Sentiment Analysis of Student Feedback Using Machine Learning and Lexicon Based Approaches

Zarmeen Nasim*, Quratulain Rajput† and Sajjad Haider‡
Faculty of Computer Science
Institute of Business Administration (IBA)
Karachi, Pakistan
Email: *znasim@khi.iba.edu.pk, †qrajput@iba.edu.pk, ‡shaider@iba.edu.pk

***Abstract*—This paper presents a combination of machine learning and lexicon-based approaches for sentiment analysis of students feedback. The textual feedback, typically collected towards the end of a semester, provides useful insights into the overall teaching quality and suggests valuable ways for improving teaching methodology. The paper describes a sentiment analysis model trained using TF-IDF and lexicon-based features to analyze the sentiments expressed by students in their textual feedback. A comparative analysis is also conducted between the proposed model and other methods of sentiment analysis. The experimental results suggest that the proposed model performs better than other methods.**

***Keywords*—sentiment analysis; text mining; natural language processing.**

I. Introduction

Evaluation of a class and the instructor by students towards the end of each semester has now become a norm in higher education institutions. The prime purpose of gathering students feedback is to assess and improve the teaching quality. The feedback helps instructors to refine their teaching methodology and enables them to better understand the students perspective. Likert-scale based scores are typically used for instructors evaluation. In this method, a set of questions is presented to students and they are asked to respond to each question using a five (or seven) point scale. The set of questions is prepared to evaluate the course instructor on different factors such as organization of lectures, presentation, punctuality of instructor, counseling, etc.

Along with the Likert-scale based questions, students also provide textual feedback. The textual feedback provides an opportunity to students to highlight certain aspects which are not directly covered by Likert-scale questions. For instance, students may like to point out any specific problem they faced during the semester. The textual feedback also allows students to share suggestions regarding future course offerings. The feedback provides useful insights to both the instructor and the academic administration as the conventional Likert-scale responses are incapable of capturing such aspects.

Sentiment Analysis, also known as opinion mining, aims to identify the orientation of textual contents in terms of positivity/negativity expressed by the author towards a target entity. It is being applied to a wide variety of domains including social media marketing, finance, business intelligence, sociology, politics and various others. The basic task involved in sentiment analysis is to determine the sentiment polarity expressed in a textual content. The sentiment polarity can be represented by discrete labels such as positive, negative, neutral.

This paper aims at identifying the sentiment polarity expressed in a textual feedback by a student. The paper employs a hybrid approach to build the predictive model for sentiment analysis. Furthermore, a comparison of presented methodology with other sentiment analysis tools is also discussed.

The rest of the paper is organized as follows. Section II discusses the related work in the area of sentiment analysis. The proposed methodology of determining sentiment polarity of students feedback is explained in Section III while experimental results are discussed in Section IV. Section V compares the presented methodology with other approaches reported in the literature. Finally, Section VI concludes this paper.

II. Related Work

Sentiment Analysis has been extensively studied during the past few years. The reported work can be broadly classified into three main approaches: (a) machine learning based, (b) lexicon-based and (c) hybrid.

A. Machine Learning based

Machine learning based approaches of sentiment analysis learn a predictive model using the provided training dataset and evaluate the performance of the learned model on the test dataset. It can be further classified into supervised learning and unsupervised learning methods.

Unsupervised learning methods do not require dataset to be annotated with true sentiment labels. Turney [1] proposed an unsupervised approach of sentiment analysis. The polarity of the textual content was determined by aggregating the polarities of phrases containing adjectives and adverbs. Pointwise mutual information (PMI) based method was employed to identify the polarity of a phrase. Fernández et al. [2] presented a method of computing semantic orientation of unstructured text based on dependency parsing technique. The proposed method leveraged the use of sentiment lexicons which were created using semi-automatic polarity expansion algorithm. Supervised machine learning approaches of sentiment analysis

978-1-5090-6255-3/17/$31.00

involve training classifiers using linguistic features that are extracted from the text. A labeled dataset of text documents is required to train classifiers. Linguistic features that have been widely used for sentiment analysis includes n-grams (Pang et al. [3]), word representations (Tang [4]), part of speech (POS) tags, punctuations and emoticons (Kiritchenko et al. [5]). Altrabsheh et al. [6] presented a supervised learning approach to predict sentiment from students feedback. The reported models were trained using n-gram features extracted from the feedback text. Naive Bayes, Maximum entropy and Support Vector Machine (SVM) algorithms were used to train models.

