Creating Multi-level Class Hierarchy for Question Classification with NP analysis and WordNet

Maheen Bakhtyar[1,2], Asanee Kawtrakul[3,4], Junaid Baber[1,2], Sher Muhammad Doudpota[5]
[1]Department of Computer Science and Information Management
Asian Institute of Technology
Thailand
[2]Department of Computer Science and IT
University of Balochistan
Pakistan
[3]Department of Computer Engineering
Kasetsart University
Thailand
[4]National Electronics and Computer Technology Center
Thailand
[5]Department of Computer Science
Sukkur Institute of Business Administration
Pakistan
Maheen.Bakhtyar@ait.asia, asanee_naist@yahoo.com, Junaid.J.Baber@ait.asia, sher@iba-suk.edu.pk

Journal of Digital Information Management

ABSTRACT: *Question answering systems provide answer to the questions using various question processing techniques and extracts answers from a set of documents. Finding the exact answer is more interesting and useful than getting a list of documents to look through and find the answer manually. Question answering is not same as a traditional document retrieval search engine where a set of relevant documents are returned in response of the query, whereas, in the question answering systems the response of the query is a concise and exact answer to the question. Typically, Question Classification (QC) is the first step in a Question Answering (QA) system. This phase is responsible for finding out the type of the expected answer by pruning out the extra information that is not relevant to extract the answer. Almost all the previous QC algorithms evaluated their work on the basis of a common class hierarchy already defined. The coarse grained classes Location, Entity and Numeric in the existing hierarchy have a fine grained class Other. We present the framework to create new fine grained classes to replace the Other classes. We also discuss the motivation behind the replacement and how the new fine grained classes may support the answer extraction. Additionally, we also present an automatic hierarchy creation method to add new class nodes using WordNet and Noun phrase parsing.*

Categories and Subject Descriptors:
I.2.7 [Natural Language Processing];Text analysis F.4.2 [**Grammars and Other Rewriting Systems]**

General Terms:
Natural Language Processing, Query Systems

Keywords: Question Classification, Question Answering, Question Class Hierarchy, Meta Information Management, Natural Language Processing

Received: 11 July 2012, Revised 29 September 2012, Accepted 7 October 2012

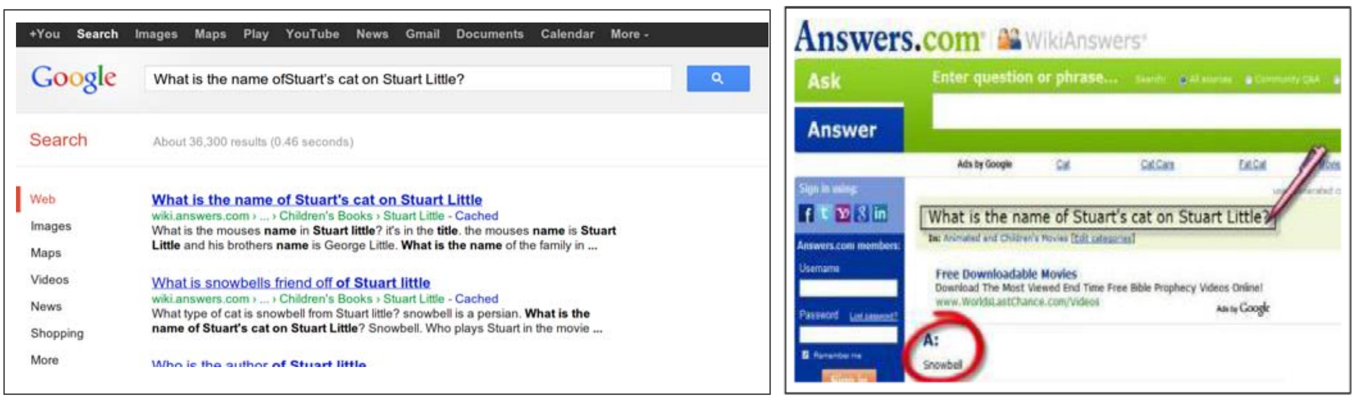
1. Introduction

Exact answer to the question is more interesting and useful than getting a list of documents in response of a query. People are usually more interested in the exact answer and do not desire to look for the answer in a long list of documents. The documents retrieved from traditional search engines are large in number (as shown in Figure 1 (a)), and users have to manually browse the documents to find the answers.

Open domain Question Answering (QA) ([6, 10]) serves the purpose of obtaining the answers in response of a question.

For example, a traditional search engine[1] retrieves the set of documents in response of the question "*What is the name of Stuart's cat on Stuart Little*?" as shown in Figure 1 (a). Whereas, QA systems such as Answers.com [1], provides the exact answer; "*Snowbell*" as shown in Figure 1 (b).

Generally, QA systems comprise of three phases, (1) Question Classification (QC), (2) Query Expansion (QE), and Answer Extraction (AE). The QC is typically the first step towards developing a QA system, this phase processes the question to find the expected answer type. The following examples[2] show the importance of this phase with respect to the AE.



(a) (b)

Figure 1. Example applications of document retrieval and Question Answering systems (QA), (a) shows the document retrieval system where 36300 documents/pages are retrieved on query question, and (b) shows the QA system that returns the exact answer of query question.

