Why Question Answering using Sentiment Analysis and Word Classes

Jong-Hoon Oh[*] **Kentaro Torisawa**[†] **Chikara Hashimoto** [‡]
Takuya Kawada[§] **Stijn De Saeger**[¶] **Jun'ichi Kazama**[‖] **Yiou Wang**[**]
Information Analysis Laboratory
Universal Communication Research Institute
National Institute of Information and Communications Technology (NICT)
{*rovellia,†torisawa,‡ch,§tkawada,¶stijn,‖kazama,**wangyiou}@nict.go.jp

Abstract

In this paper we explore the utility of sentiment analysis and semantic word classes for improving why-question answering on a large-scale web corpus. Our work is motivated by the observation that a why-question and its answer often follow the pattern that *if something undesirable happens, the reason is also often something undesirable*, and *if something desirable happens, the reason is also often something desirable*. To the best of our knowledge, this is the first work that introduces sentiment analysis to non-factoid question answering. We combine this simple idea with semantic word classes for ranking answers to why-questions and show that on a set of 850 why-questions our method gains 15.2% improvement in precision at the top-1 answer over a baseline state-of-the-art QA system that achieved the best performance in a shared task of Japanese non-factoid QA in NTCIR-6.

1 Introduction

Question Answering (QA) research for factoid questions has recently achieved great success as demonstrated by IBM's Watson at Jeopardy: its accuracy has been reported to be around 85% on factoid questions (Ferrucci et al., 2010). Although recent shared QA tasks (Voorhees, 2004; Peñas et al., 2011; Fukumoto et al., 2007) have stimulated the research community to move beyond factoid QA, comparatively little attention has been paid to QA for non-factoid questions such as *why* questions and *how to* questions, and the performance of the state-of-art non-factoid QA systems reported in the literature (Murata et al., 2007; Surdeanu et al., 2011; Verberne et al., 2010) remains considerably lower than that of factoid QA (i.e., 34% in MRR at top-150 results on why-questions (Verberne et al., 2010)).

In this paper we explore the utility of sentiment analysis (Pang et al., 2002; Turney, 2002; Nakagawa et al., 2010) and semantic word classes for improving why-question answering (why-QA) on a large-scale web corpus. The inspiration behind this work is the observation that why-questions and their answers often have the following tendency:

- *if something undesirable happens, the reason is often also something undesirable*, and
- *if something desirable happens, its reason is often also desirable*.

Consider the following question **Q1**, and its answer candidates **A1-1** and **A1-2**.

- **Q1**: Why does cancer occur?
- **A1-1**: Carcinogens such as nitrosamine and benzopyrene *may increase the risk of* cancer by altering DNA in cells.
- **A1-2**: Maintaining a healthy weight *may lower the risk of* various types of cancer.

Here **A1-1** describes an undesirable event related to cancer, while **A1-2** suggests a desirable action for its prevention. Our hypothesis suggests that **A1-1** is more appropriate for answering **Q1**. If this hypothesis holds, we can obtain a significant improvement in performance on why-QA tasks by exploiting the sentiment orientation[1] of expressions obtainable

[1] In this paper we denote the *desirable/undesirable* polarity of an expression by the term "sentiment orientation" instead of "semantic orientation" to avoid confusion with our different notion of "semantic word classes."

Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, pages 368–378, Jeju Island, Korea, 12–14 July 2012.

by automatic sentiment analysis of questions and answers.

A second observation motivating this work is that there are often significant associations between the lexico-semantic classes of words in a question and those in its answer sentence. For instance, questions concerning diseases like **Q1** often have answers that include references to specific semantic word classes such as chemicals (like **A1-1**), viruses, body parts, and so on. Capturing such statistical correlations between diseases and harmful substances may lead to higher why-QA performance. For this purpose we use classes of semantically similar words that were automatically acquired from a large web corpus using an EM-based clustering method (Kazama and Torisawa, 2008).

Another issue is that simply introducing the sentiment orientation of words or phrases in question and answer sentences in a naive way is insufficient, since answer candidate sentences may contain multiple sentiment expressions with different polarities in answer candidates (i.e., about 33% of correct answers had such multiple sentiment expressions with different polarities in our test set). For example, if **A1-2** contained a second sentiment expression with negative polarity like the example below,

> "Trusting a specific food *is not effective for* preventing cancer, but maintaining a healthy weight *may help lower the risk of* various types of cancer."

both **A1-1** and **A1-2** would contain sentiment expressions with the same polarity as that of **Q1**. Thus, it is difficult to expect that the sentiment orientation alone will work well for recognizing **A1-1** as a correct answer to **Q1**. To address this problem, we consider the combination of sentiment polarity and the contents of sentiment expressions associated with the polarity in questions and their answer candidates as well. To deal with the data sparseness problem arising in using the content of sentiment expressions, we developed a feature set that combines the polarity and the semantic word classes effectively.

We exploit these two main ideas (concerned with the sentiment orientation and the semantic classes described so far) for training a supervised classifier to rank answer candidates to why-questions. Through a series of experiments on 850 Japanese why-questions, we showed that the proposed semantic features were effective in identifying correct answers, and our proposed method obtained more than 15% improvement in precision of its top answer (P@1) over our baseline, which achieved the best performance in the non-factoid QA task in NTCIR-6 (Murata et al., 2007). We also show that our method can potentially perform with high precision (64.8% in P@1) when answer candidates containing at least one correct answer are given to our re-ranker.

