Question Type Classification using Language Modelling Enhanced with Head Words and Hypernyms

Justin Domingue – 260588454
justin.domingue@mail.mcgill.ca

Abstract

Question type classification is a particularly promising area of research which is often used in Question Answering systems to improve accuracy. This report takes a look at question type classification using language modelling enhanced with head words and hypernyms as suggested in Huang et al. (2008). More specifically, we look at extracting n-grams, word shapes and head words and their hypernyms from questions in order to classify them into coarse and fine categories. Using SGD and SVC models, we obtained accuracies of 82.8% and 84% on 50 class classification and 90.4% and 91% on 6 class classification respectively using standard benchmark dataset.

1 Introduction

Researchers have established that the next step in information retrieval is to extract concise and succinct answers to user queries rather than giving back whole documents (Sundblad, 2007).

A Question Answering (QA) system can be built to respond to open-ended [1] natural language questions by retrieving the relevant information from stored documents. However, without some sort of question type classification, the range of answers is quite large. Question type classification plays a crucial role in bounding the type of answers returned by the QA. For instance, the answer to the question *When did Beethoven finish the Eroica?* should be a **date** while that of *What is an appoggiatura in music?* should be a **definition**.

Question type classification is then defined as "the task of determining the correct type of the answer expected to a given query" (Khoury, 2011). There are few ways of accomplishing this task. Amongst others, one could use rule-based classification or a machine learning approach – learning from a labeled data set. It seems obvious that simple rules cannot account for all the nuances of English syntax. Take for example a rule-based classification system which would attribute questions with a particular wh-word to a given category. Consider the sentences *What is the capital of Canada?* and *What is an espresso?*.The first sentence is of type **location** (city) while the second is of type **definition**. Clearly, a more sophisticated set of rules would need to be devised. On the other hand, a statistical approach has the advantage that "one can focus on designing insightful features, and rely on learning process to efficiently and effectively cope with the features" (Huang et al., 2008).

Work on machine learning approaches to question classification was initiated by Li and Roth in 2002 (Li and Roth, 2006). Other work in this area include (Pinto et al., 2002) who employed entity tagging, part of speech tagging, regular expressions and language models, Sunblad (2007) who carried out a performance analysis of various machine learning algorithms applied to the question classification task and Khoury (2011) who introduced a Part-of-Speech Hierarchy coupled with informer spans.

More related to the present paper, Huang et al. (2008) have built upon the work of Li and Roth (2006) and defined new features, namely head words and hypernyms. They have found a significant increase of the model accuracy. In this paper, we follow the path of Huang et al. (2008) to confirm their results.

2 Method

In this section, we first present the data set used, then two classifiers – SGD and Linear SVC – employed in the experiment as well as the features

[1] In opposition to close-ended questions which only require a "yes" or "no" answer, open-ended questions demand more than one word answers.

extracted from the data set.

2.1 Data Set

The data set used was compiled by Li and Roth[2]. They hand labeled each question according to the hierarchy they defined[3], keeping only one label per question based on a majority vote from their annotators. The training data consists of 5,500 labeled questions (4,500 English questions published by USC, 894 TREC 8 and 9 questions and about 500 manually constructed questions for a few rare classes). The testing set consists of 1,000 label questions taken from TREC 10 and 11. Both data set contains factual (open-ended) questions only (Li and Roth, 2006).

2.2 Classifiers

Two classifiers were used in the experiment: a stochastic gradient descent classifier (SGD) and a support vector classifier (SVC). Both perform similarly although a slight edge has to be given to SVC.

2.3 Features

In this experiment, we made use of the five binary feature types described by Huang et al. (2008). They defined, from existing literature, wh-words, n-grams, and word shape. They added two new features: head words and their hypernym. While previous attempt at classifying questions sometimes used head chunks – meaningful spans of words, Huang et al. (2008) posited that using only head words and their hypernyms would drastically reduce the feature space. They argued that every feature would then have a more meaningful contribution (Huang et al., 2008).

