ORDSAENet: Outlier Resilient Semantic Featured Deep Driven Sentiment Analysis Model for Education Domain

[1]Smitha B A and [2]Raja Praveen K N
[1,2]Jain Deemed to be University, Bangalore
[1]smitha.prasaad@gmail.com, [2]rajapraveen.k.n@gmail.com

Correspondence should be addressed to Smitha B A : smitha.prasaad@gmail.com

Article Info
Journal of Machine and Comp uting (http://anapub.co.ke/journals/jmc/jmc.html)
Doi: https://doi.org/10.53759/7669/jmc202303034
Received 26 February2023; Revised from 30 May 2023; Accepted 26 June 2023.
Available online 05 October 2023.

Abstract – The high pace rising global competitions across education sector has forced institutions to enhance aforesaid aspects, which require assessing students or related stakeholders' perception and opinion towards the learning materials, courses, learning methods or pedagogies, etc. To achieve it, the use of reviews by students can of paramount significance; yet, annotating student's opinion over huge heterogenous and unstructured data remains a tedious task. Though, the artificial intelligence (AI) and natural language processing (NLP) techniques can play decisive role; yet the conventional unsupervised lexicon, corpus-based solutions, and machine learning and/or deep driven approaches are found limited due to the different issues like class-imbalance, lack of contextual details, lack of long-term dependency, convergence, local minima etc. The aforesaid challenges can be severe over large inputs in Big Data ecosystems. In this reference, this paper proposed an outlier resilient semantic featuring deep driven sentiment analysis model (ORDSAENet) for educational domain sentiment annotations. To address data heterogeneity and unstructured-ness over unpredictable digital media, the ORDSAENet applies varied pre-processing methods including missing value removal, Unicode normalization, Emoji and Website link removal, removal of the words with numeric values, punctuations removal, lower case conversion, stop-word removal, lemmatization, and tokenization. Moreover, it applies a text size-constrained criteria to remove outlier texts from the input and hence improve ROI-specific learning for accurate annotation. The tokenized data was processed for Word2Vec assisted continuous bag-of-words (CBOW) semantic embedding followed by synthetic minority over-sampling with edited nearest neighbor (SMOTE-ENN) resampling. The resampled embedding matrix was then processed for Bi-LSTM feature extraction and learning that retains both local as well as contextual features to achieve efficient learning and classification. Executing ORDSAENet model over educational review dataset encompassing both qualitative reviews as well as quantitative ratings for the online courses, revealed that the proposed approach achieves average sentiment annotation accuracy, precision, recall, and F-Measure of 95.87%, 95.26%, 95.06% and 95.15%, respectively, which is higher than the LSTM driven standalone feature learning solutions and other state-of-arts. The overall simulation results and allied inferences confirm robustness of the ORDSAENet model towards real-time educational sentiment annotation solution.

Keywords – Educational Sentiment Analysis, Semantic Features, CBOW, Bi-LSTM, Long-Term Dependency, Contextual Feature.

I. INTRODUCTION

The last few years have witnessed exponential rise in software computing, mobile technologies, internet and Big Data analytics. It has broadened the horizon for the different applications and data ecosystems enabling organizations to make situational awareness-based decisions. Amongst the major innovative technologies serving industries for data driven decisions, sentiment analysis has emerged as a decisive approach that exploits user's online reviews to classify their sentiment polarity or perception. This as a result helps organizations improving the products and services [1-3]. In the current digital data world, performing sentiment annotations or labelling to each review manually is near impossible. Moreover, with rising data heterogeneity, unstructured-ness across online review datasets makes manual sentiment classification almost infeasible with manual approaches [2][4][5]. Despite challenges involved, sentiment analysis has been playing decisive role in industries including e-Commerce, finance, manufacturing, entertainment, politics etc. [5]. To meet industrial demands, automated sentiment analysis technologies have evolved by using AI and NLP techniques [1-4][6-9]. These techniques are used to extract sentiment or polarity from input text representing review or feedback towards certain product, service or event. It enables organizations to have actionable knowledge for proactive decisions [2]. Functionally, sentiment analysis methods exploit sentiment metrics from the input text to predict respective opinion, and this approach makes it suitable to serve an array of industries including healthcare, e-Commerce, finance [7][10]; yet, its efficacy remains confined for a domain specific problem. A sentiment analysis model designed for a specific domain might perform differently for other domain's review. On the contrary, ensuring reliability of a solution over unknown input texts has been becoming inevitable for industry. Though, in the past numerous AI and NLP driven sentiment analysis approaches are developed; yet, ensuring their efficacy towards education reviews and allied polarity assessment remains inferior due to low accuracy and high false positive performance [12]. On the contrary, the rising global population and allied educational demands, whether being served online or offline have forced educational institutions to exploit student's reviews to understand their opinion to improve educational deliveries and courses for better performance [12]. With this motivation, this paper emphasizes on designing a robust sentiment analysis model for educational review dataset.

