A Language Independent Method for Question Classification

Thamar Solorio[1], **Manuel Pérez-Coutiño**[1], **Manuel Montes-y-Gómez**[1,2]
Luis Villaseñor-Pineda[1] and **Aurelio López-López**[1]
[1]Language Technologies Group, Computer Science Department
National Institute of Astrophysics, Optics and Electronics
72840 Tonantzintla, Puebla,
Mexico
[2]Departamento de Sistemas Informáticos y Computación
Universidad Politécnica de Valencia
España
{thamy,mapco,mmontesg,villasen,allopez}@inaoep.mx

Abstract

Previous works on question classification are based on complex natural language processing techniques: named entity extractors, parsers, chunkers, etc. While these approaches have proven to be effective they have the disadvantage of being targeted to a particular language. We present here a simple approach that exploits lexical features and the Internet to train a classifier, namely a Support Vector Machine. The main feature of this method is that it can be applied to different languages without requiring major modifications. Experimental results of this method on English, Italian and Spanish show that this approach can be a practical tool for question answering systems, reaching a classification accuracy as high as 88.92%.

1 Introduction

Open-domain Question Answering (QA) systems are concerned with the problem of trying to answer questions from users posed in natural language. What makes these systems a very complex and interesting research area is that the answers they retrieve must be concise, as opposed to traditional search engines that in response to a user query retrieve a list of documents believed to contain the answer. Moreover, current evaluation environments of QA systems, such as TREC QA track (Voorhees, 2001) and CLEF (Peters et al., 2003), restrict the size of the answers to a maximum of 50 bytes. Given the complexity involved in this problem, traditional approaches to QA take a divide-and-conquer strategy, where the problem is divided into several less complex subtasks that combined lead to the resolution of the questions. An important subtask of a QA system is question analysis, since it can provide useful clues for identifying potential answers in a large collection of texts. For instance, Question Classification is concerned with assigning semantic classes to questions. This semantic classification can be used to reduce the search space of possible answers, i.e. if we can determine that the question *Who is the Italian Prime Minister?* belongs to the semantic category PERSON, then we only need to look for instances of type PERSON as possible answers. Clearly, the advantage of such classification relies on having the ability of extracting from the documents such instances. In other words, a good question classification module may be useless if we lack an accurate named entity extractor for the document collection.

Results of the error analysis of an open-domain QA system showed that 36.4% of the errors were generated by the question classification module (Moldovan et al., 2003). Thus it is not surprising that an increasing interest has arisen aimed at developing accurate question classifiers (Zhang and Lee, 2003; Li and Roth, 2002; Suzuki et al., 2003). However, most of these approaches are targeted to the English language. Besides, the machine learning algorithms used are trained on features extracted by natural language processing tools that are language dependent, and for some languages these tools are not available. This implies that if we want to reproduce the results of these methods in a different language we need first to solve the problem of making available the appropriate analyzers in the given language.

We present here a flexible method for question classification. We claim that the method is language-independent since no complex nat-

ural language processing tools are needed; we use plain lexical features that can be extracted automatically from the questions. A machine learning algorithm that has proven to perform well over high dimensional data, is trained on prefixes of words and on additional attribute information gathered automatically from the Internet. The method was evaluated experimentally, achieving high accuracy on questions in three different languages: English, Italian and Spanish.

The next section briefly summarizes some of the previous approaches for question classification. Section 3 presents the learning scenario of this work, together with a brief introduction to Support Vector Machines (SVM). Section 4 shows our experimental results and we conclude with a discussion of this work and ideas for future research in Section 5.

2 Related Work

Most approaches to question classification are based on handcrafted rules (Voorhees, 2001). It is not until recently that machine learning techniques are being used to tackle the problem of question classification. In (Zhang and Lee, 2003) they present a new method for question classification using Support Vector Machines. They compared accuracy of SVM against Nearest Neighbors, Naive Bayes, Decision Trees and Sparse Network of Winnows (SNoW), with SVM producing the best results. In their work, Zhang and Sun Lee improve accuracy by introducing a tree kernel function that allows to represent the syntactic structure of questions. Their experimental results show that SVM using this tree kernel function achieves an accuracy of 90%, however, a parser is needed in order to acquire the syntactic information.

