The 2017 Conference on Computational Linguistics and Speech Processing
ROCLING 2017, pp. 295-307

Opinion Target Extraction for Student Course Feedback

Janaka Chathuranga, Shanika Ediriweera,
Pranidhith Munasinghe, Ravindu Hasantha and Surangika Ranathunga
Department of Computer Science and Engineering
University of Moratuwa, Katubedda 10400, Sri Lanka
{janaka.13, shanika.13, pranidhith.13, ravindu.13, surangika}@cse.mrt.ac.lk

Abstract

Student feedback is an essential part of the instructor - student relationship. Traditionally student feedback is manually summarized by instructors, which is time consuming. Automatic student feedback summarization provides a potential solution to this. For summarizing student feedback, first, the opinion targets should be identified and extracted. In this context, opinion targets such as “lecture slides”, “teaching style” are the important key points in the feedback that the students have shown their sentiment towards. In this paper, we focus on the opinion target extraction task of general student feedback. We model this problem as an information extraction task and extract opinion targets using a Conditional Random Fields (CRF) classifier. Our results show that this classifier outperforms the state-of-the-art techniques for student feedback summarization.

Keywords: Student Feedback Summarization; Opinion target Extraction, Conditional Random Fields

1. Introduction

Student feedback is used widely in present in order to enhance the quality of teaching and learning. Feedback is collected from students as online forms as well as handwritten documents. Since it takes a considerable effort to read and understand all the feedback given by the students, the best way is to read all the feedback and create a summary that covers all the aspects of all the feedback given.

Although many lecturers collect student feedback, comments written by students are not summarized. If a lecturer wants to get a summary of these comments, the lecturer has to manually read and summarize these comments. Manual summarization is not scalable; in a large class with more than few hundred students, it is going to be a tedious and rigorous task. Thus, a system to summarize all student feedback and giving an overall summary by categorizing students' sentiments towards different aspects of the lecture will be very useful for teachers, lecturers, schools, universities, and the education systems as a whole.

Research done in this area so far has focused only on using student feedback collected using reflective prompts [1]. With a reflective prompt, student feedback is collected by giving a specific question (prompt). For an example, a prompt such as "What are the most interesting topics of today's lecture?" is considered as a reflective prompt. In reflective prompts, the prompt decides the opinion of the feedback: positive or negative. Opinion for different aspects in student feedback cannot be measured in this approach.

In this paper, we focus on general student feedback. General feedback means that the feedback is collected using a general prompt (example: "Give feedback on today's lecture"), rather than a specific prompt where the prompt suggests the sentiment of the feedback.

Our system contains three parts:

(1) Identifying and extracting all the opinion targets in the given feedback

(2) Clustering all the targets into unique categories

(3) Determining the sentimental polarity of the targets and getting a statistic of polarity for each target cluster.

Here in this paper, we only focus on the first part of our solution, which is identifying and extracting the opinion targets from student feedback. First, we undergo a data-preprocessing step to fix errors in the dataset. Then we annotate the targets using our own annotation schema into Beginning, Inside, Outside (BIO) tags, and then we use a Conditional Random Field

(CRF) classifier as a supervised approach to extract the opinion targets.

To the best of our knowledge, there is no prior research done on general feedback summarization. Thus, we have no viable baseline, nor an annotated data set. Therefore, we have created the baseline for our system using the supervised approach used by Luo et al. [1], which was done using reflective prompts. We show that for this general feedback data set, our classifier outperforms the selected baseline system.

Even though it is suggested that deep learning techniques such as Recurrent Neural Networks(RNN)[2] perform well in extracting opinion targets, we are not able to get good results because our dataset is very small with 956 student responses in total with 4428 sentences. In order to use deep learning techniques, we need a much bigger dataset and there is no other general student feedback dataset which suits our purpose.

The rest of the paper is structured as follows; Section 2 overviews previous work and section 3 describes the data used for our experiments. Section 4 describes the details about our approach and the features used. Section 5 describes how the experiment is done and the evaluation results, followed by the conclusion in Section 6.

2. Related Work

There are two general approaches for automatic summarization: extraction and abstraction. Extractive methods work by selecting a subset of existing words, phrases, or sentences in the original text to form the summary. In contrast, abstractive methods build an internal semantic representation and then use natural language generation techniques to create a summary.

Research on student feedback summarization done up to the day has only used extractive methods such as Integer Linear Programming [3], Phrase-based approach, clustering, and ranking approaches [4] to summarize student feedback. These techniques use reflective prompt-based student feedback data sets. That means when acquiring feedback, the students are guided with a specific question such as, “Describe what you found most interesting in today’s class?” In general feedback, the student is not directed towards a specific aspect. Students are given the chance to write anything related to the lecture on their feedback. Therefore, it contains many unexpected, but useful information. Further, general feedback contains more complex content and noise compared to reflective prompt based feedback. Therefore, target extraction on general feedback is more challenging. To the best of our knowledge, there has not been any research done to summarize general student feedback.

