Predicting Accepted Answer for StackOverflow Questions

Abhishek Kanike
Indiana University
School of Informatics
Computing and Engineering (SICE)
abkanike@iu.edu

Preetham Kowshik
Indiana University
School of Informatics
Computing and Engineering (SICE)
pkowshik@iu.edu

Abstract

Community based question answering websites provide an excellent platform where user can ask and answer questions in various categories. A common challenge with such forums is that no one particular answer is adjudged as a best answer. In this project we attempt to address this problem of predicting an ideal answer from a given set of answers to a question. The fundamental task includes identifying influential features measuring association between question and answer. We carried out experiments using machine learning techniques to extract features related to contextual information which affects the quality of answer.

1 Introduction

With the evolution of the Internet, users all over the world are able to collaborate to form a common platform on the web where in they can ask questions and answer topics pertaining to their interests. Collectively these platforms are a wide spectrum of crowd sourced and administered contents covering a wide range of topics like Science Mathematics, Computer Science to commerce, art and finances. Such community based question answering platform includes Quora, Reddit, Stackoverflow, Yahoo answers and many more.

People may ask any type of questions and then wait for someone else to answer the question. If multiple answers are provided, the asker can select the most suitable one, which is called the accepted answer/the best answer or put the question into the voting stage to gain evaluation on this answer from other users. Also, for each answer users can comment to further dicuss about it. Not every asker always selects the best answer for his/her question. This could be simply due to lack of action, or due to the difficulties in deciding on the best answer. As a result, many questions are left as “not answered”.

Not answered questions do not facilitate the purpose of knowledge exchange since many users would be reluctant to use this information given that these questions are left unanswered. Furthermore, some of the websites remove the questions which are unanswered resulting in loss of information or knowledge. In order to address these problems we try to identify an answer which will be most likely accepted.

2 Related Work

Previous research in Community Question Answer platforms, used features such as number of upvotes to predict the quality of answers. Few researchers conducted a survey and learned that there was a correlation between posting a high quality question and getting a high quality answer. They found that question related features such as tags, length of the question, presence of examples enhanced user understanding of the question.

Another research performed investigation on StackExchange community and found that, users do not evaluate answers based on any criteria such as cognitive Heuristic - space of this answer or the order of it in the answer.

This paper investigates low quality answers in question answering communities. They ignore the question type as a feature that decides the attributes that should be exist to determine the answer to existing question.

While the work has provided useful insight into quality - a measure of CQA content these experimentations use only social features such as user rating and authority but ignored the textual features or content-appraisal features in their study. Also, in few previous work, they extracted features which contain information from the questions, the answers, and the users. But there is no consideration on the relationship between the answers and the questions.

3 Datset

The dataset consists of posts from Stackoverflow website which can be obtained in .csv format. from StackExchange archives or through Google Bigquery. At present, Stackoverflow consists of over 13,000,000 questions and 25,000,000 answers with over 45,000,000 comments. We decided to take a subset of these posts around 60,000 questions which were tagged as "python" or "python-2.x" or "python-3.x". As the model features varies among the communities we chose python questions in our case study. We composed the following sql query to fetch "question id", "question title", "question body", "answer body", "answerer score", "user reputation" and "comment body".

4 Methods

We chose five models viz. Bernoulli's Naive Bayes, Decision Tree, Adaboost, Random Forest and Neural Network Classifiers. In order to measure the content based features a preprocessing step was essential.

4.1 Preprocessing

Readability - Since the data was taken from Stackoverflow, questions and answers often contained information such as "code" and "hyperlinks" . As a data preprocessing step, such non human readable format tags are removed an replaced by a constant factor : "A positive code/user link" which is a 100% readable statement.