Li and Roth reported a hierarchical approach for question classification based on the SNoW learning architecture (Li and Roth, 2002). This hierarchical classifier discriminates among 5 coarse classes, which are then refined into 50 more specific classes. The learners are trained using lexical and syntactic features such as pos tags, chunks and head chunks together with two semantic features: named entities and semantically related words. They reported question classification accuracy of 98.80% for a coarse classification, using 5,500 instances for training.

A different approach, used for Japanese question classification, is that of Suzuki *et al.* (Suzuki et al., 2003). They used SVM whith a new kernel function, called Hierarchical Directed Acyclic Graph, which allows the use of structured data. They experimented with 68 question types and compared performance of using bag-of-words against using more elaborated combinations of attributes, namely named entities and semantic information. Their best results, an accuracy of 94.8% at the first level of the hierarchy, were obtained when using SVM trained on bag-of-words together with named entities and semantic information.

The idea of using the Internet in a QA system is not new. What is new, however, is that we are using the Internet to obtain values for features in our question classification process, as opposed to previous approaches where the redundancy of information available on the Internet has been used in the answer extraction process (Brill et al., 2002; Lin et al., 2002; Katz et al., 2003).

3 Learning Question Classifiers

Question classification is very similar to text classification. One thing they have in common is that in both cases we need to assign a class, from a finite set of possible classes, to a natural language text. Another similarity is attribute information; what has been used as attributes for text classification can also be extracted and used in question classification. Finally, in both cases we have high dimensional attributes: if we want to use the bag-of-words approach, we will face the problem of having very large attribute sets.

An important difference is that question classification introduces the problem of dealing with short sentences, compared with text documents, and thus we have less information available on each question instance. This is the reason why question classification approaches are trying to use other information (e.g. chunks and named entities) besides the words within the questions. However, the main disadvantage of relying on semantic analyzers, named entity taggers and the like, is that for some languages these tools are not yet well developed. Plus, most of them are very sensitive to changes in the domain of the corpus; and even if these tools are accurate, in some cases acquiring one for a particular language may be a difficult task. This is our prime motivation for searching for different, more easier to gather, information to solve the question classification problem. Our learning scenario considers as attribute information

prefixes of words in combination with attributes whose values are obtained from the Internet. These Internet based attributes are targeted to extract evidence of the possible semantic class of the question.

The next subsection will explain how the Internet is used to extract attributes for our question classification problem. In subsection 3.2 we present a brief description of Support Vector Machines, the learning algorithm used on our experiments.

3.1 Using Internet

As Kilgarriff and Grefenstette wrote, the Internet is a fabulous linguists' playground (Kilgarriff and Grefenstette, 2003). It has become the greatest information source available worldwide, and although English is the dominant language represented on the Internet it is very likely that one can find information in almost any desired language. Considering this, and the fact that the texts are written in natural language, we believe that new methods that take advantage of this large corpus must be devised. In this work we propose using the Internet in order to acquire information that can be used as attributes in our classification problem. This attribute information can be extracted automatically from the web and the goal is to provide an estimate about the possible semantic class of the question.

The procedure for gathering this information from the web is as follows: we use a set of heuristics to extract from the question a word w, or set of words, that will complement the queries submitted for the search. We then go to a search engine, in this case Google, and submit queries using the word w in combination with all the possible semantic classes for our purpose. For instance, for the question *Who is the President of the French Republic?* we extract the word *President* using our heuristics, and run 5 queries in the search engine, one for each possible class. These queries take the following form:

- "President is a person"
- "President is a place"
- "President is a date"
- "President is a measure"
- "President is an organization"

We count the number of results returned by Google for each query and normalize them by their sum. The resultant numbers are the values for the attributes used by the learning algorithm. As can be seen, it is a very straightforward approach, but as the experimental results will show, this information gathered from the Internet is quite useful. In Table 1 we present the figures obtained from Google for the question presented above, column *Results* show the number of hits returned by the search engine and in column *Normalized* we present the number of hits normalized by the total of all results returned for the different queries.