Example 1: *Who was the first American to walk in space?*. The answer sentence obtained is "*In 1965 astronaut Edward White became the first American to "walk " in space during the flight of Gemini* 4"[3]. Suppose the question is classified as *Human: Individual* by some classification mechanism. We notice that the answer line contains the matching string "*first American to walk in space*" therefore, the answer to the question is to be selected from the remaining part "1965", "*Edward White*" or "*Gemini* 4". Correct classification now leads us to the answer *Edward White*.

Example 2: *What day and month did John Lennon die*?. If this question is classified as *Number: Date*, it means that only date type will be targeted from the text. This implies that the question when correctly classified will give a hint about the answer which helps the system in judging and extracting the answer from the corpus. Research shows ([7, 13, 17]) that filtering out a wide range of candidates based on some categorization of answer types supports question answering system. It also guides the answer extraction mechanism to decide the strategy to extract the answer correctly.

The questions can be categorized mainly in two ways, i.e., considering the question word and second the answer type. Ray et al. [15] categorizes the factoid questions in the categories such as "*who*", "*why*", "*what*", "*where*", "*how*" and "*when*" and classify them based on the two level hierarchy of classes defined by Li et al. [11] and shown in Table 1.

2. Problem Statement and Motivation

Question Classification is important and helpful for extracting the answers. A correct and meaningful classification will lead the system to more efficient and correct answer extraction mechanisms. On the other hand, a wrong or meaningless classification will not improve the answer extraction and might become a cause of inaccurate final results. We deal with the significance of the question classification for extracting correct answers in the earlier section. An incorrect classification will neither improves the answer extraction but might become a cause of incorrect answers.

Most of the classification algorithms make use of the two level classes hierarchy defined by Li et al. [11] (shown in Table 1).

We address few issues related to the need and advantage of creating new classes automatically in the hierarchy and filling the gap of some missing classes.

2.1 Insufficient classes in the taxonomy

Question classes defined and labelled in UIUC[4] dataset by [11] are most widely used in the previous work ([4, 5, 8, 14, 18, 22]). Many of the researchers developed their systems using these classes and labeled the questions in dataset.

In the labeled dataset, if a question is not mapped to some class, due to the class being uncommon, it is placed into the fine grained class *Other*. The class*Other*

[1]http://google.com
[2]Questions and answer sentence containing the answer is taken from TREC-10 [19]

[3]This line is taken from the document number DOCNO: AP890527-0145 and contains the answer to this question
[4]http://cogcomp.cs.illinois.edu/Data/QA/QC/

is not very helpful in the answer extraction because it does not give any concrete meaning regarding the expected answer type. For example, in case of *Location* category, *Location*: Other will only prune out *city, country, mountain* and *state* as possible answer categories. Therefore, a close analysis of questions belonging to this class is needed and a new set of classes is required to overcome this deficiency. There may be many more classes such as *continent, ocean, river, lake etc* that need to be classified to support the remaining phases of QA. For instance, the class *river, lake* or any other *water body* is not present in the existing class taxonomy whereas some questions require such classes e.g. the question *What body of water are the Canary Islands in ?* is currently placed in class LOC: *Other* [11]. This assigned class neither gives an exact hint nor helps to filter the candidate answers. Whereas, mapping it to a class such as *waterbody* makes it more meaningful and easier to find the answers.

One of the main advantage of replacing the class *Other* with fine grained classes is that it makes assignment of a single question to multiple classes/subclasses more efficient and effective. A question will implicitly belong to all its super-classes in the hierarchy that helps answer ex tractions algorithms to be designed in a way that they can consider a superclass as another candidate if the leaf class is not sufficient.

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**

Table 1. Coarse and Fine grained classes

The coarse grained classes; *Location* and *Entity*; and all their fine grained classes; have already been discussed in our previous work [3]. In this paper, we examine the class *Numeric* and all its fine grained classes, including those that are missing and need to be added in the hierarchy and mapped to the question dataset. For example, the class "*population*", "*score*" or any other "*cardinal number*" are not addressed in the existing class taxonomy, whereas, some questions require these classes such as the questions "*What is the population of Mexico?*" and "*What is the frequency of vhf ?*" are currently mapped to the class *NUM:Other*, as discussed above, the class *NUM:Other* is not helpful to filter the candidate answers. Whereas, mapping those examples to some particular class such as *population* and *frequency*, respectively, makes them more meaningful and helpful to find the correct answers.

Similarly, the question "*Which is the largest island in Thailand ?*" is previously mapped to the class *LOC:Other*. The *LOC:Other* class does not help much to extract the answer from the given text chunk "*Phuket is now Thailand's most important tourist destination, offering a variety of beaches, attractions and exciting night life. Koh Phuket is Thailand's largest Island. It is 50 km long north to south and 21 km wide and joined to the mainland by Sarasin bridge. Phuket has been inhabited since the early days of mankind by ancient tribes and this still keeps archaeologists occupied to find out the history from the early days.*[5]". If the same question is mapped to the class *LOC:PHY:Natural:Island* or even if to the class *LOC:PHY:Natural*, it will help to locate the *natural locations* or *islands* inside Thailand from the given text chunk. Therefore, the detail of classes and subclasses are needed to cover more questions instead assigning them to class *Other*.

