Journal on Artificial Intelligence
DOI:10.32604/jai.2021.027590
Article

Tech Science Press

Answer Classification via Machine Learning in Community Question Answering

Yue Jiang, Xinyu Zhang, Wohuan Jia and Li Xu*

College of Computer Science and Technology, Harbin Engineering University, Harbin, 150001, China
*Corresponding Author: Li Xu. Email: xuli@hrbeu.edu.cn
Received: 21 January 2022; Accepted: 25 January 2022

Abstract: As a new type of knowledge sharing platform, the community question answer website realizes the acquisition and sharing of knowledge, and is loved and sought after by the majority of users. But for multi-answer questions, answer quality assessment becomes a challenge. The answer selection in CQA (Community Question Answer) was proposed as a challenge task in the SemEval competition, which gave a data set and proposed two subtasks. Task-A is to give a question (including short title and extended description) and its answers, and divide each answer into absolutely relevant (good), potentially relevant (potential) and bad or irrelevant (bad, dialog, non-English, other). Task-B is to give a YES/NO type question (including short title and extended description) and some answers. Based on the answer of the absolute correlation type (good), judge whether the answer to the whole question should be yes, no or uncertain. This paper first preprocesses this data set, and then uses natural language processing technology to perform word segmentation, part-of-speech tagging and named entity recognition on the data set, and then perform feature extraction on the preprocessed data set. Finally, SVM and random forest are used to classify on the basis of feature extraction, and the classification results are analyzed and compared. The experiments in this paper show that SVM and random forest methods have good results on the data set, and exceed the multi-classifier ensemble learning method and hierarchical classification method proposed by the predecessors.

Keywords: Community question answering; SVM; random forest

1 Introduction

With the rapid development of Internet technology and the rapid popularization of mobile Internet, socialization, personalization, and communalization have become the trends of the Internet. As a new type of knowledge sharing platform, the CQA (community question answer) website can realize the acquisition and sharing of knowledge by virtue of its good interactivity and reasonable incentive mechanism, and meet the personalized knowledge needs of different users. It is loved and sought after by the majority of users. Typical CQA websites include StackOverflow, Yahoo Answers, TurboTax [1–6]. At present, industry peers and researchers in related fields have gradually begun to pay attention to questions in the CQA, and continue to enrich the research in the field of the CQA.

Since the CQA involves a wealth of content and topics, the answers to the questions raised by the community users will accumulate more and more over time, so that the needs of the users can be solved. However, with the ever-increasing resources of users and answers, the CQA system is also facing many challenges. Understanding user behavior patterns, accurately positioning user needs, and providing high-quality answers to user's query requests have become urgent problems in the CQA system. In recent years, more and more scholars at home and abroad have participated in the research of community question answering [7–9].

In this paper, we mainly focus on the data set in SemEval. The data set contains 3 files. Each training set consists of questions and several answers. The data set is xml file format. This paper completes two subtasks, namely Task-A and Task-B. Task-A is to give a question (including short title and extended description) and its answers, and divide each answer into absolutely relevant (good), potentially relevant (potential) and bad or irrelevant (bad, dialog, non-English, other). Task-B is to give a YES/NO type question (including short title and extended description) and some answers. Based on the answer of the absolute correlation type (good), judge whether the answer to the whole question should be yes, no or uncertain [10]. This paper first extracts the attribute information and content information of each question and answer from the data set, preprocesses the data set, removes the tags after reading the xml data set, extracts the attribute information and content information, and saves it as a json file that is convenient for processing. Then use natural language processing methods to perform word segmentation, part-of-speech tagging and named entity recognition on the data set.

Then use the relevant rules to extract the features of the attribute and content information. After extracting the features, complete the preparation of label formulation and parameter setting, and finally use SVM and random forest methods for classification. Summarize and analyze the data by observing the classification results, and compare with the previous experimental results. Experiments shows that SVM and random forest have a good effect on the data set, which is better than the multi-classifier integrated learning method and hierarchical classification method proposed by predecessors.

2 Answer Classification Method

The overall implementation of the answer classification method via machine learning in the paper is shown in Fig. 1.

