Medical Document Categorization Using *a Priori Knowledge*

Lukasz Itert[1,2], Włodzisław Duch[2,3], and John Pestian[1]

[1] Department of Biomedical Informatics, 3333 Burnet Avenue, Children's Hospital Research Foundation, Cincinnati, OH 45229, USA
[2] Department of Informatics, Nicolaus Copernicus University, Toruń, Poland
[3] School of Computer Engineering, Nanyang Technological University, Singapore

Abstract. A significant part of medical data remains stored as unstructured texts. Semantic search requires introduction of markup tags. Experts use their background knowledge to categorize new documents, and knowing category of these documents disambiguate words and acronyms. A model of document similarity that includes *a priori* knowledge and captures intuition of an expert, is introduced. It has only a few parameters that may be evaluated using linear programming techniques. This approach applied to categorization of medical discharge summaries provided simpler and much more accurate model than alternative text categorization approaches.

1 Introduction

The dream of semantic Internet populated with documents annotated with XML tags remains a difficult challenge. Automatic tools that convert unstructured textual data into semantically-tagged documents are still elusive. In the medical domain the need to create these tools is acute because errors may be costly, medical vocabularies are abbreviations and acronyms are rampant. Critical differences between General English and Medical English have been analyzed in a numbers of publications [1]. The "Discovery System" (DS) data repository [2] at the Cincinnati Childrens Hospital Medical Center (CCHMC), a large pediatric academic medical center with over 700,000 pediatric patient encounters per year, contains terabytes of medical data, mostly in form of raw texts, stored in a complex, relational database integrating many electronic hospital services.

The long-term goal of our research is to create tools that automatically annotate unstructured medical texts, adding full information about all medical concepts, ambiguous terms, expanding acronyms and abbreviations, using a variety of statistical and computational intelligence algorithms to achieve this goal. The first step towards full semantic annotation and disambiguation of medical text requires discovery of the document topic, for example the main disease that has been treated. It is clear that medical expert reading a given text quickly forms a hypothesis about the particular sub-domain the text belongs to and interprets the text in the light of the background knowledge derived from medical studies, textbooks and individual experience. This is especially true if relatively short

W. Duch et al. (Eds.): ICANN 2005, LNCS 3696, pp. 641–646, 2005.

texts, such as patient's discharge summaries, containing brief medical history, current symptoms, diagnosis, treatment, medications, therapeutic response and outcome of hospitalization, are analyzed. Many medical concepts appear very rarely in such short documents, therefore document categorization algorithms that ignore background medical knowledge make many errors.

In the next section a model trying to capture expert intuition in document categorization is introduced and a simple way to take the *a priori* knowledge into account proposed. Estimation of parameters of this model is done using linear programming techniques. Numerical experiments with over 4500 discharge summaries were made to compare this approach with standard document categorization methods.

2 Model of Similarity

Documents D_j of length l_j are composed of terms (words, collocations or concepts). Term frequencies tf_{ij} for term $i = 1 \ldots n$ in document j are calculated for all documents, and transformed to obtain features that help to reflect document similarity. Weights of features that appear with high frequency, or are derived from longer documents, should be reduced using logarithmic or square root functions. Uniqueness of each feature is inversely proportional to the number of documents this feature appears in; if the term i appears in df_i out of N documents weighting for non-zero term frequencies may be calculated as [3]:

$$s_{ij} = (1 + \log tf_{ij}) \log N/df_i \tag{1}$$

In the tf $\times$ idf weighting scheme additional scaling is used, for example [3]:

$$s_{ij} = \text{round}\left(10 \times \frac{1 + \log tf_{ij}}{1 + \log l_j} \log \frac{N}{df_i}\right) \tag{2}$$

In document categorization distribution of a given term among different categories is important, therefore the logarithm of ratio $\log(K/cf_i)$ of the number of classes K to the number of classes cf_i in which term i appears, should be used in the above equation. To avoid favoring long documents all vectors $(s_{1j}, \ldots s_{nj})$ may be divided by their length to obtain final feature vectors $\mathbf{x}_{ij}$, for example:

$$\mathbf{x}_{ij} = (1 + \log tf_{ij}) \log N/df_i; \qquad \mathbf{x}_j = \mathbf{s}_j/||s_j|| \tag{3}$$

This normalization tends to favor shorter documents. More sophisticated normalization methods have been introduced to counter this effect, but unbiased normalizations are hard to find.