We show that the highest number of questions (among 500 questions in TREC 10[19]) are currently mapped to the class *LOC:Other* [3], which indicates that most of the questions will be answered during the AE phase without making use of the classes, despite the efforts put into classification phase. We also observed from the work of Li et al.[12] by examining 1000 questions from TREC 10 and 11 [20] that about 30% of the questions lie under the class *Numeric* and out of those about 8% are mapped to the class *Other*.

2.2 Unavailability of automatic class creation mechanism in the hierarchy

In the previous section we show that the new hierarchy of classes and subclasses is needed and effective for efficient answer extraction. Creating new classes manually for each and every possible question is impossible and we need an automatic mechanism to create and assign new classes. [11] presented a two level hierarchy with a fixed number of classes. Whereas a more general method to create and assign new classes to the questions is required. The new classes may be organized in any number of levels in the hierarchy and can be assigned accordingly.

Our target is to fill the gap of the unavailable classes for the course grained classes *Location, Entity and Numeric*. We propose a technique that automatically creates new classes and classify all the questions that are currently mapped to *Loc:Other,* and *Entity:Other* having the pattern "*what | which NP ...* ". Moreover, we deal with all the questions mapped to class *Num:Other*. Our technique is based on the language processing and external knowledge resources.

3. Methodology

In this section, we present our methodology for creating

[5]http://www.beachpatong.com/

the hierarchical structure to represent the classes, and the mechanism to automatically add new classes into the hierarchy. We divide the section into two parts. First part explains the approach for dealing with the classes mapped as *Num:Other* and in the second we discuss the methodology for class *Enty:Other* and *Loc:Other*.

3.1 Approach for Numeric Class

First we focus on and discuss the *Numeric* class and the questions that are mapped to class *Num:Other.*

3.1.1 Hierarchical Class Structure

We present an algorithm that creates hierarchical class taxonomy and places the existing classes into appropriate position in the tree, and add new required classes (to replace the class *Other*) that are missing in the previous taxonomy, as shown in Figure 2.

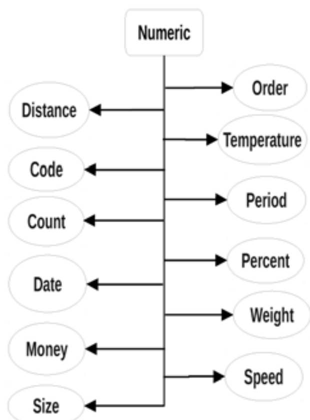


Figure 2. Numeric Class Hierarchy

3.1.2 Adding new classes in the hierarchy automatically

We examined the distribution of the questions over the existing class hierarchy. We used 5500 training questions available at UIUC and observed that about 17% of the questions belong to the class *Numeric*. We target the questions that are mapped to class *Num:Other* and observe that the highest number of questions start with the pattern "*What is | 's Noun Phrase* (*NP*)". The proportion of the questions belonging to the class *Numeric:Other* based on the starting pattern is shown in Table 2.

What is NP...	55%
What's NP ...	14%
What was NP ...	9%
What are NP ...	6%
What does ...	2%
What number ...	2%
What JJ number ...	2%
What amount ...	2%
How JJ...	6%

Table 2. Question proportion w.r.t the question starting patterns where NP is noun phrase and JJ is adjective tags by part-of-speech tagging

1. Consecutive singular/plural nouns inside the first Noun Phrase

Noun phrases have consecutive Singular Noun (NN) and/or Plural Noun (NNS) inside the first NP of the question. For example, "*What's a perfect score in a gymnastics exercise?*" or "*what is the population of ohio?* "

After the first NP is identified in the question, the next task is to determine the consecutive NN and NNS e.g. score and population in the examples above. These are the target focus of the question and also a candidate class to be added as a node in the hierarchy.

2. Adding a new class based on similarity calculation and knowledge resource

After finding the candidate class for the hierarchy, the next task is to arrange the candidate class in the hierarchy in the appropriate place. We cannot directly add the node in the hierarchy because adding each and every candidate class directly into the hierarchy will make the hierarchy grow very rapidly. We need to consider the relationship between the candidate class and the existing classes before adding a new node. Therefore, as a first step, we calculate the similarity between the new candidate and the existing nodes in the hierarchy. If the similarity value between candidate class and some existing class is greater than a threshold t then that existing class is assigned to the candidate class, otherwise a new node is added. The basic framework is shown in Figure 3.

