Evaluating WordNet Features in Text Classification Models

Trevor Mansuy and Robert J. Hilderman
Department of Computer Science
University of Regina
Regina, Saskatchewan, Canada S4S 0A2
{mansuy1t, robert.hilderman}@uregina.ca

Abstract

Incorporating semantic features from the WordNet lexical database is among one of the many approaches that have been tried to improve the predictive performance of text classification models. The intuition behind this is that keywords in the training set alone may not be extensive enough to enable generation of a universal model for a category, but if we incorporate the word relationships in WordNet, a more accurate model may be possible. Other researchers have previously evaluated the effectiveness of incorporating WordNet synonyms, hypernyms, and hyponyms into text classification models. Generally, they have found that improvements in accuracy using features derived from these relationships are dependent upon the nature of the text corpora from which the document collections are extracted. In this paper, we not only reconsider the role of WordNet synonyms, hypernyms, and hyponyms in text classification models, we also consider the role of WordNet meronyms and holonyms. Incorporating these WordNet relationships into a Coordinate Matching classifier, a Naive Bayes classifier, and a Support Vector Machine classifier, we evaluate our approach on six document collections extracted from the Reuters-21578, USENET, and Digi-Trad text corpora. Experimental results show that none of the WordNet relationships were effective at increasing the accuracy of the Naive Bayes classifier. Synonyms, hypernyms, and holonyms were effective at increasing the accuracy of the Coordinate Matching classifier, and hypernyms were effective at increasing the accuracy of the SVM classifier.

Introduction

Supervised text classification, the task of assigning predefined category labels to previously unseen documents based on learned models, has been the focus of a considerable amount of previous and recent research (de Buenaga Rodriguez, Gomez-Hidalgo, & Diaz-Agudo 1997), (Scott & Matwin 1998), (Jensen & Martinez 2000), (Kehagias *et al.* 2003), (Hotho & Bloehdorn 2004), (Rosso *et al.* 2004), (Peng & Choi 2005). When performing text classification, the classification accuracy we observe on the previously unseen documents largely depends on the quality of the training set we have used to build the category models. That is, if training information for a category model is sparse, then we can expect the category model to be a poor representation of the category, and the classification accuracy to be poor. Similarly, a training set may not necessarily be sparse, but it can contain important word relationships that a simple vector of words is not capable of modeling. For example, consider two documents, one discussing the concept flax and the other wheat. A reasonable person would most likely classify both documents as grain-related or agriculture-related because he/she recognizes the relationship between flax and wheat. A simple word vector may not be sufficient for capturing this relationship.

In an attempt to address the issue of related concepts in text classification models, a number of researchers have previously incorporated features derived from word relationships in the WordNet lexical database (de Buenaga Rodriguez, Gomez-Hidalgo, & Diaz-Agudo 1997), (Scott & Matwin 1998), (Jensen & Martinez 2000), (Kehagias *et al.* 2003), (Hotho & Bloehdorn 2004), (Rosso *et al.* 2004), (Peng & Choi 2005). WordNet is a database of words containing a semantic lexicon for the English language that organizes words into groups called synsets (i.e., synonym sets) (Miller 1995). A *synset* is a collection of synonymous words linked to other synsets according to a number of different possible relationships between the synsets (e.g., *is-a*, *has-a*, *is-part-of*, and others). When building a category model for a document, words related to a feature already in the model (and satisfying some desired WordNet relationship) are extracted from the WordNet database and incorporated into the model. The intuition is that this expanded representation has greater potential to assign semantically similar documents to the same class.

As far as the authors of this paper know, in the area of text classification approaches incorporating WordNet features, previously studied relationships include synonyms (de Buenaga Rodriguez, Gomez-Hidalgo, & Diaz-Agudo 1997), (Scott & Matwin 1998), (Jensen & Martinez 2000), (Kehagias *et al.* 2003), (Hotho & Bloehdorn 2004), (Rosso *et al.* 2004), (Peng & Choi 2005), hypernyms (Scott & Matwin 1998), (Jensen & Martinez 2000), (Hotho & Bloehdorn 2004), (Peng & Choi 2005), and hyponyms (Peng & Choi 2005). In this paper, we extend the use of WordNet relationships in text classification models by considering the role of WordNet meronyms and holonyms. Meronyms and holonyms are compositional relationships. A concept is a meronym of another if it is a component of that other con-