An additional advantage of using the Internet is that by approximating the values of attributes in this way, we take into account words or entities belonging to more than one class (polysemy).

Now that we have introduced the use of the Internet in this work, we continue describing the set of heuristics that we use in order to perform the web search.

3.1.1 Heuristics

We begin by eliminating from the questions all words that appear in our stop list. This stop list contains the usual items: articles, prepositions and conjunctions plus all the interrogative adverbs and all lexical forms of the verb "to be". The remaining words are sent to the search engine in combination with the possible semantic classes, as described above. If no results are returned for any of the semantic classes we then start eliminating words from right to left until the search engine returns results for at least one of the semantic categories. As an example consider the question posed previously: *Who is the President of the French Republic?* we eliminate the words from the stop list and then formulate queries for the remaining words. These queries are of the following form: *"President French Republic is a* s_i*"* where $s \in \{Person, Organization, Place, Date, Measure\}$. The search engine did not return any results for this query, so we start eliminating words from right to left. The query is now like this: *"President French is a* s_i*"* and given that again we have no results returned we finally formulate the last possible query: *"President is a* s_i*"* which returns results for all the semantic classes except for *Date*.

Being heuristics, we are aware that in some cases they do not work well. Nevertheless, for the vast majority of the cases they presented surprisingly good results, in the three

Query	Results	Normalized
"President is a person"	259	0.8662
"President is a place"	9	0.0301
"President is an organization"	11	0.0368
"President is a measure"	20	0.0669
"President is a date"	0	0

Table 1: Example of using the Internet to extract features for question classification

Class	Number of Instances
Person	91
Organization	41
Measure	103
Date	64
Object	12
Other	54
Place	85

Table 2: Distribution of semantic classes for the DISEQuA corpus

languages, as shown in the experimental evaluation.

3.2 Support Vector Machines

Given that Support Vector Machines have proven to perform well over high dimensionality data they have been successfully used in many natural language related applications, such as text classification (Joachims, 1999; Joachims, 2002; Tong and Koller, 2001) and named entity recognition (Mitsumori et al., 2004; Solorio and López, 2004). This technique uses geometrical properties in order to compute the hyperplane that best separates a set of training examples (Stitson et al., 1996). When the input space is not linearly separable SVM can map, by using a kernel function, the original input space to a high-dimensional feature space where the optimal separable hyperplane can be easily calculated. This is a very powerful feature, because it allows SVM to overcome the limitations of linear boundaries. They also can avoid the over-fitting problems of neural networks as they are based on the structural risk minimization principle. The foundations of these machines were developed by Vapnik, for more information about this algorithm we refer the reader to (Vapnik, 1995; Schölkopf and Smola, 2002).

4 Experimental Evaluation

4.1 Data sets

The data set used in this work consists of the questions provided in the DISEQuA Corpus (Magnini et al., 2003). Such corpus was made up of simple, mostly short, straightforward and factual queries that sound naturally spontaneous, and arisen from a real desire to know something about a particular event or situation. The DISEQuA Corpus contains 450 questions, each one formulated in four languages: Dutch, English, Italian and Spanish. The questions are classified into seven categories: *Person*, *Organization*, *Measure*, *Date*, *Object*, *Other* and *Place*. The experiments performed in this work used the English, Italian and Spanish versions of these questions.

4.2 Experiments

In the experiments performed in this work we used the evaluation technique 10-fold cross-validation which consists of randomly dividing the data into 10 equally-sized subgroups and performing 10 different experiments. We separated nine groups together with their original classes as the training set, the remaining group was considered the test set. Each experiment consists of ten runs of the procedure described above, and the overall average are the results reported here.