2 Approach

Our proposed method is composed of *answer retrieval* and *answer re-ranking*. The first step, answer retrieval, extracts a set of answer candidates to a why-question from 600 million Japanese Web corpus. The answer retrieval is our implementation of the state-of-art method that has shown the best performance in the shared task of Japanese non-factoid QA in NTCIR-6 (Murata et al., 2007; Fukumoto et al., 2007). The second step, answer re-ranking, is the focus of this work.

2.1 Answer Retrieval

We use Solr[2] to retrieve documents from a 600 million Japanese Web page corpus[3]for a given why-question. Let a set of content words in a why-question be $T = \{t_1, \cdots, t_n\}$. Two boolean queries for a why-question, "t_1 AND $\cdots$ AND t_n" and "t_1 OR $\cdots$ OR t_n," are given to Solr and top-300 documents for each query are retrieved. Note that retrieved documents by each query have different coverage and relevance to a given why-question. To keep balance between the coverage and relevance of retrieved documents, we use a set of retrieved documents by these two queries for obtaining answer candidates. Each document in the result of document retrieval is split into a set of answer candidates consisting of five subsequent sentences[4]. Subsequent answer candidates can share up to two sentences to avoid errors due to wrong document segmentation.

[2] http://lucene.apache.org/solr

[3] To the best of our knowledge, few Japanese non-factoid QA systems in the literature have used such a large-scale corpus.

[4] The length of acceptable answer candidates for why-QA in the literature ranges from one sentence to two paragraphs (Fukumoto et al., 2007; Murata et al., 2007; Higashinaka and Isozaki, 2008; Verberne et al., 2007; Verberne et al., 2010).

Answer candidate ac for question q is ranked according to scoring function $S(q, ac)$ given in Eq. (1) (Murata et al., 2007). Murata et al. (2007)'s method uses text search to look for answer candidates containing terms from the question with additional clue terms referring to "reason" or "cause." Following the original method we used *riyuu* (*reason*), *genin* (*cause*) and *youin* (*cause*) as clue terms. The top-20 answer candidates for each question are passed on to the next step, which is answer re-ranking. $S(q, ac)$ assigns a score to answer candidates like tf-idf, where $1/dist(t_1, t_2)$ functions like tf and $1/df(t_2)$ is idf for given terms t_1 and t_2 that are shared by q and ac.

$$S(q, ac) = max_{t_1 \in T} \sum_{t_2 \in T} \phi \times log(ts(t_1, t_2)) \quad (1)$$

$$ts(t_1, t_2) = \frac{N}{2 \times dist(t_1, t_2) \times df(t_2)}$$

Here T is a set of terms including nouns, verbs, and adjectives in question q that appear in answer candidate ac. Note that the clue terms are added to T if they exist in ac. N is the total number of documents (600 million), $dist(t_1, t_2)$ represents the distance (the number of characters) between t_1 and t_2 in answer candidate ac, $df(t)$ is the document frequency of term t, and $\phi \in \{0, 1\}$ is an indicator, where $\phi = 1$ if $ts(t_1, t_2) > 1$, $\phi = 0$ otherwise.

2.2 Answer Re-ranking

Our re-ranker is a supervised classifier (SVMs) (Vapnik, 1995) that uses three types of feature sets: features expressing morphological and syntactic analysis of questions and answer candidates, features representing semantic word classes appearing in questions and answer candidates, and features from sentiment analysis. All answer candidates of a question are ranked in a descending order of their score given by SVMs. We trained and tested the re-ranker using 10-fold cross validation on a corpus composed of 850 why-questions and their top-20 answer candidates provided by the answer retrieval procedure in Section 2.1. The answer candidates were manually annotated by three human annotators (not by the authors). Our corpus construction method is described in more detail in Section 4.

3 Features for Answer Re-ranking

This section describes our feature sets for answer re-ranking: features expressing morphological and syntactic analysis (MSA), features representing semantic word class (SWC), and features indicating sentiment analysis (SA). MSA, which has been widely used for re-ranking answers in the literature, is used to identify associations between questions and answers at the morpheme, word phrase, and syntactic dependency levels. The other two feature sets are proposed in this paper. SWC is devised for identifying semantic word class associations between questions and answers. SA is used for identifying sentiment orientation associations between questions and answers as well as expressing the combination of each sentiment expression and its polarity. Table 1 summarizes the respective feature sets, each of which is described in detail below.

3.1 Morphological and Syntactic Analysis

MSA including n-grams of morphemes, words, and syntactic dependencies has been widely used for re-ranking answers in non-factoid QA (Higashinaka and Isozaki, 2008; Surdeanu et al., 2011; Verberne et al., 2007; Verberne et al., 2010). We use MSA as a baseline feature set in this work.

We represent all sentences in a question and its answer candidate in three ways: morphemes, word phrases (*bunsetsu*[5]) and syntactic dependency chains. These are obtained using a morphological analyzer[6] and a dependency parser[7]. From each question and answer candidate we extract n-grams of morphemes, word phrases, and syntactic dependencies, where n ranges from 1 to 3. Syntactic dependency n-grams are defined as a syntactic dependency chain containing n word phrases. Syntactic dependency 1-grams coincide with word phrase 1-grams, so they are ignored.

Table 1 defines four types of MSA (MSA1 to MSA4). MSA1 is n-gram features from all sentences in a question and its answer candidates and distinguishes an n-gram feature found in a question from that same feature found in answer candidates. MSA2 contains n-grams found in the answer

[5] A *bunsetsu* is a syntactic constituent composed of a content word and several function words such as post-positions and case markers. It is the smallest unit of syntactic analysis in Japanese.

