





## Article

# Automated Analysis of Open-Ended Students' Feedback Using Sentiment, Emotion, and Cognition Classifications

Melanie Fargues<sup>1,2</sup>, Seifedine Kadry<sup>2,3,4,\*</sup> , Isah A. Lawal<sup>2</sup> , Sahar Yassine<sup>2</sup>  and Hafiz Tayyab Rauf<sup>5</sup> 

<sup>1</sup> Polytech Marseille, School of Engineering, Aix Marseille Université, 13288 Marseille, France

<sup>2</sup> Department of Applied Data Science, Noroff University College, 4631 Kristiansand, Norway

<sup>3</sup> Artificial Intelligence Research Center (AIRC), Ajman University, Ajman 346, United Arab Emirates

<sup>4</sup> Department of Electrical and Computer Engineering, Lebanese American University, Byblos 1102-2801, Lebanon

<sup>5</sup> Centre for Smart Systems, AI and Cybersecurity, Staffordshire University, Stoke-on-Trent ST4 2DE, UK

\* Correspondence: seifedine.kadry@noroff.no

**Abstract:** Students' feedback is pertinent in measuring the quality of the educational process. For example, by applying lexicon-based sentiment analysis to students' open-ended course feedback, we can detect not only their sentiment orientation (positive, negative, or neutral) but also their emotional valences, such as anger, anticipation, disgust, fear, joy, sadness, surprise, or trust. However, most currently used assessment tools cannot effectively measure emotional engagement, such as interest level, enjoyment, support, curiosity, and sense of belonging. Moreover, none of those tools utilize Bloom's taxonomy for students' learning-level assessment. In this work, we develop a user-friendly application based on NLP to help the teachers understand the students' perception of their learning by analyzing their open-ended feedback. This allows us to examine the sentiment and the embedded emotions using a customized dictionary of emotions related to education. The application can also classify the students' emotions according to Bloom's taxonomy. We believe our application will help teachers improve their course delivery.

**Keywords:** students' feedback; open-ended questions; sentiment analysis; emotional analysis; Bloom's taxonomy



**Citation:** Fargues, M.; Kadry, S.; Lawal, I.A.; Yassine, S.; Rauf, H.T. Automated Analysis of Open-Ended Students' Feedback Using Sentiment, Emotion, and Cognition Classifications. *Appl. Sci.* **2023**, *13*, 2061. <https://doi.org/10.3390/app13042061>

Academic Editors: Dickson K.W. Chiu and Kevin K.W. Ho

Received: 5 December 2022

Revised: 1 February 2023

Accepted: 3 February 2023

Published: 5 February 2023



**Copyright:** © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Natural Language Processing (NLP) is a part of artificial intelligence used to make computers understand words as humans do [1]. The use of NLP in educational research, including analyzing student feedback, has recently gained a lot of interest [2–5]. Students' feedback is usually in the form of answers to multiple-choice and closed-ended or open-ended questions. Analyzing this feedback helps the course instructors to understand the students' difficulties during the learning process, thus facilitating the process of course delivery improvements [6]. Unlike multiple-choice and closed-ended questions that need quantitative data analysis tools, NLP is necessary to analyze open-ended feedback because the data collected are qualitative and have a textual form [7]. However, in the past, emphasis has been placed mostly on sentiment analysis of the open-ended students' feedback by classifying them as positive, negative, or neutral using the NLP [8–11]. In recent times, the importance of analyzing the students' emotions in learning has also been discussed in some research to help the teacher understand the overall students' feelings during the learning process [12,13].

This paper presents an application based on NLP for emotion analysis and educational-related feelings classification of students' open-ended course feedback. We detect students' understanding, emotional engagement, and feelings to help teachers identify when to improve their lessons or where the students struggle. Open-ended student feedback is also an exciting source for constructing a Bloom's-taxonomy-based study [14]. Bloom's

taxonomy tool [15–17], is known for defining a hierarchical classification of learning actions and helping teachers build their lessons according to the objectives they want to reach with their students. Thus, our app also comes with a lexicon-based classification using Bloom's digital taxonomy. With this application, we hope to identify needed improvements in the lesson from a more practical and goal-oriented point of view. It is important to note that the proposed application can currently be applied to English text only. However, the techniques can also be adapted to texts written in other languages.

## 2. Related Work

One of the main applications of data analysis is making sense of the unstructured text, such as open-ended responses in course evaluation questionnaires. NLP [18], with the help of other text mining methods [19–21], is making this kind of analysis more accessible. In the following discussion, we will highlight some of the progress made in the literature for open-ended student feedback. Lexicon-based sentiment analysis has been used for open-ended students' feedback to identify their positive or negative attitudes toward the learning materials [8]. The method involves analyzing the intensifier words extracted from students' feedback to determine the polarity of words by mapping them against a database of English sentiment words. Similarly, in 2020, a customized lexicon-based analysis tool was designed to explore and analyze the students' reflective journal writing [22]. Nasim et al. [10] proposed a hybrid approach that combines machine learning and lexicon-based techniques to perform sentiment analysis on students' textual feedback. The study uses a training set of more than 1000 open-ended student feedback and a lexicon-based analysis combined with a Term Frequency-Inverse Document Frequency (TF-IDF) feature to determine the essential words and assign sentiment scores to the words. The computed sentiment scores were used to train a set of classifiers to predict the sentiment scores of previously unseen students' feedback. Yun et al. [23] proposed a fuzzy logic-based method to determine students' satisfaction with their open-ended feedback. The study assigns sentiment scores to opinion words and polarity shifters extracted from the students' responses. The generated scores are then analyzed using fuzzy logic to quantify the students' satisfaction. Ren et al. [11] present an aspect-level sentiment prediction model for course evaluation of students' feedback. The study trained three deep-learning models to predict the sentiment score in open-ended student feedback. Thanks to their large dataset of more than 4000 comments, the study achieved good results with the trained model using a topic dictionary as an input and attention mechanism. Although the proposed work could give the sentiment score of each student's comment individually, no global teaching evaluation was computed by correlating the students' assessments with each other.

Some studies also examine the students' emotions [9,22,24]. For example, Hynninen et al. [9] proposed a study to assess open-ended student feedback by focusing on emotional analysis. First, they performed sentiment analysis to find the positive, negative, and neutral texts. Then, they used the NRC lexicon [25] to calculate the emotion values for each feedback and classify emotions among anger, anticipation, disgust, fear, joy, sadness, surprise, and trust. Finally, they visualized the results using the word cloud library. Okoye et al. [24] proposed a method that uses an educational process, data mining, and a machine learning model to analyze the feedback from students' evaluation of teaching. The study extracts the sentiments and emotions expressed by the students and then performs the covariance and the Kruskal–Wallis Test to understand the influential factors in the students' comments. Chong et al. [22] proposed an NLP model that applies a lexicon-based classification system to categorize the reflection level of reflective writings, followed by a fuzzy-based approach to increase the model's accuracy. As a result, the model identified the following reflection levels: description, feelings/emotions, relation, reasoning, and critical reflection.

For this purpose, a specific reflective keyword dictionary was created using the WordNet library. In addition, the model offers a user-friendly layout to visualize the classification in HTML format in any browser. Some studies also use Bloom's taxonomy classification [17].

For example, Kumara et al. [26] combine Bloom’s taxonomy classification with NLP to categorize and evaluate examination questions. They proposed a system based on six rules, each aiming to represent a question model corresponding to Bloom’s taxonomy’s learning level. Then, the questions are compared to the rules to classify them according to the associated level and determine whether the examination paper is balanced.

The review of all the above studies shows that sentiment analysis is fundamental in analyzing open-ended students’ feedback, and efforts have been made to improve the accuracy of the sentiment analysis. However, while we know the feasibility of conducting emotion analysis and customized lexicon-based classification, no work exploits them in student feedback studies. Furthermore, no study seems to exist in the literature regarding education-related feeling classification. Finally, no study incorporates students’ feedback analysis regarding Bloom’s taxonomy.