Sentiment analysis mechanism involves sentiment annotation process designed to label a review sentence or document with corresponding sentiment emptions, where the emotions can be positive, negative or neutral. Sentiment annotation can be done for individual review or can be cumulative sentiment for multiple reviews or documents, where the sentiment annotation can be done based on respective sentiment score or contextual polarity. In education domain as well, the sentiment annotation can be done by performing micro- assessment of the quantum emotions present within the texts representing reviews for courses, pedagogy, teaching performance, learning efficacy etc. Sentiment analysis can be used for the educational domain, where the feedback provided by the students towards the different learning paradigms, approaches, course, content, teaching efficacy, practices, student teacher behavioral aspects etc. can be used to improve products and practices within setup [12-14]. Based on the sentiment observed, the educational deliveries, practices and infrastructures can be improved and can be streamlined. It can also help in predicting the student's performance to improve personalized learning [12][13]. In education domain, student's feedback towards learning material(s), pedagogy, teaching practices, performance, etc. can be exploited to understand their opinion [14]. It can help educational institutions make optimal and timely measures to improve learning, engagement and deliveries [15]. Educational institutions can exploit student's feedback after every semester to understand their opinion for the courses enrolled and the deliveries made [15]. Specifically, the use of statistical data as quantitative tool can be applied for student's feedback assessment, while qualitative feedback can be exploited by NLP and AI to understand their opinion. Such practices can involve feature extraction, feature selection and classification to perform opinion classification and labelling [16]. The preliminary step in sentiment analysis is to annotate text with respective sentiment tag (say, negative, neutral or positive). In real-world application, the level of the sentiment assessment might differ based on the application environment and allied demand [17]. There can be certain applications demanding a snippet of the student's perception and allied satisfaction report, while a few can employ topic level feedback and reviews to perform fine-grained analysis. The matter of fact is that the manual annotation of the sentiment for a large input review is near impossible and can be time-exhaustive [18].

Despite significances, ensuring precise sentiment annotation over large feedback data, especially with text heterogeneity and unstructured-ness has remained challenge [13]. In real-world data systems, there can be data heterogeneity with the different words constitutions, intends, etc. [12]. Additionally, the likelihood of data imbalance over large input text reviews can't be ruled out [3]. Deploying AI and ML models over heterogeneous, skewed data with the lack of structure can limit accuracy of the solution [10][12][19]. Unlike conventional feedback texts, typically followed by qualitative methods, the quantitative reviews over multiple opinion scores like satisfied, dissatisfied, neutral, extremely satisfied and extremely dissatisfied (say, Likert's scale) can make sentiment analysis more challenging. Thus, selecting an optimal data and computing environment is inevitable to achieve accurate sentiment analysis solution. The AI methods including machine learning, deep learning, and transformers [20] have played decisive role for student's opinion mining and student review annotations [21]. The recent development in unsupervised methods, AI driven sentiment analysis methods possess ability to alleviate the manual annotation cost; yet, ensuring efficiency over unstructured, heterogeneous data remains a challenge. The authors [3] suggested to extract student's opinion at varied levels such as entity level, sentence level, document level, which seems doable by using AI and NLP techniques [19][20]. In document level approaches, the complete reviews by a number of students can be exploited altogether to provide respective opinion label like negative, positive and neutral [22]. On the contrary, the sentence level method extracts opinion score for each sentence and label them as positive or negative. In entity level sentiment annotation methods, the reviews on an entity like course, teaching, assignment are exploited to label them. The aspect-based methods provide a fine-grained characterization of sentiment orientation on every data types. As per the educational purpose, the student's feedback data can be exploited and analyzed at the different levels for opinion annotation. For instance, an educational decision system can require opinion mining at both document-level, entity level as well as entity level. In this case, guaranteeing optimality of feature extraction, selection, learning and classification remains inevitable. Despite efforts, there are numerous challenges like class-imbalance, lack of contextual details in learning, impact of long-term dependency on the sentiment probability, local minima and convergence during training etc., which might impact the efficiency of the sentiment annotation solution. To generalize a solution, model requires robustness in terms of both data as well as feature optimality.

