Automatic Text Classification of Consumer Health Web Sites Using WordNet

Patrick Herron

INLS 170

Submitted 06 December 2005

Text categorization is the assignment of documents or other text units to predefined categories. We examine the effect of using WordNet features for feature representation on automatic classification performance for a binary classification problem. The input document set is a collection of consumer health web sites, and representations of that document set were constructed using the UMLS's SPECIALIST lexicon as well as WordNet features. While the use of WordNet features, particularly hypernymy, hold promise for feature representation building for text classification, human inter-rater reliability statistics would need to be acquired before the relative success of automatic text classification with the current domain can be determined. Further, given the flexibility of SVMs with respect to dimensionality of a problem space, additional experiments should be run without some of the feature reduction steps taken at early stages of the experiment.

1 Introduction

1.1 North Carolina Health Info

The North Carolina Health Info (NCHI) project (http://www.nchealthinfo.org/), a National Library of Medicine-funded undertaking run jointly by the School of Information and Library Science at UNC-Chapel Hill and the UNC Health Sciences Library, offers to the public a web portal of more than 2500 health-related web sites serving North Carolina. Sites added to the portals catalog are cataloged by local service term first and then by health topic. The cataloging tasks are currently executed by graduate students and are performed manually; adding even a single site to the NCHI catalog is labor-intensive. The costs of manual classification will grow out of control as the number of North Carolina-based health websites proliferate. NCHI hopes to move their cataloging model to a topic-first model and are interested in the automation of as much of the topic classification task as possible in order to reduce labor and possibly improve the quality of cataloging by topic.

1.2 Thesis

Automation of the NCHI cataloging task may be made possible by the use of automatic text classification. While research has shown it possible to automate the construction of

internet portals using text classification techniques (McCallum, Nigam, Rennie, & Seymore, 2000), little is known about doing so with health related collections. It is similarly unknown how well the NCHI's collection of web-based corpora will lend itself to automatic classification. The NCHI collection does offer an opportunity to test different text encoding schemes for automatic text classification.

The present study focuses on text classification at the web-document level. The objective of the present study is to examine the effect of different NLP-based document encoding schemes (also called representation schemes, or representations, for short) on the performance of automatic classification according to a binary topic class set. More specifically, the aims of the present study are two-fold: firstly, to demonstrate that automatic binary topic classification of the NCHI web documents can be performed using elements of two different knowledge sources, the Unified Medical Language System, and WordNet, to encode the documents; and, secondly, that the encoding of documents in terms of WordNet features significantly improves and/or otherwise benefits the performance of automatic binary topic-based document classification over automatic classification of the same documents without using WordNet features in the representation.

The working hypothesis is that a bag of words representation loses crucial semantic features relevant to topic classification, and that using WordNet synsets to represent each term will infuse a representation with term polysemy and return relevant semantic features otherwise lost in any bag of words representation. While it is suspected that the

explosion of dimensionality and ambiguity implied in re-representing terms with multiple corresponding synsets will lead to a trade-off with the benefits of adding semantic information, it is believed that using parent and possibly grandparent hypernyms to represent the synsets will offset the loss in performance due to the addition of irrelevant polysemous features and will provide a representation that is more compact and more tuned to a highly general topic-based classification.

2 Background

2.1 Context

After more than a half-century of computers and over a decade of the existence of the World Wide Web, vast amounts of digital information now exist. Lesk (1997) estimated the size of the Library of Congress Collection alone to be over 3 petabytes as of 1997; it is likely that the amount of ACII text on the World Wide Web has eclipsed 3 petabytes years ago (Lesk, 1997). Regardless of the exact figures we may safely say that there is a vast amount of digital text on-line. Organizing such a vast amount of material, if performed manually, is likely to be an impossible task given a reasonable amount of time and effort. The ability to automatically classify digital text documents offers the means to browsing, searching, and sorting vast amounts of digital data.

2.2 Classification

"Classification or categorization is the task of assigning objects from a universe to two or more classes" (Manning & Schütze, 2003, p. 575). Many natural language tasks such as word sense disambiguation can be considered classification tasks (Manning & Schütze, 2003, p. 575).

In general when classifying items according to a classification system, an item should be

first organized along a topic-subject axis (Taylor, 2000, p. 278). The NCHI cataloging model therefore essentially follows this basic principle.

2.3 Machine learning & automatic classification

Machine learning is defined in the following way:

> *A computer program is said to* ***learn*** *from experience* E *with respect to some class of tasks* T *and performance measure* P*, if its performance at tasks in* T*, as measured by* P*, improves with experience* E (Mitchell, 1997, p. 2).

Automatic or statistical classification may be generally described as follows: a set of objects called a *training set*, where each object is assigned to a member of a class set where the class set contains two or more members, is represented or encoded in a data representation model (Manning & Schütze, 2003, pp. 575-6). Automatic classification according to a predefined class is known as *supervised learning* (Jain, 2000, p. 4). Classification according to a class set of exactly two members is known as *binary* classification.

2.4 Text Classification

Text classification, or *categorization*, has been perhaps *the* archetypal text-based machine learning application according to Weiss, Indurkhya, Zhang, and Damerau (2005, p. 81) and is considered a well-understood problem (Weiss et al., 2005, p. 52). The goal of text classification is simply to assign a category of some classification scheme to a document or other arbitrary unit of text (Manning & Schütze, 2003, p. 575). Automatic classification of documents mathematically speaking provides a solution to making the binary class decision that is a function which maps documents to the class with an outcome of either true or false, such that

$$f : w \rightarrow L$$

where w is a vector of features for the documents and L is the class label (Weiss *et al*, 2005, p. 52). Supervised learning of a text classifier in effect learns which particular combination of encoded features for a document lead to a particular classification based on a *training set* of documents—documents whose class are already known. This set of previously-classified documents is called a training set because it trains the learning algorithm (Manning & Schütze, 2003, p. 577).