Table 1: Characteristics of previous WordNet classification approaches

Author	*WordNet Relationships* Synonyms	Hypernyms	Hyponyms	*Word Sense Disambiguation* Manual	All	Most Likely	Context	*Model Type* Word	Synset
(de Buenaga Rodriguez, Gomez-Hidalgo, & Diaz-Agudo 1997)	•			•				•	
(Scott & Matwin 1998)	•	•			•			•	•
(Jensen & Martinez 2000)	•	•				•		•	
(Kehagias *et al.* 2003)	•			•					•
(Hotho & Bloehdorn 2004)	•	•			•	•	•	•	
(Rosso *et al.* 2004)	•			•					•
(Peng & Choi 2005)	•	•	•			•		•	

cept. Conversely, a concept is a holonym of another if it has that other concept as a component. In particular, we incorporate words derived from these two WordNet relationships into a Coordinate Matching, a Naive Bayes, and a Support Vector Machine classifier to determine whether these "narrowing" and "broadening" relationships result in increased accuracy. We then apply the algorithms to six document collections extracted from the Reuters-21578 (Hettich & Bay 1999), USENET, and DigiTrad (Digital 2002) text corpora to determine the effectiveness of these relationships in increasing the accuracy of the text classification algorithms.

Related Work

WordNet has been applied to a variety of problems in machine learning, natural language processing, information retrieval, and artificial intelligence (WordNet 2005). In this section, we discuss a number of relevant contributions that describe approaches to incorporating WordNet semantic features into text classifiers. Characteristics of these approaches are summarized in Table 1 and described in the text that follows.

One of the first efforts toward the integration of WordNet features into a text classifier is described in (de Buenaga Rodriguez, Gomez-Hidalgo, & Diaz-Agudo 1997). Here it is proposed that accuracy may be increased if the category model for a document is expanded by incorporating WordNet synonyms of the category label. In this work, since the number of features actually incorporated by WordNet expansion was small, manual word sense disambiguation was used to determine the correct word sense. To evaluate their approach, Rocchio and Widrow-Hoff classification algorithms were used. It was found that accuracy, in general, was increased by incorporating synonyms, and, in particular, was increased when the number of categories in the training documents was sparse.

In (Scott & Matwin 1998), an approach is described where all words found in a document are considered for WordNet expansion rather than just the category label. Here, however, both synonyms and hypernyms of a category label are incorporated into the category model. A different representation of a category model is proposed where the features actually correspond to WordNet synsets rather than words. No word sense disambiguation is done in this approach, rather all senses of a word are incorporated into the category model. Using the RIPPER classification algorithm for evaluation of their approach, results were mixed, showing both statistically significant increases and decreases on various document collections.

A similar approach incorporating both synonyms and hypernyms is proposed in (Jensen & Martinez 2000). Noting that words in a synset are organized in occurrence frequency order, in their approach to word sense disambiguation, they only select the most likely sense for incorporation into the category model. Coordinate Matching, TF*IDF, and Naive Bayes classification algorithms were used to evaluate their approach, where different combinations of synonyms, hypernyms, and bigrams were incorporated into the category models. They found that incorporating hypernyms into category models is almost always appropriate.

The work described in (Kehagias *et al.* 2003) evaluates the merits of modeling senses as features rather than words. The Brown Semantic Corpus, a document collection whose words have been tagged with the correct word sense, is used such that only synsets corresponding to the features found in the document are incorporated into the category model. Consequently, hypernyms are not incorporated in this approach. Of course, word sense disambiguation is not necessary since the document collection has previously been tagged with the correct sense. They used MAP, Naive Bayes, and kNN classifiers to evaluate their approach. An increase in accuracy was obtained on most document collections considered. However, the increases were small, leading the authors to conclude that the benefits from using their approach are marginal.

The application of WordNet as an ontology in text classification problems is explored in (Hotho & Bloehdorn 2004). Their approach incorporates both synonyms and hypernyms in the category model. Three different word sense disambiguation strategies are studied in their approach. These include strategies incorporating all senses and the most likely sense. A third strategy, context, measures the degree of overlap of different WordNet features in relation to how close these features occur to one another in the document being classified. Using an AdaBoost classifier, an increase in accuracy is reported on most document collections considered.