In sync with aforesaid research gaps and allied scopes, in this paper a robust outlier resilient semantic featured deep driven sentiment analysis model (ORDSAENet) is proposed for the education domain applications. The ORDSAENet model intends to address multiple aspects including data heterogeneity, unstructured-ness, semantic feature (latent information) learning, deep driven feature learning with contextual details, long term dependency and data imbalance that makes it robust towards real-time sentiment annotation. Considering the rising bots-driven content outliers on digital media, ORDSAENet at first performs text size-constrained outlier removal where being a sentence level sentiment analysis solution, the phrases with more than 250 words were removed from further processing. It helps in removing the irrelevant topic consideration in feature extraction and learning. Moreover, it also helps in removing outlier contents. It performs pre-processing over the input qualitative reviews from the students by applying the different pre-processing methods like missing value removal, Unicode normalization, Emoji removal, Website link removal, removal of the words with numeric values, non-word character(s) or Punctuations removal, lower case conversion, Stop-word removal, Lemmatization. Subsequently, it performs tokenization over the pre-processed text-sequences and thus converts input sentences into the corresponding tokens. The tokenized outputs from each review were then passed to the Word2Vec CBOW semantic embedding module that generated low-dimensional embedding metrics with intrinsically improved latent information. The embedded latent metrics was then fed as input to the SMOTE-ENN followed by Bidirectional LSTM (Bi-LSTM) deep network. Here, SMOTE-ENN alleviated any likelihood of class-imbalance, while Bi-LSTM exploited both local as well as contextual (say, global) features to perform learning and classification. Moreover, it addresses the issue of long-term dependency in text sentences, which often remains unaddressed by researchers. The proposed Bi-LSTM model was designed in such manner that it yields higher accuracy with minimum computational costs. The Bi-LSTM model exploited ADAM optimizer that in sync with cascaded entropy loss function performed multi-class classification to annotate each review sentence as "Extremely Unsatisfied", "Unsatisfied", "Neutral", "Satisfied" and "Extremely Satisfied", and labeled it with "1", "2", "3", "4" and "5", correspondingly. To assess efficacy of the ORDSAENet sentiment annotation model, the performance comparison was done with other CBOW feature driven ML algorithms such as the Naïve Bayes (NB), k-NN, SVM, logistic regression (LR), and decision tree (DT) algorithms. The simulation over the educational sentiment review dataset exhibited that the proposed ORDSAENet model can achieve the average sentiment classification accuracy of 95.87%, precision 95.26%, recall 95.06% and F-Measure of 95.15%. The overall performance characterization confirmed robustness of the proposed model towards real-time educational sentiment annotation tasks.

The other sections of this manuscript are divided as follows. Section II presents the related work, which is followed by the problem statement and research questions in Section III. Overall proposed model and its implementation is discussed in Section IV, while the research conclusion is given in Section V. The references used in this work is given at the end of the manuscript.

II. RELATED WORK

The sentiment analysis methods by using unsupervised and supervised approaches are discussed as follows:

Unsupervised Sentiment Annotation

The majority of the unsupervised sentiment analysis methods employ two approaches broadly the lexicon-based method and the corpus-based methods. Some of these methods designed towards educational sentiment analysis tasks are given as follows:

Lexicon-Based Sentiment Annotation

The matter of fact is that the opinion terms, also called sentiment word(s) plays vital role in identifying the sentiment polarity of a given text or sentence [23]. Identifying aforesaid sentiment words within a sentence it can label that sentence or document for its polarity in unsupervised manner [22]. The majority of the lexicon-based methods use sentiment dictionary containing lexical units such as the words or phrases and allied sentiment polarity like real values in the range of −1 to +1, polarity values like positive, negative, or neutral. Though, such dictionaries can have more fine-grained classes as well encompassing very positive, positive, neutral, negative and very negative [24-28]. In the past, the different lexicon-based methods are proposed English language texts sentiment annotation. Some of the key methods using sentiment dictionary for the sentiment polarity annotations are SentiWordNet [29], Opinion Finder [30], Bing Liu's Opinion Lexicon [31], MPQA subjectivity lexicon [32], Harvard General Inquirer, AFINN [33], SentiFul [34], Vader [35], TextBlob [36] etc. Tzacheva and Easwaran [37] used National Research Council (NRC) [38][39] lexicon dictionary to annotate student feedback in terms of the emotions like joy, fear, trust, anger, sadness, disgust, and anticipation in teaching. They used lexicon-based methods to assess the impact of active learning [40] and lightweight teams [41][42]. Though, such methods used emotion detection for sentiment analysis. To improve computation Rosalind and Suguna [43] focused on improving the Bing lexicon tool and proposed a customized sentiment lexicon (CSL), which was applied to assess sentiment polarity over the student's feedback towards courses provided. More specifically, the authors applied Bing lexicon with CSL method to perform sentiment analysis. For better efficiency, they tokenized the input sentences, which was followed by the estimation of the polarity score for each word by using Bing and CSL tools. Finally, the cumulative score was used to predict the sentiment class for a given feedback sentence. The use of lexicon-based methods can perform unsupervised labelling of the feedback data without manual labelling; yet, understanding the domain and context remains a challenge for these methods. The existing dictionary-based methods, which map certain keywords with allied polarity are found limited to label a sentence or document with the words out of the aforesaid dictionary. In other words, the likelihood of words beyond the predefined dictionary can't be ruled out in educational feedback data. To address this problem, corpus-based sentiment analysis methods are proposed.

Corpus-based Sentiment Annotation

A corpus represents a group of texts representing a specific domain and subjective topic. The corpus-based sentiment analysis methods employ concurrence statistics and word's syntactic patterns within the text corpora to annotate sentiment. It enables enhancing the sentiment lexicons with prior information related to the words throughout the semantic polarity of sentiment. The approach behind the corpus-based methods is to measure the semantic distance between a word and a set of polarity terms to measure the semantic polarity of the target word. Its efficiency can be understood by the fact that it can help to adapt the domain-independent sentiment lexicon to a domain-specific lexicon [44].

Supervised annotation techniques

Though, the above discussed lexicon and corpus-based sentiment annotation techniques are capable of labelling the student's feedback data with respective sentiment polarity; yet, annotating the opinions over a large number of reviews can be a challenge [45]. To alleviate it, AI and NLP methods have gained widespread attention where its ability to get trained over certain pre-processed NLP data enables predicting the sentiment class of an input sentence or document. For supervised learning-based sentiment annotations, three key approaches or techniques are developed. These are, the machine learning based methods, deep-learning based method and transformer-based methods.

