An Experimental Study in Automatically Categorizing Medical Documents

Berthier Ribeiro-Neto and Alberto H.F. Laender
Computer Science Department, Federal University of Minas Gerais, 31270-901, Belo Horizonte, MG, Brazil. E-mail: {berthier, laender}@dcc.ufmg.br

Luciano R.S. de Lima
Medical Informatics Group, Sarah Hospital Network, 30510-000, Belo Horizonte, MG, Brazil. E-mail: luciano@bhz.sarah.br

In this article, we evaluate the retrieval performance of an algorithm that automatically categorizes medical documents. The categorization, which consists in assigning an International Code of Disease (ICD) to the medical document under examination, is based on well-known information retrieval techniques. The algorithm, which we proposed, operates in a fully automatic mode and requires no supervision or training data. Using a database of 20,569 documents, we verify that the algorithm attains levels of average precision in the 70–80% range for category coding and in the 60–70% range for subcategory coding. We also carefully analyze the case of those documents whose categorization is not in accordance with the one provided by the human specialists. The vast majority of them represent cases that can only be fully categorized with the assistance of a human subject (because, for instance, they require specific knowledge of a given pathology). For a slim fraction of all documents (0.77% for category coding and 1.4% for subcategory coding), the algorithm makes assignments that are clearly incorrect. However, this fraction corresponds to only one-fourth of the mistakes made by the human specialists.

Introduction

Most medical organizations produce an abundance of medical documents that are used to support a variety of processes within these organizations. For instance, each treatment (such as a clinical evaluation) of a patient in a hospital starts a documentation/billing process that is likely to produce dozens of distinct records. All these patient records are usually filed together in a *user medical portfolio,* a process that has been largely automated nowadays (Cimino, 1994; Hersh, 1995). Further, these records are useful to governmental agencies and health insurance companies for gathering statistics on health care services and to support billing.

To help with the organization and understanding of all this information, the records in the user medical portfolio are usually categorized according to preestablished knowledge hierarchies. Particularly, it is usually of great interest and value to organize the medical records according to the patient diseases and their causes. Because this categorization is a nontrivial task, medical organizations usually include a team of specialists dedicated to it.

A categorization hierarchy that is largely used throughout the world for organizing patient medical records is the *International Code of Diseases (ICD)* (Cimino, 1995) (other categorization hierarchies, such as MeSH, are more useful for organizing knowledge resources composed of documents from sources such as MEDLINE and journal articles). The ICD standard specifies a hierarchy of medical concepts and category (or subcategory) codes, which allows categorizing medical records for later inspection and querying. The categorization consists in assigning a set of codes to each medical document (which corresponds to placing the document in a number of classes). Because this procedure is usually carried out manually by the coding specialists, automatic algorithms for performing the code assignment are highly desirable. The results of these algorithms can then be quickly inspected (and corrected) by the specialists, a procedure that can save time and lead to improved coding.

The major goal of this article is to thoroughly investigate experimentally the overall quality of the codes generated by an automatic coding algorithm we proposed in Lima, Laender, and Ribeiro-Neto (1998). That algorithm presented very good results for a small test collection composed of 77 inpatient discharge summaries. In here, we study the quality of the codes generated by our algorithm

Received January 24, 2000; revised February 23, 2000; accepted February 12, 2000.

• Published online 5 February 2001

considering a collection of 20,569 medical documents (inpatient discharge summaries, inpatient evolution clinicals and anamnesis). We again observe quite good results. We also include a thorough analysis of those cases in which the codes indicated by the specialists are quite distinct from the codes computed by our algorithm. One interesting conclusion is that the coding specialists might make quite surprising mistakes (i.e., they are fallible).

Our algorithm for the automatic categorization of medical documents is based on the International Code of Diseases (ICD-9) (Cimino, 1995) proposed by the World Health Organization and takes advantage of the hierarchical structure of the ICD to categorize the documents with high precision. Despite being based on the ICD, the model is quite general, and could be equally applied to other hierarchical coding standards such as the Systematized Nomenclature of Medicine (SNOMED), the International Classification of Primary Care (ICPC), and the Read Clinical Codes (RCC) (Cimino, 1995).

The remainder of the article is organized as follows. The Related Work section discusses work related to ours. In the ICD-9 section we present an overview of the International Code of Diseases, version 9, which is the coding standard adopted in our experiments. In the Automatic Code section, we briefly review our code assignment algorithm and also present a comparison with a coding algorithm based on the classic vector space model. In the Experimental Results section, we provide a thorough analysis of our experimental results that consider a database composed of 20,569 medical documents. Finally, the last section concludes the article.

Related Work

Various approaches have been proposed in the literature to address the problem of automatic text categorization, as described in (Baeza-Yates & Ribeiro-Neto, 1999; Sebastiani, 1999). The results of those studies have been used in a variety of applications such as document organization, document filtering, and information retrieval systems. Concisely, we can define the problem as follows. Let $C = \{c_1, c_2, \ldots, c_p\}$ be a set of categories, $c_i = \{a_1, a_2, \ldots, a_q\}$ be a set of attributes of a category $c_i \in C$, $D = \{d_1, d_2, \ldots, d_n\}$ be a set of documents to be categorized, and $d_j = \{e_1, e_2, \ldots, e_m\}$ be a set of elements that compose the document $d_j \in D$. By comparing the elements that form each document d_j with the attributes of a category c_i, we can compute a weight w_{ij}, which is associated with the pair $[c_i, d_j]$. This weight is interpreted as a quantitative degree of membership of the document d_j in the category c_i.

The task of designing algorithms that automatically assign codes to medical documents can also be seen as a problem of automatic text categorization. In this case, the codes are the categories and the terms in the medical documents are the basic elements used to define the categories. This is the basic strategy that we adopt in this work. Other approaches have been discussed in the literature, but they differ from ours in several technical aspects, as we now discuss.

