The Role of Semantic Information in Learning Question Classifiers*

Xin Li **Dan Roth** **Kevin Small**
Department of Computer Science
University of Illinois at Urbana-Champaign
{xli1,danr,ksmall}@uiuc.edu

Abstract

Question Classification is commonly used in question answering systems to perform a semantic classification of the target answer in an effort to provide additional information to downstream processes. It is different from the common text categorization task in the sense that questions are relatively short and contain less word-based information compared with classification of the entire text. This work presents a machine learning approach to this task. Our approach is to augment the questions with syntactic and semantic analysis, as well as external semantic knowledge, as input to the text classifier. It is shown that, in the context of question classification, augmenting the input of the classifier with appropriate semantic category information results in significant improvements to classification accuracy.

1 Introduction

Open-domain question answering (Moldovan et al., 2002; Ravichandran and Hovy., 2002) has become an increasingly important direction in natural language processing. The purpose of the question answering (QA) task is to seek an accurate and concise answer to a free-form factual question[1] contained in a large text corpora, as opposed to a document judged relevant through its similarity to the query. The difficulty of pinpointing and verifying the precise answer makes question answering more challenging than the common information retrieval task performed by readily available search engines.

*Research supported by NSF grants IIS-9801638 and ITR IIS-0085836 and an ONR MURI Award.

[1]It does not address questions like 'Do you have a light?', which calls for an action

Recent works (Hovy et al., 2001) have shown that locating an accurate answer hinges on first filtering out a wide range of candidates based on some categorization of answer types given a question, which is also demonstrated empirically in this paper. Specifically, this classification task has two purposes. First, it provides constraints on the answer types that allow further processing to precisely locate and verify the answer. Second, it provides information that downstream processes may use in determining answer selection strategies that may be answer type specific. For example, when considering the question: **Q:** *What Canadian city has the largest population?*, we do not want to test every noun phrase in a document to see whether it provides an answer. The hope is, at the very least, to classify this question as having answer type **city**, implying that only candidate answers that are cities need consideration.

At a high level, question classification may be viewed as a text categorization task (Sebastiani, 2002). However, there exist characteristics of question classification that distinguish it from the common task. On one hand, questions are relatively short and contain less word-based information compared with classifying the entire text. On the other hand, short questions are amenable for more accurate and deeper-level analysis. Our approach is, therefore, to augment the questions with syntactic and semantic analysis, as well as external semantic knowledge, as input to the text classifier. In this way, this work on question classification can be also viewed as a case study in applying semantic information to text classification.

Similar to syntactic information such as part-of-speech tags, a fairly clear notion of how to use lexical semantic information is to replace or augment each word by its semantic class in the given context, then generate a feature-based representation and learn a mapping from this representation to the desired property. This general scheme leaves several issues open that make the analogy to syntactic categories nontriv-

ial. First, it is not clear which semantic categories are appropriate and how to acquire them. Second, it is not clear how to handle the more difficult problem of semantic disambiguation when augmenting the representation of a sentence.

Therefore, there have been very few attempts to study these problems in the context of classification. In the context of prepositional phrase attachment, both (Brill and Resnik, 1994) and (Krymolowski and Roth, 1998) were able to show small improvements by using Wordnet semantic classes to augment the raw representation of sentences. Lin and Pantel (Pantel and Lin, 2002) have done several works on acquiring semantic classes and using the acquired information but, in most cases, this was not done in a classification framework. The semantic classes acquired by them will be used in the current work.

This work systematically studies several possible semantic information sources and their contribution to classification. For the first problem, we compare four types of semantic information sources that differ in their granularity, method of acquisition, and size: (1) automatically acquired named entity categories, (2) word senses in WordNet 1.7 (Fellbaum, 1998), (3) manually constructed word lists related to specific categories of interest, and (4) automatically generated semantically similar word lists (Pantel and Lin, 2002). For the second problem above, in all cases, we define semantic categories of words and incorporate the information into question classification in the same way: if a word w occurs in a question, the question representation is augmented with the semantic categories of the word.

Clearly, a word may belong to different semantic categories in different contexts. For example, the word *water* has the meaning *liquid* or *body of water* in different sentences. Without disambiguating the sense of a word we cannot determine which semantic category is more appropriate in a given context. At this point, our solution is to extract all possible semantic categories of a word as features, without disambiguation, and allowing the learning process to handle this problem, building on the fact that the some combinations of categories are more common than others and more indicative to a specific class label. As we show later, our experiments support this decision, although we have yet to experiment with the possible contribution of a better way to determine the semantic class in a context sensitive manner.