Machine learning

In education domain, the learning management system (LMS) have gained attention globally for its potential to provide courses and allied data access for both online as well as offline learning and engagement. However, its efficacy often remains centered on the affinity of students. In this reference, the authors [45] assessed student feedback collected from a university towards the LMS driven course being provided. They applied machine learning method to classify student's feedback towards the course materials into sentiment classes like positive, negative, or neutral so as to help university for further improvement in the course materials and deliveries. They used six ML classifiers including multinomial logistic regression (LR), decision tree (DT), multi-layer perceptron (MLP), XGBoost, support vector classifier (SVM), Gaussian Naive Bayes (GNB), and k-nearest neighbors (k-NN) to perform student's sentiment annotation. The depth performance characterization revealed that the logistic regression method can yield superior results over the other state-of-art machine learning algorithms. Lwin et al. [46] exploited students review along with the quantitative ratings to perform sentiment classification. They performed clustering over the quantitative rating scores by using K-means clustering algorithm. They applied six ML algorithms including SVM, LR, MLP, and random forest (RF) to classify the clustered dataset. They manually labelled the textual feedbacks as positive or negative as their sentiment. Subsequently, a NB was applied to train the labelled dataset for further two-class classification (i.e., negative and positive). Faizi [47] performed learner's sentiment analysis towards YouTube education videos by exploiting reviews dataset. The authors used different ML algorithms including RF, LR, NB and SVM to learn the feedback datasets and classify feedback as positive or negative. They claimed that the SVM classifier could achieve superior accuracy (92.82%) than other machine learning algorithms. Kaur et al. [48] applied three machine learning algorithms employing RF, LR and SVM classifiers to perform sentiment classification. They applied lexicon-based tool called SentiwordNet to label student reviews and labelled them as positive or negative labels, which were used further to train the proposed hybrid classifier to perform sentiment classification. The authors found that their proposed hybrid classifier performed better than the standalone SVM classifier. Dolianiti et al. [49] performed document and sentence level sentiment annotation by using IBM Watson Natural Language Understanding, Microsoft Azure Text Analytics API, OpinionFinder 2.0, Repustate, and Senti-strength in educational domain text reviews data. The authors used two student's reviews data pertaining to the learning management system. Ahmad et al. [50] performed document level sentiment annotation model in faculty performance review data. They used SVM and NB on a pre-processed 5000 reviews data, where they achieved respective sentiment annotation accuracy of 72.80% and 81%, respectively. Sivakumar and Reddy [51] assessed semantic relatedness by using cosine similarity method for sentence level sentiment annotation in student feedback data. They used three machine learning methods DT, SVM, and NB to perform sentence level opinion annotation. Despite their effort to use lexicon-based SentiWordNet for sentence level annotation; it failed in addressing challenges like class-imbalance and data heterogeneity, which is common in real-world data systems. Similarly, a few methods [52] intended to extract one or more aspects from input review phrases or sentences and annotate them in two polarities like positive or negative. To improve efficacy, though they applied SVM, cascade classifier, and rule-based methods altogether that exhibited F-Measure of 0.94. Though, the use of SVM with a dictionary-based method exhibited 0.83. Yang [53] extracted automatic sentiment annotation model by exploiting educational reviews pertaining to the content, teacher, lesson, and curriculum. Ding et al. [54] used TF– IDF and Doc2Vec for sentiment annotation. To achieve better accuracy Gradient Boosting Tree and Linear SVM algorithms. Li and Lu [55] exploited latent information for entity-level sentiment annotation to label each entity as positive, negative or neutral sentiment. Shaik et al. [11] performed fine-grained assessment of sentiment opinion in educational review data. The authors suggested to use latent semantic information to perform sentiment annotation. Rosalind and Suguna [56] proposed an aspect-based sentiment analysis (ABSA) model by extracting the student's opinion towards online course by using machine learning method. Their approach at first segmented the student reviews by applying unsupervised and semi-supervised linear discriminant analysis (LDA) method. It was achieved by segmenting the input data into the sentences for aspect-based analysis for polarity annotation. They trained the maximum entropy classifier to annotate sentiment for multiple aspects; yet, the highest accuracy of 80.67% questions its generalizability. Edalati et al. [57] performed ABSA on student review data to assess their experience towards the lecturers and online teaching deliveries. The authors applied RF, SVM, DT, and deep learning method to label student opinions. The RF ensemble learning model exhibited an F1-Score of 98.01%. Despite the claims to have almost 97% F-Score, Wehbe et al. [58] could not address the problem of data imbalance amongst the students' reviews. Amongst the different machine learning classifiers RF was found performing superior towards fine grained sentiment classification in six emotions including happiness, surprise, sad, anger, fear, and disgust. Dehbozorgi and Mohandoss [59] developed a multi-class sentiment classification model for students' speech analysis and annotation. In their method, rule-based POS tagging was performed to extract aspects, which was followed by k-NN based prediction to perform student's performance by interpolating aspects and (student's) sentiment. Kastrati et al. [60] performed student's opinion mining and annotations in MOOCs related data, where CNN learning model exhibited the F-Measure of 82.10%.

