Question Classification using Naive Bayes Classifier and Creating Missing Classes using Semantic Similarity in Question Answering System

Jeena Mathew[1], Shine N Das[2]

[1]M.tech Scholar, [2] Associate Professor

[1,2] College Of Engineering, Munnar, Kerala, India

***Abstract*— Question Classification is the core component of the Question Answering System. The quality of the question answering system depends on the results of the question classification. Almost all the question classification algorithms are based on the classes defined by Li and Roth [2].In this paper, a question classification algorithm based on Naïve Bayes Classifier and question semantic similarity is proposed. This paper mainly focuses on *Numeric* and *Location* type questions. Naive Bayes Classifier is adopted to classify the questions into *Numeric* and *Location* classes and semantic similarity is used to classify the questions into their fine-grained classes. According to Li and Roth, the coarse grained class *Numeric* and *Location* has fine-grained class *Other*. In this paper, we also present the method to replace the *Other* class in *Numeric* and *Location* classes by creating new classes and adding the newly created classes in the hierarchy.**

***Keywords*— Naïve Bayes Classifier, Natural Language Processing, Question Answering, Question Class Hierarchy, Question Classification, Semantic Similarity.**

I. INTRODUCTION

The Internet or the World Wide Web is surely a tremendous and surprising addition in our lives. The internet can be known as a form of global meeting place where people from all parts of the world come together. In other words, people use it as a medium to link with other people, sharing files, amusement, data and lots of other actions that are effective and good in many terms.

The amount of data on the web increases tenfold every five years. Increase in data on the web has got many troubles and challenges for information retrieval. Information has gone from scarce to superabundant. That brings huge new benefits, but also big headaches, says Kenneth Cukier.It is obvious that existing search engines have many truly remarkable potentialities. But there is a very important capability which they do not have-deduction capability-the capability to synthesize an answer to question by drawing on bodies of information which reside in various parts of the knowledge base [5]. Millions of users search over the internet to find the answers to their questions. The current search engine retrieves a list of documents in response to a user's query and the user has to navigate through each and every document to get the exact answer. To solve this information overloading problem Question answering system came into play. A Question Answering system gives an exact answer to the questions. For the question" county did Ravi Shastri play for?"[2], the QA system provides "Glamorgan" as the exact answer, whereas the traditional search engine retrieves a list of documents in response to the user's question.

Most systems treat question answering as three different distinct sub-tasks: question processing, document processing, and answer processing [4].Question classification is one part of the question processing stage. During this phase, expected answer type is derived.

Example 1. *What year did the Titanic sink?* [2] The answer sentence obtained with the help of search engine is "RMS titanic was a British passenger liner that sank in the North Atlantic Ocean in the early morning of 15 April 1912 after colliding with an iceberg during her maiden voyage from Southampton, UK to New York City, US".Suppose the question is classified as Numeric: Year by some classification mechanism. It will help to locate the year value from the given answer sentences.

Example 2. Consider another question *"Which country gave New York the Statue of Liberty?"*[2] The answer sentence obtained from the search engine is "The Statue of Liberty, a gift of friendship from the people of France to the people of the United States, is dedicated in New York Harbor by President Grover Cleveland".If this question is classified as Location: Country, it means that only country type will be targeted from the text. This means that the question when correctly classified will give a clue about the answer which helps the system in guessing and extracting the answer from the text chunk. It is found that filtering out a wide range of candidates based on some categorization of answer types supports question answering system.

In this paper, a question classification algorithm based on Naïve Bayes Classifier and question semantic similarity is proposed. This paper mainly focuses on *Numeric* and *Location* type questions. Naive Bayes Classifier is adopted to classify the questions into *Numeric* and *Location* classes and semantic similarity is used to classify the questions into their fine-grained classes.

The rest of this paper is organized as follows. In the section II, we begin with a review on related works. Section III is about Naïve Bayes Classifier. Section IV about the semantic similarity measure. Problem statement is described in section V.Our proposed method is described in section VI.The experimental result for question classification is described in section VII. Conclusion is described in section VIII.

II. RELATED WORK

Question Classification is the most important phase of a QA System. The original method for question classification is primarily rule-based approach. These rules are very effective for particular question taxonomy. But the problem is that, large human effort is needed to create these rules. Some other systems employed machine learning approaches to classify questions.

X. Li and D.Roth [2] presented a machine learning approach to question classification. They 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. Their experimental results prove that the question classification problem can be solved quite accurately using a learning approach, and exhibit the benefits of features based on semantic analysis.