### 3. Components of the Application

The components of the proposed App for mining the open-ended students’ feedback (see Figure 1) includes, after the pre-processing of the input data module, five main modules: classical NLP analysis, sentiment analysis, emotional analysis, customized emotional analysis, and classification according to Bloom’s taxonomy. The pre-processing module prepares the input file for analysis by removing files with empty or blank spaces and those with only images. The classical NLP analysis module generates a complete overview of the feedback by computing metrics such as word counts and frequencies accordingly. The sentiment analysis module provides the sentiments observed in the students’ feedback, including positive, negative, and neutral. On the other hand, the customized emotion analysis module allows us to analyze students’ understanding, engagement, and ease of feeling regarding the evaluated course. Finally, Bloom’s taxonomy module classifies the actions related to the learning level of the students as observed in the feedback.

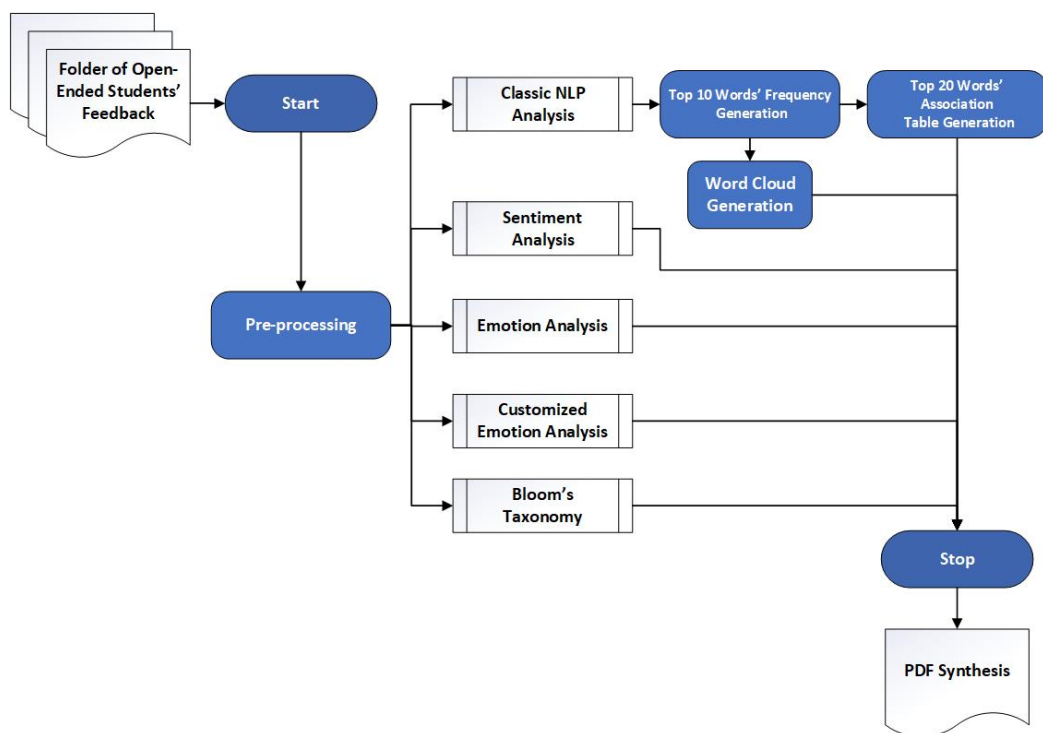


Figure 1. A block diagram illustrating the components of the proposed app.

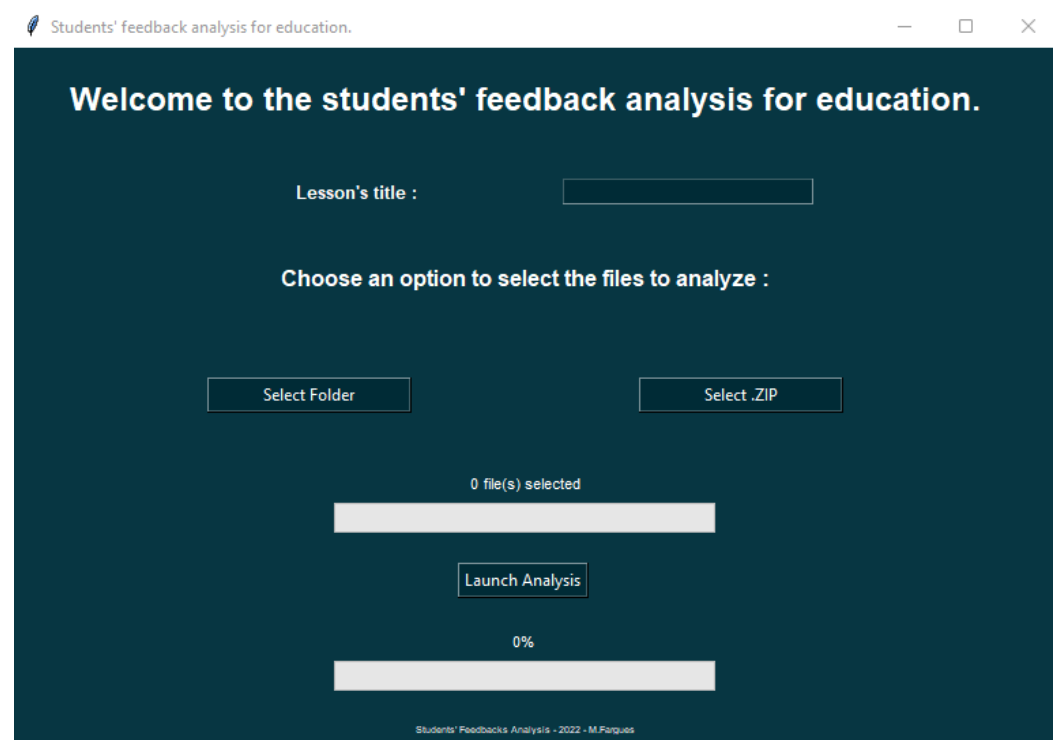
### 4. Implementation and Analysis

This section presents the implementation of the proposed application and how it works. All the application modules were implemented using Python and its built-in NLP libraries [27]. Firstly, we created an activity on the Moodle platform (this is also applicable

to any learning management system) in four lessons, where students must write their feedback and upload it as a PDF file to Moodle at the end of a lecture. Then, the instructor downloads all the submitted feedback in one zipped folder as inputs to the application.

#### 4.1. A User-Friendly Interface

Our application has a user-friendly interface where users can choose the lesson name (see Figure 2). Then, according to the selected button, the feedback files can be added directly from the zip folder. If the folder has already been unzipped, it can be imported like that. Once the desired folder is uploaded, the progress bar begins to evolve to indicate to the user how the importation is going. Once all the files are imported, the user can click the “Launch Analysis” button to generate a synthesized PDF report. A progress bar again indicates to the user how complete the output is; when it is 100%, a message box displays the path of the PDF report and opens it while it closes the interface.



**Figure 2.** Main screen of the user interface.

#### 4.2. The PDF Report

A PDF report is generated for each of the investigated lessons. This PDF report includes all the synthesized analyses conducted on the gathered students’ feedback. The following sections will explain all the implemented analyses and the generated results. The chart at the beginning of each PDF report shows the number of students’ feedback files included in the synthesis. This includes the number of files with images that are ignored for that reason (those can then be read individually by the user) and the number of files that are removed because they were blank files, etc. Table 1 summarises the details of the collected files for each investigated lesson. The items under the number of analyzed files column are the number of files used in our analysis, and we will refer to the lessons by their aliases henceforth.

