Information Sciences 261 (2014) 101–115

Contents lists available at ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

Discovering high quality answers in community question answering archives using a hierarchy of classifiers

Hapnes Toba [a,1], Zhao-Yan Ming [b,*], Mirna Adriani [a], Tat-Seng Chua [b]

[a] Information Retrieval Laboratory, Faculty of Computer Science, Universitas Indonesia, Depok 16424, West Java, Indonesia
[b] Lab of Media Search, School of Computing, National University of Singapore, 117417, Singapore

ARTICLE INFO

Article history:
Received 21 February 2013
Received in revised form 10 September 2013
Accepted 26 October 2013
Available online 1 November 2013

Keywords:
Question answering system
User generated content
Answer quality
Question type analysis
Hierarchical supervised learning

ABSTRACT

In community-based question answering (CQA) services where answers are generated by human, users may expect better answers than an automatic question answering system. However, in some cases, the user generated answers provided by CQA archives are not always of high quality. Most existing works on answer quality prediction use the same model for all answers, despite the fact that each answer is intrinsically different. However, modeling each individual QA pair differently is not feasible in practice. To balance between efficiency and accuracy, we propose a hybrid hierarchy-of-classifiers framework to model the QA pairs. First, we analyze the question type to guide the selection of the right answer quality model. Second, we use the information from question analysis to predict the expected answer features and train the type-based quality classifiers to hierarchically aggregate an overall answer quality score. We also propose a number of novel features that are effective in distinguishing the quality of answers. We tested the framework on a dataset of about 50 thousand QA pairs from Yahoo! Answer. The results show that our proposed framework is effective in identifying high quality answers. Moreover, further analysis reveals the ability of our framework to classify low quality answers more accurately than a single classifier approach.

© 2013 Elsevier Inc. All rights reserved.

1. Introduction

Since it was introduced in the 1960s [17,28,55,56], the task of question answering system (QAS) has always been at the forefront of technology advances. In its early stage of development, QAS is restricted to structured domains with limited natural language processing [20,35,25]. Along with the advances in the fields of information retrieval, computational linguistics, and Internet technology, research on QAS were broadened into unstructured textual documents in open domains, and with collaborative users [15,23]. Evaluation forums for QAS, such as TREC [15] and CLEF [41], have steered the development of QAS into an established and large-scale research methodologies and evaluations. Despite the advances that QAS technologies have achieved in TREC and CLEF evaluation forums, existing systems are mostly used on document-based (reputed) resources, such as the newspapers and web pages. In a document-based QAS, the quality of answer candidates provided in the retrieval results are typically of good quality.

Recent advances in Internet technologies offer more freedom for the users in the ways they express their opinions and interact with one another. Web applications, such as Yahoo! Answers[2] (Y!A), Twitter, Facebook, and Flickr are good examples of how people are connected to each other and share their interests, opinions, knowledge, and social activities. In comparison to

* Corresponding author. Tel.: +65 94590298.
E-mail address: mingzhaoyan@nus.edu.sg (Z.-Y. Ming).

[1] This work was done when the first author was a research intern at NUS.

[2] http://answers.yahoo.com/.

0020-0255/$ - see front matter © 2013 Elsevier Inc. All rights reserved.
http://dx.doi.org/10.1016/j.ins.2013.10.030

a traditional QAS, which use documents as information sources, a community-based question answering system (CQA), such as the Y!A, relies on users to provide the answers (sometimes also called the user-generated content (UGC)) [12].

In CQA services, each authorized user can post questions or answer other users' questions. In this way, each question might have a variety of different answers. Users can also search for related questions to their needs and expect to find good answers in the system. Users would expect better answers from a CQA as compared to a traditional QAS since the answers are generated by human [8,22,53,11]. However, there is a possibility that the answers provided by a CQA are of poor or low quality[3] quality [2]. Some examples of low quality answers are given below:

- Case 1: The answers only contain URL links to other information sources without any related explanations. For example, **Question** [Category: History] [Question Type: Procedure]: How did the Holocaust really happen? **Answer**: http://www.ushmm.org/wlc/article.php?lang=en&ModuleId=10005143.
- Case 2: The answers show opinions or sentiments of other unrelated issues. For example, **Question** [Theatre & Acting] [Factoid]: Which Shakespearean play has the most parts for women? **Answer**: will be sure not to spam your answer box, I'll keep a look out for other spammers too. All the best:)
- Case 3: The answers give only a short reasoning or fact with limited external proofs or evidences. For example, **Question** [Books & Authors] [Opinion]: Do you think William S. Burroughs was heavily influenced by James Joyce? **Answer**: I've never read any Burroughs, but a guy I work with is a big fan and told me James Joyce was a huge influence on him, Naked Lunch in particular.
- Case 4: The answers are not properly written or use informal writing style. For example, **Question** [Fashion & Accessories] [YesNo]: Men in ugg boots- yay or nay? **Answer**: BOOOO

It is common to analyze the quality of the answer provided by CQA through its textual representation features [6], such as the length of a question, length of an answer, overlapped words between a question and its answer, length ratio between a question and its answer. Another common features used in quality analysis is to utilize popularity and social interaction measures [6,24,47,29], such as the number of best answers assigned by users, user or editor recommendations, quality ratings, and answerer's acceptance ratio.

Most recent studies in CQA answer quality analysis utilize user information as the main indicators to predict the answer quality. The links between users and their related questions and answers are mostly used to compute the quality of an answer [1]. Based on user information, many approaches try to learn the rank of answer retrieval results based on users' votings [7,31,43,49]. Other approaches rely on the assumption that similar questions provide similar answers. Such CQA systems thus try to retrieve similar questions in CQA and use the answer of the most similar question as the final answer, such as in [5,14,38,52,61]. A common problem with existing approaches that utilize user metadata is that such information is often unavailable in real world application. For example, when an answer is newly posted, no rating is available. When a user's account is banned by the administrator or the user has closed his/her account, the user metadata cannot be retrieved. Besides the difficulty of obtaining complete user metadata, another problem is that active users who are usually associated with good answers may not be able to answer all the questions. Many questions are answered by common users whose metadata may not be indicative of the quality of their answers. On the other hand, those answers provided by active users can be easily distinguished by solely looking at the user information, they also exhibit some intrinsic features that can be captured by text analysis.

Moreover, although many recent studies [1,36,38,52] have taken the question and answer features into consideration, they do not identify how a question should be answered. In practice, different types of questions often require different approaches of answering them. For example, a factoid-based question requires specific nouns as the answer, while a procedure question requires some explanation or references in the answer. In other words, we need to find out which features will be useful to answer a certain type of question. Thus, after classifying the question type, we can further measure the quality of its answers with higher confidence by utilizing the influential features in the specific question types.

In a broader perspective, we try to explore a large spectrum of intrinsic information about the question and answer pair as a way to determine the quality of answers. To justify our choice of using intrinsic features only instead of involving metadata features, we did a preliminary experiment on a sample dataset with complete user information such as the percentage of best answers, number of best answers, number of questions asked, the achieved level, and the gained points. We found that the use of intrinsic features alone could achieve comparable performance to the systems that either utilize the metadata, or combining metadata with intrinsic features. We conjecture that given the well designed set of intrinsic features and classification framework, the quality indicators derived from metadata are redundant in conjunction with those from intrinsic features. Based on this and the fact that metadata is sparse in our dataset and also real applications, we thus proceed with designing and experimenting the system without metadata. Our main strategy is to identify the question type before the evaluation of the quality of the answers based on the predicted question type. Specifically, we use the information from question analysis component and trained quality classifier to hierarchically aggregate an overall answer quality score. The research questions which we try to answer in this study are as follows:

[3] We use the terms *high/low* and *good/bad* quality interchangeably.

- What are the intrinsic criteria of high quality answers in CQA?
- How can we model and assess the quality criteria for answers of varying types and quality?
- How can we integrate the question type analysis and answer quality information in a unified framework?