X. Li and D.Roth [3] presented the first work on a machine learning approach to question classification. Guided by a layered semantic hierarchy of answer types, they developed a hierarchical classifier that classifies questions into fine-grained classes. This work also performed a systematic study of the use of semantic information sources in natural language classification tasks. It showed 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.

M.Bakhtyar and A.Kawtrakul [6] proposed a new hierarchy for the questions that earlier belonged to the class *Location: Other* or *Entity: Other*. Classifying the questions into “Other” is not very useful for the answer extraction phase. These two classes are now represented as a hierarchy which is populated using some NLP techniques and knowledge resources i.e. WordNet and DBPedia. They also analysed how the new hierarchy helped to prune out the extra unnecessary details for efficient answer extraction. They focused on the question with a specific pattern for generating the new hierarchy using knowledge resources and presented an automatic hierarchy creation method to add new class nodes using the knowledge resources and shallow language processing. They also showed how language processing and knowledge resources are important in the question processing and its advantage on Answer Extraction phase.

Jinzhong Xu and Yanan Zhou [8] proposed a question classification algorithm based on SVM and question semantic similarity .It is applied in a real-world on-line interactive question answering system in tourism domain. In the two level question classification method, Support Vector Machine model is adopted to train a classifier on coarse categories; question semantic similarity model is used to classify the question into sub-categories. The use of concept of domain terms construction will improve the feature expression of Support Vector Machine and question semantic similarity. The experimental result show that the accuracy of the classification algorithm is up to 91.49%.

M.Bakhtyar and A. Kawtrakul [7] proposed a new hierarchy for the questions that earlier belonged to the class *Numeric: Other*. Almost all the previous question classification algorithms evaluated their work by using the classes defined by Li and Roth [1]. The coarse grained class Numeric has fine grained class *Other*. In this paper, we target and present the mechanism to create new classes to replace the *Other* class in Numeric class. We present an automatic hierarchy creation method to add new class nodes using the knowledge resources and shallow language processing.

III. NAIVE BAYES CLASSIFIER

The Naïve Bayes Classifier technique is based on the so-called Bayesian theorem and is particularly fitted when the dimensionality of the inputs is high. Naive Bayes can outperform more sophisticates classification methods. It comes handy since it can be trained rapidly. In Naïve Bayes, the concept of ‘probability’ is used to classify new entities. Here we are using Naïve Bayes Classifier with weka. Weka provides implementation of wide range of machine learning based classifiers. A trained classifier can be used for the classification of data in a particular domain which depends on the training set.

To train a classifier we need a training set. Here, before developing the training set we build a feature vector. All features are put in feature vector. Then we create an empty training set and give its initial capacity as 10.If required we can double the capacity of the training set. The next step is to make message into instance and add instance to training data. Thus a training set is created. Finally, choose the Naïve Bayes Classifier and create the model. Thus we create and trained a classifier.

IV. SEMANTIC SIMILARITY MEASURE

Semantic similarity is a measure of informativeness.It is computed based on the properties of the concepts and their relationships. Semantic similarity has been a part of computational linguistics and artificial intelligence for many years. Many semantic similarity measures have been developed in the past years. In general, all measures can be classified into two classes. The first one makes use of a large corpus to figure out the semantic similarity. The second one makes use of the relations and the hierarchy of a synonym finder such as Word Net. Here we are finding the semantic similarity of words using WordNet.WordNet is a freely usable software package. It provides six measures of similarity. Three similarity measures are based on path lengths between concepts. The remaining similarity measures are based on information content. Information content is based on the specificity of a concept. Here we are using the Lin similarity measure to find the semantic similarity between two words. Lin is one of the six similarity measures based on information content. It uses the amount of data required to fully depict two terms as well as the commonality between the two concepts.

V. PROBLEM DEFINITION

Question Classification is the core component of the Question Answering System. The quality of the question answering system depends on the results of the question classification. Almost all the question classification algorithms are based on the classes defined by Li and Roth [2] (shown in Table I).

According to Li and Roth, the coarse grained class ENTY, LOC and NUM has fine grained class *Other*. The problem with the fine grained class *Other* is that, it will not help in answer extraction process. It does not give any correct meaning regarding the expected answer type. For example, *what hemisphere is the Philippines in?* [2] is previously mapped to *LOC: Other*. This assigned answer never gives a clue or helps to extract the answer. Instead mapping it to *LOC: City: hemisphere* makes it more meaningful and helpful in extracting the answer.