**Table 1.** The synthesis of resources: Number of students' files in each lesson.

Lesson ID	Alias	Total Number of Students' Files	NO. of Ignored Files (with Images)	NO. of Analysed Files
101496550	Lesson 1	31	15	16
102078874	Lesson 2	30	15	15
103999562	Lesson 3	40	20	20
104161929	Lesson 4	42	21	21

#### 4.3. Pre-Processing of Input Files

Because the input files contain open-ended student feedback, some texts could be polluted by uninformative titles, formatting elements, or unreadable objects (e.g., images). Thus, we proposed several steps for cleaning the feedback comments before analysis. Firstly, we deleted any image in the feedback files, but we kept a feedback counter with the deleted images to allow the teacher to keep track of them. Secondly, we deleted sections with empty comments to avoid blank processing files in our analysis. Thirdly, we deleted titles from the input. To do this, we split the texts each time an "end of the line" (" $\backslash n$ ") was detected and followed by another end of the line or by an uppercase letter. Then, we tokenized each portion of the split, which means we split them word by word in an intelligent way, and we post-tagged them. Post-tagging consists of finding the class of a word to determine whether the word is a noun, a verb, an adjective, etc. Thanks to the post-tagging process, it was possible to identify any isolated line that did not have a verb. These lines without verbs were considered titles and so removed. Finally, because our analyses are case-sensitive, we converted all the texts to their lower cases.

#### 4.4. Word Frequencies and Word Clouds

After completing the pre-processing of the comments, we then proceeded to obtain an overview of the main idea of the texts to have a base for comparing our subsequent analyses. We proposed several visualizations of the most used words. First, we determined the top ten most used words using the "most common" function of the "Counter" Python class. Then, we used the "WordCloud" Python class to complete the word cloud overview. Finally, we proposed an association table to show the other words most often associated with the top ten words. This was performed by creating a document-term matrix of the top ten words, showing each word's frequency. Then, we calculated a similarity matrix that gives the correlation score between each top ten word and the other words in the text. Finally, we kept associations with a correlation over 0.5.

#### 4.5. Sentiment Analysis

As presented in the related work discussion, sentiment analysis applied to open-ended student feedback is essential in automating students' feedback. Therefore, we explore sentiment analysis in our studies. We use the "sentiment.vader.SentimentIntensityAnalyzer"; a ready-to-use Python library to realize a simple sentiment analysis. The module was imported from the Natural Language Toolkit, a Python tool for English language processing. The sentiment analyzer based on the VADER lexicon works well for classifying social media texts, whether positive, negative, or neutral. Finally, because we had a little test set for our experiments, we manually checked the VADER-based classification to confirm that the outputs of the analyzer were correct.

#### 4.6. Emotional Analysis

Like the sentiment analysis, we knew that emotion classifiers were pre-existing, so we looked to compare them. We used two different libraries: the "text2emotion" Python library and the "nrclx" NRC lexicon-based classifier, for the emotional analysis. Our experiments found that the NRC lexicon-based classifier released by the National Research Council Canada was more accurate and had more classes.

#### 4.7. Customized Emotional Analysis

Our customized classification aimed to analyze students’ understanding, engagement, and ease of feeling regarding the evaluated course. Therefore, we first identified the following six classes of emotions: understanding, misunderstanding, interesting, boring, easy, and hard. Then, we used the same “nrclex” and the “text2emotion” libraries to automate our classification. The following sections discuss the various steps taken to analyze customized emotions.

##### 4.7.1. Building the Dictionary

A dictionary of the word is needed for both of the libraries used. Like [22], we identified as many words (verbs, adjectives, and adverbs) as possible related to each emotional class. To reduce this hand-made work, we kept only the infinitive form of verbs. The result was a list of “word: class” pairs in a text file. Figure 3 shows a summary of the customized dictionary developed for this application. In most lexicon-based works [8,22], the dictionary is complemented by including synonyms and antonyms in the WordNet library. We tried to do it, but unfortunately, we observed that the synonyms we found were often not adapted to education’s vocabulary. As a result, when we began to remove the incorrect synonyms manually, it led to removing too many words. So, we concluded that this time-consuming work poorly improved the initial lexicon, and we decided to abandon this idea and keep only our hand-made dictionary. We also found the text2emotion’s codes interesting in handling negations. For example, we managed negation with a first function that was to delete any negation’s contraction, which means replacing any “n’t” with “not”.

'understood': 'understood',	'problem': 'hard',	'unfocused': 'boring',
'understand': 'understood',	'push': 'hard',	'dreamy': 'boring',
'figure out': 'understood',	'adjust': 'hard',	'dream': 'boring',
'clear': 'understood',	'search': 'hard',	'disinterested': 'boring',
'precise': 'understood',	'look for': 'hard',	'disinteresting': 'boring',
'transparent': 'understood',	'help': 'hard',	'uninterested': 'boring',
'certain': 'understood',	'determine': 'hard',	'uninteresting': 'boring',
'trust': 'understood',	'determination': 'hard',	'trivial': 'boring',
'confident': 'understood',	'rigorous': 'hard',	'tire': 'boring',
'well-defined': 'understood',	'rigor': 'hard',	'sleepy': 'boring',
'well-define': 'understood',	'uneasy': 'hard',	'sleep': 'boring',
'explicit': 'understood',	'relentless': 'hard',	'monotonous': 'boring',
'convince': 'understood',	'disorganize': 'hard',	'too easy': 'boring',
'sure': 'understood',	'unknown': 'hard',	'numb': 'boring',
'questionless': 'understood',	'unknow': 'hard',	'pleasureless': 'boring',
'secure': 'understood',	'time consuming': 'hard',	'inanimate': 'boring',
'relax': 'understood',	'time consume': 'hard',	'unexciting': 'boring',
'at ease': 'understood',	'lot of time': 'hard',	'unexceptional': 'boring',
'untroubled': 'understood',	'long time': 'hard',	'unsurprising': 'boring',
'unworried': 'understood',	'lot of work': 'hard',	'unsurprise': 'boring',
'comfortable': 'understood',	'fronted': 'hard',	'unsurprised': 'boring',
'sure': 'understood',	'front': 'hard',	'waste': 'boring',
'limpid': 'understood',	'require': 'hard',	'wait': 'boring',
'nonambiguous': 'understood',	'try': 'hard',	'too slow': 'boring',
'comprehensible': 'understood',	'trial': 'hard',	'distract': 'boring',
'intelligible': 'understood',	'error': 'hard',	'unimportant': 'boring',
'doubtless': 'understood',	'easy': 'easy',	'useless': 'boring',
'capable': 'understood',	'easily': 'easy',	'disappoint': 'boring',
'able': 'understood',	'facile': 'easy',	'elementary': 'boring',
'competent': 'understood',	'fluid': 'easy',	'motivate': 'interesting',
'strong': 'understood',	'painless': 'easy',	'motivation': 'interesting',
'autonomous': 'understood',	'painlessly': 'easy',	'challenge': 'interesting',
'independent': 'understood',	'simple': 'easy',	'absorb': 'interesting',
'meaningful': 'understood',	'simply': 'easy',	'engage': 'interesting',
'achieve': 'understood',	'soft': 'easy',	'fascinate': 'interesting',
'achievement': 'understood',	'smooth': 'easy',	'intriguing': 'interesting',
'manage': 'understood',	'smoothly': 'easy',	'intrigued': 'interesting',
'find': 'understood',	'effortless': 'easy',	'involve': 'interesting',
'solve': 'understood',	'accessible': 'easy',	'inspire': 'interesting',
'misunderstood': 'misunderstood',	'manageable': 'easy',	'captivate': 'interesting',
'misunderstand': 'misunderstood',	'manage': 'easy',	'amaze': 'interesting',
'doubt': 'misunderstood',	'uncomplicated': 'easy',	'hypnotize': 'interesting',
'wonder': 'misunderstood',	'undemanding': 'easy',	'surprise': 'interesting',
'ask': 'misunderstood',	'untroublesome': 'easy',	'eye-opening': 'interesting',
'question': 'misunderstood',	'offhand': 'easy',	'open-eyed': 'interesting',
'guess': 'misunderstood',	'casual': 'easy',	'entertain': 'interesting',
'worry': 'misunderstood',	'no problem': 'easy',	'entertainment': 'interesting',
'anguish': 'misunderstood',	'feasible': 'easy',	'amuse': 'interesting',
'fear': 'misunderstood',	'realizable': 'easy',	'nice': 'interesting',

Figure 3. Customised hand-made emotions dictionary.