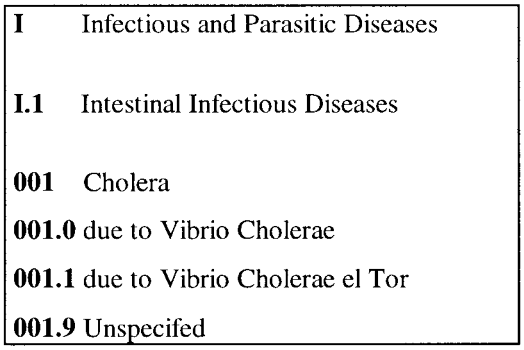
In Hersh and Greenes (1990) and Hersh and Hickam (1995a, 1995b), the project SAPHIRE (Semantic and Probabilistic Heuristic Information Retrieval Environment) is presented. The main objective of this project is the development of methods for indexing and searching collections of medical documents. Further, the project also aims at establishing reference collections (i.e., collections that can be used to compare distinct information retrieval systems) in the medical domain. SAPHIRE proposes to index and search medical collections using a semantic network of medical terms and concepts. The semantic network is based on the metathesaurus defined by the National Library of Medicine within the Unified Medical Language System (UMLS) Project (UMLS, 1994). The terms in this metathesaurus are obtained from various controlled vocabularies such as MeSH (MeSH, 1994), SNOMED (Rothwell, Cote, Cordeau, & Boisvert, 1993), and ICD (Cimino, 1995). The metathesaurus also includes relationships between synonym concepts which allows automatically moving from one controlled vocabulary to the other. SAPHIRE provides categorization (or indexing) of medical documents by automatically assigning codes (in the form of concepts) to a given medical document. This is accomplished by comparing terms in the document with terms associated to concepts in the semantic network. After all concepts have been identified, the similarity between each concept and the document is determined using a formula based on a *tf-idf* weighting scheme (Salton & Buckley, 1988). This is very much like the computation of similarity in the classic vector space model (Baeza-Yates & Ribeiro-Neto, 1999). As discussed in Lima, Laender, and Ribeiro-Neto, (1998), the assignment of concepts (or codes) to medical documents based on a vector-based computation yields results whose precision decreases considerably at high levels of recall (typically, precision falls within 20–30% for recall levels within 60–70%). An approach to improve the precision at higher recall levels is to take advantage of the hierarchical structure present in vocabularies such as MeSH, SNOMED, and ICD. This is one of the major contributions of this work, as we later discuss.

In Tuttle et al. (1998), a multivocabulary system based on the UMLS project is presented. The system provides assistance to a human specialist in the task of determining which (ICD) codes are related to a given medical concept. Although the system can be used to assist with the task of categorizing medical documents, it does not allow the direct and automatic categorization of new documents.

In Larkey and Croft (1996), three different types of classifiers were used (individually or in combination) in the automatic assignment of ICD-9 codes to dictated inpatient discharge summaries. The classifiers used are of three types: k-nearest-neighbor, relevance feedback, and Bayesian. The relationship between medical terms and medical codes was determined using a large training database. The experimental results presented show good results in some situations but have some disadvantages when compared to our approach. For instance, the results depend on a large training set composed of medical documents that have been previously classified. In addition, the approach completely ignores the hierarchical structure of a coding scheme such as the ICD-9 alphabetical index. Other studies that are also based on the usage of general classifiers are discussed in Aronow, Soderland, Ponte, Feng, Croft, and Lehnert (1995), Mladenic and Grobelnik (1999), Rajashekar and Croft (1995), and Yang and Chute (1994).

A rather distinct approach for dealing with the automatic categorization of medical documents is to use techniques from the area of Natural Language Processing (NLP). A detailed overview of categorization algorithms based on this idea can be found in (Spyns, 1996). In this case, the codes are treated as concepts of the medical language whose meanings are defined through sentences in natural language. To assign codes (or classes) to the documents, a natural language processor extracts medical knowledge from the documents and tries (through deduction) to determine the medical concepts involved. The basic algorithm is composed fundamentally of three steps: parsing, transformation/regularization, and encoding. During the parsing, syntatic, and semantic objects are identified and extracted from the medical documents. In the phase of regularization, the semantic objects previously built are standardized into a canonical tree of basic language sentence types. In the encoding, the terms that compose these basic language sentences are mapped into a controlled vocabulary associated with the concepts or categories. The natural language processors are, in general, built using the theoretical foundations of Expert Systems (Graham & Jones, 1988) or Neural Networks (Hertz, Krogh, & Palmer, 1991). The approach yields good results particularly when restricted to specific areas of the medical domain. However, it is in general an excessively complex approach for a problem whose domain vocabulary is clearly small when compared to the vocabulary of any existing language. Despite this drawback, natural language processing techniques have been applied to various specific problems in the medical arena, as we now discuss.

In Friedman, Alderson, Austin, Cimino, and Johnson (1994), and Pellegrin, Bastien, and Roux (1994), experimental natural-language processors are applied to the identification of specific medical concepts (related to the thyroid gland and to clinical radiology). The main contribution in this case is on the construction of natural-language processors specialized in medical environments. In Pietrzyk (1991) and Wehri and Clack (1995), important theoretical aspects associated with the construction of natural-language processors for medical documents are discussed. In Satomura and Amaral (1992), Sager, Lyman, Bucknall, Nham, and Tick (1994), Sager, Lyman, Nhan, and Tick (1995), Delamarre, Burgun, Seka, and Beux (1995), and Chute, Cohn, Campbell, Oliver, and Campbell (1996), the design of generic natural-language algorithms for application to patient data representation and automatic classification (through the use of controlled medical vocabularies (Cimino, 1995) such as ICD and SNOMED) is discussed. These are advanced and complex studies on the automatic processing of medical documents that do not include practical results that can be compared to the experimental results in our work.



I	Infectious and Parasitic Diseases
I.1	Intestinal Infectious Diseases
001	Cholera
001.0	due to Vibrio Cholerae
001.1	due to Vibrio Cholerae el Tor
001.9	Unspecifed

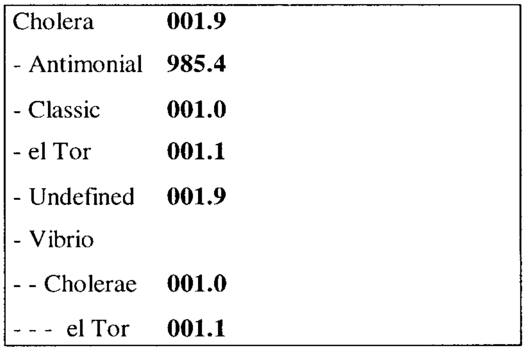
FIG. 1. ICD-9 tabular list of codes (translated to English).

ICD-9: International code of diseases

In this section, we briefly describe the structure of the International Code of Diseases, which we refer to as ICD. This structure reflects the hierarchical nature of the ICD coding scheme that is important here because, distinctly from other approaches for automatic code assignment discussed in the literature, it heavily influences the code assignment algorithm we propose.

The ICD code scheme, which has been proposed by the World Health Organization (WHO), defines a vocabulary of medical terms for describing diseases, injuries, and death causes. This vocabulary is largely used by health organizations throughout the world for assigning ICD codes to a large variety of medical documents. The ninth version of the ICD, called ICD-9, is the most widely used, and is adopted in our study. Because our experiments are based on a medical database from a Brazilian hospital, we use a Portuguese version of the ICD-9 (CID-OMS, 1980). We point out, however, that our algorithm does not depend on the language used for the medical documents and for the International Code of Diseases.

The ICD-9 is organized hierarchically in two documents: a tabular list of codes and an alphabetical index of vocabulary terms. The first ICD-9 document, the *tabular list of codes,* is divided in parts called chapters, sections, categories, and subcategories. Figure 1 illustrates a small portion of the ICD-9 tabular list of codes. The level *I* is a chapter, the level *I*.1 is a section, the level 001 corresponds to the category *cholera,* and the level 001.1 corresponds to the subcategory *cholera due to vibrio cholerae el Tor.* Despite this division, in general, only the levels of category and subcategory are used as reference codes. The other two levels are too generic, and do not provide proper code descriptions. Thus, in this study we focus on the automatic assignment (to medical documents) of ICD-9 category and subcategory codes only.



