An Entity-Based approach to Answering Recurrent and Non-Recurrent Questions with Past Answers

Anietie Andy
Howard University
anietie.andy@bison.howard.edu

Mugizi Rwebangira
Howard University
rweba@scs.howard.edu

Satoshi Sekine
New York University
sekine@cs.nyu.edu

Abstract

Community question answering (CQA) systems such as Yahoo! Answers allow registered-users to ask and answer questions in various question categories. However, a significant percentage of asked questions in Yahoo! Answers are unanswered. In this paper, we propose to reduce this percentage by reusing answers to past resolved questions from the site. Specifically, we propose to satisfy unanswered questions in entity rich categories by searching for and reusing the best answers to past resolved questions with shared needs. For unanswered questions that do not have a past resolved question with a shared need, we propose to use the best answer to a past resolved question with similar needs. Our experiments on a Yahoo! Answers dataset shows that our approach retrieves most of the past resolved questions that have shared or similar needs to unanswered questions.

1 Introduction

Community question answering (CQA) systems such as Yahoo! Answers are online systems that allow signed-in users to ask, answer, and view questions and answers in a predetermined number of question categories. In Yahoo! Answers, there are two parts to a question: (I) the title - a brief description of the question, and (II) the content - a detailed description of the question (Dror et al., 2011). Despite the active user participation in Yahoo! Answers, a significant percentage of questions remain unanswered (Li and King, 2010). An analysis of Yahoo! Answers data showed that 15% of questions did not receive any answer; however, approximately 25% of questions, at the title-level, in certain Yahoo! Answers categories were recurrent (Shtok et al., 2012), thereby showing the potential of reusing the best answers to past resolved questions to satisfy unanswered questions, with shared needs. Some unanswered questions do not have a past resolved question with a shared need. For example, given the unanswered question *"How can one win a trip to 2006 FIFA World Cup in Germany?"*, the following past resolved question could be recommended, *"How do I buy FIFA 2006 tickets in US?"*. These two questions do not have a shared need but they do have a similar need, namely *"attending the FIFA world cup"*.

In this paper we claim that using cosine similarity with an entity-linking and knowledge base (KB) approach in question categories with high entity usage retrieves most of the past resolved questions with shared or similar needs to unanswered questions. We investigated this claim by labelling a sample dataset of 50 question pairs from the *Sports* and *Entertainment & Music* categories, that exhibited a shared need. We chose these question categories because of the prevalent use of named entities and their variations. Each question pair was associated with a label described below:

This work is licensed under a Creative Commons Attribution 4.0 International Licence. Licence details: http://creativecommons.org/licenses/by/4.0/

Proceedings of the Open Knowledge Base and Question Answering (OKBQA) Workshop,
pages 39–43, Osaka, Japan, December 11 2016.

- *Potential answer*: given a question pair, (Q_{given},[Q_{past}, *Answer*]), *Answer* is a ”potential answer” if it can be used to satisfy Q_{given}.
- *Similar question*: Q_{past} is similar to Q_{given} if they both refer to the same topic[1], but the answer to Q_{past} cannot be used to satisfy Q_{given}.
- *Related question:* Q_{past} is related to Q_{given} if it contains a common entity as Q_{given}, but refers to a different topic[1] from Q_{given}.

We used equation 1 below to calculate the entity ratio of the sample question pair dataset.

$$2 * NC/(NQ1 + NQ2) \tag{1}$$

where, *NQ1* is the number of entities in the unanswered question, *NQ2* is the number of entities in the past resolved question, and *NC* is the number of common entities in both questions. Figure 1 is a plot of the cosine similarity and entity ratio overlap of the sample dataset. It shows that the *potential answer* question pairs have a higher cosine similarity and can be distinguished from *similar* and *related* question pairs. However, *similar* and *related* question pairs are not easily distinguishable. Our proposed algorithm will aim to distinguish between these question pairs.

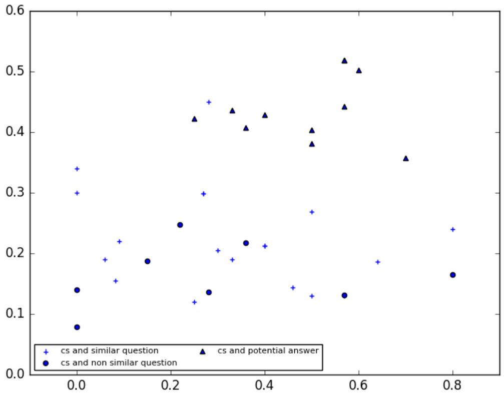

Figure 1: Cosine similarity and number of similar entities in question pair

From this sample dataset, we noticed that the higher the number of common entities or entity variations in a question pair, the easier it is to use cosine similarity to distinguish the question pair categories i.e. *potential answers, similar questions, related questions*. entities in a KB (Guo et al., 2013), from the title and content sections of questions.