Figure 1: The overall implementation framework

In machine learning, there are many models that can be used in classification problems, and the selection of different classification algorithms in different scenarios often leads to good experimental results. Classification problems in general can be divided into two processes, training process and classification process.

Training process: “learning” or “training” is the process of getting the desired model from the data, usually this process needs to run the corresponding learning algorithm to get, “training samples” is the data used in the training process. The set of training samples is often called the “training set”. The learned model shows some internal underlying pattern of the data in question, and the training process is to find some underlying pattern in the data, which can be seen as the instantiation of the training algorithm on a given data and parameter space.

Classification process: The purpose of classification learning is to learn a classification function or classification model, also often called a classifier, from a given dataset of manually labeled classification training samples. When new data arrives, predictions can be made based on this function, mapping the new data items to one of the classes in the given category [11–13].

2.1 Data Preprocessing

The initial data set is a question-answer xml file, which contains some key information on the CQA website. Need to be pre-processed. First remove the tags and symbols in the file, extract key information, and then use natural language processing to process some basic words through the methods of word segmentation, part-of-speech tagging and named entity recognition to extract normative data for features [14–15].

2.2 Feature Extraction

We adopted the idea of extracting attribute information and content information features separately,

and considered extracting features from both perspectives. In Task-A, each question-answer pair has 100 dimensional features. Among these 100-dimensional features, question (QBody field) and answer (CBody field) have 25 dimensional features each. Before extracting the features, all the links in the text were removed, but the html tags were retained. The question-answer also has 50 features. In feature selection, the features selected for Task-A and Task-B are the same.

3 Data Sets

In this paper, experiments were conducted for the English corpus only. For the English corpus, each question has a title and description, and a list consisting of many answers, as shown in Fig. 2. Among them, YES/NO type questions account for about 10% of the total number of questions, which is harder to process using machine learning for Task-B because of the small amount of data. It can be further seen that on average there are 6 responses per question, and for each question specifically, the minimum number of responses is 1 and the maximum is 143. about half of the responses are good, 10% are potentially useful, and the rest are not good. Note that for classification purposes, Bad is a heterogeneous class that consists of 50% Bad, 50% Dialogue, and a small fraction of Non-English and Other. The purpose of breaking Bad down into multiple tags is to consider it for use in other systems. About 40–50% of the CGOLD_YN tags for the responses labeled YES/NO were Yes, with the remaining portion of No and Unsure each accounting for half. However, the number of QGOLD_YN labels in questions labeled YES/NO was higher for Unsure than for No. Overall, the label distribution of CGOLD values is basically similar for the development and test datasets compared with the training dataset, but the label distribution of QGOLD_YN is more different.


```
mainKey:    QID
{
    "QID": "Q2601",
    "QCATEGORY": "Life in Qatar",
    "QDATE": "2010-11-24 14:41:45",
    "QUSERID": "U5424",
    "QTYPE": "GENERAL",
    "QSubject": "from DUBAI to QATAR",
    "QBody": "i am currently working here in dubai and i got an offer from qatar company.. they offered 8000Q/
    "comments": [
        {
            "CID": "Q2601_C1",
            "CUSERID": "U3098",
            "CSubject": "Hi",
            "CBody": "\n If you are single then its ok you can enjoy.",
            "CGOLD": "Potential"
        },
        {
            "CID": "Q2601_C2",
            "CUSERID": "U35",
            "CSubject": "depends",
            "CBody": "depends on where the accommodation is.. how many people will live with you, what kind o
            "CGOLD": "Good"
        },
        {
            "CID": "Q2601_C3",
            "CUSERID": "U554",
            "CSubject": "If the company is from Oil",
            "CBody": "If the company is from Oil and Gas Industry or working for that Industry, then their ac
            "CGOLD": "Good"
        },
        {
            "CID": "Q2601_C4",
            "CUSERID": "U37",
            "CSubject": "Transport in the city is a",
            "CBody": "Transport in the city is a nightmare.",
            "CGOLD": "Good"
        },
        {
            "CID": "Q2601_C5",
```


Figure 2: Question-answer lists

4 Experiments and Results

4.1 Label Development

For the Task-A, there are six categories in the description given in the dataset, i.e., the number of labels is 6, so the labels are numbered from 0 to 5, corresponding to Good, Bad, Potential, Dialogue, Non-English,

and Other. However, by observing the actual data, we found that the tags Non-English and Other did not appear in the corpus. So we set two tagging methods according to this situation, A-full is for tagging 6 categories, and A-sim is for tagging only three categories.