Based on the research questions above, we conduct comprehensive experiments on answer quality prediction on CQA data. We use a real world dataset consisting of about 50 thousand question answer pairs from Yahoo!Answer in 14 question categories. The main contributions of this paper are as follows:

1. We propose a novel hierarchy of classifiers framework for accurate and efficient CQA quality prediction. We design a broad spectrum of features which are universally applicable even when an answer does not have metadata or is newly posted. The performance is even better than systems that make extensive use of metadata in identifying both high and low quality answers.
2. We conduct a large scale comprehensive study on answer quality prediction and perform deep analysis on the influence of the proposed approaches on different question types and domains.

In the rest of this paper, Section 2 summarizes some recent related studies on UGC answer quality prediction, while Section 3 gives the detail of our methodology. Our experimental set-up, results, and analysis are presented in Section 4. Finally, Section 5 contains our conclusion and future work.

2. Related work

CQA services, such as Y!A, have emerged as popular sources of information. In CQA, users can post questions and expect quick and accurate answers from other users or experts on any topic. However, in CQA systems, the answers vary in terms of quality as they are provided by a broad range of users [1,50]. We categorize recent studies in answer quality estimation in three main approaches: link analysis based approaches such as [1,50], log metadata analysis based approaches such as [31,47], and ranking and classification methods such as [21,52].

2.1. Link analysis

Such methods employ link-based ranking algorithms, such as PageRank [39] and HITS [27], to analyze the user graphs that consist of users as vertices and the ask/answer links between them as edges. ExpertiseRank [60], an extension of PageRank gives additional score to users who were able to provide answers to other people with certain degree of expertise. The principal graph-based framework [1] exploits the interactions between users around the questions and answers. It is modeled as a tri-partite graph with stochastic gradient boosting. The use of such interaction graphs enables the system to assign higher probability to more active users who returned better quality answers. An alternative to enhancing the interaction graphs is by adding influential feature sets in the answer quality [1]. It is recommended that the contextual features, such as the *n*-gram, and answer length, are among the significant features for answer quality prediction. The quality-aware framework in [50] exploits the textual relevance model and the way users answer questions, and concludes that the collaborative users, *i.e.*, users who have many related question–answering activities, tend to produce higher quality (high relevance) answers. Similar to the interaction graphs method, the performance of the method in [50] relies more on the links between users rather than the textual features.

Despite the success of user linking based methods, it is equally important to exploit answers to questions which are not popular, as such questions have lower chance of being answered by collaborative users. Moreover, for questions that can be answered in many ways, more than one relevant answers are expected, thus some answers from new users and not so active users should be considered valuable. In both cases, there is a need to evaluate the quality of answers based on the intrinsic textual features. In this study, we thus try to generalize the variations in which a question should be answered. Such generalization may also be useful for other types of user-generated-content for knowledge building.

2.2. Log analysis

In the log analysis approach, answers quality is usually measured by using the *past performance* of users for similar questions [9,18,31,47]. User log, such as session/login duration, and question/answer behaviors are important sources for examining how answerers choose their questions. Log analysis based approaches aimed to identify users with same interests [47] or highly reputed users [40,51]. By analyzing users' behaviors, they postulate that reputed users in a specific topic will be able to generate high quality answers in the same topic. Based on this hypothesis, subjective features such as askers' and answerers' profiles, are used in a learning-to-rank framework [31] and other classical learning method [47], to predict whether a given answer will be selected by the asker as the best answer.

One of the main advantages of using user log data is that a small amount of high quality content can be easier identified in the community [3,16]. However, given the sparsity of active users and the large numbers of questions, the log analysis based method may have low coverage of questions that it can be applied to. Given that a question might be answered by a wide range of users, the log based method may also be subjected to the unavailability of user metadata. Therefore, an adaptive method should be able to account for the huge variety in the quality of answers whenever they are provided by an active user or a new user.

2.3. Ranking and classification methods

Ranking and classification methods have been applied to text analysis tasks [57,4,37,42], such as distinguish high quality content [38,49]. Recent works apply the analogical principle which hypothesizes that similar questions would attract similar answers. An example is the Bayesian analogical reasoning [52] method. It performs answers re-ranking by utilizing a set of statistical and social interaction features from the question–answer pairs. In another approach by [21], the user's best answer model is combined with a query expansion and re-ranking approach in finding relevant question. In these methods, the answer for a specific question is evaluated over a set of answer candidates which have been generated by social interaction. The main advantage of modeling quality as question–answer pairs is that one might combine various features for modeling the relevance between a question and its answer candidates. One of the main problems of using such relevance models is the inherent noise of data, as reported in [21]. We conjecture that this may be rooted in the nature of relevance models: they compare a given question (query) to some answer candidates based on the overlapping contents only.

Since the quality of answers are sometimes affected by the quality of the question, early studies [1,14,38] in CQA which separate the question and answer models. In the same spirit, our research tries to first classify the category of question type. Unlike [1] which uses users-questions link features, linguistic features [38], or pattern-based question detection [14], we use the textual features exclusively. Combination of models has been found to be effective in a number of retrieval tasks [58,59] or in multi-classification task such as the question type classification [30,45]. Following these related research findings, we propose to exploit the hierarchical classification approach in learning the question types and answer quality in connected layers.

3. Methodology

3.1. Motivation

Most existing work on answer quality prediction use the same model for all types of answers, despite the fact that every question–answer pair is intrinsically different. However, attempting to model each individual question–answer pair differently will move towards to another extreme which is not feasible in practice. In this work, we propose a compromise framework which first analyzes the content of the question to guide the selection of a right model to be used for answer quality characterization, before performing the answer quality analysis.

We describe the quality model as an aggregated score of the probability functions of the question types and the answer quality:

$$score(Q, A) = op(P(QuestionType|Q) \cdot P(Quality|QuestionType, A)) \quad (1)$$

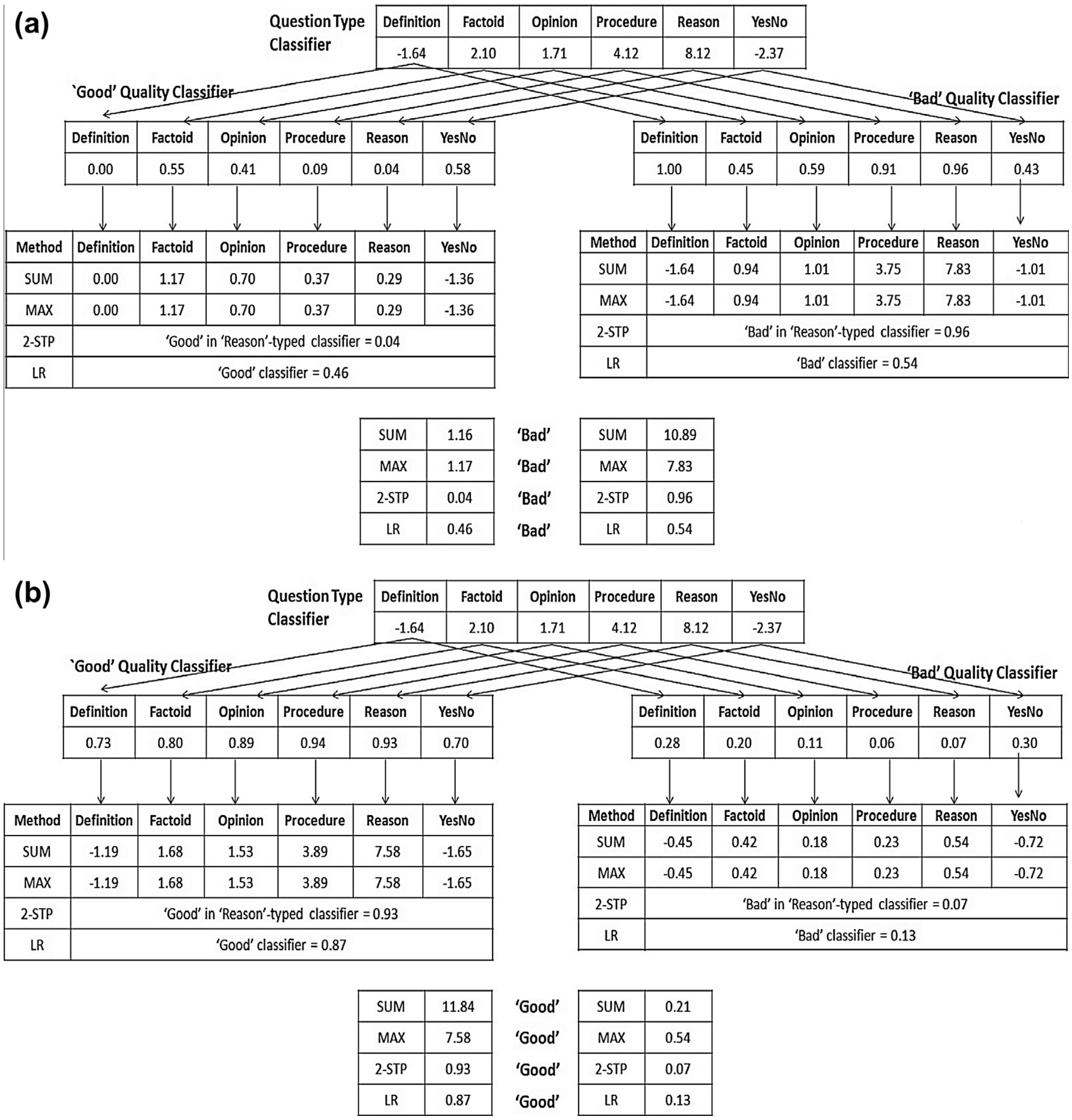
where Q is a question, A is an answer, and $op(\cdot)$ is the decision operator of the probability function as we will describe in the following section.