[6] http://nlp.ist.i.kyoto-u.ac.jp/EN/index.php?JUMAN

[7] http://nlp.ist.i.kyoto-u.ac.jp/EN/index.php?KNP

MSA1	Morpheme n-grams, word phrase n-grams, and syntactic dependency n-grams in a question and its answer candidate, where n ranges from 1 to 3. n-grams in a question and those in an answer candidate are distinguished.
MSA2	**MSA1**'s n-grams in an answer candidate that contain a question term.
MSA3	**MSA1**'s n-grams that contain a clue term including *riyuu* (reason), *genin* (cause) and *youin* (cause). These n-grams in a question and those in an answer candidate are distinguished.
MSA4	The ratio of the number of question terms in an answer candidate to the total number of question terms.
SWC1	Word class n-grams in a question and its answer candidate. These n-grams in a question and those in an answer candidate are distinguished.
SWC2	**SWC1**'s n-grams in an answer candidate whose original **MSA1**'s n-grams contain any question term.
SA@W1	Word polarity n-grams in a question and its answer candidate. These n-grams in a question and those in an answer candidate are distinguished.
SA@W2	**SA@W1**'s n-grams in an answer candidate whose original **MSA1** n-grams contain any question term.
SA@W3	Joint class-polarity n-grams in a question and its answer candidate. These n-grams in a question and those in an answer candidate are distinguished.
SA@W4	**SA@W3**'s n-grams in an answer candidates whose original **MSA1** n-grams contain any question term.
SA@P1	The indicator for polarity agreement between sentiment phrases, one in a question and the other in an answer candidate: 1 if any pair of such sentiment phrases has polarity in agreement, 0 otherwise.
SA@P2	The phrase-polarity, positive or negative, of a pair of sentiment phrases for which the indicator in SA@P1 is 1.
SA@P3	Morpheme n-grams, word phrase n-grams, and syntactic dependency n-grams in sentiment phrases are coupled with their phrase-polarity, where n ranges from 1 to 3. These n-grams in a question and those in an answer candidate are distinguished.
SA@P4	**SA@P3**'s n-grams in an answer candidates that contain a question term.
SA@P5	The ratio of the number of question terms in sentences that have sentiment phrases in answer candidates to the total number of question terms.
SA@P6	Word class n-grams in sentiment phrases are coupled with phrase-polarity. These n-grams in a question and those in an answer candidate are distinguished.
SA@P7	**SA@P6**'s n-grams in an answer candidates, whose original **MSA1**'s n-grams include any question term.
SA@P8	Joint class-polarity n-grams in sentiment phrases of a question and its answer candidate are coupled with phrase-polarity of the sentiment phrases. These n-grams in a question and those in an answer candidate are distinguished.
SA@P9	**SA@P8**'s n-grams in an answer candidates, whose original **MSA1**'s n-grams include any question term.
SA@P10	A pair of **SA@P6**'s n-grams, one from sentiment phrases in a question and the other from those in an answer candidate, where the two sentiment phrases should have the same sentiment orientation.

Table 1: Features used in training our re-ranker

candidates that themselves contain a term from the question (e.g., "*types of cancer*" in example **A1-2**). MSA3 is the n-gram feature that contains one of the clue terms used for answer retrieval (*riyuu* (reason), *genin* (cause) or *youin* (cause)). Here too, n-grams obtained from the questions and answer candidates are distinguished. Finally, MSA4 is the percentage of the question terms found in an answer candidate.

3.2 Semantic Word Class

Semantic word classes are sets of semantically similar words. We construct these semantic word classes by using the noun clustering algorithm proposed in Kazama and Torisawa (2008). The algorithm follows the distributional hypothesis, which states that semantically similar words tend to appear in similar contexts (Harris, 1954). By treating syntactic dependency relations between words as "contexts," the clustering method defines a probabilistic model of noun-verb dependencies with hidden classes as:

$$p(n, v, r) = \sum_c p(n|c)p(\langle v, r\rangle|c)p(c) \qquad (2)$$

Here, n is a noun, v is a verb or noun on which n depends via a grammatical relation r (post-positions in Japanese), and c is a hidden class. Dependency relation frequencies were obtained from our 600-million page web corpus, and model parameters $p(n|c)$, $p(\langle v, r\rangle|c)$ and $p(c)$ were estimated using the EM algorithm (Hofmann, 1999). We successfully clustered 5.5 million nouns into 500 classes. For each noun n, EM clustering estimates a probability distribution over hidden variables representing semantic classes. From this distribution we obtained discrete semantic word classes by assigning each noun n to semantic class $c = argmax_{c^*}\ p(c^*|n)$. The resulting classes actually form clean semantic categories such as *chemicals*, *nutrients*, *diseases* and *conditions*, in our examples of Q1 and Q2. The following are the top-10 words (English translation) according to $p(c|n)$ for these classes.

chemicals: acetylene, hydrogenation product, phosphoric monoester, methacrylate, levoglucosan, ammonium salt, halogenated organic compound, benzonitrile, alkyne, nitrosamine

nutrients: glucide, carbonhydrate, mineral, salt, sugar, water, fat, vitamin, nutrients, protein

diseases: pneumonia, neuritis, cancer, oral leukoplakia, pachymeningitis, acidosis, encephalitis, abdominal injury, valvulitis, gingivitis

conditions: proficiency, decrepitude, deficiency, impurity, abnormalities, floated, crisis, displacement, condition, shortage

Semantic word class (SWC) features are used to capture associations between semantic classes of words in the question and those in the answer candidates. For example:

- **Q2**: Why does rickets ($W_{disease}$) occur in children?
- **A2**: Deficiency ($W_{condition}$) of vitamin D ($W_{nutrients}$) can cause rickets ($W_{disease}$).

