Alaa Asim Qaffas

AN IMPROVED CONTEXT-AWARE SENTIMENT ANALYSIS OF STUDENT COMMENTS ON SOCIAL NETWORKS BASED ON ChatGPT

Abstract *The widespread use of social networks has provided a variety of active, dynamic, and popular platforms for students to express their opinions and sentiments. These data are increasingly being exploited and integrated into university information systems to better govern and manage universities and improve educational quality. The analysis of such data can offer valuable insights into student experiences and attitudes towards various educational aspects including courses, professors, events, and facilities. However, automatic opinion mining in this context is challenging due to the difficulty of analyzing some languages such as Arabic, the variety of used languages, the presence of informal language, the use of emoticons and emoji, sarcasm, and the need to consider the surrounding context. To deal with all these challenges, we propose a novel approach for an effective sentiment analysis of student comments on the X platform (Twitter). The proposed approach allows the collection of student comments from public Twitter pages and automatically classifies comments into positive, negative, and neutral. The new approach is based on ChatGPT capabilities, supports three languages: English, Arabic, and colloquial Arabic, and integrates a new scoring method that measures both the positiveness and subjectivity of student comments. Experiments performed on simulated and real public Twitter pages of five Saudi high education institutions showed the performance of the proposed tool to analyze and summarize collected data automatically.*

Keywords sentiment analysis, educational social networks, student opinion mining, Lexicon-based approach, transformer-based methods

Citation Computer Science 26(1) 2025: 133–164

1. Introduction

Social media platforms, such as “Facebook”, “Twitter”, “Instagram”, “LinkedIn” and “TikTok” have emerged as popular spaces where people can freely express their thoughts, opinions, and personal experiences. These platforms offer users an open and free environment to create profiles, establish connections with friends, communicate with family members, and share several types of online content including documents, photos, videos, and many other online resources. Social platforms enable boosting interactions and engagements in the network by allowing users to easily like, comment on, or share each other’s posts. The large number of users and the wide range of activities on these platforms make them valuable resources across various domains. For example, in the marketing domain, social networks provide a powerful platform for businesses to promote their products or services [9, 19]. In healthcare, social networks are utilized to disseminate health-related information and connect patients with support groups [6, 16].

In the educational domain, social networks have been extensively used by educational institutions such as schools, faculties, colleges, and universities to improve their digital presence and engage with their students, alumni, and broader communities [12, 13, 20]. These pages can be either official, created by the educational institutions, or unofficial, created by other parties. In any case, these pages on social networks allow students to express their feedback on various aspects of their academic journey, including courses, professors, campus facilities, and overall university experiences. Data collected from these platforms are crucial for universities to understand student thoughts, identify areas of improvement, and enhance the overall educational experience [20]. For example, identifying the opinion of students regarding course content allows universities to identify the courses that need new evaluations and improvements. This task is referred to as **sentiment analysis** (or **opinion mining**) of student comments.

The sentiment analysis of student comments on social networks presents several challenges that must be addressed to accurately detect the emotional tone expressed in collected comments. One major challenge is the unstructured and informal nature of student comments on social media platforms. These comments often include slang, abbreviations, emojis, and other informal language constructs that complicate the text processing and analysis stage. The unstructured textual comments require comprehensive pre-processing tasks to organize them into a structured format to facilitate their effective processing by learning algorithms. Another significant challenge is the inherent ambiguity and subjectivity of natural language. Sentiment analysis algorithms must discern the context and nuanced meanings of words to accurately detect the correct sentiment. The same term can convey different sentiments depending on its context. Texts containing subjective opinions often do not clearly indicate a positive or negative sentiment and usually require sophisticated natural language processing techniques to effectively interpret contextual meanings. Another critical challenge in sentiment analysis is the difficulty of processing texts written in languages with complex morphological structures or texts written in colloquial languages. For instance, the Arabic language presents unique challenges due to its rich morphology, special characters, and right-to-left writing direction. Similarly, colloquial languages or regional dialects pose significant challenges due to their informal nature and regional variations. These dialects often include specific terms, idiomatic expressions, and regional variations that are not well-represented in formal language corpora. Existing sentiment analysis tools, that can be effective for standard versions of languages, often do not perform well with Arabic and colloquial languages. Advanced language processing techniques are required to handle the complexities of these languages and accurately analyze sentiments in diverse linguistic contexts.

To deal with all the discussed challenges, we propose in this paper an innovative approach, centered around an improved context-aware sentiment analysis method, able to effectively collect, analyze, and classify student comments on Twitter public pages related to higher education institutions into positive, negative, and neutral categories. The proposed approach uses automated techniques to gather a substantial volume of student comments from Twitter pages and then pre-process the unstructured textual comments to facilitate a comprehensive analysis of student sentiment. The proposed approach has been designed to support formal and informal languages to better recognize student opinions and thoughts. By combining the strengths of lexicon-based and transformer-based methods, the proposed approach builds sentiment polarity scores for student comments through three phases leveraging both TextBlob and ChatGPT capabilities and taking into account language nuances and subjectivity of student comments.

The major contributions of this paper can be summarized as follows:

- Design of a New Hybrid Sentiment Analysis Model. We introduce an innovative hybrid sentiment analysis model that effectively detects sentiments in student comments collected from social networks. The model combines the strengths of transformer-based methods (such as ChatGPT) and lexicon-based approaches (such as TextBlob), providing a comprehensive and accurate sentiment detection framework.
- Exploration of Transformative Models for Complex Languages. We explore and evaluate the capability of transformer models, such as GPT-turbo-3.5, in enhancing sentiment analysis for languages with complex morphological structures and non-rich corpora, such as Arabic. We also examine their effectiveness in processing text written in colloquial languages, addressing the challenges of informal and region-specific dialects.
- Comparative Analysis of Model Effectiveness. A comparative analysis of the proposed hybrid model against conventional machine learning, deep learning, and transformer-based models is conducted. The comparison demonstrates the superior performance of the hybrid contextual-based approach in accurately detecting sentiments in student comments across different languages, including English, Arabic, and colloquial Arabic.

The rest of this paper is organized as follows: Section 2 gives an overview and a literature review of sentiment analysis in the educational domain. Then, Section 3 describes the steps and data analytic components of the proposed sentiment analysis of student comments on the Twitter platform whereas Section 4 gives the empirical experiments that were performed to show the effectiveness of the proposed approach. Finally, Section 5 gives concluding remarks and future direction to improve this work.

2. Sentiment analysis in the educational domain

2.1. Sentiments analysis

Sentiment analysis is an automatic processing of textual data that automatically detects the emotional tone expressed in texts such as positiveness, negativeness, sadness, happiness, etc. Automatic sentiment analysis has been applied in several domains such as e-commerce, politics, and sports. For instance, in e-commerce, sentiment analysis has enabled companies to gain insights into customer satisfaction levels, and marketing strategies based on the sentiment analysis of customer reviews and feedback. Another example can be cited from politics where sentiment analysis is applied to understand voter sentiment by automatically assessing positiveness towards candidates, policies, political campaigns, etc.

Independently of the specificity of each domain, the conventional sentiment analysis process is usually based on text analytic techniques and typically involves four steps: Text Pre-processing step, Tokeneziation and feature extraction step, Sentiment classification step, and Sentiment interpretations as described in Figure 1.

In the first step, text pre-processing, textual data are collected, cleaned, and pre-processed to remove any irrelevant information, such as special characters, punctuation, and stop-words. Then, in the Tokenization step, the text is divided into smaller units called tokens (i.e. words, phrases, or even sentences). Tokenization enables the breaking down of large textual data into meaningful units ready to be analyzed. Once tokens are prepared, the most relevant features are extracted from the tokens to better represent the studied text. These features can include words, n-grams, parts of speech, or any other syntactic patterns. The choice of selected and relevant features depends on the specific sentiment analysis technique being used. Once the text is represented by features, a classification model is applied to assign sentiment labels to the text. Various sentiment analysis approaches can be used to determine the sentiment orientation of textual units including lexicons-based, corpus-based, machine learning, and deep learning approaches. The final step, sentiment interpretations, aims to interpret the sentiment scores or categories that range from a binary sentiment classification (positive or negative) to a more fine-grained sentiment analysis that includes several categories or sentiment intensity scores.

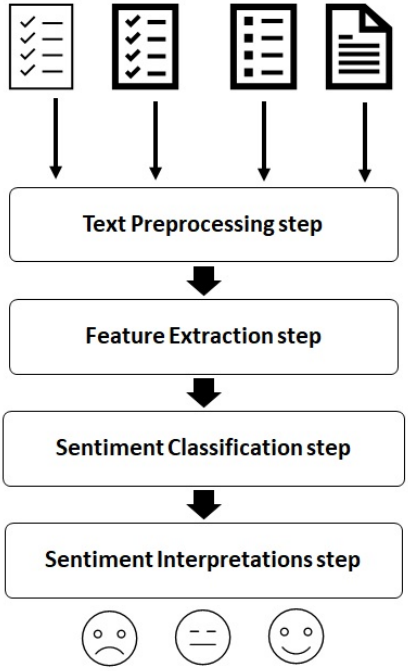


Figure 1. Four steps of the sentiment analysis process: text pre-processing, feature extraction, sentiment classification, sentiment interpretations

2.2. Literature review of sentiment analysis techniques in the educational domain

In the educational domain, sentiment analysis has gained significant attention as it allows an automatic analysis of student feedback on various educational resources such as course contents, teaching experiences, and instructor abilities. Several approaches have been adopted to perform such automatic sentiment analysis within the educational domain. Existing approaches can be classified into three categorie: rule-based, machine learning, and hybrid approaches. Rule-based approaches utilize predefined sets of sentiment words and linguistic rules to classify sentiment polarity. On the other hand, machine learning techniques involve training classifiers on labeled datasets to automatically recognize sentiment patterns. Hybrid models leverage the strengths of both approaches for improved accuracy and flexibility.

