Question Classification using Semantic, Syntactic and Lexical features

Megha Mishra[1] ,Vishnu Kumar Mishra [2] and Dr. H.R. Sharma[3]

[1] Research Scholoar, SOA University
megha16shukla@gmail.com
[2]CSE Dept RSR,Bhilai
vshn_mshr@rediffmail.com
[3]Dean R&D, RCET Raipur
hrsharmaji@indiatimes.com

ABSTRACT

Question classification is very important for question answering. This paper present our research work on question classification through machine learning approach. In order to train the learning model, we designed a rich set of features that are predictive of question categories. An important component of question answering systems is question classification. The task of question classification is to predict the entity type of the answer of a natural language question. Question classification is typically done using machine learning techniques. Different lexical, syntactical and semantic features can be extracted from a question. In this work we combined lexical, syntactic and semantic features which improve the accuracy of classification. Furthermore, we adopted three different classifiers: Nearest Neighbors (NN), Naïve Bayes (NB), and Support Vector Machines (SVM) using two kinds of features: bag-of-words and bag-of n grams. Furthermore, we discovered that when we take SVM classifier and combine the semantic, syntactic, lexical feature we found that it will improve the accuracy of classification. We tested our proposed approaches on the well-known UIUC dataset and succeeded to achieve a new record on the accuracy of classification on this dataset.

KEYWORDS

Question Classification, Question Answering Systems, Lexical Features, Syntactical Features, Semantic Features, Combination of Features, Nearest Neighbors (NN), Naïve Bayes (NB), Support Vector Machines (SVM).

1. INTRODUCTIONSS

A question answering (QA) system aims at automatically finding concise answers to arbitrary questions phrased in natural language. It delivers only the requested information, unlike search engines which refer to full documents. For example, given the question *"What was the name of the first German Chancellor?"*1, ideally a QA system would respond with *"Konrad Adenauer"*. This usage is intuitive; it saves time and allows a satisfactory result presentation even on compact mobile devices. Recently QA has been drawing attention: *True Knowledge* [1] is an English language web-based system, and IBM's *Watson* [2] has successfully participated in a quiz show.*LogAnswer* [3,4] is a web-based QA system for the German language It works with a knowledge base (KB) derived from the entire German Wikipedia, and the answers are produced using a synergistic combination of natural language processing (NLP), machine learning (ML) algorithms and automated theorem proving (ATP). Those methods analyze and parse complex question to multi simple questions and use existing techniques for answering them [5**].**

Svetlana Stoyanchev [5] presented a document retrieval experiment on a question answering system, and evaluates the use of named entities and of noun, verb, and prepositional phrases as exact match phrases in a document retrieval query. While [6,7] presented simplest approach to improve the accuracy of a question answering system might be restricting the domain it covered. Paloma Moreda, Hector Llorens et al [8] presented two proposals for using semantic information in QAS, specifically in the answer extraction step. Its aim is to determine the improvement in performance of current QA systems, especially when dealing with common noun questions. Liang Yunjuan , Ma Lijuan, [9,10] discussed the design of dynamic knowledge-based full-text retrieval system, inverted index technology research and analysis, given some of indexing code, in order to improve the retrieval accuracy and to achieve a reasonable. The following table presents comparison about different types of question answering system and methods used in this system. For correct answer extraction, some patterns should be defined for system to find exact type of answer and then sends to document processing. [11][12].

2. Question Classification

2.1. Question type taxonomy

The set of question categories (classes) are usually referred as *question taxonomy* or *question ontology*. Different question taxonomies have been proposed in different works, but most of the recent studies are based on a two layer taxonomy proposed by Li and Roth[13].

Table 1: The coarse and fine grained question classes

Coarse	Fine
ABBR	abbreviation, expansion
DESC	definition, description, manner, reason
ENTY	animal, body, color, creation, currency, disease, event, food, instrument, language, letter, other, plant, product, religion, sport, substance, symbol, technique, term, vehicle, word
HUM	description, group, individual, title
LOC	city, country, mountain, other, state
NUM	code, count, date, distance, money, order, other, percent, percent, period, speed, temperature, size, weight

2.2 Classification Algorithms

In our experiments, three different types of supervised classifiers were used.For all experiments involving SVM, we employed the LIBSVM [14] implementation with a linear kernel, and trained the classifiers using the one-versus-all multi-class strategy. As for the the Naive Bayes and *k* Nearest Neighbors implementation, we adopted the LingPipe [15]software package.