2.3.1 Wh-word

Wh-word simply refers to the question word contained in the question text. The wh-word is taken to be the first word of the question if it is a wh-word, or *rest* otherwise: *what, which, when, where, who, how, why, rest*.

2.3.2 N-grams

N-grams are sequences of n contiguous words in a text. Unigrams, bigrams and trigrams were extracted from the question text.

2.3.3 Word shape

Word shape refers to the letter case of each word in the question text. Five cases are distinguished: all lowercase, all uppercase, mixed, all digits or other. The individual word shapes are then combined to form a string of word shapes. N-grams are then extracted from that string. The idea behind this feature is that questions with the same type can usually share a same word shape. Amongst others, word shape can help classify questions with proper nouns by serving as a loose named entity recognizer and questions with digits (dates, codes, counts, money, etc.) by recognizing digits and their relation to the other words.

2.3.4 Head word

Early question classification attempts have included head chunks – the first noun chunk and the first verb chunk of a sentence (Li and Roth, 2006), or *informer spans* – contiguous span of tokens (Huang et al., 2008). Huang et al. (2008) have observed that these features introduced much noise. As an example, they gave the sentence *What is a group of turkeys called ?* and explained that both its head chunk and its informer span is *group of turkeys*. They argued that while the word *turkeys* contribute to a classification type ENTY:Animal, the word *group* was misleading the classifier into giving type HUMAN:group. They noted that the issue here is that *turkeys* and *group* are treated equally. Thus, they proposed to introduce a new feature: head words. Head words are defined as "one single word specifying the object that the question seeks" (Huang et al., 2008). Given the previous example, they identify the head word of the question as *turkeys*. The relevant word *turkeys* now has more weight than the misleading word *group*.

Extracting the head word of a question necessitates a syntactic parser. A set of rules are then used to traverse the syntactic tree to find the head word. For syntactic parsing, we made use of the BLLIP reranking parser, also known as Charniak-Johnson parser[4]. Each question in the data set was parsed using that state-of-the-art parser. After obtaining a parse, two sets of rules were applied.

In their paper, Huang et al. (2008) introduced a set of regular expressions to help identify head words. They observed that in questions like *What are string quartets?* or *What is an appog-*

[2] Available here: http://cogcomp.cs.illinois.edu/Data/QA/QC/

[3] See their question classification taxonomy here: http://cogcomp.cs.illinois.edu/Data/QA/QC/definition.html

[4] available at https://github.com/BLLIP/bllip-parser

giatura?, the head word – respectively, *quartets* and *appoggiatura* is of little help at classifying them as **definitions**. They thus defined a regular expression matching any sentence beginning with *what is/are*, followed by an optional determiner plus one or two words. Hence, if a question matches a regular expression, a binary feature is introduced to the feature set of the question. The researchers have given the following regular expressions (Huang et al., 2008):

DESC:def pattern 1 The question begins with what is/are and follows by an optional a, an, or the and then follows by one or two words.

DESC:def pattern 2 The question begins with what do/does and ends with mean.

ENTY:substance pattern The question begins with what is/are and ends with composed of/made of/made out of.

DESC:desc pattern The question begins with what does and ends with do.

ENTY:term The question begins with what do you call.

DESC:reason pattern 1 The question begins with what causes/cause.

DESC:reason pattern 2 The question begins with What is/are and ends with used for.

ABBR:exp pattern The question begins with What does/do and ends with stand for.

HUM:desc pattern The question begins with Who

If none of these regular expressions matches the question, then the head word is found using (modified) Collins rules (Collins, 2003). In his PhD thesis, Collins defined a set of rules for finding *syntactic* head words. The rules are of the following form (tag, direction, candidates) where tag is a tag in the Penn Treebank Tag-set[5], $candidates$ is a list of tags and $direction$ is the direction in which to look at the child of the tree starting at tag. For instance, given a tree rooted at an SQ node, the head word finder would look at the immediate children of the root, from left to right, and return the head word of the first VBZ. If there is no VBZ, then it would return the head word of the first VBD, If there is no VBD, it would return the first head word of the first VB, etc. Note that head finding should be done bottom-up – meaning that the head word of sentence is found by finding head word of partial trees from the ground up. Refer to Collins' thesis (2003) for a full account of head word finding.