Rule-based approaches can be based on a single lexicon or on a whole corpus. The first, lexicon-based, relies on sentiment lexicons or dictionaries that contain words or phrases with pre-assigned sentiment polarity, (i.e. positive, negative, and neutral). Sentiment scores are assigned to text based on the frequency of sentiment words in the lexicon. The lexicon-based approach has been used to label educational data without a need for an excessive manual labeling step. In [31] the authors evaluated the impact of two active learning methods, flipped classrooms and lightweight teams, on student emotions by using a lexicon-based sentiment analysis approach. The authors utilized the National Research Council (NRC) lexicon to assign fine-grained emotions including joy, fear, trust, anger, sadness, disgust, and anticipation. The study was based on both quantitative and qualitative student feedback. Quantitative feedback was measured through Likert scales whereas qualitative feedback was determined through student feelings and opinions about the course. The results indicate that the implementation of these active learning methods is associated with increased positivity in student emotions. Some work [5, 25] tried to improve lexicon-based sentiment analysis in the educational domain either by using open textual feedback or by proposing customized sentiment lexicons for the educational domain. Rajput et al. used open-ended student feedback to evaluate teachers. Textual student feedback was analyzed by employing various text-analytic techniques to build a sentiment analysis-based metric to determine the teacher score. This technique allowed a deeper evaluation of teachers' performance. in another lexicon-based approach, Chauhan et al. [5] proposed to improve the existing Bing lexicon and developed a customized sentiment lexicon. They applied the proposed customized sentiment lexicon to calculate sentiment polarity from academic course feedback texts. The process involved tokenizing sentences using the bag-of-words model and determining the polarity score of each word using both Bing and a new customized lexicon. The authors showed that combining the two lexicons effectively improves both the detection of opinion words and the polarity scoring process. In fact, the quality of the lexicon-based approach is highly relative to the richness of the used dictionary. For this reason, a corpus-based approach can be adopted to enhance sentiment lexicons by incorporating prior information about words and their semantic orientation. The corpus-based approach estimates the semantic polarity of the target word by calculating the semantic distance between a word and a set of positive or negative words.

The second category of sentiment analysis approaches is based on supervised and unsupervised annotation techniques to automatically classify or predict the sentiment orientation of textual data. Several machine learning methods have been used to analyze textual student feedback. In the work of [24] student feedback was analyzed to assess course materials. The authors used the logistic regression technique to classify textual evaluations of course materials into positive, negative, or neutral. In [32] the authors propose a support-vector-machine-based learning model for the early identification of students who are likely to fail in an academic course. The proposed model gives educators the opportunity to early detect academic failure and support students to become self-regulated learners. Other machine learning techniques were also used to automatically detect sentiment orientation including multinomial logistic regression, decision tree, multi-layer perceptron, XGBoost, Gaussian Naive Bayes, and k-nearest neighbors [1, 7, 11, 15, 28]. In order to improve the accuracy of machine learning-based approaches, deep learning techniques, especially Recurrent Neural Networks (RNNs) and Convolutional Neural Networks (CNNs) have been evaluated for sentiment analysis of student feedback. These deep learning models were trained on large textual student feedback to learn the sentiment patterns. In the works of [30], the CNN model was used to evaluate teachers' effectiveness based on student feedback collected through a questionnaire including both structured and unstructured data. In the works of [26], the authors evaluated the ability of RNN to effectively detect the right word emotion based on an ensemble Long Short-Term Memory (LSTM) with attention layers. The authors showed that the use of multiple layers largely improves the result compared with conventional machine learning techniques. Other works also proposed the fusion of several deep learning models. In [3], the authors combined CNN and Bi-LSTM models to detect users' feelings. The authors showed that the fusion of CNN and BiLSTM improves sentiment detection accuracy compared with conventional machine learning techniques. In fact, all the deep learning models, including RNN, CNN and LSTM, have shown effectiveness in capturing the sequential dependencies in textual data and capturing local patterns and features in educational comments, feedback, and texts. However, these deep learning-based approaches have shown their limits in detecting contextual nuances and semantic relationships of texts.

More recent learning-based models [8,17,22,29] tried to solve the issue of contextual nuances by using pre-trained transformer models, such as Bi-directional Encoder Representations from Transformers (BERT) and Generative Pre-trained Transformer (GPT). In [8], the authors uses the BERT model to handle the complex relationships between students, teachers, and courses, as well as to tackle various challenges encountered during the learning process. The authors demonstrated the effectiveness of BERT in enhancing the analysis of student data. Similarly, in [17], the authors combine deep learning techniques with BERT to accurately detect sentiment polarity from collected comments. Their proposed model, BERT-CNN, outperforms traditional machine learning models, showcasing the power of BERT in sentiment analysis of student data. Moreover, the authors in [18] and [22] explored the use of BERT for improved contextual sentiment analysis of textual data. These studies demonstrated the effectiveness of BERT in capturing nuanced sentiment information in the context of student feedback. In addition to BERT, recent research has explored the effectiveness of GPT-based models in sentiment analysis of student feedback. In [23] and [3], the authors investigated the sentiment classification capabilities of various GPT-based models, including prompt-based and fine-tuning GPT models. The results demonstrated the superior predictive performance of GPT approaches, along with their ability to understand contextual information and detect sarcasm. This highlights the potential of GPT-based models in capturing nuanced sentiment information in the analysis of student feedback. Furthermore, a comparative study conducted by [22] further supports the effectiveness of GPT-based models in understanding and interpreting student sentiments in diverse research scenarios. In fact, Both BERT and GPT-based models have proven to be powerful tools for sentiment analysis in the context of education. They offer the advantage of leveraging pre-training on large-scale datasets, enabling them to capture contextual information and nuances in sentiment expression without the need for extensive feature engineering [10].

The third category of sentiment analysis approaches in the educational domain combines both lexicon-based and learning-based methods to leverage the strengths of each method. In [27], the authors proposed a deep learning-based opinion mining system based on a two-layered Long Short-Term Memory model: aspect extraction and sentiment polarity detection layers. The first layer predicts the aspects of student feedback while the second specifies the orientation (positive, negative, and neutral). The system is enforced using a domain embedding dictionary in both layers. Also, the authors in [2] proposed to use BERT, TextBlob, machine learning, and ensemble methods for an improved user feedback analysis. BERT and TextBlob were used for text annotations while machine learning and ensemble methods were used for sentiment classification.

Similarly in [21] and [14], the authors combined machine learning and lexicon-based approaches for sentiment analysis of students' feedback. Textual feedback was trained using a TF-IDF representation enhanced with contextual lexicon-based features. The authors showed that the combination of both approaches effectively improves results compared to conventional techniques. However, combining both approaches may be computationally expensive while requiring additional prepossessing steps, feature engineering, and model integration. Despite these limits, hybrid sentiment analysis can offer advantages by leveraging the strengths of both lexicon-based and machine learning-based approaches. A hybrid approach can provide a deeper understanding of sentiment by combining explicit and implicit rule-based heuristics.

2.3. Challenges of implementing automatic sentiment analysis in the educational domain

The automatic sentiment analysis of student comments on social networks faces several challenges that need to be addressed to effectively analyze and determine the emotional tone expressed in textual data. First, the unstructured nature of textual data makes automatic analysis of these data highly difficult and requires a pre-processing step to transform the data into a numerical format that can be easily used as inputs for automatic sentiment analysis algorithms. This task includes word extractions, removing punctuation, converting to lowercase, handling special characters, and many other text pre-processing tasks. All these tasks would be more difficult to perform on student feedback collected from social platforms given that students often use informal language, slang, abbreviations, and emojis in their social media comments. These linguistic elements add complexity to sentiment analysis as they may have connotations specific to social platforms and student interactions. It would be challenging to perform automatic sentiment analysis algorithms on such data types.

Another challenge is the ambiguity in natural languages. In fact, words can have multiple meanings depending on the context in which they are used. This ambiguity makes it difficult to accurately determine the sentiment expressed in a particular text. Contextual understanding and disambiguation techniques are necessary to overcome this challenge. Additionally, student comments on social networks may express subjective opinions and thoughts that do not clearly indicate positiveness or negativeness. Developing robust systems that can effectively capture and quantify subjective student comments is of high importance to effectively identify the right expressed sentiments. To deal with all these challenges, we propose in the next section a contextual aware sentiment analysis system that can effectively solve all the discussed problems.

3. Proposed approach

We propose in this section a new ChatGPT-based approach for the sentiment analysis of student comments in social networks. The proposed approach, referred to as Twitter Sentiment Analysis of Student Comments (TSASC), aims to automatically detect the polarity of student comments on the Twitter social platform.

TSASC combines the advantages of lexicon-based and transformer-based methods in order to build the sentiment polarity of student comments. Sentiment polarity scores are built using both TextBlob and ChatGPT through a three-phase approach as described in Figure 2. We describe in the following the different three phases and the different tasks for each phase.

3.1. Phase 1: Twitter student comments cleaning

The first phase aims to build a relevant and clean textual database of student comments. By using “Tweepy”, we collected more than 5000 comments from Twitter pages related to five Saudi high educational institutions. Most of the collected comments are in Arabic and colloquial Arabic. Only a few comments are written in English. Given that many comments contain advertisements or request academic or non-academic information and are not relevant to the sentiment analysis task, we removed all the comments containing an advertisement, containing a question mark, or written in a question form. After that, by using the “re” library, we tried to clean the student comments by removing all mentioned URLs, users, tags, dates, and numbers.