B. Lexicon based Method

Lexicon based approach of sentiment analysis makes use of a sentiment lexicon to determine the polarity of a given textual content. A lexicon or dictionary represents a list of words with associated sentiment polarity. The lexicon can be constructed either manually or automatically. Hu and Liu [7] utilized an online lexical resource WordNet to predict the semantic orientation of an opinion word. Taboada et al. [8] proposed another lexicon-based approach that determines the polarity of a word by using the dictionaries constructed.

A number of general purpose and domain specific lexicons have been constructed such as MPQA subjectivity lexicon [1], Harvard General Inquirer [2], Linguistic Inquiry and Word Counts database [3] and many others. One drawback of lexicon based approaches is that the contextual and domain-specific semantic orientation of a word is generally ignored.

Rajput and Haider [9] described the use of lexicon to determine the sentiment polarity of the feedback given by a student. They modified a general-purpose sentiment dictionary to determine the polarity of an opinion in the context of the academic domain. Their results suggested that the use of domain-specific sentiment lexicon achieved better results as compared to any general purpose sentiment lexicon.

C. Hybrid Approach

Hybrid Approaches use sentiment lexicon in machine learning methods. Zhang et al. [10] proposed a hybrid approach for sentiment analysis of Twitter data. An opinion lexicon was used to label training dataset with sentiment polarities. The labeled dataset was then used to train a binary classifier to predict sentiment polarity on the evaluation dataset. Appel et al. [11] performed sentiment analysis at the sentence level using a hybrid approach. Their approach was based on a sentiment lexicon extended using SentiWordNet and fuzzy sets to determine sentiment polarity of a sentence.

This paper also presents a hybrid approach that combines the use of sentiment dictionary and machine learning methods to determine the semantic orientation of a textual feed provided by students.

[1]http://mpqa.cs.pitt.edu/

[2]http://www.wjh.harvard.edu/~inquirer/

[3]http://liwc.wpengine.com/

TABLE I
SAMPLE COMMENTS FROM DATASET

S.No	Student Feedback	Sentiment Labels
1.	timings are very odd such courses should not be offered at such late timings, as for programming PERSON NEED FRESH MIND. Till the time of our class we r all dead tired and sleepy.	negative
2.	Inefficient, boring, confusing.	negative
3.	She is a very hard working instructor, actually helps us a lot however the course is way too irrelevant for ACF students	positive
4.	An excellent course, taught very well. The lectures and lab are well organized, and the instructor explains things very well. But perhaps the course might be too easy.	positive
5.	Give more programming assignments and enhance the level of the course to include critical thinking and solutions as it is required in CS research.	neutral

III. METHODOLOGY

The presented methodology classifies sentiment polarity as positive, negative and neutral. The processes workflow is shown in Fig. 1 and is further described in the following subsections.

A. Dataset Description

The dataset used in this paper comprises of 1230 comments extracted from our institutes educational portal. The dataset was manually labeled with sentiment polarity labels { *positive, negative, neutral*}. Table I shows few examples of student comments.

B. Preprocessing

Student feedback data represents an unstructured text. To extract useful information from the unstructured text, several preprocessing steps are applied to remove spelling errors, grammatical mistakes, URLs, etc. from the text. During preprocessing stage, the following tasks were performed using Python's NLTK [12] library was used to perform preprocessing.