Cholera	**001.9**
- Antimonial	**985.4**
- Classic	**001.0**
- el Tor	**001.1**
- Undefined	**001.9**
- Vibrio	
- - Cholerae	**001.0**
- - - el Tor	**001.1**

FIG. 2. ICD-9 alphabetical index (translated to English).

Complete descriptions of category and subcategory codes are found in the second ICD-9 document, the *alphabetical index* of vocabulary terms, as shown in Figure 2. An entry point in this index is marked by a term that is not preceded by a dash mark. For instance, in Figure 2, *cholera* is an entry to which a classification code might be assigned to. In this case, the code 001.9 is assigned to the entry point *cholera.* Dash marks are used to indicate terms that are hierarchically dependent of an entry. For instance, *classic* is a hierarchical descend of *cholera* in Figure 2. To the combination *classic cholera* is assigned the code 001.0, which is more specific than the code 001.9. For simplicity, we refer to a combination of hierarchically dependent terms that starts at an entry point as a *codepath.* In Figure 2, the code 985.4 is associated with the codepath *antimonial cholera,* while the code 001.1 is associated with the codepaths *cholera el Tor* and *cholera vibrio cholerae el Tor.* Notice that a same code might be associated with two or more codepaths.

Furthermore, notice that the alphabetical index in Figure 2 includes a subcategory code 001.0 relative to *classic cholera,* which is not present in the tabular list of codes. Also, the entry point for the category *cholera* includes not only the code category 001 but also the subcategory 001.9, which corresponds to *cholera due to a nonspecified* (*or unknown*) *cause.* This subcategory code does not necessarily appear explicitly in the tabular list of codes.

Because the ICD-9 alphabetical index is more complete, our automatic code assignment algorithm is based on it.

Automatic Code Assignment Algorithm

In this section, we describe an algorithm for the automatic assignment of ICD-9 codes to medical documents. The algorithm considers, as one of its fundamental premisses, the hierarchical structure of the ICD-9 alphabetical index. While here we focus on the ICD-9 alphabetical index, we notice that the algorithm can be easily adapted to consider other medical coding schemes such as SNOMED, ICPC, and RCC (Cimino, 1995).

Computing a Categorization Scheme

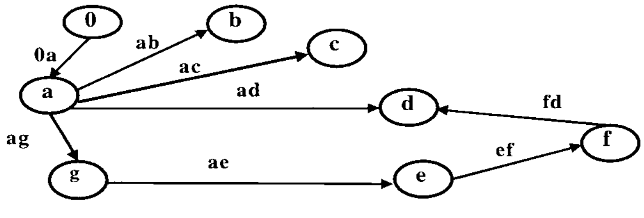
Our categorization algorithm uses, as a fundamental data structure, a directed acyclic graph (DAG) built to represent the hierarchical coding scheme under consideration. This DAG is referred to as a *categorization scheme* (CS) and, in the case of the ICD-9, is built as follows.

1. Let CS-9 be a categorization scheme corresponding to the ICD-9 alphabetical index.
2. To each entry point t_i in the alphabetical index associate a node n_i in the CS-9. These nodes have no ascendants and are called *root* nodes. We use the notation $t(n_i)$ to refer to the term associated with the node n_i.
3. To each term t_j that is hierarchically dependent on an entry point t_i also associate a node n_j in the CS-9. Insert a directed edge e_{ij} from the node n_i associated with e_i to the node n_j associated with t_j.
4. Repeat the step above recursively to the terms hierarchically dependent on the terms that have already been mapped into the CS-9.
5. Label each node in the CS-9 with the respective term in the ICD-9 alphabetical index.
6. To each edge e_{ij} assign all ICD-9 category (or subcategory) codes that are associated to the term t_j, hierarchical descendent of the term t_i, in the ICD-9 alphabetical index.
7. If a term t_i in the ICD-9 also includes a synonym (which appears immediately after the term, separated by a comma), label all edges e_{ij} with the synonym (this indicates that the synonym can be applied to the term t_i in the context defined by the pair of term-nodes (n_i, n_j)).

To illustrate, consider the CS-9 graph corresponding to the portion of the ICD-9 alphabetical index presented in Figure 2. This graph is portrayed in Figure 3. We notice that, in this particular case, there are no synonyms attached to the edges because they did not appear in Figure 2.

The Code Assignment Algorithm

Given the CS-9 graph for all the ICD-9 alphabetical index, our algorithm computes a *degree of confidence* (or rank) in the categorization of a medical document d_j into an



Where:

Nodes = {0 (Root), a (Cholera), b (Antimonial), c (Classic), d (El Tor), e (Vibrio), f (Cholerae), g (Undefined)}

Edges = {0a (Null; 001.9), ab (Null; 985.4), ac (Null; 001.0), ad (Null; 001.1), ae (Null; Null), ef (Null; 001.0), fd (Null; 001.1), ag (Null; 001.9)}

FIG. 3. CS-9 directed acyclic graph corresponding to the ICD-9 example in Figure 2.