Text classification generally proceeds in a stepwise process. First a corpus is (or corpora are) acquired. Secondly the corpus is transformed into a feature representation in a step commonly referred to as preprocessing. Preprocessing frequently involves a great deal of

language parsing and careful cleaning to ensure a minimum of non-linguistic noise in the data set. Preprocessing may be followed by additional efforts to transform, enhance, and or reduce the feature set by the use of advanced NLP techniques. Once the preprocessing and additional representational transformational steps are complete, document instances are associated with their manually assigned classes. At this point of the text classification process the input for the machine learning process is ready. An appropriate machine learning algorithm is selected and the machine learning application learns a predictive model for classification of documents via training from the input data. Statistics about the performance of the machine learning step are generated, and the process is ready for analysis and further refinement.

Performance of the classification model is measured by applying the learned classification model to a *test set* (Manning & Schütze, 2003, p. 577). Typical measures of classification performance include *precision* and *recall* (Manning & Schütze, 2003, p. 577). Precision ($a/a+b$) and recall ($a/a+c$) measures are based upon a *contingency table*, shown below, that shows which documents were correctly classified by the learned classifier (Manning & Schütze, 2003, p. 577).

	L is correct	¬L is correct
L was assigned	*a*	*b*
¬L was assigned	*c*	*d*

Table 1: Contingency table

Documents belonging to *a* are known as true positives, to *d*, true negatives, to *c*, false

negatives, and to *d*, false positives.

Classification can be evaluated either by direct application of the model to the entire set of documents, herewith called training evaluation, or by various *holdout methods* (Frank & Witten, 2000, p. 125). Cross-validation is considered a holdout method such that the original data set is divided into a number of evenly divided groups (called *folds*); with one fold removed, the classifier is trained on the remaining documents, and then the learned classifier is tested on the fold (Frank & Witten, 2000, p. 126). Further cross-validation requires that every instance is used once for the test set (Frank & Witten, 2000, p. 126). For example, 10-fold cross-validation withholds one-tenth of the documents for testing and trains over the other nine-tenths; and this process is repeated 10 times in such a way that every document appears in the sum total of the 10 tests sets exactly once. *Stratified* cross-validation implies that the distribution of the class over the test and train sets in the withholding process is the same as the distribution of the class over the entire set of documents (Frank & Witten, 2000, p. 126). Perhaps the single best measure for evaluating text classification is precision based on stratified 10-fold cross-validation (Frank & Witten, 2000), as the goal of building an automatic classifier is to be able to pluck out with a high degree of precision novel documents that belong to the positive document class. Stratified 10-fold cross-validation gives a good picture of the relative generality of the classification model built by the learning application.

Automated text classification dates back to the 1960s (Sebastiani, 2002, p. 1) and is perhaps the quintessential intersection of information retrieval and machine learning. A

number of statistical classification techniques have been applied to text classification problems, including Expectation-Maximization (Nigam, McCallum, Thrun, & Mitchell, 2000), naïve Bayes (McCallum & Nigam, 1998), support vector machines (SVM) (Dumais, Platt, Heckerman, & Sahami, 1998; Joachims, 1998; Leopold & Kindermann, 2002), and rule induction (Apté, Damerau, & Weiss, 1994).

2.5 Text classification with SVM

Text classification projects tasks face a hurdle generally known as the *curse of dimensionality* (Belman, 1961). In short, representations of collections of documents are usually highly dimensional regardless of the particular encoding scheme chosen. By highly dimensional, we mean that the collection of documents require a great number of features or attributes to represent them. It is not unusual to encounter 10,000 or more attributes describing a collection of texts (Joachims, 1997, p.3). The curse of dimensionality means that the more features needed to describe a problem space, the less general the model. Another way of saying this is that the hypervolume[1] formed by the feature space becomes exponentially less tractable as the number of features increases linearly (Belman, 1961). The goal of any learning experiment is to devise a model general enough to be reapplied to new instances not included in the current data set.

A common approach to treating the curse of dimensionality in automatic text

1 Hypervolume – the *n*-dimensional space implied by the *n* features used to describe a problem space

classification, for which there are many examples (*e.g.*, Apté *et al*., 1994), is to treat as many features as possible as irrelevant. The process of eliminating dimensions by treating them as irrelevant is known as *feature selection*. Feature selection, or *feature reduction*, responds to the curse of dimensionality by eliminating dimensions of the representation being input into the learner. It has been shown however that few features are in actuality irrelevant and that feature selection usually implies a loss of information (Joachims, 1997).

Texts can be characterized as having highly dimensional input sets with few irrelevant features (Joachims, 1997). In addition, vectors representing individual documents in a corpus are usually very sparse. Document feature vectors that represent each document contain few nonzero features. In other words, if we use the set of all words of a corpus as the feature set for that corpus, each document will likely have no more than a mere fraction of that total set of words, and so the vectors used to represent each set will be dominated with zero-valued features.

Finally texts are said to be generally linearly separable with respect to classification (Joachims, 1997). Failures in linear separation are usually due either to dubious documents or to misclassification of documents by human classifiers (Joachims, 1997).

Support vector machines (SVMs) are well-suited to text classification tasks given the characteristics of text. SVMs handle high dimensionality well because they have built-in overfitting protection. SVMs optimally identify attributes that facilitate linear separation,

thus eliminating the need for significant efforts at feature reduction including stemming, stop lists, and lemmatization (Joachims, 1997).