Creating new classes manually for each and every possible question is impossible. To overcome that more general method to create and assign new classes to the questions is required. Our technique is based on the natural language processing and external knowledge resources.

TABLE I
COARSE AND FINE GRAINED CLASSES

Coarse	Fine
ABBR	abbreviation, expansion
DESC	definition,description,manner, reason
ENTY	animal, body, color, creation, currency,disease/medical,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, period, speed,temperature, size, weight

VI. PROPOSED TECHNIQUE

In this paper, we propose a new method to classify the questions into *Numeric* and *Location* classes. We also present our methodology for making the hierarchal structure to symbolize the classes and the mechanism to add new classes in the hierarchy for the *Numeric* and *Location* classes. Fig.1 shows the architecture of the proposed system.

A. Target Question

For our experiment we are using a limited set of questions from UIUC [2] dataset.

B. Classifier

A supervised learning system that does classification is known as a learner or, a classifier. A training data is first fed into the classifier in which each item is already labeled with the correct label or class. The learning algorithm is trained using this data. It creates models that can then be used to label/classify similar data. Here we are using Naïve Bayes classifier.

1) Naïve Bayes Classifier: The question is given as the input to the classifier and it act as the message to be classified. The classifier classifies it into *Numeric* or *Location* classes.

2) Location and Numeric Class Hierarchy: The commonly used question category criteria is a two level class hierarchy proposed by Li and Roth [2].This hierarchy contains 6 coarse classes and 50 fine classes. In this paper, we focus on the coarse-grained classes *Numeric* and *Location* and their fine-grained classes. The *Numeric* class hierarchy is shown in Fig.2.