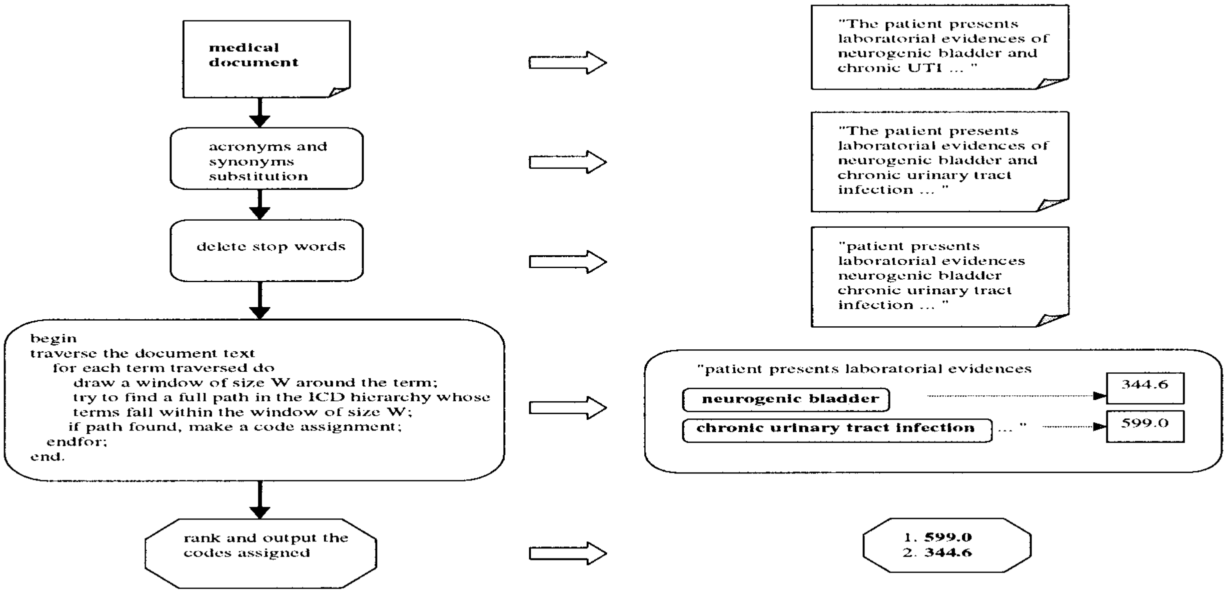

medical document
acronyms and synonyms substitution
delete stop words
begin
traverse the document text
for each term traversed do
draw a window of size W around the term;
try to find a full path in the ICD hierarchy whose terms fall within the window of size W;
if path found, make a code assignment;
endfor;
end.
rank and output the codes assigned
"The patient presents laboratorial evidences of neurogenic bladder and chronic UTI ... "
"The patient presents laboratorial evidences of neurogenic bladder and chronic urinary tract infection ... "
"patient presents laboratorial evidences neurogenic bladder chronic urinary tract infection ... "
"patient presents laboratorial evidences
neurogenic bladder
chronic urinary tract infection ... "
344.6
599.0
1. 599.0
2. 344.6

FIG. 4. General overview of our coding strategy. The adoption of sliding windows (and the requirement that coding paths must fall entirely within them) is a key feature that ensures increased precision regarding the final assignment of codes.

ICD-9 category or subcategory (i.e., ICD-9 codes) c_i. For this, it requires two additional data structures, acronyms and general synonyms, which we now define.

Definition 1 *An acronym dictionary is a set* A = {a_1, a_2, . . . , a_{na}}, na ≥ 0. *Each element* a_i *is a pair given by* ($acron_i$, str_i) *where* $acron_i$ *is a reference to an acronym and* str_i *is a string (composed solely by ICD-9 terms), which can be referred to by* $acron_i$.

Acronyms are useful because doctors use them frequently when writing down medical documents. In this work, we adopt the list of acronyms provided in the ICD-9 documentation.

Definition 2 *A synonym dictionary is a set* S = {s_1, s_2, . . . , s_{ns}}, ns ≥ 0. *Each element* s_i *is a pair given by* (syn_i, str_i) where syn_i *and* str_i *are synonym strings composed of ICD-9 terms.*

As acronyms, synonyms are useful because they are popular among doctors. In this work, we use a list of synonyms generated by the doctors of the Sarah Hospital Network.

Besides acronyms and synonyms, our categorization algorithm requires the notion of a *path,* as follows:

Definition 3 *A path* P *in the classification scheme CS-9 is a sequence of edges* $e_{k_1k_2}$, $e_{k_2k_3}$, . . . , $e_{k_{p-1}k_p}$ *in which* (a) *the first node* n_{k_1} *in the sequence is a root node, and* (b) *consecutive edges share a common end point* (*as indicated by the notation we adopt*). *The number* p *of nodes in the path* P *is its path length. Further, the edge* $e_{k_{p-1}k_p}$ *is called the terminal edge of* P.

Given the CS-9 graph, the acronyms, the synonyms, and the notion of a path, our categorization algorithm takes a medical document d_j and classifies it into one or more c_i categories (and subcategories) of the CS-9 graph. Figure 4 illustrates a high level view of our coding strategy.

Considering that each medical document is divided into sections, the algorithm operates as follows: For each section s_{kj} of the medical document d_j do

1. Substitute each acronym or synonym in the text of s_{kj} by its respective term string.
2. Traverse the text of s_{kj} sequentially looking for terms associated to the root nodes in CS-9.
3. For each root node found, determine the largest path length among all paths which satisfy the following condition: for each edge e_{ij} in the path, the terms $t(n_i)$ and $t(n_j)$ appear both in a same text window (belonging to the text of s_{kj}) of size W centered around $t(n_i)$.
4. Given the set of all largest paths found, let $p_{\min}$ be the smallest path length and $p_{\max}$ be the largest path length or $p_{\min} = p_{\max} - 1$, if $p_{\min} = p_{\max}$.
5. For each largest path P found (one per root node), let p be its path length and C_t be the set of codes (i.e., categories or subcategories) associated to its terminal edge; then, to each code $c_i \in C_t$ generate the code classification assignment (d_j, c_i, r) where the rank r is given by

$$r = \frac{p - p_{\min}}{p_{\max} - p_{\min}} \times (1 - \Delta) + \Delta \qquad (1)$$

The adoption of a text window of size W ensures that the paths traversed (in the CS-9 graph) are composed of nodes whose terms (i.e., the labels) are closely related in the medical document (i.e., these terms appear near one another in the text). For the test collection used in our experiments, a good value for W is 7. In the computation of the rank r, the parameter Δ is used as a threshold, which ensures that all ranks generated here are in the range $[\Delta, 1.0]$. The reason is that, as discussed in the next section, we also evaluate variations of our retrieval algorithm (which consider synonyms and approximate string matching) that are less precise and thus, retrieve a larger number of classification codes. To these codes, we assign ranks whose values are smaller than Δ. We observe that our computation of the r ranks is quite simple (the complexity is in finding the appropriate largest paths). Despite its simplicity, our ranking computation procedure is quite effective, as we now discuss.

Comparison With the Vector Space Model

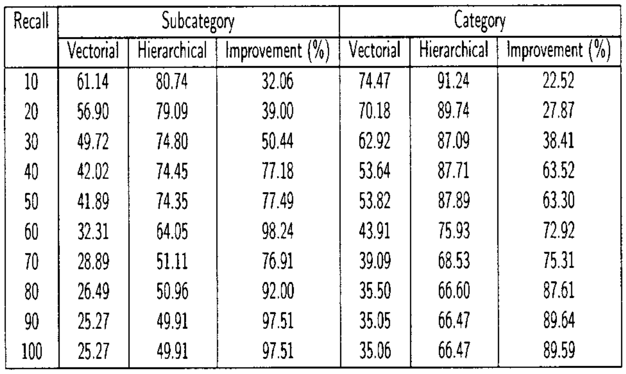
In Lima, Laender, and Ribeiro-Neto (1998), we compare our code assignment algorithm with the classic vector space model adapted to the task of assigning ICD-9 codes to medical documents. We now briefly summarize the results of that study.

