Practical Sentiment Analysis for Education: The Power of Student Crowdsourcing

Robert Kasumba, Marion Neumann

Washington University in St. Louis
St. Louis, Missouri, USA
rkasumba@wustl.edu, m.neumann@wustl.edu

Abstract

Sentiment analysis provides a promising tool to automatically assess the emotions voiced in written student feedback such as periodically collected unit-of-study reflections. The commonly used dictionary-based approaches are limited to major languages and fail to capture contextual differences. Pretrained large language models have been shown to be biased and online versions raise privacy concerns. Hence, we resort to traditional supervised machine learning (ML) approaches which are designed to overcome these issues by learning from domain-specific labeled data. However, these labels are hard to come by – in our case manually annotating student feedback is prone to bias and time-consuming, especially in high-enrollment courses. In this work, we investigate the use of student crowdsourced labels for supervised sentiment analysis for education. Specifically, we compare crowdsourced and student self-reported labels with human expert annotations and use them in various ML approaches to evaluate the performance on predicting emotions of written student feedback collected from large computer science classes. We find that the random forest model trained with student-crowdsourced labels tremendously improves the identification of reflections with negative sentiment. In addition to our quantitative study, we describe our crowdsourcing experiment which was intentionally designed to be an educational activity in an introduction to data science course.

Introduction

Enrollment numbers in computer science courses have been on the rise since 2006 in most US universities (Loyalka et al. 2019; NASEM 2018). This has come with a similarly widening feedback gap between students and course instructors. Periodically collecting student feedback in the form of free-form text or Likert-style survey questions is one approach to bridge this feedback gap. Assessing this feedback in a timely manner allows instructors to learn about course materials students struggle with, gain awareness of teams that have issues, or even identify students that fall behind (Ahadi et al. 2015; Presler-Marshall, Heckman, and Stolee 2022; Gitinabard et al. 2022; Neumann and Linzmayer 2021). Likert-type survey questions are easy to analyze but need to be carefully tailored to specific contexts and the responses tend to be unreliable (Holzbach 1978; Leising et al. 2016; Murphy 1993). What is more, they offer limited detail as to what causes issues. Unit-of-study reflections (USRs) in the form of free-form text are widely recognized as being beneficial for students promoting critical thinking, self-regulated learning, and problem-solving skills (Tarricone 2011; Dewey 1933). In addition, they are easy to collect using general prompts that trigger students to report their experiences or reflections for certain course units or activities. However, the large volume of text is cumbersome to manually read and analyze.

Supervised machine learning (ML) methods have been employed for sentiment analysis in general and specifically to analyze student feedback by course instructors. To use these methods, one needs to have ground truth labels, which are both subjective and time-consuming to collect through manual human annotation. Collecting labels from the student authors directly is straightforward but those labels tend to be less reliable. Thus, we investigate the use of *student crowdsourced labels* to predict the sentiment in student textual feedback. We collected assignment reflections and self-reported labels in several large computing courses over multiple years. In addition, we ran a crowdsourcing experiment in two offerings of a data science course to collect multiple student-generated labels for the same feedback texts to aggregate them into high-quality training labels. We demonstrate that using these cheaper-to-get crowdsourced labels achieves comparable performance to using the expensive-to-get dedicated human expert labels in predicting the sentiment of student feedback. Next to providing training labels, this experiment also serves as a hands-on educational activity introducing sentiment analysis, crowdsourcing, and data collection challenges to computer science or data science students.

Use Cases of Sentiment Analysis in Education

Studying the use of crowdsourced labels gained from a student learning experience to train the ML models for supervised sentiment analysis directly benefits many learning analytics use cases in (computing) education. Our approach has the potential to directly improve the following existing practices and processes.

Improving Teaching Student feedback is analyzed both in real-time during the semester or as end-of-semester eval-

uations with the goal of improving teaching. Early identification of areas of concern for students in specific units or assignments can help instructors prepare for review sessions or improve future course offerings. Recent approaches include smartphone applications that generate dashboards with key insights that can guide instructors (Bijlsma et al. 2019). Existing SA approaches use various sources of text such as Twitter data, MOOC forum posts, and teaching evaluations (Dalipi, Zdravkova, and Ahlgren 2021; Li et al. 2022; Adinolfi et al. 2016; Nasim, Rajput, and Haider 2017).

Identifying Students at Risk Automatically detected negative USRs are critical for course instructors who seek to help students who fall behind in high-enrollment courses. Once identified, these students at risk can be offered personalized support in the form of interventions to improve their course experience (Akram et al. 2022).

Improving Team Work Manually analyzed weekly surveys to monitor software engineering students to identify and help struggling teams have proven useful (Presler-Marshall, Heckman, and Stolee 2022).

A dashboard aggregating interaction logs from learning management and discussion systems helps to understand how team members contributed to a group project (Gitinabard et al. 2019).

Understanding Student Populations In the context of large computer science classes, dictionary-based SA revealed that students' grades and their emotional experiences were not correlated thus emphasizing the need to monitor student emotions in addition to assignment or exam performance when managing high-enrollment courses (Neumann and Linzmayer 2021).

Research Questions and Experience Report

To assess the efficacy of student-crowdsourced data in training machine learning (ML) models to identify emotions in student feedback, particularly unit-of-study reflections (USRs), we formulated two guiding research questions.