$W_{condition}$, $W_{disease}$ and $W_{nutrients}$ represent semantic word classes of *conditions*, *diseases* and *nutrients*, respectively. If this question-answer pair is given to the classifier as a positive training sample, we expect it to learn that if a disease name appears in a question then, everything else being equal, answers including nutrient names are more likely to be correct. Note that in principle the same association could be learned between word pairs, i.e., *rickets* and *vitamin D*. However, we found that word level associations are often too specific, and because of data sparseness this knowledge cannot easily be generalized to unseen questions. This is our main motivation for introducing broad coverage semantic word classes into the feature set.

We call the feature set with the word classes SWC and use two types of SWC, as shown in Table 1. To obtain the first type (SWC1), we convert all nouns in the MSA1 n-grams into their respective word classes, and keep all n-grams that contain at least one word class. We call these features *word class n-grams*. Again, word class n-grams obtained from questions are distinguished from the ones in answer candidates. For example, we extract "$W_{disease}$ occur" as a word class 2-gram from **Q2**.

The second type of SWC, SWC2, represents word class n-grams in an answer candidate, in which question terms are replaced by their respective semantic word classes. For example, $W_{disease}$ in word class 2-gram "cause $W_{disease}$" from **A2** is the semantic word class of *rickets*, one of the question terms. These features capture the correspondence between semantic word classes in the question and answer candidates.

3.3 Sentiment Analysis

Sentiment analysis (SA) features are classified into word-polarity and phrase-polarity features. We use *opinion extraction tool*[8] and sentiment orientation lexicon in the tool for these features.

3.3.1 Opinion Extraction Tool

Opinion extraction tool is a software, the implementation of Nakagawa et al. (2010). It extracts linguistic expressions representing opinions (henceforth, we call them sentiment phrases) from a Japanese sentence and then identifies the polarity of these sentiment phrases using machine learning techniques. For example, *rickets occur* in **Q2** and *Deficiency of vitamin D can cause rickets* in **A2** can be identified as sentiment phrases with a negative polarity. The tool identifies sentiment phrases and their polarity by using polarities of words and dependency subtrees as evidence, where these polarities are given in a word polarity dictionary.

In this paper, we use a trained model and a word polarity dictionary (containing about 35,000 entries) distributed via the ALAGIN forum[9] for our sentiment analysis. Table 2 shows the performance of *opinion extraction tool*, precision (P), recall (R) and F-value (F), in this setting (reported in the Japanese homepage of this tool). In the evaluation of sentiment-phrase extraction, an extracted sentiment phrase is determined as correct if its head word is the same as one in the gold standard. Polarity classification is evaluated under the condition that all of the sentiment phrases are correctly extracted.

	P	R	F
Sentiment-phrase extraction	0.602	0.408	0.486
Polarity classification (pos.)	0.873	0.893	0.883
Polarity classification (neg.)	0.866	0.842	0.854

Table 2: The performance of *opinion extraction tool*

3.3.2 Word Polarity (SA@W)

Polarities of words are identified by simply looking up the word polarity dictionary of *opinion ex-*

[8] Available at http://alaginrc.nict.go.jp/opinion/index_e.html

[9] http://www.alagin.jp/index-e.html. Only the members of the ALAGIN forum can access these resources.

traction tool. Word polarity features are used for identifying associations between the polarity of words in a question and that in a correct answer. For example:

- **Q2**: Why does rickets (W^-) occur in children?
- **A2**: Deficiency (W^-) of vitamin D can cause rickets (W^-).

Here, W^- represents negative word polarities. We expect our classifier to learn from this question and answer pair that if a word with negative polarity appears in a question then its correct answer is likely to contain a negative polarity word as well.

SA@W1 and SA@W2 in Table 1 are sentiment analysis features from *word polarity n-grams*, which contain at least one word that has word polarities. We obtain these n-grams by converting all nouns in MSA1 n-grams into their word polarities through dictionary lookup. For example, from Q2 in the above example we extract "W^- occur" as a word polarity 2-gram. SA@W1 is concerned with *all* word polarity n-grams in questions and answer candidates. For SA@W2, we restrict word polarity n-grams from SA@W1 to those whose original n-gram include a question term.

Furthermore, word polarities are coupled with semantic word classes so that our classifier can identify meaningful combinations of both. For example, *deficiency* in **A2** can be represented as $W^-_{condition}$ by its respective semantic word class and word polarity, which allows for the representation of *undesirable conditions*. This in turn lets our system learn meaningful correlations between words expressing these kind of negative conditions and their connection to questions asking about diseases. SA@W3 and SA@W4 are features from this combination. They are defined in the same way as SA@W1 and SA@W2 except that word polarities are replaced with the combination of semantic word classes and word polarities. We call n-grams in SA@W3 and SA@W4 *joint (word) class-polarity n-grams*.

3.3.3 Phrase Polarity (SA@P)

Opinion extraction tool is applied to question and its answer candidate to identify sentiment phrases and their phrase-polarities. In preliminary tests we found that sentiment phrases do not help to identify correct answers if answer sentences including the sentiment phrases do not have any term from the question. So we restrict the target sentiment phrases to those acquired from sentences containing at least one question term. From these sentiment phrases we extract three categories of features.

First, SA@P1 and SA@P2 are features concerned with phrase-polarity agreement between sentiment phrases in a question and its answer candidate. We consider all possible pairs of sentiment phrases from the question and answer. If any such pair agrees in phrase-polarity, an indicator for the agreement and its polarity in the agreement become features SA@P1 and SA@P2, respectively.