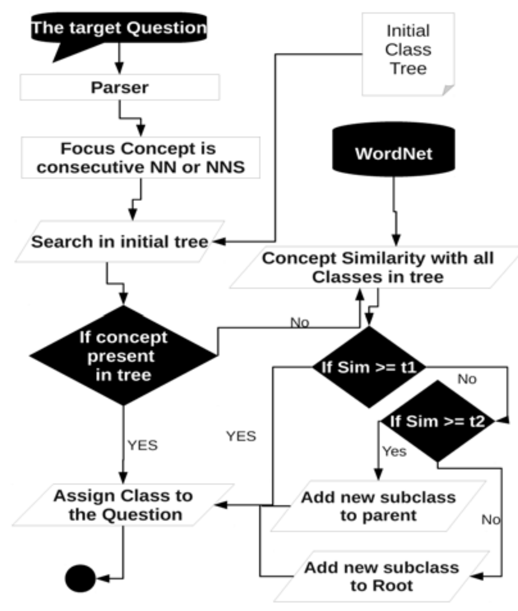


Figure 3. Numeric Classification Framework

There are various metrics and measures to find the similarity between pair of words, several based on WordNet ([9, 21, 16]). We use the similarity metric by Wu and Palmer [21] for calculating similarity between pair of words. After calculating the similarity value based on the WordNet, we compare it with the threshold values. We use two threshold values, t_1 is to classify the question using existing classes and t_2 is to add new node as a sub-class of some existing class, where $t_1 > t_2$.

If the similarity value is less than both of the thresholds, then the new node is created as a child of the root node. The basic algorithm is shown in Algorithm 1. In the algorithm, *AssignClass*(*Q, some_class*) classifies the question *Q* as some_class. *InsertChildToParent* (*some_child, some_parent*) creates the class *some_child* as a child of class *some_parent*.

Algorithm 1 Classification (Numeric)

Require: A natural language question Q
Require: Threshold values t_1 and t_2
NP := First Noun Phrase after the Question Word
candidate := Extracted consecutive Noun (singular/plural)
root := Root of the tree
n := Number of tree nodes
for $i = 1$ to n **do**
  similarity :=SimWN(*node*[*i*]; candidate) using WordNet
  if similarity $\geq t_1$ **then**
    AssignClass(Q, node[i])
    BREAK LOOP
  end if
  if similarity $\geq t_2$ **then**
    InsertChildToParent(candidate,[node[i]])
    AssignClass(Q, candidate)
    BREAK LOOP
  end if
  InsertChildToParent(candidate, root)
  AssignClass(Q, candidate)
  BREAK LOOP
end for

A high value of *similarity function* (*SimWN*) means the concepts are highly related. This value depends on the distance between one concept to another in the taxonomy such as WordNet in our case.

There is a relationship between the similarity values calculated and the size of the hierarchy. If similarity of concepts is low, it compels to add new nodes into the tree. This means the size of the tree will depend on the thresholds set for the addition of new nodes. If the threshold values are large, then tree size will increase because most of the new candidate classes will be added as new node. Therefore, it is required to maintain reasonable values for t_1 and t_2 to obtain a reasonable number of nodes in the tree. We have set the threshold values manually in our framework based on try and error, where $t_1 = 0.7$ and $t_2 = 0.5$. If the similarity of candidate class with any of the existing classes is greater than t_1, then the question is mapped to that existing class and no new node is added in the tree. Few examples below depict the different cases.

EXAMPLE 1. The question "What 's a perfect score in a gymnastics exercise?" has NP ***perfect score*** *and the NN is* ***score*** *which is a candidate new node in the hierarchy.*

EXAMPLE 2. The question "What is ***the population*** *of Ohio?" has NP the population and the consecutive NN is population which is a candidate new node in the hierarchy.*

EXAMPLE 3. The question "What is the ***normal resting heart rate*** *of a healthy adult?" has NP normal resting heart rate and the consecutive NN are* ***heart rate*** *which is a candidate new node in the hierarchy.*

EXAMPLE 4. The question "What amount of folic acid should an expectant mother take daily?" has NP and NN ***amount*** *which is a candidate new node in the hierarchy.*

EXAMPLE 5. The question "What was the death toll at the eruption of mount pinatubo?" has NP ***the death toll*** *and the consecutive NN are* ***death toll*** *which is a candidate new node in the hierarchy.*

Existing-New class	Sim WN	Decision
count-score	0.86	Assign Existing Class
count-population	0.80	Assign Existing Class
speed-heart_rate	0.86	Assign Existing Class
count-amount	0.69	Insert Child to Parent
amount-death_toll	0.49	Insert Child to Root

Table 3. Similarity calculation (Numeric)

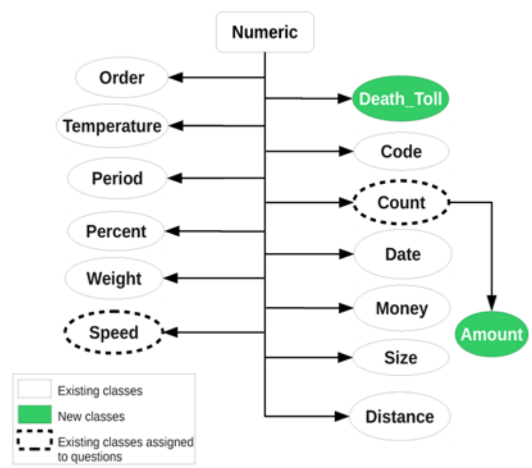


Figure 4. Example Hierarchy for *NUM:Other*

The hierarchy after implementing the algorithm for the examples above is shown in Figure 4.

3.2 Approach for "*Location*" and "*Entity*" class

In this section we discuss the methodology for *Loc:Other* and *Enty':Other* classes.