The key contribution of this paper is to propose an entity-based algorithm to reduce the number of unanswered questions in entity rich question categories by recommending the best answer to past resolved questions with shared needs to a an unanswered question, if it exists, otherwise recommend past resolved questions with similar needs.

2 Cosine similarity and Entity-based approach

Cosine similarity has been widely used to find similar questions and sentences (Salton and McGill , 1986). However, due to the lack of uniformity in CQA users writing styles (Khalid et al., 2008) and the

[1] A topic is an activity or event along with all directly related events and activities. A question is on topic when it discusses events and activities that are directly connected to the topic’s seminal event

frequent use of entity name variations in question categories with high entity and entity variation usage, similar questions could have a low cosine similarity. Hence we propose an entity-based algorithm to satisfy unanswered questions by reusing the best answer to past resolved questions with either a shared or similar need to the unanswered questions.

The proposed algorithm has two stages:

2.1 Stage One

In this stage, we select a past resolved question as a candidate similar question to a given question if the ”question-title” section of both questions have a cosine similarity greater than a threshold, (0.08) and the ”question-title” + ”question-content” of both questions contain one or more common entities, entity variations, or KB anchor phrases.

2.2 Stage Two

In stage two, the answer to the candidate past resolved question selected in stage one is assessed as a valid answer to the given question (Shtok et al., 2012). Features are extracted from the unanswered question and past resolved question and we train a classifier that validates whether the best answer to a past resolved question can be used to satisfy an unanswered question.

2.2.1 Features

Entity and KB features: We collect the following entity and KB statistics from the question pair: the number of common entities, the number of commom entity disambiguations, the number of common KB anchor phrases, the number of common words and phrases in the the question pair. These features measure the similarity of the entities and words in the question pair.

Surface level features: We extract the following statistics from the question pair: maximal IDF within all terms in the text, minimal IDF, average IDF. Various IDF statistics over query terms have been found to be correlated to query difficulty in ad-hoc retrieval (Hauff et al., 2008; He and Ounis, 2008). We extract the difference between the word-length of $\mathrm{Q}_{\mathrm{given}}$ and $\mathrm{Q}_{\mathrm{past}}$ and the stopword count. These features try to identify the focus, complexity and informativeness of the text (Shtok et al., 2012). We also, extract bigrams and trigrams from the question pair.

Lexical Analysis: We classify words in the question pair into their parts-of-speech and extract the number of matching nouns, verbs, and adjectives, if they exist.

Cosine similarity: Cosine similarity is popularly used to show the similarity between documents (Salton and McGill , 1986). We calculate the cosine similarity of the ”question-title” and ”question-title” + ”question-content” of the question pair.

2.2.2 *Classifier model:*

For learning, we used the Random Forest algorithm with its default parameter settings as implemented by Weka machine learning workbench (Shtok et al., 2012; Jeon et al., 2009) with a 5-fold cross validation.

3 Experiments

3.1 Data Construction and Labeling

The dataset used to train and evaluate our system contains questions pairs, ($\mathrm{Q}_{\mathrm{given}}$,[$\mathrm{Q}_{\mathrm{past}}$, *Answer*]), with labels *potential answers, similar question*, and *related question* , described in *section 1.*

To generate the given question and past resolved question pair, we selected 3000 and 5000 past resolved questions from the *Sports* and *Entertainment & Music* question categories respectively from the language data section of Yahoo labs Webscope^{TM} dataset,and Yahoo! Answers dataset (Chang et al., 2008). Given a question from the selected dataset of past resolved questions, we selected a candidate similar question from the selected dataset if it had a common named entity, entity variation or anchor phrase as the given question and a cosine similarity ($>$ 0.08) . We had three independent reviewers label the question pairs as either a *potential answer*, *similar question*, or *related question*. We selected a question pair if at least two of the reviewers agreed on the question pair label. We calculated the degree of agreement between the reviewers by using Fleiss' kappa [1]. The kappa of our reviewers was *0.448*.