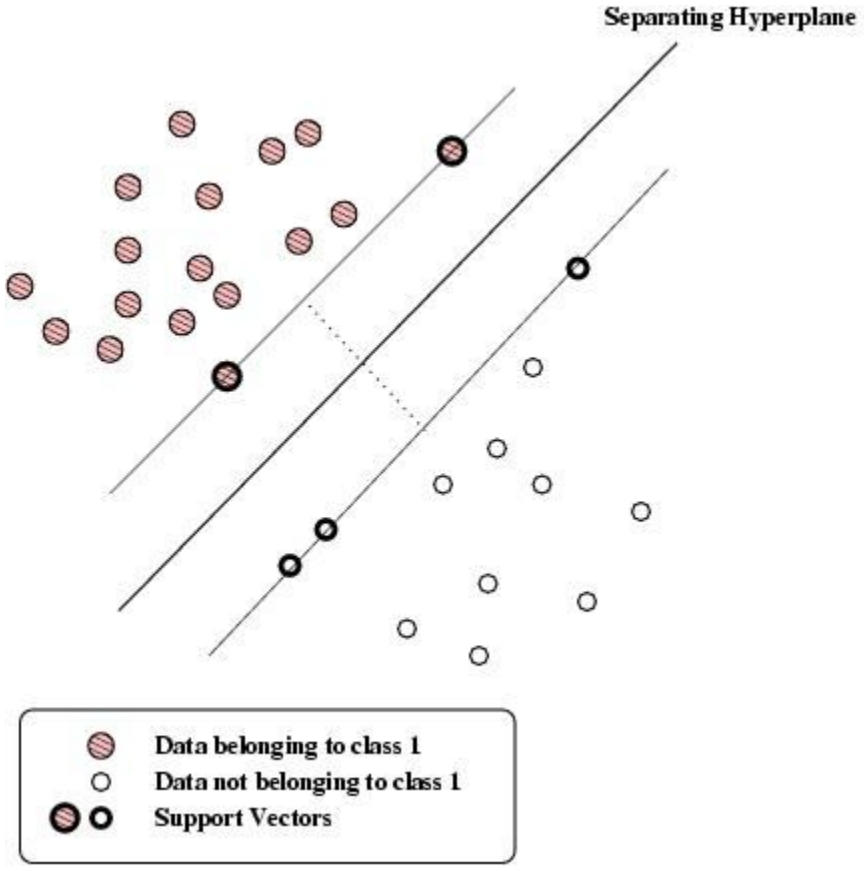

*Illustration 1: A Support Vector Machine (from **http://tinyurl.com/83lmv**)*

Support Vector Machines (Vapnik, 1995) were first proposed in 1979 yet have only become popular in the learning community in the last 10 years (Dumais *et al.*, 1998). An SVM is in short a hyperplane that separates two classes such that the boundary is set by support vectors with maximal margin. Determining the maximum margin is treated by SVMs as an optimization problem and is resolved as a quadratic programming problem.

2.6 Health-related texts & classification

Numerous efforts have been made to perform automatic document classification on biomedical-related text collections (de Bruijn & Martin, 2002). Text classification has been performed on various subsections of the biomedical science literature including documents found inside MEDLINE (*e.g.*, Yang & Chute, 1994), molecular biology texts (Wilbur & Yang, 1996), cell biology texts (Mostafa & Lam, 2000), and to clinical narratives (Wilcox & Hripcsak, 2003; Wilcox & Hripcsak, 2000). Automatic classification of medical texts has been of great interest since the early 1990s given not only the large volume of biomedical texts but also the need to not only expedite extraction of relevant medical information & evidence from the research literature but also apply that knowledge to particular clinical situations (de Bruijn & Martin, 2002).

While numerous text classification projects can be found with respect to the biomedical research literature and to clinical text data, few if any such projects can be found with respect to consumer health information. Price & Hersch (1999) have *suggested* the utility of various computational approaches to consumer health document filtering, but little if any work has been done on automatic topic classification of consumer health text collections.

2.7 Knowledge sources & classification

There are many ways to encode text documents. Most of the text classification research starts at the preprocessing stage with the generation of what is known as a *bag of words* representation. A bag of words representation simply means that the entirety of the attribute set for the representation is made of words from the corpus, and every word in the corpus is represented by the feature set. In other words, a bag of words representation is simply the corpus vocabulary. While language has many features, such as part of speech, lemmas, term meaning, and so forth, bag of words has been regularly been used for its simplicity. Using a corpus' vocabulary makes the preprocessing step of text classification quite simple. It is fairly usual for bag of words to be transformed slightly by (a) the removal of stop words (commonly occurring words like determiners and prepositions) and (b) stemming of the vocabulary.

While the use of bag of words representations has its advantages, individual word tokens do not cover the full set of information conveyed by a corpus. Individual words in a context have specific meanings (or may be used in a polysemous fashion and therefore have multiple meanings), and those meanings contain information about the documents. With respect to topic classification, these meanings have information that bears on the topic class to which each document belongs. Additionally, bag of words representations can contain nonsense words (*e.g.,* 'blah') or misspellings. In an ideal world our representations would somehow incorporate knowledge embodied in a lexicon and even contain some information about the relationships between various meanings of each word. Preferably the knowledge sources utilized would be tuned in some way to

correspond well with the particular domain of subject matter to which the corpora belong.

2.7.1 The SPECIALIST lexicon and text classification

The United Medical Language System (UMLS) project, an open source effort by the National Library of Medicine (NLM), offers linguistic knowledge resources tuned to medical-oriented texts. Of particular usefulness for the development of feature representations is the SPECIALIST lexicon, one of three knowledge sources in the UMLS system (*n.d.*, 2005). Use of the SPECIALIST lexicon (SL) as a knowledge source for representing a feature set with respect to semantic information offers the advantage of a reduction in the amount of labor needed to build any medical-related language processing system (Johnson, 1999). At the very least the SL offers a comprehensive way to control the term set of a feature representation for the purposes of filtering out nonsense words and misspellings without a loss of actual words. In fact the very spirit motivating the creation of the SL centered around creating normalized word and string indices (Humphreys, Lindberg, Schoolman & Barnett, 1998).

Use of UMLS tools such as the SPECIALIST lexicon has proved useful to text mining efforts in the past (Aronson, 2001). Bodenreider (2000) successfully applied UMLS semantics to the automatic classification of broad disease categories in a clinical trials database. Wilcox & Hripcsak (2000, 2003) have used information from the UMLS-like MedLEE system for text classification tasks in the medical domain as well, specifically

by applying MedLEE to hospital clinical data reports.