#### 4.7.2. Infinitive Text

We limited the verbs to their infinitive form to make building the dictionary easier. Moreover, because our classification relies on word counting, we keep the conjugated verbs. To count the verbs in all forms, we developed a function that identified conjugated verbs and put them into their infinitive form. Firstly, we tokenized our texts and then post-tagged the tokenized texts. Then, we found the infinitive form of the tokens identified as verbs. Next, we used the WordNetLemmatizer function of the nltk.stem library on each of the tokens identified as a verb by the post-tagging. This function lemmatizes the word by finding its lemma, the form of the word in the dictionary. The problem with this solution is that it ignored the past participle of verbs conjugated with an auxiliary. This is because the verbs are tagged as adjectives. To overcome this issue, we removed any verb identified as an auxiliary, such as 'be', 'can', 'could', 'dare', 'do', 'have', 'may', 'might', 'must', 'need', 'ought', 'shall', 'should', 'will', and 'would'. This was further simplified by removing negation contractions in the previous step. After removing the auxiliaries, we post-tagged the text a second time to make the past participles identified as verbs and repeated the steps. Finally, we converted the text to their plural nouns from their singular form to fit our dictionary completely.

#### 4.7.3. Taking Care of Negations: Not before Words

Because our classification is based on word counting, it naturally does not care about the "not" before words. To manage that, we were inspired by the text2emotion library and adapted the code for our work. To take care of negations, we created a "contrary dictionary". Our emotion classes were chosen so that each one was paired with its opposite; as we can see, a lesson can be understood or misunderstood, interesting or boring, and easy or hard. Thus, we built an antonym dictionary where each word composing the initially built dictionary becomes preceded by "not" and associated with its opposite class. For instance, the pair "difficult: hard" found in the initial dictionary becomes "not difficult: easy" for the opposite. Likewise, we put the whole text into its infinitive form. We changed them again to put them in their positive form by replacing any "not + word" association found in the contrary dictionary with their corresponding class name, knowing that for each class, the initial dictionary has an entry "word: class" where word = class.

#### 4.7.4. Word Counting and Exploitation

The final step in the customized emotion analysis consists of word counting. We provided the infinitive and positive forms of our texts as input to a function that counts every appearance of a word in our customized dictionary before summing the number of times each class is encountered. In this way, we have the frequencies of each education-related feeling. Thus, we can have an idea of whether the course was globally understood or misunderstood, interesting or boring, and easy or hard.

#### 4.8. Classification According to Bloom's Taxonomy

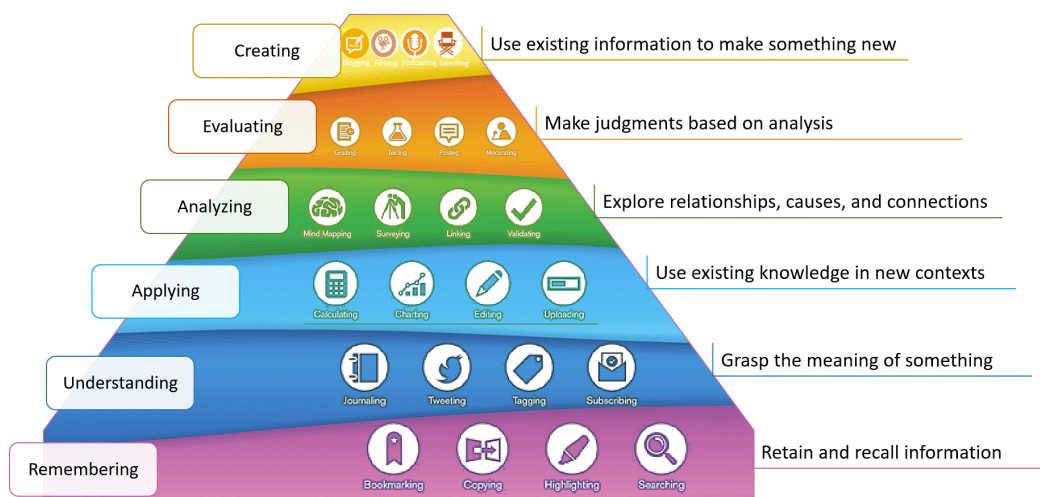
Bloom's digital taxonomy aims to identify actions related to each learning level in open-ended student feedback to evaluate the student's knowledge regarding the lesson. The previous work on the customized emotion analysis can be applied here as well, using the classes of Bloom's taxonomy. We tried to implement the previous work to identify actions of remembering, understanding, applying, analyzing, evaluating, and creating to finally estimate the level of students' learning thanks to the hierarchy between those classes. As in the customized emotion analysis, a few steps were necessary. Among these steps, removing negations' contractions and putting the text in its infinitive form remained the same as previously discussed. The following are some of the steps that were altered to achieve the classification.

### 4.8.1. Building the Dictionary

To build the dictionary that pairs each class to its action verbs, we directly used Bloom’s digital taxonomy, as shown in Figure 4 and Table 2. Here we just chose to use the digital taxonomy instead of the classical one to fit more of today’s methods and tools students have for learning and practicing. As a result, we had a list of “word: class” pairs in a text file as we did in the customized emotion dictionary.

**Table 2.** Bloom’s Taxonomy Digital Planning Verbs [14].

Remembering	Understanding	Applying	Analyzing	Evaluating	Creating
Copying	Annotating	Acting out	Calculating	Arguing	Bloggng
Defining	Tweeting	Articulate	Breaking- down	Validating	Building
Finding	Associating	Reenact	Correlating	Testing	Animating
Locating	Tagging	Loading	Deconstructing	Scoring	Adapting
Quoting	Summarizing	Determining	Linking	Assessing	Collaborating
Listening	Relating	Displaying	Mashing	Criticizing	Composing
Googling	Categorizing	Judging	Mind-mapping	Commenting	Directing
Repeating	Paraphrasing	Executing	Organizing	Debating	Devising
Outlining	Predicting	Examining	Appraising	Defending	Podcasting
Highlighting	Comparing	Implementing	Advertising	Detecting	Writing
Memorizing	Contrasting	Sketching	Dividing	Grading	Filming
Networking	Commenting	Experimenting	Deducing	Hypothesizing	Programming
Searching	Interpreting	Hacking	Distinguishing	Measuring	Simulating
Identifying	Grouping	Interviewing	Illustrating	Moderating	Role playing
Selecting	Inferring	Painting	Questioning	Posting	Solving
Duplicating	Estimating	Preparing	Structuring	Predicting	Mixing
Matching	Extending	Playing	Integrating	Rating	Facilitating
Bookmarking	Gathering	Integrating	Attributing	Reflecting	Managing
Bullet-pointing	Exemplifying	Presenting	Estimating	Reviewing	Negotiating
	Expressing	Charting	Explaining	Editorializing	leading



**Figure 4.** Bloom’s Taxonomy: activities with digital tools [14].

### 4.8.2. Taking Care of Negations: Not before Words

After the negations’ contractions were removed and the texts were put in their infinitive form, we needed to construct the contrary dictionary. It was unique in the case of Bloom’s taxonomy because we established that a “not” before an action verb of the classification was canceling the learning action. Therefore, the contrary dictionary only consisted of adding each action verb preceded by “not” in pairs with nothing. So, for instance, the pair “repeat: remembering” found in the initial dictionary becomes “not repeat” “on the contrary one”. In this way, we could easily apply the same function as we had in the



customized emotion analysis that replaced any “not + word” with its corresponding class on the contrary dictionary, which is, in this case, nothing.