[RQ1] Are crowdsourced labels a suitable measure of the sentiment polarity of USRs?

[RQ2] Do ML approaches trained on crowdsourced labels perform well at predicting emotions of USRs?

[Experience Report] We further report on the logistical and practical implications of running a crowdsourcing experiment in an introductory data science course.

Background and Related Work

Sentiment analysis (SA) for Education SA refers to the task of determining the sentiment polarity of a piece of text.

Both supervised and unsupervised ML approaches are used to perform SA in education. Unsupervised approaches are popular since they are straightforward to use without requiring training data. Existing approaches predict emotions in unit-of-study evaluations or learning diaries using latent semantic analysis or non-negative matrix factorization (Kim and Calvo 2010; Munezero et al. 2013) or maintain a static lexicon mapping the polarity generally associated with a given word or set of words to a sentiment score. To predict the sentiment in a given sentence the latter approaches aggregate the sentiments of the individual words or phrases. These so-called lexicon- or dictionary-based SA approaches are popular since they are easy to implement and do not require labeled training data (Alencar and Netto 2020). One commonly used example dictionary approach is VADER (Valence Aware Dictionary for Sentiment Reasoning) (Hutto and Gilbert 2014). Other lexicon-based libraries include TextBlob (Gujjar and Kumar 2021) and Flair (Akbik et al. 2019). Unsupervised SA methods have two major drawbacks. They are limited to major languages where lexicons exist and they are less respectful of cultural diversity in Higher Education institutions (Grimalt-Álvaro and Usart 2023). Second, they fail to capture contextual differences as phrases may have different meanings and thus different sentiment polarities in different contexts. Some words that are known to have a positive sentiment may be neutral in other contexts (Kumar and Garg 2020). For example, the use of the word "problem" in student feedback is not necessarily negative, cf. "*I like problem one on the assignment.*". The static nature of their lexicons limits the application of dictionary-based methods to new and dynamic contexts such as the prediction of sentiments in computer science courses with varying topics and diverse student populations.

Supervised ML approaches are designed to overcome these issues and have been applied to Twitter data and real-time student feedback to evaluate student satisfaction (Candra Permana, Rosmansyah, and Abdullah 2017; Dhanalakshmi, Bino, and Saravanan 2016). Existing automatic analysis approaches of USR data focus on the prediction of categories of reflective writing (Kovanović et al. 2018; Ullmann 2019). A hybrid custom lexicon and ML approach was used to predict sentiments in end-of-semester student feedback (Nasim, Rajput, and Haider 2017). Previous work using labels for the training of supervised ML approaches or evaluation of sentiment polarity predictors has relied on dedicated raters to assign sentiment labels to student reflections (Ullmann 2019; Neumann and Linzmayer 2021).

Crowdsourcing Crowdsourcing refers to human-computation systems where a large number of online users perform tasks that would typically be done by a designated agent or expert (Law and von Ahn 2011). Crowdsourcing has proven useful to cheaply label large SA datasets used in applications outside of education (Heidari and Shamsinejad 2020). In the educational context crowdsourcing has been utilized for the design and use of crowdsourced learning analytics tasks (Ahn et al. 2021), as well as, to interpret learners' reviews of MOOCs (Li et al. 2022), but neither to gather sentiment labels nor to serve as a hands-on learning activity for students themselves.

Crowdsourcing Sentiment Labels

In this section, we describe our study data, report on our experiences with running the student crowdsourcing experiment, and outline our process to create training labels for supervised machine learning.

Semester	Course	Students	USRs	SR	CS	HE
Sp16	CC	10	86	✓		
Fl16	CC	13	114	✓		
Sp17	CC	41	348	✓		
Fl17	CC	45	269	✓		
Sp18	CC	97	740	✓		
Fl18	CC	97	722	✓	✓	✓
Fl19	CC	85	563	✓	✓	
Sp20	INTRO	603	3181	✓		

Table 1: Dataset details: semester, course (Cloud Computing (CC) and Introduction to Computer Science (INTRO)), number of students, number of USRs, and available labels (Self-Reported (SR), crowdsourced (CS), and Human Expert-Annotated (HE)).

Study Setting and Feedback Data

Our dataset consists of 6023 student homework reflections collected in eight large Computer Science courses from Spring 2016 to 2020 at Washington University in St. Louis, a research-focused institution with institutional approval to study human subjects. Table 1 summarizes the data. Students were asked to provide feedback about their experience with the homework assignments in the form of textual USRs of no less than 50 words as well as a star rating (1 to 5) referred to as self-reported (SR) labels with 1 representing the most negative and 5 the most positive sentiment.

This data was collected for all homework assignments during each of the eight courses. For courses offered before Fl18, no minimum length was required and students were asked to provide one of three labels (*positive*, *neutral*, and *negative*) instead of a 5-star rating. For this study, we decided to use those three sentiment categories as classes for the classification problem. This has several benefits, first, we were able to use all our data. Second, reducing the number of classes mitigates some challenges when training the ML algorithms, and what is more, it removes noise in the labels; especially since the nuanced difference between 1 and 2 or 4 and 5 is not essential for the use cases outlined above. Further, we have human expert-annotated (HE) labels for Fl18, which were derived from the median of the star ratings provided by three independent human annotators and crowdsourced (CS) labels for Fl18 and Fl19.