1) Punctuations: Punctuations, numbers and other special characters were removed as these characters do not carry useful information related to sentiment analysis.
2) Tokenization: Tokenization is the process of splitting text stream into a list of words.
3) Case Conversion: After tokenization, words were transformed into lower case.
4) Stop words: : In natural language processing, words that are frequently used such as helping verbs, prepositions, articles are termed as stop-words. Stop-words generally do not provide any useful information and therefore were removed from the feedback text.

C. Data Partition

For training and evaluation purposes, the manually labeled dataset of students feedback, as shown in Table I, was randomly split into train set and test set. 70% of the dataset

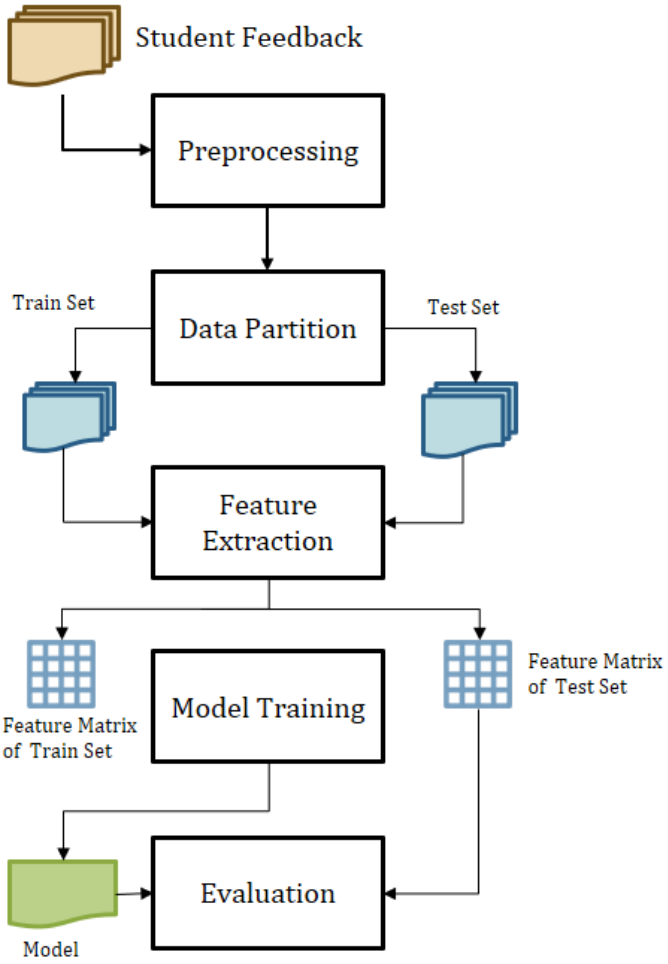

Fig. 1. Methodology of Student Feedback Sentiment Analysis

was used for training and the remaining dataset was used for the evaluation purpose. Table II presents the distribution of sentiment labels in the training and the testing datasets. As shown in the table, the distribution of labels was highly skewed towards positive sentiments.

TABLE II
DISTRIBUTION OF SENTIMENT LABEL IN DATASETS

Sentiment Label	Trainset (sentences)	Testset (sentences)
positive	713	306
negative	126	55
neutral	22	8

D. Feature Extraction

After data splitting, feature extraction was applied on both training and testing datasets. During the feature extraction stage, the preprocessed text was converted into a numerical feature vector using *Ngram* and *TF-IDF*. The following features were extracted during feature extraction phase.

1) Term Frequency-Inverse Document Frequency (TF-IDF): TF-IDF metric determines the importance of a word to a document in a given corpus. It assigns a higher weight to the words that occur frequently in a set of documents labeled with a particular sentiment polarity but least occurring in a corpus. In general, there are different libraries available for computing TF-IDF metric. In this study, TF-IDF vectorizer of scikit-learn library [13] was used. The library transformed a set of documents into a TF-IDF feature matrix. Fig. 2 shows top ten words in positive and negative feedbacks having highest TF-IDF scores.