In (Rosso *et al.* 2004), an approach is proposed whereby vector of words in a category model are replaced with a vector of WordNet synsets. Manual word sense disambiguation is done before classification. A kNN classifier showed an increase in accuracy on most document collections considered.

An approach is proposed in (Peng & Choi 2005) where synonyms, hypernyms, and hyponyms are incorporated into a category model as well as features found in the document. Category models attempt to capture the relationship between synsets rather than simply measuring the density of the various WordNet features. A variation of the most likely sense disambiguation strategy is used where the frequency of each sense in the context of the document collection is considered before WordNet expansion. Using a TF*IDF classifier where only hypernyms are incorporated into a category model, increases in accuracy on particular document collections are obtained that greatly exceed those reported by any other authors. However, the methodology seems simplistic and is not well-documented (e.g., was 10-fold cross-validation used, for instance), so the reported results are suspect.

Incorporating WordNet Features

In WordNet, synsets are connected according to a number of lateral, hierarchical, and compositional relationships. In this work, we are concerned with the relationships defined between noun synsets, which we now review, as follows. A synonym is a lateral relationship where a concept X is similar to a concept Y (i.e., an X is a Y and a Y is an X). A hypernym is a hierarchical relationship where a concept Y is a superclass of a concept X (i.e., every X is a kind of Y). A hyponym is a hierarchical relationship where a concept X is a subclass of a concept Y (i.e., a Y includes every X). A meronym is a compositional relationship where a concept X is a component of a concept Y (i.e., an X is part of Y). A holonym is a compositional relationship where a concept Y has a concept X as a component (i.e., a Y has an X).

To illustrate the general approach to text classification incorporating WordNet features, we present a simple, representative classification task. Sample synsets for the concepts `pup`, `dog`, and `cat` are shown in Figure 1 and sample documents are shown in Figure 2. In Figure 1, the sample synsets are connected to other synsets by the various WordNet relationships. In Figure 2, the documents are shown to contain keywords that will be used to facilitate classification. `Doc1` is the only labeled document and represents the training set. Our task is to attempt to automatically assign a category label to `Doc2` through `Doc5`.

We begin by attempting to classify the documents without incorporating WordNet features. Without WordNet features, `CategoryModel(Pets)=`⟨`dog`⟩. Since `dog` is not found in ⟨`Words(Doc2), Words(Doc3), Words(Doc4), Words(Doc5)`⟩, we are unable to classify any of the documents.

We now consider situations where WordNet features are incorporated. For example, when incorporating synonyms, we have `CategoryModel(Pets)=`⟨`dog, pup`⟩. That is, from the `dog` synset in Figure 1, we extract the synonym `pup` and add it to the category model for `Pets`. In this case, `pup` is found in ⟨`Words(Doc2)`⟩, so `Label(Doc2)=Pets`. The synonym relationship enabled the classification of a document that did not refer to any concepts found in the training set, but did contain a similar concept.

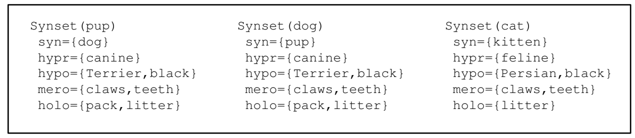

Figure 1: Sample synsets

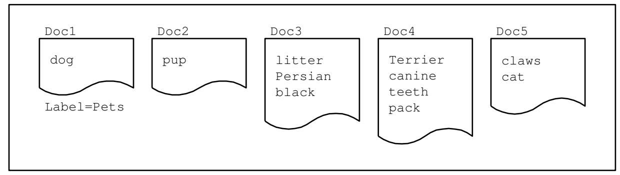

Figure 2: Sample documents

Similarly, to assign the same category label to documents that refer to more specific concepts, hyponyms can be incorporated. For example, when incorporating synonyms and hyponyms, we have `CategoryModel(Pets)=`⟨`dog, pup, Terrier, black`⟩. Since `pup` is found in ⟨`Words(Doc2)`⟩, `black` is found in ⟨`Words(Doc3)`⟩, and `Terrier` is found in ⟨`Words(Doc4)`⟩, then `Label(Doc2)=Label(Doc3)=Label(Doc4)=Pets`. This example shows how concepts that share some common, more specific characteristic can be assigned the same category label even though they may contain different keywords. That is, since `pups` and `cats` both come in black varieties, the classifier was able to assign the `Pets` label to a cat-related document (i.e., `Doc3`) even though the original training set was dog-related.