We also removed special characters, punctuation, and symbols that do not contribute in detecting the sentiment polarity. We also converted all comments to lowercase to ensure consistency and avoid case differences for English comments. However, concerning emoticons and emojis, by using the “Emojis” library, we defined a list of positive and negative emoticons and then replaced all these emoticons with the words “positive” or “negative” for English comments and with the words “ijabi” and “salbi” for Arabic and colloquial Arabic to refer to positiveness and negativeness in each comment. The other non-defined emoticons and emojis are deleted from student comments.

Figure 3 shows an example of Twitter emoticons that were used in student comments and were replaced by positive and negative words in English, Arabic, and colloquial Arabic.

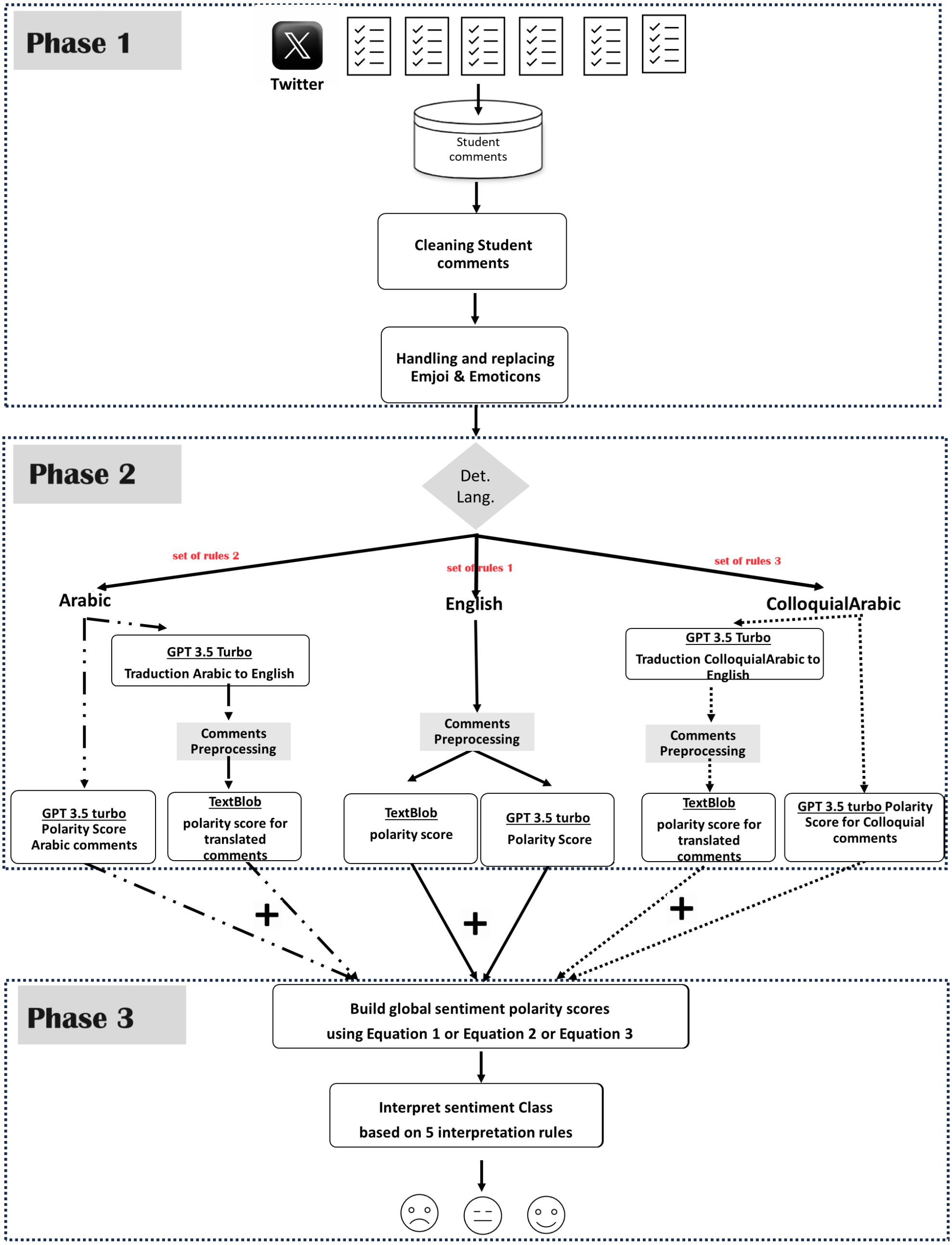

Phase 1
Twitter
Student comments
Cleaning Student comments
Handling and replacing Emjoi & Emoticons
Phase 2
Det. Lang.
set of rules 2
set of rules 1
set of rules 3
Arabic
English
ColloquialArabic
GPT 3.5 Turbo Traduction Arabic to English
Comments Preprocessing
GPT 3.5 turbo Polarity Score Arabic comments
TextBlob polarity score for translated comments
Comments Preprocessing
TextBlob polarity score
GPT 3.5 turbo Polarity Score
GPT 3.5 Turbo Traduction ColloquialArabic to English
Comments Preprocessing
TextBlob polarity score for translated comments
GPT 3.5 turbo Polarity Score for Colloquial comments
+
+
+
Phase 3
Build global sentiment polarity scores using Equation 1 or Equation 2 or Equation 3
Interpret sentiment Class based on 5 interpretation rules

Figure 2. Proposed hybrid sentiment analysis of student comments in social platforms. Sentiment polarity scores are built using both TextBlob, as a lexicon-based method, and ChatGPT as a transformer-based method

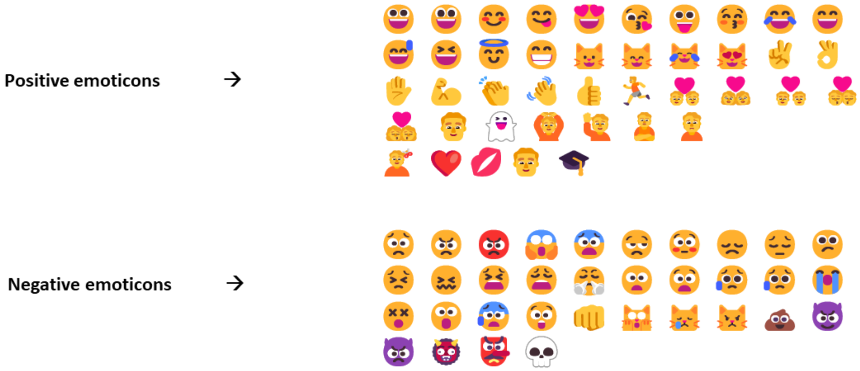


Figure 3. Examples of positive and negative Twitter emoticons. All positive emoticons are replaced by the words "positive" while negative emoticons are replaced by the words "negative"

3.2. Phase 2: building sentiment polarity and subjectivity scores

The second phase aims to build sentiment polarity and subjectivity scores of student comments using TextBlob and ChatGPT. The first, TextBlob, is a Python Natural Language Processing (NLP) library that provides simple and various NLP operations and functionalities including part-of-speech tagging, tokenization, spell-checking and sentiment analysis. It facilitates common NLP tasks and provides for developers with a convenient interface to analyze and process the textual data. The sentiment analysis function in TextBlob begins by the creation of an object by passing the text to be analyzed as a parameter. Then, the text is tokenized into a bag of words for further analysis. After that, the polarity score is calculated for each single word in the text using a sentiment lexicon (pre-defined dictionary). TextBlob utilizes its built-in sentiment lexicon using a pre-defined collection of words with associated sentiment scores. After assigning individual scores to all the words, the global sentiment for the input text is calculated by a pooling operation like taking an average of all single sentiment scores. The sentiment polarity score is normalized to a value between -1 and 1 where -1 indicates highly negative sentiment, 1 indicates highly positive sentiment, and 0 for neutral sentiment. TextBlob allows also building a subjectivity score for each comment that represents the degree of subjectivity or objectivity in a given text. The score is a numeric value between 0 and 1 where 0 indicates highly objective text and 1 indicates highly subjective. A subjectivity score close to 1 indicates that the comment may contain personal opinions or express an emotion. Figure A1 gives a small example of building sentiment polarity and subjectivity scores for the input text *"I absolutely love the teacher methodology! It's amazing!"*. TextBlob builds a positive sentiment polarity equal to 0.6875 and also a high subjectivity for the input text equal to 0.75.

The second tool used for building sentiment polarity scores is ChatGPT [33] which allows the generation of human-like responses based on a given user request, also called a user prompt. This tool has been trained on diverse huge textual data and supports several human languages including several colloquial languages. We explored the ChatGPT natural language processing capabilities by using the recently proposed OpenAI API and based on GPT-3.5-turbo learning model. The OpenAI GPT-3.5-turbo learning model has been trained to understand human natural language and allows to provide text outputs in response to a user prompt. The design of a user prompt is similar to designing instructions in a computer program describing how to successfully complete a task. Several tasks can be defined including text analyzer, question answer about a knowledge base, translating text, or interpreting the sentiments in a given text.