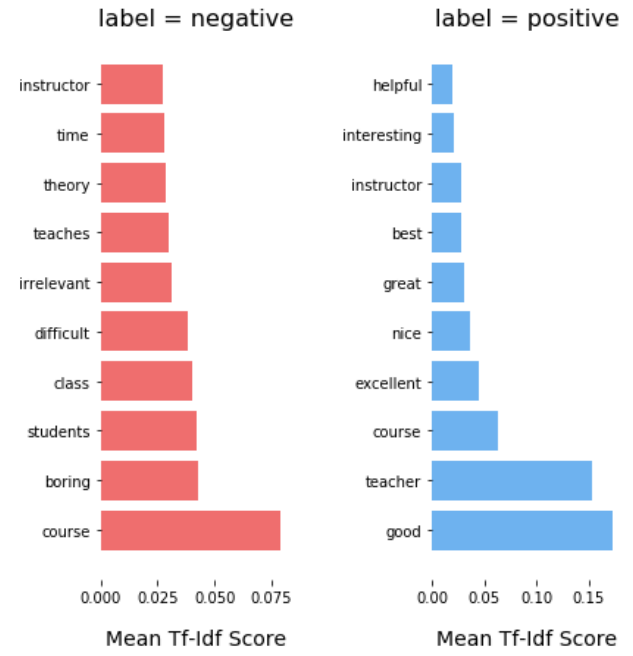

Fig. 2. TOP 10 WORDS IN POSITIVE AND NEGATIVE COMMENTS USING TF-IDF METRIC

2) N-gram Features: N-gram refers to the contiguous sequence of N words in a given text stream. We extracted unigram (1 word) and bigram (2 words) features from the students feedback dataset. A higher values of N was leading to a very high dimensional feature space, therefore, only bigrams were considered in this paper.

3) Lexicon Features: After computing N-gram and TF-IDF based features, lexicon based feature was extracted from the preprocessed text. In this study, we used a sentiment lexicon developed during an earlier research work [9] by our group. The sentiment lexicon was the modified version of MPQA subjectivity lexicon [14] which is a general purpose lexicon. We found words such as fine, miss, lecture, and fun labeled as negative words in the original version of MPQA lexicon. However, when considered in the context of student-teacher domain, these words are used in student feedback to express positive sentiment. Considering the following sentences,

1) I enjoyed her lectures.
2) His classes are full of fun.
3) The teacher is fine.

In the above sentences, the words *lectures, fun and fine* are used as positive opinion words. In the earlier work, for each

TABLE III
SENTIMENT ANALYSIS MODEL EVALUATION RESULTS

Algorithms	**Random Forest**		**SVM**	
Features	**Accuracy**	**F-Measure**	**Accuracy**	**F-Measure**
Unigrams	0.894	0.878	0.890	0.880
Unigram + Lexicon Scores	0.921	0.911	0.910	0.910
Bigrams	0.872	0.847	0.880	0.870
Bigrams + Lexicon Scores	0.907	0.890	0.910	0.910
Unigrams + Bigrams	0.886	0.865	0.886	0.874
Unigrams + Bigrams + Lexicon Scores	0.900	0.892	0.910	0.900
TF-IDF	0.900	0.887	0.890	0.880
TF-IDF + Lexicon Scores	**0.934**	**0.926**	0.929	0.920

given sentence, sentiment score was computed by subtracting the count of negative words from the count of positive words. The semantic orientation of a word was determined using a sentiment dictionary. Sentiment scores were then used as a feature vector while training sentiment analysis model. This paper also handles negation while computing sentiment score of a given textual feedback. A list of negation words was defined in the system which includes {*not, no, never, donot, dont, didnt, didnot, wont, havent, havenot, hasnot, hasnt, nor, doesnot*}. The polarity of the opinion word found from the sentiment dictionary was then modified if the opinion word was preceded by a negation word.

Following examples illustrates the process of computing sentiment score of a textual feedback with the help of a sentiment dictionary built for an academic domain.