In our experiments we used the WEKA implementation of SVM (Witten and Frank, 1999). In this setting multi-class problems are solved using pairwise classification. The optimization algorithm used for training the support vector classifier is an implementation of Platt's sequential minimal optimization algorithm (Platt, 1999). The kernel function used for mapping the input space was a polynomial of exponent one.

The most common approach to question classification is bag-of-words, so we decided to compare results of using bag-of-words against using just prefixes of the words in the questions. In

Language	Words	Prefix-5	Prefix-4	Internet
ENGLISH	**81.77%**	81.32%	80.21%	67.77%
ITALIAN	88.03%	87.59%	**88.70%**	60.79%
SPANISH	79.90%	**81.45%**	76.97%	68.86%

Table 3: Experimental results of accuracy when training SVM with words, prefixes and Internet-based attributes

order to choose an appropriate prefix size we compute the average length of the words in the three languages used in this work. For English the average length of words is 4.62, for Italian is 4.8 while for Spanish the average length is 4.75. So we decided to experiment with prefixes of size 4 and 5. In Table 3 we can see a comparison of classification accuracy of training SVM using all the words in the questions, using prefixes of size 4 and 5 and using only the Internet-based attributes. As we can see for English the best results were obtained when using words as attributes, although the difference between using just prefixes and using words is not so large. For Spanish however, the best results were achieved when using prefixes of size 5. This can be due to the fact that some of the interrogative words, that by themselves can define the semantic class of questions in this language, such as *Cuándo* (When) and *Cuánto* (How much) could be considered as the same prefix of size 4 i.e. *Cuán*. But if we consider prefixes of size 5, then these two words will form two different prefixes: *Cuánd* and *Cuánt*, thus reducing the loss of information, as oppose to using prefixes of size 4. For Italian language the best results were obtained from using prefixes of size 4. And for the three languages the Internet-based attributes had rather low accuracies, the lowest being for Italian. When we analyzed the results computed for Italian, using our Internet-based attributes, we realized that in many cases we could not get any results to the queries. One plausible explanation for this lack of information, is that the number of Italian documents available on Internet is much smaller than for English and Spanish. Estimates reported in (Kilgarriff and Grefenstette, 2003) show that for Italian the web size in words is 1,845,026,000; while for English and Spanish the web sizes are 76,598,718,000 and 2,658,631,000 respectively. Thus our method was not able to extract as much information as for the other two languages.

4.3 Combining Internet-based Attributes with Lexical Features

Results presented in the previous subsection show how by using just lexical information we can train SVM and achieve high accuracies in the three languages. But our goal is to discover the usefulness of using Internet in order to extract attributes for question classification. We performed other experiments combining the lexical attributes with the Internet information in order to discover if we can further improve accuracy. Table 4 show experimental results of this attribute combination and Figure 1 shows a graphical representation of these results.

It is interesting to note that for English and Spanish we did gain accuracy when using the Internet features in all the cases. In contrast, for Italian classification accuracy was decreased when incorporating Internet-based attributes to words and prefixes of size 5. We believe that this drop in accuracy for Italian may be due to the weakly supported information extracted from the Internet, Table 3 shows that SVM trained only on the coefficients from the Internet performed worse for Italian. It is not surprising that adding this rather sparse information to the attributes in the Italian language did not produce an advantage in the classifiers performance.

5 Conclusions

We have presented here experimental results of a language independent question classification method. The method is claimed to be language independent since the features used as attributes in the learning task can be extracted from the questions in a fully automated manner; we do not use semantic or syntactic information because otherwise we will be restricted to work on languages for which we do have parsers that can extract this information. We believe that this method can be successfully applied to other languages, such as Romanian, French, Portuguese and Catalan, that share the morphologic characteristics of the three languages

Language	Words+Internet	Prefix-5 + Internet	Prefix-4 + Internet
ENGLISH	82.88%	82.66%	**83.55%**
ITALIAN	87.34%	86.93%	**88.92%**
SPANISH	83.43%	**84.09%**	81.45%

Table 4: Experimental results combining Internet-valued attributes with words and prefixes