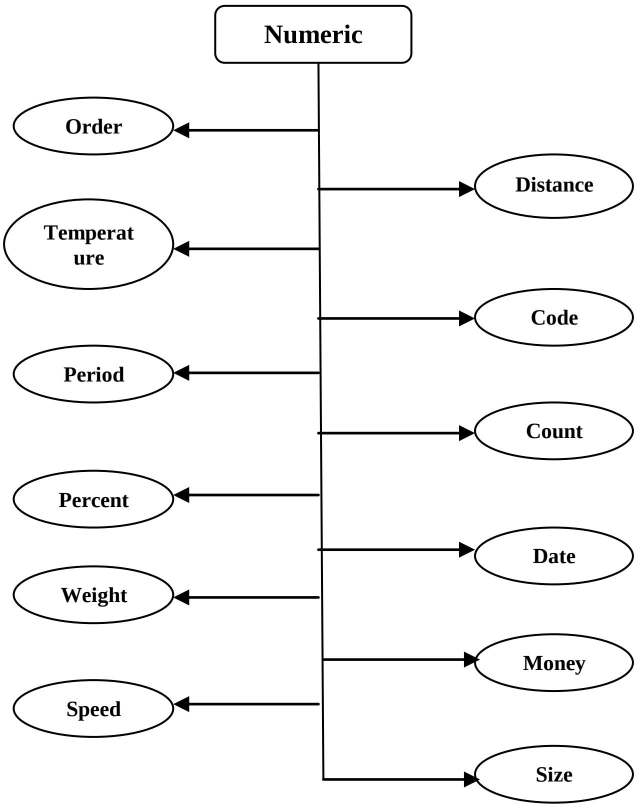


Fig.2 Numeric Class Hierarchy

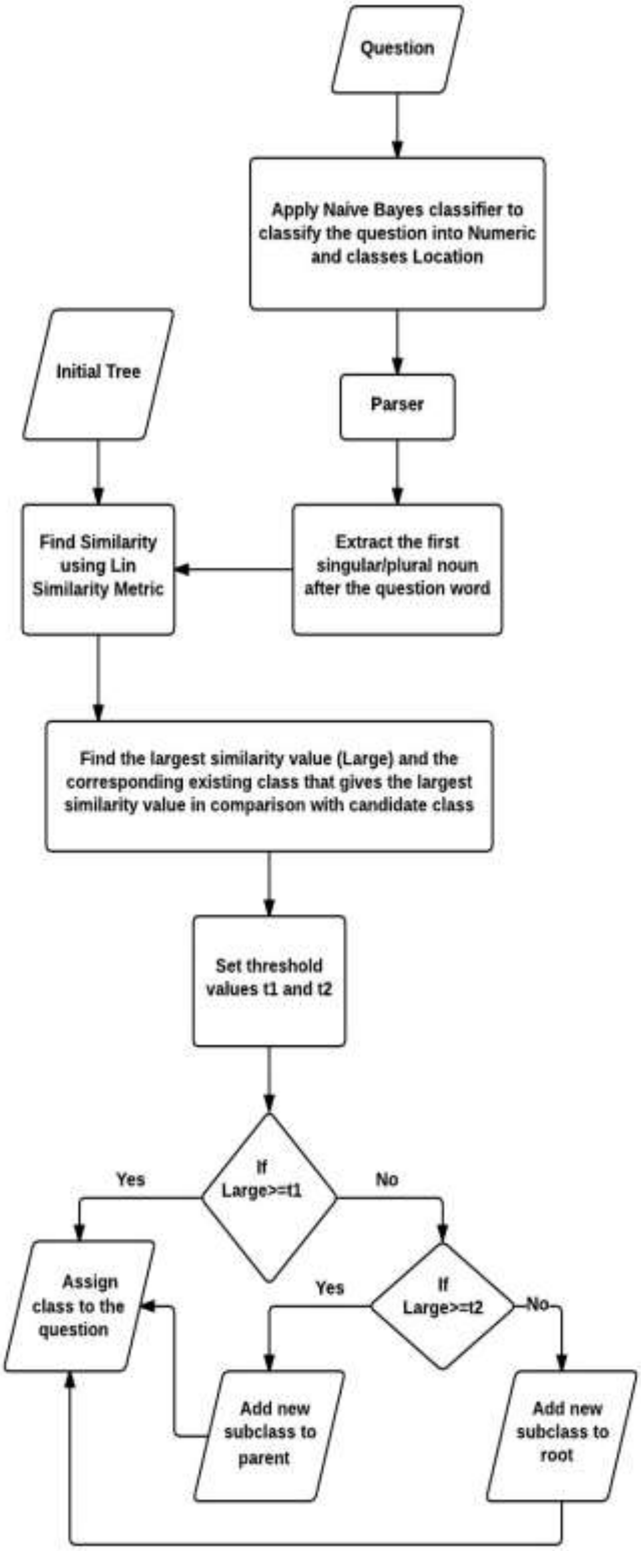

Question
Apply Naive Bayes classifier to classify the question into Numeric and classes Location
Initial Tree
Parser
Find Similarity using Lin Similarity Metric
Extract the first singular/plural noun after the question word
Find the largest similarity value (Large) and the corresponding existing class that gives the largest similarity value in comparison with candidate class
Set threshold values t1 and t2
If Large>=t1
Yes
No
Assign class to the question
Yes
If Large>=t2
No
Add new subclass to parent
Add new subclass to root

Fig.1 Architecture of the Proposed System

In Figure 2, *Numeric* is the base class and has 12 subclasses. If the question does not match with any of the existing subclasses it is assigned to the subclass *Other*. The *Location* class hierarchy is shown in Fig 3.

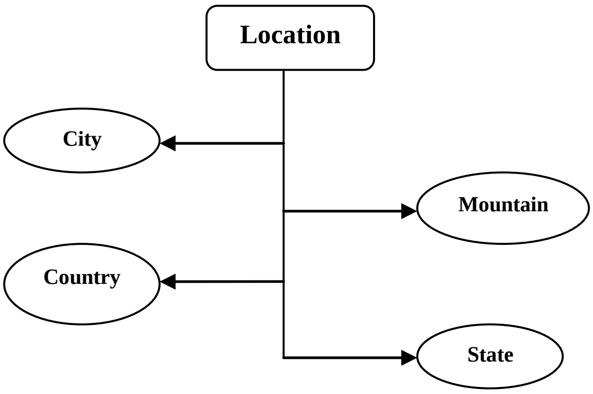


Fig.3 Location Class Hierarchy

In Figure 3, *Location* is the base class and has 4 subclasses. If the question does not match with any of the existing subclasses it is assigned to the subclass *Other*.

C. Parser

After the question is given as the input, the next step is to tag the words in the question. For that we are using Stanford Parser. It is a natural language parser that figures out the grammatical relation of sentences such as which words is the subject or object of a verb or which groups of words go together. In our case, Maxent tagger of Stanford Parser allows as to find the part-of-speech tag of each word of the question. That is, for each word, the tagger gets whether it is a verb, a noun etc and assigns the result to the word.

D. Extract Noun

After each word of the question is assigned the part-of-speech tag, the next job is to find out the first singular noun (NN) or plural noun (NNS) after the question word. For example, *"What is the temperature at the center of the earth?" or "what is the population of India?"*[2]The first singular noun or plural noun after the question word for the above example is *temperature* and *population.* These are the main focus of the question. It also acts as the candidate class to be added as a node in the hierarchy.