3.2. Hierarchy of classifiers framework

The use of question analysis or query classification has been found to be effective in information retrieval and question answering. In the same spirit, under the hierarchical framework we assume that each question type has a number of specific features that indicate how a question should be answered. Fig. 1 presents an example of the proposed framework. For instance, a *Reason*-based question is expected to have some reasoning in the answer using specific words such as: *because* ... and *in that reason* .... The main problem of a typical question analysis component is that some features that appear across a range of question types are of low discriminative power. For instance, a question which uses *What* question word could be classified as *Factoid*, such as: "*What is considered as the best guitar in the world?*", or as *Opinion*, such as: "*In your opinion, what is the best guitar in the world?*".

We define a decision operator $op(\cdot)$ as a manner to handle the obscurity of the answer quality based on a range of question type typicality. This operator acts as a controller in the way we combine the classifiers. We follow the assumption that the hierarchical classifiers, *i.e.*, the question type and the answer quality, are independent of each other. This independence assumption is important to make sure that each classifier does not influence the performance of the other classifier.

Our basic algorithm to compute the quality model is given in Algorithm 1. We compute the quality as the maximum of the sum-product of the question type and the quality probabilities. In this way we try to find which quality expectation is maximized after the aggregation of the quality in each question type. The question type classifiers include the following types: *Definition, Factoid, Opinion, Procedure, Reason* and *YesNo*; and the quality classifiers produce two probabilities each: the likelihood of *Good-quality* or *Bad-quality*.



(a)

Question Type Classifier	Definition	Factoid	Opinion	Procedure	Reason	YesNo
	-1.64	2.10	1.71	4.12	8.12	-2.37

`Good' Quality Classifier	Definition	Factoid	Opinion	Procedure	Reason	YesNo
	0.00	0.55	0.41	0.09	0.04	0.58

'Bad' Quality Classifier	Definition	Factoid	Opinion	Procedure	Reason	YesNo
	1.00	0.45	0.59	0.91	0.96	0.43

Method	Definition	Factoid	Opinion	Procedure	Reason	YesNo
SUM	0.00	1.17	0.70	0.37	0.29	-1.36
MAX	0.00	1.17	0.70	0.37	0.29	-1.36
2-STP	'Good' in 'Reason'-typed classifier = 0.04					
LR	'Good' classifier = 0.46					

Method	Definition	Factoid	Opinion	Procedure	Reason	YesNo
SUM	-1.64	0.94	1.01	3.75	7.83	-1.01
MAX	-1.64	0.94	1.01	3.75	7.83	-1.01
2-STP	'Bad' in 'Reason'-typed classifier = 0.96					
LR	'Bad' classifier = 0.54					

SUM	1.16	'Bad'	SUM	10.89
MAX	1.17	'Bad'	MAX	7.83
2-STP	0.04	'Bad'	2-STP	0.96
LR	0.46	'Bad'	LR	0.54

(b)

Question Type Classifier	Definition	Factoid	Opinion	Procedure	Reason	YesNo
	-1.64	2.10	1.71	4.12	8.12	-2.37

`Good' Quality Classifier	Definition	Factoid	Opinion	Procedure	Reason	YesNo
	0.73	0.80	0.89	0.94	0.93	0.70

'Bad' Quality Classifier	Definition	Factoid	Opinion	Procedure	Reason	YesNo
	0.28	0.20	0.11	0.06	0.07	0.30

Method	Definition	Factoid	Opinion	Procedure	Reason	YesNo
SUM	-1.19	1.68	1.53	3.89	7.58	-1.65
MAX	-1.19	1.68	1.53	3.89	7.58	-1.65
2-STP	'Good' in 'Reason'-typed classifier = 0.93					
LR	'Good' classifier = 0.87					

Method	Definition	Factoid	Opinion	Procedure	Reason	YesNo
SUM	-0.45	0.42	0.18	0.23	0.54	-0.72
MAX	-0.45	0.42	0.18	0.23	0.54	-0.72
2-STP	'Bad' in 'Reason'-typed classifier = 0.07					
LR	'Bad' classifier = 0.13					

SUM	11.84	'Good'	SUM	0.21
MAX	7.58	'Good'	MAX	0.54
2-STP	0.93	'Good'	2-STP	0.07
LR	0.87	'Good'	LR	0.13

Fig. 1. Example of answer prediction. The LR-method is the usual logistic regression classification.

Algorithm 1. *SUM* quality model

```
input: $Q$ question, $t$ question type classifiers, $A$ answer,
$qua$ answer quality classifiers,
output: $score(Q,A)$
for $i = 1,\ldots,$ number of quality classifiers $qua$ do
  for $j = 1,\ldots,$ number of question type classifiers $t$ do
    $score(Q,A) =: argmax\left(\sum_i \left(\prod_{i,j} P(t_j|Q) \cdot P(qua_{i,j}|t_j,A)\right)\right)$
    end
  end
return $score(Q,A)$
```

As an alternative, we define our second algorithm which maximizes the quality by using the maximum expectation value of the question type classification. This approach is presented in Algorithm 2.

Algorithm 2. *MAX* quality model

```
input: $Q$ question, $t$ question type classifiers, $A$ answer,
qua answer quality classifiers,
output: $score(Q, A)$
for $i = 1, \ldots,$ number of quality classifiers qua do
  for $j = 1, \ldots,$ number of question type classifiers $t$ do
    $score(Q, A) =: argmax\left(\max_j\left(\prod_{i,j} P(t_j|Q) \cdot P(qua_{i,j}|t_j, A)\right)\right)$
    end
  end
return $score(Q, A)$
```

In Algorithm 2, the quality model is dependent on the accuracy of the question type component. In this manner we can analyze the behavior of the questions more intensively. As a tie-break between the first and the second algorithm, we also define a *2-step* approach. In this approach, the quality is determined based on hard-decision. First it determines in which question type a question is classified rather than the probabilities of the types. Next, the second classifier is used to determine the final quality score. The *2-step* method is described in Algorithm 3.