The features used in Task-B are the same as in Task-A, but in terms of questions, so the feature vector for each question is the mean of the feature vectors of the options marked as good in the svm training results under A-sim labeling.

4.2 Parameter Settings

In the process of concrete implementation, this experiment uses libsvm and xgboost as machine learning tools to serve as implementation tools for support vector machine and random forest machine learning algorithms, respectively. And in processing, nltk is utilized to deal with word separation, lexical annotation and named entity recognition, etc. nltk is a powerful third-party library for python, called Natural Language Toolkit, which can easily perform many natural language processing tasks, including word separation, lexical annotation, named entity recognition and syntactic analysis. The question fields and answer fields are processed by the module functions provided by nltk.

After a simple parameter adjustment, both Task-A and Task-B use linear kernel SVM. The parameter-c of Task-B is 1, and the parameter -c of the two labels of Task-A is 0.5. But in random forest, both Task-A and Task-B tasks use multisoftmax as the objective function, the maximum tree depth is 5, the learning rate eta is 0.2, and the number of training rounds is 50 rounds. When evaluating, Task-A is the same as the evaluation script on the official website, and only three categories are considered. The evaluation indicators of Task-A and Task-B are both macro-f1 and accuracy.

4.3 Indicators

The paper evaluates the algorithm performance from the macro-averaged F1-score, the accuracy, etc.

The macro-averaged F1-score is calculated as:

$$\text{macro}-\text{F1}=\frac{\sum_{i=1}^{NumC}\text{F1}_i}{NumC} \tag{1}$$

where $NumC$ is the number of class in test set, F1_i is the F1 value for class i in test set. F1-good, F1-bad indicators are Detail Class F1-score，And F1 value is calculated as:

$$\text{F1}=\frac{2\times P\times R}{P+R} \tag{2}$$

where P and R is the precision and recall of test results for a class in test set. When calculating F1-good and F1-bad, P and R are the precision and recall when the categories are good and bad, respectively.

The accuracy is calculated as:

$$Accuracy=\frac{totalRighNum}{totalTestCaseNum} \tag{3}$$

4.4 Results

In this paper, Train datasets were used for training, and natural language processing was used for extracting features. SVM, Random Forest model was trained on Train dataset and tested on Devel and Test datasets. The experimental statistics of two indicators, Accuracy and Macro-F1. Table 1 shows the results of Task-A. A-sim classifies the answers into Good, Potential and Bad labels. Dialog, Non-English and Other are treated as Bad labels, while A-full classifies the answers into Good, Potential, Bad, Dialog, Non-English and Other labels. Table 2 shows the experimental results of Task-B.

Table 1: Experimental results on Task-A

	Macro-F1	Accuracy	F1-Good	F1-Bad
A-full-svm-dev	47.39%	67.42%	76.05%	61.66%
A-full-svm-test	49.41%	70.73%	76.70%	67.99%
A-Sim-svm-dev	48.46%	69.36%	77.60%	68.49%
A-Sim-svm-test	50.56%	72.77%	77.86%	73.42%
A-full-rf-dev	48.51%	68.94%	77.12%	64.96%
A-full-rf-test	49.57%	70.70%	77.08%	68.49%
A-Sim-rf-dev	48.77%	69.85%	77.40%	68.79%
A-Sim-rf-test	50.24%	72.27%	77.16%	73.17%