Crowdsourcing Experiment

To establish CS labels, we set up an experiment to collect multiple labels for each USR in the Fl18 and Fl19 datasets. We asked the students in an Introduction to Data Science course to read and label the de-identified homework reflections using a password-protected web-based interface as shown in Figure 1 that randomly displays a USR and records the assigned rating. For each USR, the labelers assigned a 5-point Likert scale rating with 1 representing strong negative and 5 representing strong positive emotions. To ensure quality, the labelers were incentivized with lab quiz credit to provide ratings that were within one rating from the median rating of all other students' ratings. This might have led students to hesitate to provide extreme valued ratings. This was another reason why we converted the 5-star ratings to a 3-point scale $[-1, 0, +1]$, representing negative, neutral, and positive, which mitigates this bias. For the Fl18 dataset, we received 4043 labels with each USR receiving six labels on average and each student labeling 31 USRs on average. For the Fl19 dataset, we received a total of 3037 labels with each USR receiving at least three ratings; on average each student labeled 23 reflections and each reflection received five labels. With this easy-to-set-up experiment, we were able to reliably label 1285 USRs in just two-course sessions.

Crowdsourcing as a Learning Opportunity

We ran our crowdsourcing activity in the last lab of the Fl21 and Sp22 offerings of an intro-level data science course. In addition to providing an efficient way of collecting ground truth labels, our experiment serves as a hands-on learning activity introducing crowdsourcing to students while also illustrating that real-world data collection bears challenges. Here are some students' comments on the activity that indicate that this learning activity is indeed meaningful:

"*I learned what crowdsourcing is and how it works. I had heard of it before but never knew what it actually was, and now I do!*"

"*I learned about crowdsourcing and the difficulty of getting labeled data to train models.*"

"*I got a chance to try crowdsourcing data, so I learned about one of the ways data scientists can collect data in a scalable way.*"

In addition to the crowdsourcing experiment, we introduced the prediction of student emotions from textual feedback as an intuitive example of a real-world SA application that students can immediately relate to themselves. Further, this activity shows that collecting data is not trivial and care needs to be taken when annotating data:

"*I learned about the value of crowdsourcing and putting in the time to give honest ratings.*"

"*I [...] learned that in terms of crowdsourced labeling, the incentive needs to ensure high-quality labeling, rather than completion and that there is a reasonably easy way of judging quality ...* "

All students' comments were collected in Fl21 as (parts of) responses to the lab quiz question: "*What is the one thing you learned from today's lab? Write 1-2 full sentences.*" This question was part of each lab quiz in the course.

Label Aggregation Process

The crowdsourcing experiments on the FL18 and FL19 datasets yielded numerous sentiment ratings assigned by different labelers for each USR. To distill the significance of each labeler's contribution to a particular reflection, we applied the weighted majority procedure. Given the variance in skills and engagement levels among the various labelers, we opted for the *weighted majority vote* (WMV) aggregation method to ascertain the genuine label. WMV inherently acknowledges the diverse accuracy levels of crowd workers

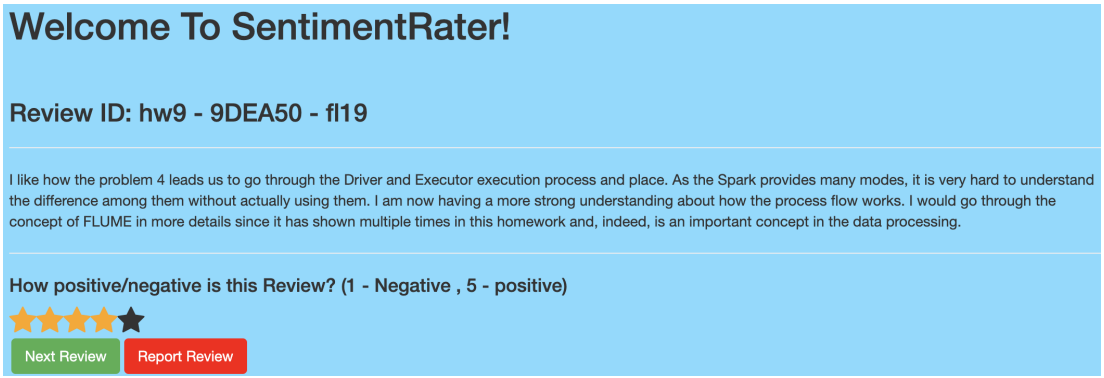

Figure 1: Web-based interface used by students in our crowdsourcing experiment to provide labels.

due to differing experiences. Consequently, distinct weights are attributed to each labeler's input, culminating in the determination of the true label. To weight the contributions from each labeler, we utilized the expectation-maximization (EM) approach for maximum likelihood estimation to ascertain worker accuracies (Dawid and Skene 1979) which we used to weight the contribution of ratings from each labeler.

The EM algorithm starts by assuming an accuracy p_i of 0.8 for all labelers. The M-step computes the labels of each USR using a weighted majority vote with a weight of $2p_i - 1$. The E-step then computes the new p_i of each worker assuming the computed labels are connected. The algorithm runs until there is no further change in the labels and labeler accuracies. The initialization of the labeler accuracy for any $p_i > 0.5$ did not affect the final results.