To build sentiment polarity and subjectivity scores for a given input text using the GPT-3.5-turbo model through the OpenAI API, we designed two functions, “analyze-sentiment-polarity” and “analyze-subjectivity”, and requested the “Chat Completions API” to build responses containing the calculated scores. We give in the following a description of the two designed functions:

```
def analyze_sentiment_polarity(sentence):
    response = client.chat.completions.create(
      model="gpt-3.5-turbo",
      messages=[
        {
          "role": "system",
          "content": "You will be provided with a given textual
          ↪   sentence in the context of education, and your task is
          ↪   to calculate a sentiment score between -1 and 1."

        },
        {
          "role": "user",
          "content": sentence
        }
      ],
      temperature=0.2,
      max_tokens=10,
      top_p=1
    )

    sentiment_score = response.choices[0].message.content
    return sentiment_score


```

```
def analyze_subjectivity(sentence):
    response = client.chat.completions.create(
      model="gpt-3.5-turbo",
      messages=[
        {
          "role": "system",
          "content": "You will be provided with a given textual
          ↪  sentence in the context of education, and your task is
          ↪  to calculate a subjectivity score between 0 and 1"
        },
        {
          "role": "user",
          "content": sentence
        }
      ],
      temperature=0.2,
      max_tokens=10,
      top_p=1
    )

    subjectivity_score = response.choices[0].message.content

    return subjectivity_score
```

For the first function, “analyze-sentiment-polarity,” we utilize the OpenAI *client.chat.completions.create()* function with various parameters. We start by specifying the GPT-3.5-turbo model as our learning model. In the design of the message parameter, we define the system and user roles. In the system's role, we request the model to generate a sentiment score between -1 and 1, representing the sentiment polarity of a given input textual sentence. We explicitly mention the context by using the term “education” to provide contextual information. By incorporating the intended context, the fine-tuned GPT-3.5-turbo model can generate more accurate sentiment scores that are contextually aware. Regarding the user's role, we indicate that the user will provide an input textual sentence for sentiment analysis. Additionally, we set the *temperature* parameter to a low value of 0.2 to ensure more deterministic and stable sentiment scores. The *max_token* parameter is set to 10, allowing us to effectively receive and store the score results. Finally, we set $top\_p = 1$, which considers all possible tokens without the need for sampling. These same parameters are also applied to the second function, “analyze-subjectivity”, which we designed to calculate a subjectivity score using the GPT-3.5-turbo model. In this case, the system role is responsible for generating a subjectivity score between 0 and 1 with an wxlocit indication of the “education” context. For a complete description and call of the defined functions through the OpenAI API please refer to Appendix 2.

To provide readers with a clearer understanding of the sentiment polarity and subjectivity scores obtained using TextBlob and ChatGPT, Table 1 presents the results for different sample English messages. These scores were generated using TextBlob and ChatGPT tools and reveal variations in sentiment and subjectivity assessments. We show that TextBlob tends to assign slightly lower sentiment scores compared to ChatGPT. This difference suggests that TextBlob may be more conservative in its sentiment analysis, potentially leading to a more neutral sentiment classification for some messages. On the other hand, ChatGPT often exhibits lower subjectivity scores, indicating a more objective interpretation of the given statements. It is also important to note that the reported results are specific to the English language given that TextBlob is limited to English comments and cannot be applied to Arabic or colloquial Arabic comments.

Table 1

Sentiment analysis results

#	Message	TextBlob	ChatGPT

1	"the teacher gives a good textbook for the operating system course"	sentiment = 0.7
		subjectivity = 0.6
			sentiment = 0.7
			subjectivity = 0.3

2	"I'm not sure how I feel about exams"	sentiment = –0.25
		subjectivity = 0.88
			sentiment = –0.4
			subjectivity = 0.7

3	"Do you get nervous before an exam?"	sentiment = 0
		subjectivity = 0
			sentiment = –0.3
			subjectivity = 0.7

4	"This was a helpful example but I would prefer another one"	sentiment = 0
		subjectivity = 0
			sentiment = 0.2
			subjectivity = 0.8

5	"I absolutely love the teacher methodology! It's amazing!"	sentiment = 0.68
		subjectivity = 0.75
			sentiment = 0.9
			subjectivity = 0.1

To leverage the strengths of both tools and create a more effective sentiment analysis score, we propose to combine scores from TextBlob and ChatGPT to take advantage of the nuanced analysis provided by each tool. Respecting to the comment detected language, polarity and subjectivity scores of student comments are calculated based on a set of rules and tasks as described in the second phase of Figure 2. We give in the following a detailed description of these tasks and rules:

- **Set of rules 1** (detected language is English). A pre-processing of English comments is performed by removing stop words that do not contribute in determining the sentiment polarity such as "from", "this", "on", "was", etc. We used a list of English stop words available in the Python Natural Language Toolkit [4]. Then, a stemming task is performed to reduce words in each comment to their root forms. This can help in consolidating similar words and reducing vocabulary size when using TextBlob. After that, the pre-processed comments are used as inputs for both TextBlob and ChatGPT. For each comment EN_i, we generated four scores: $P_{Blob}(EN_i)$ and $S_{Blob}(EN_i)$ that represents sentiment polarity and subjectivity scores calculated using TextBlob and $P_{GPT}(EN_i)$ and $S_{GPT}(EN_i)$ that represents sentiment polarity and subjectivity score calculated using ChatGPT 3.5 turbo. For ChatGPT, we added a contextual indication "education" in the textual request of building a sentiment score between -1 and 1 and also in detecting the sentiment subjectivity of the whole comment. The contextual indication allows the pre-trained ChatGPT model to give a more accurate sentiment analysis of student comments.

- **Set of rules 2** (detected language is Arabic). Concerning sentiment scores built by ChatGPT, Arabic comments are given as inputs without any text pre-processing tasks. ChatGPT is pre-trained to analyze such data and build an effective sentiment polarity score. However, a contextual indication "م" that refers to education is added to improve the contextual analysis when building the sentiment polarity $P_{GPT}(AR_i)$ and the subjectivity $S_{GPT}(AR_i)$ scores. On the other hand, to build sentiment scores using TextBlob, we translated Arabic comments to English using ChatGPT, then pre-processed the output text by removing stop words and finally the sentiment polarity $P_{Blob}(AR_i)$ and the subjectivity scores $S_{Blob}(AR_i)$ are built using TextBlob sentiment function.
- **Set of rules 3** (detected language is colloquial Arabic). Similarly to Arabic comments, colloquial Arabic comments are given as inputs to ChatGPT without a pre-processing task. Only a contextual indication "م" is added to the ChatGPT request to refer to comments relative to the education domain for building the sentiment polarity $P_{GPT}(CA_i)$ and the subjectivity $S_{GPT}(CA_i)$ scores. However, colloquial Arabic comments are translated to English using ChatGPT and then pre-processed by removing stop words and given as inputs to TextBlob to generate the sentiment polarity $P_{Blob}(CA_i)$ and the subjectivity $S_{Blob}(CA_i)$ scores.

3.3. Phase 3: building global sentiment polarity score and interpreting sentiment classes

This phase aims to assign a sentiment class for each student comment based on polarity scores built in the previous step. In the first step, building global sentiment polarity scores, a global sentiment score is calculated for each student comment based on a combination of polarity scores obtained using TextBlob and ChatGPT. To build a global score for each comment EN_i, $AR_i$and CA_i, we defined a combination function that uses the subjectivity scores as weights for calculating global sentiment polarity scores in the overall function as follows:

- **English comments**: for each English comment EN_i, given the two sentiment polarity scores $P_{Blob}(EN_i)$ and $P_{GPT}(EN_i)$ and the two subjectivity scores $S_{Blob}(EN_i)$ and $S_{GPT}(EN_i)$ built using TextBlob and ChatGPT, the global sentiment score $P(EN_i)$ is calculated by:

$$P(EN_i) = \frac{(S_{GPT}(EN_i) \cdot P_{GPT}(EN_i)) + (S_{Blob}(EN_i) \cdot P_{Blob}(EN_i))}{S_{GPT}(EN_i) + S_{Blob}(EN_i)} \tag{1}$$

- **Arabic comments**: for each arabic comment AR_i, given the two sentiment polarity scores $P_{Blob}(AR_i)$ and $P_{GPT}(AR_i)$ and the two subjectivity scores $S_{Blob}(AR_i)$ and $S_{GPT}(AR_i)$ calculated using TextBlob and ChatGPT, the global sentiment score for each arabic comment AR_i is calculated by:

$$P(AR_i) = \frac{(S_{GPT}(AR_i) \cdot P_{GPT}(AR_i) + S_{Blob}(AR_i) \cdot P_{Blob}(AR_i))}{S_{GPT}(AR_i) + S_{Blob}(AR_i)} \tag{2}$$

- **Colloquial Arabic comments**: for each colloquial arabic comment CA_i, given the two sentiment polarity scores $P_{Blob}(CA_i)$ and $P_{GPT}(CA_i)$ and the two subjectivity scores $S_{Blob}(CA_i)$ and $S_{GPT}(CA_i)$ calculated using TextBlob and ChatGPT, the global sentiment score for each colloquial arabic comment $P(CA_i)$ is calculated by:

$$P(CA_i) = \frac{(S_{GPT}(CA_i) \cdot P_{GPT}(CA_i) + S_{Blob}(CA_i) \cdot P_{Blob}(CA_i))}{S_{GPT}(CA_i) + S_{Blob}(CA_i)} \tag{3}$$

We notice in Equation (1), (2) and (3) that a bigger weight is assigned for ChatGPT scores compared to those built using TextBlob. This choice is based on our study of small examples of comments that showed better performance of ChatGPT compared to TextBlob specifically for Arabic and colloquial Arabic comments. One can adjust these weights by assigning similar weights or giving more importance to any selected tool.