1) **Sentence:** Difficult course but he made our concepts clear.
   After preprocessing: difficult course made concepts clear
   Negative words: {*difficult*}
   Positive words: {*clear*}
   Sentiment Score: 1-1 = 0

2) **Sentence:** No complaints as such, nice teacher.
   After preprocessing: no complaints nice teacher
   Negative words: {*complaints*}
   Positive words: {}
   Negation word: {*no*}
   After negation handling the polarity of the word *complaints* was reversed.
   Sentiment Score: 1-0 = 1

3) **Sentence:** The timings for this course is not good.
   After preprocessing: timings course not good
   Negative words: {}
   Positive words: {*good*}
   Negation word: {*not*}
   After negation handling, the polarity of the word *good* was reversed.
   Sentiment Score: 0-1 = -1

E. Model Training

After the extraction of features from the train and test dataset, learning algorithms were applied for training model. The hybrid model for sentiment analysis was trained using unigrams, bigrams, TF-IDF and lexicon-based features. A brief description of the learning algorithms is given below:

1) Random Forest: Random Forest Algorithm was proposed by [15]. In this study, scikit-learn [13] implementation of Random Forest algorithm was used. The hyper parameters were tuned using three fold cross validation.

2) Support Vector Machines (SVM): The scikit-learn implementation of SVM [13] with linear kernel was used to train model.

IV. EXPERIMENTAL RESULTS

The evaluation of learned models was performed using the test dataset. The following evaluation metrics were used:

1) Accuracy
   Accuracy is defined as the ratio between number of correct predictions made by the model and the number of rows in the dataset.
   $$Accuracy = \frac{\#correct\ predictions}{\#data\ points}$$
2) F-Measure
   F-Measure is another commonly used metric in the multi-class classification task. It is defined as the geometric mean of *precision* and *recall*. *Precision* is defined as the ratio between the correct predictions and the total predictions made by the system. *Recall* is the ratio between the correct predictions made by the model and the total number of true sentiment labels. F-measure is an effective metric of measuring the performance of a model where the data is highly imbalance.
   $$F - measure = \frac{2 * Precision * Recall}{Precision + Recall}$$
   In this study, weighted f-measure was computed to account for sentiment labels imbalance distribution.

Table III presents the results of sentiment analysis model evaluation on the test dataset. As shown in the table, the hybrid approach which includes TF-IDF and lexicon based features for model training provides better results as compared with N-gram and TF-IDF features used without lexicon features. It

was also observed that the performance of classifier decreased with bigrams as compared to unigrams. This was due to the increase of feature matrix dimension. The feature matrix dimension increased to 4345 feature columns with bi-grams. In high dimensional feature space, an enormous amount of training sample is required which was not available for this study.

Table IV shows the confusion matrix of the best model that employed TF-IDF and domain-specific lexicon features for training. The model classified most of the positive and negative student feedbacks correctly. For the *neutral* class, the predictions were not so good. This is due to the insufficient number of neutral samples. We believe that the performance can be further improved by increasing the number of neutral comments in the training set.

TABLE IV
CONFUSION MATRIX

	negative	**neutral**	**positive**
negative	**40**	0	15
neutral	2	**1**	5
positive	1	1	**304**

TABLE V
PREDICTED AND TRUE SENTIMENT LABELS OF SELECTED STUDENT FEEDBACKS

S. No	**Student Feedback**	**Actual Sentiment Label**	**Predicted Sentiment Label**
1.	Useless course.	negative	negative
2.	Course is not too much interesting.	negative	positive
3.	The teacher was very helpful and cleared my previous concepts	positive	positive
4.	She is a good teacher. Sometimes she is unable to relate things practically.	neutral	positive

Table V shows few samples of the test dataset with actual sentiment labels and the predicted sentiment labels. For feedback 1 and 3, our model made correct predictions. However, in feedback 2 it is observed that the model did not handle negation properly as the word *not* occurred within a window of size 3 in contrast to a window of size 2 and thus made an incorrect prediction. For the neutral feedbacks, our model classified them as positive and negative. This was due to the insufficient number of training samples for the neutral class.

