Dependency Extraction for Knowledge-based Domain Classification

Lokesh Kumar Sharma
Dept. of Computer Science & Engineering
Malaviya National Institute of Technology
Jaipur, Rajasthan, India
2013rcp9007@mnit.ac.in

Namita Mittal
Dept. of Computer Science & Engineering
Malaviya National Institute of Technology
Jaipur, Rajasthan, India
nmittal.cse@mnit.ac.in

Abstract

Question classification is an important part in Question Answering. It refers to classifying a given question into a category. This paper presents a learning based question classifier. The previous works in this field have used UIUC questions dataset for the classification purpose. In contrast to this, we use the WebQuestions dataset to build the classifier. The dataset consists of questions with the links to the Freebase pages on which the answers will be found. To extract the exact answer of a question from a Freebase page, it is very essential to know the domain of the answer as it narrows down the number of possible answer candidates. Proposed classifier will be very helpful in extracting answers from the Freebase. Classifier uses the questions' features to classify a question into the domain of the answer, given the link to the freebase page on which the answer can be found.

1 Introduction

Question classification refers to finding out the class to which a question belongs (Loni, 2011). In traditional question answering systems, the answers were extracted from the corpora. But more recent question answering systems use structured knowledge bases for extracting answers. One of the popular knowledge bases is the Freebase. Freebase is an online collection of structured data harvested from many sources. Freebase aims to create a global resource which allows people (and machines) to access common information more effectively. It was developed by the American software company Metaweb. It has over 39 million topics about real-world entities like people, places and things. The freebase data is organized and stored in the form of a graph and each node in a graph has a unique id. The data is classified into one of the 71 domains like people, location, sports etc. The domains comprise of types and types comprise of properties. To extract an answer from the freebase we write MQL queries to query over Freebase (Yao et. al, 2014A, Yao et. al,2014B). For example: *"Who are the parents of Justin Bieber?"*. The MQL query to find out the answer will be The above MQL query, searches the freebase page with the id "/en/justin_bieber". The answer field /people/person/parents is left blank. The query returns the answer by filling in the answer fields. The /people is the domain which has /person as a type and this type has /parents as a property. It is shown in Fig. 1. If we do this manually by going to the freebase page named Justin Bieber, we have to scan through all the properties to find the answer. But by knowing the domain i.e. *people*, only the answers which fall under the *people* category are scanned. Hence number of possible answer candidates are reduced which considerably reduces our effort and time in finding the answer. In this approach, we build a learning based classifier which can classify a question into the domain of the answer. In case of multiple domains, we select the most suitable

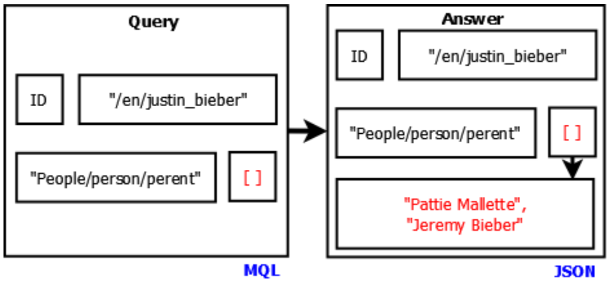


Figure 1: MQL query and answer from Freebase

D S Sharma, R Sangal and E Sherly. Proc. of the 12th Intl. Conference on Natural Language Processing, pages 215–218,
Trivandrum, India. December 2015. ©2015 NLP Association of India (NLPAI)

domain. We have assumed that only the common nouns can be the headwords of the questions. As a result, only ~50% of the questions have headwords present in them. We then use the feature set for these ~500 questions for training using a LIBSVM classifier. We get an accuracy of 76.565%.

2 Related Work

The previous works in this field have classified the UIUC dataset published by Li and Roth (2004). Li and Roth also defined a taxonomy with 6 course and 50 fine grained classes. They used four question features namely, (1) automatically acquired named entity categories, (2) word senses in WordNet 1.7, (3) manually constructed word lists related to specific categories of interest, and (4) automatically generated semantically similar word lists. They obtained an accuracy of 92.5% for 6 coarse classes and 89.3% for 50 fine grained classes using a learning bases classification approach. Silva et al. (2011) used 5 features namely, (1) wh-word, (2) headword, (3) WordNet semantic feature, (4) N-grams and (5) word shape. They get an accuracy of 89.2% for coarse grained and 93.4% for fine grained classes, using learning based approach. Silva et al. (2011) obtained an accuracy of 90.8% on fine grained and 95.0% on coarse grained classes which is the highest accuracy reported on this dataset. They used a hybrid of the rule based and the learning based approach.

We use the before mentioned three features because they have contribute the most to the classification process in the previous works mentioned above. Thus we have a relatively smaller but rich feature set.