E. Similarity Measurement using WordNet

On finding the candidate class, the next step is to add the resulted candidate class in the hierarchy. Candidate class cannot be directly added into the hierarchy. This can be done by adding every candidate class in the hierarchy thus making the hierarchy grow very quickly. To avoid this we consider the relationship between the existing classes and the resulted candidate class.

As a first step, we calculate the similarity between the existing classes and the candidate class. For calculating the similarity we are using the Lin similarity measure provided by WordNet [1].In the previous paper, wu and palmer similarity metric provided by WordNet [1] is used. The similarity value provided by wu and palmer metric is not accurate. To overcome this disadvantage we are using the Lin similarity measure. Lin similarity measure is based on information content. It uses the amount of data required to fully depict two terms as well as the commonality between the two concepts to find the similarity value.

After calculating the similarity, we find out the largest similarity value out of all similarity values. And also we find out the corresponding existing and candidate class that gives the largest similarity value.

After finding out the largest similarity value, we compare it with two threshold values t1 and t2, t1 is used to classify the questions using existing classes and t2 is used to add the candidate class as the subclass of existing classes and also t1 is always greater than t2.Firstly the similarity value is compared with t1. If the similarity is greater than t1, the existing class that gives the largest similarity in comparison with the candidate class is assigned to the question. If the similarity is less than the t1, then the largest similarity value is compared with the t2 value. If the similarity is greater than t2 then candidate class is added as the subclass of an existing class that gives the largest similarity in comparison with the candidate class and the candidate class is assigned to the question. Otherwise the candidate class is added as the child node of the base class and also the candidate class is assigned to the question. The proposed algorithm is shown below.

Proposed Algorithm

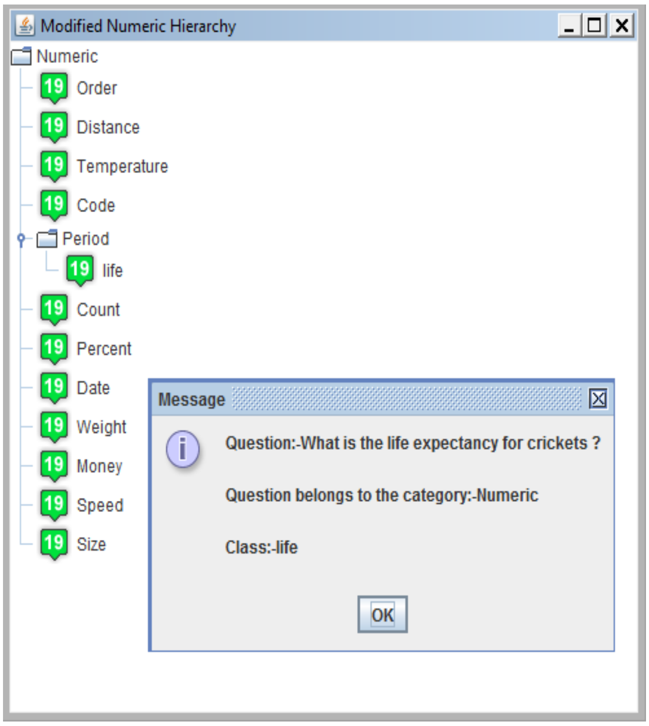
```
Require: A natural language question Q
Require: Threshold values t1 and t2
    candidate: = First noun after the question word
    root: = root of the tree
    n: = Number of tree nodes
    for i=1 to n do
        similarity: =Sim (node[i],candidate) using Lin metric
        largest: = Largest (similarity)
        if largest >=t1 then
            AssignClass (Q, node[i])
        else if largest>=t2 then
            InsertChildToParent (candidate, node[i])
            AssignClass (Q, candidate)
        else
            InsertChildToParent (candidate, root)
            AssignClass (Q, candidate)
        end if
        end if
    end for
```

VII. EXPERIMENTAL RESULTS

In this experiment, the corpus contains the training set of 250 questions, and test set of 150 questions. We have developed a user interface in which the test set questions are applied one by one through this interface. Each question is tokenized and POS of words are tagged, then the features of the questions are extracted by the method in the paper.

We adopted Naïve Bayes Classifier to classify the questions into *NUM* and *LOC* coarse classes and semantic similarity to obtain their fine-grained classes. We use 250 questions to train the classifier.