The term frequency-inverse document frequency (TF-IDF) and unigram features are found to be effective in sentiment analysis of textual feedback provided by students. Moving to higher order n-grams did not improve the performance, rather the accuracy was decreased.

The best so far model achieved during training and evaluation phase was trained used lexicon score and TF-IDF features. The use of domain-specific sentiment lexicon also highlights the need to construct an extensive sentiment lexicon for the academic domain. To the best of our knowledge, there is no sentiment lexicon publicly available for the academic domain.

TABLE VI
COMPARATIVE ANALYSIS BETWEEN PROPOSED APPROACH AND OTHER SENTIMENT ANALYSIS APPROACHES

	Accuracy	**F-Measure**
Aylien Text API	0.67	0.76
Alchemy Language API	0.49	0.57
Text Analytics API	0.79	0.74
Lexicon-based Approach	0.91	0.90
Proposed Hybrid Approach (TF-IDF with Domain Specific Lexicon)	**0.93**	**0.92**

V. COMPARATIVE ANALYSIS

In addition to the presented approach of sentiment analysis on student feedbacks, we also compared our methodology with other sentiment analysis tools available on the web. This section presents a comparison of the hybrid approach presented in this paper with lexicon based approach and other application programming interface (API) based services used for sentiment analysis. Three APIs namely Text Analytics API [4] by Microsoft, Alchemy Language API [5] and Aylien Text API [6] were evaluated on the student feedback dataset.

A. Text Analytics API

Text Analytics API returns a numeric sentiment score between 0-1, where sentiment score close to 1 represents that the provided textual content is highly positive and score close to 0 represents a textual content is highly negative. The API used classification techniques using n-gram and word embeddings features.

A trial version of API was used for evaluation purpose. The student comments present in the dataset were provided to the API. The API returned a numeric score instead of discrete sentiment labels. Therefore, to compare it with discrete sentiment labels manually labeled in the dataset, a decision tree classification algorithm was used to train a model to predict discrete sentiment labels from the numeric sentiment score on train dataset and the evaluation was performed on the test dataset.

B. Alchemy Language API

Alchemy Language API returns discrete sentiment labels defined as {*positive, negative, neutral*}. We used a free subscription plan of the API for evaluation purpose. Student comments in the test dataset were passed to the sentiment analysis method of API. The resultant set of sentiment labels were then compared with the manually annotated sentiment labels in the test dataset to measure accuracy.

C. Aylien Text API

Aylien Text API is another text analysis API available for sentiment analysis. The sentiment analysis method of the API requires text input to analyze and returns discrete sentiment

[4]https://docs.microsoft.com/en-us/azure/machine-learning/machine-learning-apps-text-analytics

[5]https://www.ibm.com/watson/developercloud/alchemy-language.html

[6]http://aylien.com/text-api

polarity labels dened as {*positive, negative, neutral*}. We used free subscription plan of API for evaluation purpose.

Student feedback content present in test dataset was passed to the API and the returned sentiment labels were than compared against manually annotated sentiment polarity labels.

D. Lexicon Based Approach

The lexicon-based approach of sentiment analysis was presented in our earlier work [9]. With this approach, sentiment scores of each comment in the dataset were computed using sentiment lexicon. The discrete sentiment labels were then assigned to the sentiment score. The feedback was considered as *positive* if the sentiment score was greater than zero. Sentiment label *neutral* was assigned to the comments where sentiment score was zero and the label *negative* was assigned to the comments with negative sentiment scores.

E. Proposed Hybrid Approach

The best performing model trained on TF-IDF and lexicon score features was chosen for comparison with the approaches discussed above.

The comparison results presented in Table VI shows that the results produced by our proposed hybrid approach outperform the predictions made by Aylien Text API, Alchemy Language API and Text Analytics API. The inclusion of domain specific features made our model more suitable for the academic domain as compared to general-domain models used by these APIs. In comparison to the lexicon-based approach, the presented approach achieves 2% improvement in the results.