The general approach should now be clear. That is, the first step to incorporating WordNet features is to expand the category model with the words contained in the appropriate WordNet relationship. In the second step, classification simply proceeds as usual.

Experimental Results

Text Classifier Overview

We implemented a text classifier that can incorporate the various WordNet features into a category model. We also constructed the text classifier so that the classification algorithm for a particular series of experiments can be ”plugged in” at run-time. The results reported here were obtained using Coordinate Matching, Naive Bayes, and SVM classification algorithms. However, it is important to note that the focus of our evaluation is not concerned with the general performance of the algorithms, per se. Specifically, the focus of our evaluation is to determine how incorporating combinations of synonyms, hypernyms, hyponyms, meronyms and holonyms affects accuracy in a “simple” classification algorithm (i.e., Coordinate Matching) and in more “sophisticated” ones (i.e., Naive Bayes and SVM). The Coordinate Matching classifier is considered to be “simple” because all features within a category model are considered. The Naive Bayes and SVM classifiers are considered to be “sophisticated" because they use more intelligent approaches. The Naive Bayes classifier considers all features within a category model, but the importance of each feature is weighted by its probability of occurring in a particular class. The SVM classifier only considers those features that add discernability to the training set when building the category model. The text classifier was implemented in Visual C++ version 6.0 and run under Windows XP on an IBM compatible PC with a 3.0 GHz Pentium 4 processor and 1 GB of memory.

Document Collections

The text classifier was run on six different document collections drawn from three different text corpora: Reuters-21578, USENET, and DigiTrad. These particular document collections have been used extensively in previous text classification studies (e.g., see Related Work). The characteristics of the six document collections are shown below in Table 2 (reproduced from (Jensen 2000)). In Table 2, the *Corpora* column describes the origin of the corresponding dataset, the *Class* column describes a single semantic label attached to each document in the dataset, the *No. of Documents* column describes the number of documents assigned to each class, the *Average Length* column describes the average number of words in each document in the corresponding class, and the *WordNet Nouns* column describes the percentage of words in the corresponding documents that are WordNet nouns.

Table 2: Document collection characteristics

Corpora	*Class*	*No. of Documents*	*Average Length*	*WordNet Nouns (%)*
USENET	Micro	163	72	54
	Neuro	117	77	54
USENET	Taxes	170	69	58
	History	79	88	53
Reuters-21578	Corn	168	99	60
	Wheat	221	83	62
Reuters-21578	Livestock	113	98	63
	Gold	134	84	61
DigiTrad	Marriage	200	125	47
	Murder	224	146	46
DigiTrad	Politics	194	115	51
	Religion	238	91	50

Methodology

Text classification in this, and other work, is a two-step process. In the first step, a category model for each category in a corpus of labeled training documents is built. In the second step, a classification algorithm compares unlabeled documents to the learned category models to assign (hopefully) the correct category label to the unlabeled documents. In this work, a category model for a document consists of a bag of words and synsets where each feature in the model is either a word actually found in the document, a synset corresponding to a feature found in the document, or a synset linked to another synset already in the category model. The most likely sense strategy is used for word sense disambiguation.

Each document collection has been previously tagged with a Brill part of speech tagger. When building a category model, a word is compared against a list of common stop words. If the word is determined to be a stop word, it is simply discarded. If the word is not determined to be a stop word, but is tagged as a noun, it is incorporated into the category model and then used to query WordNet for additional features (i.e., synonyms, hypernyms, hyponyms, meronyms, holonyms) which are also incorporated into the category model. Finally, if the word is not determined to be a stop word, and is not tagged as a noun, it is incorporated into the category model without WordNet expansion.

For each classification algorithm, six classification tasks were performed for each document collection, one with the "base" classifier (i.e., when no WordNet features are incorporated) and one where combinations of WordNet features (i.e., synonyms, synonyms and hypernyms, synonyms and hyponyms, synonyms and meronyms, and synonyms and holonyms) are incorporated into the classifier. Results for each classification task reflect the average accuracy obtained using 10-fold cross-validation.