- K-nearest neighbor algorithm
- Naive Bayesian classifier
- Support vector machine
    - Linear SVM
    - Nonlinear SVM
    - Multi-class SVM

3. Evaluation Measures

The evaluation measure used to assess the performance of the question classifier is accuracy – i.e., the fraction of the total number of questions that have been correctly classified. Additionally, the performance of the classifier for each particular class *c* is evaluated using precision and recall.

$$Precision(c) = \frac{\#\text{ of correctly classified questions of category } c}{\#\text{ of predicted questions of category } c}$$

$$Recall(c) = \frac{\#\text{ of correctly classified questions of category } c}{\#\text{ of questions of category } c}.$$

4. Experimental Results

The first experiment was designed to evaluate the individual performance of the three classifiers: Naive Bayes, SVM ,kNN using simple unigrams as features, and under the coarse grained category presented in Table 2. This was an expected finding, since previous literature on this task – such as [18] , had already reported similar results.

Table2. Question classification accuracy using different machine learning algorithms and different training set sizes, under the coarse grained category

	Number of questions in training set				
Algorithm	1000	2000	3000	4000	5500
SVM	78.4%	83.6%	85.0%	86.2%	**88.2%**
Naïve Bayes	70.4%	72.4%	73.0%	75.6%	78.6%
***k*NN**	62.6%	68.6%	72.4%	73.0%	75.2%

5. Question feature set

5.1 Lexical features

Lexical features refer to *word related* features that are extracted directly from the question. In this work,we use word level *n*-grams as lexical features

5.1.1 Word level *n*-grams

A word level *n*-gram is a sequence of *n* consecutive words from a given question. The rationale behind this feature is that questions of the same category tend to share word *n*-grams.For instance, the unigram *city* appears often in questions of type LOCATION:CITY, which can be a good indicator that the question belongs to this category. Another example is the bigram *Who was* which tends to appear associated with questions of type HUMAN:DESCRIPTION.

5.1.2 Stemming and Stopword removal

Stemming is a technique that reduces words to their grammatical roots or *stems*, by removing their Affixes. First, we represent the question using the bag-of-words model Second, we apply Porter's stemming algorithm [24] to transform each word into its stem.

5.1.3 word shapes

It refers to apparent properties of single words. Huang et al.[20] introduced 5 categories for word shapes: *all digit*, *lower case*, *upper case*, *mixed* and *other*.

5.2 Syntactic features

Syntactic features denote *syntax related* features, that require an analysis of the grammatical structure of the question to be extracted.

5.2.1 Question headword

The question headword is a word in a given question that *represents* the information that is being sought after, for example What is Australia's national **flower**? Here the headword is in bold face. the headword *flower* provides the classifier with an important clue to correctly classify the question to ENTITY:PLANT.

For natural language sentences written in English language, English grammar rules are used to create syntax tree.There are successful parsers that can parse a sentence and form the syntax tree[21]. These parsers are statistical-based parsers which parse an English sentence based on *Probabilistic Context-Free Grammars* (pcfg) in which every rule is annotated with the probability of that rule being used. The rule's probabilities was learned based on a supervised approach on a training set of 4,000 parsed and annotated questions known as treebank (Judge et al., 2006). These parsers typically maintain an accuracy of more than 95%. Jurafsky and Martin [25] provided a detailed overview of parsing approaches. The list of English pos tags which is used for parsing syntax tree is listed in appendix A. In this work we used Standford pcfg parser[21] has suggested the combined feature approach.

Algorithm 1 Headword extraction algorithm
procedure Extract-Question-Headword (tree)
if IsTerminal(tree) **then**
return tree
else
head-child ← Apply-Rules(tree)
return Extract-Question-Headword (head-child)
end if
end procedure

5.2.2 Question patterns

here are still some (albeit few) ques tion categories for which our definition of headword doesn't help classification. For instance, in DESCRIPTION:DEFINITION questions such as *What is a bird?*,the headword *bird* is futile because the question is asking for a definition. To prevent some of these pitfalls, we compile a small set of patterns (some of which are adapted from [20], so that when a question matches one of the patterns, a placeholder is returned instead of the question headword.

Algorithm 2 Question headword feature extraction algorithm
procedure HEADWORD-FEATURE(*question*)
if PATTERN-MATCHES?(*question*) **then**
return *placeholder*
else
return EXTRACT-QUESTION-HEADWORD(*question.tree, rules*) ◃ Algorithm 1
end if
end procedure

5.2.3 Part-of-speech tags

the parse tree of a question, we extract the pre-terminal nodes to use as features. These nodes represent the part-of-speech (POS) tags or grammatical classes of the question tokens. For example, the POS tags of the question "*What is the capital of France?*" are: *WP*, *VBD*, *DT*, *NN*, *IN*, and *NNP*.