Resulting Labels We explored two settings for label aggregation. In the first setting, we exclusively utilized the ratings provided by the crowd labelers. Subsequently, we shall refer to the resultant aggregated labels as crowdsourced (CS) labels. In the alternate approach, we expanded the crowdsourced labels by incorporating the self-reported labels by the student authors. This procedure involved treating the students who authored the original reflections as an additional group of labelers. Consequently, each USR received an extra label, which was then aggregated alongside the crowdsourced labels, leading to the formation of CS+SR labels. Notably, both label aggregation settings employed the same algorithm to generate the final labels, denoted as CS and CS+SR labels, respectively.

Answering RQ1

To assess the quality of crowdsourced labels we computed the root mean square difference (RMSD) and weighted Cohen's kappa (κ) between CS, CS+SR, SR, and human expert-annotated (HE) labels as well as randomly generated ones. Table 2 summarizes the scores. Measures involving random labels were averaged over 500 iterations using independently generated random samples. Interestingly, we discovered that SR labels exhibited a smaller RMSD value (1.21) when compared to randomly generated labels. This value was lower than that of labels annotated by human experts (1.26) and those obtained through crowdsourcing

RMSD					
	HE	SR	CS	CS+SR	RAND
HE	0	0.78	0.75	0.72	1.26
SR	–	0	0.87	0.80	1.21
CS	–	–	0	0.61	1.28
CS+SR	–	–	–	0	1.28

Weighted Cohen's kappa					
	HE	SR	CS	CS+SR	RAND
HE	1	0.53	0.61	0.64	0.0
SR	–	1	0.39	0.48	0.0
CS	–	–	1	0.74	0.0
CS+SR	–	–	–	1	0.0

Table 2: Agreement measured by RMSD (the lower the better) and weighted Cohen's kappa (the higher the better) for human-expert annotated (HE), student self-reported (SR), crowdsourced (CS), combined (CS+SR), and randomly generated (RAND) labels in the FL18 dataset.

(1.28). The fact that SR labels are closer to random indicates that they are less reliable. Cohen's kappa gauges the inter-rater reliability among human coders: $\kappa < 0$ indicates no agreement, 0-0.20 suggests slight agreement, 0.21-0.40 signifies fair agreement, 0.41-0.60 denotes moderate agreement, 0.61-0.80 indicates substantial agreement and 0.81-1 represents almost perfect agreement (Landis and Koch 1977). κ between SR and HE labels is 0.53, only indicating a moderate agreement, whereas κ between CS and HE labels is 0.61 indicating substantial agreement. The agreement is even stronger between HE and CS+SR with a score of 0.64.

Additionally, the SR label distribution shown in Figure 2 is extremely left-skewed with a relatively high number of neutral labels, whereas both the distribution of HE, CS, and CS+SR labels is closer to a bi-modal distribution. Neutral labels are considered an easy way out for students to not take sides (Neumann and Linzmayer 2021). Furthermore, the SR labels often disagree with the written text. We hypothesize that this disagreement is caused by various biases that occur when students report on their own experiences as an answer

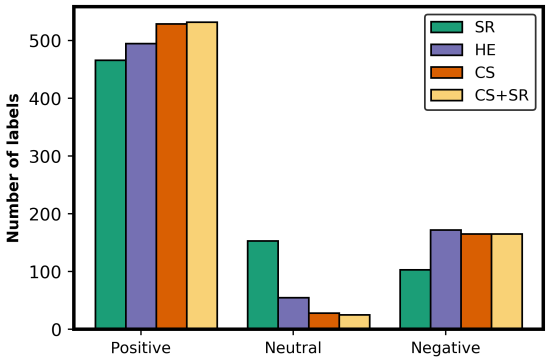

Figure 2: Distribution of self-reported (SR), human expert-annotated (HE), crowdsourced (CS), and combined (CS+SR) labels in the FL18 dataset.

to a 5-point Likert-scale question. These issues include honesty, the ability to assess themselves accurately, the question prompt may have different meanings for different students, as well as, response and sampling bias. The fact that computer science students often come from diverse cultural backgrounds aggravates these issues, especially in the cases where we see positive SR labels for reflections that express negative experiences. Due to this noise, student SR labels are not suitable for training supervised sentiment prediction approaches. Crowdsourcing offers a promising framework to obtain more reliable labels. Specifically, when designing interventions triggered by (negative) sentiment predictions one might argue that the self-reported ratings are the best indicator of whether a student needs attention or not since the information comes directly from the students themselves. However, this would mean that we miss reaching out to all those students who provide positive SR labels, but voice negative experiences in their reflections. This is exactly the student population that we would like to identify. So, even when evaluating models for sentiment predictions with the goal of helping students who struggle, we argue that the ground truth should follow the actual written text rather than the self-reported Likert-scale answers.

In summary, we answer RQ1 affirmatively and conclude that crowdsourced labels are a suitable measure to quantify emotions voiced in USRs.

Predicting Emotions

In this section, we evaluate how ML models trained on the different kinds of labels perform in predicting sentiments from the written text.