Once global polarity scores for student comments are calculated, the next step aims to interpret the sentiment classes based on the following defined five interpretation rules:

1. **Interpretation rule 1** (*if the global polarity score is above a threshold* 0.2*:*) Classify the comment as positive
2. **Interpretation rule 2** (*if the global polarity score is under a threshold of* -0.2*:*) classify the comment as negative
3. **Interpretation rule 3** (*if the global polarity score of a comment is in the interval* $[-0.2, -0.1]$*:*) if the calculated subjectivity of TextBlob (S_{Blob}) and ChatGPT (S_{GPT}) are both above -0.2 and both calculated polarities are negative then classify the comment as negative else classify the comment as neutral.
4. **Interpretation rule 4** (*if the global polarity score of a comment is in the interval* $[0.1, 0.2]$*:*) if calculated subjectivity of TextBlob (S_{Blob}) and ChatGPT (S_{Blob}) are both above 0.2 and both calculated polarities for the comment are positive then classify the comment as positive else classify the comment as neutral.
5. **Interpretation rule 5** (*if the global polarity score is in the interval* $[-0.1, 0.1]$ *(within a range around* 0*):*) classify the comment as neutral.

We considered for all these rules a subjectivity threshold equal to 0.2 and a polarity threshold equal to 0.1. These thresholds are fixed based on an experimental study of subjectivity and polarity scores built for all the comments. One can adjust these thresholds based on the specificity of the application or the specificity of the studied data.

4. Experiments and empirical evaluations

4.1. Data set description

In order to evaluate the quality of the proposed sentiment analysis approach, we manually annotated an ensemble of 1266 student comments that were extracted from Twitter pages of five Saudi higher colleges on the Twitter platform posted between April 2022 and June 2023. The annotated dataset contains 1266 comments including 134 in English, 460 in Arabic and 672 in colloquial Arabic. The selected comments were tagged by a human annotator with sentiment labels including positive, negative, and neutral categories for each language. The statistics for annotated comments are described in Table 2. The final distribution of the sentiment labels in the overall dataset is as follows: 607 are labeled as neutral, 355 are labeled as positive, and 304 are labeled as negative.

Table 2

Statistics of the build dataset from five university pages on the Twitter platform

Dataset	Language	Classes
Twitter student comments dataset (1266)	English (134)	Positive (39) Negative (48) Neutral (47)
	Arabic (460)	Positive (108) Negative (117) Neutral (235)
	Colloquial Arabic (672)	Positive (188) Negative (139) Neutral (325)

As described in Section 3.1, several pre-processing steps were applied to enhance the quality of the extracted textual comments. Irrelevant comments including advertisements were filtered out. English comments were transformed into lowercase, tokenized, and split into individual words. Also, stop words, punctuation marks, and special characters were removed. The resulting pre-processed dataset is used as input for the sentiment classification methods.

To better explore the built student comments dataset, we visualized the word cloud of English, Arabic, and colloquial Arabic comments. The word clouds provide an overview of the keywords and topics discussed by students giving insights into their sentiments, concerns, and experiences related to education and university life. Word clouds are generated by analyzing the frequency of words in the comments and representing them graphically such that the size of each word indicates its relative occurrence. The most frequently used words appear larger and bolder while less frequent words are visualized smaller. For English comments, we colored positive and negative words in green and red based on the TextBlob dictionary. The coloring cannot be performed for Arabic and colloquial Arabic given that TextBlob only gives polarity of English terms. Figure 4 shows the most used English terms in the built dataset including "course", "teacher", "instructor", "material", etc. This figure also shows a few detected positive words (i.e. "easy", "much" and "challenging") and a few detected negative words (i.e. "unfair", "difficult" and "minimal"). We notice that all the colored terms have been well classified whether for positive or negative polarities.

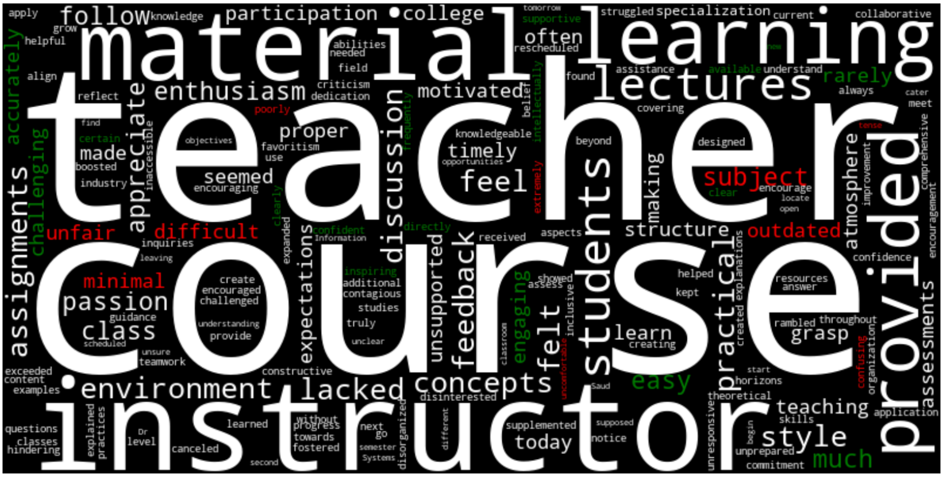


Figure 4. Wordcloud of extracted student comments in English. Positive and negative words are schematized in green and red colors based on the TextBlob dictionary

Although the high precision of classifying positive and negative terms, Figure 4 shows the limit of using TextBlob for polarity detection since several positive and negative words are not detected. For example, evident words that express positiveness such as "timely" and "motivated" and also words that express negativeness such as "inaccessible", "unsupported" and "disinterested" are not detected and classified by TextBlob. This shows the limit of using lexicon-based algorithms, based on TextBlob dictionary, for detecting the sentiment polarity of student comments. Concerning Arabic and colloquial Arabic comments, Figure 5 and Figure 6 show the most frequent Arabic and colloquial Arabic words. By examining the prominent words in the two figures, we notice the existence of several Arabic words expressing positiveness and negativeness related to the educational field such as م and " و " that refer to "professor" and "training" respectively. These two figures also show the effectiveness of ChatGPT in separating Arabic from colloquial Arabic comments. Figure 6 visualizes several words from the Arabic colloquial language such as "bidina = أ" and " yji = آ" as are pronounced in Arabic to refer to "we can do something" and "possible".

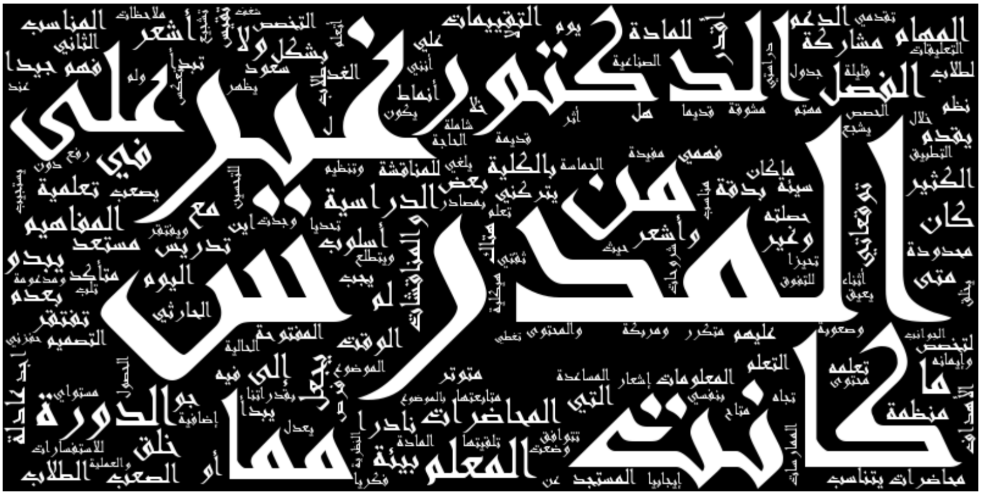


Figure 5. Wordcloud of extracted student comments in Arabic

Figure 6. Wordcloud of extracted student comments in colloquial Arabic

4.2. Experimental results and discussions

In order to evaluate the performance of the proposed approach compared with existing methods, we used four evaluation measures: *Precision*, *Accuracy*, *Recall*, and *F1-score*. These measures are used to evaluate whether the predictions of sentiment classes are correct with respect to the underlying correct sentiment labels.

Precision measures the proportion of correctly predicted comments for a specific sentiment class over the total comments predicted as that class, and is calculated using the formula:

$$Precision = \frac{TP}{TP + FP} \tag{4}$$

where TP denotes true positives (correctly predicted instances) and FP represents false positives (incorrectly predicted instances). Accuracy evaluates the overall correctness of sentiment class predictions across all classes, and is defined as:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{5}$$

where TN denotes true negatives (correctly predicted instances of sentiment classes other than the evaluated class) and FN represents false negatives (incorrectly predicted instances of the evaluated class). Concerning the Recall measure, it evaluates the proportion of correctly predicted instances of a specific sentiment class over the total comments that belong to that class, and is calculated by:

$$Recall = \frac{TP}{TP + FN} \tag{6}$$

Finally, *F1-score* combines precision and recall to provide a single metric that balances both measures, and is computed using the formula:

$$F1\text{-}score = \frac{2 \cdot Precision \cdot Recall}{Precision + Recall} \tag{7}$$

All these evaluation measures are computed individually for each sentiment class (positive, negative, neutral), and then an overall weighted score is generated for the entire dataset, considering the importance of each sentiment class.

We compared the effectiveness of the proposed sentiment analysis approach with several existing methods including lexicon-based, machine learning, deep learning, and transformer-based methods. Concerning machine learning methods, we evaluated the effectiveness of four machine learning methods which are Random Forest (RF), Support Vector Machine (SVM) using both linear and polynomial kernels and K-Nearest Neighbors (KNN). For all machine learning methods, features are extracted using the bag-of-words model while frequencies are calculated by using the Term Frequency-Inverse Document matrix (TF-IDF) technique. Evaluation measures are built by calculating the average of five-fold cross-validation. For each loop, a split of data into training and testing, with sizes 60% and 40%, is performed. Concerning deep learning methods, we evaluated two recurrent neural network methods which are LSTM (Long Short-Term Memory) and GRU (Gated Recurrent Unit). Finally, for transformer-based methods, we assessed the performance of both BERT$_{base}$ and RoBERTa which is an optimized version of the BERT pre-trained model.