5.3 Semantic features contribution

Semantic features are extracted based on the semantic meaning of the words in a question. We extracted different type of semantic features. Most of the semantic features requires a third party data source such as WordNet [23], or a dictionary to extract semantic information for questions.

5.3.1Hypernyms

WordNet is a lexical database of English words which provides a lexical hierarchy that associates a word with higher level semantic concepts namely *hypernyms*. For example a hypernym of the word "city" is "municipality" of which the hypernym is "urban area" and so on. As hypernyms allow one to abstract over specific words, they can be useful features for question classification. Extracting hypernyms however, is not straightforward. There are four challenges that should be addressed to obtain hypernym features:

1. For which word(s) in the question should we find hypernyms?
2. For the *candidate* word(s), which part-of-speech should be considered?
3. The candidate word(s) augmented with their part-of-speech may have different senses in WordNet. Which sense is the sense that is used in the given question?
4. How far should we go up through the hypernym hierarchy to obtain the optimal set of hypernyms?

To address the first challenge we considered two different scenarios: either to consider the headword as the candidate word for expansion or expanding all the words in the question by their hypernyms. For the second issue the pos tag which extracted from syntactical structure of question is considered as the target pos tag of the chosen candidate word. To tackle the third issue, the right sense of the candidate word should be determined to be expanded with its hypernyms. We adopted Lesk'sWord Sense Disambiguation (wsd), (Lesk, 1986) algorithm to determine the true sense of word according to the sentence it appears. To address the fourth challenge we found that expanding the headword with hypernyms of maximum dept 6 will have the best result. In the next chapter we will show the influence of hypernym's dept on classification accuracy.

5.3.2 Question Category

We extracted a successful semantic feature namely *question category* which is obtained by exploiting WordNet hierarchy based on the idea of Huang et al. [20]s. We used WordNet hierarchy to calculate the similarity of question's headword with each of the classes. The class with highest similarity is considered as a feature and will be added to the feature vector. In fact this is equal to a *mini-classi_cation* although the acquired class will not be used as final class; since it is not as accurate as the original classifier.

6. Comparison with other works

6.1 Lexico-syntactic features contribution

We trained a SVM classifier using different combinations of both lexical and syntactic features, in order to assess their individual and combined contribution to question classification. In sum, we can conclude that the most prominent and discriminative lexico-syntactic features are the question headword and unigrams and word shapes which contradicts the results obtained by (F. Li et al., 2008). In the experiment that follows, we experiment the use of these three features in combination with semantic features.

6.2 Semantic features contribution

The experiment was designed to measure the contribution of semantic features to question classification. Specifically, we experimented different combinations of semantic features with the lexico-syntactic features that yielded the most informative results in the previous experiment.The best accuracy attained in this experiment for both coarse- and fine-grained classification – 96.2% and 91.1%, respectively which we can see from fig1. and fig2. this is achieved by using the combination of the question headword, hypernyms(wordnet),word shapes,question category and unigrams.

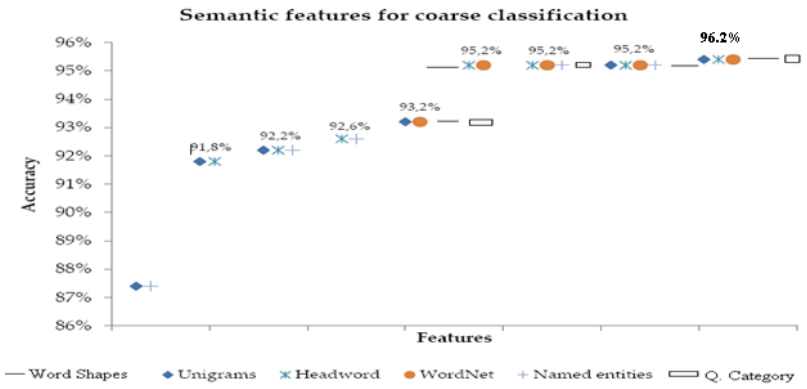

Figure 1.: Question classification accuracy using different combinations of lexical, syntactic, and semantic features,under the coarse-grained category. Juxtaposed symbols represent a combination of the corresponding symbols'features