The SPECIALIST lexicon is limited in that it does not contain information about semantic relationships such as *hypernymy, meronymy,* or *polysemy*. Further, the UMLS MetaMap tool, designed to incorporate semantic relationship information into a usable NLP code base, suffers from performance problems and implementation complexities beyond the scope of the current project.

2.7.2 WordNet and text classification

While MetaMap may not be convenient for use in feature representation, the WordNet lexical database (Miller *et al.*, 1994) offers information related to various semantic relationships among words. WordNet according to Miller *et al.* is a "lexical reference system" whereby "English nouns, verbs, and adjectives are organized into synonym sets, each representing one underlying lexical concept" such that "different relations link the synonym sets." (1993, p. 1). Synonym sets, referred to in WordNet as *synsets*, are related to each other by a series of pointers (Miller, 1995) called synset ids. Pointers between synset ids may represent linguistic relationships such as hyponymy ("is-a"), antonymy, and meronymy ("has-a"). WordNet has the potential to be used in order to represent concepts rather than words and the relationships between such concepts. Scott and Matwin (1998, 1999) have made extensive use of WordNet, in particular its hypernymy pointers, in text classification tasks to mixed success. Benkhalifa, Mouradi, and Bouyakif

(2001) complimented feature representations with WordNet information in text classification tasks resulting in significant improvements to F1 and accuracy of the classification tasks.

2.8 Web-based Classification

While library items in general should be targeted for the most specific subject headings possible (Taylor, 2000), the classification of web-based items often focuses upon topic selection needs. Given the parameters of web-based collection representation such as portals, where navigating content often starts from short but broad-ranging list of topics that cover and include all of the web materials in some meaningful way, starting with classification of topics makes good sense. Classification needs for web-based collections should focus upon usability of the collections, in particular navigability of the content. Foremost in navigability is the topic menu, usually customized to fit exactly the contents of a particular web site/corpus.

McCallum et al. (2000) successfully executed automatic text classification of web-based documents according to top-level topic categories through the use of keyword representations of documents. Yang, Slattery,and Ghani (2002), in efforts to automate classification of web documents, focused efforts on using hypertext features contained in HTML tags in building feature representations for web documents. However it can easily be recognized that the WWW corpora are comprised of a vast array of document formats,

hardly limited to HTML, thereby limiting the value of using HTML features in light of extracting and weighting them. Further, the use of HTML features moves any text classification system away from generalizability, even with respect to web collections.

3 Experimental Design

3.1 Classification Models

The text classification model being followed for the present experiment involves five general stages: corpus collection, preprocessing, feature representation building, machine learning, and analysis of results.

3.2 NCHI Corpora

2576 web pages were downloaded from the NCHI website in June 2004. Site spidering performed by Catherine Blake's research team was performed without controls for zero-length files or occurrence of non-HTML or content-less HTML (e.g., outer frames) files. The pages were downloaded to and stored on a directory residing on UNC SILS's Jade server. At this stage it was unknown exactly how many real web pages this downloading step resulted in. The collection of web pages comprise not a single corpus but rather a set of corpora, as the NCHI web site contains numerous and diverse consumer health-oriented web collections from different content providers. All of the documents were classified by employees of the UNC Health Sciences Library with respect to various topics and terms and the relevant cataloging data was stored in a set of tables ported by the present author to an Oracle database hosted on another SILS server named Pearl.

3.3 Preprocessing

Numerous attempts were made to remove the HTML from the web pages. One of the objectives of preprocessing included the complete removal of HTML information with the exception of paragraph and section breaks[2]. Conventional HTML parsing, given the vast irregularity of HTML styles among the varied web corpora, proved to leave behind significant amounts of HTML code fragments. A bash script was written to contain a set of seven other bash scripts, each containing a small filtration step. Further steps prepared the data so as to create a set of data containing document identifier/term pairs in a format loadable into an Oracle database. The preprocessing proceeded as follows:

1. Each HTML file was first run through a Java class designed to decode HTML entities into literal strings.
2. The results of step 1 were run through a call to a simple implementation of HTMLEditorKit.Parser, a standard Java class that recognizes well-formed HTML tag pairs as events and removes them. I extended the class so that in processing it would replace HTML heading tags with header flags and HTML paragraph tags with non-HTML paragraph flags.
3. The results of step 2 still left behind a good deal of HTML code, both well-formed and otherwise. These results were run through an

2 Attempts to use paragraph and section information were abandoned.

application called lynxdump, a standard extension of standard UNIX and LINUX distributions Essentially lynxdump allows command-line-level plain text dumps of HTML content. These results were stripped of blank paragraphs, blank lines, and excess whitespace.

4. The results of step 3 were filtered in such a way that paragraph end flags inserted in step 2 were replaced with newline characters. End-header flags were also replaced, and then white space and empty sections were again filtered out.
5. Step 4 results were run through a custom Java process written to remove Javascript contents,
6. Step 5 results were formatted into a single plain text file containing document id/term pairings, with one pair for every occurrence of every whole string in the remaining contents.
7. The resulting text file from step 6 was manually stripped of all decoded symbols, numbers, and other non-word-level noise and then loaded into the database.
8. The table of data loaded was further screened for remaining symbols, diacritical marks, and numerals.

1733 documents remained from the original set after the preprocessing steps were executed; 379,339 document id/term pairs provide values for the 23,536 distinct term features. Hundreds of files in the original spidered collection that did not survive the preprocessing steps were blank, PDFs or other binary files, HTML files devoid of

content, HTML redirects, HTML outer frame files, HTML files containing only Javascript, or were impartial/broken spider downloads that either terminated or initiated prematurely therefore rendering incomplete HTML files.