Our experimental study focuses on comparing the contribution of different syntactic and semantic features to the classification quality. In the experiments, we observe that classification accuracies over 1,000 TREC (Voorhees, 2002) questions reach 92.5% for 6 coarse classes and 89.3% for 50 fine-grained classes, and that the semantic information is critical to support this level of accuracy. A 28.7% error reduction can be achieved when semantic features are incorporated into fine-grained classification. This result is even better than the SVM and kernel methods (but with fewer features) in (Zhang and Lee, 2003; Suzuki et al., 2003), which is a proof of the necessity of informative features in this task.

The paper is organized as follows: Sec. 2 presents the question classification problem, its value to the overall question answering task, and our learning approach; Sec. 3 illuminates how the sources of semantic information are incorporated as features and describes all the features defined for this task. Sec. 4 presents our experimental study and results. In Sec. 5 we conclude by discussing a few issues left open by our study.

2 Question Classification

Many important natural language inferences can be framed as resolving ambiguity, either syntactic or semantic, based on properties of the surrounding context. These are typically modeled as classification tasks (Roth, 1998). Examples include part-of-speech tagging where a word is mapped to its part-of-speech tag in the context of a given sentence, context-sensitive spelling correction where a word is mapped to a similarly spelled word appropriate within the context of the sentence, and many other problems such as word-sense disambiguation or word choice selection in machine translation. Similarly, we define Question Classification (QC) here to be the multi-class classification task that one seeks a mapping $g : X \rightarrow \{c_1, ..., c_n\}$, that maps an instance $x \in X$ (e.g., a question) to one of n classes $c_1, ..., c_n$, which provides a semantic constraint on the sought-after answer.

2.1 Question Hierarchy

We define a two-layered taxonomy, which represents a natural semantic classification for typical answers. The hierarchy contains 6 coarse classes (ABBREVIATION, ENTITY, DESCRIPTION, HUMAN, LOCATION and NUMERIC VALUE) and 50 fine classes, shown in Table 1. Each coarse class contains a non-overlapping set of fine classes.

Coarse Class	Fine Classes
ABBREV.	abbreviation,expression
ENTITY	animal,body,color,creative,currency,disease, event,food,instrument,lang,letter,other, plant,product,religion,sport,substance, symbol,technique,term,vehicle,word
DESCRIPTION	definition,description,manner,reason
HUMAN	group,individual,title,description
LOCATION	city,country,mountain,other,state
NUMERIC	code,count,date,distance,money,order,other, period,percentage,speed,temp,volume-size,weight

Table 1: Question Classification Taxonomy.

2.2 QC in Question Answering - A Case Study

In this section, we provide evidence that using question classification results for reranking the passages supplied by a passage retrieval engine, which is a common component in question answering systems, can dramatically improve the precision of returned passages. The evaluation is performed on the questions in TREC 2002 (Voorhees, 2002).

The passage retrieval method we use is an adaptation of the Okapi BM25 document retrieval algorithm (Robertson et al., 1998) with typical parameter values ($b = 0.75$, $k_1 = 1.2$, $k_3 = \infty$) under the implementation provided by LEMUR[2]. We divide each document into overlapping 3-sentence passages for indexing. The question itself is directly used as the query, and 1000 passages were retrieved for each query. This original ranking is then compared against a reranking where any passage containing a term of the fine grained semantic class decided by classification of this question is ranked higher than those do not. One caveat here is that we removed from our test set questions of semantic classes which can not be annotated by our named entity tagger and those without known answers.

We evaluate *average precision of relevant passages* ($\frac{1}{n}\sum_{i=1}^{n}\frac{i}{rank\ of\ the\ ith\ relevant\ passage}$, where n is the total number of relevant passages containing the correct answer in the 1,000 returned.) for each question. The relative precision increases (%) averaged on questions by reranking are seen in Table 2.

Coarse Classes	NUM	ENTY	LOC	HUM	TOTAL
# of Questions	147	53	101	88	389
# Increase/# Decrease.	116/0	12/0	75/3	64/1	267/4
Avg. Rel. Prec. Increase	43.7%	54.0%	71.3%	27.2%	48.5%

Table 2: Relative Precision Increase of Reranked Passages. The rows show coarse classes of questions evaluated, the total number of questions tested in each coarse class, the number of questions whose retrieving precisions are increased and decreased by reranking, and the average relative precision increase over questions in each coarse class respectively.