Deep learning

Yu et al. [61] designed a deep learning based educational sentiment analysis model by CNN network. They applied CNN to learn over the structured data including attendance, grades and unstructured text reviews collected from a total of 181 students. They performed manual annotation of the text reviews with two classes; positive and negative using Self-Assessment Manikin rating method [62]. The extracted features were then trained by using SVM as well as cross-entropy driven Softmax classifier. The CNN-based model exhibited superior efficacy with an average F-measure of 0.74. Sutoyo et al. [63] applied CNN model to assess teacher's efficacy by exploiting student review data. They claimed to have achieved accuracy, precision, recall, and F-Measure of 87.95%, 87%, 78%, and 81%, correspondingly. To improve contextual learning abilities, they suggested to append attention layers which could enable identification of the sadness impact of words on emotion. Sangeetha and Prabha [64] designed a multi-head attention fusion concept for sentiment analysis in student's feedback data, where the input text sequences were processed in parallel throughput the multi-head attention layer by using Glove and Cove embedding. They designed multi-head attention layer with the aforesaid embedding layers that produced word-embedding matrix which were passed to the LSTM for learning. To improve learning accuracy, they regulated the dropouts of the different layers. Thus, applying this method they classified reviews as positive, negative or neutral. They inferred that the amalgamation of the multi-head attention with embedding layer and LSTM performs better than multi-head attention and LSTM individually.

Transformers

To exploit contextual details from the text reviews, the authors [65] designed a bidirectional encoder representation from transformers (BERT) for educational sentiment analysis. They used BERT to learn the student's feedback that helped identifying the top terms representing the negative and the positive polarities. They used K-Means clustering with cosine similarity over 300 MOOCs titles obtained from Udemy learning platform. Here. K-Means clustering obtained a total of 14 clusters, where the top terms were obtained from each cluster. Finally, the BERT model was applied to assess the associations between student–teacher, student-course, and learner's issues (text) descriptions to perform sentiment analysis. It indicated more negative sentiment towards courses than to the teachers. Li et al. [66] designed BERT-CNN model to perform sentiment analysis over the learning comments. They used BERT-CNN with a self-attention process which performed same as the classical BERT model even with significantly reduced design parameters. The BERT-CNN model performed better than other lexicon-based solutions with an accuracy of 92.8%, and F1-Score of 95.2% and 81.3% for Positive and negative class, correspondingly.

III. PROBLEM FORMULATION

The AI and NLP driven sentiment analysis methods can have the decisive role towards education industry where the institutions might exploit student reviews and feedback to improve at hand courses, learning pedagogies, teaching efficiency, infrastructures etc. However, disparity amongst the feedback over gigantically large reviews, especially encompassing the feedback sentences or phrases with different word constituents, intends etc. makes annotation more challenging. Approaches employing lexicon can't be optimal with such heterogenous data as there can be the large number of sentences which might fall beyond the predefined lexicon dictionary. Moreover, it can yield false positive results over phrases with mixed emotions. Unlike conventional lexicon and machine learning driven approaches, though, in the last few years numerous efforts have been made by exploiting deep learning models towards sentiment analysis; however, the efforts towards education domain are rare. In addition, most of the classical deep network driven approaches such as CNN, LSTM etc. have used local sentiment features from the input sentences and often fail in addressing long term dependency problem. This lack of long-term dependency limits learning to exhibit high reliability. It requires the use of a robust deep network which could exploit both local sentiment features as well as long-term contextual features for better learning and hence accuracy.