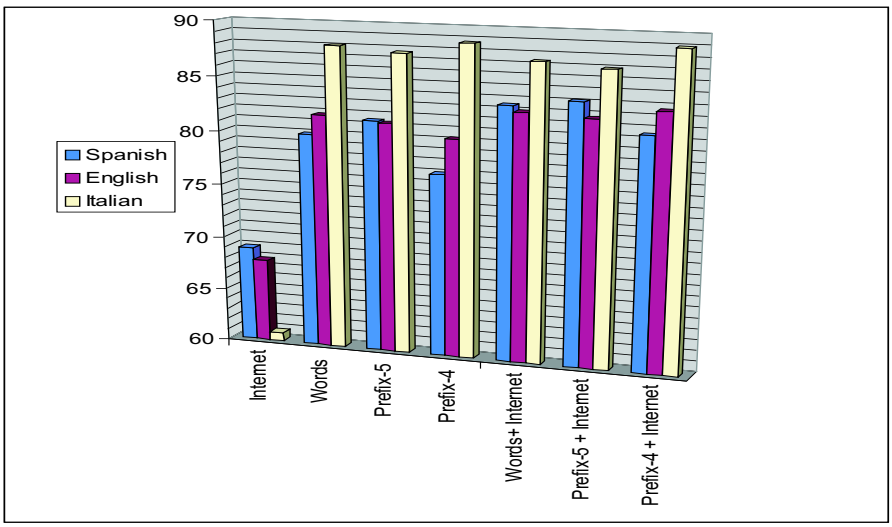

Figure 1: Graphical comparison of question classification accuracies

tested here.

Comparing our results with those of previous works we can say that our method is promising. For instance Zhang and Sun Lee (Zhang and Lee, 2003) reported an accuracy of 90% for English questions, while Li and Roth (Li and Roth, 2002) achieved 98.8% accuracy. However, they used a training set of 5,500 questions and a test set of 500 questions, while in our experiments we used for training 405 for each 45 test questions (10-fold-cross-validation). When Zhang and Sun Lee used only 1,000 questions for training they achieved an accuracy of 80.2%. It is well known that machine learning algorithms perform better when a bigger training set is available, so it is expected that experiments of our method with a larger training set will provide improved results.

As future work we plan to investigate active learning with SVM for this problem. Given that manually labelling questions is a very time consuming task, active learning can provide a faster approach to build accurate question classifiers. Instead of randomly selecting question instances to label manually and then provide them to the learner, the learner can analyze the unlabeled instances and select for labelling the instances that seem more relevant to the task.

Another interesting line for future work is exploring the advantage of using mixed languages corpora lo learn question classification. The Romance languages, for instance, such as Italian, French and Spanish have stems in common. Then it is feasible that questions for several languages may help to train a classifier for a different language. The advantage of this idea will be the availability of larger corpora for languages for which a large enough corpus is not available, counting in favor of languages that are underrepresented on the Internet. We could circumvent this lack of presence on the Internet for some languages by using information available on other, more well represented, languages.

6 Acknowledgements

We would like to thank CONACyT for partially supporting this work under grants 166934, 166876 and U39957-Y, and Secretaría de Estado de Educación y Universidades de España.

References

E. Brill, S. Dumais, and M. Banko. 2002. An analysis of the AskMSR question-answering system. In *2002 Conference on Empirical Methods in Natural Language Processing.*

T. Joachims. 1999. Transductive inference for

text classification using support vector machines. In *Proceedings of the Sixteenth International Conference on Machine Learning (ICML)*, pages 200–209. Morgan Kaufmann.

T. Joachims. 2002. *Learning to Classify Text using Support Vector Machines: Methods Theory and Algorithms*, volume 668 of *The Kluwer International Series in Engineering and Computer Science*. Kluwer Academic Publishers.

B. Katz, J. Lin, D. Loreto, W. Hildebrandt, M. Bilotti, S. Felshin, A. Fernandes, G. Marton, and F. Mora. 2003. Integrating web-based and corpus-based techniques for question answering. In *Twelfth Text REtrieval Conference (TREC 2003)*, Gaithersburg, Maryland, November.