Such *ad hoc* term weights do not take into account *a priori* knowledge. Before the document is examined the probability that it belongs to category C_k should be equal to the prior probability $p(C_k)$. The background knowledge about reference documents from class C_k may be represented using weighted frequencies $R_{ik} = R_k(tf_i)$ for the term i. These frequencies are collected in the reference vector R_k (more than one vector per class may be needed). The following algorithm seems to capture human intuitions of the document categorization process:

1. Initial distance between document D and the reference vectors R_k should be proportional to $d_{0k} = ||D - R_k|| \propto 1/p(C_k) - 1$.
2. If a term i appears in R_k with frequency $R_{ik} > 0$ but does not appear in D the distance $d(D, R_k)$ should increase by $\Delta_{ik} = a_1 R_{ik}$.
3. If a term i does not appear in R_k but it has non-zero frequency D_i the distance $d(D, R_k)$ should increase by $\Delta_{ik} = a_2 D_i$.
4. If a term i appears with frequency $R_{ik} > D_i > 0$ in both vectors the distance $d(D, R_k)$ should decrease by $\Delta_{ik} = -a_3 D_i$.
5. If a term i appears with frequency $0 < R_{ik} \leq D_i$ in both vectors the distance $d(D, R_k)$ should decrease by $\Delta_{ik} = -a_4 R_{ik}$.

Coefficients $a_1, \ldots a_4 > 0$ are adaptive constants. If a term appears in both D and R than the distance is decreased by a constant times the smaller of the two frequencies, because for small term frequencies this situation may happen by pure chance. A term that appears only in documents from the C_k class should be more important for this class than terms appearing in all classes, therefore term specificity is given by the class-conditional probability $p(i|C_k) = p(tf_i > 0|C_k)$. Given the document D, and reference vector R_k, probability that the class is C_k should be proportional to:

$$S(C_k|D; R_k) = 1 - \sigma\left(\beta\left[d_{0k} + \sum_i p(i|C_k)\Delta_{ik}\right]\right) \tag{4}$$

Here Δ_{ik} depends on adaptive parameters $a_1, \ldots a_4$ that may be specific for each class, and the distance depends on the d_{0k} which may also be treated as an adaptive parameter; the slope β is an additional parameter, giving 6 adaptive parameters per class. Weighted distance contributions may sum to a negative number therefore a logistic function $\sigma(\cdot)$ is used. Probabilities are estimated after softmax normalization $p(C_k|D; R_i) = S(C_i|D; R_i)/\sum_k S(C_k|D; R_k)$.

This approach seems to capture some human intuitions when texts are analyzed using background knowledge. Parameters $a_1, \ldots a_4$ may be estimated using neural networks with RBF-like architecture and $S(C_i|D; R_i)$ functions (4) in each hidden node i, and a soft-max function for the output node. An alternative is to use linear programming techniques for parameter optimization, solvable in polynomial time using interior point based methods. PCx algorithm has been used here [6]. Condition

$$d_{0k} + \sum_i p(i|C_k)\Delta_{ik} = \min \tag{5}$$

maximizes similarity between documents and reference vectors, Eq. 4, and should be used with the following constraints:

$$\sum_i p(i|C_j)\Delta_{ij} - \sum_i p(i|C_k)\Delta_{ik} \geq d_{0k} - d_{0j}; \quad k \neq j = 1 \ldots K \tag{6}$$

where k indicates the correct class. For all N training vectors (documents) $K - 1$ constraints are created. Two cases have been considered: a common set of $a_1, \ldots a_4$ parameters for all classes, and a separate set for each class. Satisfying

all $K-1$ inequalities for one document D guarantees that its similarity measure (4) is maximal for the correct class and provides correct classification.