3 Proposed Approach

A question can be treated as a bag of features. These features then help us to map the question with a class. We have considered three features namely, the wh-tag, the similarities of the headword with the four domains (obtained using WordNet Similarity package) and the question unigrams as features. We use the WebQuestions dataset for training and testing the classifier. The WebQuestions dataset was created by the Stanford NLP group. It consists of 3,778 training examples and 2,032 test examples. On WebQuestions, each example contains three fields: utterance: natural language utterance. TargetValue: The answer provided by AMT workers, given as a list of descriptions. url: Freebase page, where AMT workers found the answer. We use the training samples from the above dataset. We find out all the factoid questions (which have a single answer) from the dataset. There are about 2575 factoid questions out of 3778.

We find out the headwords of all the given questions. A headword is a word which the question is seeking (a noun). For example: "*Who are the parents of Justin Bieber?*" Here the word *parents* is the headword. Out of 2575 nearly 1303 questions have headwords present. We then use ~500 questions from these questions for the training of our classifier. We manually classify the questions by navigating to the url given with each question and searching for the answer. Then the domain of the answer is noted as the question class. All the questions fall into one of the four classes (domains) namely, people, location, government and sports. The classifier uses question features for training the dataset. In this approach, the question is treated as a bag of features. We use three features namely, the wh-tag, the similarities of the headword with the four domains (obtained using WordNet Similarity package) and the question unigrams as features.

3.1. Wh-word

A Wh-word is the word starting with *wh* with which the question begins. In our dataset it is one of *what, which, why, where* and *who*. If the wh-word is *who*, then the question has a high probability of falling into the *people* class. Similarly, for the wh-word being *where*, the question may fall in the *location* class.

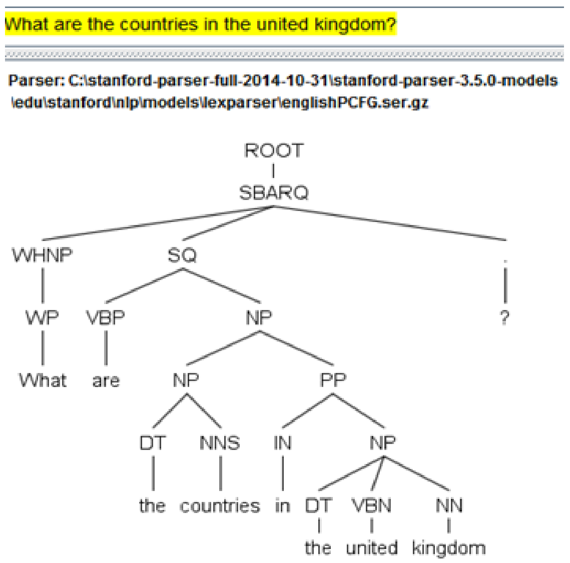
3.2. Headword similarities with the four classes

A headword is a word the question is seeking. It plays an important role in classifying a question. For example if the headword is brother, the question is seeking for a person and will probably fall into the *people* class. This is because the word brother is semantically related to the class *people* more than the other three classes. Thus, we can use the similarities of the headword with the four classes as four separate features and take the active

Table I
Dependency tree's priority list

Parent	Direction	Priority List
S	Left	VP S FRAG SBAR ADJP
SBARQ	Left	SQ S SINV SBARQ FRAG
SQ	Left	NP VP SQ
NP	Right by position	NP NN NNP NNPS NNS NX
PP	Left	WNP NP WHADVP SBAR
WHNP	Left	NP
WHPP	Right	WHNP WHADVP NP SBAR

feature as the one corresponding to the class to which the headword is the most similar. To find out the similarities we make use of WordNet (Miller, 1995). WordNet is a large English lexicon in which meaningfully related words are connected via cognitive synonyms (synsets). The WordNet is a useful tool for word semantics analysis and has been widely used in question classification. We make use of the WordNet Similarity package to find out the similarities. For a pair of words, this package computes the length of the path to reach from one of the words to the other via the WordNet network. It then computes the similarity based on the path. The class which has the highest similarity with the headword is marked as an active feature. Starting from the root of the parse tree shown in Fig. 2, the current node is matched with a parent. Depending upon the parent, all the child nodes of the current node are then compared with the corresponding priority list elements by the manner specified by direction. If the direction of search is left by category then the algorithm starts from the leftmost child and check it against items in priority list and if it matches any, then the matched item will be returned as head. Otherwise if the algorithm reaches the end of the list and the child does not match with any of the items, it continues the same process with the next child. On the other hand, if the search direction is left by position, then the algorithm first starts checking the items in priority list and for each item it tries to match it with every child from left to right. The first matched item is considered as head. we applied the head rules starting from the root. The root of the tree is SBARQ. On matching it with the parent tag on table I, we get the rule comparing the children of SBARQ with the right hand side of the rule we get a match at SQ. We then match it with the table and get the rule NP matches the right hand side and in this way the procedure continued till we reach at NN and capital is returned as the headword. Apart from the head rules mentioned in table I, there are three more rules which have been used to find out the similarities are shown in table II. Thus, the question will fall into the class *government* as it has the highest similarity with the question headword. Hence we use the semantic meaning of the headword for classification.