Results

The accuracy values shown in Tables 3 through 6 represent the percentage or relative percentage of documents in the corresponding document collection that have been correctly classified.

The accuracy obtained by the Coordinate Matching, Naive Bayes, and SVM base classifiers is shown in Table 3. Although it is generally believed that SVM classifiers have higher accuracy than Naive Bayes classifiers, and that both SVM and Naive Bayes classifiers have higher accuracy than Coordinate Matching classifiers, this does not seem to be the case for the six document collections shown here. Specifically, in Table 3, there is no statistically significant difference between the accuracy of the three base classifiers. Analysis of Variance and the *F* Ratio Test were used to determine statistical significance using a 90% level of significance (i.e., $\alpha = 0.10$) and a null hypothesis that there is no difference between means.

Table 3: Accuracy of the three base classifiers

Class	*Coordinate Matching*	*Naive Bayes*	*SVM*
Micro/Neuro	56.90	61.07	64.14
Taxes/History	81.55	87.81	71.97
Corn/Wheat	65.59	75.53	84.77
Livestock/Gold	98.07	97.02	97.07
Marriage/Murder	71.93	75.72	83.40
Politics/Religion	78.64	80.13	82.44

The accuracy obtained using the Coordinate Matching, Naive Bayes, and SVM classifiers for each combination of document collection and WordNet features is shown below in Tables 4 through 6, respectively. In Tables 4 through

6, the *Base* column describes the accuracy obtained when no WordNet features are incorporated into the classifier. The *Syn*, *Syn+Hypr*, *Syn+Hypo*, *Syn+Mero*, and *Syn+Holo* columns describe the relative difference in accuracy obtained when the various WordNet features are incorporated. For example, in Table 4, the accuracy obtained for Micro/Neuro in the *Syn* column is 60.73, which we show as a relative increase of $+3.83$ over the 56.90 shown in the *Base* column. Columns shown in bold represent a statistically significant difference in accuracy from those obtained by the base classifier. The Wilcoxon Signed Rank Test was used to determine statistical significance using a 90% level of significance (i.e., $\alpha = 0.10$) and a null hypothesis that there is no difference between medians (i.e., a two-tailed test).

Table 4: Accuracy using the Coordinate Matching classifier

Class	*Base*	***Syn***	***Syn+ Hypr***	*Syn+ Hypo*	*Syn+ Mero*	***Syn+ Holo***
Micro/Neuro	56.90	**+3.83**	**+7.47**	+1.85	+6.56	**+4.71**
Taxes/History	81.55	**+0.71**	**+5.71**	-1.34	-2.77	**+2.14**
Wheat/Corn	65.59	**+5.00**	**+6.08**	+3.41	+4.06	**+4.43**
Livestock/Gold	98.07	**+0.39**	**+0.39**	-0.84	-0.38	**+0.77**
Marriage/Murder	71.93	**+1.29**	**+2.32**	-3.68	-0.05	**+0.82**
Politics/Religion	78.64	**+0.05**	**+1.49**	-5.14	-1.61	**-2.04**

Table 5: Accuracy using the Naive Bayes classifier

Class	*Base*	*Syn*	*Syn+ Hypr*	***Syn+ Hypo***	*Syn+ Mero*	*Syn+ Holo*
Micro/Neuro	61.07	+0.57	+5.26	**-1.25**	-1.53	-0.73
Taxes/History	87.81	-0.71	-1.60	**-16.34**	-19.58	-3.11
Corn/Wheat	75.53	+0.60	-3.57	**-4.09**	+2.14	+0.83
Livestock/Gold	97.02	-0.38	+1.23	**-2.06**	0.00	+0.53
Marriage/Murder	75.72	0.00	-1.27	**-2.11**	-2.84	-1.00
Politics/Religion	80.13	+1.23	-1.16	**-5.03**	-1.83	+0.05

Table 6: Accuracy using the SVM classifier

Class	*Base*	*Syn*	***Syn+ Hypr***	*Syn+ Hypo*	*Syn+ Mero*	*Syn+ Holo*
Micro/Neuro	64.14	-1.67	**+0.14**	+0.12	+0.91	-1.22
Taxes/History	71.97	+2.14	**+5.84**	+8.11	+6.26	+1.85
Corn/Wheat	84.77	+1.16	**+0.94**	-1.05	+1.30	+2.07
Livestock/Gold	97.07	+0.41	**+0.41**	-3.58	-0.89	+0.02
Marriage/Murder	83.40	-1.29	**-0.36**	-1.68	-2.04	-0.36
Politics/Religion	82.44	-0.26	**+1.85**	-4.23	-0.79	+0.84