3.2.1 Classes in form of a hierarchy

We propose an algorithm that creates hierarchical class taxonomy by placing the existing classes into appropriate position in the tree, and add new classes that are missing in the previous taxonomy, as shown in Figure 5 (a). The Entity Class hierarchy is shown in Figure 5 (b).

3.2.2 Automatic class creation in the hierarchy

We examined the distribution of the target pattern questions over the existing class hierarchy. We used 1500 questions consisting of 1000 training questions available at UIUC, and 500 questions from TREC 10. We observed that 23% of the questions belong to class *Entity* and 16% belong to class *Location*. Out of these, 16% and 54% belong to classes *ENTY:Other* and *LOC:Other* respectively. We also observed that 30% of the *ENTY:Other* and 54% of the *LOC:Other* questions are of our target pattern "*what | which NP ...* ". It shows that the proportion of question matching this pattern seems adequate to get started. Therefore, we will focus on the class *ENTY:Other* and its subclasses as shown in Figure ??. The similar approach can be applied to the *LOC:Other* class with separate set of patterns e.g. the question "*which part of the university has most trees?*" is a *Location* question having no defined class in the initial hierarchy as well as in the newly created hierarchy shown in Figure 5 (a). To keep the initial work simple, we target the subset of question classes and question patterns for *Location* and *Entity* classes. Once we have developed a system to add new nodes for this set of questions, we can define similar algorithms for the other set of questions.

The questions starting with the *What* and *Which* question words, followed by a Noun Phrase (NP), have their expected answer type inside the NP. The expected answer type/question class will be the focus of the NP. The steps to create and assign new class is as follows:

1. Noun Phrase and Head Noun

Noun phrases have a head noun surrounded by some modifiers such as possessives, adjectives. For example, "*Which Thailand's island has highest number of tourists?*" or "*Which dark green plant is beneficial to fight cancer?*".

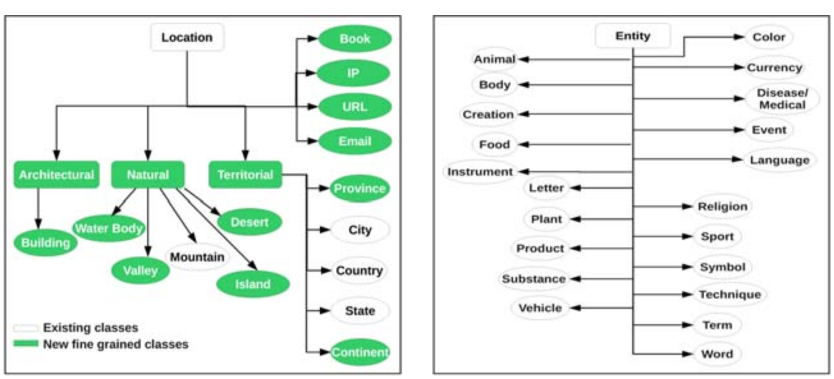
After the first NP is identified in the question, the next task is to determine the head noun e.g. island and plant in the examples above. Head noun is the target focus of the question and also a candidate class to be added as a node in the hierarchy. For example, in the question *what four forms does gold occur in?* has NP *four forms* and the head noun in this NP is the *forms* which is a candidate new node in the hierarchy. Similarly, the question "*which fungi cause the skin infection?*" has NP and head noun fungi and is a candidate for the class in the classes hierarchy.

After finding the focus of the NP i.e. the candidate class for the hierarchy, we cannot directly add the node in the hierarchy as discussed earlier. We need to consider the relationship between the candidate class and the existing classes before adding a new node and we use the similarity metric by Wu and Palmer [21]. Bakhtyar et al. [3]used DBpedia[6], a structured ontology representing a Wikipedia[7] page, for similarity calculation in addition to WordNet. We observe that because DBpedia is not complete and smaller in size with missing concepts, unlike WordNet, and might affect the overall similarity value in an unwanted manner (few cases observed in [2]). We also observed that WordNet gave adequate concepts' similarities therefore, we do not use DBPedia for our experiments. The basic framework is shown in Figure 6.

The basic algorithm is shown in Algorithm 2. In the algorithm, *AssignClass*(Q, *some_class*) classifies the question Q as *some_class*. *InsertChildToParent* (*some_child, some_parent*) creates the class *some_child* as a child of class *some_parent*.

Algorithm 2 Classification

Require: A natural language question Q
Require: Threshold values t_1 and t_2
NP := First Noun Phrase after the Question Word
candidate := Extracted Head Noun from NP
root := Root of the tree
n := Number of tree nodes
for i = 1 to n **do**
  similarity := Sim_{WN}(*node* [i]; candidate) using WordNet
  if similarity $\geq t_1$ **then**
    AssignClass(Q, node[i])
    BREAK LOOP
  end if
  if similarity $\geq t_2$ **then**
    InsertChildToParent(candidate, [node [i])
    AssignClass(Q, candidate)
    BREAK LOOP
  end if
  InsertChildToParent(candidate, root)
  AssignClass(Q, candidate)
  BREAK LOOP
end for



(*a*) (*b*)

Figure 5. Class Hierarchies (a) Location Class Hierarchy (b) Entity Class Hierarchy