Algorithm 3. *2-STEP* quality model

```
input: $Q$ question, $t$ question type classifiers, $A$ answer,
qua answer quality classifiers,
output: $score(Q, A)$
for $i = 1, \ldots,$ number of quality classifiers qua do
  for $j = 1, \ldots,$ number of question type $t$ do
    $score(Q, A) =: argmax(P(qua_i|\max(P(t_j|Q), A))$
    end
  end
return $score(Q, A)$
```

To illustrate how the proposed methods work, we take as an example the question of the first example in Section 1, *i.e.*, "*How did the Holocaust really happen?*". Consider the following answers obtained from the CQA (real from our dataset):

1. Answer 1: `http://www.ushmm.org/wlc/article.php?lang=en&ModuleId=10005143`.
2. Answer 2: `Adolf Hitler gradually but effectively advanced through the ranks of German politics and through various intricate economic, political, and propaganda schemes managed to persuade the military powers of Germany that Jews were an inferior race that deserved to be systematically slaughtered.`

According to our annotation, the first answer is a low quality answer while the second is a high quality one. The calculation and the final decision following our methods as explained above in Fig. 1. In this example, we compare our method with the logistic regression classifier (as will be described in Section 4, logistic regression is better than other single classifiers). In Fig. 1(a), all of the methods predict the final quality of the first example as *bad*. Comparing the result of our methods in Fig. 1(a) to the logistic regression classification, we can see that all of our methods produce a higher prediction for the quality of the answer which is classified as *bad* – than the logistic regression. In Fig. 1(b), all of the methods predict the final quality of the second example as *good*. Again, all of our proposed methods have a higher prediction margin than that produced by the logistic regression.

3.3. The question classification layer

Our question analyzer follows the technique reported in [44,45]. We classify the question into six types, *i.e.*, *Definition, Factoid, Opinion, Procedure, Reason*, and *YesNo*. The question type model is trained using supervised approach with the following feature groups:

- **Lexical Features.** This feature group is used to identify the (co)-occurrences of words in specific question type. For example in the *opinion*-type questions, the word sequence of "*What is your opinion about ...?*" is expected to occur frequently.
  - **Unigram**: sequences of single word in the question.
  - **Bigram**: sequences of neighboring 2-words in the question.
  - **Trigram**: sequences of neighboring 3-words in the question.

- **Syntax-driven Semantic Features**. This feature group is used to analyze the syntactic structure occurred in a typical question type. By analyzing the syntactic structure, we can determine for instance, the difference between: *How many* ... in a *factoid*-type questions, and *How do* ... in a *procedure*-type questions.
  - **Focus Type**: This feature is applicable to questions with a *wh*-word of *what* or *which*. The focus word is identified using a manually-compiled set of syntactic patterns which are matched against a syntactic parse tree of the question. For example, in "*What is your favorite play by Shakespeare?*", *play* is the focus word. The syntactic parse tree of this question is as follows: *(ROOT (SBARQ (WHNP (WP What)) (SQ (VBZ is) (NP (NP (PRP$ your) (JJ favorite) (NN play)) (PP (IN by) (NP (NNP Shakespeare))))) (.?)))*. The syntactic tree is matched against the following pattern:

```
(ROOT(SBARQ(WHNP(WPWhat)) (SQ(VP(VBZ is) (NP(NP(DT the) (JJ xx) (NN xx)) (PP(IN xx)(*NP xx)))))))
```

which extracts a node using the specified tree templates by looking for the last consecutive NN or JJ that follows a question word *What*.

  - **Focus Adjective**: This feature identifies the focus of the *how* question, for example, the word *old* in "*How old do you have to be to be able to audition for america's best dance crew?*"
  - **Main Verb**: This feature specifies the main verb of the question following Collin's style head-rules of a syntactic parse tree [13].
  - **Question Word**: This feature is used to determine the question words: *wh*-word, and *how*.
  - **Question Word Determiner**: this feature indicates a *wh*-word as a question word or a determiner.

An example of the question type prediction result for "*How did Holocaust really happen?*" can be seen below:
Prediction:

```
[Class: Procedure 14.21097129083157] [Class: Procedure 1.0] [compact instance/null: BIGRAM.
Holocaust-really BIGRAM.did- Holocaust BIGRAM.really-happen FOCUS_TYPE.- MAIN_VERB.did TRIGRAM.
Holocaust-really-happen TRIGRAM.did-Holocaust-really UNIGRAM.Holocaust UNIGRAM.did UNIGRAM.happen
UNIGRAM.really]
```

Although the question type classification technique was initially designed to classify factoid questions, it achieves around 70% accuracy on our CQA test dataset, which is comparable to the performance reported in [45]. Though the question type classification performance is not optimum, it works for our framework. Consider the extreme case that the type prediction is removed the hierarchical framework is receded into a normal classification approach.

3.4. The quality prediction layer

The quality prediction layer consists of several basic classifiers for distinguishing high or low answer quality. The basic classifier can use any supervised machine learning algorithms, such as the Support Vector Machine, Logistic Regression, and other models which are appropriate for the specific applications. We leave the selection of classifiers to the experimental section, and focus on the design of the quality prediction features.

We adopt the basic textual features, such as the question–answer similarity, question/answer length statistics, and punctuation density, which are commonly used in answer quality prediction [1]. In addition, we propose some novel features such as the readability and the sentiment polarity. The complete feature groups are summarized as follows:

- **Similarity Features** (12 in total). This feature group gives overlapped terms proportion between a question and its answer, expressed in *n*-gram similarity with *n* up to 3. We expect that a *good* answer contains a considerable proportion of terms co-occurring in the question.
  - Resemblance (Resemblance-*n*-g). This feature gives the proportion of the set of overlapping *n*-grams, and the set of all *n*-grams for the question and its answer [10]. Denoting $S(Q)$ as the set of question *n*-grams and $S(A)$ as the set of answer *n*-grams, this feature can be expressed as:

$$Resemblance(Q,A) = \frac{|S(Q) \bigcap S(A)|}{|S(Q) \bigcup S(A)|} \tag{2}$$

where $S(Q) \bigcap S(A)$ is the set of overlapping *n*-grams, and $|S(Q) \bigcup S(A)| = (|S(Q)| + |S(A)|) - (2 * |S(Q) \bigcap S(A)|)$ is the set of all *n*-grams.

  - Containment (Containment-*n*-g). The containment gives the proportion of *n*-grams from the answer that also appear in the question [34], which can be expressed as:

$$Containment(Q,A) = \frac{|S(Q) \bigcap S(A)|}{|S(A)|} \tag{3}$$

  - Cosine distance (Cosine-*n*-g). This feature gives the cosine similarity between a question and its answer as follows:

$$Cosine(Q,A) = \frac{|S(Q) \bigcap S(A)|}{\sqrt{|S(Q)| * |S(A)|}} \tag{4}$$

- Word Overlap (Overlap-n-g). This feature gives the proportion of the number of overlapping n-grams between the answer and the question:

$$Overlap(Q,A) = \frac{|S(A)|}{|S(Q)|}. \tag{5}$$

- **Statistical Features** (9 in total). This feature set includes the statistics of the new surface semantic features. It gives the number of prominent countable terms and surface semantic features of a question and its answer. We expect that a *good* answer has a good structure and contains a reasonable number of shallow syntactic features, such as the verbs and nouns.
  - Raw length in question (Q-Raw-Length) in words.
  - Raw length in answer (A-Raw-Length) in words.
  - Number of nouns in question (Q-Nouns).
  - Number of nouns in answer (A-Nouns).
  - Number of verbs in question (Q-Verbs).
  - Number of verbs in answer (A-Verbs).
  - Number of sentences in answer (A-Sentences).
  - Number of stop words (A-Stopw).
  - Number of non-stop words (A-NonStopw) in answer.
- **Answer Question Ratio** (1 in total). This feature gives the ratio between the raw lengths of a question and its answer (Ratio-A-Q). We expect that a *good* answer has a considerable word length. In other words, a *good* answer should contain some explanations to support its credibility.
- **Density** (6 in total). This feature set gives the density of special tokens which is found in a question and its answer. We expect that a *good* answer contains a considerable low proportion of unused punctuations or special symbols, such as:-).
  - Punctuations density in question (Q-Punct-Dens), computed as:

$$Q\_Punct\_Dens = \frac{number\_of\_punctuations\_in\_question}{number\_of\_characters\_in\_question} \tag{6}$$

  - Punctuations density in answer (A-Punct-Dens), computed following the same approach as Q-Punct-Dens, this time using the number of punctuations in the answer.
  - Punctuations density in question–answer pair (Punct-Dens), computed following the same approach as Q-Punct-Dens, this time using the number of punctuations in the question and answer together as a pair.
  - Non-ASCII characters in question (Q-NonAscii-Dens), computed as:

$$Q\_NonAscii\_Dens = \frac{number\_of\_non\_ascii\_chars\_in\_question}{number\_of\_characters\_in\_question} \tag{7}$$

  - Non-ASCII characters in answer (A-NonAscii-Dens), computed following the same approach as Q-NonAscii-Dens, this time using the number of punctuations in the answer.
  - Non-ASCII characters in question–answer pair (NonAscii-Dens), computed following the same approach as Q-NonAscii-Dens, this time using the number of non-ASCII characters in the question and answer together as a pair.