An example of a question that benefits from this method is: *(NUM:money) What is the GDP of China?*. Too many passages contain *China* in the document collection and *GDP* does not help much to filter out irrelevant passages since it is not completely trivial to unify it with all of its equivalent representations. Therefore, associating the question classification outcome indicating that we are looking for a *dollar amount* drastically increases the precision of retrieved passages.

2.3 Learning a Question Classifier

To adapt to the layered semantic hierarchy of answer types, we adopt a hierarchical learning classifier (Li and Roth, 2002) based on the sequential model of multi-class classification, as described in (Even-Zohar and Roth, 2001). The basic idea of this model is to shrink the set of possible class labels (*confusion set*) of a given question step by step by concatenating a sequence of simple classifiers. In order to allow a simple classifier to output more than one class label in each step, the classifier's output activation is normalized into a density over the class labels and is thresholded.

The question classifier is built by combining a sequence of two simple classifiers. The first classifies questions into coarse classes (*Coarse Classifier*) and the second into fine classes (*Fine Classifier*). Each of them utilizes the Winnow algorithm within the SNoW (Sparse Network of Winnow (Carlson et al., 1999)) learning architecture which learns a separate linear function over the features for each class label efficiently. A feature extractor automatically extracts the same features for them based on multiple syntactic, semantic analysis results and external knowledge of the question. The second classifier depends on the first in that its candidate labels are generated by expanding the set of retained coarse classes from the first into a set of fine classes; this set is then treated as the confusion set for the second classifier. Figure 1 shows the basic

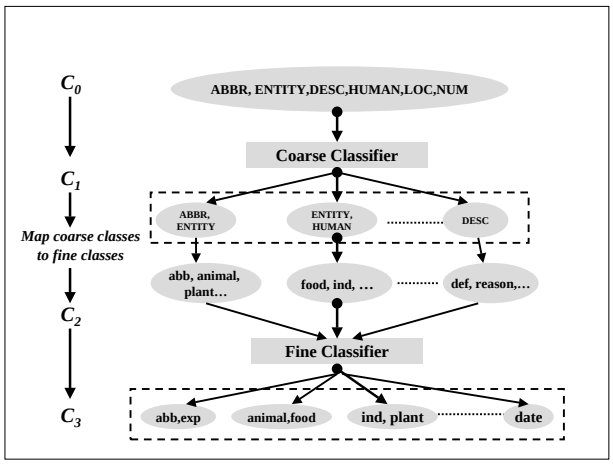

Figure 1: The hierarchical classifier

structure of the hierarchical classifier. During either the training or the testing stage, a question is processed along one single path top-down to get classified. The confusion set of the question is shrank from the set of all possible coarse classes C_0 to C_3 by the classifier.

For both the coarse and fine classifiers, the same decision model is used to choose class labels for a question ($C_0 \rightarrow C_1$ and $C_2 \rightarrow C_3$). Given a confusion set, SNoW outputs a density over the classes derived from the activation of each class. After ranking the classes in the decreasing order of density values, we have the possible class labels $C = \{c_1, c_2, \ldots, c_n\}$, with their densities $P = \{p_1, p_2, \ldots, p_n\}$ (where, $\sum_{i=1}^{n} p_i = 1$, $0 \leq p_i \leq 1$, $1 \leq i \leq n$). For each question we output

[2]The Lemur Toolkit for Language Modeling in Information Retrieval. *See http://www.cs.cmu.edu/ lemur*.

the first k classes ($1 \leq k \leq 5$), $c_1, c_2, \ldots c_k$ where k satisfies,

$$k = min(argmin_t(\sum_{i=1}^{t} p_i \geq T), 5),$$

T is a threshold value in [0,1] ($T = 0.95$ is chosen in the experiments.). For evaluation purpose, only one coarse class in C_1 and one fine class in C_3 with the highest rank are counted as the final output of the classifier.

3 Features in Question Classification

Machine Learning based classifiers typically take as input a feature-based representation of the domain element (e.g., a question). For the current task, a question sentence is represented as a vector of features and treated as a training or test example for learning. The mapping from a question to a class label is a linear function defined over this feature vector.

In this work, several primitive feature types are derived from multiple sources of syntactic and lexical semantic analysis of questions, each of which in itself could be a learning process, described later in this section. Over those primitive feature types, a set of operators are used to compose more complex features, such as conjunctive (n-grams) and relational features. Only 'active' features are listed in our representation so that despite the large number of potential features — about 500,000 in the whole feature space, the size of each example is small — hundreds of active features.