Huang et al. (2008) have given indications on possible modifications to extract the *semantic* head words: the head preference is transferred from verb or verb phrase to noun or noun phrase for the rules to find head word of phrases SBARQ, SQ, VP and SINV.

Here is the final algorithm given by Huang et al. (2008) in their paper (copied as-is for convenience) and also the one used in this experiment:

Algorithm 1 Question head word extraction

```
Require: : Question q
Ensure: : Question head word
if $q.type == when|where|why$ then
    return null
end if
if $q.type == how$ then
    return the word following word ”how”
end if
if $q.type == what$ then
    for any aforementioned regular expression r (except HUM:desc pattern)
    if q matches r then
        return r.placehold-word
    end if
    end if
    if q.type == who && q matches HUM:desc pattern then
        return ”HUM:desc”
    end if
    String candidate = head word extracted from question parse tree
    if candidate.tag starts with NN then
        return candidate
    end ifreturn the first word whose tag starts with NN
```

2.3.5 Hypernyms

The hypernym of a word *w* is a word with a broader meaning that encompasses *w*. Introducing head word hypernyms can help improve the performance by reducing the effect of data sparsity. For this feature, WordNet[6], an extensive English lexicon, is used. More particularly, the WordNet interface defines hypernyms for each senses of a word. Extracting the hypernym of a head word given a chosen depth then involves going up the hypernyms chain. Huang et al. (2008) have defined direct and indirect hypernyms. They have found that the use of direct hypernyms enhanced the accuracy of the model the most. For this experiment, we chose to only implement direct hypernyms. First, we assumed that the part-of-speech tag of the head word is noun. we then found the

[5] See https://www.comp.leeds.ac.uk/ccalas/tagsets/upenn.html.

[6] https://wordnet.princeton.edu

most likely sense of the head word given that it is a noun. Finally, we traversed the hypernym chain up n degrees, where n is the depth to consider.

3 Experimental Results

This section presents the results of the experiments conducted. We have studied both the individual and incremental feature contributions on the question classification accuracy by training two classifiers – SGD and Linear SVC as discussed earlier – on the UIUC training data set containing 5,500 questions. The accuracy of the models has been evaluated with the UIUC test data set of 1,000 questions. Note that Huang et al. (2008) have used the accuracy as the evaluation measure as it facilitated the comparison of their experimental results with those of previous researchers. We have decided to do the same.

3.1 Individual Feature Contribution

Evaluating the individual contribution of predictors is a good way to get a sense of what features bring insightful information. Table 1 presents the question classification accuracy of individual features using stochastic gradient descent and linear support vector classification for 6 coarse and 50 fine classes. According to this table, bigrams (consisting of unigrams and bigrams) are the most informative feature amongst the one considered. At 89.8 for coarse and 83.0 for fine using SVC, the accuracy of bigrams outperforms the best n-grams feature of Huang et al. (2008) by 1.8% and 2.6% for coarse and fine classes respectively. Further data normalization in the present experiment might explain this. In fact, with the pointers given by Sunblad (2007) in their problematic question analysis, cross validation was used to determine the following optimal normalization parameters. We determined that lemmatization of every word in a question increases the accuracy of the model by 0.6%, probably because the data sparsity is reduced and can compensate for the relatively low number of training examples; both normalizing the case of every word and removing punctuation symbols had no significant impact; however, removing stop words negatively impacted the accuracy significantly: a drop of about 2.4% has been observed.

Our wh-word + head word feature does not perform as well. Huang et al. (2008) have reported an accuracy of 92.2% and 82.0% for the wh-word + head word feature for 6 class, ME model and 50 class, ME model respectively. To tell the truth, Huang et al. haven't been very explicit in describing how they extracted semantic head word. They also haven't proposed a metric to evaluate their head finder. It is thus difficult to compare the two implementations. In fact, we got an maximum accuracy of 86.8% for coarse classification, SVC and 76.6% for fine classification.