In the first step, the question is given as the input to the classifier and the classifier classifies it into *NUM* and *LOC* classes. In the second step, we obtain the main focus of the question and calculate the similarity value based on our Proposed Algorithm. In third step, we populate the hierarchy and assign the classes to the questions. The resulting hierarchy for the question *what is the life expectancy for crickets?* [2] shown in Fig.4.

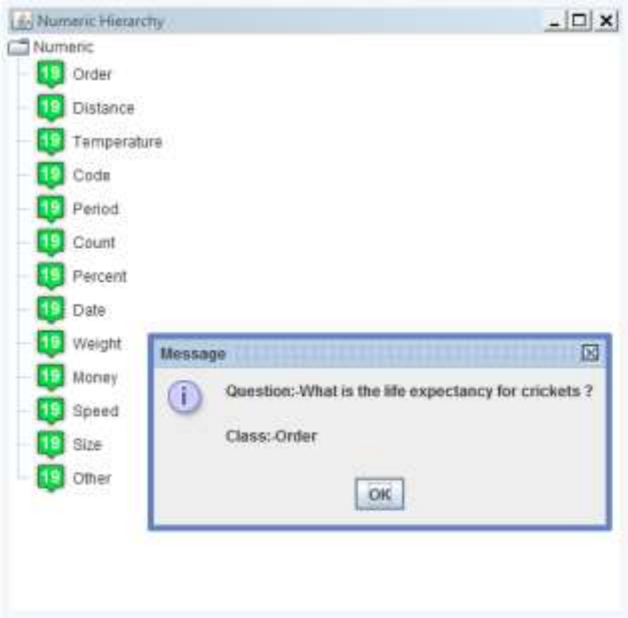


Question:-What is the life expectancy for crickets ?

Question belongs to the category:-Numeric

Class:-life

Fig.4 Proposed Numeric Hierarchy



Question:-What is the life expectancy for crickets ?

Class:-Order

Fig.5 Existing Numeric Hierarchy

Compared with the existing system our proposed system gives more accurate results. From figure 4 and figure 5 it is clear that for the question *what is the life expectancy for crickets?* [2] the class *NUM: Period: life* helps to extract the correct answer from the text chunks than *NUM: Order.*

VIII.CONCLUSION

We propose a new question classification mechanism based on Naïve Bayes Classifier and Semantic Similarity for the questions that belongs to the class *NUMERIC* and *LOCATION*. We showed that replacing fine grained class "*Other*" is helpful in extracting the exact answer. Also we add the newly created class that replaces the "*Other*" class in the hierarchy.

In the future work, we can implement a method that combines both accuracy and time consumption in getting the exact answer.

REFERENCES

[1] Z. Wu and M. Palmer, "Verbs semantics and lexical selection," in Proceedings of the 32nd annual meeting on Association for Computational Linguistics, ser. ACL '94. Stroudsburg, PA, USA: Association for Computational Linguistics, 1994, pp. 133–138. [Online]. Available: http://dx.doi.org/10.3115/981732.981751

[2] X. Li and D. Roth, "Learning question classifiers," in *Proceedings of the 19th international conference on Computational linguistics*. Morristown, NJ, USA: Association for Computational Linguistics, 2002, pp. 1–7.

[3] X. Li and D. Roth, "Learning question classifiers: the role of semantic information," Natural Language Engineering, vol. 12, no. 03, pp. 229–249,2006.[Online].Available:http://dx.doi.org/10.1017/S135132490500 3955

[4] L.A.Zadeh, "From search engines to question answering systems—The problems of world knowledge, relevance, deduction and precisiation." *Capturing Intelligence* 1 (2006): 163-210.

[5] H.Sundblad, "Question Classification in Question Answering Systems." (2007).

[6] M.Bakhtyar and A.Kawtrakul, "Integrating knowledge resources and shallow language processing for question classification," in *Proceedings of the KRAQ11 workshop*. Chiang Mai: Asian Federation of Natural Language Processing, November 2011, pp. 22–28. [Online]. Available: http://www.aclweb.org/anthology/W11-3104

[7] M.Bakhtyar et al. "Creating missing classes automatically to improve question classification in question answering systems." *Digital Information Management (ICDIM), 2012 Seventh International Conference on*. IEEE, 2012.

[8] Xu.Jinzhong, Y.Zhou, and Y.Wang. "A classification of questions using SVM and semantic similarity analysis." *Internet Computing for Science and Engineering (ICICSE), 2012 Sixth International Conference on*. IEEE, 2012.