3.1 Syntactic Features

In addition to the information that is readily available in the input instance, it is common in natural language processing tasks to augment sentence representation with syntactic categories, under the assumption that the sought-after property, for which we seek the classifier, depends on the syntactic role of a word in the sentence rather than the specific word (Roth, 1998).

Our baseline classifier makes use of the standard POS information and phrase information extracted by a shallow parser. Specifically, we use *chunks* (non-overlapping phrases) and *head chunks*, extracted using a publicly available chunker described in (Punyakanok and Roth, 2001). The following example illustrates the information available when generating the syntax-augmented feature-based representation.

Question: *Who was the first woman killed in the Vietnam War?*

Chunking: *[NP Who] [VP was] [NP the first woman] [VP killed] [PP in] [NP the Vietnam War] ?*

The head chunks denote the first noun or verb chunk after the question word in a question. For example, in the above question, the first noun chunk after the question word *Who* is 'the first woman'. The features are represented as abstract tags in each example.

3.2 Semantic Features

Similar logic can be applied to semantic categories. In many cases, the property seems not depend on the specific word used in the sentence – that could be replaced without affecting this property – but rather on its 'meaning'. For example, given the question: *What Cuban dictator did Fidel Castro force out of power in 1958?*, we would like to determine that its answer should be a name of a person. Knowing that *dictator* refers to a person is essential to correct classification.

This work systematically studies four semantic information sources and their contribution to classification: (1) automatically acquired named entity categories - *NE*, (2) word senses in WordNet 1.7 (Fellbaum, 1998) - *SemWN*, (3) manually constructed word lists related to specific categories of interest - *SemCSR*, and (4) automatically generated semantically similar word lists (Pantel and Lin, 2002) - *SemSWL*.

For the four external semantic information sources, we define semantic categories of words and incorporate the information into question classification in the same way: if a word w occurs in a question, the question representation is augmented with the semantic category(ies), of the word. For example, in the question: *What is the state flower of California?* given that *plant* (for example) is the only semantic class of flower, the feature extractor adds *plant*, an abstract label to the question representation.

Named Entities

A named entity (NE) recognizer assigns a semantic category to some of the noun phrases in the question. The scope of the categories used here is broader than the common named entity recognizer. With additional categories that could help question answering, such as *profession*, *event*, *holiday*, *plant*, *sport*, *medical* etc., we redefine our task in the direction of semantic categorization. The named entity recognizer was built on the shallow parser described in (Punyakanok and Roth, 2001), and was trained to categorize noun phrases into one of 34 different semantic categories of varying specificity. Its overall accuracy ($F_{\beta=1}$) is above 90%. For the question *Who was the woman killed in the Vietnam War ?*, the named entity tagger will return: **NE:** *Who was the [Num first] woman killed in the [Event Vietnam War] ?* As described above, the identified named entities are added to the question representation.

WordNet Senses

In WordNet (Fellbaum, 1998), words are organized according to their 'senses' (meanings). Words of the same sense can, in principle, be exchanged in some contexts. The senses are organized in a hierarchy of hypernyms and hyponyms. Word senses provide another effective way to describe the semantic category

of a word. For example, in WordNet 1.7, the word *water* belongs to 5 senses. The first two senses are:

Sense 1: binary compound that occurs at room temperature as a colorless odorless liquid;

Sense 2: body of water.

Sense 1 contains words {H2O, water} while Sense 2 contains {water, body of water}. Sense 1 has a hypernym (**Sense 3**: binary compound); and one hyponym of Sense 2 is (**Sense 4:** tap water).

For each word in a question, all of its sense IDs and direct hypernym and hyponym IDs are extracted as features. This approach possibly introduces significant noise to classification since only a small proportion of senses are really related.

Class-Specific Related Words

Each question class frequently occurs together with a set of words which can be viewed as semantically related to this class. We analyzed about $5,000$ questions and constructed manually a list of related words for each question class. Those lists are different from ordinary named entity lists in a way that they cross the boundary of the same syntactic role. Below are some examples of the word lists.