		6 class		50 class	
		SGD	SVC	SGD	SVC
wh-word + head word		86.6	86.8	76.4	76.6
wh-word +	depth=1	82.4	87.2	75.8	77.4
head word +	depth=3	85.6	86.6	76.2	77.7
hypernym	depth=6	83.8	86.2	76,8	77.6
unigrams		87.4	87.0	81.2	81.6
bigrams		89.2	89.8	81.8	83.0
trigrams		88.2	89.0	79.8	80.8
word shape		44.0	46.6	33.2	33.8

Table 1: Question classification accuracy of individual features using SGD and Linear SVC for 6 and 50 classes

The next feature analyzed was the WordNet semantic expansion (hypernyms). For coarse classification, we have found hypernyms to hinder the performance of the SGD model but it improved by 0.4% that of the SVC model for a depth of 1. Any other combination of model/depth has seen a decrease in the performance. It is interesting to note that the hypernym feature seems to have been mostly helpful for fine classification. In fact, a hypernym depth of 3 with an SVC model increased the accuracy by as much as 1.1%. This is in line with what Huang et al. (2008) reported in their paper: "WordNet hypernym benefits mainly on the 50 fine classes classification" (Huang et al., 2008). In sum, a depth of 1 brings the highest accuracy of 6-class SVC to 87.2% and a depth of 2, a highest accuracy of 50-class SVC to 77.7%.

It is worth mentioning that while Huang et al. (2008) report having used Lesk's algorithm for word sense disambiguation during hypernym extraction, in our test, doing so impeded the performance. Hence, we decided to always choose the most common sense for a word with part-of-speech "noun".

Finally, the word shape feature has an accuracy of 44% and 46.6% under SGD and SVC for coarse classification and an accuracy of 33.2% and 33.8% under SGD and SVC for fine classification. These results are very high compared to what Huang et al. (2008) found – 18.8% for 6-class and 10/4% for 50-class classifications. We suspect the treatment of word shape wasn't the same. Not only did we extract word shapes for the entire sentence, we also extracted smaller sequences of shapes, thus catching word patterns more finely.

3.2 Incremental Feature Contribution

Based on our observations of the contribution of individual features, we trained an SGD and an SVC model on the training data for coarse 6 class and fine 50 class question classification. First, we considered bigrams only, then added word shape to the bag of features, wh-word + head word and finally hypernym. As shown in table 2, the best accuracy for coarse (6 class) classification is 91% , achieved using bigrams and word shape only with an SVC model while the best accuracy for fine (50 class) classification is 84.6% using bigrams, word shape and wh-word + head word and SVC model. As expected from individual feature contribution, hypernyms hinder the performance of the models. This might due to the fact the our head word finder is not as sophisticated as the one developed by Huang et al. (2008). Hence, augmenting the feature space with the hypernym of the head word only brings noise.

Features	6 class		50 class	
	SGD	SVC	SGD	SVC
bigrams	89.2	89.8	82.2	83.0
+ word shape	89.4	**91.0**	81.6	84.0
+ wh-word + head word	90.4	90.2	82.8	**84.6**
+ hypernym	90.4	90.0	81.8	84.0

Table 2: Question classification accuracy with incremental features

We also looked at the performance of the model on a more fine-grained level shown in Table 3. This table highlights the most difficult classes for the SVC model. For example, the subclasses of **abbreviation** have F1-scores 1.00 and 0.93. However, many subclasses of **numerical** have low F-1 scores (around 0.5). Also, note that the UIUC test set is not representative of the category distribution: the relatively small category **description** contains more questions (138) than the biggest category **entity** (94 questions). Some categories are barely tested – e.g. creativity, food, title or abbreviation.