A. Kilgarriff and G. Grefenstette. 2003. Introduction to the special issue on the web as corpus. *Computational Linguistics*, 29(3):333–347.

X. Li and D. Roth. 2002. Learning question classifiers. In *COLING'02*.

J. Lin, A. Fernandes, B. Katz, G. Marton, and S. Tellex. 2002. Extracting answers from the web using knowledge annotation and knowledge mining techniques. In *Eleventh Text REtrieval Conference (TREC 2002)*, Gaithersburg, Maryland, November.

B. Magnini, S. Romagnoli, A. Vallin, J. Herrera, A. Peñas, V. Peinado, F. Verdejo, and M. de Rijke. 2003. Creating the DISEQuA corpus: a test set for multilingual question answering. In Carol Peters, editor, *Working Notes for the CLEF 2003 Workshop*, Trondheim, Norway, August.

T. Mitsumori, S. Fation, M. Murata, K. Doi, and H. Doi. 2004. Boundary correction of protein names adapting heuristic rules. In Alexander Gelbukh, editor, *Fifth International Conference on Intelligent Text Processing and Computational Linguistics, CICLing 2004*, volume 2945 of *Lecture Notes in Computer Science*, pages 172–175. Springer.

D. Moldovan, M. Paşca, S. Harabagiu, and M. Surdeanu. 2003. Performance issues and error analysis in an open-domain question answering system. *ACM Trans. Inf. Syst.*, 21(2):133–154.

C. Peters, M. Braschler, J. Gonzalo, and M. Kluck, editors. 2003. *Advances in Cross-Language Information Retrieval, Third Workshop of the Cross-Language Evaluation Forum, CLEF 2002. Rome, Italy, September 19-20, 2002. Revised Papers*, volume 2785 of *Lecture Notes in Computer Science*. Springer.

J. Platt. 1999. Fast training of support vector machines using sequential minimal optimization. In *Advances in Kernel Methods –Support Vector Learning, (B. Schölkopf, C.J.C. Burges, A.J. Smola, eds.)*, pages 185–208, Cambridge, Massachusetts. MIT Press.

B. Schölkopf and A. J. Smola. 2002. *Learning with Kernels: Support Vector Machines, Regularization, Optimization and Beyond*. MIT Press.

T. Solorio and A. López López. 2004. Learning named entity classifiers using support vector machines. In Alexander Gelbukh, editor, *Fifth International Conference on Intelligent Text Processing and Computational Linguistics, CICLing 2004*, volume 2945 of *Lecture Notes in Computer Science*, pages 158–167. Springer.

M. O. Stitson, J. A. E. Wetson, A. Gammerman, V. Vovk, and V. Vapnik. 1996. Theory of support vector machines. Technical Report CSD-TR-96-17, Royal Holloway University of London, England, December.

J. Suzuki, H. Taira, Y. Sasaki, and E. Maeda. 2003. Question classification using HDAG kernel. In *Workshop on Multilingual Summarization and Question Answering 2003*, pages 61–68.

S. Tong and D. Koller. 2001. Support vector machine active learning with applications to text classification. *Journal of Machine Learning Research*, 2:45–66.

V. Vapnik. 1995. *The Nature of Statistical Learning Theory*. Number ISBN 0-387-94559-8. Springer, N.Y.

E. Voorhees. 2001. Overview of the TREC 2001 question answering track. In *Proceedings of the 10th Text REtrieval Conference (TREC01)*, NIST, pages 157–165, Gaithersburg, MD.

I. H. Witten and E. Frank. 1999. *Data Mining, Practical Machine Learning Tools and Techniques with Java Implementations*. The Morgan Kaufmann Series in Data Management Systems. Morgan Kaufmann.

D. Zhang and W. Sun Lee. 2003. Question classification using support vector machines. In *Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 26–32, Toronto, Canada. ACM Press.