Table 3 presents obtained performance measures including *Accuracy* (Acc.), Precision (Prec.), Recall (Rec.), and F1-score (F1) for each sentiment class (Positive, Negative, and Neutral), as well as an overall average over the three classes computed for each evaluation measure. This table shows that the proposed approach, TSASC, achieved the highest overall scores evaluated by 0.81, 0.86, 0.82 and 0.85 for accuracy, precision, recall, and F1-score respectively. All these obtained scores largely exceed all the other built scores in Table 3. For the Machine Learning category, we show that all evaluated methods do not exceed an overall F1-score of 0.43 indicating the low sentiment analysis performance of machine learning methods in identifying the true sentiment classes of student comments. We show that SVM method with a linear kernel gives the lowest result in predicting the sentiment of student comments with an overall accuracy of 36% while RF method showed a slight improvement compared to the other machine learning methods with an overall accuracy of 0.45%. For the deep learning category, we show that RNN with a Long Short-Term Memory (LSTM) architecture achieves an overall F1-score of 0.56, while RNN with a Gated Recurrent Unit (GRU) architecture achieves an overall F1-score of 0.59. These results indicate that Deep Learning models, based on both LSTM and GRU architectures, outperform the Machine Learning methods in detecting sentiment classes of student comments. In the Lexicon-based approaches, both TextBlob and Vader achieve nearly similar overall performances of around 0.65. This suggests that these lexicon-based methods performed better than both machine learning and deep learning approaches. Although the dictionary of TextBlob and Vader does not include an exhaustive list of words with positive and negative polarities, the lexicon-based approach has achieved better performance compared to machine learning and deep learning approaches in detecting sentiment classes of student comments. Similarly, transformer-based methods, including BERT_{base} and RoBERTa have shown good results compared to machine and deep learning approaches. For instance, the overall F1 scores attained by BERT*base* and RoBERTa were 0.71 and 0.75 respectively. Nevertheless, these two methods have demonstrated limitations in accurately detecting neutral student comments. Specifically, the RoBERTa method obtained a precision, recall, and F1 score of 0.52, 0.55, and 0.52 respectively.

Table 3

Comparison of the effectiveness of the proposed TSASC approach compared to conventional sentiment analysis methods

	Method	Class	Performance measures			
			Acc.	Prec.	Rec.	F1
Lexicon-based	TextBlob	Positive	0.65	0.70	0.62	0.65
		Negative	0.64	0.68	0.61	0.63
		Neutral	0.76	0.79	0.73	0.75
		Overall	0.70	0.73	0.69	0.71
	Vader	Positive	0.65	0.70	0.62	0.64
		Negative	0.64	0.71	0.61	0.63
		Neutral	0.66	0.69	0.63	0.65
		Overall	0.65	0.70	0.62	0.64

Table 3 cont.

	Method	Class	Performance measures			
			Acc.	Prec.	Rec.	F1
Machine learning	RF	Positive	0.45	0.48	0.42	0.44
		Negative	0.44	0.49	0.41	0.43
		Neutral	0.46	0.47	0.43	0.44
		Overall	0.45	0.47	0.42	0.43
	SVM (Linear Kernel)	Positive	0.39	0.31	0.36	0.32
		Negative	0.38	0.32	0.35	0.33
		Neutral	0.40	0.40	0.37	0.38
		Overall	0.39	0.37	0.36	0.36
	SVM (Polynomial Kernel)	Positive	0.41	0.43	0.49	0.45
		Negative	0.40	0.44	0.48	0.45
		Neutral	0.42	0.42	0.40	0.40
		Overall	0.41	0.43	0.42	0.41
	KNN	Positive	0.42	0.45	0.40	0.41
		Negative	0.41	0.46	0.49	0.47
		Neutral	0.43	0.44	0.41	0.42
		Overall	0.42	0.44	0.43	0.43
Deep learning	RNN (LSTM)	Positive	0.55	0.58	0.52	0.54
		Negative	0.54	0.59	0.53	0.55
		Neutral	0.56	0.57	0.53	0.54
		Overall	0.56	0.58	0.52	0,54
	RNN (GRU)	Positive	0.59	0.51	0.56	0.54
		Negative	0.58	0.52	0.55	0.53
		Neutral	0.60	0.61	0.61	0.61
		Overall	0.59	0.59	0.59	0.59
Transformer-based	BERT$_{base}$	Positive	0.75	0.77	0.71	0.72
		Negative	0.69	0.70	0.76	0.72
		Neutral	0.46	0.48	0.49	0.48
		Overall	0.70	0.73	0.69	0.71
Transformer-based	RoBERTa	Positive	0.79	0.78	0.79	0.79
		Negative	0.80	0.81	0.79	0.80
		Neutral	0.51	0.52	0.55	0.52
		Overall	0.72	0.71	0.72	0.75
Proposed approach	**TSASC**	Positive	0.85	0.80	0.82	0.81
		Negative	0.84	0.82	0.91	0.85
		Neutral	0.86	0.89	0.73	0.79
		Overall	0.85	0.86	0.82	0.81

Obtained results in Table 3 show that our TSASC approach has significantly outperformed all other conventional methods including machine learning, deep learning, as well as transformer-based and lexicon-based approaches. This finding shows the effectiveness of the proposed approach in accurately identifying sentiment classes on the student comments dataset and also shows the effectiveness of leveraging ChatGPT capability for sentiment classification. Reported results also show an interesting performance of lexicon-based and transformer based approaches compared to deep learning and machine learning approaches.

In order to effectively evaluate the performance of ChatGPT in giving better contextual-improved sentiment analysis of student comments, we evaluated its effectiveness compared to Google translation. Concerning our approach, rather than directly detecting the sentiment polarity of Arabic and colloquial Arabic comments using ChatGPT, we translated all these comments into English language using Google Translate and then detected sentiment polarity by using TextBlob. Table 4 presents the results of the comparison of ChatGPT and Google Translation and their impacts on the sentiment analysis of student comments by using machine learning, deep learning, and lexicon-based approaches.

Table 4

Comparison of the impacts of ChatGPT and Google Translator on the sentiment analysis of student comments using machine learning, deep learning, lexicon-based, and the proposed approach. ChatGPT and Google Translator are used as automatic tools to translate Arabic and colloquial Arabic comments into English

	Method	Google Translate		ChatGPT translation service	
		Acc.	F1	Acc.	F1
Lexicon-based	TextBlob	0.58	0.59	0.69	0.71
	Vader	0.54	0.56	0.65	0.64
Machine learning	RF	0.37	0.39	0.45	0.43
	SVM (linear kernel)	0.34	0.36	0.39	0.36
	SVM (polynomial kernel)	0.40	0.41	0.41	0.41
	KNN	0.40	0.40	0.42	0.43
Deep learning	RNN(LSTM)	0.55	0.57	0.56	0.58
	RNN(GRU)	0.56	0.57	0.59	0.59
Transformer-based	BERT_{base}	0.65	0.67	0.70	0.71
	RoBERTa	0.69	0.71	0.72	0.75
Proposed approach	TSASC	0.67	0.70	0.85	0.81

The performance of each method is evaluated using an overall score of *Accuracy* (Acc.) and *F1-score* (F1) measures. For the machine learning and deep learning category, we do not show a high impact on obtained results. For example, the Random Forest method achieves an accuracy and *F1-score* of around 0.40 by using ChatGPT and Google translation. In fact, for machine learning and deep learning category, the computational framework does not take into account the semantics of words and only focuses on the similarity of words between student comments to build sentiment classes. This fact explains the nearly same results obtained by ChatGPT and Google Translator for all machine learning and deep learning methods. Furthermore, the effectiveness of transformer-based methods remains consistent across different translation services. For instance, the RoBERTa method demonstrated comparable accuracy and F1 scores by using Google Translate and ChatGPT translation services. When employing Google Translate, the achieved scores were 0.69 and 0.71, while utilizing the ChatGPT translation service yielded scores of 0.72 and 0.75. Despite the superior performance of transformer-based methods compared to machine and deep learning approaches, no discernible impact was observed concerning the choice of translation service on the final results. However, when using a lexicon-based approach, Table 4 shows a high improvement of results by using ChatGPT compared to Google translator for both TextBlob and Vader. For example, when using ChatGPT TextBlob achieves an accuracy of 0.69 and an *F1-score* of 0.71 compared to an accuracy of 0.65 and an *F1-score* of 0.64 when using Google Translator. Furthermore, transformer based methods also show high performance compared to machine In the same way, we show a high improvement when using ChatGPT for our proposed approach. ChatGPT with TSASC achieves an accuracy of 0.85 and an *F1-score* of 0.81 compared to 0.67 and 0.7 by using Google Translate. This obtained result shows the effectiveness of directly detecting sentiment polarity of Arabic and colloquial Arabic comments rather than considering a translating step of the comment into English. This obtained result also confirms the contextual improved capability of the proposed approach compared to other approaches and its ability to detect more accurate sentiment polarity of student comments through using ChatGPT capability.

4.3. Ethical considerations of student sentiment analysis

The sentiment analysis of student data collected from their activities on social networks has gained significant attention leading to the development of various tools, packages, and methods that enable effective and automatic analysis of these data. However, the collection and analysis of such data has lead to crucial ethical considerations that require careful attention. One primary ethical concern is the privacy protection. The collected student data may include personal and sensitive information pertaining to their daily lives. It is crucial that sentiment analysis tools incorporate pre-processing steps to anonymize data and remove any personally identifiable information to prevent potential privacy breaches. It is worth noting that available tools may be limited to accessing only public student information obtained from public university pages.