In summary, we draw six main conclusions. First, incorporating synonyms, synonyms and hypernyms, and synonyms and holonyms into the Coordinate Matching classifier resulted in a significant increase in accuracy from that of the base classifier. Second, incorporating synonyms and hyponyms, and synonyms and meronyms into the Coordinate Matching classifier did not result in accuracy that was significantly different from that of the base classifier. Third, incorporating synonyms, synonyms and hypernyms, synonyms and meronyms, and synonyms and holonyms, respectively, into the Naive Bayes classifier did not result in accuracy that was significantly different from that of the base classifier. Fourth, incorporating synonyms and hyponyms into the Naive Bayes classifier resulted in a significant decrease in accuracy from that of the base classifier. Fifth, incorporating synonyms and hypernyms into the SVM classifier resulted in a significant increase in accuracy from that of the base classifier. And last, incorporating synonyms, synonyms and hyponyms, synonyms and meronyms, and synonyms and holonyms did not result in accuracy that was significantly different from that of the base classifier.

In Table 4, the *Syn*, *Syn+Hypr*, and *Syn+Holo* columns show a statistically significant increase in accuracy from the base classifier, where accuracy increased in six out of six, six out of six, and five out of six classes, respectively. The *Syn+Hypo* column shows accuracy increased for the Micro/Neuro and Wheat/Corn classes over the base classifier when synonyms and hyponyms were incorporated, specifically $+1.85\%$ and $+3.41\%$, respectively. In contrast, a decrease in accuracy was obtained for Taxes/History, Livestock/Gold, Marriage/Murder, and Politics/Religion, specifically, -1.34%, -0.84%, -3.68%, and -5.14%, respectively. Similarly, for the *Syn+Mero* column, accuracy increased by $+6.56\%$ and $+4.06\%$ for Micro/Neuro and Wheat/Corn, respectively, while Taxes/History, Livestock/Gold, Marriage/Murder, and Politics/Religion decreased by -2.77%, -0.38%, -0.05%, and -1.61%, respectively. However, there is no statistically significant difference between the accuracy of the base classifier and the accuracy shown in the *Syn+Hypo* and *Syn+Mero* columns.

In Table 5, the *Syn* and *Syn+Hypr* columns show that increases and decreases in accuracy were evenly split between the classes. However, there is no statistically significant difference between the accuracy of the base classifier and the accuracy shown in these columns. The accuracy shown for these columns is similar to the results shown in (Jensen & Martinez 2000). We speculate that the slight variance between the accuracy we observe and the accuracy reported in (Jensen & Martinez 2000) is due to variations in the split of the document collection during 10-fold cross validation. For the *Syn+Hypo* column, accuracy decreased in all six classes (over 16% for Taxes/History). The accuracy shown in this column is a statistically significant decrease from the accuracy of the base classifier. For the *Syn+Mero* column, accuracy decreased in four of the six classes (almost 20% for Taxes/History) with no or a slight increase in the other two. For the *Syn+Holo* column, increases and decreases in accuracy were evenly split between the classes. Again, there is no statistically significant difference between the accuracy of the base classifier and the accuracy reported in these columns.

In Table 6, the *Syn+Hypr* column shows a statistically significant increase in accuracy between the base classifier and the accuracy when synonyms and hypernyms are incorporated. There is no statistically significant difference between the accuracy of the base classifier and the accuracy shown in the *Syn*, *Syn+Hypo*, *Syn+Mero*, and *Syn+Holo* columns.

Discussion

The impact on accuracy by incorporating the various WordNet features into the three classifiers is summarized below in Table 7, where -, ○, and + represent a decrease, no change, and an increase in accuracy, respectively.