3.4 Feature representation construction

3.4.1 Representation definitions

The goal of the feature representation step was to take the initial preprocessed content, essentially a bag of words representation in its own right, and render it in several distinct and normalized fashions. The goal of the present experiment is, after all, to evaluate the effect of different feature representations on automatic classification performance.

SPECIALIST – The baseline representation of the present experiment. The preprocessed data was joined on a table containing the SL in such a way that the only terms to be included in this representation are necessarily members of the SL. One of the goals behind using the SL was to act as a crude filter, removing misspellings, dubious strings, and proper names such as cities and names of people. This step however should not remove health-related terms that are based on proper names (*e.g.*, Graves).

SPECIALIST_POS – Given that WordNet only contains nouns, adjectives, verbs, and adverbs, some accounting would need to be made for the benefits of reducing the possible

parts of speech to these four in a classification task. This representation, along with the other two part-of-speech-reduction representations (SPECIALIST_NV and SYNSET_NV), were designed primarily to act as bases from comparison of the effects of limiting part of speech in a given representation. WordNet was used as the part-of-speech (POS) filter in order to simplify matters and give a raw intersection of the possibilities of SPECIALIST and WordNet combined in the current experiment's feature universe. In other words, SPECIALIST_POS is equivalent to the intersection of the SPECIALIST and WordNet lexicons inside the present problem's feature space. No POS tagging was done at this stage in order to reduce possible noise. This was chosen because there is risk of losing critical information in POS tagging, and if a term could either be, say, a noun or an adverb, we want to keep it regardless of its true grammatical function in each context.

SPECIALIST_NV - WordNet hypernymic relations exist only between nouns and verbs; any representation based solely on hypernymic relations is necessarily limited to nouns and verbs included in WordNet. This representation serves as an intermediary representation then between the SPECIALIST representation and the two hypernym representations.

SYNSET - The SYNSET representation, like the SPECIALIST representation, was derived from a simple join—in this case, every SPECIALIST term in the feature set was replaced with its representative synset identifier. Given the vast polysemy of the English language, this step necessarily implies an explosion of the feature space. It adds more information, but it also adds more noise; it remains to be seen whether the trade-off is

beneficial to automatic classification performance. Neither POS tagging nor word-sense disambiguation was performed as intermediary steps in building this representation. The choice was made in light of the efficiencies of the SVM classifier as well as the ease by which the representation could be arrived upon.

SYNSET_NV – Like the other POS reductions, this representation serves as a sort of intermediary benchmark between for the purposes of measuring the amount of benefit/cost based solely upon limiting POS in the feature space. In this case, SYNSET_NV is an intermediary between SYNSET and HYPERNYM1; the HYPERNYM1 encoding can only include nouns and verbs.

HYPERNYM1 – This representation contains a hypernym representation of the SYNSET representation. In short, every hypernym of every synset in the SYNSET representation is used to replace the SYNSET. If a synset id does not have a corresponding hypernym id, the original synset id is retained. It is hoped that this will reduce the feature space while, more importantly, maintaining and possibly concentrating all of the information necessary for the topic-based classification process. Since topic identification is in some sense a task of simplification of many words to one, this sort of effort makes at least good sense intuitively. I may also refer to this representation as the parent hypernym representation, given that the hypernyms selected for this representation are direct hypernyms.

HYPERNYM2 – hypernyms of HYPERNYM1 or grandparent hypernyms of SYNSET. This representation is being employed to test whether the added conceptual generalization

add benefit to the classification task?

All seven representations were normalized with respect to document set, document length, maximum term/synset frequency, and minimum term frequency. In short, only terms or synsets occurring more than five times across the corpora were kept. Further, terms occurring more than 1950 times were removed. This step removed only the most trivial terms, such as 'North,' 'Carolina,' and 'health.' After each of these representations were filtered for minimum and maximum term frequencies, it was then determined which documents provided at least five doc id/feature pairings. In other words, if one document in one of the seven representation had only three non-zero features, that document was removed from every representation even if it had more than five non-zero features in every other representation. Normalizing the document set in this way ensured that the experiments would all be performed on the same set of documents, in this case, 1576 documents.

3.4.2 Properties of the feature universe and feature representations[3]

A *feature universe* is the set of all possible features of a feature set. For example, if our document collection is solely in the English language and a bag of words approach is selected, then the feature universe is simply all words in the English language. In the present experiment, the feature universe for each representation is bounded by the

3 I chose to analyze the feature representations and use of knowledge sources at this stage because such analysis is part of building the experiment rather than analyzing the results of the experiment.

members of the SL, and, in the case of every feature representation except SPECIALIST constructed for the classification experiments, the feature universe is bounded by the intersection of the sets of SL members and strings included in WordNet. Given that each representation is bounded by the SL, it might help if some understanding of what was lost from the original preprocessing set when it was initially bounded by the SL.

I sampled 95 terms from the set of 9312 unique features lost in the join between the original preprocessing results and the SPECIALIST representation (before document normalization). The terms lost can be characterized as following ($p<0.05$):

Type of term lost	**Percentage of total lost** ($p<0.05$, +/- 10%)
proper names, including acronyms, surnames, first names, and geographic names	73%
non-English/non-words/misspelling	20%
words that should be in the lexicon	7%

Table 2: Types of terms lost by using SPECIALIST

By these figures it appears that transforming the data from the simple bag of words representation produced directly from the preprocessing steps to the SPECIALIST representation shed about three pieces of noise and fifteen meaningless names for every real word lost in the step. Whether this step creates a performance increase for the classification task is beyond the scope of the present study, yet it seems inevitable that such a reduction will invariably speed up the learning computational process significantly given the drastic reduction of the feature space it entails.