Question Class: Food

{*alcoholic apple beer berry breakfast brew butter candy cereal champagne cook delicious eat fat feed fish flavor food fruit intake juice pickle pizza potato sweet taste ...*}

Question Class: Mountain

{*hill ledge mesa mountain peak point range ridge slope tallest volcanic volcano...*}

The question class can be viewed as a 'topic' tag for words in the list, a type of semantic categories. It's a semantic information source similar to the keyword information used in some earlier work (Hermjakob, 2001). The difference is that they are converted into features here and combined with other types of features to generate an automatically learned classifier.

Distributional Similarity Based Categories

Distribution similarity (Lee, 1999) of words captures the likelihood of them occurring in the same syntactic structures in text. Depending on the type of dependencies used to determine the distributional similarity, it can be argued that words with high distribution similarity have similar meanings. For example, the words used in the following syntactic structures are likely to be U.S. states.

... appellate court	campaign in ...
... capital	governor of ...
... driver's license	illegal in ...
... 's sales tax	senator for ...

Pantel and Lin (Pantel and Lin, 2002) proposed a method to cluster words into semantically similar groups based on their distributional similarity with respect to dependencies in a large collection of text. They built similar word lists for over $20,000$ English words. All the words (generally hundreds) in a list corresponding to a target word are organized into different senses. For example, the word *water* has the following similar words:

Sense 1: {*oil gas fuel food milk liquid ...*}
Sense 2: {*air moisture soil heat area rain snow ice ...*}
Sense 3: {*waste sewage pollution runoff pollutant...*}

One way of applying these lists in question classification is to treat the target word of a list as the semantic category of all the words in the list and in line with our general method, and add this semantic category of the word as a feature.

Comparison of Semantic Sources

In an effort to compare the semantic information sources, Table 3 presents the average number of semantic features extracted for each test question from each source. This indicates the increase in information, which significantly differs among them. Named entities provide the least semantic information while Pantel's categories provide the most as each of the category reflects the semantical similarity among a much broader range of words.

Feature Type	avg. # of features
NE	0.23
SemWN	16
SemCSR	23
SemSWL	557

Table 3: The average number of semantic features extracted for each test question based on different types of semantic features. For example, there are 16 SemWN features extracted for each question on average.

Among the four sources, named entity recognition is the only context sensitive semantic analysis of words. The other three sources are likely to add some degree of noise to the representation of a question due to lack of sense disambiguation. Furthermore, SemCSR is the only partially task-specific semantic analysis and all other sources can be applied in general classification tasks.

4 Experimental Study

Our experimental study focuses on (1) testing the performance of the learned classifier in classifying factual questions into coarse and fine classes, and (2) comparing the contribution of different syntactic and semantic features to the classification quality.

Based on the same framework of the hierarchical classifier described before, we construct different classifiers utilizing different feature sets and compare them in experiments. The first group of classifiers compared take as input an incremental combination of syntactic features (words, POS tags, chunks and head chunks).

In particular, the classifier takes as input all the syntactic features is denoted as SYN. Then, another group of classifiers are constructed by adding different combinations of semantic features to the input of the SYN classifier.

Three experiments are conducted for the above purposes. The first evaluates the accuracies of the hierarchical classifier for both coarse and fine classes using only syntactic features. The second evaluates the contribution of different semantic features (all 15 possible combinations of semantic feature types are added to the SYN classifier and compared this way.). In the third experiment we hope to find out the relation between the contribution of semantic features and the size of the training set by training the classifier with training sets of different sizes.

The 1000 questions taken from TREC 10 and 11 serve as an ideal test set for classifying factual questions. 21,500 training questions are collected from three sources: 894 TREC 8 and 9 questions, about 500 manually constructed questions for a few rare classes, and questions from the collection published by USC (Hovy et al., 2001). In the first two experiments, the classifiers are trained on all these questions. 10 other training sets with incremental sizes of 2,000, 4,000, ..., 20,000 questions built by randomly choosing from these questions are used in the third experiment. All the above questions were manually labelled according to our question hierarchy, with one label per question according to the majority of our annotators[3].

Performance is evaluated by the global accuracy of the classifiers for all the coarse or fine classes (**Accuracy**), and the accuracy of the classifiers for a specific class c (**Precision[c]**), defined as follows:

$$Accuracy = \frac{\#\ of\ correct\ predictions}{\#\ of\ predictions}$$

$$precison[c] = \frac{\#\ of\ correct\ predictions\ of\ class\ c}{\#\ of\ predictions\ of\ class\ c}$$

Note that since all questions are being classified, the global accuracy is identical to both precision and recall that are commonly used in similar experiments. Moreover, for specific classes, precision and recall are dependent although different — high precision on all specific classes implies high recall, so, only precision[c] is shown in Figure 5. Note that only one coarse class and one fine class with the highest rank in their density value are counted as correct in evaluation.