3 Numerical Experiments

Customized SQL queries were created to retrieve discharge summaries from the database. Overall 4534 patients discharge summary records were used. All documents are short, less than 3000 characters, with the average length below 2000 characters. They are labeled by 10 distinct disease names, with "asthma" being the majority class that covers 19.1%, followed by Epilepsy (14.1%), Pneumonia (13.4%), Gastroenteritis (12.9%), Anemia (12.0%), Otitis media (10.8%), Urinary tract infection (UTI) (6.6%), Cystic fibrosis (6.2%), Cerebral palsy (3.9%), and the Juvenile Rheumatoid Arthritis (JRA) with 0.9%. Except for the last class that contained only 41 documents all the other classes were among the most common in the database containing discharge records.

The name of the disease used as the category label plays a dual role: it is one of the features used to describe the document, and it is also the class label. For example, documents from the "asthma" class frequently contain the name "asthma" as a part of some concept (such as "allergic asthma"), but they may also contain the names of other diseases. The frequency of appearance of each of the 10 disease names in the documents may be taken as an indicator of the class, giving a more informed base rate distribution. Using this approach leaves 55.3% of documents unclassified (including ties with several identical highest frequencies), 34.6% correctly classified and 10.1% errors.

To define the feature space each record has to be subject to several text processing techniques: exhaustive sets of parsing rules are used to handle punctuation issues and stop-word list of common English words to remove words that do not contribute to document categorization. MetaMap Transfer (MMTx) program package [5] has been used to discover UMLS Metathesaurus concepts [4] in these texts. To prevent any false-positive mapping a very restrictive MMTx settings has been used during string matching. Concepts are assigned to 135 semantic types, but only 26 types representing specific, medical concepts were found useful for document categorization. They include anatomical structures, body parts, functions, biological organisms, drugs and pharmacological substances, clinical procedures, disease and syndromes, symptoms, and test results. Using the UMLS ontology as a base all common words may be filtered out, and all unnecessary medical terms excluded. The final number of features included in the "native" space based on concepts discovered in medical records was 7220.

The reference texts were taken from MedicineNet [7], Children's Hospital Boston Child Health A to Z [8], and MedlinePlus: Medical Encyclopedia [9]. Documents describing each of the 10 selected diseases have been processed and 1097 unique UMLS concepts have been identified. In the discharge summaries only 807 of these concepts appeared and these concepts have been used as the feature space. Background knowledge contained in features that appear only in the reference space, but not in the limited selection of medical records taken

for analysis, could be useful in future for categorization of new texts. Discharge summaries contain many more UMLS concepts than reference texts, but in most cases there is little or no correlation between names of these additional concepts and diseases. Thus *a priori* knowledge helps in feature selection and definition of the feature space.

All calculations presented below were done using 10-fold crossvalidation. Features were based on term frequencies (M0), binary present/absent values, and 4 popular weighting schemes [3]. Poor results of the SSV decision tree [10] (similar results are obtained with C4.5 tree) show that the similarity-based approach may be more appropriate here, and that the reference vectors containing *a priori* knowledge may help. To check how the nearest neighbor classifier performs using a single reference vector per class kNN with a cosine distance function has been used [3]. Direct application of Euclidean distance has no sense because reference vectors have different norms, and the shortest one will almost always be the closest (accuracies are between 6-15%). Best accuracy is obtained with unweighted term frequencies (60.1%), worst accuracy with binary vectors (43.8%) and 56-59% accuracy with M2-M5 tf weightings. A very large neural network is needed (300 neurons and ~ 250 thousand parameters) to reach 71-72% accuracy on this data. SVM has never given such good results, with Gaussian kernel results at the level of 40% only and linear kernels in the range of 60%. Standard deviation was between 1.5-2.5%.