The SPECIALIST lexicon as implemented in the current project has a total feature universe of 339,695 unique lower case strings. WordNet, as implemented by a port of version 2.0 to MySQL rewritten by the author for Oracle, contains 96,727 unique lower case strings. The feature universe created by the intersection of these two sets results in a total feature universe size of 47,888 distinct lower case strings, considerably smaller than the feature universe of the SL's 340,000 or so unique terms. Further, 60,550 unique synsets in total correspond to this universe of 47,888 distinct lower case terms created by the intersection of the SL and WordNet; 33033 of those synsets represent nouns, 11,495 are verbs, 13946 are adjectives, and only 2076 are adverbs.

As expected, the construction of the six representations by employing WordNet in various ways significantly altered the size of the feature space for each representation. Limiting SPECIALIST to nouns, verbs, adjectives and adverbs in WordNet reduced the feature space by 25%, and limiting the feature space to only nouns and verbs found in WordNet reduced that feature space by an additional 25%. As expected, re-representing the data set in terms of un-disambiguated synsets resulted in an explosion of the feature space. Representing that set of synsets by its hypernyms reduced the feature space fourfold, and the grandparent hypernym feature space was less than half the size of the HYEPRNYM1 representation.

Representation	Unique features	As % of SPECIALIST
SPECIALIST	5991	100.0%
SPECIALIST_POS	4402	73.5%
SPECIALIST_NV	3022	50.4%
SYNSET	15045	251.1%
SYNSET_NV	11503	192.0%
HYPERNYM1	5091	85.0%
HYPERNYM2	2411	40.2%

Table 3: Size of feature spaces

3.5 Vector model

All of the feature representations were transformed into sparse document vectors using a set of Java classes written in part by the present author. Each sparse document vector was associated and labeled by its human-assigned topic category. The most frequently assigned topic term was chosen as the class of interest for the present experiment ("Health Facilities"); 567 of the 1576 documents were assigned by human catalogers to this class while 1009 were determined not to belong to the class.

3.6 Experiments

The Weka Machine Learning Environment (Frank & Witten, 2000) was chosen to run the actual classification learning experiments. Weka is implemented in Java and is therefore portable; further, it is simple to implement, and it also has the advantage of having an

efficient and easy-to-use implementation of the SVM supervised learning classifier, the classifier used in the present experiment given its resistance to overfitting flexibility with respect to large feature spaces. Each set of 1576 sparse document vectors for each feature representation were rendered into a file format readable by Weka using code written by the author which was executed on Jade. The learning experiments, seven in all, were run using Baobab, UNC's high performance Beowulf computing cluster.

4 Results

4.1 Result 1: All representations perform similarly well within the current set of documents with respect to direct training data.

Precision and recall statistics for the training tests of the automatically-derived classification model show apparently near-perfect results (see Table 4 below). The use of WordNet synsets and parent hypernyms seem to improve performance in terms of recall. These results however seem to indicate that, despite SVM's resistance to overfitting, the model is quite possibly overfitting the data and that analysis for a more general model using cross-validation techniques may show significantly worse results.

		TRAINING		
Representation	*Number of features, as % of base representation*	*Precision*	*Recall*	*F-measure*
1. Specialist (base representation)	100%	**1.00**	0.92	**0.98**
2. Specialist, {N, V, ADJ, ADV} only	73%	0.99	0.88	0.93
3. Specialist {N, V} only	50%	0.99	0.80	0.88
4. WordNet Synsets	251%	**1.00**	**0.94**	0.97
5. WordNet Synsets, {N, V} only	192%	0.99	0.92	0.96
6. Parent Hypernyms of 5	85%	0.98	0.84	0.90
7. Grandparent hypernyms of 5	40%	0.95	0.69	0.80
Averages				
Avg, all POS reduction	115%	0.98	**0.85**	**0.91**
Avg., Specialist POS reduction	62%	**0.99**	0.84	**0.91**
Avg, all WN	142%	0.98	**0.85**	**0.91**

Table 4: Training data

4.2 Result 2: General application of the learning model performs significantly worse than the specific application to the document set

4.3 Result 3: Use of the Parent Hypernym representation provides slight performance advantages with respect to precision

4.4 Result 4: Little variation in retrieval performance is seen between the different representations

		10-FOLD STRATIFIED CROSS-VALIDATION		
Representation	*Number of features, as % of base representation*	*Precision*	*Recall*	*F-measure*
1. Specialist (base representation)	100%	0.59	0.55	**0.57**
2. Specialist, {N, V, ADJ, ADV} only	73%	0.58	0.52	0.55
3. Specialist {N, V} only	50%	0.57	0.47	0.51
4. WordNet Synsets	251%	0.56	0.55	0.56
5. WordNet Synsets, {N, V} only	192%	0.56	**0.56**	0.56
6. Parent Hypernyms of 5	85%	**0.60**	0.53	**0.57**
7. Grandparent hypernyms of 5	40%	0.59	0.45	0.51
Averages				
Avg, all POS reduction	115%	0.58	0.51	0.54
Avg., Specialist POS reduction	62%	0.57	0.50	0.53
Avg, all WN	142%	0.58	**0.52**	**0.55**
Avg., all hypernym	63%	**0.60**	0.49	0.54

Table 5: Classification performance, 10-fold stratified cross validation

As seen in Table 5, the generalizability of the models from all the representations suffers with respect to the models applied directly to the document set. It appears that the SVM, contrary to the theory, overfit the data to a significant degree.

Of particular interest is the relatively significant improvement in precision for the parent hypernyms over the synsets (see Table 6). The four percent increase in precision from the synset-based representations (SYNSET, SYNSET_NV) to the parent hypernym representation seems somewhat surprising since I would expect performance degradation due to information loss. I would suspect such information loss because the hypernym representation's feature set is one-fourth the size of the synset one.