The first step in general feedback summarization is opinion target (aspect) extraction. Opinion target (aspect) is an entity that respondents have raised their opinions about.

Aspect extraction has been studied by many researchers in the domain of sentiment analysis. There are two main approaches: supervised and unsupervised.

When supervised approach has been used for opinion target extraction [5], the sequence labeling scheme known as BIO labeling has been commonly used. However, this research is limited to extract only course names and instructor names as entities.

The system [7] that we consider as the baseline for our work also has used a BIO labeling scheme for candidate phrase extraction. A Conditional Random Fields (CRF) [6] classifier is used as the sequence labeler. Since their dataset is responses to reflective prompts, they extract noun phrases as candidate phrases.

Double Propagation method [7] is an unsupervised approach to solve opinion target extraction. The basic idea of this approach is to extract opinion words (or targets) iteratively using known and extracted (in previous iterations) opinion words and targets through the identification of syntactic relations. At the beginning, opinion word is given as a seed word. Thus, it can also be viewed as a semi-supervised method. Improvement of this method has been proposed by Luo et. al [1].

These two approaches hold the promise for the task of extracting opinion targets from student responses for small data sets.

3. Data

In a student feedback summarization task, the first thing is to identify the entities or aspects students have raised their opinions about. Although currently there are datasets containing student feedback collected by asking them a specific reflection prompt (question), a reasonable sized feedback set that contains feedback about almost every aspect of a course is missing. In this work, we created a new dataset in order to fulfill this purpose.

Our data consists of student responses collected from an undergraduate Computer Science and Engineering Course. General responses were collected from 27 Lectures and Workshops . They contain 956 student responses in total with 4428 sentences.

The prompts we used to collect responses were general prompts. Therefore, student had the freedom to write regarding any aspect of the lecture .In addition, there was no sentence limitation for providing feedback.

This feedback consists of many opinionated responses. Each of those responses focuses their opinion towards a target entity, which is called an opinion target. Some opinion targets have both positive and negative opinions towards them. For example, consider the following sentence.

"The lecture slides were **uploaded to Moodle every week** and I think **it would have been much better if you could upload them on Sunday**".

Here the student expresses his opinions about "lecture slides": positive opinion for uploading them every week and negative opinion for not uploading it on Sunday**.**

In our work, we used our own way of annotating student feedback.

That is mainly because of the nature of the data. Data used in previous work [1][3][4] only had opinion targets in them whereas the positive / negative expressions were in the prompt itself. The following cases were identified in responses, which contain both opinion targets and positive/ negative expressions.

In the dataset, many different types of opinion targets and opinion expressons were found.

- Multi word opinion targets

Ex: - I think time and weight for documentation of the project is **too much**.

Opinion target is “time and weight for documentation of the project”.

- Single target, single opinion expression.

Ex: - Lectures were **really good**.

- Single target, multiple opinion expressions.

Ex: - Overall lecture session was **great**, **well organized** and **very helpful**.

Here the target “Overall lecture session” has three positive opinion expressions towards it.

- Single opinion multiple opinion targets

Ex: - Keeping interactions with students, asking questions, giving in-class activities and discussing them within the class were **greatly helpful for me** to develop my oop skills.

A positive opinion is expressed here for all the following aspects/ targets of the lecture: “Keeping interactions”, “asking questions”, “giving in class activities”.

- Ambiguity about which opinion target to take

E.g.: - Both lecturers did a great job on delivering the subject matter.

Here, two aspects can be identified: “Both lecturers” and “delivering the subject matter”. It is difficult to find on which target the opinion is focused on.

We manually annotated 20 feedback files out of 27 using this method. This annotation scheme first identifies sentences or phrases with opinions and then marks the opinion target.

Since we annotated both the target and the opinion towards the target, we had to use unique BIO tags for both target and the opinion expression. Therefore, we used B-T (Beginning-Target) for the beginning of the Target, I-T (Inside-Target) for the inside of the target, B-PO (Beginning-Positive Opinion) for the beginning of the positive opinion expression, I-PO (Inside-Positive Opinion) for the inside of the positive opinion expression, B-NO (Beginning-Negative Opinion) for the beginning of the negative opinion expression, I-NO (Inside-Negative Opinion) for the inside of the negative expression, and O for the outside words that are not annotated.