Beside the above surface textual features (referred to as **SF-TXT** features later), we also use some novel features (referred to as **NOVEL** features later), which hypothetically will be useful to distinguish the high quality from the low quality answers. They are:

- **Readability** (3 in total). This feature set gives the scores of three popular readability models [48], which estimate the educational grade level necessary to understand a portion of text based on the number of syllables detected in the given portion of text. We expect that a *good* answer contains a considerable amount of formal words and written in a good structure.

  Fog score (S_{Fog}). The readability score is computed as:

$$S_{Fog} = 0.4(average\_text\_length + \%\_of\_Hard\_Words) \tag{8}$$

where *Hard Words* is the number of words with more than two syllables of a given text.

  - Flesch score (S_{Flesch}), the score is computed as:

$$S_{Flesch} = 206.835(1.015 * ASL)(84.6 * ASW) \tag{9}$$

where *ASL* is the average sentence length (number of words divided by number of sentences); and *ASW* is the average word length in syllables (number of syllables divided by number of words)

  - Flesch-Kincaid score ($S_{Kincaid}$), the score is computed as:

$$S_{Kincaid} = 0.39 * \left(\frac{total\_words}{total\_sentences}\right) + 11.8 * \left(\frac{total\_syllables}{total\_words}\right) - 15.59 \tag{10}$$

- **Monolingual word translation**. This feature gives the probability of the semantic relatedness of a question word and an answer word. We use the statistical translation model as conducted in [5,26]. We expect that in some question types, there will be some word-to-word relatedness between question and answer, for instance, in a *reason*-type questions, the answer are expected to contain the word *because* ...
- **Number of links/URL** in the answer. This feature gives the number of URL found in the answer. We expect that some facts or explanation of a *good* answer are shown as URL in the answer.
- **Sentiment polarity** of the answer. The polarity is computed as an enumeration of the number of words in the answer which express the following sentiment: strong-negative, weak-negative, strong-positive, and weak-positive. We use the opinion word list from [54]. We expect that a *good* answer is written in a *positive* sense.

4. Experiments

4.1. Experimental set-up

In order to set up our experiment, we collected data from Yahoo! Answers. All together, 5854 questions with 46,821 answers covering 12 categories have been collected from April 2006 to April 2012 period. Table 1 gives the statistic of our data collection. Our collection consists of around 80% of *good* quality answers. We conjecture that in reality people tend to give *good* answers, thus a CQA system usually has more *good* answers than *bad* answers. It is thus important for a CQA system to reject the low quality answers during retrieval, while maintaining the overall performance.

We label each question answer pair manually with a quality label in *good* and *bad* judgement. Beside the quality label, we also annotate each question answer pair with a specific question type. Two annotators spent 10 working days on the annotation independently. When discrepancy happened, they would discuss to assign a final label. To justify our quality label, we compare our annotation with the original *best answer* (selected by the cQA users) of each question. We used Cohen's kappa statistics for this purpose and achieved 0.97 agreement. After the development of the dataset, we conducted the following studies:

- Explore various machine learning algorithms to select the most appropriate basic classifiers and features.
- Evaluate various hierarchy combinations of classifiers based on the question types.
- Analyze various feature sets to see their influence on different question domains.

Our data is a naturally imbalanced collection [46,33], thus we experiment on both the natural and balanced data. As the baseline, we used the basic feature set from question answer pairs by performing 10 folds cross-validation on the **natural dataset** (about 80%:20% of good:bad, the ratio in a natural dataset according to our annotation) and **balanced dataset** (50%:50% of good:bad ratio, intentionally designed balanced data for experiments on machine learning methods). We used the *accuracy* as our evaluation metric in each experiment, *i.e.*, the number of correct classification according to the manual labeling.

Each classifier in our framework is trained using a supervised machine learning algorithm. The evaluation for the most appropriate machine learning algorithm is conducted by running some experiments using the baseline feature sets on a number of popular algorithms: support vector machines, logistic regression, random forest, sequential minimal optimization, and voted perceptron.

In order to evaluate the influence of the feature set that we used, we conducted feature selection experiment by using random subset approach as proposed in [32]. Moreover, we conduct deep analysis on the effects of proposed hierarchical methods on different question types and question domains.

4.2. Experimental results

We begin our study with the initial performance analysis of cross validation experiments on the natural and balanced datasets. The main objectives are to determine the best machine learning algorithm as the basic classifier and to provide some enhancements on the chosen algorithm in terms of feature selection. In our setting, the best algorithm not only depends on the overall accuracy performance, but also the one which can classify *bad* quality answers more accurately. Beside

Table 1
Dataset statistics.

Quality	Question type						Total
	Definition	Factoid	Opinion	Procedure	Reason	YesNo	
Good	714	6190	17,279	3510	7282	2066	37,041
Bad	225	1782	4287	925	1540	1021	9780
Total	939	7972	21,566	4435	8822	3087	46,821

the analysis of machine learning performance, we also report our explorations of the importance of the features on each question type and Y!A category.

4.2.1. Selection of base classifiers

Table 2 presents results of the baseline models with the **SF-TXT** features. We can see that all machine learning algorithms have comparable accuracies in predicting *good* samples, while SVM and logistic regression achieve the best accuracy and are stable for all the feature sets.

We use the baseline accuracies to select the most appropriate classifier as the base classifier in our framework. A preference score function is defined to predict the appropriateness of each classifier. The function is defined as: $preference_{ML} = \max \sum w_i \times acc_{ML}$, where w_i is a weighting factor of the *good* data samples, and acc_{ML} is the accuracy of a machine learning algorithm of a dataset. In our case, the weights will be 0.8 for the natural dataset and 0.5 for the balanced dataset. Based on these weights, we can infer from Table 2 that logistic regression is the most preferred machine learning algorithm, *i.e.*: $0.8 * 79.08 + 0.5 * 59.45 = 92.99$. The choice of logistic regression classifier supports our discussion in Section 3 that linear combination is an appropriate approach for our features.

4.2.2. Analysis of feature importance

4.2.2.1. Overall feature importance. To analyze the importance and the combination effects of each feature, we implemented the random feature selection method with correlation-based subset evaluation [32]. Fig. 2 displays the most influential **SF-TXT** features. These selected features will be expanded with the novel feature sets for further experiments. To compute the weights of each feature, we conduct a feature ranking evaluation by applying the chi-squared statistic, with importance value normalized to between 0 and 100. For analysis purposes, we split our dataset into 75% (35,116 question–answer pairs) of train- and 25% test (11,705 question–answer pairs) data randomly.

Fig. 2 shows that the most influential feature is the number of non-stop words in the answer and the answer length. This result suggests that in a CQA system, the length of an answer is a very important indication for its quality. This feature is strengthened by the readability score, indicating that a long (usually informative) answer with careful writing style is usually considered as a good quality of answer. To conclude, a good quality of answer is indicated by:

1. Its length: the overlapped text with the question.
2. Its content: consists of a number of facts (*i.e.*, the number of nouns).
3. Its structure: (*i.e.*, the number of verbs and punctuations, and the readability score).

4.2.2.2. Feature importance vs. question type. Table 3 describes that the performance of our selected features (SF-TXT) together with the **NOVEL** features has a comparable accurate result as the baseline in Table 2. This results indicate that the selected features performed effectively, and can differentiate more *bad* answers. Further analysis on the question types shows that the performance of the **SF-TXT** + **NOVEL**-features outperforms the baseline in the *Definition, Reason,* and *Opinion* question types. This also indicates that the features can differentiate more *good* answers for the questions which require explanations and appropriate sentence structure in the answers.

To analyze the feature importance in each question type, we implemented the random feature selection method [32] with correlation-based subset evaluation [19] on the **SF-TXT** + **NOVEL** feature sets.