4.1 Experimental Results

All the classifiers are trained on the 21,500 training questions and tested on the 1,000 TREC 10 and 11 questions in the experiments except the case of studying the influence of training sizes.

[3]Available at http://l2r.cs.uiuc.edu/∼cogcomp/data/QA/.

Classification Using Only Syntactic Features

Table 4 shows the classification accuracy of the hierarchical classifier with different sets of syntactic features in the first experiment. **Word**, **POS**, **Chunk** and **Head(SYN)** represent different feature sets constructed from an incremental combination of syntactic features. For example, the feature set **Chunk** actually contains all the features in Word, POS, and adds chunks. **Head(SYN)** contains all the four types of syntactic features. Overall, we get a 92.50% accuracy for coarse classes and 85.00% for the fine classes using all the syntactic features. The reason why the classifier has a much lower performance in classifying fine classes compared with coarse classes is because there are far more fine classes and because they have less clear boundaries. Although it is not shown that chunks contribute to the classification quality in this experiment, in some other experiments, chunks are shown to contribute when combined with other types of features. The fact that head chunk information contributes more than generic chunks indicates that the syntactic role of a chunk is a factor that can not be ignored in this task.

Classifier	Word	POS	Chunk	Head(SYN)
Coarse	85.10	91.80	91.80	**92.50**
Fine	82.60	84.90	84.00	**85.00**

Table 4: Classification Accuracy of the hierarchical classifier for coarse and fine classes using an incremental combination of syntactic features.

Contribution of Semantic Features

Although minor improvements occur in classifying questions into coarse classes after semantic features are also used in the second experiment, significant improvements are achieved for distinguishing between fine classes. Figure 2 presents the accuracy of the classifier for fine classes after semantic features are input together with the SYN feature set.

The best accuracy for classifying fine classes in this experiment is 89.3%, using a combination of feature types {SYN, NE, SemCSR, SemSWL}. This is a 28.7% error reduction (from 15% to 10.7%) over the SYN classifier. For simplicity, this feature set {SYN, NE, SemCSR, SemSWL} is denoted as 'SEM' in the later experiments. The results reflect that lexical semantic information has significant contribution to fine-grained classification, even without word sense disambiguation. In this experiment we also noticed that the class-specific word lists (SemCSR), and similar word lists (SemSWL) are the most independent beneficial sources of semantic information because the coverage and the usage of them. All the relevant semantic categories (synonyms in some sense) are output as features for a word, so a broader range of words will output the same feature which enforce its influence in learning.

An interesting phenomenon is that named entity and

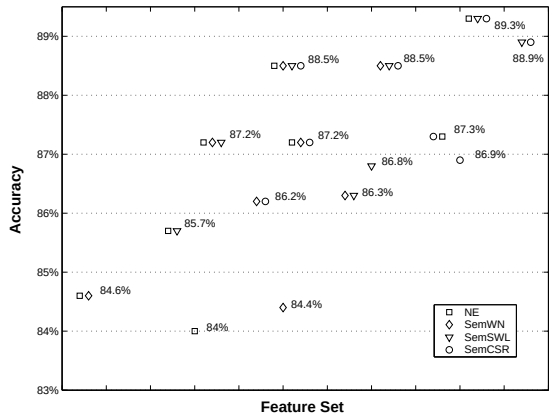

Figure 2: Classification Accuracy for fine classes after adding different combinations of semantic features to the input of the SYN classifier. The X-axis is just a random arrangement of feature sets without a unit. Shapes in the graph represent the four types of semantic feature {NE, SemWN, SemCSR, SemSWL} defined in Sec. 3.2 and a juxtaposition of symbols represents the use of a combination of different types(in addition to SYN). For example, $\bigtriangledown\bigcirc$ denotes that the classifier takes as input a combination of feature types {SYN, SemCSR, SemSWL}.