We annotated 400 question pairs from the *Sports* and *Entertainment & Music* question categories and the number of question pairs assigned to each label is as follows: 208 *Potential answers* , 136 *Similar questions*, and 56 *Related question*. We intend to make this dataset available to the research community.

4 Results

We tested two state of the art entity linking tools, AlchemyAPI (Turian, 2013) and Babelfy (Moro et al., 2014) on a sample dataset of questions from the *Sports* and *Entertainment & Music* categories of Yahoo! Answers and AlchemyAPI identified more named entities, entity disambiguations, and KB anchor phrases in the sample dataset. We used AlchemyAPI to extract named entities, named entity disambiguations, and KB anchor phrases from a given question and a past resolved question. AlchemyAPI extracts anchor phrases from the following KBs, dbpedia and freebase.

We carried out experiments using the proposed algorithm on two classes of classifiers, Random Forest and SVM. Table 1 shows that Random Forest performed better than SVM by correctly predicting 87% of the question pair.

Classifier	Percentage of correct predictions
SVM	85%
Random Forest	87%

Table 1: Percentage of correctly predicted question pairs by clasifiers

We also tested the proposed algorithm on *similar* and *related* question pairs. Our aim was to see if our algorithm will distinguish the *similar* questions from the *related* questions. Table 2 shows that the proposed algorithm correctly predicted 77% of the *similar question* pairs. We also tested the proposed algorithm on *potential answer* question pairs and the proposed algorithm predicted 80% of the *potential answer* pairs.

Question Category	Percentage of correct predictions
Potential answers	80%
Similar questions	77%

Table 2: Percentage of correctly predicted potential answers and similar question question pair

5 Conclusion

In this paper, we showed that using cosine similarity and exploiting named entities, entity variations, and KB anchor phrases is effective in searching for past resolved questions in entity rich categories.

[1] Fleiss kappa assesses the reliability of the agreement between the raters when assigning labels to the question pairs.

Acknowledgements

This work was supported by the Center for Science of Information (CSoI), an NSF Science and Technology Center, under grant agreement CCF-0939370.

References

Shtok, Anna and Dror, Gideon and Maarek, Yoelle and Szpektor, Idan 2012. *Learning from the past: answering new questions with past answers Proceedings of the 21st international conference on World Wide Web* 759–768

Dror, Gideon and Maarek, Yoelle and Szpektor, Idan 2011. *I want to answer; who has a question?: Yahoo! answers recommender system Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining* 1109–1117

Khalid, Mahboob Alam and Jijkoun, Valentin and De Rijke, Maarten 2008. *Advances in Information Retrieval Springer* 705–710

Chang, Ming-Wei and Ratinov, Lev-Arie and Roth, Dan and Srikumar, Vivek 2008. *Importance of Semantic Representation: Dataless Classification* In *proceedings AAAI* 830–835

Salton, Gerard and McGill, Michael J 1986. *Introduction to modern information retrieval* McGraw-Hill, Inc.

Guo, Stephen and Chang, Ming-Wei and Kiciman, Emre 2013. *To Link or Not to Link? A Study on End-to-End Tweet Entity Linking HLT-NAACL* 1020–1030

Jeon, Jiwoon and Croft, W Bruce and Lee, Joon Ho 2009. *The WEKA data mining software: an update* Journal ACM SIGKDD explorations newsletter, 11(1):10–18

Li, Baichuan and King, Irwin 2010. *Routing questions to appropriate answerers in community question answering services.* In *Proceedings of the 19th ACM international conference on Information and knowledge management* 1585–1588

Hauff, Claudia and Hiemstra, Djoerd and de Jong, Franciska 2008. *A survey of pre-retrieval query performance predictors.* In *Proceedings of the 17th ACM conference on Information and knowledge management* 1419–1420

He, Ben and Ounis, Iadh 2004. *Inferring query performance using pre-retrieval predictors.* In *Proceedings of International Symposium on String Processing and Information Retrieval* 43–54

Guo, Stephen and Chang, Ming-Wei and Kiciman, Emre 2013. *To Link or Not to Link? A Study on End-to-End Tweet Entity Linking.* In *Proceedings of HLT-NAACL* 1020–1030

Moro, Andrea and Raganato, Alessandro and Navigli, Roberto 2014. *Entity linking meets word sense disambiguation: a unified approach* Journal Transactions of the Association for Computational Linguistics, 2:231–244

Turian, Joseph 2013. *Using AlchemyAPI for Enterprise-Grade Text Analysis.* Technical report, AlchemyAPI