#### 4.8.3. Word Counting and Exploitation

As in the customized emotion analysis, we finished by conducting the word counting to have the frequencies of each Bloom’s taxonomy level from the student’s learning actions. Eventually, to evaluate the student’s learning level, we assumed that it was given by the one the student talked about the most.

### 5. Results and Discussion

In this section, we will demonstrate the outputs of the proposed application, and we will discuss their usefulness. Because the number of files containing the students’ feedback that we need to analyze is small, we manually validated the generated outputs. Thus, the validation processes were conducted by reading, reviewing, and checking all the students’ feedback files against the generated outputs by all the authors involved in this work. For all the files we manually validated, the corresponding generated outputs reflect the same sentiments we observed in the students’ feedback.

#### 5.1. Classic NLP Analysis

Figure 5 shows the top 10 most used words in each lesson and their number of appearances in the gathered feedback files. The chart gives a great idea of what students retained from the lessons. Most of the time, only the root of words appears because we stemmed the text before. The first top 10 words chart is complemented by the word cloud, which allows us to see more of the most used words and their importance. Figure 6 shows the generated word cloud for the case studied in our work. Finally, a table of the most frequent words association including the top 10 words was generated based on a minimum correlation coefficient of 0.5 (see Table 3). The table gives an idea of the context in which the students used the words.

**Table 3.** Samples of the word association table extracted from all lessons.

	Correlation
database, nosql	0.7445189320116243
database, not	0.6990950777289033
database, store	0.676294010356556
database, data	0.6109001287282151
database, type	0.5527805688124333
database, relate	0.5065160024362998
data, store	0.6503310481300625
data, database	0.6109001287282151
data, table	0.5059633235464172
use, table	0.6715488196512599
use, relate	0.6300322498066675
use, inform	0.6047209393116116
use, store	0.5161765495475054
use, nosql	0.5052344607758218

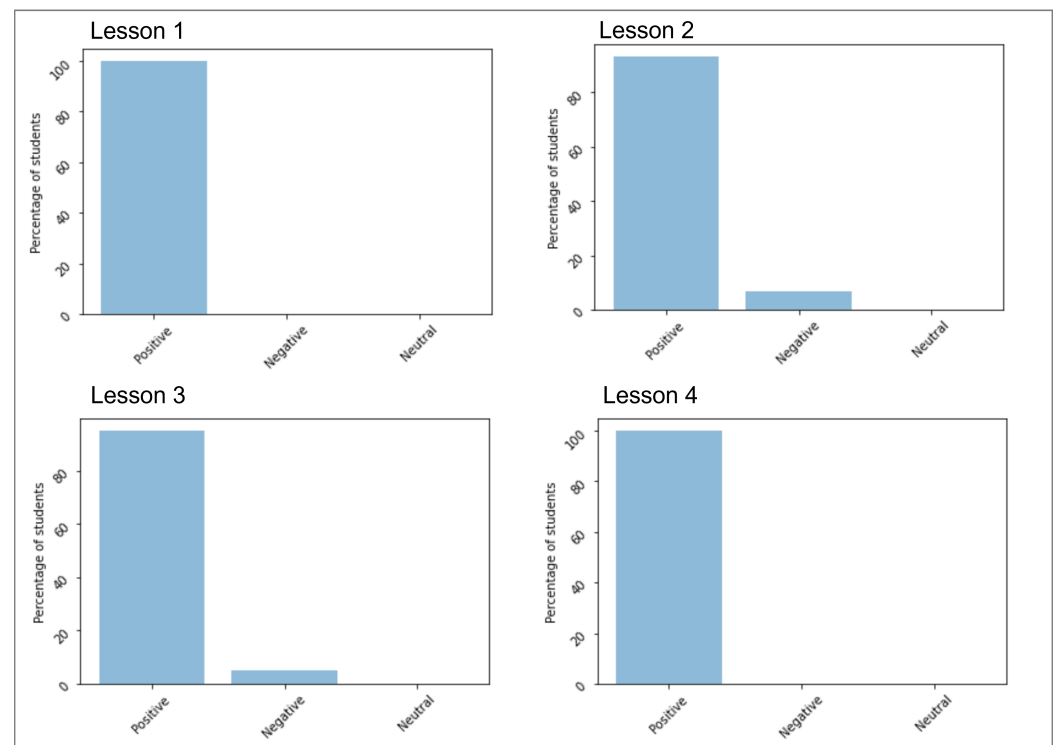


### 5.2. Sentiment Analysis

As we mentioned earlier, we applied the sentiment analysis using the sentiment analyzer based on the VADER lexicon. This technique helps to detect the text polarity and classify it into three different classifications: positive, neutral, or negative. The generated PDF synthesis reports show the results of the sentiment analysis (see Figure 7). As in [28,29] we used the following thresholds to classify the scores of sentiment analysis:

- Positive (score between 0.05 and 1);
- Negative (score between  $-0.05$  and  $-1$ );
- Neutral (score between  $-0.05$  and 0.05).

Figure 7 shows the percentage of the students' feedback that is positive, negative, or neutral for the feedback files reviewed in our experiments. By looking at the results, which show the sentiment scores to the text collected from each lesson, we can conclude the overall sentiment of the students are positive, which means that their feedback includes a majority of positive words that conveys their positive emotions about the lessons.



**Figure 7.** Sentiment analysis for each lesson.

### 5.3. Emotion Analysis

We applied the emotion analysis using an NLP tool that calculates the number of times anger, anticipation, disgust, fear, joy, sadness, surprise, and trust emotions are identified in the text. The pie charts shown in Figure 8 summarize the frequency of detecting each emotion in the students' feedback of each lesson. We can conclude that the dominant feelings in all the lessons are trust and anticipation, which are positive feelings that reflect the students' positive feedback about the lessons. Table 4 shows an overview of the number of students who expressed each specific emotion. We can conclude that most of the students experienced and showed the same emotions in their feedback.