For example, consider the sentence “Lectures were really good”. This sentence was annotated as shown below:

- Lectures/B-T were/O really/O good/B-PO

4. Aspect extraction

For the task of classification, we choose to use a Conditional Random Fields (CRF) classifier [6]. CRFs are a class of statistical modeling methods often applied in pattern recognition and machine learning and used for structured prediction. CRFs fall into the sequence modeling family. Whereas a discrete classifier predicts a label for a single sample without considering "neighboring" samples, a CRF can take context into account; e.g., the linear chain CRF (which is popular in natural language processing) predicts sequences of labels for sequences of input samples. This has been used for many other sequence labeling tasks such as Named Entity Recognition (NER) as well[8] [9][10].

The CRF labeler is trained using the training data set containing 956 responses.

4.1 Features

As the baseline features we use the features used by Luo et al. [1]. These features are based on sentence syntactic structure and word importance to signal the likelihood of a word being included in the target.

Local Features

- Word trigram within a 5-word window
- Part-of-Speech tag trigram within a 5-word window
- Chunk tag trigram within a 5-word window
- Whether the word is in the prompt

- Whether the word is a stop word

Global Features

- Total number of word occurrences (stemmed)
- Rank of the word's term frequency

These local and global features are used for supervised target extraction. Local features are extracted from one student's response. Global features are extracted using all student responses in one lecture.

4.2 New Features

We increased the accuracy of target extraction by adding following features.

4.2.1 Capitals, Punctuation marks and Numbers (*CPN*)

These features check whether the word is a capital letter, whether the first character is a capital letter, whether all characters are capital letters, whether the word is a punctuation mark, whether all characters are punctuation marks, whether the word contains punctuation marks, whether the word is a number, and whether all characters are numbers. These features are applied as unigram features within a 3-word window.

4.2.2 Word Embedding Features

Previous research [11][12][13] has shown that utilization of unlabeled data can improve the quality of the Named Entity Recognition, which also used a CRF classifier. Therefore, we tried out following word embedding features to improve the target extraction process.

4.2.2.1 Brown Clusters

Brown's algorithm is a hierarchical clustering algorithm that clusters words that have a higher mutual information of bigrams [14]. We created Brown clusters using the given corpus and some other un-annotated feedback data (**this set contains 3970 sentences, which were collected in 37 workshops).** The output of the algorithm is a dendrogram. A path from the root of the dendrogram represents a word and can be encoded with a bit sequence. We used the prefix of the bit sequence as a feature. We used the first 5, 7, 11 bits as three features. Those numbers were discovered by trying different numbers on the same data set. The combination of above numbers gave the best output.

4.2.2.2 Clark Clusters

Clark's algorithm groups words that have similar context distribution and morphological clues starting with the most frequent words [15]. We created 100 clusters using the non-annotated corpus. Clark clusters were used as unigram, bi-gram, tri-gram and 4-gram features within a 9-word window. The window size was determined by trying different window sizes. 9-word window gave best results.

4.2.2.3 Word to vector feature clusters

We trained a word to vector model [16] using the non-annotated data set, and used it to create 100 clusters using k-medoids algorithm. The output was used as a unigram feature within a one-word window.

5. Experiment

We first corrected the spelling mistakes in the dataset using the Bing Spell Check API [17]. Then the data set was annotated according to above described annotation scheme. Annotated data was converted into BIO tags and was used to train the CRF classifier to extract targets. Here CRF is used because our dataset is small and because of that the deep learning techniques cannot be applied on our dataset. Accuracy of the CRF classifier was measured using 10- fold cross validation. Table 1 shows experiment results.

Table 1. Results

Features	**Precision**	**Recall**	**F1**
Baseline	0.76923	0.60437	0.67690
Baseline + CPN	0.76081	0.66788	0.71132
Baseline + Brown	0.74648	0.63174	0.68434
Baseline + Clark	0.79733	0.62991	0.70380
Baseline + Stemmed Word	0.80348	0.61161	0.69454
Baseline + Word2Vec K-medoids	0.76627	0.61939	0.68504
All	0.79566	0.67154	0.72835

When considering the precision and recall, only exact matches were considered. Partial matches were considered as false negatives. CPN has improved the result considerably compared to other features. One of the major reasons could be usage of capital letters in feedback. Some of the mentioned entities did appear at the beginning of the sentence. Further, many targets are named entities, and there is a high probability for them to appear in title case. CPN is much sensitive to title case because it matches whether word contains a capital letter.

Brown clusters, Clark clusters and Word2Vec K-medoids are word embedding features. They provide a cluster representation on words depending on their relative meanings. May be the dataset size being small can be a reason to obtain lower results by Word2Vec K-medoids feature, given that in previous work Word2Vec models were trained on a much larger dataset[18]. Among the word embedding features, Clark clusters has improved the results, but even its improvement is considerably less compared to CPN. Stemmed word feature is also like clustering. For an example, both “Lecture” and “Lectures” will be clustered in to their stemmed word “lecture”. It has a less improvement in F-score compared to Clark clusters but it has improved precision considerably.