VI. Conclusion

The paper described a hybrid approach for performing sentiment analysis on student feedbacks. The presented approach employed machine learning methods along with sentiment lexicons. The paper also investigated other APIs available for sentiment analysis and compared the results with the presented hybrid approach. It was found that the best performing model was achieved using TF-IDF and domain-specific sentiment lexicon. In contrast to lexicon-based approach, the proposed approach of combining the use of sentiment lexicon with machine learning techniques was capable of predicting the sentiment of the textual content even if the opinion words do not exist in the lexicon. However, the presented approach is limited to the computation of overall sentiment of the student feedback. It is often observed that the students discuss multiple aspects of the instructor in their feedback which includes teaching style, lecture organization, knowledge, and punctuality. The future work would include fine-grained analysis of student feedbacks at the aspect level.

References

[1] P. D. Turney, "Thumbs up or thumbs down?: semantic orientation applied to unsupervised classification of reviews," in *Proceedings of the 40th annual meeting on association for computational linguistics*. Association for Computational Linguistics, 2002, pp. 417–424.

[2] M. Fernández-Gavilanes, T. Álvarez-López, J. Juncal-Martínez, E. Costa-Montenegro, and F. J. González-Castaño, "Unsupervised method for sentiment analysis in online texts," *Expert Systems with Applications*, vol. 58, pp. 57–75, 2016.

[3] B. Pang, L. Lee, and S. Vaithyanathan, "Thumbs up?: Sentiment classification using machine learning techniques," in *Proceedings of the ACL-02 Conference on Empirical Methods in Natural Language Processing - Volume 10*, ser. EMNLP '02. Stroudsburg, PA, USA: Association for Computational Linguistics, 2002, pp. 79–86. [Online]. Available: https://doi.org/10.3115/1118693.1118704

[4] D. Tang, "Sentiment-specific representation learning for document-level sentiment analysis," in *Proceedings of the Eighth ACM International Conference on Web Search and Data Mining*, ser. WSDM '15. New York, NY, USA: ACM, 2015, pp. 447–452. [Online]. Available: http://doi.acm.org/10.1145/2684822.2697035

[5] S. Kiritchenko, X. Zhu, and S. M. Mohammad, "Sentiment analysis of short informal texts," *Journal of Artificial Intelligence Research*, vol. 50, pp. 723–762, 2014.

[6] N. Altrabsheh, M. Cocea, and S. Fallahkhair, "Learning sentiment from students feedback for real-time interventions in classrooms," in *Adaptive and Intelligent Systems*. Springer, 2014, pp. 40–49.

[7] M. Hu and B. Liu, "Mining and summarizing customer reviews," in *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2004, pp. 168–177.

[8] M. Taboada, J. Brooke, M. Tofiloski, K. Voll, and M. Stede, "Lexicon-based methods for sentiment analysis," *Computational linguistics*, vol. 37, no. 2, pp. 267–307, 2011.

[9] Q. Rajput, S. Haider, and S. Ghani, "Lexicon-based sentiment analysis of teachers evaluation," *Applied Computational Intelligence and Soft Computing*, vol. 2016, p. 1, 2016.

[10] L. Zhang, R. Ghosh, M. Dekhil, M. Hsu, and B. Liu, "Combining lexicon-based and learning-based methods for twitter sentiment analysis," *Technical Report*, 2011.

[11] O. Appel, F. Chiclana, J. Carter, and H. Fujita, "A hybrid approach to the sentiment analysis problem at the sentence level," *Knowledge-Based Systems*, vol. 108, pp. 110–124, 2016.

[12] S. Bird, E. Klein, and E. Loper, *Natural language processing with Python: analyzing text with the natural language toolkit*. " O'Reilly Media, Inc.", 2009.

[13] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[14] J. Wiebe and E. Riloff, "Creating subjective and objective sentence classifiers from unannotated texts," in *International Conference on Intelligent Text Processing and Computational Linguistics*. Springer, 2005, pp. 486–497.

[15] L. Breiman, "Random forests," *Machine learning*, vol. 45, no. 1, pp. 5–32, 2001.