Secondly, following the original hypothesis underlying this paper, we assume that sentiment phrases often represent the core part of the correct answer (e.g., A2 above) and it is important to express the content of the sentiment phrases in features. SA@P3 and SA@P4 were devised for this purpose. SA@P3 represents this sentiment phrase contents as n-grams of morphemes, words, and syntactic dependencies of sentiment phrases, together with their phrase-polarity. Furthermore, SA@P4 is the subset of SA@P3 n-grams restricted to those that include terms found in the question, and SA@P5 indicates the percentage of sentiment n-grams from the question that are found in a given answer candidate.

Finally, features SA@P6 through SA@P9 use semantic word classes to generalize the content features mentioned above. These features consist of word class n-grams and joint class-polarity n-grams taken from sentiment phrases, together with their phrase polarity. Similar to the definition of SA@P4, for SA@P7 and SA@P9 we restrict ourselves to n-grams containing a question term. SA@P10 represents the semantic content of two sentiment phrases with the same sentiment orientation (one from a question and the other from an answer candidate) using word class n-grams, together with the phrase-polarity in agreement.

4 Test Set

We prepared three sets of why-questions (QS1, QS2 and QS3) and used these questions to build two test sets for our experiments.

Why-questions in **QS1** are taken from the Japanese version of *Yahoo! Answers* (called *Yahoo! Chiebukuro*)[10]. We automatically extracted

[10] We used "Yahoo! Chiebukuro Data (2nd edition)" which is

questions consisting of a single sentence and containing the interrogative *naze* (*why*), and our annotators verified that these questions are meaningful without further context. For example, they discarded questions like "Why doesn't the WBC (world boxing council) make an objection to the WBC (World baseball classic)?" (the object of the objection is unclear) and "Why do minors trade at the auction even though it is disallowed by the rules" (information about which auction is not provided).

Because questions in *Yahoo! Answers* are aimed at human readers, users often "set the stage" by giving lots of background information about their question. This often leads to large stylistic differences between the questions in *Yahoo! Answers* and those typically posed to a QA system. We therefore created a second set of why-questions, **QS2**, whose style should be more appropriate for a QA system (examples showing these differences are given in the supplementary materials of this paper). Six human annotators (not the authors) were asked to create why-questions in their own words, keeping in mind that the questions they create are for a QA system. In addition, the annotators were asked to verify on the Web that the questions they created ask about some real event or phenomena. For example, a question like "Why does Mars appear blue?" is disallowed in QS2 because "Mars appears blue" is false. Note that the correct answer to these questions does not have to be either in our target corpus or in real-world Web texts. These two sets of why-questions, QS1 and QS2, are used to build a test set for evaluating our proposed method.

Finally, **QS3** contains why-questions that have at least one answer in our target corpus (600 million Japanese Web page corpus). For creating such why-questions, four human annotators (not the authors) were given a text passage composed of three continuous sentences and asked to locate the reasons for some event as described in this passage. Then they created a why-question for which the description is a correct answer. Because randomly selected passages from our target corpus have little chance of generating good why-questions we extracted passages from our target corpus that include at least one of the clue terms used in our answer retrieval step (i.e. *riyuu* (*reason*), *genin* (*cause*), or *youin* (*cause*)). This setting may not necessarily reflect a "real world" distribution of why-questions, in which ideally a wide range of people ask questions that may or may not have an answer in our corpus. However, QS3 allows us to evaluate our method under the *idealized* conditions where we have a perfect answer retrieval module whose answer candidates always contain at least one correct answer (the source passage used for creating the why-question). This setting allows us to estimate the *ideal-case* performance of our method. Under these circumstances we found that our method achieves almost 65% precision in P@1, which suggests that it can potentially perform with high precision if the answer candidates given by the answer retrieval module contain at least one correct answer. This is the main purpose of QS3. Additionally, we use QS3 for building training data, to check whether questions that do not reflect the real-world distribution of why-questions are useful for improving the system's performance on "real-world" questions (see Section 5.1).

In addition, we checked QS1, QS2 and QS3 for questions having the same topic, to avoid the possibility that the distribution of questions is biased towards certain topics. We manually extracted the questions' topic words and randomly selected a single representative question from all questions with the same topic. For example, "Why does Twitter only allow 140 characters?" and "Why is Twitter so popular?" both have as topic *Twitter*. In the end we obtained 250 questions in QS1, 250 questions in QS2 and 350 questions in QS3.

For evaluation we prepared two test sets, Set1 and Set2. *Set1* contains question-answer pairs whose questions are taken from QS1 and QS2. In our experiment, we evaluate systems with 10-fold cross validation on Set1. *Set2* has question-answer pairs whose questions are from QS3. Set2 is mainly used for estimating estimate the ideal-case performance of our method with a perfect answer retrieval module. Furthermore Set2 is used as additional training data in evaluating systems with 10-fold cross validation on Set1. We used our answer retrieval system to obtain the top-20 answer candidates for each question, and all question-answer (candidate) pairs were checked by three annotators, where their inter-rater agreement (Fleiss' kappa) was 0.634, indicating substantial agreement. Finally, correct answers to each question were determined by majority vote.

provided by Yahoo Japan Corporation and contains 16 million questions asked from April, 2004 to April 2009.