In most question types, except for the Yes/No-type, the length of the answer and the overlap context are the most important. This fact is manifested in the following features: *A-Raw-Length, Ratio-A-Q, Overlap-1-g,* and *Containment-1-g*. However, in the YesNo-type, people tends to answer a question in short forms, such as: *yes, yup, no, and nope*. Therefore, the sentiment feature set, either in positive or negative responses, is the most influential feature set in the YesNo-type. Another interesting finding is that the translation model feature is one of the influence features for the reason and opinion question types. We conjecture that this is a logical consequence from the tendency of the answers in these two question types where people usually include some specific phrases in their answers, such as: *...because of ..., I think ..., My opinion on this is ..., You shouldn't ...*.

For certain question types, such as, Procedure, Factoid, and Yes/No, where the **SF-TXT** + **NOVEL** features perform worse than that of the baseline, we find that these question types require a number of facts or evidences in the answers, or the *number of nouns in answer* feature. However, as sometimes the noun recognizer fails to identify all the nouns, thus such important evidences could not be properly utilized in the classification process. This is a common problem when NLP tools developed for proper text are used on UGC type of text.

Table 2
Accuracy of baseline results with different base classifiers. The numbers in the brackets show the accuracy of predicting *bad* answers for each feature set.

Dataset	Classifiers				
	SVM	LR	RF	SO	VP
Natural dataset	79.11 (0.00)	79.08 (0.07)	78.56 (0.64)	79.11 (0.00)	79.11 (0.02)
Balanced dataset	59.24 (65.88)	59.45 (67.83)	58.32 (69.95)	59.36 (68.08)	59.38 (66.69)

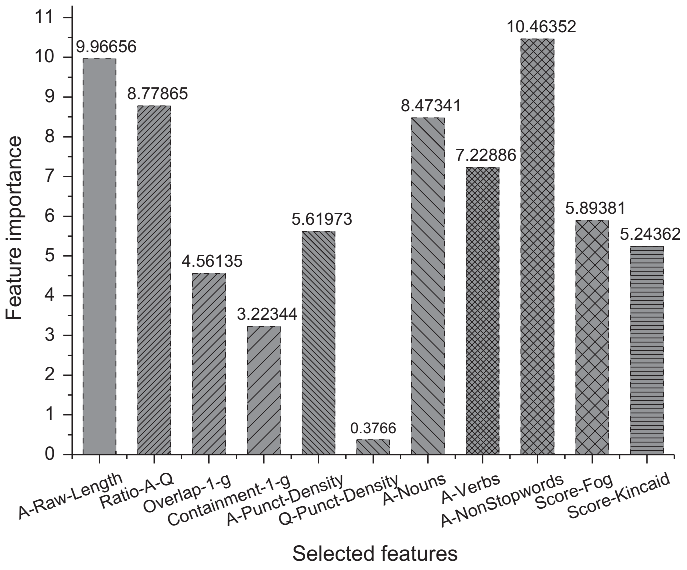


Fig. 2. The most influential features (feature names encoding in Section 3).

Table 3
Experiments using selected **SF-TXT** features and **NOVEL** features. The numbers in the brackets show the accuracy of *bad* answers.

Q. Type	SF-TXT (S)	S + Trans	S + URL	S + Polar	All
Definition	80.74	80.74	80.74	80.74	80.74
Procedure	79.30	79.30	79.30	79.39	79.39
Reason	83.00	83.04	83.00	83.05	83.00
Opinion	80.01	80.01	80.01	80.01	80.03
Factoid	76.61	76.61	76.61	76.66	76.66
YesNo	65.54	65.54	65.54	65.54	65.63
Overall	78.98 (0.77)	78.99 (0.73)	78.98 (0.73)	79.01 (0.77)	79.01 (0.77)

4.2.3. Effectiveness of hierarchy of classifiers method

In the next experiment, we explore the effectiveness of the proposed hierarchy of classifiers method. As the baseline in this experiment we use the result of the **SF-TXT** + **NOVEL** features, the right most column in Table 3. We evaluate our approach in *all*- and *selected* per question type-features settings.

Table 4 indicates that the *Max* and *2-step* methods have a comparable performance in overall accuracy as compared to the baseline, and performed better than the *Sum* method. The performance of the *Max* approach is slightly better than the *2-step* method. The *2-step* method is a kind of *hard*-styled classification system. It requires two classification steps, first to determine the question type and then the question quality. As a consequence, it also requires an accurate question type classification otherwise it will fail to determine the final question quality. Due to this characteristic, the *2-step* method is more depended on the accuracy of the question type classifier than the other methods.

Table 4
Experiments of hierarchical methods with *all features* and *selected features per question type*. The numbers in the brackets show the accuracy in predicting *bad* answers. Here † means that the difference to the respective baseline is statistically significant with $p = 5\%$. The bold values show the best aggregation result for each aggregation method.

Q. Type	All features			Selected features		
	Sum	Max	2-step	Sum	Max	2-step
Definition	73.77	81.56	79.92	74.59	80.33	80.33
Procedure	78.85	79.75	79.84	78.49	80.02	75.45
Reason	82.31	83.00	82.77	82.45	82.82	82.58
Opinion	78.38	79.57	79.35	78.72	79.77	79.39
Factoid	75.70	76.15	75.80	75.19	76.15	75.34
YesNo	64.38	65.28	65.28	65.28	65.54	64.90
Overall	77.68 (6.91†)	**78.74** (5.00†)	78.51 (5.89†)	77.82 (5.85†)	**78.82** (4.14†)	77.98 (5.03†)

Based on the experiment results, we conclude that the question classifier performed moderately. This is supported by the fact that the *2-step* method performed better than the *Sum* method. We actually expect the *Sum* and *Max* methods to capture more overlapped question features, such as the question words, which sometimes make it hard to distinguish the question types. The best result in our experiment is achieved by the *Max* method. Its accuracy of the *bad* quality questions is significantly better than that of the baseline, while the overall accuracy is comparable. It indicates the potential of our methods which can distinguish *bad* answers more accurately than the common machine learning approaches.

In all the methods, the best performance is achieved on the *Reason* and *Definition* question types, while the worst is for the *YesNo*-type. Based on this fact, we consider the answer quality of *YesNo*-type questions to be the most difficult type to predict. These results also suggest that in *Reason* and *Definition* question types, people tend to answer the questions completely, in good sentence structure, and with substantial facts and evidences.

Comparing the effects of the features in Table 4, we can see that the overall accuracy of the selected features is as good as its complete version, and even better for the *Sum* and *Max* methods.

4.2.4. Analysis of domain adaption

To see how our proposed methods adapt to different domains, we conduct analysis on the per-domain performance. As our experimental data is collected from Yahoo! Answers, where questions are already grouped into different categories, we can conveniently use the category information as the domain information for this study. The main objective of this analysis is to show that the proposed methods work on most categories and to find out in which category the answer quality is hard to be predicted. The accuracy of each method in different domains is presented in Table 5.

Based on the results in Table 5 we can see that the questions in the *Poetry*-category are the most difficult to predict. All methods have only an accuracy of around 60%. An exception to this result can be found in the *selected*features of the *2-step* method in Table 5. One of the reason of this exception is that in the *2-step* method, the *hard* decision of the question type classification leads the prediction to the most appropriate quality classifier, especially for the *Opinion*-type questions. Further analysis for this exception leads us to the most discriminated features for the *Poetry* category, *i.e.*, the translation model feature. A typical question in the *Poetry* category is in asking others' opinion about a poem, for example "*I wrote this? Your opinion of it please?*" or "*A poem from the heart. Is it any good?*". The answers to this kind of question usually contain opinion-based words or phrases, such as: *I think your poem was just really outstanding, please due keep up the good work!!* ... or *Your poem is more than just good* .... In such cases, the translation model will give significant contribution on the prediction results.

Compared with the baseline, we can see that in the *Dancing*, *Painting*, and *Poetry* where the question type is dominated by *Definition*, *Opinion*, *Procedure*, and *Reason*-types, our methods have better accuracy. This results indicate that our methods performed their best in questions where the well-structured answers are expected.