2. Adding a class based on similarity calculation and knowledge resources

[6]http://dbpedia.org/About
[7]http://en.wikipedia.org/wiki/Main_Page

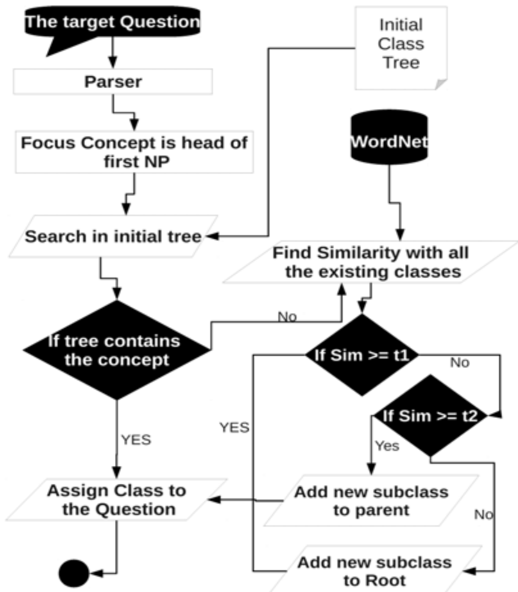


Figure 6. Entity Classification Framework

A high value of similarity function means the concepts are highly related. This value depends on the distance between one concept to another in WordNet. Following are the few examples of various scenarios:

EXAMPLE 6. *the question "what meter was invented by c.c. magee in* 1935 *?", the head noun is meter. Now, to decide whether this node should be added or not, we find the similarity with the nodes in the existing hierarchy. A high similarity is observed* (*see Table* 4), *therefore existing class is assigned to the question and no new node is added.*

EXAMPLE 7. *The question "what kind of science is cosmology ?" has the candidate class science*. Table 4 shows the similarity values for some concepts and Figure 9 shows the concepts in the hierarchy.

	Similarity WN	Decision
Instrument-meter	0.9	Assign Existing Class
Technique-Science	0.7	Insert Child to Parent

Table 4. Similarity calculation (Entity)

Figure 7 show the example how these nodes are added in the tree.

3.3 Noun Phrase Ambiguity

As we already discussed that question word followed by a NP has the question focus in head noun. This rule is applicable to almost all the questions from the corpus to determine the candidate class. But we do observe some patterns in human language that are sometimes ambiguous. Such ambiguous noun phrases in the questions can cause issues in the correct classification. As an example, consider the question "*which teacher´s bag is black ?* ". The noun phrase in this question is "*teacher´s bag*" but we are unsure if the target focus of the question is "*bag*" or "*a teacher*". It basically gives dual meaning. First meaning is to name the "*teacher*" who has

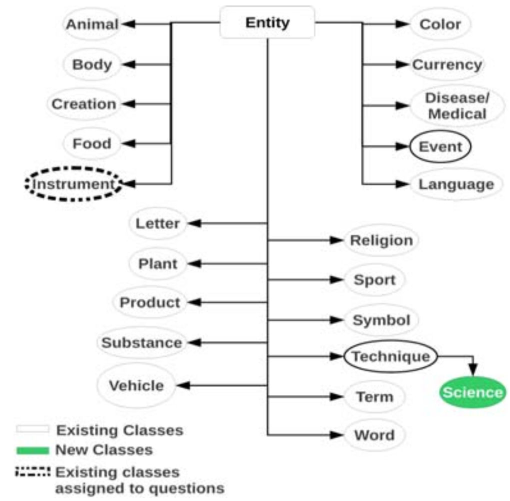


Figure 7. Adding new node

a black bag and in the second sense there are many bags and all of them belong to teachers and what is asked is that among those bags which one is black. In the first sense the candidate class might be a human whereas, in the second sense the candidate class might be an entity, bag or a product. Similarly, an other example is " which programmer's algorithm is based on recursion" where the meaning could be identify an algorithm or could be a person. We currently do not focus on dealing with this problem at the moment but we only discuss the problem and present a shallow and partial idea to deal with it.

One way to look at the problem is to make use of large document corpus and observe and predict the co-occurrences of both the meanings. For instance, in our example, system must extensively parse through the documents and based on the patterns and frequency of occurrences, should predict the correct meaning of the question. This technique can be more effective in domain specific systems such as medical, agriculture, architecture etc.

4. Experiments and Discussion

In this section we present the experiments, results and discussion

4.1 Numeric Class

We performed our experiments on the set of 50 questions from UIUC dataset. These 50 questions are previously mapped to the class *NUM:Other*. In first step, we obtain the focus (consecutive NN or NNS) of all the questions and calculate the similarity value based on the steps defined in Algorithm 1. In second step, we populate the hierarchy and assign the classes to the questions using our proposed algorithm. The resulting hierarchy is shown in Figure 8.

We compare our hierarchy with the coarse and fine grained classes defined by Li et al.[11] (shown in Table 1). Few example questions are examined below to show how the the new classes in the hierarchy can be more helpful to extract the answer.

The question "*What* **amount** *of folic acid should an expectant mother take daily?* " was mapped to the class *NUM:Other*, whereas, using our hierarchy, it is now mapped to the class *amount* which is a subclass of another existing class count. Suppose, some answer extraction module make use of amount instead of *NUM:Other*, it will skip extra information and will only look for some *amount* or some measurement in the text corpus to locate the correct answer. Similarly, the question "*What is the* **population** *of japan?*" is mapped to the class *population* making it more informative and useful for extracting the answer more efficiently.