The vector space model considers that documents and queries are indexed by keywords. To each keyword k_i in a document d is assigned a weight w_{id}, which is usually based on a *tf-idf* (i.e., term frequency and inverse document frequency) scheme (Salton & Buckley, 1988). To each keyword k_i in a query q is also assigned a weight w_{iq}. The similarity (or rank) $sim(d, q)$ of the document d with respect to the query q can be computed, for instance, by the cosine of the angle between the two vectors as given by Equation (2).

$$\mathrm{sim}(d, q) = \frac{\sum_{\forall i} w_{id} \times w_{iq}}{\sqrt{\sum_{\forall i} w_{id}^2} \times \sqrt{\sum_{\forall i} w_{iq}^2}} \qquad (2)$$

We can apply the vector space model to the ICD-9 code assignment problem as follows:

1. Each medical document is interpreted as a query Q, while the ICD-9 codes are viewed as documents to be retrieved. Thus, the document collection in this case is the set of all ICD-9 codes.
2. With each medical document is associated a vector of terms extracted from its text. Each acronym or synonym is replaced by its respective term string. All stop words are filtered out.
3. With each classification code c_i in the ICD-9 alphabetical index is associated a set of vectors of terms. One vector for each codepath leading from a root node to the code c_i. Each of these vectors is composed of the terms in the respective codepath. Again, all stop words are filtered out.
4. The weights are computed as *tf-idf* (i.e., within-document frequency and inverse document frequency) factors in standard fashion (Salton & Buckley, 1988).



Recall	Subcategory			Category		
	Vectorial	Hierarchical	Improvement (%)	Vectorial	Hierarchical	Improvement (%)
10	61.14	80.74	32.06	74.47	91.24	22.52
20	56.90	79.09	39.00	70.18	89.74	27.87
30	49.72	74.80	50.44	62.92	87.09	38.41
40	42.02	74.45	77.18	53.64	87.71	63.52
50	41.89	74.35	77.49	53.82	87.89	63.30
60	32.31	64.05	98.24	43.91	75.93	72.92
70	28.89	51.11	76.91	39.09	68.53	75.31
80	26.49	50.96	92.00	35.50	66.60	87.61
90	25.27	49.91	97.51	35.05	66.47	89.64
100	25.27	49.91	97.51	35.06	66.47	89.59

FIG. 5. Relative improvements in precision obtained by our code assignment algorithm (referred to as *hierarchical*) over an assignment generated using the vector space model (referred to as *vectorial*).

Because more than one vector of term weights might be assigned to a same classification code c_i (because, as discussed earlier, the ICD-9 alphabetical index might associate more than one codepath with c_i), we compute the similarity $sim(c_i, Q)$ between the code c_i and the medical discharge summary Q as the maximum of all similarities between Q and each of the term weight vectors associated with c_i.

For the set of experiments now described, we used 77 inpatient discharge summaries obtained from the Sarah Hospital Network, a national network of hospitals maintained by the Federal Government. Associated with each summary, there is a set of ICD-9 codes assigned by a group of coding specialists that is referred to as the *ideal code set.* By comparing the results of a coding algorithm with this set, we can make an assessment of its overall quality. In our study, this assessment is done through the standard precision and recall measures (Baeza-Yates & Ribeiro-Neto, 1999).

In the results below, our coding assignment algorithm (referred to as the *hierarchical* algorithm) is enriched with synonyms, acronyms, and the notion of approximate matching at the level of terms. Regarding synonyms and acronyms, their usage is the same both in our algorithm and in the vector-based coding algorithm. Regarding approximate string matching (i.e., *musc* provides an approximate match for the ICD-9 term *muscular*), this feature could not be used with the vector space model because the results deteriorated when it was considered (Lima, Laender, & Ribeiro-Neto, 1998). Figure 5 illustrates the relative improvements in precision obtained by our hierarchical algorithm. We notice that average precision at high recall levels (i.e., above 50%) is quite good. Also the relative improvements in precision at high recall levels provided by the hierarchical algorithm are considerable (both for category and subcategory coding).

We notice that average precision is around 25–35% for recall levels around 60–80%, when a vectorial ranking of codes is adopted. Results roughly of the same magnitude have been reported previously in the literature (Hersh, 1996; Hersh & Hickam, 1995b), for vector-based coding algorithms applied to subcollections composed of documents from MEDLINE (Haynes, McKiibbon, Walker, & Sinclair, 1990). While the document subcollections used in those experiments are rather distinct from the one we used in our experiments (and thus, a direct comparison is not possible), the results in both cases do suggest that vector-based coding tends to yield low average precision figures at high recall levels. We also observe that, at high recall levels, our hierarchical coding algorithm yields relative improvements in precision close to 100% (i.e., precision is almost doubled with regard to the results obtained by a vector-based coding algorithm). The key reasons for such gains in average precision are: (a) the adoption of a sliding text window as illustrated in Figure 4, and (b) the constraint that the coding is dependent on the recognition of a full path (in the ICD hierarchy) composed solely of terms that appear within a same sliding text window.

Experimental Results

In this section, we describe a set of experiments that we use to evaluate the retrieval performance of our categorization algorithm. The experiments described here use a database that includes more than 20,000 medical documents to be categorized. In this respect, they are thorough and conclusive when compared with the experiments in Lima, Laender, and Ribeiro-Neto (1998), which used only 77 medical documents.

Characterization of the Experiments

We use the ICD-9 alphabetical index (in Portuguese) and its category (and subcategory) codes to generate a corresponding CS-9 graph. In this structure, the largest path has size 18, and the number of distinct terms in the vocabulary is 14,078.

The medical documents used in our experiments are a set of 20,569 medical documents (inpatient discharge summaries, inpatient evolution clinicals and anamnesis—all in Portuguese) obtained from the Sarah Hospital Network. To each of these documents is assigned a set of category and subcategory codes (a task carried out by specialists in codification of the Sarah Hospital Network). We refer to the set of codes assigned by the specialists to a given document as its *ideal code set.*

Evaluation of our classification algorithm is done through recall and precision figures. These figures are generated by examining the codes (generated by our algorithm) according to their order of importance and verifying whether these codes are in the ideal code set. At any given point of this traversal, recall is the fraction of ideal codes that have been seen, while precision is the fraction of traversed codes that are in the ideal code set (Baeza-Yates & Ribeiro-Neto, 1999).

The characteristics of our experiments can be summarized as follows:

- Number of medical documents: 20,569.
- Documents for which no code was found: 918 (4.46%).
- Documents for which a code was found: 19,651 (95.54%).
- Documents for which at least 1 *ideal* category code was found: 17,358.
- Documents for which a code was found but with no *ideal* category code: 2,293.
- Documents for which no ideal category code was found: 3,211 (i.e., 918 + 2,293).
- Documents for which at least 1 *ideal* subcategory code was found: 15,709.
- Documents for which a code was found but with no *ideal* subcategory code: 3,942.
- Documents for which no ideal subcategory code was found: 4,860 (i.e., 918 + 3,942).
- Number of distinct ICD-9 terms (vocabulary): 14,078.
- Average number of ICD-9 terms per document: 15.
- Average time for coding a document: 44 miliseconds.

The expression *ideal code* is a reference to an ICD-9 code that was found automatically and appears in the ideal code set (i.e., the set of codes determined by the coding specialists). We notice that our algorithm could not found an ideal category code for 3,211 documents (i.e., 2,293 + 918). Also, the algorithm could not found an ideal subcategory code for 4,860 documents (i.e., 3942 + 918). For these documents, the average precision (for category and subcategory, respectively) is zero. We discuss this issue in great detail later on.