Another ethical consideration concerns the non-disclosure of student perceptions, experiences, or opinions regarding subjects, courses, or teachers. This information should only be shared with individuals who are responsible for improving the student experience. It is essential to safeguard this information from other parties looking to exploit it for commercial purposes. Individuals involved in such projects may sign a consent form affirming non-disclosure of the results for any commercial use. Furthermore, individuals working on such projects should seek approval from relevant ethics committees or institutional review boards to ensure compliance with ethical guidelines and regulations.

This leads to a more generic ethical consideration that concerns the security of student data. Student data must be protected against unauthorized access, breaches, or misuse. Employing encryption techniques, secure data storage systems, and implementing restricted access controls are essential measures to ensure the integrity and confidentiality of the data.

5. Conclusion

We proposed a novel approach for sentiment analysis of student comments on the Twitter platform. Our proposed approach addresses the challenges of language diversity, the presence of informal language, and the presence of emoticons, and supports three languages: English, Arabic, and colloquial Arabic. Based on ChatGPT's capabilities of addressing language nuances, we incorporated a new scoring method to build sentiment classes based on both polarity and subjectivity. Experiments conducted on real student comments dataset demonstrated the effectiveness of our proposed approach compared to conventional sentiment analysis methods.

Although the proposed approach is described and evaluated on the Twitter social platform, it can be generalized to the analysis of student comments on other platforms such as Facebook and LinkedIn. Another area of improvement could be the expansion of language support beyond English, Arabic, and colloquial Arabic. As the proposed approach is based on ChatGPT to address language nuances, the integration of additional languages is still possible. Moreover, it would be valuable to develop a user-friendly sentiment analysis toolkit or API based on the proposed approach and make it accessible to researchers, educators, and educational institutions. Further improvements of the presented work can also explore the effectiveness of recent proposed ChatGPt models, such as GPT4o and GPT4-turbo, in building sentiment scores for student comments. Conducting experimental evaluations that compare the performance and capabilities of these available ChatGPT models would provide valuable insights into their respective strengths and weaknesses in generating accurate sentiment scores in educational contexts.

Acknowledgements

This work was funded by the University of Jeddah, Jeddah, Saudi Arabia, under grant No. (UJ-21-DR-5). Therefore, the authors thank the University of Jeddah for its technical and financial support.

References

[1] Alencia Hariyani C., Nizar Hidayanto A., Fitriah N., Abidin Z., Wati T.: Mining Student Feedback to Improve the Quality of Higher Education through Multi Label Classification, Sentiment Analysis, and Trend Topic. In: *2019 4th International Conference on Information Technology, Information Systems and Electrical Engineering (ICITISEE)*, pp. 359–364, 2019. doi: 10.1109/ICITISEE48480.2019.9003818.

[2] Baker M.R., Taher Y.N., Jihad K.H.: Prediction of People Sentiments on Twitter Using Machine Learning Classifiers during Russian Aggression in Ukraine, *Jordanian Journal of Computers and Information Technology (JJCIT)*, vol. 09(03), pp. 189–206, 2023. doi: 10.5455/jjcit.71-1676205770.

[3] Baker M.R., Utku A.: Unraveling user perceptions and biases: A comparative study of ML and DL models for exploring twitter sentiments towards ChatGPT, *Journal of Engineering Research*, vol. 13(2), pp. 1658–1665, 2025. doi: 10.1016/j.jer.2023.11.023.

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

[5] Chauhan A., Mohana R.: Implementing LDA Topic Modelling Technique to Study User Reviews in Tourism. In: *2022 Seventh International Conference on Parallel, Distributed and Grid Computing (PDGC)*, pp. 357–360, 2022. doi: 10.1109/PDGC56933.2022.10053153.

[6] Chirumamilla S., Gulati M.: Patient Education and Engagement through Social Media, *Current Cardiology Reviews*, vol. 17(2), pp. 137–143, 2021. doi: 10.2174/1573403X15666191120115107.

[7] Dhanalakshmi V., Bino D., Saravanan A.M.: Opinion mining from student feedback data using supervised learning algorithms. In: *2016 3rd MEC International Conference on Big Data and Smart City (ICBDSC)*, 2016. doi: 10.1109/ICBDSC.2016.7460390.

[8] Dyulicheva Y., Bilashova E.: Learning Analytics of MOOCs based on Natural Language Processing. In: *CS&SE@SW 2021: 4th Workshop for Young Scientists in Computer Science & Software Engineering, December 18, 2021, Kryvyi Rih, Ukraine*, pp. 187–197, 2021. https://ceur-ws.org/Vol-3077/paper15.pdf.

[9] Ebrahimi P., Basirat M., Yousefi A., Nekmahmud M., Gholampour A., Fekete-Farkas M.: Social Networks Marketing and Consumer Purchase Behavior: The Combination of SEM and Unsupervised Machine Learning Approaches, *Big Data and Cognitive Computing*, vol. 6(2), 35, 2022. doi: 10.3390/bdcc6020035.

[10] Hao Z., Cheah Y.N., Alyasiri O.M., An J.: Exploring aspect-based sentiment quadruple extraction with implicit aspects, opinions, and ChatGPT: a comprehensive survey, *Artificial Intelligence Review*, vol. 57, 2024. doi: 10.1007/s10462-023-10633-x.

[11] Hew K.F., Hu X., Qiao C., Tang Y.: What predicts student satisfaction with MOOCs: A gradient boosting trees supervised machine learning and sentiment analysis approach, *Computers & Education*, vol. 145, 103724, 2020. doi: 10.1016/j.compedu.2019.103724.

[12] Jasim Y., Saeed M., Raewf M.B.: Analyzing Social Media Sentiment: Twitter as a Case Study, *ADCAIJ: Advances in Distributed Computing and Artificial Intelligence Journal*, vol. 11, pp. 427–450, 2023. doi: 10.14201/adcaij.28394.

[13] 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, *Applied Sciences*, vol. 11(9), 2021. doi: 10.3390/app11093986.

[14] Kaur W., Balakrishnan V., Singh B.: Social media sentiment analysis of thermal engineering students for continuous quality improvement in engineering education, *Journal of Mechanical Engineering*, vol. 1, pp. 263–272, 2017.

[15] Kavitha R.: Sentiment Research on Student Feedback to Improve Experiences in Blended Learning Environments, *International Journal of Innovative Technology and Exploring Engineering*, pp. 159–163, 2019. doi: 10.35940/ijitee.K1034.09811S19.

[16] Laranjo L., Arguel A., Neves A.L., Gallagher A.M., Kaplan R., Mortimer N.: The influence of social networking sites on health behavior change: a systematic review and meta-analysis, *Journal of the American Medical Informatics Association*, vol. 22(1), pp. 243–256, 2015. doi: 10.1136/amiajnl-2014-002841.

[17] Li X., Zhang H., Ouyang Y., Zhang X., Rong W.: A Shallow BERT-CNN Model for Sentiment Analysis on MOOCs Comments. In: *2019 IEEE International Conference on Engineering, Technology and Education (TALE)*, 2019. doi: 10.1109/TALE48000.2019.9225993.

[18] Liu Y., Ott M., Goyal N., Du J., Joshi M., Chen D., Levy O., *et al.*: RoBERTa: A Robustly Optimized BERT Pretraining Approach, *ArXiv*, vol. abs/1907.11692, 2019. doi: 10.48550/arXiv.1907.11692.

[19] Manzoor U., Baig S.A., Hashim M., Sami A.: Impact of Social Media Marketing on Consumer's Purchase Intentions: The Mediating Role of Customer Trust, *International Journal of Entrepreneurial Research*, vol. 3(2), pp. 41–48, 2020. doi: 10.31580/ijer.v3i2.1386.

[20] Misuraca M., Scepi G., Spano M.: Using Opinion Mining as an educational analytic: An integrated strategy for the analysis of students' feedback, *Studies in Educational Evaluation*, vol. 68, 100979, 2021. doi: 10.1016/j.stueduc.2021.100979.

[21] Nasim Z., Rajput Q., Haider S.: Sentiment analysis of student feedback using machine learning and lexicon based approaches. In: *2017 International Conference on Research and Innovation in Information Systems (ICRIIS)*, 2017. doi: 10.1109/ICRIIS.2017.8002475.

[22] Nguyen D.Q., Vu T., Nguyen A.T.: BERTweet: A pre-trained language model for English Tweets. In: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pp. 9–14, 2020. doi: 10.18653/v1/2020.emnlp-demos.2.

[23] Obinwanne T., Brandtner P.: Enhancing Sentiment Analysis with GPT – A Comparison of Large Language Models and Traditional Machine Learning Techniques. In: A.K. Nagar, D.S. Jat, D. Mishra, A. Joshi (eds.), *Intelligent Sustainable Systems. Selected papers of WorldS4 2023, Volume 3*, Lecture Notes in Networks and Systems, vol. 803, pp. 187–197, Springer Nature, Singapore, 2024. doi: 10.1007/978-981-99-7569-3_17.

[24] Osmanoğlu U.Ö., Atak O.N., Çağlar K., Kayhan H., Can T.: Sentiment Analysis for Distance Education Course Materials: A Machine Learning Approach, *Journal of Educational Technology and Online Learning*, vol. 3, pp. 31–48, 2020. doi: 10.31681/jetol.663733.

[25] Rajput Q., Haider S., Ghani S.: Lexicon-Based Sentiment Analysis of Teachers Evaluation, *Applied Computational Intelligence and Soft Computing*, vol. 2016(1), 2385429, 2016. doi: https://doi.org/10.1155/2016/2385429.