The initial taxonomy for type *Numeric* contained 12 fine grained classes. After performing our experiments on the 50 questions, our hierarchy contain total of 19 classes. This means that 37% of the hierarchy consists of new classes. The 29% of the newly created classes are added as the sub-classes of some existing classes and the remaining are added as direct child of the root. In addition to this, 43% of the new classes are mapped to more than one question which shows that the newly created classes are reusable in the same way as the existing classes.

Our target is to develop a more informative class hierarchy. The hierarchy contains more informative fine grained classes which will help the answer extraction phase to locate the answer more precisely.

4.2 Entity Class

We performed the experiments on set of 50 questions of the pattern "*what | which NP...*" selected from UIUC dataset. To test the system, we first obtain the focus (head noun of NP) of the questions and then check the similarity based on the steps defined in Algorithm 1. Using the rules in the algorithm we populated the hierarchy and assign the classes to the questions. A visual chunk of the resulting tree is shown in Figure 9.

We compare the hierarchy built using our approach (Figure 9) with the course and fine grained classes (shown in Table 1) defined by Li et al. [11]. We examine some questions and show how the the new classes in the hierarchy can be more helpful to extract the answer.

The question "*what stringed weapon fires a bolt?*" is currently mapped to *ENTY:Other* whereas using our hierarchy it is mapped to the class *ENTY:Instrument*. If any answer extraction module uses our class instead of *ENTY:Other*, it will skip extra information and will only look for names or information about some instrument since weapon can be considered as an instrument. Another example is the question "*what kind of puzzle first appeared in the u.s. in the new york world on december* 21 , 1913 *?*" the new class "*puzzle*" is added as a new node and makes answer extraction convenient.

The initial taxonomy for type *Entity* contained 21 fine grained classes. Our hierarchy, after performing our this initial experiments increased this number to total of 31 classes which means that 32% of the hierarchy consists of the new classes. 60% of these newly created classes are added as the sub-classes of some existing class and the remaining are added as direct child of the root.

Our main focus was to develop a more informative class hierarchy. The hierarchy contains more informative fine grained classes that will help the answer extraction phase to locate the answer more precisely. We do not present a complete classification scheme but initially only for the specific pattern of questions as discussed earlier.

4.3 Further Discussion

Another point worth mentioning here is that we observed that though we targeted the classes that were mapped to "*ENTY: Other*" and "*LOC:other*" in the first part, however, the same strategy can be used for other questions of the same pattern i.e. questions starting with pattern "*What | Which NP ...*" regardless of ther existing type. We did not conduct complete set of expeirments but we show some examples in Table 5 and compare that how our approach can be used for the various questions that are already classified.

Answer extraction phase requires the question to be classified in some manner. If a classification mechanism is developed by using our set of classes, then answer extraction technique be more helpful to extract the answer.

5. Conclusion

We propose a new hierarchy for the questions that earlier belonging to the class Other . We show that classifying the questions into "Other" is not very useful for the answer extraction phase. We represent these classes in a hierarchy that is re now represented as a hierarchy which is populated using some NLP techniques and knowledge resources i.e. WordNet and DBPedia. We also analyzed how the new hierarchy helped to prune out the extra unnecessary details for efficient answer extraction.

This is the initial work carried out with extremely limited questions. We only focused on the question with a specific pattern for generating the new hierarchy using knowledge resources. We plan to work on the remaining question types and patterns in the future. Moreover, we also plan to target the other coarse classes, "*NUM* " having sub-type "*Other*". Additionally, we plan to label the questions and publish with the hierarchy obtained for all the questions set so a new set of classes is obtained and is comparable for the other researchers.

We propose a new hierarchy for the questions that earlier belonging to the class Numeric:Other. We show that classifying the questions into "Other" is not very useful for theanswer extraction phase[3]. The target class is now represented as a hierarchy which is populated using some NLP techniques and knowledge resources i.e. WordNet. We also analyzed how the new hierarchy helped to prune out the extra unnecessary details for efficient answer extraction.

Question and Current Class	what cocktail inspired john doxat to write the book stirred-not shaken ? (ENTY:food)
Discussion	the head noun "*cocktail* " has the highest similarity with the class "*Food*" therefore the class assigned is "*Food* "
Question and Current Class	what country has the best defensive position in the board game diplomacy ? (LOC:country)
Discussion	the head noun "*country*" is already in the hierarchy therefore the class "*country*" is assigned.
Question and Current Class	what time of year is air travel the heaviest ? (NUM:date)
Discussion	the head noun "*year*" is a date with highest similarity therefore class "*date*" is assigned.
Question and Current Class	what color is the cross on switzerland 's flag ? (ENTY:color)
Discussion	the head noun "*color*" is already in the hierarchy therefore the class "*color*" is assigned.
Question and Current Class	which type of soda has the greatest amount of caffeine ? (ENTY:food)
Discussion	the head noun "*soda*" is a food therefore the class "*food* " is assigned.