SA Problem and ML Models

The SA classification problem was set up with three classes, $y_i \in \{-1, 0, +1\}$. We compare the performance of support vector machine (SVM) and random forest (RF). These models have shown the best performance in related sentiment prediction tasks (Altrabsheh, Cocea, and Fallahkhair 2014; Dhanalakshmi, Bino, and Saravanan 2016) as well as for general ML classification problems. An initial set of experiments showed weak performance for other models such as the regularized linear model and multinomial naïve Bayes. All experiments were implemented using the SciKit Learn package (version 0.22.1) in Python 3.7 and executed on a 16-core CPU. To deal with our imbalanced input data, we used balanced versions of the RF and SVM training methods. For SVM this means that the C-value of each class is multiplied by an automatically adjusted weight that is inversely proportional to the class frequency of the input class labels used for training. For RF we experimented with two balancing methods, one that uses the same weights as in SVM to scale the impurity criteria and one where the weights are computed based on the bootstrap sample for every tree grown. The choice of balancing method is a hyperparameter (cf. *balanced* and *balanced_subsample* in Table 3) and learned during hyperparameter tuning.

Features

The input text documents r_i are the assignment reflections (USRs). To generate features $\mathbf{x}_i$ the lower-case USRs were preprocessed to remove stop words and punctuation symbols. We then generated various numeric features $\mathbf{x}_i$ using term frequency-based scores. We also experimented with document embedding features using the pre-trained BERT model with 768 dimensions (Alaparthi and Mishra 2021). However, with the exception of RF when optimizing for weighted recall, the ML models did not perform well with these features compared to TF-IDF. We use unigrams, bigrams, and trigrams as TF-IDF features where the number of features was learned during hyperparameter tuning. We enhanced the TF-IDF features with a 1-dimensional VADER-based feature following previously introduced successful hybrid approaches (Nasim, Rajput, and Haider 2017). The dictionary-based method VADER (Valence Aware Dictionary and sEntiment Reasoner) was designed for micro-blog text (Hutto and Gilbert 2014). VADER takes as input a USR r_i and computes a compound score $c_i \in \mathbb{R}$ based on the normalized sum of the sentiment arguments in its dictionary. Only using the VADER score as a single feature performed poorly. Adding it to the hybrid approach improved performance compared to using TF-IDF alone.

Training and Evaluation

For training and evaluation, we consider three datasets FL18, FL18+FL19, and REST. The FL18 data, where we have ground truth (HE labels) in addition to SR and CS labels, was split via stratified 5-fold cross-validation with 80% of this dataset added to the respective other data used in the various training settings. The 20% was held out for testing as shown in Figure 3. This setup allows us to use SR, CS, or CS+SR labels in the FL18+FL19 portion of the training set to compare the effect of using different labels on the model performance. To study the effect of adding more training data (with noisy labels) we use the REST portion of the data which is equipped with SR labels. Only HE labels were used for performance evaluation.

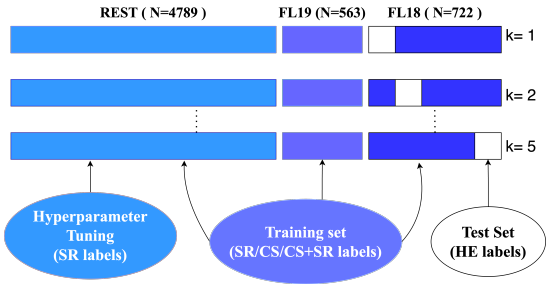

Figure 3: Cross-validation scheme used for model evaluation. Note that a separate 5-fold cross-validation on the hyperparameter tuning set is used to learn the model hyperparameters and the number of TF-IDF features.

Hyperparameter Tuning We used grid search to choose hyperparameters of all ML models using 5-fold cross-validation on the hyperparameter tuning set (REST) optimizing for weighted accuracy. Since the dataset is imbalanced, weighted accuracy ensures that the models perform well across all sentiment categories c. We used the grid search package in SciKit Learn to choose the hyperparameters. Table 3 shows the used parameter search spaces.

Performance Measures Four metrics were used for evaluation: *weighted recall*, *weighted precision*, *weighted F-score*, and *negative class recall*. Given the dataset's imbalance, these weighted metrics evaluate model performance across all sentiment categories. Weighted scores are derived by combining standard metrics for each class, weighted by their support, i.e., the number of true instances per class. Maximizing weighted recall reduces "false negatives", and maximizing weighted precision decreases "false positives" in each class. The weighted F-score is the harmonic mean of recall and precision, calculated for each class and weighted by class support. Additionally, negative class recall, $\text{Recall}_{-1} = \frac{true\ negatives}{all\ negatives}$, is of special interest for our work, as it helps minimize instances where struggling students are incorrectly classified as positive or neutral, ensuring they receive appropriate support.

Results

We designed a series of experiments to assess whether ML approaches trained on crowdsourced labels perform well at predicting emotions of USRs compared to using noisy student self-reported labels (RQ2). We use the human expert-annotated (HE) labels in the FL18 dataset as ground truth for evaluation in all experiments.