4.2 Features selection

- Features from the answer body (Readability): The quality of an answer to a question depends on how structured the answer is described or how readable is the answer. Intuitively, this parameter becomes an essential feature in deciding the contextual measure of the answer. Readability is calculated using the standard Flesch-Kincaid readability score. Below (Figure 1.) is the formula

$$206.835 - 1.015\left(\frac{\text{total words}}{\text{total sentences}}\right) - 84.6\left(\frac{\text{total syllables}}{\text{total words}}\right)$$

Figure 1: Readability Formula

- Feature as answer context: Similarity : Similarity Measure is a real-valued function that quantifies the similarity between two objects. Both question and answer are converted into TF-IDF vectors. Term Frequency-Inverse Document Frequency, is a numerical statistic that is intended to reflect how important a word is to a document in a collection or corpus. Before converting into TF-IDF vectors all the html tags in both question and answer body is removed and stop words were removed from the document. Cosine Similarity is used to measure the similarity between question vector and answer vector.

- Feature from StackOverflow metrics: Answer Score and Answerer Reputation - Quite often the answer with more number of votes tends to have higher chances of classifying as best answer. Also, an answerer with high reputation would be considered as expert in the domain. As a result, these two features would also help in determining the selection of answer as best answer.

- Feature as a Question-Answer Relationship: Faster the question is answered, higher chances the asker will read and opt as best answer. Hence the time lag between the postings of question and answer is selected as feature to the model.

- Feature as answer's comment: A new feature is extracted as the answer comment score. Sentiment analysis is added to measure the degree of the user satisfaction to an answer. Textblob sentiment analyzer package is used to measure the polarity(-1, 1) of the comment added to an answer.

4.3 Classification

We model a prediction system as a binary classifier, which classifies an answer as an "accepted answer" or "not an accepted answer".

4.3.1 Gaussian Naive Bayes

Gaussian Naive Bayes is useful when features which take up continues values. Gaussian Naive Bayes makes an assumption that the continuous values associated with each class are distributed according to a Gaussian distribution. We used scikit-learns GaussianNB package to train and make predictions.

4.3.2 Adaboost

The idea behind Adaboost is that a set of weak classifiers put together can give a strong classifier. The weighted vote of the each weak classifiers is taken during predictions. We used scikit-learns Adaboost package to train and do predictions. We used decision trees with depth 2 as weak classifiers with n_estimator as 200.

4.3.3 Random Forest

Random forests or random decision forests[1][2] are an ensemble learning method for classification that operate by constructing a large number of decision trees during training and classifiying based on maximum votes among the decision trees. Random decision forests correct for decision trees' habit of overfitting to their training set. Scikit-learns RandomForestClassifier package Is used to train and make predictions. We set the number of decision trees to 7.

4.3.4 Neural Networks

MLPClassifier : Multilayer Perceptron Classifier is a supervised learning algorithm which is based on a collection of connected units or nodes called artificial neurons. These classifier build a network of neurons consisting of many layers and each layer having many activation units(neurons).

Scikit-learns MLPClassifier package Is used to train and make predictions. The network consists of 1 hidden layer with 5 activation units and logistic function as activation function. The solver for weight optimization is lbfgs. The regularization parameter was set to 1e-5 .

5 Evaluation Metric

We consider F1 score as our measure of test accuracy, which considers precision and recall to compute the score. Precision is calculated by counting the number of true positive results divided by total number of positive results. Recall parameter is calculated as number of correct positive results divided by number of positive results. And finally F1 score is computed as harmonic mean of precision and recall.

$$F_1 = 2 \cdot \frac{1}{\frac{1}{\mathrm{recall}} + \frac{1}{\mathrm{precision}}} = 2 \cdot \frac{\mathrm{precision} \cdot \mathrm{recall}}{\mathrm{precision} + \mathrm{recall}}.$$

Figure 2: F1 score formula

6 Results

We followed cross validation hypothesis technique. Around 59,000 posts are taken as training data. Remaining 1000 posts were taken as test data. We plotted the correlation among the features which we selected as part of our principal component analysis and found that none of the features are independent of one another. Following (Figure 3) is a heat map of correlation between features.