Q1:二酸化炭素などの温室効果ガスが増えると海面水位が上昇するといわれているのはなぜですか？ (Why does the increase of greenhouse gases such as carbon dioxide in the atmosphere lead to a rise of ocean level?)
A1: .. 化石燃料等の使用が増えるにつれて、温室効果ガスが大 気中に大量に放出され、その濃度が増加し、大気中に吸収される 熱が増えたことにより、地球規模での気温上昇が進行しています。 これが地球温暖化です。 ... 温暖化による海水膨張と両極の氷解で、海面が平均９～８８ｃｍ上昇すると警告しています。 (The burning of fossil fuels contributes to the increase of atmospheric concentrations of greenhouse gases and this makes the atmosphere absorb more thermal radiation. As a result, Earth's average surface temperature increases. This is global warming. ... There are warnings that the increase of sea water and melting of polar ice due to the global warming may cause sea-surface height to rise by 9–88 cm on average.
Q2:ヘモグロビンが不足すると体が酸素不足になるのはなぜですか？ (Why does hemoglobin deficiency cause lack of oxygen in the human body?)
A2:... ヘモグロビンは酸素を体の中に運び、いらなくなった二酸化炭素を持ち帰り、肺から外に出すなど重要な働きをしています。もし鉄分が不足してヘモグロビンが少ししか作られないと、全身に運ばれる酸素の量が減少し、カラダが酸素不足になります。.. (... Hemoglobin has an important role in the human body of carrying oxygen to the organs and transferring carbon dioxide back to the lungs, to be dispensed from the organism. If the amount of hemoglobin produced by the body is insufficient due to iron deficiency, the amount of oxygen delivered throughout the body decreases, causing oxygen deficiency. ...)

Table 3: Correct question-answer pairs in our test set

Table 3 shows a sample of correct question-answer pairs in our test set. Please see the supplementary materials of this paper for more examples.

Note that word and phrase polarities are not considered by the annotators in building our test sets and these polarities are automatically identified using a word polarity dictionary and *opinion extraction tool*. We confirmed that about 35% of questions and 40% of answer candidates had at least one sentiment phrase by *opinion extraction tool*, and about 45% of questions and 85% of answer candidates contained at least one word having polarity by a word polarity dictionary.

5 Experiments

We use TinySVM[11] with a linear kernel for training our re-ranker. Evaluation was done by P@1 (Precision of the top answer) and MAP (Mean Average Precision). P@1 measures how many questions have a correct top answer candidate. MAP, widely used in evaluation of IR systems, measures the overall quality of the top-n answer candidates (n=20 in this experiment) using the formula:

$$MAP = \frac{1}{|Q|} \sum_{q \in Q} \frac{\sum_{k=1}^{n} (Prec(k) \times rel(k))}{|A_q|} \quad (3)$$

Here Q is a set of why-questions, A_q is a set of correct answers to why-question $q \in Q$, $Prec(k)$ is the precision at cut-off k in the top-n answer candidates, $rel(k)$ is an indicator, 1 if the item at rank k is a correct answer in A_q, 0 otherwise.

We evaluated all systems using 10-fold cross validation in two ways. In the first setting we performed 10-fold cross validation on Set1. Set1 consists of 10,000 question-answer pairs (500 questions with their 20 answer candidates), and was partitioned into 10 subsamples such that the questions in one subsample do not overlap with those of the other subsamples. 9 subsamples (9,000 question-answer pairs) were used as training data and the remaining subsample (1,000 question-answer pairs) was retained as test data. This experiment is called **CV(Set1)**. It shows the effect of answer re-ranking when evaluating our proposed method with training data built with real world why-questions alone. In the second setting, we used the same 10 subsamples of Set1 as in CV(Set1) and exploited Set2 (composed of 7,000 question-answer pairs) as additional training data for 10-fold cross validation. As a result, in each fold 16,000 question-answer pairs (9,000 from Set1 and 7,000 from Set2) were used as training data for re-rankers, and all systems were evaluated on the remaining 1,000 question-answer pair subsample from Set1. We call this setting **CV(Set1+Set2)**. It verifies whether training data that does not necessarily reflect a real-world distribution of why-questions can improve why-QA performance on real-world questions.

[11] http://chasen.org/∼taku/software/TinySVM/

5.1 Results

Table 4 shows the evaluation results of six different systems. For each system, we represent the performance in P@1 and MAP. B-QA is a system of our answer retrieval and the other five re-rank top-20 answer candidates using their own re-ranker.

B-QA: our answer retrieval system, our implementation of Murata et al. (2007).

B-Ranker: a system that has a re-ranker trained with morphological and syntactic analysis (MSA) features alone.

System	CV(Set1)		CV(Set1+Set2)	
	P@1	MAP	P@1	MAP
B-QA	0.222 (0.368)	0.270 (0.447)	0.222 (0.368)	0.270 (0.447)
B-Ranker	0.256 (0.424)	0.319 (0.528)	0.274 (0.454)	0.323 (0.535)
B-Ranker+CR	0.262 (0.434)	0.319 (0.528)	0.278 (0.460)	0.325 (0.538)
B-Ranker+WN	0.257 (0.425)	0.320 (0.530)	0.275 (0.455)	0.325 (0.538)
Proposed	**0.336 (0.56)**	**0.377 (0.624)**	**0.374 (0.619)**	**0.391 (0.647)**
UpperBound	0.604 (1)	0.604 (1)	0.604 (1)	0.604 (1)

Table 4: Comparison of systems

B-Ranker+CR: a system has a re-ranker trained with our MSA features and the causal relation (CR) features used in Higashinaka and Isozaki (2008). The CR features include binary features indicating whether an answer candidate contains a causal relation pattern, which causal relation pattern the answer candidate has, and whether the question-answer pair contains a causal relation instance — cause in the answer, effect in the question). We acquired causal relation instances from our target corpus using the method from (De Saeger et al., 2009), and exploited the top-100,000 causal relation instances and the patterns that extracted them for CR features. Note that these CR features are introduced only for comparing our semantic features with ones in Higashinaka and Isozaki (2008) and they are not a part of our method.