Table 2: Experimental results on Task-B

	Macro-F1	Accuracy	F1-Yes	F1-Unsure	F1-No
B-svm-dev	54.32%	55.80%	66.67%	40.00%	50.00%
B-svm-test	43.39%	51.72%	62.86%	46.15%	20.00%
B-rf-dev	52.77%	55.88%	68.75%	45.45%	42.86%
B-rf-test	57.50%	68.97%	82.35%	58.82%	28.57%

5 Discussion

The results of Xgboost on Task-B are better than the SVM method on the test dataset, but the difference is not much on the dev dataset. It may be an accident due to the smaller test set of Task-B. In Task-A, when setting labels, we used two ways of label setting, the first one only considers good, bad and potential, and adopts one label bad, dialog, non-English, and other, and the second way sets 6 labels, and from the experimental results, the former method of setting labels works slightly better.

After the results were obtained, the data were compared with the results of some related articles also for this dataset, and the results of Task-A and Task-B for the comparison are shown in Table 3 and Table 4. In the tables, Hou represents the method of reference [16], Quan represents the method of reference [17], and their method was the best one at the time of measuring the competition, although they did not do Task-B. Belinkov represents the method of reference [18], and this method was the best one at the time of measuring the competition Task-B worked the best. In [16], ensemble learning and hierarchical classification were proposed for answer selection. Tran et al. [17] combined 16 features belong to 5 groups to predict answer quality. And Belinkov et al. [18] described their experience using continuous word and phrase vectors as a source of features. Since Hou et al. [16–18] also processed Task-A and Task-B of this dataset and gave their measured indicators separately, the results of our experiments were compared with theirs.

Table 3: Horizontal comparison on Task-A

	Accuracy	Macro-F1	F1-Bad	F1-Good
Task-A-SVM	72.77%	50.56%	73.42%	77.86%
Task-A-RF	72.27%	50.24%	73.17%	77.16%
Quan	72.52%	57.29%	78.96%	78.96%
Hou	69.43%	69.43%	72.58%	78.87%
Belinkov	70.45%	49.54%	-	-

Table 4: Horizontal comparison on Task-B

	Accuracy	Macro-F1	F1-No	F1-Unsure	F1-Yes
Task-B-SVM	51.72%	43.39%	20.00%	46.15%	62.86%
Task-B-RF	68.97%	57.50%	28.57%	58.82%	82.35%
Hou	64.00%	53.60%	36.36%	44.44%	80.00%
Belinkov	72.00%	63.70%	-	-	-

6 Conclusions

Based on the above comparison, we can conclude that the difference between the Random Forest and SVM methods is not very big in Task-A, while in Task-B, Random Forest has better results. In the comparison with the previous methods, the Accuracy of our Random Forest and SVM methods in Task-B and Task-A is higher than that of the method used by Hou, so we can conclude that our Random Forest and SVM methods exceed the integrated multi-classifier learning method and the hierarchical classification method used by Hou, and there is a slight difference.

So there are still many places that can be improved. For example, this experiment only uses traditional machine learning methods, not deep learning algorithms, if there is enough time, the experiment can be conducted later using deep network models, such as CNN; this experiment only randomly selected some hyperparameters of the model, without setting a fine-grained analysis of the hyperparameter selection scheme, later the hyperparameters can be adjusted; there is also the feature extraction aspect of this experiment can be improved, later the experiment can add more features from different perspectives. Later, more features can be added from different angles for the experiments.

Funding Statement: The authors received no specific funding for this study.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] Y. Xiao, M. Wang, W. Wei, M. Khabsa and A. H. Awadallah, "Adversarial training for community question answer selection based on multi-scale matching," *AAAI*, Honolulu, Hawaii, USA, pp. 395–402, 2019.

[2] J. Hu, Q. Fang, S. Qian and C. Xu, "Multi-modal attentive graph pooling model for community question answer matching," in *Proc. of the 28th ACM Int. Conf. on Multimedia*, Seattle, WA, USA, pp. 3505–3513, 2020.

[3] S. Lyu, W. Ouyang, Y. Wang, H. Shen and X. Cheng, "What we vote for? answer selection from user expertise view in community question answering," in *The World Wide Web Conf. on WWW'19*, New York, NY, USA, pp. 1198–1209, 2019.