4.3. Case studies

In this section we provide a number of case studies of *bad* quality answers to show the potential of our proposed methods in discriminating the *bad* quality answers from the *good* ones. Each case study is taken from real world CQA scenario from our dataset. The case studies can be followed in Table 6. We can make the following observations that help to strengthen our analysis in the previous sub-section:

1. Our methods can generally be applied accurately across various categories and question types. In Table 6 we present 7 categories in all question types from our dataset.
2. Our methods work well on questions which expect direct answers with well-structured sentences, such as the *Factoid* and *Definition* types. However, for questions that expect more personal understanding, such as the *Opinion* and *Reason* type, the intrinsic features including our **SF-TXT** features and **NOVEL** features might not be adequate. Some external background knowledge might be needed.

Table 5
Accuracy in Y!A categories using **selected** feature sets.

Category	Sum	Max	2-step
Books & Author	90.54	91.01	78.24
Dancing	77.51	79.73	75.95
Drawing & Illustration	81.85	81.51	74.32
Fashion & Accessories	77.69	78.87	74.32
Genealogy	74.77	75.23	75.92
History	77.77	77.43	80.36
Painting	81.65	82.26	77.02
Performing Arts	79.44	81.13	79.44
Philosophy	82.12	83.88	79.71
Photography	82.85	84.24	75.87
Poetry	60.33	60.55	79.07
Theater & Acting	79.17	80.04	79.82

Table 6
Answer quality case study of the proposed methods. Note that all the listed examples should be labeled as *bad*. The columns are: *Cat* = Category, *QT* = Question Type, *L* = Logistic Regression, *S* = Sum, *M* = Max, and *2s* = 2-step. The prediction of the answer quality is shown as: *G* = *Good* and *B* = *Bad*. The categories are: *His* = History, *TA* = Theatre & Acting, *BA* = Books & Authors, *FA* = Fashion & Accessories, *Phi* = Philosophy, *Pai* = Paint, and *Poe* = Poetry. The question types are: *Pro* = Procedure, *Fac* = Factoid, *Opi* = Opinion, *YN* = YesNo, *Def* = Definition, and *Rea* = Reason. The bold values give the prediction of our aggregation methods which discriminate 'bad'-quality answers more accurately than a base-classifier alone.

No.	Question	Community answer	Cat	QT	L	S	M	2s
1	How did the Holocaust really happen?	http://www.ushmm.org/wlc/article.php?lang=en&ModuleId=10005143	His	Pro	B	B	B	B
2	Which Shakespearean play has the most parts for women?	I will be sure not to spam your answer box, I'll keep a look out for other spammers too. All the best:)	TA	Fac	G	G	G	G
3	Do you think William S. Burroughs was heavily influenced by James Joyce?	I've never read any Burroughs, but a guy I work with is a big fan and told me James Joyce was a huge influence on him, Naked Lunch in particular.	BA	Opi	G	G	G	G
4	Men in ugg boots- yay or nay?	BOOOO	FA	YN	B	B	B	B
5	Would you fit the definition of "normal"?	I could pass for normal at a distance-maybe.	Phi	Def	G	**B**	**B**	**B**
6	Why are barns always usually red!?	Excellent Question... so the cows don't run into them???	Pai	Rea	G	G	G	G
7	What might be your thoughts, on forty-eight years old in 1 h time by Suzi Quatro?	Happy Birthday ... I send you 48 virtual hugs. (()) (()) (()) ...	Poe	Opi	G	**B**	**B**	**B**

3. Our selected features performed as good as the full feature sets, as can be seen in all the examples, indicating that in real applications only the selected features need to be extracted for efficiency purpose. This shows that our proposed approach can achieve both good efficiency and accuracy at the same time.

5. Conclusion and future work

In this study we proposed a novel and generic hierarchy-of-classifiers framework in predicting the quality of answers in CQA. Our initial experiments show that logistic regression is the most suitable machine learning model that fits our problem (Section 4.2.1). With further experiments we found that the hierarchical classification approach with weighted confidence of prediction probability is an effective method to discover high quality answers and distinguish low quality answers in a CQA system. Among the 3 variants of the aggregation methods (Section 3), *i.e.*, *Sum*, *Max*, and *2-step*, the *Max* method works best for the quality prediction task.

By performing feature selection, we identified that good quality answers are dominated by long answers completed with convenience facts or evidences, and written in a good structure (Section 4.2.2). Our proposed selected (*SF-TXT*) feature sets achieve a comparable accuracy as the *full* feature sets, this indicates the efficiency of our approach. The significant finding in our experiments is that our methods can distinguish *bad* answers more accurately than the common machine learning approaches. Our methods have shown their potentials in discriminating the *bad* quality answers by more than five times better accuracies than the baseline machine learning algorithm, while maintaining the overall accuracy (Section 4.2.3).

By analyzing the question category, we showed the most influential feature sets for each type of answer quality classifiers in different categories (Section 4.2.4). We found that in CQA the two most difficult question types are the *Factoid* and *YesNo*-type, as they expect more evidences and background knowledge. For other types of answers, such as *Definition*, *Opinion*, *Procedure*, and *Reason*; people tend to answer those questions in a proper manner, and our methods have predicted their quality more accurately.

For future work, we plan to extend our framework to other types of user-generated-content, such as microblog, tweets, and forum postings. We conjecture that with the new design of type-dependent classification, *i.e.*, the first layer of the hierarchy of classifiers, our proposed framework and wide spectrum of intrinsic text feature should be able to capture the quality-related information of various types of contents more completely and accurately.

Acknowledgement

This research is supported by the Singapore National Research Foundation under its International Research Centre @ Singapore Funding Initiative and administered by the IDM Programme Office, and by I-MHERE Project of Faculty of Computer Science, Universitas Indonesia, IBRD Loan No. 4789-IND & IDA Credit No. 4077-IND, Directorate General of Higher Education Ministry of Education and Culture, Republic of Indonesia.

References

[1] E. Agichtein, C. Castillo, D. Donato, A. Gionis, G. Mishne, Finding high-quality content in social media, in: Proceedings of WSDM, 2008.
[2] M. Allahbakhsh, B. Benatallah, A. Ignjatovic, H.R. Motahari-Nezhad, E. Bertino, S. Dustdar, Quality control in crowdsourcing systems: issues and directions, Internet Computing, IEEE 17 (2) (2013) 76–81.
[3] A. Anderson, D. Huttenlocher, J. Kleinberg, J. Leskovec, Discovering value from community activity on focused question answering sites: a case study of stack overflow, in: Proceedings of KDD, 2012.

[4] A.R. Backes, D. Casanova, O.M. Bruno, Texture analysis and classification: a complex network-based approach, Information Sciences (2012).

[5] D. Bernhard and I. Gurevych, Combining lexical semantic resources with question and answer archives for translation-based answer finding, in: Proceedings of ACL, 2009.

[6] J. Bian, Y. Liu, E. Agichtein, H. Zha, Finding the right facts in the crowd: factoid question answering over social media, in: Proceedings of WWW, 2008.

[7] M.J. Blooma, A.Y.K. Chua, D.H.-L Goh, Predictive framework for retrieving the best answer, in: Proceedings of TAC, 2008.

[8] M.J. Blooma, J.C. Kurian, Research issues in community based question answering, in: Proceedings of PACIS, 2011.

[9] M. Bouguessa, B. Dumoulin, S. Wang, Identifying authoritative actors in question–answering forums: the case of Yahoo! answers, in: Proceedings of KDD, 2008.

[10] A.Z. Broder, On the resemblance and containment of documents, in: Proceedings of Compression and Complexity of Sequences, 1997.

[11] L. Chen, D. Zhang, M. Levene, Question retrieval with user intent, in: Proceedings of SIGIR, 2013.

[12] A.Y. Chua, S. Banerjee, So fast so good: an analysis of answer quality and answer speed in community question–answering sites, Journal of the American Society for Information Science and Technology (2013).

[13] M. Collins, Three generative, lexicalised models for statistical parsing, in: Proceedings of ACL/EACL, 1997.

[14] G. Cong, L. Wang, C.-Y. Lin, Y.-I. Song, Y. Sun, Finding question–answer pairs from online forums, in: Proceedings of SIGIR, 2008.