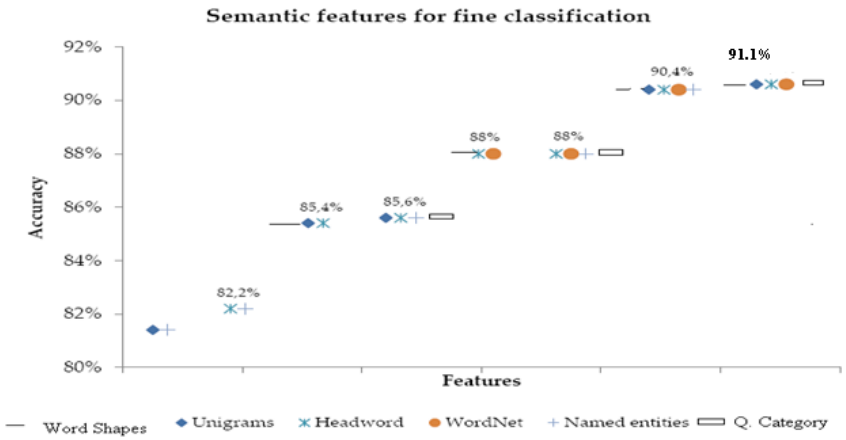

Figure 2.: Question classification accuracy using different combinations of lexical, syntactic, and semantic features,under the fine-grained category. Juxtaposed symbols represent a combination of the corresponding symbols'features

6.3 Comparison with other works

We now compare our results with others reported in the literature. Table1. summarises the question classification accuracy reached by other relevant works in the literature for this particular task. All works were evaluated using similar settings as this work, with the same question type taxonomy, and the same training and test sets.

Table 1.: Comparison of different supervised learning studies on question classification

Study	**Classifier**	**Features**	**Accuracy**	
			Coarse	**Fine**
Krishnan et al. (2005)	Linear SVM	U+B+T+IS+HY	94.2%	88.0%
Blunsom et al. (2006)	ME	U+B+T+P+H+NE+more	92.6%	86.6%
Merkel et al. (2007)	Language Modeling	U+B	-	80.8%
Li et al. (2008)	SVM+CRF	U+L+P+H+HY+NE+S	-	85.6%
Pan et al. (2008)	Semantic tree	U+NE+S+IH	-	94.0%
Huang et al. (2008)	ME	U+WH+WS+H+HY+IH	93.6%	89.0%
Huang et al. (2008)	Linear SV	U+WH+WS+H+HY+IH	93.4%	89.2%
Loni et al. (2011)	Linear SV	U+B+WS+H+HY+R	93.6%	89.0%
This Work	Linear SVM	U+H+HY+WS+QC	**96.2**%	**91.1%**

From the comparison we can see that in our approach we can get the accuracy for coarse grain 96.2% and for fine grain 91.1% which is much better from previous one.

7. Conclusion

We presented a machine learning-based approach to question classification, modeled as a supervised learning classification problem. In order to train the learning algorithm, we developed a rich set of lexical, syntactic, and semantic features, among which are the *question headword* and hypernym, , which we deemed as crucial for accurate question classification. We then proceeded with a series of experiments to determine the most discriminative set of features, which proved to be the combination of *unigrams*,Q.category,word shapes, *question headword*, and the *semantic headword* feature. Using an SVM trained on these features, we attained 96.2% and 91.1% accuracy for coarse- and fine-grained classification, respectively, which, as we write, outperforms every other state-of-the-art result reported in the literature. Furthermore, we also suggested how these results could be improved, by using a better training and test set, and extended question type taxonomy.

REFERENCES

1. William Tunstall-Pedoe. True knowledge: Open-domain question answering using structured nowledge and inference. *AI Magazine*,31(3):80–92, 2010.
2. David Ferrucci, Eric Brown, Jennifer Chu-Carroll, James Fan, David Gondek, Aditya A. Kalyanpur, Adam Lally, J. William Murdock, Eric Nyberg,John Prager, Nico Schlaefer, and ChrisWelty. BuildingWatson:An overview of the DeepQA project. *AI Magazine*,31(3):59–79, 2010.
3. Ulrich Furbach, Ingo Gl¨ockner, and Bj¨orn Pelzer. An application of automated reasoning in natural language question answering. *AI Communications*,23(2-3):241–265, 2010. PAAR Special Issue.
4. Ingo Gl¨ockner and Bj¨orn Pelzer. The LogAnswer project at CLEF 2009. In *Working Notes for the CLEF 2009 Workshop*, Corfu, Greece, September 2009.