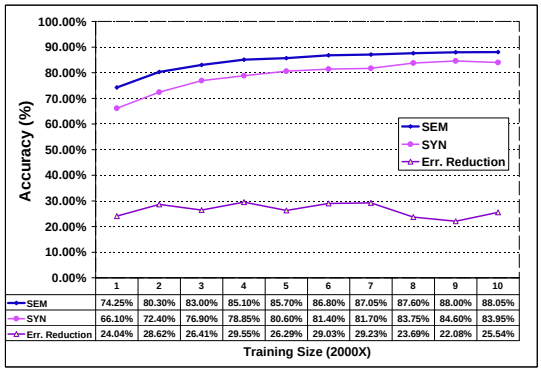

Figure 3: Classification Accuracy versus training size. 'SYN' and 'SEM' represent the learning curves of the SYN classifier and the SEM classifier respectively. 'Err. Reduction' denotes the error reduction from the SYN classifier to the SEM classifier. The training size is $2000 \times X$ and the test set is 1,000 TREC questions.

WordNet features degrades the classification accuracy to below baseline when used independently. The possible reasons behind this are: named entities have a very small coverage over the words (0.23 active features per question); and WordNet adds more noise compared with other semantic sources[4]. However, when combined with other sources they do achieve an improvement in accuracy.

Classification Performance vs. Training Size

The relation between classification accuracy of the SYN classifier and the SEM classifier, and training size, is tested in the third experiment and results are given in Figure 3. The error reduction from the SYN classifier to the SEM classifier on the 1,000 TREC questions is stable over 20% over all training sizes, also proving the distinctive contribution of semantic features.

4.2 Further Analysis

Some other interesting phenomena have also been observed in our experiments. The classification accuracy of the SEM classifier for individual fine classes is given by Table 5. The results indicate that accuracies for them are far from uniform, reflecting different difficulties in identifying them. Questions belonging to **Desc (description)** and **Entity:other (uncommon entities)** are the most difficult, since their boundaries with other classes are ill-defined.

Class	#	Precision[c]	Class	#	Precision[c]
abb	2	100%	desc	25	36%
exp	17	94.11%	manner	8	87.5%
animal	27	85.18%	reason	7	85.71%
body	4	100%	gr	19	89.47%
color	12	100%	ind	154	90.25%
creative	13	76.92%	title	4	100%
currency	6	100%	desc	3	100%
disease	4	50%	city	41	97.56%
event	4	75%	country	21	95.23%
food	6	100%	mount	2	100%
instru	1	100%	LOC:other	116	89.65%
lang	3	100%	state	14	78.57%
ENTY:other	24	37.5%	count	24	91.66%
plant	3	100%	date	145	100%
product	6	66.66%	dist	37	97.29%
religion	1	100%	money	6	100%
sport	4	75%	NUM:other	15	93.33%
substance	21	80.95%	period	20	85%
symbol	2	100%	perc	9	77.77%
termeq	22	63.63%	speed	8	100%
veh	7	71.42%	temp	4	100%
def	125	97.6%	weight	4	100%
TOTAL	**1000**	**89.3%**			

Table 5: Classification Accuracy for specific fine classes with the feature set SEM. # denotes the number of predictions made for each class and Precision[c] denotes the classification accuracy for a specific class c. The classes not shown do not actually occur in the test collection.

To better understand the classification results, we also split the 1,000 test questions into different groups according to their question words, that is, *What*, *Which*, *Who*, *When*, *Where*, *How* and *Why* questions. A baseline classifier, **Wh-Classifier**, is constructed by classifying each group of questions into its most typical fine class. Table 6 shows recall (defined as

[4]Only one or two sense features for each word is correct when using WordNet. SemSWL may have a lower ambiguity.

$\frac{\#\ of\ correct\ predicted\ questions}{\#\ of\ test\ questions}$) of the Wh-Classifier and the SEM classifier on different groups of questions. The typical fine classes in each group and the number of questions in each class are also given. The distribution of *What* questions over the semantic classes is quite diverse, and therefore more difficult to classify than other groups. From this table, we have also noticed that classifying questions simply based on question words (1) does not corresponds well to the desired taxonomy, and (2) is too crude since a large fraction of the questions are 'What' questions.

Question Word	#	Wh	SEM	Classes(#)
What	598	21.07%	85.79%	ind.(36), def.(126), loc–other(47)
Which	21	33.33%	95.24%	ind.(7), country(5)
Who	99	93.94%	96.97%	group(3), ind.(93), human desc.(3)
When	96	100%	100%	date(96)
Where	66	90.01%	92.42%	city(1), mount.(2), loc–other(61)
How	86	30.23%	96.51%	count(21), dist.(26), period(11)
Why	4	100%	100%	reason(4)
Total	1000	41.3%	89.3%	

Table 6: Classification Accuracy of the Wh-Classifier and the SEM classifier on different question groups. Typical fine classes in each group and the number of questions in each class are also shown by **Classes(#)**.

