focused on identifying patient records that have evidence of congestive heart failure. It may be worth while pursuing an approach that would use the two classifiers in tandem. The classifier with the highest PPV would be used to make the first cut to maximize recall and the more accurate classifier would be used to rank the output for subsequent review.

Acknowledgements

We are thankful to the investigators working on the “Heart Failure” grant RO1-HL-72435 who have provided valuable input and recommendations for this research.

References

1. Horn, L., *A Natural History of Negation.* 1989, Chicago: University Of Chicago Press.
2. Yang, Y. and C. Chute. *A linear least squares fit mapping method for information retrieval from natural language texts*. in *14th International Conference on Computational Linguistics (COLING)*. 1992.
3. Lewis, D. *Naive (Bayes) at forty: The independence assumption in information retrieval*. in *ECML-98*. 1998.
4. Johnson, D., et al., *A deci-sion-tree-based symbolic rule induction system for text categorization.* IBM Systems Journal, 2002. **41**(3).
5. Nigam, K., J. Lafferty, and A. McCullum. *Using Maximum Entropy for Text Classification*. in *IJCAI-99 Workshop on Machine Learning for Information Filtering*. 1999.
6. Yang, Y. *Expert Network: Combining Word-based Matching and Human Experiences in Text Categorization and Retrieval*. in *SIGIR*. 1994.
7. Aronow, D., F. Fangfang, and, and B. Croft, *Ad Hoc Classification of Radiology Reports.* Journal of Medical Informatics Association, 1999. **6**(5).
8. Wilcox, A., et al. *Using Knowledge Sources to Improve Classification of Medical Text Reports*. in *KDD-2000*. 2000.
9. NLM, *UMLS*. 2001, National Library of Medicine.
10. Jain, N. and C. Friedman. *Identification of finding suspiciois for breast cancer based on natural language processing of mammogram reports*. in *AMIA*. 1997.
11. Damerau, F., et al. *Experiments in High Dimensional Text Categorization*. in *ACM SIGIR International Conference on Information Re-trieval*. 2002.
12. Carlson, A.J., et al., *SNoW User’s Guide*, Cognitive Computations Group - University of Illinois at Urbana/Champaign.
13. Manning, C. and H. Shutze, *Foundations of Statistical Natural Language Processing*. 1999, Cambridge, MA: MIT Press.
14. Anderson, J., *Introduction to Neural Networks*. 1995, Boston: MIT Press.
15. NLM, *Fact sheet Medical Subject Headings (MeSH®)*. 2000.
16. Commission on Professional and Hospital Activities, *Hospital Adaptation of ICDA*. 2nd ed. Vol. 1. 1973, Ann Arbor, MI: Commission on Professional and Hospital Activities.