In the first set of experiments, we solely used the FL18 dataset trained on HE, SR, CS, and CS+SR labels respectively. Table 4 summarizes the results. As expected both models (RF and SVM) performed best across all measures when using HE labels for training. Comparing the performance of SR labels versus crowdsourced (CS or CS+SR) labels reveals that training on student crowdsourced labels yields higher performance scores across all measures and both ML models with only one exception (weighted recall and SVM). For RF the CS labels result in higher scores than

	Hyperparameter	Values
RF	max_depth	[16, 32, 64, 128, 256]
	n_estimators	[200, 300, 400, 450, 500]
	class_weight	[balanced , balanced subsample]
	criterion	[entropy, gini]
	min_samples_leaf	[7, 9]
	min_samples_split	[7, 9]
	num_features_tfidf	[1000, 2000, ..., 10000]
SVM	kernel	[linear, poly, rbf, sigmoid]
	gamma	[scale, auto]
	decision_function_shape	[ovo, ovr]
	class_weight	[balanced]
	num_features_tfidf	[100, 200, ..., 1000]
TF-IDF	n-grams	[unigrams; uni- & bigrams; uni-, bi-, & trigrams]

Table 3: Hyperparameter search space for RF and SVM as well as TF-IDF features.

SR and CS+SR across all measures. For SVM CS+SR labels yield higher scores than SR and CS. From this experiment, we can conclude that human expert-annotated labels are best and that crowdsourced labels are superior to student self-reported labels. Also, note that the supervised ML methods perform better than the unsupervised VADER (VD) approach for all performance measures.

In the second set of experiments, we explored the following scenario. Starting with a baseline dataset labeled with SR labels, does adding more data with (cheap but noisy) student self-reported labels improve performance (sanity check)? But more importantly, does using better quality labels (CS instead of SR) improve performance? Then, we also looked into the combination of both, adding more data and using better labels where available. Last, we investigated the performance of the combined CS+SR label set in this scenario. Table 5 summarizes the main results. The baseline dataset uses the FL18+Fl19 dataset with SR labels. When adding more data we added the REST dataset with SR labels for training. First, we see that for the baseline weighted precision, F1-score, and Recall_{-1} increase compared to when only training on the FL18 data set (cf. SR row in Table 4). It is worth noting that Recall_{-1} for SVM trained on the baseline is extremely low (15.4% and 19.8%). This is caused by the fact that the negative class in USR data is extremely underrepresented, cf. Figure 2. And when using student self-reported labels this is even more aggravated due to the various previously discussed biases. Once we add the REST dataset which comprises 80% of the total number of USRs, all performance scores increase, notably Recall_{-1} to 63.0% for SVM. Using better labels on the FL18+FL19 dataset namely CS instead of SR labels improves performance, especially weighted recall (from 62.1% to 72.7% for RF and 62.6% to 73.7% for SVM) and negative class recall (from 64.5% to 81.1% for RF and 19.8% to 70.1% for SVM). Combining better labels with more data (CS labels on FL18+FL19 and SR labels on REST) improves weighted precision and Recall_-1 for both ML methods (83.8% for RF and 79.5% for SVM). Finally, we investigated the per-

Labels	Weighted Recall			Weighted Precision			Weighted F1-score			Recall_{-1}		
	VD	RF	SVM	VD	RF	SVM	VD	RF	SVM	VD	RF	SVM
HE	69.9	**72.6**	**70.1**	68.1	**72.6**	**70.2**	68.6	**71.8**	**72.2**	55.7	**74.3**	**61.6**
SR		70.6	68.0		69.6	67.0		61.2	61.4		50.8	15.4
CS		72.0	66.8		71.6	66.8		71.2	69.1		67.4	*51.5*
CS+SR		69.9	67.8		69.6	67.8		71.2	69.8		59.0	52.9

Table 4: Model performance of RF and SVM when trained on the FL18 dataset with different labels. VD represents the baseline performance of the VADER approach. Bold emphasizes the best-performing labels for a given ML model

Training Dataset	Labels	Weighted Recall		Weighted Precision		Weighted F1-score		Recall_{-1}	
		RF	SVM	RF	SVM	RF	SVM	RF	SVM
FL18+FL19	SR	62.1	62.6	71.2	69.8	64.7	62.4	64.5	19.8
	CS	**72.7**	**73.7**	71.5	70.0	70.3	**71.4**	81.1	70.1
	CS+SR	71.1	72.0	71.5	69.1	**70.7**	70.0	76.0	68.0
FL18+FL19 + REST (SR labels)	SR	65.5	65.6	**72.7**	**72.4**	66.2	68.3	78.1	63.0
	CS	63.7	69.9	72.0	71.3	64.6	69.6	83.8	79.5
	CS+SR	65.0	70.0	71.6	71.6	65.2	69.6	**84.1**	**80.2**

Table 5: Model performance of RF and SVM when trained on FL18+FL19 and FL18+FL19+REST with different labels. The REST portion of the dataset uses SR labels. Bold indicates the best setting for the respective measure and ML method.

formance of RF and SVM when using CS+SR labels on the FL18+FL19 dataset in addition to SR labels on REST. When using the combined labels negative class recall increased even more (to 84.1% for RF and 80.2% for SVM) which is good to see since this is the measure we care most about in our educational use cases. Overall, SVM performs better on weighted recall and F1-score with the highest scores being attained for CS labels on the FL18+FL19 only, whereas RF excels when optimizing for weighted precision and negative class recall with the highest score being attained when using more data in combination with CS+SR labels.