[26] Sangeetha K., Prabha D.: RETRACTED ARTICLE: Sentiment analysis of student feedback using multi-head attention fusion model of word and context embedding for LSTM, *Journal of Ambient Intelligence and Humanized Computing*, vol. 12, pp. 4117–4126, 2020. doi: 10.1007/s12652-020-01791-9.

[27] Sindhu I., Muhammad Daudpota S., Badar K., Bakhtyar M., Baber J., Nurunnabi M.: Aspect-Based Opinion Mining on Student's Feedback for Faculty Teaching Performance Evaluation, *IEEE Access*, vol. 7, pp. 108729–108741, 2019. doi: 10.1109/ACCESS.2019.2928872.

[28] Srinivas S., Rajendran S.: Topic-based knowledge mining of online student reviews for strategic planning in universities, *Computers and Industrial Engineering*, vol. 128, pp. 974–984, 2019. doi: 10.1016/j.cie.2018.06.034.

[29] Susnjak T.: *Applying BERT and ChatGPT for Sentiment Analysis of Lyme Disease in Scientific Literature*, vol. 2742, pp. 173–183, 2024. doi: 10.1007/978-1-0716-3561-2_14.

[30] Sutoyo E., Almaarif A., Yanto I.T.R.: Sentiment Analysis of Student Evaluations of Teaching Using Deep Learning Approach. In: J.H. Abawajy, K.K.R. Choo, H. Chiroma (eds.), *International Conference on Emerging Applications and Technologies for Industry 4.0 (EATI'2020)*, Lecture Notes in Networks and Systems, vol. 254, pp. 272–281, Springer, Cham, 2021. doi: 10.1007/978-3-030-80216-5_20.

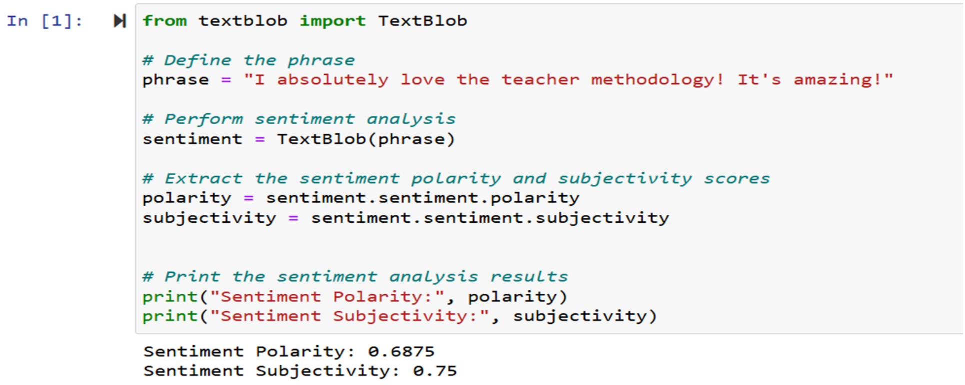
[31] Tzacheva A.A., Easwaran A.: Emotion Detection and Opinion Mining from Student Comments for Teaching Innovation Assessment, *International Journal of Education*, vol. 9(2), pp. 21–32, 2021. doi: 10.5121/ije2021.9203.

[32] Yu L.C., Lee C.W., Pan H.I., Chou C.Y., Chao P.Y., Chen Z.H., Tseng S.F., *et al.*: Improving early prediction of academic failure using sentiment analysis on self-evaluated comments, *Journal of Computer Assisted Learning*, vol. 34(4), pp. 358–365, 2018. doi: https://doi.org/10.1111/jcal.12247.

[33] Zhang Y., Sun S., Galley M., Chen Y.C., Brockett C., Gao X., Gao J., *et al.*: DIALOGPT: Large-Scale Generative Pre-training for Conversational Response Generation. In: A. Celikyilmaz, T.H. Wen (eds.), *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pp. 270–278, Association for Computational Linguistics, 2020. doi: 10.18653/v1/2020.acl-demos.30.

Appendix 1

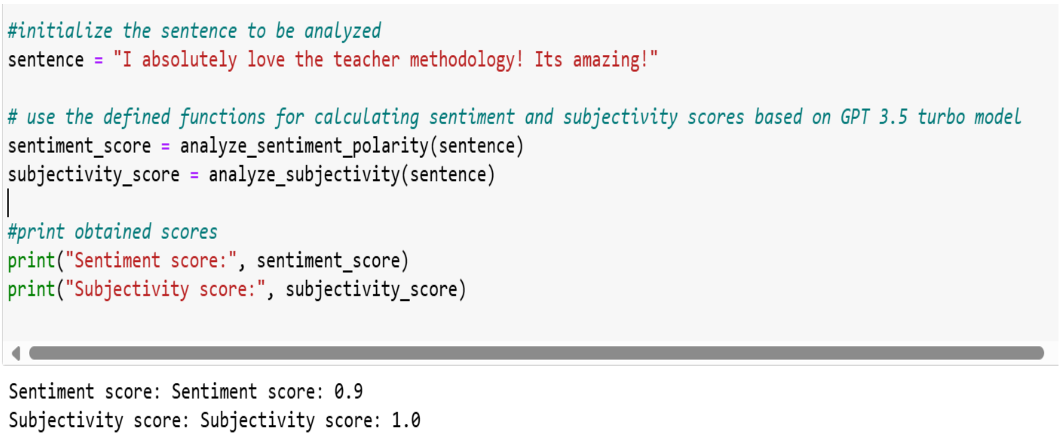
In Figure A1 and Figure A2, we provide examples of calculating sentiment polarity and subjectivity scores using two different tools: TextBlob and ChatGPT (the GPT-3.5-turbo model). These examples are based on the input sentence "I absolutely love the teacher's methodology! It's amazing!". As depicted in these figures, each tool generates distinct scores, highlighting the variations in sentiment analysis results.



```
from textblob import TextBlob

# Define the phrase
phrase = "I absolutely love the teacher methodology! It's amazing!"

# Perform sentiment analysis
sentiment = TextBlob(phrase)

# Extract the sentiment polarity and subjectivity scores
polarity = sentiment.sentiment.polarity
subjectivity = sentiment.sentiment.subjectivity


# Print the sentiment analysis results
print("Sentiment Polarity:", polarity)
print("Sentiment Subjectivity:", subjectivity)
```

```
Sentiment Polarity: 0.6875
Sentiment Subjectivity: 0.75
```

Figure A1. Example of building sentiment and subjectivity scores using the TextBlob python library for the sentence "I absolutely love the teacher methodology! It's amazing!".



```
#initialize the sentence to be analyzed
sentence = "I absolutely love the teacher methodology! Its amazing!"

# use the defined functions for calculating sentiment and subjectivity scores based on GPT 3.5 turbo model
sentiment_score = analyze_sentiment_polarity(sentence)
subjectivity_score = analyze_subjectivity(sentence)

#print obtained scores
print("Sentiment score:", sentiment_score)
print("Subjectivity score:", subjectivity_score)
```

```
Sentiment score: Sentiment score: 0.9
Subjectivity score: Subjectivity score: 1.0
```

Figure A2. Example of building sentiment polarity and subjectivity scores using ChatGPT through the OpenAI API and using the GPT-3.5-turbo model for the sentence "I absolutely love the teacher methodology! It's amazing!"

Appendix 2

This is a simple example that demonstrates how to utilize the OpenAI API in Python to generate sentiment and subjectivity scores for a given textual sentence. The code includes two designed functions that enables the calculation of sentiment polarity and subjectivity scores based on the provided textual input sentence.

```python
"""
@author: alaa
"""
import openai

openai.api_key =
↪  'sk-proj-03pxTK4NoD6GZv02pDumT3BlbkFJiLacDv1wId4A4SZ5UFVM'   # My
↪  OpenAI API key

from openai import OpenAI
client = OpenAI(
  api_key=openai.api_key,
)

def analyze_sentiment_polarity(sentence):
    response = client.chat.completions.create(
      model="gpt-3.5-turbo",
      messages=[
        {
          "role": "system",
          "content": "You will be provided with a given textual
          ↪  sentence in the context of education, and your task is
          ↪  to calculate a sentiment score between -1 and 1"

        },
        {
          "role": "user",
          "content": sentence
        }
      ],
      temperature=0.2,
      max_tokens=10,
      top_p=1
    )
    sentiment_score = response.choices[0].message.content
    return sentiment_score
```

```
def analyze_subjectivity(sentence):
    response = client.chat.completions.create(
      model="gpt-3.5-turbo",
      messages=[
        {
          "role": "system",
          "content": "You will be provided with a given textual
          ↪  sentence in the context of education, and your task is
          ↪  to calculate a subjectivity score between 0 and 1"
        },
        {
          "role": "user",
          "content": sentence
        }
      ],
      temperature=0.2,
      max_tokens=10,
      top_p=1
    )

    subjectivity_score = response.choices[0].message.content
    return subjectivity_score

#initialize the sentence to be analyzed
sentence = "I absolutely love the teacher methodology! Its amazing!"

# use the defined functions for calculating sentiment and subjectivity
↪  scores based on GPT 3.5 turbo model
sentiment_score = analyze_sentiment_polarity(sentence)
subjectivity_score = analyze_subjectivity(sentence)

#print obtained scores
print("Sentiment score:", sentiment_score)
print("Subjectivity score:", subjectivity_score)
```

Affiliations

Alaa Asim Qaffas
University of Jeddah, College of Business, aaqaffas@uj.edu.sa

Received: 5.03.2024
Revised: 7.06.2024
Accepted: 9.06.2024