Table 5. Comparison of classes assigned

6. Acknowledgements

We thank the scholarship donor University of Balochistan, NECTEC and NAIST lab, Kasetsart University for the support during the research.

References

[1] Answers.com. Getting answers to questions.

[2] Maheen Bakhtyar, Junaid Baber, Asanee Kawtrakul, Doudpota, S. M. (2012). Creating missing classes automatically to improve question classification in question answering systems. *In*: 7th International Conference on Digital Information Management (ICDIM), Macau.

[3] Maheen Bakhtyar, Asanee Kawtrakul, (2011). Integrating knowledge resources and shallow language processing for question classification. *In*: Proceedings of the KRAQ11 workshop, p. 22–28, Chiang Mai, November. *Asian Federation of Natural Language Processing*.

[4] Boldrini, E., Ferrández, S., Izquierdo, R., Tomás, D., Ferrández, O., Vicedo, J. L. (2009). A proposal of expected answer type and named entity annotation in a question answering context. *In*: Proceedings of the 2nd conference on Human System Interactions, HSI'09, p. 315–319, Piscataway, NJ, USA. IEEE Press.

[5] Davide Buscaldi, Paolo Rosso, José Manuel Gómez-Soriano, Emilio Sanchis. (2010). Answering questions with an n-gram based passage retrieval engine. *J. Intell. Inf. Syst.*, 34, 113–134, April.

[6] Sanda, M., Harabagiu, Dan, I., Moldovan, Marius Pasca, Rada Mihalcea, Mihai Surdeanu, Razvan, C., Bunescu, Roxana Girju, Vasile Rus, Paul Morarescu. (2000). Falcon: Boosting knowledge for answer engines. *In*: TREC, p. 479–488.

[7] Eduard Hovy, Laurie Gerber, Ulf Hermjakob, Chin-Yew Lin, Deepak Ravichandran. (2001). Toward semanticsbased answer pinpointing. In HLT '01: Proceedings of the first international conference on Human language technology research, p. 1–7, Morristown, NJ, USA. *Association for Computational Linguistics*.

[8] Peng Huang, Jiajun Bu, Chun Chen, Guang Qiu. (2007). An effective feature-weighting model for question classification. *In*: Proceedings of the 2007 *International Conference on Computational Intelligence and Security*, CIS'07, p. 32–36, Washington, DC, USA. IEEE Computer Society.

[9] Claudia Leacock, Martin Chodorow. (1998). Combining Local Context and WordNet Similarity for Word Sense Identification, 11, 265–283. The MIT Press, May.

[10] Lehnert, W. (1986). Readings in natural language processing. chapter A conceptual theory of question answering, p. 651–657. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.

[11] Xin Li, Dan Roth. (2002). Learning question classifiers. In Proceedings of the 19th *international conference on Computational linguistics*, p. 1–7, Morristown, NJ, USA. Association for Computational Linguistics.

[12] Xin Li, Dan Roth. (2006). Learning question classifiers: the role of semantic information, *Natural Language Engineering*, 12 (3) 229–249.

[13] Dan Moldovan, Marius Pa¸sca, Sanda Harabagiu, Mihai Surdeanu. (2003). Performance issues and error analysis in an open-domain question answering system. *ACM Trans. Inf. Syst.*, 21:133–154, April.

[14] Xiaojun Quan, Liu Wenyin, Bite Qiu. (2011). Term weighting schemes for question categorization. *IEEE Trans. Pattern Anal. Mach. Intell.*, 33 (5) 1009–1021.

[15] Santosh Kumar Ray, Shailendra Singh, Joshi, B. P. (2010). A semantic approach for question classification using wordnet and wikipedia. Pattern Recognition Letters, 31 (13) 1935 – 1943. *Meta-heuristic Intelligence Based Image Processing.*

[16] Philip Resnik. (1995). Using information content to evaluate semantic similarity in a taxonomy. *In*: Proceedings of the 14th international joint conference on Artificial intelligence - 1, 448–453, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

[17] Roth, D., Cumby, C., Li, X., Morie, P., Nagarajan, R., Punyakanok, V., Rizzolo, N., Small, K., Yih, W. (2002). Questionanswering via enhanced understanding of questions. In TREC.

[18] Wanpeng Song, LiuWenyin, Naijie Gu, Xiaojun Quan, Tianyong Hao, (2011). Automatic categorization of questions for user-interactive question answering. Inf. Process. Manage., 47, 147–156, March.

[19] Text-REtrieval-Conference-10. Trec-10 question answering data, (2001).

[20] Ellen M. Voorhees. (2002). Overview of the trec 2002 question answering track. In In Proceedings of the Eleventh Text Retrieval Conference (TREC), p. 115–123.

[21] Zhibiao Wu, Martha Palmer. (1994). Verbs semantics and lexical selection. In Proceedings of the 32nd annual meeting on Association for Computational Linguistics, ACL ’94, p. 133–138, Stroudsburg, PA, USA. Association for Computational Linguistics.

[22] Zhengtao Yu, Lei Su, Lina Li, Quan Zhao, Cunli Mao, Jianyi Guo, (2010). Question classification based on cotraining style semi-supervised learning. Pattern Recogn. Lett., 31, 1975–1980, October.