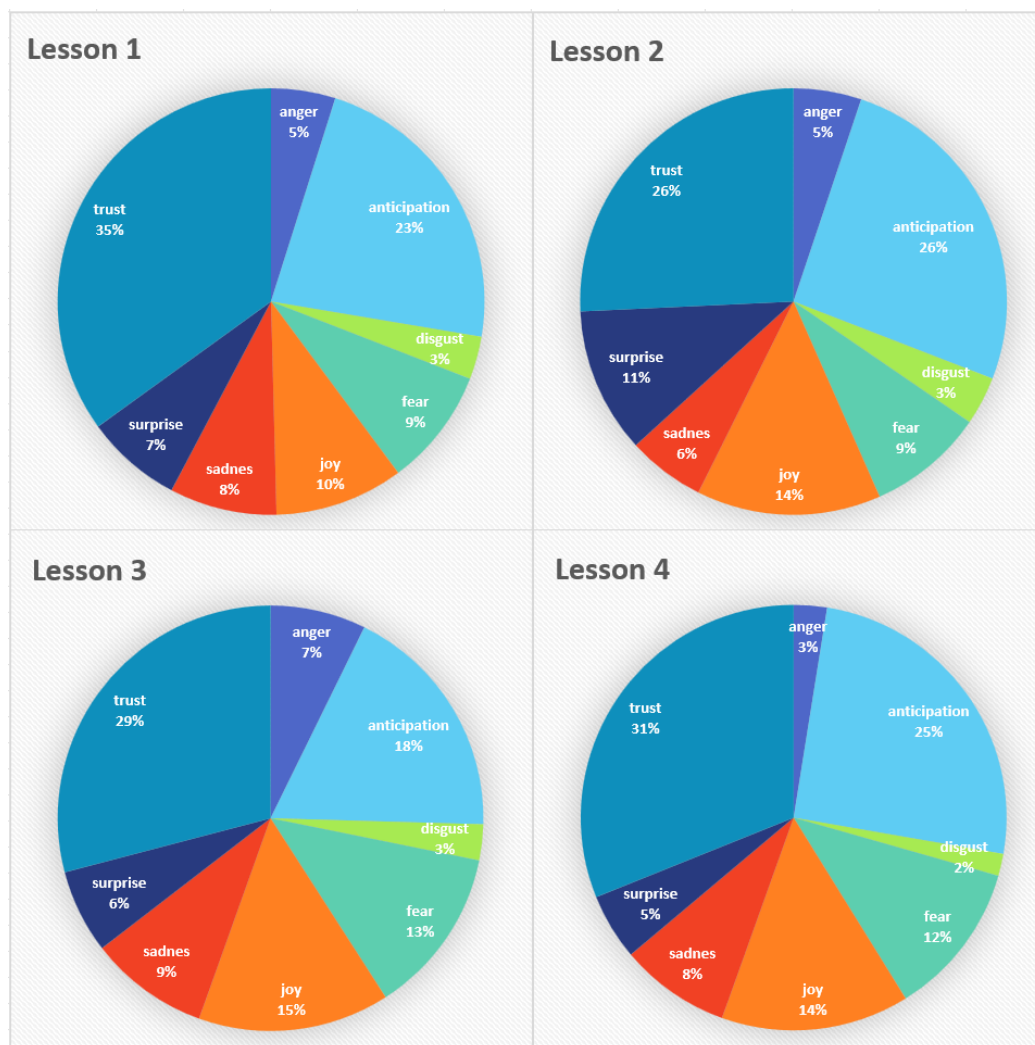


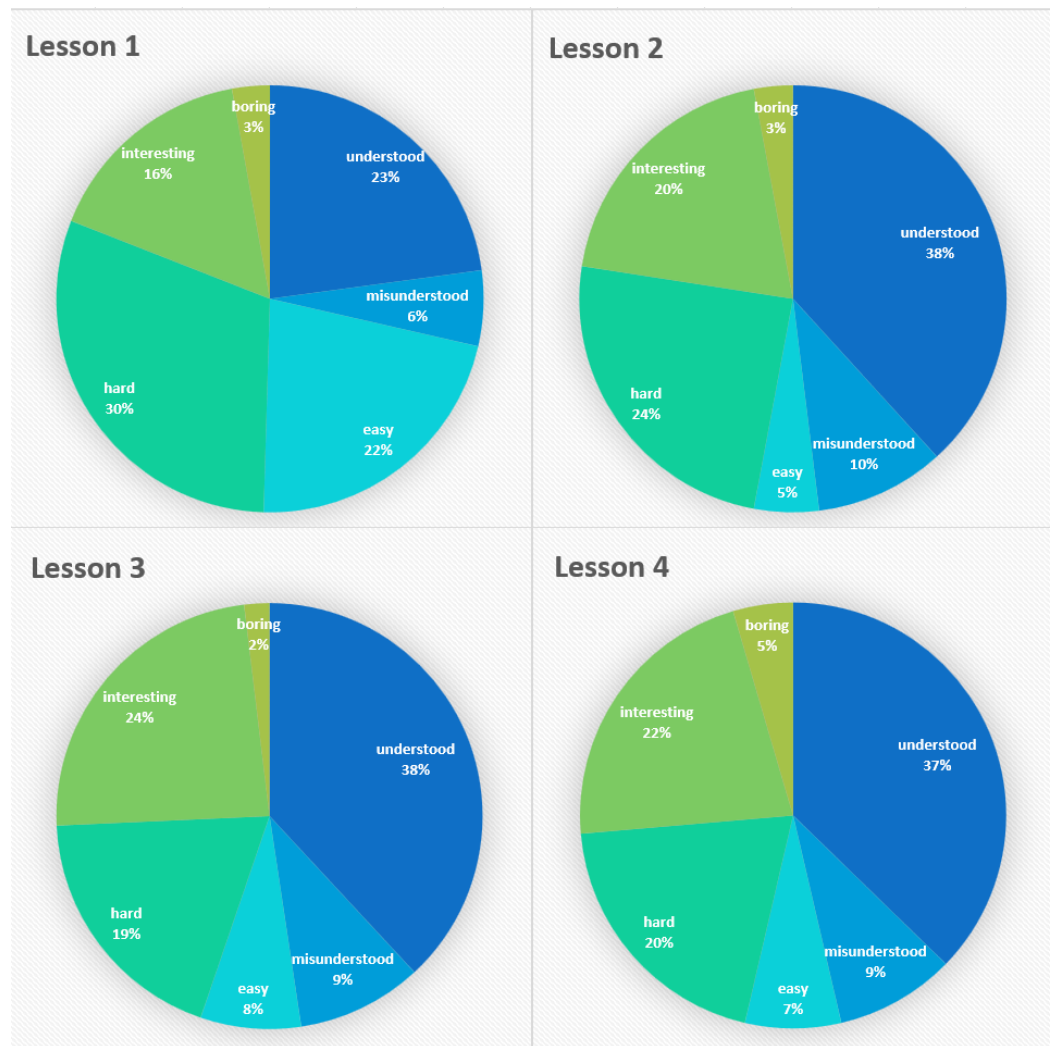
Figure 8. Frequency of emotions detection in each lesson.

Table 4. Number of students who expressed each emotion.

Lesson ID	Lesson	Anger	Anticipation	Disgust	Fear	Joy	Sadness	Surprise	Trust
101496550	Lesson 1	15	16	15	16	16	16	16	16
102078874	Lesson 2	13	15	8	13	14	12	13	13
103999562	Lesson 3	16	19	8	17	17	14	15	18
104161929	Lesson 4	14	20	11	18	19	16	16	19

#### 5.4. Customized Emotion Analysis

The customized emotion analysis considers the educational needs in evaluating the quality of the lessons by using the customized dictionary built in the implementation phase. The dictionary was built to detect the following feelings: understood, misunderstood, easy, hard, interesting, or boring. Similar to the process implemented in the emotion analysis, the pie charts shown in Figure 9 summarize the frequency of detecting each of the customized emotions in the students’ feedback of each lesson. We can conclude that some of the students feel that the lessons were hard but still interesting and understandable.



**Figure 9.** Frequency of customised emotions detection in each lesson.

### 5.5. Bloom's Taxonomy Analysis

Applying Bloom's taxonomy analysis to our students' feedback helps to classify their learning actions based on some developed abilities and skills. Our analysis detects different students' actions in their feedback. Those actions exhibit the different levels of abilities described in Bloom's taxonomy which include remembering, understanding, analyzing, evaluating, and creating. Following the same analytical pattern, the pie charts shown in Figure 10 summarize the frequency of the detected actions and verbs representing Bloom's abilities. We can conclude that the most detectable ability in all lessons is the remembering ability which is the lowest cognitive level of outcomes. As expected, this level is recognizable by observation and recalling a wide range of information. It is also attached to the use of more general verbs such as search, finds, and select. In Table 5, we show the number of students expressing actions related to each of Bloom's abilities at least once. We can conclude that the collected feedback succeeded in representing the lowest cognitive level which is the remembering level for most of the students. The collected feedback also included some indications for the highest cognitive level, which is the creating level for approximately half of the students.