In our experiments, we ran a version of the coding algorithm with the following characteristics: (a) adoption of acronyms and synonyms derived from the daily work routine in a hospital of the Sarah Network and from the ICD Manual of Operation (CID-OMS, 1980), (b) usage of approximate term matching restricted to at most two editing errors or incomplete words (i.e., a word that is missing characters to its right is recognized even if the number of missing characters is greater than 2) (Wu & Manber, 1991), (c) usage of a categorization scheme (CS-9) that was built using the full ICD-9 alphabetical index, and (d) adoption of a text window of size *W* equal to 7 (determined empirically).

Analysis of the Results

We first examine the behavior of our coding algorithm without synonyms, acronyms, and approximate string matching. Figure 6 illustrates the results. When all documents are considered, average precision figures are around 60–70% for category coding and around 50–60% for subcategory coding. If we do not consider the documents for which no code was found (these documents can be separated and shown to a coding specialist), these average precision figures increase by roughly 10%, as can be observed in Figure 6.

It is important to notice that, in the case of Figure 6, the number of documents for which no code was found climbed up to 2,299. The utilization of synonyms, acronyms, and approximate string matching reduces the number of such

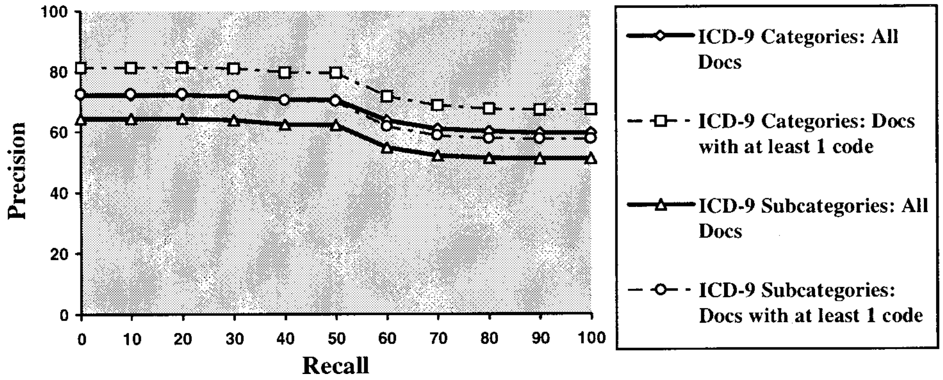


FIG. 6. Retrieval performance of our coding algorithm without considering synonyms, acronyms, and approximate string matching. Results for category and subcategory codes are displayed separately. In both cases, two situations are distinguished: (a) all documents considered, and (b) only documents for which at least one code was automatically generated are considered.

documents to 918, as discussed in the immediately following.

Figure 7 illustrates the retrieval performance of our code assignment algorithm when synonyms, acronyms, and approximate string matching are used. The results are displayed separately for two cases: category, and subcategory ICD-9 codes. In both cases, the average precision at all recall levels is quite good and always greater than 60%. The curve labeled "*All Docs*" includes all the 20,569 documents. For these, the average precision is between 70 and 80% for category coding and between 60 and 70% for subcategory coding. The curve labeled "*Docs with at least 1 code*" includes only the documents for which at least one code was found (i.e., these results exclude 918 documents that returned no code). This is reasonable to consider because documents without a code can be detected automatically and passed directly to a coding specialist. We notice that average precision figures increase slightly in this case.

By examining our results, we observed that a number of medical documents received codes generated automatically that included no ideal code. This happened both in the case of category coding as in the case of subcategory coding. For category coding, the number of documents without an ideal code assignment was 3,211 (15.61% of all documents) while for subcategories this number was 4,860 (23.63% of all documents). Notice that one or more codes were frequently found for each of these documents (only 918 documents returned no code whatsoever), but these codes were not specified by the coding specialists (and thus, are not part of the ideal code set for the document). Therefore, the average precision for these documents is always zero.

We proceeded by analyzing all the documents for which no ideal code was found and investigating what our coding algorithm could have done better. The table in Figure 8 summarizes the results of this analysis. The documents in the column labeled category sum up to 3,211, while the documents in the column labeled subcategory sum up to 4,860. As can be seen, we distinguish five different cases that we now discuss.

The case *no-code-found* includes all documents for which no ICD-9 code was found by our algorithm. This was due to the fact that the ICD-9 alphabetical index is not complete in the sense that it does not cover all the semantics of the medical world. Thus, there are inevitable cases for which it is not possible to do a code assignment based on the original ICD-9 index only. To deal with this problem, the

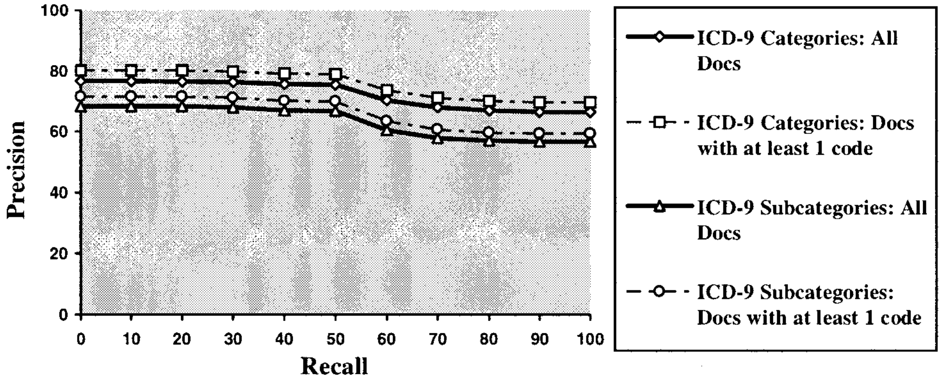
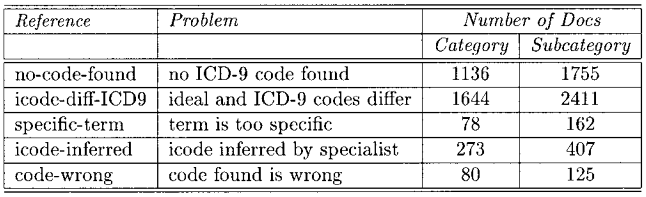


FIG. 7. Retrieval performance of our coding algorithm when synonyms, acronyms, and approximate string matching are used. Results are displayed separately for category and subcategory codes. In both cases, two situations are distinguished: (a) all documents considered, and (b) only documents for which at least one code was automatically generated are considered.



Reference	*Problem*	*Number of Docs*	
		Category	*Subcategory*
no-code-found	no ICD-9 code found	1136	1755
icode-diff-ICD9	ideal and ICD-9 codes differ	1644	2411
specific-term	term is too specific	78	162
icode-inferred	icode inferred by specialist	273	407
code-wrong	code found is wrong	80	125

FIG. 8. The five distinct cases for which the codes assigned by our algorithm yielded a precision figure equal to zero.

coding specialist usually introduces new entries (i.e., new terms and dependent terms for it) into the original ICD-9 index. These new entries then make the coding possible. Thus, the documents that fit in this case can only be assigned codes in a manual or semiautomatic fashion. A typical example in our test database is provided by a patient discharge summary in which the doctor specified that the patient suffered from *encephalomyelopathy mitochondrial,* while the corresponding entry in the ICD-9 alphabetical index is labeled as *encephalomyelitis* (which leads to the code 323.9). The specialist solves this case by adding an annotation to her ICD-9 alphabetical index that indicates that *encephalomyelopathy mitochondrial* constitutes an alternative path to the code 323.9. This indicates that the development of an interactive tool that assists the specialist in analyzing, editing, and modifying the alphabetical index can reduce this problem drastically.