Note that our best result feature space for 6 class classification is composed of 35,851 binary features with about 20,000 features active per class. As Huang et al. (2008) have noted, this is a sharp contrast with the 200,000 feature size reported in Li and Roth (2006). It means that our feature space is much more informative.

class	#	P	R	F1
ABBR	9			
abb	1	1.00	1.00	1.00
exp	8	1.00	0.88	0.93
DESC	138			
def	123	0.88	0.98	0.93
desc	7	0.50	0.86	0.63
manner	2	0.67	1.00	0.80
reason	6	1.00	1.00	1.00
ENTITY	94			
animal	16	0.67	0.62	0.65
body	2	1.00	0.50	0.67
color	10	1.00	0.90	0.95
creat	0	0.00	0.00	0.00
currency	6	1.00	1.00	1.00
dismed	2	0.33	0.50	0.40
event	2	0.00	0.00	0.00
food	4	0.80	1.00	0.89
instru	1	1.00	1.00	1.00
lang	2	1.00	1.00	1.00
other	12	0.43	0.50	0.46
plant	5	1.00	0.80	0.89
product	4	1.00	0.25	0.40
sport	1	0.50	1.00	0.67
substance	15	0.88	0.47	0.61
techmeth	1	1.00	1.00	1.00
termeq	7	0.75	0.86	0.80
veh	4	1.00	0.75	0.86
HUMAN	65			
desc	3	1.00	1.00	1.00
gr	6	1.00	0.67	0.80
ind	55	0.89	1.00	0.94
title	1	0.00	0.00	0.00
LOC	81			
city	18	0.86	0.67	0.75
country	3	1.00	1.00	1.00
mount	3	0.67	0.67	0.67
other	50	0.82	0.80	0.81
state	7	0.86	0.86	0.86
NUM	113			
count	9	0.64	1.00	0.78
date	47	0.98	1.00	0.99
dist	16	1.00	0.56	0.72
money	3	0.50	0.33	0.40
other	12	1.00	0.42	0.59
perc	3	0.50	0.33	0.40
period	8	0.67	1.00	0.80
speed	6	1.00	0.50	0.67
temp	5	1.00	0.60	0.75
weight	4	1.00	0.50	0.67

Table 3: Support, precision, recall and F-1 score for question classification of fine grained classes on the test set using SVC

4 Discussion and Conclusion

From our experiment, head words and their hypernym do not bring significant improvement to language modelling – in fact, in some cases, it hindered the accuracy of the model tested, in sharp contrast with the findings of Huang et al. (2008). Table 3 suggested that more work can be done to specifically optimize difficult classes for the model – either by adding more questions to these categories or merging categories, as suggested in Sunblad (2007) or devising new features to handle them. We would also like for a metric to test head word extractors to be introduced – it would most likely be sufficient to collect a set of syntactic trees and hand label the semantic head word. Do note that more work has been done around finding syntactic head words (Collins, 2003) than semantic head words and so adapting that work might be insightful.

References

Michael Collins. 2003. Head-driven statistical models for natural language parsing. *Comput. Linguist.*, 29(4):589–637, December.

Zhiheng Huang, Marcus Thint, and Zengchang Qin. 2008. 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, Stroudsburg, PA, USA. Association for Computational Linguistics.

Richard Khoury. 2011. Question type classification using a part-of-speech hierarchy. In Mohamed Kamel, Fakhri Karray, Wail Gueaieb, and Alaa Khamis, editors, *Autonomous and Intelligent Systems*, volume 6752 of *Lecture Notes in Computer Science*, pages 212–221. Springer Berlin Heidelberg.

Xin Li and Dan Roth. 2006. Learning question classifiers: The role of semantic information. *Nat. Lang. Eng.*, 12(3):229–249, September.

David Pinto, Michael Branstein, Ryan Coleman, W. Bruce Croft, Matthew King, Wei Li, and Xing Wei. 2002. Quasm: A system for question answering using semi-structured data. In *In Proceedings of the Joint Conference on Digital Libraries (JCDL) 2002*, pages 46–55.

Håkan Sundblad. 2007. *Question Classification in Question Answering Systems*. Ph.D. thesis, Linköping Studies in Science and Technology.