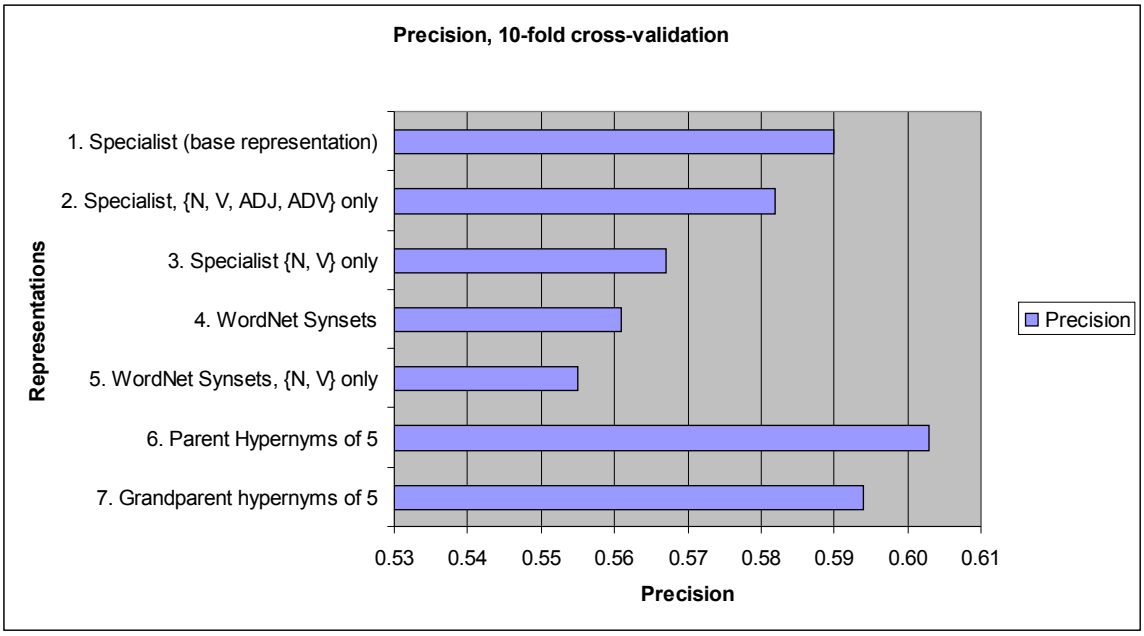

Table 6: Precision results

Equally surprising is the result showing that the representations each perform relatively similarly. I had expected performance to *vary* by the amounts of information contained in each representation, and I had similarly expected information to depend to a significant degree on size of the feature set and its spread across its respective feature universe.

5 Summary & Conclusion

A previous examination of automatic classification of the NCHI collection according to ten different topics as well as ten different local terms demonstrated quite strongly that the precision and recall of the automatic text classifier is highly dependent upon the particular class being assigned. The weakest results seemed to come from the most frequent topic, the topic in focus in the present experiment. This variation across topics seems to suggest that, given that the automatic classification model is built upon a training set based on human manual classification, a better understanding of the success of the automatic classifier can only be had with a thorough examination of the performance of the human/manual classification performance. Knowing the value of the automatic classification model requires knowing how well different humans perform the same manual classification task for each topic; knowing the inter-rater reliability of the manual task may shed light on the slight degree of lack of generality of the automatic classification model for the top topic as well as variance in recall and precision between the models for different topics. In other words, we cannot know how well the automatic classifier is truly performing without knowing the consistency of human classification. The closer the automatic classifier's performance to the human classifier's, the greater the actual utility of the automatic classification system.

Several improvements to the present experiment should be made. Foremost, given that few if any text features are irrelevant, and given the theoretical responsiveness of SVMs

to highly dimensional feature spaces, many of the feature reduction steps should be eliminated from the feature representation construction phase. First, minimum and maximum term frequency bounds should be removed. In particular, the setting of a minimum term frequency resulted in the removal of approximately 9000 dimensions, leaving only roughly 6000 dimensions. It seems that this is at best only a crude approximation of what an SVM can do gracefully and optimally with respect to handling individual features. The representations should also be built without use of stop lists. The basic document normalization steps, however, should be retained. Removing all dimension reduction efforts at the early stages may improve retrieval statistics by as many as 30 points, when based on the performance of experiments run on the same document set using minimal feature reduction with a different classification algorithm. Additional experiments should be run on other topic classes as well.

WordNet's feature universe with respect to health terminology may be a limiting factor to its utility for feature representation for a consumer health collection such as NCHI's. The gains in information from semantic relation information are likely offset by the sheer number of words WordNet excludes from its domain. A WordNet-like medical ontology with a large number of medical-specific terms and concepts likely offers significant improvements in classification tasks. But we may not need to wait for such an imaginary tool. The use of hypernymy information seems to already offer great potential. It may be best, however, to *add* WordNet information to a bag-of-words or lexicon-filtered representation rather than using WordNet features to *replace* the previous information. It appears that SVMs are well-suited to handle the increase in attributes.

6 Bibliography

--. (2000). UMLS Knowledge Sources (Version 2000 Edition) Washington, DC: U.S. Dept of Health and Human Services, National Institutes of Health, National Library of Medicine.

--. (2005) *Specialist Lexicon Fact Sheet*. [Web Page]. URL http://www.nlm.nih.gov/pubs/factsheets/umlslex.html [2005, November 29].

--. WordNet 2.0 data files in MySQL format : Android Technologies.

Apté, C., Damerau, F., & Weiss, S. M. (1994). Automated Learning of Decision Rules for Text Categorization. *ACM Transactions on Information Systems, 12* (3), 233-251.

Aronson, A. R. (2001). Effective Mapping of Biomedical Text to the UMLS Metathesaurus: the MetaMap Program. *Journal of the American Medical Informatics Association,* 17-21.

Bellman, R. (1961). *Adaptive Control Processes: A Guided Tour* . Princeton, NJ: Princeton University Press.

Benkhalifa, M., Mouradi, A., & Bouyakhf, H. (2001). Integrating WordNet Knowledge to Supplement Training Data in Semi-Supervised Agglomerative Hierarchical Clustering for Text Categorization. *International Journal of Intelligent Systems, 16* (8), 929-947.

Bodenreider, O. (2000). Using UMLS Semantics for Classification Purposes. *Journal of the American Medical Informatics Association,* 86-90.