[15] H.T. Dang, Overview of the TAC 2008 opinion question answering and summarization tasks, in: Proceedings of Text Analysis Conference, 2008.

[16] R. Gazan, Social Q&A, Journal of The American Society for Information Science and Technology 62 (12) (2012) 2301–C2312.

[17] B. Green, A. Wolf, C. Chomsky, K. Laughery, BASEBALL: an automatic question answerer, in: Proceedings of the Western Joint Computer Conference, 1961.

[18] J. Guo, S. Xu, S. Bao, Y. Yu, Tapping on the potential of Q&A community by recommending answer providers, in: Proceedings of CIKM, 2011.

[19] M.A. Hall, Correlation-based Feature Subset Selection for Machine Learning, Hamilton, New Zealand, 1998.

[20] I. Heim, The Semantics of Definite and Indefinite Noun Phrases, Ph.D. Thesis, University of Massachusetts, 1982.

[21] F. Hieber, S. Riezler, Improved answer ranking in social question–answering portals, in: Proceedings of SMUC, 2011.

[22] L. Hirschman, R. Gaizauskas, Natural language question answering: the view from here, Natural Language Engineering 7 (4) (2001) 275–300.

[23] R. Hong, M. Wang, G. Li, L. Nie, Z.-J. Zha, T.-S. Chua, Multimedia question answering, IEEE MultiMedia 19 (4) (2012) 72–78.

[24] J. Jeon, W.B. Croft, J.H. Lee, S. Park, A Framework to predict the quality of answers with non textual features, in: Proceedings of SIGIR, 2006.

[25] H. Kamp, A theory of truth and semantic representation, in: J. Groenendijk, T.M. Janssen, M. Stokhof (Eds.), Truth, Interpretation and Information: Selected Papers from the 3^{rd} Amsterdam Colloquium, Dordrecht – Holland/ Cinnaminson – USA Foris, 1984, pp. 1–41.

[26] M. Karimzadehgan, C.X. Zhai, Estimation of statistical translation models based on mutual information for ad hoc information retrieval, in: Proceedings of SIGIR, 2010.

[27] J.M. Kleinberg, Authoritative sources in a hyperlinked environment, Journal of the ACM 46 (5) (1999) 604–632.

[28] W.G. Lehnert, A conceptual theory of question answering, in: Proceedings of the 5th International Joint Conference on Artificial Intelligence, 1977.

[29] J. Lou, Y. Fang, K.H. Lim, J.Z. Peng, Contributing high quantity and quality knowledge to online q&a communities, Journal of the American Society for Information Science and Technology 64 (2) (2013) 356–371.

[30] X. Li, D. Roth, Learning question classifiers, in: Proceedings of COLING, 2002.

[31] Q. Liu, E. Agichtein, Modeling answerer behavior in collaborative question answering systems, in: Proceedings of ECIR, 2011.

[32] H. Liu, R. Setiono, A probabilistic approach to feature selection – a filter solution, in: Proceedings of ICML, 1996.

[33] V. López, A. Fernández, S. García, V. Palade, F. Herrera, An insight into classification with imbalanced data: empirical results and current trends on using data intrinsic characteristics, Information Sciences (2013).

[34] C. Lyon, J. Malcolm, B. Dickerson, Detecting short passages of similar text in large document collections, in: Proceedings of EMNLP, 2001.

[35] M.G. Main, D.B. Benson, Denotational semantics for natural language question answering programs, Association for Computational Linguistics (1983).

[36] Z.-Y. Ming, K. Wang, T.-S. Chua, Vocabulary filtering for term weighting in archived question search, Advances in Knowledge Discovery and Data Mining (2010) 383–390.

[37] Z.-Y. Ming, K. Wang, T.-S. Chua, Prototype hierarchy based clustering for the categorization and navigation of web collections, in: Proceeding of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval, ACM, 2010, pp. 2–9.

[38] A. Moschitti, S. Quarteroni, Linguistic kernels for answer re-ranking in question answering systems, Journal of Information Processing and Management 47 (2011) 825C–842.

[39] L. Page, S. Brin, R. Motwani, T. Winograd, The PageRank Citation Ranking: Bringing Order to the Web, Technical Report, Stanford Digital Library Technologies Project, 1998.

[40] A. Pal, F.M. Harper, J.A. Konstan, Exploring question selection bias to identify experts and potential experts in community question answering, ACM Transactions on Information Systems 30 (2) (2012). Article No. 10.

[41] A. Peñas, P. Forner, A. Rodrigo, R. Sutcliffe, C. Forascu, C. Mota, Overview of ResPubliQA 2010: Question Answering Evaluation over European Legislation, Working Notes CLEF Labs, 2010.

[42] J.A. Sáez, M. Galar, J. Luengo, F. Herrera, Tackling the problem of classification with noisy data using multiple classifier systems: analysis of the performance and robustness, Information Sciences (2013).

[43] T. Sakai, D. Ishikawa, N. Kando, Y. Seki, K. Kuriyama, C.-Y. Lin, Using graded-relevance metrics for evaluating community QA answer selection, in: Proceedings of WSDM, 2011.

[44] N. Schlaefer, P. Gieselmann, T. Schaaf, A. Weibel, A pattern learning approach to question answering within Ephyra framework, LNAI 4188 (2006) 687–694.

[45] N. Schlaefer, J. Ko, J. Betteridge, G. Sautter, M. Pathak, E. Nyberg, Semantic extensions of the Ephyra QA system for TREC 2007, in: Proceedings of TREC, 2007.

[46] C. Seiffert, T.M. Khoshgoftaar, J. Van Hulse, A. Folleco. An empirical study of the classification performance of learners on imbalanced and noisy software quality data, Information Sciences, 259 (2014) 571–595.

[47] C. Shah, J. Pomerantz, Evaluating and predicting answer quality in community QA, in: Proceedings of SIGIR, 2010.

[48] L. Si, J. Callan, A statistical model for scientific readability, in: Proceedings of CIKM, 2001.

[49] M. Surdeanu, M. Ciaramita, H. Zaragoza, Learning to rank answers to non-factoid questions from web collections, Computational Linguistics 37 (2) (2011) 351–383.

[50] M.A. Suryanto, E.-P. Lim, A. Sun, R.H.L. Chiang, Quality-aware collaborative question answering: methods and evaluations, in: Proceedings of WSDM, 2009.

[51] Y.R. Tausczik, J.W. Pennebaker, Predicting the perceived quality of online mathematics contributions from users reputations, in: Proceedings of CHI, 2011.

[52] X.-J. Wang, X. Tu, D. Feng, L. Zhang, Ranking community answers by modeling question–answer relationships via analogical reasoning, in: Proceedings of SIGIR, 2009.

[53] I. Weber, A. Ukkonen, A. Gionis, Answers, not links: extracting tips from yahoo! answers to address how-to web queries, in: Proceedings of WSDM, 2012.

[54] T. Wilson, J. Wiebe, P. Hoffmann, Recognizing contextual polarity in phrase-level sentiment analysis, in: Proceedings of HLT-EMNLP, 2005.

[55] T. Winograd, Understanding Natural Language, Academic Press, New York, 1972.

[56] W.A. Woods, R.M. Kaplan, B.L. Nash-Webber, The Lunar Sciences Natural Language Information System: Final Report, Technical Report 2378, BBN, 1972.

[57] R. Xia, C. Zong, S. Li, Ensemble of feature sets and classification algorithms for sentiment classification, Information Sciences 181 (6) (2011) 1138–1152.

[58] R. Yan, J. Yang, A.G. Hauptmann, Learning query-class dependent weights in automatic video retrieval, in: Proceedings of ACM MM, 2004.
[59] H. Yang, L. Chaisorn, Y. Zhao, S.-Y. Neo, T.-S. Chua, VideoQA: question answering on news video, in: Proceedings of ACM MM, 2003.
[60] J. Zhang, M.S. Ackerman, L. Adamic, Expertise networks in online communities: structure and algorithm, in: Proceedings of WWW, 2007.
[61] T.C. Zhou, M.R. Lyu, I. King, A classification-based approach to question routing in community question answering, in: Proceedings of WWW Companion, 2012.