The case *icode-diff-ICD9* includes all documents for which the ideal code set has no intersection with the codes indicated by the ICD-9 alphabetical index. This happens in four distinct situations: (1) the specialist is wrong, (2) the specialist did not find the default code, (3) the specialist indicated a code that is not supported by the medical document, and (4) the specialist opted for a code distinct from the code most indicated by the ICD-9 index. A typical example of the first situation is as follows: the patient discharge summary indicates a diagnosis of *mielophaty cervical,* a well-known patology of the cervical spine (related to the ICD-9 subcategory codes 336.3 and 721.1), but the specialist chose the subcategory code 336.9 related to a *nonspecified disease of the spinal column.* Thus, the specialist is simply wrong in this situation. In the second situation, the patient discharge summary does not indicate explicitly what is the subcategory (e.g., the doctor indicates a diagnosis of *osteomyelitis* without saying whether it is of the type *chronic, acute,* or *purulent*)—a case for which the ICD-9 recommends the adoption of a default code—and the specialist chose a code that is not the default one. In the third situation, the specialist chose a code that is not supported by the patient discharge summary. For instance, the doctor wrote a diagnosis of *nontraumatic paraplegia* and the specialist chose a code related to *encephalities.* In the fourth situation, the ICD-9 alphabetical index points to a preferred code, but the specialist chose a different (but related) one. For instance, the specialist chose the code 368.9 related to *visual disturbances,* while the ICD-9 index indicated that the preferred code is 369.2 related to *subnormal vision in both eyes.* We observe that an interactive tool that automatically suggests ICD-9 codes to the specialists will greatly reduce the mistakes in the first, second, and fourth situations described above. Further, it can also reduce the problems in the third situation because the specialist will not feel compelled to *guess* a code (because the algorithm suggests one).

The case *specific-term* includes all documents for which finding the correct code requires interpreting the meaning of a term (written by the doctor) that is quite specific, i.e., it requires a process of semantic generalization. A typical example is as follows: the doctor indicates a diagnosis of *fracture of the spinal column L1-L2,* which is the same as *fracture of the lumbar spine.* Cases such as this are not currently considered by our coding algorithm, but could be dealt with by adding a semantic network to the model for modeling specialized concepts.

The case *icode-inferred* includes all documents for which the coding specialist uses clinical, radiological, and laboratorial pieces of evidence to deduce an ICD-9 code. This is a situation in which the presence of a human specialist is absolutely required.

The case *code-wrong* includes all documents for which the code assigned by our algorithm is incorrect. This occurs, for instance, because the whole evidence required for the coding is composed of terms that are located far apart in the patient discharge summary. An example of this is as follows: the doctor wrote a diagnosis of "*sequel of fracture of the 2, 3 e 4 finger of hand,*" whose correct parsing requires a window of size 11 while the window we adopted has size 7. Notice that we cannot simply increase the window size to 11 in general because this could cause ambiguity for the coding of the majority of the documents. In fact, the window size we adopted was determined experimentally, and works quite well for documents in Portuguese.

From this analysis, we conclude that a semiautomatic coding procedure in which ICD-9 codes are automatically suggested by our algorithm and subsequently verified by a specialist will provide great gains in the time spent with the coding task and in the quality of the sets of codes generated.

Conclusions

In this article, we thoroughly analyzed the retrieval performance of an algorithm that we proposed for the automatic categorization of medical documents. We first summarized previous experimental results (based on a small database) that show that the algorithm considered is far superior to an alternative coding algorithm based on the classic vector model. Following, we presented a detailed analysis of experimental results with a much larger medical database. In our experiments, we considered a collection of 20,569 medical documents to be categorized. To facilitate interpreting the results, we divided our analysis in two distinct scenarios: the computation of category codes, and the computation of subcategory codes.

In the computation of category codes, we observed that our algorithm categorized 19,651 documents with an average precision between 70 and 80% for all recall levels, which is an excellent retrieval performance. Of these, 2,293 documents were poorly categorized which resulted in a precision equal to zero (i.e., no code indicated by the specialists was found by our algorithm). Further, our algorithm was completely unable to find even a single code for 918 documents. For the 3,211 (i.e., 2,293 + 918) documents that were not categorized properly, we provided a complete analysis for the discrepancy between the set of codes generated by the specialists and the set of codes computed by our algorithm. We noticed, for instance, that in a number of cases (to be exact, in 589 documents or 2.86% of all documents) the specialists simply provided codes that were incorrect. In the case of the other 391 documents (1.9% of all documents), the specialists provided codes that were not directly supported by the text of the medical document (this might indicate, for instance, that the specialists obtained additional information on the diagnosis or disease). In 158 cases (a slim 0.77% of all documents), our algorithm made assignments of category codes that were incorrect. Thus, our algorithm makes incorrect (category code) assignments less frequently than the human specialists. Almost all the remaining cases represented situations in which an ICD-9 code cannot be assigned automatically without the assistance of a human subject. These cases include: the ICD-9 index is not complete (in the sense that not all medical semantics is represented within the index), the specialists opted for a code which, even if not incorrect, is distinct from the default code recommended by the index, the specialists used knowledge about the semantics of specific terms that does not appear in the index, and the specialists deduced the code using additional information on the text of the medical document. In all these cases, the difficulties in finding a proper code can be greatly alleviated by the use of an interactive tool, such as SAPHIRE (Hersh & Greenes, 1990), that assists the user with the codification task. We are currently working in the implementation of one such tool. This tool will allow, for instance, online editing of the ICD-9 alphabetical index (to include new terms and concepts).

In the computation of subcategory codes, we observed that our algorithm categorized 19,651 documents with an average precision between 60 and 70% for all recall levels, which is, again, an excellent retrieval performance. Of these, 3,942 documents were poorly categorized, which resulted in a precision equal to zero. Further, no subcategory code was found for 918 documents. For the 4,860 (i.e., 3,942 + 918) documents that were not categorized properly, the problems observed were analogous to those found in the assignment of category codes. In the case of 1,131 documents (or 5.5% of all documents), the specialists provided subcategory codes that were incorrect. Comparatively, our algorithm assigned subcategory codes that were improper in the case of 288 documents (or 1.4%). Thus, our algorithm makes incorrect (subcategory code) assignments less frequently than the human specialists.

Our experimental evaluation indicates that the algorithm we propose is very effective for the task of automatic categorization of medical documents. Further, our results demonstrate that it is reasonable to expect more mistakes from human coding specialists than from the algorithm.