B-Ranker+WN: its re-ranker is trained with our MSA features and the WordNet features in Verberne et al. (2010). The WordNet features include the percentage of the question terms and their synonyms in WordNet synsets found in an answer candidate and the semantic relatedness score between a question and its answer candidate, the average of the concept similarity between each question term and all of the answer terms by *WordNet::Similarity* (Pedersen et al., 2004). We used the Japanese WordNet 1.1 (Bond et al., 2009) for these WordNet features. Note that the Japanese WordNet 1.1 has 93,834 Japanese words linked to 57,238 WordNet synsets, while the English WordNet 3.0 covers 155,287 words linked to 117,659 synsets. Due to this lower coverage, the WordNet features in Japanese may have a less power for finding a correct answer than those in English used in Verberne et al. (2010).

Proposed: our proposed method. All of the MSA, SWC and SA features are used for training our re-ranker.

UpperBound: a system that ranks all n correct answers as the top n results of the 20 answer candidates if there are any. This indicates the performance upperbound in this experiment. The relative performance of each system compared to *UpperBound* is shown in parentheses.

The proposed method achieved the best performance both in CV(Set1) and CV(Set1+Set2). Our method shows a significant improvement (11.4–15.2% in P@1 and 10.7–12.1% in MAP) over our answer retrieval method, B-QA. Its improvement over B-Ranker, B-Ranker+CR and B-Ranker+WN (7.6–10% in P@1 and 5.7–6.6% in MAP) shows the effectiveness of our proposed feature set over the features used in previous works. Both B-Ranker+CR and B-Ranker+WN did not show significant performance improvement over B-Ranker. At least in our setting, the causal relation and WordNet features did not prove effective. The performance gap between B-Ranker and B-QA (3.4–5.2% in P@1 and 4.9–5.3% in MAP) suggests the effectiveness of re-ranking. All systems consistently show better performance in CV(Set1+Set2) than CV(Set1). This suggests that training data built with why-questions that does not reflect real-world distribution of why-questions is useful in training re-rankers.

We investigate the contribution of each type of features to the performance by removing one feature set from the all feature sets in training our re-ranker. In this experiment, we split SA into $SA@W$ (features expressing words and their polarity) and $SA@P$ (features expressing phrases and their polarity) to investigate their contribution either. The results are summarized in Table 5.

In Table 5, **MSA+SWC+SA** represents our proposed method using all feature sets. The performance gap between **MSA+SWC+SA** and the others confirms that all the features contributed to a higher

System	CV(Set1)		CV(Set1+Set2)	
	P@1	MAP	P@1	MAP
SWC+SA	0.302	0.324	0.314	0.332
MSA+SWC	0.308	0.349	0.318	0.358
MSA+SA	0.300	0.352	0.314	0.364
MSA+SWC+SA@W	0.312	0.358	0.325	0.365
MSA+SWC+SA@P	0.323	0.369	0.358	0.384
MSA+SWC+SA	**0.336**	**0.377**	**0.374**	**0.391**
UpperBound	0.604	0.604	0.604	0.604

Table 5: Evaluation with different combination of feature sets used in training our re-ranker

performance. The significant performance improvement by SA (features from sentiment analysis) and SWC (features from semantic word classes) (The gap between **MSA+SWC+SA** and **MSA+SWC** was 2.8–6% and that between **MSA+SWC+SA** and **MSA+SA** was 3.6%–6% in P@1) supports the hypothesis for sentiment analysis and semantic word classes in this paper.

Though the performance gap between **MSA+SWC+SA** and **MSA+SWC+SA@P** (1.3%–1.6% in P@1) shows that SA@W is useful in training our re-ranker, we found that **MSA+SWC+SA@W** made only 0.4–0.7% improvement over **MSA+SWC**. We believe that this is mainly because SA@W and SWC are based on semantic and sentiment information at the word level, and these often capture a similar type of information. For instance, disease names that are grouped together into one class in SWC are typically classified as negative in SA@W. Therefore the similarity in the information provided by SA@W and SWC causes a classifier trained with both of these features to obtain only a minor improvement over a classifier using only one of the features.

To estimate the *ideal-case* performance of our proposed method, we made another experiment by using Set1 as training data for our re-ranker and Set2 as test data for evaluating our proposed method. Here, we assume a perfect answer retrieval module that adds the source passage that was used for generating the original why-question in Set2 as a correct answer to the set of existing answer candidates, giving 21 answer candidates. The performance of our method in this setting was 64.8% in P@1 and 66.6% in MAP. This evaluation result suggests that our re-ranker can potentially perform with high precision when at least one correct answer in answer candidates is given by the answer retrieval module.

6 Related Work

In the QA literature, Higashinaka and Isozaki (2008), Verberne et al. (2010), and Surdeanu et al. (2011) are closest to our work. The first two deal with why-questions, the last with how-questions. Similar to our method, they use machine learning techniques to re-rank answer candidates to non-factoid questions based on various combinations of syntactic, semantic and other statistical features such as the density and frequency of question terms in the answer candidates and patterns for causal relations in the answer candidates. Especially for why-QA, Higashinaka and Isozaki (2008) used causal relation features and Verberne et al. (2010) exploited WordNet features as a kind of semantic features for training their re-ranker, where we used these features, respectively, for B-Ranker+CR and B-Ranker+WN in our experiment.

Our work differs from the above approaches in that we propose semantic word classes and sentiment analysis as a new type of semantic features, and show their usefulness in why-QA. Sentiment analysis has been used before on the slightly unusual task of opinion question answering, where the system is asked to answer subjective opinion questions (Stoyanov et al., 2005; Dang, 2008; Li et al., 2009). To the best of our knowledge though, no previous work has systematically explored the use of sentiment analysis in a general QA setting beyond opinion questions.