Answering RQ2

Although student self-ratings are noisy, they can be useful in training ML models. For better performance, higher quality labels collected via crowdsourcing should be used. To obtain these labels, instructors can set up a data collection activity in any data science or ML class they or one of their colleagues teach. Label aggregation via weighted majority vote is simple to implement and our experiments suggest that annotating training data from previous offerings of the same or similar courses suffices. Supervised SA especially pays off for instructors who wish to use periodically collected written feedback to identify students who have negative emotional experiences, as crowdsourced labels greatly improved negative class recall in our experiments. Training ML methods with crowdsourced labels performs extremely well at predicting emotions of USRs both compared to unsupervised baseline models as well as using student self-reported labels alone. This answers RQ2 affirmatively.

Threats to the Validity

In this work, we chose HE labels as our ground truth but we only have HE labels for the FL18 dataset due to the cost of obtaining them. This could affect the performance scores used in this study. We did not address the issue that the written text itself could be inaccurate, since USRs could also suffer from biases. When writing a reflection a student could be dishonest or inaccurately assess themselves, and the collected data might suffer from response and sampling bias.

Conclusions

Student crowdsourcing is a powerful approach that makes the use of supervised sentiment analysis in education more attainable for course instructors. Crowdsourcing provides an easy way to obtain quality labels and is a worthwhile classroom activity that enhances the learning experience of students in computer science or data science courses. With a negative class recall of 84% supervised ML models trained on crowdsourced labels are extremely promising for what we believe is the most beneficial use case of sentiment analysis in education: the timely identification of students, teams, or course materials at risk.

In future work, we plan to crowdsource ground truth labels on reflections from more courses as well as to expand our crowdsourcing experiment in order to investigate how many labels per USR are necessary for optimal machine learning model performance. Additionally, we plan to expand our work to develop more elaborate ML models including large language models and ones that can simultaneously learn from all the different labels available in the training phase. Last and most importantly, we would like to integrate our approach of predicting student emotions from written feedback into a learning analytics tool and study its usefulness for course instructors to trigger interventions or improve course materials in a classroom study.

Acknowledgements

This work is under the oversight of the IRB at Washington University in St. Louis. We would like to thank Zach Mekus, Kevin Fu, Robin Linzmayer, and the cse217a students in FL21 and SP22 for providing labels, Max Torop for implementing the labeler interface, and Bill Siever and Connor Pepin for their assistance in collecting student consent.

References

Adinolfi, P.; D'Avanzo, E.; Lytras, M.; Novo-Corti, M. I.; and Picatoste, X. 2016. Sentiment Analysis to Evaluate Teaching Performance. *International Journal of Knowledge Society Research*, 7: 86–107.

Ahadi, A.; Lister, R.; Haapala, H.; and Vihavainen, A. 2015. Exploring machine learning methods to automatically identify students in need of assistance. In *Proceedings of the eleventh annual international conference on international computing education research*, 121–130.

Ahn, J.; Nguyen, H.; Campos, F.; and Young, W. 2021. Transforming Everyday Information into Practical Analytics with Crowdsourced Assessment Tasks. In *LAK21: 11th International Learning Analytics and Knowledge Conference*, LAK21, 66–76.

Akbik, A.; Bergmann, T.; Blythe, D.; Rasul, K.; Schweter, S.; and Vollgraf, R. 2019. FLAIR: An easy-to-use framework for state-of-the-art NLP. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics*, 54–59.

Akram, B.; Fisk, S.; Yoder, S.; Hunt, C.; Price, T.; Battestilli, L.; and Barnes, T. 2022. Increasing Students' Persistence in Computer Science through a Lightweight Scalable Intervention. In *To appear in the Proceedings of the 27th Annual Conference on Innovation and Technology in Computer Science Education (ITiCSE)*.

Alaparthi, S.; and Mishra, M. 2021. BERT: A sentiment analysis odyssey. *Journal of Marketing Analytics*, 9(2): 118–126.

Alencar, M.; and Netto, J. F. 2020. Measuring Student Emotions in an Online Learning Environment. In *Proceedings of the 12th International Conference on Agents and Artificial Intelligence - Volume 2: ICAART*, 563–569. INSTICC, SciTePress. ISBN 978-989-758-395-7.

Altrabsheh, N.; Cocea, M.; and Fallahkhair, S. 2014. Learning sentiment from students' feedback for real-time interventions in classrooms. In *International conference on adaptive and intelligent systems*, 40–49. Springer.

Bijlsma, H. J.; Visscher, A. J.; Dobbelaer, M. J.; and Veldkamp, B. P. 2019. Does smartphone-assisted student feedback affect teachers' teaching quality? *Technology, Pedagogy and Education*, 28(2): 217–236.

Candra Permana, F.; Rosmansyah, Y.; and Abdullah, A. 2017. Naive Bayes as opinion classifier to evaluate students satisfaction based on student sentiment in Twitter Social Media. *Journal of Physics: Conference Series*, 893: 012051.

Dalipi, F.; Zdravkova, K.; and Ahlgren, F. 2021. Sentiment Analysis of Students' Feedback in MOOCs: A Systematic Literature Review. *Frontiers in Artificial Intelligence*, 4.

Dawid, A. P.; and Skene, A. M. 1979. Maximum likelihood estimation of observer error-rates using the EM algorithm. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 28(1): 20–28.

Dewey, J. 1933. How we think : a restatement of the relation of reflective thinking to the educative process. *American Journal of Psychology*, 46: 528.