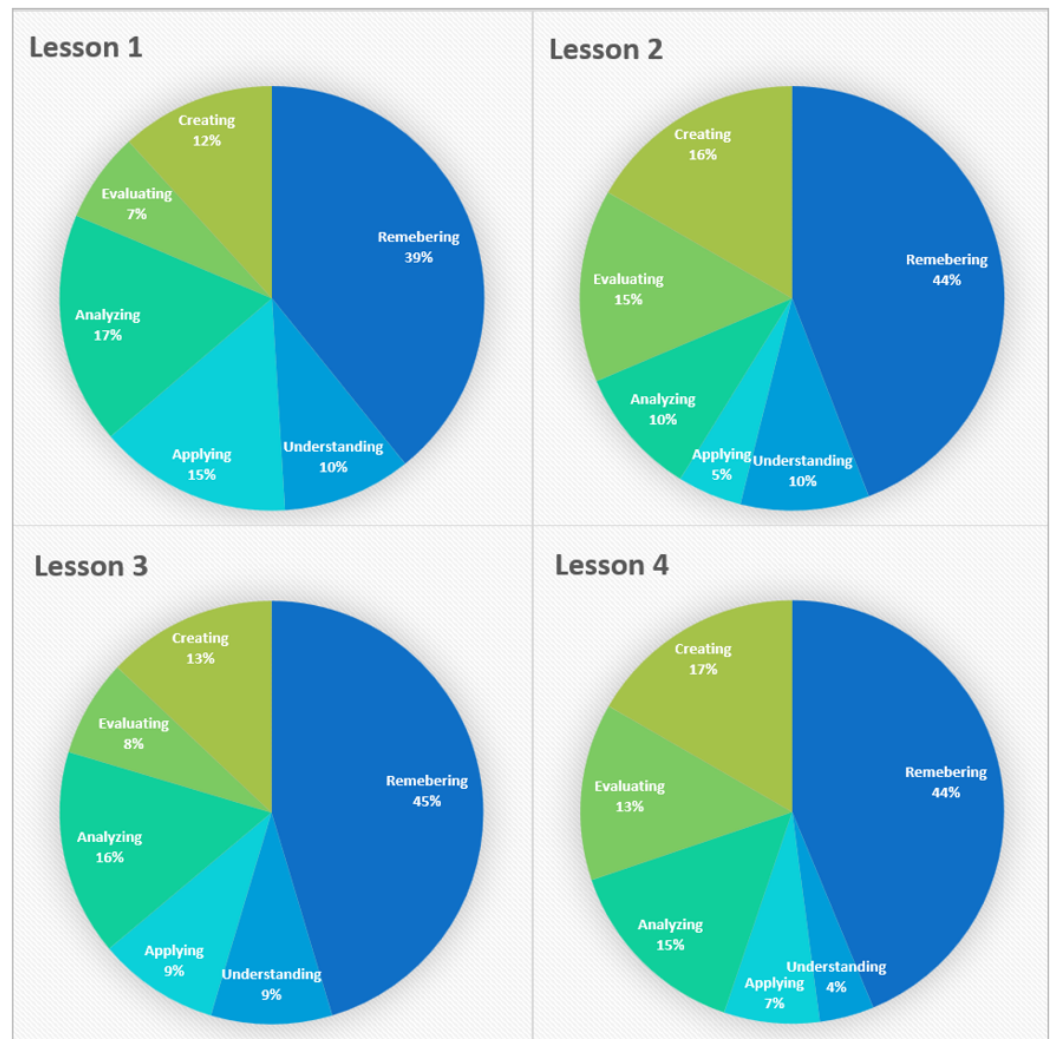
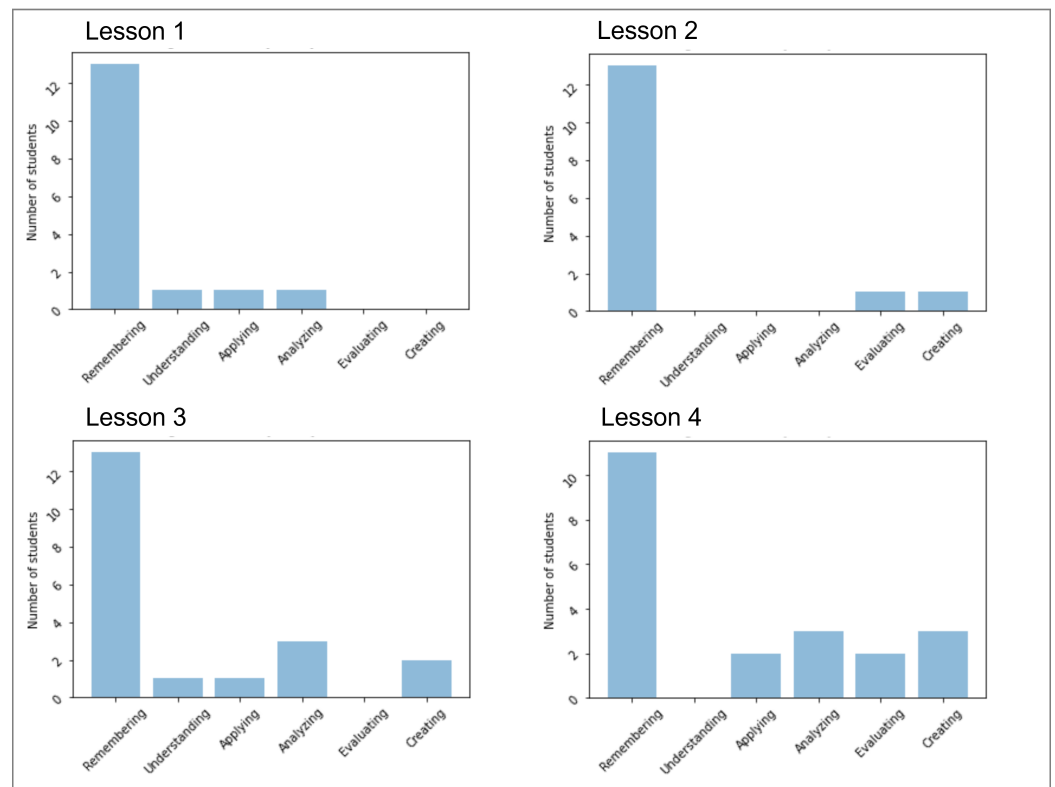


Figure 10. Frequency of Bloom’s abilities detected in each lesson.

Table 5. Number of students who expressed each ability.

Lesson	Remembering	Understanding	Applying	Analyzing	Evaluating	Creating
Lesson 1	16	12	15	15	10	8
Lesson 2	13	9	8	11	11	9
Lesson 3	18	10	9	11	7	10
Lesson 4	18	8	7	12	7	15

By detecting digital learning verbs and actions, we also managed to estimate the learning level of each student in each lesson based on the different abilities detected in their feedback. Figure 11 shows the student learning level in each of the lessons according to the taxonomy. Based on the students’ general feedback, it is expected that the detected level for most of the students is the remembering level because they discussed and informed their general feedback without narrowing that into more detailed and technical aspects, which could indicate achieving the upper cognitive levels.



**Figure 11.** Students' learning level in each lesson according to Bloom's taxonomy.

## 6. Conclusions

In this paper, we explore sentiment and emotion analysis techniques to develop a tool that can assist teachers in effectively analyzing students' open-ended feedback at the end of a course review, thus helping with improving their course delivery. First, a dictionary dedicated to the classification of education-related feelings, such as understanding, misunderstanding, interesting, boring, easy, and hard, was developed for identifying key issues of lessons through open-ended student feedback. Then, an automated learning level detection based on Bloom's taxonomy was proposed. For both classifications, we based the solutions on lexicon and counting of occurrences of words found in the students' feedback. Finally, we implemented a user-friendly interface with a ready-to-read PDF report of students' feedback analysis as outputs. This research effort is a step in facilitating the creation of an automated student feedback analyzer, whose application can also be extended to other domains where open-ended feedback is the norm. The tool is currently useful for mainly analyzing feedback written in English. However, the proposed techniques used can be adapted to text written in other languages by generating language-specific educational-related dictionaries, etc. This paper offers improvement opportunities such as enhancing the customized dictionary, analyzing and comparing multiple feedback to see the students' progress with time, and adding an automatic recommendation based on the feedback analysis.