The approach described in Sec. 2 has been used to calculate coefficients $a_1, \cdots a_4$ in each crossvalidation using linear programming techniques. For each test vector these coefficients were used to compute similarity to all 10 reference vectors, selecting the highest similarity as class indicator. In the first case the same coefficients were used for each class. Parameter $\beta = 0.01$ was used, making the logistic transformation almost linear; higher values of β lead to sharp increase in the number of ties. For each crossvalidation (CV) step on average around 95% of all constraints were satisfied, however the number of vectors for which all constraints were fulfilled was only 61%, leading to the classification accuracy of 61.1%. Optimizing coefficients $a_1, \cdots a_4$ separately for each class decreased the percentage of all satisfied constraints to 92%, but increased the number of vectors for which all constraints were fulfilled by approximately 10%. The final CV accuracy was then 71.6±2.1% with tf frequencies and similar for various scalings. This is quite remarkable for a system with 4 parameters per class, considering the improvement over standard feature weighting techniques, and the size of the MLP network needed to reach similar results. Prototypes generated

Table 1. 10-fold crossvalidation accuracies in % for different feature weightings. M0: tf frequencies; M1: binary data; M2: $\sqrt{tf}$, M3: $1 + \log(tf)$, M4: Eq. (1); M5: Eq. (2).

	M0	M1	M2	M3	M4	M5
kNN	48.9	50.2	51.0	51.4	49.5	49.5
SSV	39.5	40.6	31.0	39.5	39.5	42.3
MLP (300 neurons)	66.0	56.5	60.7	63.2	72.3	71.0
SVM (C opt)	59.3 (1.0)	60.4 (0.1)	60.9 (0.1)	60.5 (0.1)	59.8 (0.01)	60.0 (0.01)
10 Ref. vectors	71.6	–	71.4	71.3	70.7	70.1

using LVQ method from all training data (one prototype/class) gave 66.3±1.6% using the same method, showing importance of the *a priori* knowledge.

It is also worth noting that the whole calculations for linear programming with prototypes on a 3.6 GHz PC took about 1.5 hour, while SVM with Gaussian kernel (with optimized C=10 and dispersion=0.1) or MLP takes more than 10 times longer. Linear SVM takes twice as much time and is much less accurate (calculations were done using the GhostMiner package [10]).

4 Conclusions

Categorization of documents should be treated as the first step towards full annotation, facilitating subsequent disambiguation of terms and concepts . Medical texts are very specific, containing very large number of unique concepts. Standard approach to the document classification, based on vector representation using the tf×idf weighting scheme [3] leads to quite poor results using the nearest neighbor and decision trees approaches. Knowledge contained in medical records, such as the discharge summaries analyzed here, is by itself not sufficient to categorize them. Therefore reference texts have been introduced, systematically describing each disease documents can be classified to. New approach to the term weighting and evaluation of similarity of documents that refers to the background knowledge and that seems to capture human intuitions has been presented and tested on medical records.

Even the simplest implementation of a prototype-based classifier with linear programming for optimization of parameters reported here gave substantial improvement in accuracy. Background knowledge should obviously be stored in more than one prototype. Finding the simplest decomposition of medical records into classes using either sets of logical rules or minimum number of prototypes, is an interesting challenge. The approach presented here seems to be a step in right direction.

References

1. D. Campbell, S.B. Johnson, Comparing syntactic complexity in medical and non-medical corpora. Proc. of the AMIA Annual Symposium, 2001: 90-5.
2. J. Pestian, B. Aronow, K. Davis, *Design and Data Collection in the Discovery System.* Proc. Int. Conf. on Math. and Eng. Techniques in Medicine and Biological Sciences, CSREA Press, Providence, USA 2002.
3. C.D. Manning and H. Schütze, Foundations of Statistical Natural Language Processing MIT Press, Cambridge, MA 1999.
4. UMLS: http://www.nlm.nih.gov/research/umls
5. MetaMap: http://mmtx.nlm.nih.gov
6. J. Czyzyk, S. Mehrotra, M. Wagner and S. J. Wright: PCx: An Interior-Point Code for Linear Programming, Optim. Method. Softw. 12 (1999) 397-430.
7. MedNet: http://www.medicinenet.com
8. C.H. Boston: http://web1.tch.harvard.edu/cfapps/A2Z.cfm
9. Medline Plus: http://www.nlm.nih.gov/medlineplus/encyclopedia.html
10. GhostMiner: http://www.fqspl.com.pl/ghostminer/