Dhanalakshmi, V.; Bino, D.; and Saravanan, A. M. 2016. Opinion mining from student feedback data using supervised learning algorithms. In *2016 3rd MEC international conference on big data and smart city (ICBDSC)*, 1–5. IEEE.

Gitinabard, N.; Heckman, S.; Barnes, T.; and Lynch, C. 2022. Designing a Dashboard for Student Teamwork Analysis. In *Proceedings of the 53rd ACM Technical Symposium on Computer Science Education*, 446–452.

Gitinabard, N.; Xu, Y.; Heckman, S.; Barnes, T.; and Lynch, C. F. 2019. How Widely Can Prediction Models Be Generalized? Performance Prediction in Blended Courses. *IEEE Transactions on Learning Technologies*, 12(2): 184–197.

Grimalt-Álvaro, C.; and Usart, M. 2023. Sentiment analysis for formative assessment in higher education: a systematic literature review. *Journal of Computing in Higher Education*.

Gujjar, J. P.; and Kumar, H. P. 2021. Sentiment analysis: Textblob for decision making. *Int. J. Sci. Res. Eng. Trends*, 7(2): 1097–1099.

Heidari, M.; and Shamsinejad, P. 2020. Producing An Instagram Dataset For Persian Language Sentiment Analysis Using Crowdsourcing Method. In *2020 6th International Conference on Web Research (ICWR)*, 284–287. IEEE.

Holzbach, R. L. 1978. Rater bias in performance ratings: superior, self-, and peer ratings. *Journal of applied psychology*, 63(5): 579.

Hutto, C.; and Gilbert, m. 2014. Vader: A parsimonious rule-based model for sentiment analysis of social media text. In *Proceedings of the international AAAI conference on web and social media*, volume 8, 216–225.

Kim, S. M.; and Calvo, R. A. 2010. Sentiment Analysis in Student Experiences of Learning. In *Proceedings of the 3rd International Conference on Educational Data Mining*.

Kovanović, V.; Joksimović, S.; Mirriahi, N.; Blaine, E.; Gašević, D.; Siemens, G.; and Dawson, S. 2018. Understand Students' Self-Reflections through Learning Analytics. In *Proceedings of the 8th International Conference on Learning Analytics and Knowledge*, LAK '18, 389–398.

Kumar, A.; and Garg, G. 2020. Systematic literature review on context-based sentiment analysis in social multimedia. *Multimedia tools and Applications*, 79(21): 15349–15380.

Landis, J. R.; and Koch, G. G. 1977. The Measurement of Observer Agreement for Categorical Data. *Biometrics*, 33(1): 159–174.

Law, E.; and von Ahn, L. 2011. *Human Computation*. Morgan & Claypool Publishers, 1st edition. ISBN 1608455165.

Leising, D.; Locke, K. D.; Kurzius, E.; and Zimmermann, J. 2016. Quantifying the association of self-enhancement bias

with self-ratings of personality and life satisfaction. *Assessment*, 23(5): 588–602.

Li, L.; Johnson, J.; Aarhus, W.; and Shah, D. 2022. Key factors in MOOC pedagogy based on NLP sentiment analysis of learner reviews: What makes a hit. *Computers & Education*, 176: 104354.

Loyalka, P.; Liu, O. L.; Li, G.; Chirikov, I.; Kardanova, E.; Gu, L.; Ling, G.; Yu, N.; Guo, F.; Ma, L.; et al. 2019. Computer science skills across China, India, Russia, and the United States. *Proceedings of the National Academy of Sciences*, 116(14): 6732–6736.

Munezero, M.; Montero, C. S.; Mozgovoy, M.; and Sutinen, E. 2013. Exploiting Sentiment Analysis to Track Emotions in Students' Learning Diaries. In *Proceedings of the 13th Koli Calling International Conference on Computing Education Research*, Koli Calling '13, 145–152.

Murphy, K. R. 1993. Modesty bias in self-ratings of performance: A test of the cultural relativity hypothesis. *Personnel Psychology*, 46(2): 357–363.

Nasim, Z.; Rajput, Q.; and Haider, S. 2017. Sentiment analysis of student feedback using machine learning and lexicon based approaches. In *2017 International Conference on Research and Innovation in Information Systems (ICRIIS)*, 1–6.

Neumann, M.; and Linzmayer, R. 2021. Capturing Student Feedback and Emotions in Large Computing Courses: A Sentiment Analysis Approach. In *Proceedings of the 52nd ACM Technical Symposium on Computer Science Education*, SIGCSE '21, 541–547.

NASEM. 2018. *Assessing and responding to the growth of computer science undergraduate enrollments*. National Academies of Sciences, Engineering, and Medicine (NASEM). National Academies Press.

Presler-Marshall, K.; Heckman, S.; and Stolee, K. T. 2022. Identifying Struggling Teams in Software Engineering Courses Through Weekly Surveys. In *Proceedings of the 53rd ACM Technical Symposium on Computer Science Education*, 126–132.

Tarricone, P. 2011. *The taxonomy of metacognition*. Psychology Press.

Ullmann, T. 2019. Automated Analysis of Reflection in Writing: Validating Machine Learning Approaches. *International Journal of Artificial Intelligence in Education*, 29.