7 Conclusion

In this paper, we have explored the utility of sentiment analysis and semantic word classes for ranking answer candidates to why-questions. We proposed a set of semantic features that exploit sentiment analysis and semantic word classes obtained from large-scale noun clustering, and used them to train an answer candidate re-ranker. Through a series of experiments on 850 why-questions, we showed that the proposed semantic features were effective in identifying correct answers, and our proposed method obtained more than 15% improvement in precision of its top answer (P@1) over our baseline, a state-of-the-art IR based QA system. We plan to use new semantic knowledge such as semantic orientation, excitatory or inhibitory, proposed in Hashimoto et al. (2012) for improving why-QA.

References

Francis Bond, Hitoshi Isahara, Sanae Fujita, Kiyotaka Uchimoto, Takayuki Kuribayashi, and Kyoko Kanzaki. 2009. Enhancing the japanese wordnet. In *Proceedings of the 7th Workshop on Asian Language Resources*, pages 1–8.

Hoa Tran Dang. 2008. Overview of the TAC 2008 opinion question answering and summarization tasks. In *Proc. TAC 2008*.

Stijn De Saeger, Kentaro Torisawa, Jun'ichi Kazama, Kow Kuroda, and Masaki Murata. 2009. Large scale relation acquisition using class dependent patterns. In *Proc. of ICDM 2009*, pages 764–769.

David A. Ferrucci, Eric W. Brown, Jennifer Chu-Carroll, James Fan, David Gondek, Aditya Kalyanpur, Adam Lally, J. William Murdock, Eric Nyberg, John M. Prager, Nico Schlaefer, and Christopher A. Welty. 2010. Building Watson: An overview of the DeepQA project. *AI Magazine*, 31(3):59–79.

Junichi Fukumoto, Tsuneaki Kato, Fumito Masui, and Tsunenori Mori. 2007. An overview of the 4th question answering challenge (QAC-4) at NTCIR workshop 6. In *Proc. of NTCIR-6*.

Zellig Harris. 1954. Distributional structure. *Word*, 10(23):146–162.

Chikara Hashimoto, Kentaro Torisawa, Stijn De Saeger, Jong-Hoon Oh, and Jun'ichi Kazama. 2012. Excitatory or inhibitory: A new semantic orientation extracts contradiction and causality from the web. In *Proceedings of EMNLP-CoNLL 2012*.

Ryuichiro Higashinaka and Hideki Isozaki. 2008. Corpus-based question answering for why-questions. In *Proc. of IJCNLP*, pages 418–425.

Thomas Hofmann. 1999. Probabilistic latent semantic indexing. In *Proc. of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '99, pages 50–57.

Jun'ichi Kazama and Kentaro Torisawa. 2008. Inducing gazetteers for named entity recognition by large-scale clustering of dependency relations. In *Proc. of ACL-08: HLT*, pages 407–415.

Fangtao Li, Yang Tang, Minlie Huang, and Xiaoyan Zhu. 2009. Answering opinion questions with random walks on graphs. In *Proc. of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2 - Volume 2*, pages 737–745.

Masaki Murata, Sachiyo Tsukawaki, Toshiyuki Kanamaru, Qing Ma, and Hitoshi Isahara. 2007. A system for answering non-factoid Japanese questions by using passage retrieval weighted based on type of answer. In *Proc. of NTCIR-6*.

Tetsuji Nakagawa, Kentaro Inui, and Sadao Kurohashi. 2010. Dependency tree-based sentiment classification using CRFs with hidden variables. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 786–794, Los Angeles, California, June. Association for Computational Linguistics.

Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. 2002. Thumbs up? Sentiment classification using machine learning techniques. In *Proc. of the 2002 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 79–86.

Ted Pedersen, Siddharth Patwardhan, and Jason Michelizzi. 2004. WordNet::Similarity: measuring the relatedness of concepts. In *Demonstration Papers at HLT-NAACL 2004*, HLT-NAACL–Demonstrations '04, pages 38–41.

Anselmo Peñas, Eduard H. Hovy, Pamela Forner, Álvaro Rodrigo, Richard F. E. Sutcliffe, Corina Forascu, and Caroline Sporleder. 2011. Overview of QA4MRE at CLEF 2011: Question answering for machine reading evaluation. In *CLEF*.

Veselin Stoyanov, Claire Cardie, and Janyce Wiebe. 2005. Multi-perspective question answering using the opqa corpus. In *Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*, HLT '05, pages 923–930.

Mihai Surdeanu, Massimiliano Ciaramita, and Hugo Zaragoza. 2011. Learning to rank answers to non-factoid questions from web collections. *Computational Linguistics*, 37(2):351–383.

Peter D. Turney. 2002. Thumbs up or thumbs down?: semantic orientation applied to unsupervised classification of reviews. In *Proc. of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, pages 417–424.

Vladimir N. Vapnik. 1995. *The nature of statistical learning theory*. Springer-Verlag New York, Inc., New York, NY, USA.

Suzan Verberne, Lou Boves, Nelleke Oostdijk, and Peter-Arno Coppen. 2007. Evaluating discourse-based answer extraction for *why*-question answering. In *SIGIR*, pages 735–736.

Suzan Verberne, Lou Boves, Nelleke Oostdijk, and Peter-Arno Coppen. 2010. What is not in the bag of words for why-QA? *Computational Linguistics*, 36:229–245.

Ellen M. Voorhees. 2004. Overview of the TREC 2004 question answering track. In *TREC*.