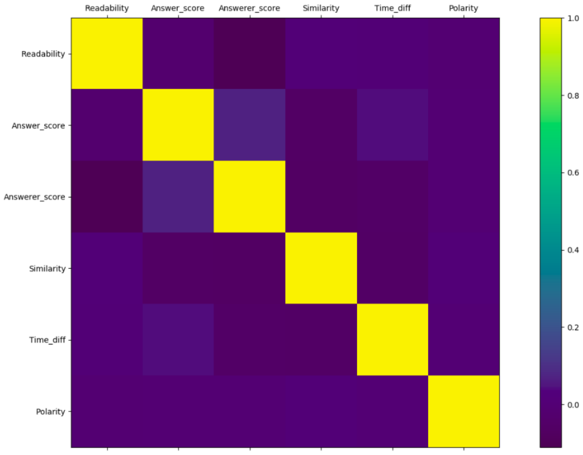

Figure 3: Correlation Heat Map

Initially we ran all our 5 models with all features and results are shown in Figure 4. Naive Bayes classifier gave the best accuracy of 80% among the other classifiers.

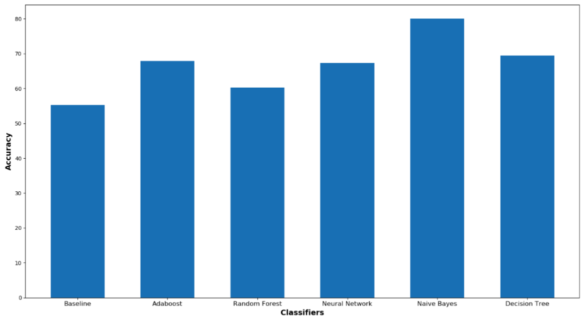

Figure 4: All feature accuracy

Furthermore, we also tried removing features of StackOverflow metrics - Answer Score and Answerer Score in-order to understand the effect of contextual features and results are shown in Figure 5.

We can see that all the classifiers gave higher accuracies when these features were dropped.Naive Bayes, Neural Networks and Decision Trees gave accuracies of 80%.

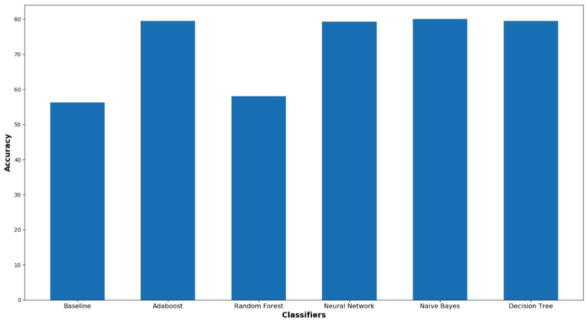

Figure 5: Accuracy after dropping question based features

Summarizing the result we see that apart from answer specific features such as Answer Score and Answerer Score,contextual features such as Readability and Similarity play an important role in classifying an answer as an Accepted Answer. Following is the F1 scores of all classifier with (Table 1) and without (Table 2) answer based features.

Classifiers	**F1 Score**
Baseline Classifiers	0.668
AdaBoost	0.778
RandomForest	0.690
Neural Network	0.789
Naive Bayes	0.887
Decision Trees	0.797

Table 1: F1 Score with all features

Classifiers	**F1 Score**
Baseline Classifiers	0.668
AdaBoost	0.878
RandomForest	0.724
Neural Network	0.884
Naive Bayes	0.887
Decision Trees	0.879

Table 2: F1 Score after dropping answer based feature

7 Conclusion

In this project we try to predict if the answer will be selected as the best answer by the asker or not based on StackOverflow posts. We considered contextual features which are the features that address the relation between the question and answers, time lag between the posting time of question and the time of the response content and features which are the analytical features of the answer itself as well as sentimental analysis of the comments. We found that contextual features play a major role in prediction decisions.

References

[1] Qiongjie Tian and Peng Zhang and Baoxin L *Towards Predicting the Best Answers in Community-Based Question-Answering Services*. Proceedings of the Seventh International AAAI Conference on Weblogs and Social Media.

[2] Dalia Elalfy and Walaa Gad and Rasha Ismail. *hybrid model to predict best answers in question answering communities* The 10th ACM/IEEE International Symposium

[3] Fabio Calefato, Filippo Lanubile and Nicole Novielli. *Moving to Stack Overflow: Best-Answer Prediction in Legacy Developer Forums* The 10th ACM/IEEE International Symposium

[4] F1 Score- Precision and Recall. `https://en.wikipedia.org/wiki/F1_score`