The overall accuracy of our learned classifier is satisfactory. Nevertheless, it is constructive to consider some cases in which the classifier fails. Below are some examples misclassified by the SEM classifier.

- *What imaginary line is halfway between the North and South Poles ?* The correct label is **location**, but the classifier outputs an arbitrary class. Our classifier fails to determine that 'line' might be a location even with the semantic information, probably because some of the semantic analysis is not context sensitive.
- *What is the speed hummingbirds fly ?* The correct label is **speed**, but the classifier outputs **animal**. Our feature extractor fails to determine that the focus of the question is 'speed'. This example illustrates the necessity of identifying the question focus by analyzing syntactic structures.
- *What do you call a professional map drawer ?* The classifier returns **other entities** instead of **equivalent term**. In this case, both classes are acceptable. The ambiguity causes the classifier not to output **equivalent term** as the first choice.

5 Conclusion and Future Directions

This paper presents a machine learning approach to question classification. We developed a hierarchical classifier that is guided by a layered semantic hierarchy of answers types, and used it to classify questions into fine-grained classes. Our experimental results exhibit benefits of the enhanced feature representation from lexical semantic analysis. While the contribution of syntactic information sources to the process of learning classifier has been well studied, we hope that this work can inspire the systematic studies of the contribution of sematic information to classification.

One future step along this line of work would be to improve the selection of the semantic classes using context sensitive methods for most of the semantic information sources and to enlarge the coverage of the named entity recognizer. Furthermore, we hope to extend this work to support interactive question answering. In this task, the question answering could be able to interact with users which may require even larger coverage of semantic classes and more robustness.

References

E. Brill and P. Resnik. 1994. A rule-based approach to prepositional phrase attachment disambiguation. In *COLING*.

A. Carlson, C. Cumby, J. Rosen, and D. Roth. 1999. The SNoW learning architecture. Technical Report UIUCDCS-R-99-2101, UIUC Computer Science Department, May.

Y. Even-Zohar and D. Roth. 2001. A sequential model for multi-class classification. In *EMNLP*, pages 10–19.

C. Fellbaum, editor. 1998. *WordNet: An Electronic Lexical Database*. The MIT Press, May.

U. Hermjakob. 2001. Parsing and question classification for question answering. In *ACL-2001 Workshop on Open-Domain Question Answering*.

E. Hovy, L. Gerber, U. Hermjakob, C. Lin, and D. Ravichandran. 2001. Toward semantics-based answer pinpointing. In *the DARPA HLT conference*.

Y. Krymolowski and D. Roth. 1998. Incorporating knowledge in natural language learning: A case study. In *COLING-ACL'98 workshop on the Usage of WordNet in Natural Language Processing Systems*.

L. Lee. 1999. Measures of distributional similarity. In *ACL*, pages 25–32.

X. Li and D. Roth. 2002. Learning question classifiers. In *COLING*, pages 556–562.

D. Moldovan, S. Harabagiu, R. Girju, P. Morarescu, F. Lacatusu, A. Novischi, A. Badulescu, and O. Bolohan. 2002. Lcc tools for question answering. In E. Voorhees, editor, *TREC*.

P. Pantel and D. Lin. 2002. Discovering word senses from text. In *KDD*.

V. Punyakanok and D. Roth. 2001. The use of classifiers in sequential inference. In *NIPS*, pages 995–1001. MIT Press.

D. Ravichandran and E.H. Hovy. 2002. Learning surface text patterns for a question answering system. In *ACL*.

S. Robertson, S. Walker, and M. Beaulieu. 1998. Okapi at trec-7: Automatic ad hoc, filtering, vlc and interative track. In *TREC*.

D. Roth. 1998. Learning to resolve natural language ambiguities: A unified approach. In *AAAI*.

F. Sebastiani. 2002. Machine learning in automated text categorization. *ACM Computing Surveys*, 34(1):1–47.

J. Suzuki, H. Taira, Y. Sasaki, and E. Maeda. 2003. Question classification using hdag kernel. In *ACL Workshop on Multilingual Summarization and Question Answering*.

E. Voorhees. 2002. Overview of the TREC-2002 question answering track. In *TREC*, pages 115–123.

D. Zhang and W. Lee. 2003. Question classification using support vector machines. In *the 26th ACM SIGIR*. ACM Press.