5. Demner-Fushman, Dina, "*Complex Question AnsweringBased on Semantic Domain Model of Clinical Medicine*",OCLC's Experimental Thesis Catalog, College Park, Md.:University of Maryland (United States), 2006.
6. Mohammad Reza Kangavari, Samira Ghandchi, Manak Golpour," Information Retrieval : Improving Question Answering Systems by Query Reformulation and Answer Validation" World Academy of Science, Engineering and Technology 48 2008.
7. Chiyoung Seoa,, Sang-Won Leeb, Hyoung-Joo Kima" An efficient inverted index technique for XML documents using RDBMS" Information and Software Technology 45 (2003) 11–22.
8. Paloma Moreda , Hector Llorens, Estela Saquete, Manuel Palomar" Combining semantic information in question answering systems" Information Processing and Management 47 (2011) 870–885.
9. Liang Yunjuan , Ma Lijuan, Zhang Lijun, Miao Qinglin" Research and Application of Information Retrieval Techniques in Intelligent Question Answering System" 978-1-61284-840-2/11/$26.00 ©2011 IEEE.
10. Li Peng, Teng Wen-Da, Zheng Wei, Zhang Kai-Hui "Formalized Answer Extraction Technology Based on Pattern Learning", IFOST 2010 Proceedings.
11. Figueira, H. Martins, A. Mendes, A. Mendes, P. Pinto, C. Vidal, D,Priberam's "Question Answering System in a Cross- Language Environment",LECTURE NOTES IN COMPUTER SCIENCE, Volume 4730, 2007,PP. 300-309.
12. Dan Moldovan, Sanda Harabagiu, Marius Pasca, Roxana Girgu, " The Structure and Performance of an Open-domain Question Answering System", Proceedings of the 38th Annual Meeting on Association for Computational Linguistics Hon Kong, 2000, PP. 563-570.
13. Li, X., & Roth, D. (2002). Learning question classifiers. In *Proceedings of the 19th international conference on computational linguistics* (pp. 1–7). Morristown, NJ, USA: Association for Computational Linguistics.
14. Chang, C.-C., & Lin, C.-J. (2001). *LIBSVM: a library for support vector machines.* (Software available at http://www.csie.ntu.edu.tw/˜cjlin/libsvm).
15. 15 .Alias-i. (2008). *LingPipe 3.8.1.* (Software available at http://alias-i.com/lingpipe).
16. Hsu, C., & Lin, C. (2001). *A comparison on methods for multi-class support vector machines* (Tech. Rep.). Department of Computer Science and Information Engineering, National Taiwan University, Taipei,Taiwan.
17. Rifkin, R., & Klautau, A. (2004). In defense of one-vs-all classification. *J. Mach. Learn. Res., 5*, 101–141.
18. Zhang and W. S. Lee. 2003. Question classification using support vector machines. In Proceedings of the26th Annual International ACM SIGIR Conference on Research and Development in Informaion Retrieval,pages 26–32.
19. Phil Blunsom, Krystle Kocik, and James R. Curran. Question classification with loglinear models. In *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '06, pages 615–616, New York, NY, USA, 2006. ACM.
20. Zhiheng Huang, Marcus Thint, and Zengchang Qin. Question classification using head words and their hypernyms. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, (EMNLP '08), pages 927–936, 2008.
21. Slav Petrov and Dan Klein. Improved inference for unlexicalized parsing. In *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics, Proceedings of the Main Conference*, pages 404–411, 2007.
22. Babak Loni, Gijs van Tulder, Pascal Wiggers, Marco Loog, and David Tax. Question classification with weighted combination of lexical, syntactical and semantic features. In *Proceedings of the 15th international conference of Text, Dialog and Speech*, 2011.
23. Fellbaum, C. (Ed.). (1998). *WordNet: An electronic lexical database*. MIT Press
24. Porter, M. F. (1980). An algorithm for suffix stripping. *Program*, *14*(3), 130–137.
25. Amaral, C., Cassan, A., Figueira, H., Martins, A., Mendes, A., Mendes, P., et al. (2008).

Authors

Mrs. Megha Mishra received her M.E degree from CSVTU University, India. This author has been published so many paper in various reputed journal. She is a research scholar in SOA University and life member of Indian society of Technical Education. She is having overall teaching experience of 9 years including professional colleges. His major research Interests are in Fuzzy logic, Expert systems, Artificial Intelligence, Data Mining and Computer-Engineering. She is currently working as an Associate Professor in Rungta groups of colleges Short Biography

Mr. Vishnu Mishra received his M.Tech from KIIT University, India in 2006. This author has been published so many paper in various reputed journal. A life member of Indian society of Technical Education. He is currently working as an Associate Professor in Rungta groups of colleges. He is having overall teaching experience of 10 years including professional colleges. His major research Interests are in Fuzzy logic, Expert systems, Artificial Intelligence and Computer-Engineering.