**Author Contributions:** M.F., carried out this research as part of her internship research project; S.K., I.A.L. and S.Y., were involved in planning and supervising the work and drafting and reviewing the manuscript; H.T.R. participated as an external team member to moderate and guide the research towards positive outcomes. The methodology, analysis, and discussion were conducted by all participants. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Data are available upon request.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Hedderich, M.A.; Lange, L.; Adel, H.; Strötgen, J.; Klakow, D. A Survey on Recent Approaches for Natural Language Processing in Low-Resource Scenarios. In Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Online, 6–11 June 2021; pp. 2545–2568. [\[CrossRef\]](#)
2. Alhawiti, K.M. Natural Language Processing and its Use in Education. *Int. J. Adv. Comput. Sci. Appl.* **2014**, *5*, 72–76. [\[CrossRef\]](#)
3. Litman, D. Natural Language Processing for Enhancing Teaching and Learning. In Proceedings of the AAAI Conference on Artificial Intelligence, Phoenix, AZ, USA, 12–17 February 2016; Volume 30. [\[CrossRef\]](#)
4. Elouazizi, N.; Birol, G.; Jandciu, E.; Öberg, G.; Welsh, A.; Han, A.; Campbell, A. Automated Analysis of Aspects of Written Argumentation. In Proceedings of the 7th International Learning Analytics & Knowledge Conference (LAK'17), Vancouver, BC, Canada, 13–17 March 2017; Association for Computing Machinery: New York, NY, USA, 2017; pp. 606–607. [\[CrossRef\]](#)
5. Gao, Y.; Davies, P.M.; Passonneau, R.J. Automated Content Analysis: A Case Study of Computer Science Student Summaries. In Proceedings of the 13th Workshop on Innovative Use of NLP for Building Educational Applications, New Orleans, LA, USA, 5 June 2018; Association for Computational Linguistics: New Orleans, Louisiana, 2018; pp. 264–272. [\[CrossRef\]](#)
6. Altrabsheh, N.; Cocea, M.; Fallahkhair, S. Learning Sentiment from Students' Feedback for Real-Time Interventions in Classrooms. In *Adaptive and Intelligent Systems*; Bouchachia, A., Ed.; Springer International Publishing: Cham, Switzerland, 2014; pp. 40–49.
7. Cutrone, L.A.; Chang, M. Automarking: Automatic Assessment of Open Questions. In Proceedings of the 10th IEEE International Conference on Advanced Learning Technologies, Sousse, Tunisia, 5–7 July 2010; pp. 143–147. [\[CrossRef\]](#)
8. Kastrati, Z.; Dalipi, F.; Imran, A.S.; Pireva Nuci, K.; Wani, M.A. Sentiment Analysis of Students' Feedback with NLP and Deep Learning: A Systematic Mapping Study. *Appl. Sci.* **2021**, *11*, 3986. [\[CrossRef\]](#)
9. Hynninen, T.; Knutas, A.; Hujala, M. Sentiment analysis of open-ended student feedback. In Proceedings of the 43rd International Convention on Information, Communication and Electronic Technology, Opatija, Croatia, 28 September–2 October 2020; pp. 755–759. [\[CrossRef\]](#)
10. Nasim, Z.; Rajput, Q.; Haider, S. Sentiment analysis of student feedback using machine learning and lexicon based approaches. In Proceedings of the International Conference on Research and Innovation in Information Systems, Seoul, Republic of Korea, 16–17 July 2017; pp. 1–6. [\[CrossRef\]](#)
11. Ren, P.; Yang, L.; Luo, F. Automatic scoring of student feedback for teaching evaluation based on aspect-level sentiment analysis. *Educ. Inf. Technol.* **2022**, *28*, 797–814. [\[CrossRef\]](#)
12. Brown, R.B. Contemplating the Emotional Component of Learning: The Emotions and Feelings Involved when Undertaking an MBA. *Manag. Learn.* **2000**, *31*, 275–293. [\[CrossRef\]](#)
13. Fineman, S. Emotion and Management Learning. *Manag. Learn.* **1997**, *28*, 13–25. [\[CrossRef\]](#)
14. Churches, A. Bloom's digital taxonomy. In *Bloom's Revised Digital Taxonomy Map*; Tech & Learning: Washington, DC, USA, 2008; pp. 6–8.
15. Anderson, L.W.; Sosniak, L.A.; Bloom, B.S. *Bloom's Taxonomy: A Forty-Year Retrospective*; University of Chicago Press: Chicago, IL, USA, 1996.
16. Krathwohl, D.R. *A Taxonomy for Learning, Teaching and Assessing: A Revision of Bloom's Taxonomy of Educational Objectives*; Longman: New York, NY, USA, 2008.
17. Bloom, B.S. *Taxonomy of Educational Objectives, Handbook I: The Cognitive Domain*; David McKay Co., Inc.: Philadelphia, PA, USA, 1956.
18. Khurana, D.; Koli, A.; Khatter, K.; Singh, S. Natural language processing: State of the art, current trends and challenges. *Multimed. Tools Appl.* **2023**, *82*, 3713–3744. [\[CrossRef\]](#) [\[PubMed\]](#)
19. Ferreira-Mello, R.; André, M.; Pinheiro, A.; Costa, E.; Romero, C. Text mining in education. *WIREs Data Min. Knowl. Discov.* **2019**, *9*, e1332. [\[CrossRef\]](#)
20. Ahadi, A.; Singh, A.; Bower, M.; Garrett, M. Text Mining in Education & A Bibliometrics-Based Systematic Review. *Educ. Sci.* **2022**, *12*, 210. [\[CrossRef\]](#)
21. Lugini, L.; Litman, D.; Godley, A.; Olshefski, C. Annotating Student Talk in Text-based Classroom Discussions. In Proceedings of the 13th Workshop on Innovative Use of NLP for Building Educational Applications, New Orleans, LA, USA, 5 June 2018; Association for Computational Linguistics: New Orleans, Louisiana, 2018; pp. 110–116. [\[CrossRef\]](#)
22. Chong, C.; Sheikh, U.U.; Samah, N.A.; Sha'ameri, A.Z. Analysis on Reflective Writing Using Natural Language Processing and Sentiment Analysis. *IOP Conf. Ser. Mater. Sci. Eng.* **2020**, *884*, 012069. [\[CrossRef\]](#)
23. Asghar, M.Z.; Ullah, I.; Shamshirb, S.; Khundi, F.M.; Habib, A. Fuzzy-Based Sentiment Analysis System for Analyzing Student Feedback and Satisfaction. *Comput. Mater. Contin.* **2020**, *62*, 631–655. [\[CrossRef\]](#)
24. Okoye, K.; Arrona-Palacios, A.; Camacho-Zuñiga, C.; Achem, J.A.; Escamilla, J.; Hosseini, S. Towards teaching analytics: A contextual model for analysis of students' evaluation of teaching through text mining and machine learning classification. *Educ. Inf. Technol.* **2022**, *27*, 3891–3933. [\[CrossRef\]](#) [\[PubMed\]](#)



25. Zad, S.; Jimenez, J.; Finlayson, M. Hell Hath No Fury? Correcting Bias in the NRC Emotion Lexicon. In Proceedings of the 5th Workshop on Online Abuse and Harms, Online, 5–6 August 2021; pp. 102–113. [[CrossRef](#)]
26. Banage, T.G.S.; Kumara, A.B.; Paik, I. Bloom’s Taxonomy and Rules Based Question Analysis Approach for Measuring the Quality of Examination Papers. *Int. J. Knowl. Eng.* **2019**, *5*, 20–24.
27. Bird, S.; Klein, E.; Loper, E. *Natural Language Processing with Python: Analyzing Text with the Natural Language Toolkit*; O’Reilly Media, Inc.: Sebastopol, CA, USA, 2009.
28. Al-Shabi, M.A. Evaluating the performance of the most important Lexicons used to Sentiment analysis and opinions Mining. *Int. J. Comput. Sci. Netw. Secur.* **2020**, *20*, 51–57.
29. Bonta, V.; Kumares, N.; Janardhan, N. A Comprehensive Study on Lexicon Based Approaches for Sentiment Analysis. *Asian J. Comput. Sci. Technol.* **2019**, *8*, 1–6. [[CrossRef](#)]

**Disclaimer/Publisher’s Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.