Acknowledgments

This work was partially supported by FINEP/PRONEX under Project SIAM, grant number 76.97.1016.00.

We would like to thank Tania Rozalina, Olimpia Leal, and Rita Coelho, coding specialists of the Sarah Hospital in Belo Horizonte, for their advice on the ICD code assignment problem. Data used in our experiments have been made available to us by the Sarah Hospital Network. We also thank two anonymous referees, whose comments allowed us to considerably improve the article.

References

Aronow, D.B., Soderland, S., Ponte, J.M., Feng, F.F., Croft, W.B., & Lehnert, W.G. (1995). Automated classification of encounter notes in a computer based medical record. In Proceedings of the eighth world congress on medical informatics (pp. 8–12). Vancouver, Canada.

Baeza-Yates, R., & Ribeiro-Neto, B. (1999). Modern information retrieval. Harlow, England: Addison Wesley Longman.

Chute, C.G., Cohn, S.P., Campbell, K.E., Oliver, D.E., & Campbell, J.R. (1996). The content coverage of clinical classifications. Journal of the American Medical Informatics Association, 3(3), 224–233.

CID-OMS. (1980). Classificação internacional de doenças, revisão 9 (Volumes 1–2). São Paulo, Brazil (in Portuguese): Organização Mundial da Sáude.

Cimino, J.J. (1994). Data storage and knowledge representation for clinical workstation. International Journal of Biomedical Computing, 34, 185–194.

Cimino, J.J. (1995). Vocabulary and health care information technology: State of the art. Journal of the American Society for Information Science, 46(10), 777–782.

Delamarre, D., Burgun, A., Seka, L.P., & Beux, P.L. (1995). Automated coding of patient discharge summaries using conceptual graphs. Methods of Information in Medicine, 34(4), 345–351.

Friedman, C., Alderson, P.O., Austin, M., Cimino, J.J., & Johnson, S.B. (1994). A general natural-language text processor for clinical radiology. Journal of the American Medical Informatics Association, 1(2), 161–174.

Graham, I., & Jones, P.L. (1988). Expert system: Knowledge, uncertainty and decision. New York: Chapman and Hall.

Haynes, R.B., McKibbon, K.A., Walker, C.A., & Sinclair, J.C. (1990). On-line access to MEDLINE in clinical setting. A study of use and usefulness. Annals of Internal Medicine, 112(1), 78–84.

Hersh, W.R. (1995). The electronic medical record: Promises and problems. Journal of the American Society for Information Science, 46(10), 772–776.

Hersh, W.R. (1996). Information retrieval—A health care perspective. New York: Springer-Verlag (Series: Computer and Medicine).

Hersh, W.R., & Greenes, R.A. (1990). SAPHIRE—An information retrieval system featuring concept matching, automating indexing probalistic retrieval, and hierarchical relationships. Computers and Biomedical Research, 23, 410–425.

Hersh, W.R., & Hickam, D.H. (1995a). Information retrieval in medicine: The SAPHIRE experience. Journal of the American Society for Information Science, 46(10), 743–747.

Hersh, W.R., & Hickam, D.H. (1995b). An evaluation of interactive Boolean and natural language searching with an online medical and textbook. Journal of the American Society for Information Science, 46(7), 478–489.

Hertz, J., Krogh, A., & Palmer, R.G. (1991). Introduction to the theory of neural computation. Reading, MA: Addison Wesley.

Larkey, L.S., & Croft, W.B. (1996). Combining classifiers in text categorization. In Proceedings of the 19th annual international ACM-SIGIR conference on research and development in information retrieval (pp. 289–297). Zurich, Switzerland.

Lima, L.S.R., Laender, A.H.F., & Ribeiro-Neto, B.A. (1998). A hierarchical approach to the automatic categorization of medical documents. In Proceedings of the 1998 ACM-CIKM international conference on information and knowledge management (pp. 132–139). Bethesda, MD.

MeSH. (1994). Medical subject heading: tree structure and alphabetic—The library. Bethesda, MD: National Library of Medicine.

Mladenic, D., & Grobelnik, M. (1999). Feature selection for unbalanced class distribution and naive bayes. In Proceedings of the 16th international conference on machine learning (pp. 258–267). Bled, Slovenia.

Pellegrin, L., Bastien, C., & Roux, M. (1994). Representation of medical concepts of thyroid gland by physicians in anatomy and pathology. Methods of Information in Medicine, 33(4), 382–389.

Pietrzyk, P.M. (1991). A medical text analysis system for german—Syntax analysis. Methods of Information in Medicine, 30(4), 275–283.

Rajashekar, T.B., & Croft, W.B. (1995). Combining automatic and manual index representation in probabilistic retrieval. Journal of the American Society for Information Science 46(4), 272–283.

Rothwell, D.J., Cote, R.A., Cordeau, J.P., & Boisvert, M.A. (1993). Developing a standart data structure for medical language—The SNOMED proposal. In Proceedings of the 17th annual symposium on computer application in medical care (pp. 695–699). Washington, DC.

Sager, N., Lyman, M., Bucknall, C., Nham, N., & Tick, L.J. (1994). Natural language processing and the representation of clinical data. Journal of the American Medical Informatics Association, 1(2), 142–160.

Sager, N., Lyman, M., Nhan, N.T., & Tick, L.J. (1995). Medical language processing: applications to patient data representation and automatic encoding. Methods of Information in Medicine, 34(1), 140–146.

Salton, G., & Buckley, C. (1988). Term-weighting approaches in automatic retrieval. Information Processing & Management, 24(5), 513–523.

Satomura, Y., & Amaral, M. (1992). Automated diagnostic indexing by natural language processing. Medical Informatics, 17(3), 149–163.

Sebastiani, F. (1999). A tutorial on automated text categorisation. In First Argentinian symposium on artificial intelligence (pp. 7–35). Buenos Aires, Argentina.

Spyns, P. (1996). Natural language processing in medicine: An overview. Methods of Information in Medicine, 35(4), 285–301.

Tuttle, M.S., Olson, N.E., Keck, K.D., Cole, W.G., Erlbaum, M.S., Sheretz, D.D., Chute, C.G., Elkin, P.L., Atkin, G.E., Kaihoi, B.H., Safran, C., Rind, D., & Law, V. (1998). Metaphase—An aid to the clinical conceptualization and formalization of patient problems in healthcare enterprises. Methods of Information in Medicine, 37(5), 373–383.

UMLS. (1994). UMLS—Knowledge sources 5th experimental edition. Bethesda, MD: National Library of Medicine.

Wehri, E., & Clack, R. (1995). Natural language processing, lexicon and semantics. Methods of Information in Medicine, 34(1), 68–74.

Wu, S., & Manber, U. (1991). Fast text searching allowing errors. Communications of the ACM, 35(10), 83–91.

Yang, Y., & Chute, C. (1994). An application of expert network clinical classification and MEDLINE indexing. In Proceedings of the 18th annual symposium on computer applications in medical care (pp. 157–161). Washington, DC.