Cohen, A. M., & Hersh, W. R. (2005). A Survey of Current Work in Biomedical Text Mining. *Briefings in Bioinformatics, 6* (1), 57-71.

De Bruijn, B., & Martin, J. (2002). Getting to the (C)Ore of Knowledge: Mining Biomedical Literature. *International Journal of Medical Informatics, 67* (1-3), 7-18.

Dumais, S. T., Platt, J., Heckerman, D., & Sahami, M. (1998). Inductive learning algorithms and representations for text categorization. *Proceedings of the seventh international conference on Information and knowledge management (ACM-CIKM98)* (pp. 148-155). ACM Press.

Frank, E., & Witten, I. H. (2000). *Data Mining: Practical machine learning tools and techniques with Java implementations*. San Francisco: Morgan Kaufmann Publishers.

Humphreys, B. L., Lindberg, D. A. B., Schoolman, H. M., & Barnett, G. O. (1998). The Unified Medical Language System: an Informatics Research Collaboration. *Journal of the American Medical Informatics Association, 5* (1), 1-11.

Jain, A. K., Duin, R. P. W., & Mao, J. C. (2000). Statistical Pattern Recognition: a Review. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 22* (1), 4-37.

Joachims, T. (1997). *Text Categorization with Support Vector Machines: Learning with Many Relevant Features*. (LS-8 Report 23). University of Dortmund Computer Science Department.

Joachims, T., & Sebastiani, F. (2002). Guest Editors' Introduction to the Special Issue on Automated Text Categorization. *Journal of Intelligent Information Systems, 18* (2-3), 103-105.

Johnson, S. B. (1999). A Semantic Lexicon for Medical Language Processing. *Journal of the American Medical Informatics Association, 6* (3), 205-218.

Leopold, E., & Kindermann, J. (2002). Text Categorization With Support Vector Machines. How to Represent Texts in Input Space? *Machine Learning, 46* (1-3), 423-444.

Lesk, M. (1997) *How Much Information Is There In the World?* [Web Page]. URL http://www.lesk.com/mlesk/ksg97/ksg.html [2005, January 12].

Manning, C. D., & Schütze, H. (2003). *Foundations of Statistical Natural Language Processing*. Cambridge, MA: The MIT Press.

McCallum, A., & Nigam, K. (1998). A comparison of event models for Naive Bayes text classification. *AAAI-98 Workshop on Learning for Text Categorization* (AAAI Press.

McCallum, A. K., Nigam, K., Rennie, J., & Seymore, K. (2000). Automating the Construction of Internet Portals With Machine Learning. *Information Retrieval, 3* (2), 127-163.

Miller, G. A. (1995). WordNet - a Lexical Database for English. *Communications of the ACM, 38* (11), 39-41.

Miller, G. A., Beckwith, R., Fellbaum, C. D., Gross, D., & Miller, K. (1993). *Five Papers on WordNet*. Princeton, NJ: Princeton University.

Miller, G. A., Fellbaum, C. D., Tengi, R., Wolff, S., Wakefield, P., Langone, H., & Haskell, B. (2004). WordNet (Version 2.0) Princeton, NJ: Princeton University Cognitive Science Laboratory.

Mitchell, T. M. (1997). *Machine Learning*. New York, NY: McGraw-Hill.

Mostafa, J., & Lam, W. (2000). Automatic Classification Using Supervised Learning in a Medical Document Filtering Application. *Information Processing & Management, 36* (3), 415-444.

Nigam, K., McCallum, A. K., Thrun, S., & Mitchell, T. (2000). Text Classification From Labeled and Unlabeled Documents Using EM. *Machine Learning, 39* (2-3), 103-134.

Price, S. L., & Hersch, W. R. (1999). Filtering web pages for quality indicators: an empirical approach to finding high quality consumer health information on the World Wide Web. *Proc AMIA Symp.,* 911-915.

Scott, S., & Matwin, S. (1999). Feature Engineering for Text Classification. *Proceedings of 16th International Conference on Machine Learning (ICML-99)* (pp. 379-388). San Francisco: Morgan Kaufmann Publishers.

Scott, S., & Matwin, S. (1998). Text Classification Using WordNet Hypernyms. *Usage of WordNet in Natural Language Processing Systems: Proceedings of the Workshop (COLING-ACL'98)* (p. 45--52).

Sebastiani, F. (2002). Machine Learning in Automated Text Categorization. *ACM Computing Surveys, 34* (1), 1-47.

Taylor, A. G. (2000). *Wynar's Introduction to Cataloging and Classification*. Englewood, CO: Libraries Unlimited, Inc.

Vapnik, V. N. (1995). *The Nature of Statistical Learning Theory*. Berlin: Springer-Verlag..

Weiss, S. M., Indurkhya, N., Zhang, T., & Damerau, F. (2005). *Text Mining: Predictive Methods for Analyzing Unstructured Information*. New York, NY: Springer.

Wilbur, W. J., & Yang, Y. M. (1996). An Analysis of Statistical Term Strength and Its Use in the Indexing and Retrieval of Molecular Biology Texts. *Computers in Biology and Medicine, 26* (3), 209-222.

Wilcox, A., & Hripcsak, G. (2000). Medical Text Representations for Inductive Learning. *Journal of the American Medical Informatics Association,* 923-927.

Wilcox, A. B., & Hripcsak, G. (2003). The Role of Domain Knowledge in Automating

Medical Text Report Classification. *Journal of the American Medical Informatics Association, 10* (4), 330-338.

Yang, Y. M., & Chute, C. G. (1994). An Example-Based Mapping Method for Text Categorization and Retrieval. *ACM Transactions on Information Systems, 12* (3), 252-277.

Yang, Y. M., Slattery, S., & Ghani, R. (2002). A Study of Approaches to Hypertext Categorization. *Journal of Intelligent Information Systems, 18* (2-3), 219-241.