In sync with aforesaid inferences, this paper proposed ORDSAENet for educational sentiment analysis and annotation. Recalling fine-grained reviews and gradience of sentiment, this work considered primary student's review dataset encompassing both qualitative reviews as well as quantitative ratings for each review. In other words, almost 1.07 Lakhs student's reviews were collected with their textual feedbacks (say, qualitative feedback), which were also annotated in terms of the sentiment gradience levels like "Extremely Unsatisfied", "Unsatisfied", "Neutral", "Satisfied" and "Extremely Satisfied", which were annotated as "1", "2", "3", "4" and "5", respectively. In this work, the aforesaid quantitative ratings were obtained on Five Point's Likert's scale to have fine grained sentiment perception for each review or qualitative reviews. Thus, obtaining the qualitative reviews and corresponding quantitative ratings, ORDSAENet performed different pre-processing tasks including missing value removal, Unicode normalization, Emoji removal, Website link removal, removal of the words with numeric values, non-word character(s) or Punctuations removal, lower case conversion, Stop-word removal and Lemmatization. Considering online bot-driven fake reviews and promotional spams (say, outlier) in datasets, in addition to the above stated pre-processing tasks, an outlier removal strategy was applied that removed those review sentences having length of more than 250 words. Subsequently, to transform input text sequences in each review data, tokenization was performed that transformed text sequences into corresponding set of tokens. To ensure maximum possible intrinsic feature with efficient long-term dependency (learning) ability, the proposed ORDSAENet performs semantic word-embedding method by using CBOW word-embedding method. The embedded semantic metrics from the tokens pertaining to each qualitative review was processed for SMOTE-ENN resampling that alleviated any likelihood of class-imbalance in input data. Subsequently, the resampled embedding matrix was passed to the Bi-LSTM deep feature extraction and learning. Unlike traditional LSTM or other RNN networks, the use of Bi-LSTM helped extracting both local as well as contextual features (also called the global features) to perform learning and classification. The proposed ORDSAENet model designed and implemented Bi-LSTM model with ADAM optimizer, learning rate of 0.0001. Applying cascaded cross-entropy loss function it performed multi-class classification to annotate each review as "Extremely Unsatisfied", "Unsatisfied", "Neutral", "Satisfied" and "Extremely Satisfied", and annotated them with the label "1", "2", "3", "4" and "5", respectively. To assess whether the proposed CBOW semantic feature driven Bi-LSTM model (i.e., ORDSAENet) can achieve superior efficacy, the comparison was done with the different ML algorithms (here, NB, LR, k-NN, SVM, and DT) designed with CBOW feature driven sentiment analysis. The efficacy of the proposed ORDSAENet model was examined in terms of accuracy and F-Measure, which is compared with other sentiment analysis tools.

IV. RESEARCH QUESTIONS

In sync with the research goals, this research intends to achieve an optimal and justifiable answer(s) for the following questions:

RQ1: Can the use of the pre-processing methods like missing value removal, Unicode normalization, Emoji removal, Website link removal, removal of the words with numeric values, non-word character(s) or Punctuations removal, Lower case conversion, Stop-word removal, Lemmatization and tokenization be effective to address data heterogeneity and unstructured-ness problem of the real-time educational review datasets for sentiment analysis?

RQ2: Can the use of Word2Vec CBOW semantic embedding method with SMOTE-ENN resampling provide sufficiently large latent information for further feature extraction and learning towards educational sentiment analysis?

RQ3: Can the use of CBOW semantic embedding assisted Bi-LSTM deep model be effective (with local and contextual details with long-term dependency learning ability) towards educational sentiment analysis and annotation?

RQ4: Can the strategic amalgamation of aforesaid feature model (i.e., CBOW) and machine learning algorithms be effective towards real-time educational sentiment analysis?

RQ5: Can the proposed ORDSAENet be more efficient than the other state-of-art educational sentiment analysis solutions?

V. SYSTEM MODEL

The proposed ORDSAENet model encompasses these steps:

Phase-1 Data Acquisition,
Phase-2 Pre-Processing,
Phase-3 Semantic Word2Vec (CBOW) Embedding,
Phase-4 SMOTE-ENN Resampling,
Phase-5 Bi-LSTM Feature and Learning, and
Phase-6 Performance Analysis.

Data Acquisition

This is the matter of fact that there exist very few data available towards sentiment analysis in education domain. Despite rising significant the lack of efforts towards educational sentiment analysis, the limited data availability can be hypothesized. However, in sync with at hand research goal, we prepared own data by scrapping online reviews from EdTech companies including Udemy, Coursera etc. The reviews were collected from the different online educational websites, where scrapping the feedback the dataset was pre-labelled on the basis of their corresponding ratings. The data contained three components; first course identification code, second review, and third label. Here, course ID represents the unique identifier for the course, while review states the actual course review. On the contrary label states the rating for the course review. The considered dataset encompassed labels or the ratings on Five Point Likert's scale, where the students have provided label for each review in the form of Likert scale (i.e., 1-Extremely Dissatisfied, 2-Dissatisfied, 3-Neutral, 4-Satisfied, and 5-Extremely Satisfied). The considered dataset encompassed 1,07,018 number of reviews. The overall dataset encompassed qualitative as well as quantitative feedbacks pertaining to the online courses. Here, qualitative review presented student's reviews towards the course material, performance, faculty and allied cumulative perception and performance. On the contrary, the quantitative review possessed the ranking over 5-Points Likert scale. In other words, the students provided the rank or review rank to each feedback presenting qualitative feedback towards the education being provided and allied pedagogy (say, cumulative perception). The rating distribution for the considered dataset is given in **Table 1.**