What are the countries in the united kingdom?

Parser: C:\stanford-parser-full-2014-10-31\stanford-parser-3.5.0-models \edu\stanford\nlp\models\lexparser\englishPCFG.ser.gz

Figure 2: Parse tree traversal based on priority list

Table II
Domain classes and their similarities

Classes	Similarity
Capital and People	0.1429
Capital and Location	0.2500
Capital and Government	0.3333
Capital and Sports	0.1111

headwords for some of the exceptions to the head rules. parse tree traversal shown in figure 2. By default the comparison is by category. They are:

1) When SBARQ has a WHXP child with at least two children, WHXP is returned.
2) After applying the first rule, if the resulting head is a WHNP containing an NP that ends with a possessive pronoun (POS), we return the NP as the head.
3) If the extracted headword is name, kind, type, part, genre or group, and its sibling node is a prepositional phrase PP, we use PP as the head and proceed with the algorithm.

After finding out the headword we use the WordNet similarity package to find out the semantic similarity of the headword with the four classes.

3.3. N-grams

An *n*-gram is a contiguous sequence of *n* items from a given sequence of text or speech. We have considered the question unigrams as features because they are simple and contribute in the question classification process better than bigrams and trigrams. We have not used the proper nouns in the unigram feature set as the proper nouns cannot help in the classification process. Further we have also removed the stop words from the unigrams because of their triviality. For example for the question: *What is the capital of India?* The unigram feature set will be {(What,1),(capital,1)}. The Wh-word will also be a part of the unigrams.

4. Experimental Setup and Analysis

Out of 500 questions 379 questions are classified correctly shown in table III. The resultant matrix shown in table IV shows the accuracy of the classified questions. In the idea case the matrix should have all the non diagonal elements as 0. We see that the there is a discrepancy in the classification matrix. The discrepancy is maximum for the sports class. The discrepancy occurs because we have tried to classify the dataset using only a compact set of 312 features. To improve the accuracy, we can increase the number of features and the no of questions used for training. Also the domains under the freebase cannot be classified using the conventional classification techniques. We cannot always correctly classify the questions into their freebase domains by the question's features. Thus the classifier performs average as expected for the freebase domain classification as compared to the conventional classification using the taxonomy proposed by Li and Roth (2004).

TABLE III
CORRECT AND INCORRECT CLASSIFIED INSTANCES OUT OF 500

Correctly Classified	379	76.565%
In-Correctly Classified	116	23.434%

TABLE IV
RESULTANT MATRIX FOR FOUR CATEGORIES

A	B	C	D	CLASSIFIED AS
223	24	2	4	a = location
36	69	2	9	b = people
4	8	47	1	c = sports
12	11	3	40	d = government

5. Conclusion and Future Work

The accuracy obtained in classification is ~76% which can be improved by increasing the size of the feature set by adding more features like bigrams, N-grams, Word-Shapes, Question-Length, Hypernyms, Indirect-Hypernyms, Synonyms, Name Entities and Related-Words can be added to the feature set. The Wordnet similarity used in the project gives us the general similarity between two words. Hence a sense disambiguation technique can be used to improve the classification. Also we have worked on the domain classification, which is a new field of question classification. It will be helpful during answer extraction from the Freebase. We have ~312 features for each question, and the number of questions used is ~500. Thus with a comparatively compact dataset we have received an accuracy of 76% which is promising for any future work in this field.

References

Babak Loni, *"A Survey of State-of-the-Art Methods on Question Classication"*, Delft University of Technology, Tech. Rep, 2011.

João Silva, Luísa Coheur, Ana Cristina Mendes, Andreas Wichert, *"From symbolic to sub-symbolic information in question classification"*, 2011.

George A. Miller, *"WordNet: A Lexical Database for English"* Communications of the ACM Vol. 38, No. 11: 39-41, 1995.

Xin Li and Dan Roth, "*Learning Question Classifiers: The Role of Semantic Information",* 2004.

Xuchen Yao and Benjamin Van Durme, *"Information Extraction over Structured Data: Question Answering with Freebase"*, Proceedings of ACL 2014.

Xuchen Yao, Jonathan Berant, Benjamin Van Durme, "*Freebase QA: Information Extraction or Semantic Parsing",* Proceedings of ACL 2014.