[4] A. Sutedi, M. A. Bijaksana and A. Romadhony, "Answer selection using word alignment based on part of speech tagging in community question answering," *Journal of Physics: Conference Series*, vol. 1192, 012035, 2019.

[5] D. Bogdanova, C. dos Santos, L. Barbosa and B. Zadrozny, "Detecting semantically equivalent questions in online user forums," in *Proc. of the Nineteenth Conf. on Computational Natural Language Learning*, Beijing, China, pp. 123–131, 2015.

[6] D. Charlet and G. Damnati, "Simbow at SemEval-2017 task 3: Soft-cosine semantic similarity between questions for community question answering," in *Proc. of the 11th Int. Workshop on Semantic Evaluation*, Vancouver, Canada, pp. 315–319, 2017.

[7] W. Chan, X. Zhou, W. Wang and T. S. Chua, "Community answer summarization for multi-sentence question with group l1 regularization," in *Proc. of the 50th Annual Meeting of the Association for Computational Linguistics*, Jeju Island, Korea, pp. 582–591, 2012.

[8] W. Zhang, Z. Chen, C. Dong, W. Wang, H. Zha *et al.,* "Graph-based tri-attention network for answer ranking in CQA." in *Proc. of the AAAI Conf. on Artificial Intelligence*, Vancouver, British Columbia, Canada, vol. 35, no.

16, pp. 14463–14471, 2021.

[9] R. Zanibbi, B. Mansouri, A. Agarwal and D. W. Oard, "ARQMath: A new benchmark for math-aware CQA and math formula retrieval," *SIGIR Forum*, vol. 54, no. 2, pp. 1–9, 2020.

[10] P. Nakov, L. Màrquez, W. Magdy, A. Moschitti, J. Glass *et al.,* "SemEval-2015 task 3: Answer selection in community question answering," *arXiv:1911.11403*, 2019.

[11] A. Karim, A. Azhari, M. Shahroz, S. B. Belhaouri and K. Mustofa, "LDSVM: Leukemia cancer classification using machine learning," *Computers, Materials & Continua*, vol. 71, no. 2, pp. 3887–3903, 2022.

[12] J. D. Lee, H. S. Cha, S. Rathore and J. H. Park, "M-IDM: A multi-classification based intrusion detection model in healthcare IoT," *Computers, Materials & Continua*, vol. 67, no. 2, pp. 1537–1553, 2021.

[13] Y. Xue, H. Zhu and J. Y. Liang, "Adaptive crossover operator based multi-objective binary genetic algorithm for feature selection in classification," *Knowledge-Based Systems*, vol. 227, no. 5, pp. 1–9, 2021.

[14] R. Chen, L. Pan, Y. Zhou and Q. Lei, "Image retrieval based on deep feature extraction and reduction with improved CNN and PCA," *Journal of Information Hiding and Privacy Protection*, vol. 2, no. 2, pp. 67–76, 2020.

[15] M. Zaffar, M. A. Hashmani, R. Habib, K. Quraishi, M. Irfan *et al.,* "A hybrid feature selection framework for predicting students performance," *Computers, Materials & Continua*, vol. 70, no. 1, pp. 1893–1920, 2022.

[16] Y. Hou, C. Tan, X. Wang, Y. Zhang, J. Xu *et al.,* "HITSZ-ICRC: Exploiting classification approach for answer selection in community question answering," in *Proc. of the 9th Int. Workshop on Semantic Evaluation*, Denver, Colorado, pp. 196–202, 2015.

[17] Q. H. Tran, V. Tran, T. Vu, M. Nguyen and S. Bao Pham, "JAIST: combining multiple features for answer selection in community question answering," in *Proc. of the 9th Int. Workshop on Semantic Evaluation*, Denver, Colorado, pp. 215–219, 2015.

[18] Y. Belinkov, M. Mohtarami, S. Cyphers and J. Glass, "VectorSLU: A continuous word vector approach to answer selection in community question answering systems," in *Proc. of the 9th Int. Workshop on Semantic Evaluation*, Denver, Colorado, pp. 282–287, 2015.