Table 7: Impact of WordNet features on accuracy

Classifier	*Syn*	*Syn+ Hypr*	*Syn+ Hypo*	*Syn+ Mero*	*Syn+ Holo*
Coordinate Matching	+	+	○	○	+
Naive Bayes	○	○	-	○	○
SVM	○	+	○	○	○

Incorporating synonyms, hypernyms, and holonyms seems to have have the most significant affect on the Coordinate Matching classifier. A possible explanation for the increase in accuracy is likely due to the nature of these WordNet relationships. Both hypernyms and holonyms represent a "narrowing" relationship, while hyponyms and meronyms represent a "broadening" relationship. That is, with hypernyms, we are going from the more specific to the more general, but a given sense of a word will typically have only a single more general concept. Conversely, a given general word might have many more specific instances of words that can be derived from it. Similar reasoning applies to meronyms and holonyms where a single object will generally have many parts, and, consequently, many meronyms. At the same time, a word describing a part of something will only belong to a few more complicated objects, consequently, having few holonyms. The general pattern that we observe is that the narrowing relationships enhance the category models in the Coordinate Matching classifier, while the broadening relationships do not. That is, when adding a large number of features to a category model, the potential for adding poor features is also large. So, when we add a poor feature that only occurs in a single category, we expect it to have little impact. However, when we add a large number of poor features to both categories, we can expect accuracy to decrease because we are blurring the line between the two classes.

We believe the simplicity of the Coordinate Matching classifier makes it most susceptible to slight changes in the category model. Thus, while the results suggest that adding features derived from holonyms may benefit the Coordinate Matching classifier, the Naive Bayes and SVM classifiers were not affected by the addition of features from either meronyms or holonyms.

Conclusion and Future Work

We incorporated features from WordNet relationships into text classification models. We found that synonyms, hypernyms, and holonyms increase accuracy in the Coordinate Matching classifier. Hyponyms actually decrease accuracy in the Naive Bayes classifier. Hypernyms increase accuracy in the SVM classifier. Future work will focus on feature weighting similar to that in (Jensen & Martinez 2000), where hypernyms were weighted based upon their depth in the WordNet database.

Acknowledgments

We thank Canada's Natural Science and Engineering Research Council (NSERC) for their financial support. We are grateful to George Miller for WordNet 2.0, and Ian Witten and Frank Eibe for the Naive Bayes and SVM classifiers in the Weka 3.4 open source software issued under the GNU General Public License. We are also grateful to Lee Jensen for providing us with his Coordinate Matching classifier algorithm and the six document collections used in this work.

References

de Buenaga Rodriguez, M.; Gomez-Hidalgo, J.; and Diaz-Agudo, B. 1997. Using WordNet to complement training information in text categorization. In *Proceedings of the 2nd International Conference on Recent Advances in Natural Language Processing (RANLP'97)*, 150–157.

Digital Tradition. 2002. [http://www.mudcat.org/AboutDigiTrad.cfm].

Hettich, S., and Bay, S. 1999. The UCI KDD archive [http://kdd.ics.uci.edu].

Hotho, A., and Bloehdorn, S. 2004. Boosting for text classification with semantic features. In *Proceedings of the Workshop on Mining for and from the Semantic Web at the 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'04)*, 70–87.

Jensen, L., and Martinez, T. 2000. Improving text classification by using conceptual and contextual features. In *Proceedings of the Workshop on Text Mining at the 6th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'00)*, 101–102.

Jensen, L. 2000. Improving text classification by using conceptual and contextual features. Master's thesis, Brigham Young University.

Kehagias, A.; Petridis, V.; Kaburlasos, V.; and Fragkou, P. 2003. A comparison of word- and sense-based text classification using several classification algorithms. *Journal of Intelligent Information Systems* 21(3):227–247.

Miller, G. 1995. WordNet: A lexical database for English. *Communications of the ACM* 38(11):39–41.

Peng, X., and Choi, B. 2005. Document classifications based on word semantic hierarchies. In *Proceedings of the International Conference on Artificial Intelligence and Applications (AIA'05)*, 362–367.

Rosso, P.; Ferretti, E.; Jiminez, D.; and Vidal, V. 2004. Text categorization and information retrieval using WordNet senses. In *Proceedings of the 2nd Global Wordnet Conference (GWC'04)*, 299–304.

Scott, S., and Matwin, S. 1998. Text classification using WordNet hypernyms. In *Proceedings of the Workshop on Usage of WordNet in Natural Language Processing Systems (Coling-ACL'98)*, 45–52.

WordNet Bibliography. 2005. [http://mira.csci.unt.edu/wordnet].