In both precision and recall wise, the maximum accuracy came by combining all features but only for the recall, maximum accuracy was obtained by baseline and stemmed word feature added, but it has a relatively lower recall. The evaluation of the result shows that adding more features increases the recall.

6. Conclusion

In this work, we have focused on opinion target extraction task of the general student feedback, which is the first sub task of summarizing student feedback. We used a CRF classifier to address this information extraction task. As the baseline, we used the supervised approach used by Luo et al. [1]. Experimental results show that our method yields better opinion targets extraction performance than this previous work [1], which is done on reflective prompts feedback.

Future work includes the other two subtasks of student feedback summarization process, which are clustering the extracted opinion targets using a suitable clustering algorithm, and identifying the student's sentiment towards the opinion target.

7. References

[1] W. Luo, F. Liu, and D. Litman, “An Improved Phrase-based Approach to Annotating and Summarizing Student Course Responses,” *Proc. 26th Int. Conf. Comput. Linguist.*, pp. 53–63, 2016.

[2] P. Liu, S. Joty, and H. Meng, “Fine-grained Opinion Mining with Recurrent Neural

Networks and Word Embeddings,” *Proc. 2015 Conf. Empir. Methods Nat. Lang. Process.*, no. September, pp. 1433–1443, 2015.

[3] W. Luo, F. Liu, Z. Liu, and D. Litman, “Automatic Summarization of Student Course Feedback,” *North Am. Chapter Assoc. Comput. Linguist.*, no. Duc 2004, pp. 80–85, 2016.

[4] W. Luo and D. Litman, “Summarizing Student Responses to Reflection Prompts,” pp. 1955–1960, 2015.

[5] C. Welch, R. Mihalcea, H. Street, and A. Arbor, “Targeted Sentiment to Understand Student Comments,” *Proc. 26th Int. Conf. Comput. Linguist.*, no. 1, pp. 2471–2481, 2016.

[6] J. Holst, A. L. Szymczak-Workman, K. M. Vignali, A. R. Burton, C. J. Workman, and D. A. A. Vignali, “Generation of T-cell receptor retrogenic mice,” *Nat. Protoc.*, vol. 1, no. 1, pp. 406–417, 2006.

[7] G. Qiu, B. Liu, J. Bu, and C. Chen, “Opinion Word Expansion and Target Extraction through Double Propagation,” *Comput. Linguist.*, vol. 37, no. 1, pp. 9–27, 2011.

[8] C. Lee, Y.-G. Hwang, and M.-G. Jang, “Fine-grained named entity recognition and relation extraction for question answering,” in *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval - SIGIR ’07*, 2007, p. 799.

[9] B. Settles, “Biomedical named entity recognition using conditional random fields and rich feature sets,” in *JNLPBA ’04 Proceedings of the International Joint Workshop on Natural Language Processing in Biomedicine and its Applications*, 2004, pp. 104–107.

[10] a McCallum and W. Li, “Early results for named entity recognition with conditional

random fields," *Proc. CoNLL-2003*, pp. 188–191, 2003.

[11] R. K. Ando and Z. (Yahoo R. Tong, "A Framework for Learning Predictive Structures from Multiple Tasks and Unlabeled Data," *J. Mach. Learn. Res.*, vol. 6, pp. 1817–1853, 2005.

[12] J. Turian, L. Ratinov, Y. Bengio, and J. Turian, "Word Representations: A Simple and General Method for Semi-supervised Learning," *Proc. 48th Annu. Meet. Assoc. Comput. Linguist.*, no. July, pp. 384–394, 2010.

[13] J. Suzuki, H. Isozaki, X. Carreras, and M. Collins, "An empirical study of semi-supervised structured conditional models for dependency parsing," *Conf. Empir. Methods Nat. Lang. Process.*, pp. 551–560, 2009.

[14] P. F. Brown, P. V DeSouza, R. L. Mercer, V. J. Della Pietra, and J. C. Lai, "Class-Based n-gram Models of Natural Language," *Comput. Linguist.*, vol. 18, pp. 467–479, 1992.

[15] A. Clark, "Combining distributional and morphological information for part of speech induction," *Proc. tenth Conf. Eur. chapter Assoc. Comput. Linguist. - EACL '03*, vol. 1, p. 59, 2003.

[16] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient Estimation of Word Representations in Vector Space," pp. 1–12, 2013.

[17] "Microsoft Cognitive Services—Bing Spell Check API | Microsoft Azure." .

[18] S. Poria, E. Cambria, and A. Gelbukh, "Aspect extraction for opinion mining with a deep convolutional neural network," *Knowledge-Based Syst.*, vol. 108, pp. 42–49, 2016.