Table 1. Data Characteristics

Rating Type (Quantitative)	Signifier	No of Samples	Examples (Qualitative)
1	Extremely Dissatisfied	2251	A lot of speaking without any sense. Skip it at all cost
2	Dissatisfied	2469	Course was not easy to grasp, lost interest after the first week and I barely got through the quiz at the end.
3	Neutral	5071	Full Individuals speaking
4	Satisfied	18054	Course is comprehensive and detailed. But is a bit fast paced for people new to IT Finance
5	Extremely Satisfied	79173	Great course - I recommend it for all, especially IT and Business Managers!

Observing the data (Table I), it can be found that one class (Extremely Satisfied) has 79173 reviews, while 2469 reviews possessed dissatisfied related reviews. The similar pattern can be found with other classes as well. It signifies the data imbalance case and hence training a model with such dataset can force the model to undergo skewed learning and classification. It can cause false positive or false negative performance. To alleviate this problem, the model requires proper data modelling with the ability to address class-imbalance. In sync with the data problems, in this paper the emphasis is made on designing a robust sentiment analysis model that addresses both data limitations as well as allied computational efficacy. To ensure optimal data quality, different pre-processing methods were applied. Though, the considered dataset possessed uniform data structure with minimum data heterogeneity or noise elements. The word-cloud of the input dataset for the different sentiment labels is given in Fig. 1.

Pre-processing

Though the considered dataset didn't have significant noise elements; however, in sync with the real-time application ecosystem, especially in Big Data, such data conditions can't be ruled out. In real-world data the text reviews can possess emotional as well as aspect's disparity, where the users can give feedback with the different words, text-compilation and emotions. Therefore, suppressing differences resembling the noise elements can be vital. The likelihood of outliers, signifying the text or phrases which are irrelevant to the target sentiment domain (here, education domain) can't be ruled out. Training any ML model over such irrelevant data and allied features might impact overall sentiment classification accuracy, and hence to achieve a robust and reliable opinion mining model over the real-world datasets, performing pre-processing is must. In this work varied pre-processing methods are used. These methods are:

- *Outlier removal,*
- *Hashtag and Website removal,*
- *Emoji removal,*
- *Missing value removal,*
- *Punctuation removal*
- *Numeric word removal,*
- *Unicode Normalization,*

- *Stopping word removal,*
- *Lower case conversion,*
- *Lemmatization,*
- *Tokenization.*

A brief of these pre-processing models is given as follows:

Outlier Removal

Typically, on digital data platforms including social media, blogs, review platforms etc., it is found that in addition to the subject specific discussion or reviews the users dump irrelevant text content as well. Such practices have been done by manual posting as well as by automated AI-bots driven solutions, where the tools identify the web-location to drop its target data including backlinks, and other unknown or unexpected text content. For example, a platform named "Quora" discussing the reviews about new education policy (NEP) can have the non-NEP related texts, intentionally created by the bloggers, search engine optimization professionals and even AI-bots. Considering aforesaid non-ROI text data and allied feature extraction and training might impact overall performance. Therefore, it becomes vital to remove such elements from the source text. Additionally, the aforesaid outliers might put a large chunk of text data encompassing dozens of the sentences with ambiguous text-components or intends. Training any model over such ambiguous data might result false positive or false negative result. To alleviate it, outlier removal can be of great significance. A basic approach to remove such irrelevant text outlier reviews can be to constrain the phrases with certain text-length. In ORDSAENet, we defined the text size as 250 words. In this method, once identifying the relevant text or subject type each review sentence was trimmed or cropped to retain merely 250 words of the data, while the remaining texts were dropped down from any further computation. Though, identifying a relevant text type required pre-defined data map or dictionary driven solution, we have kept that out of this research scope, which would be done in future to strengthen ORDSAENet with outlier data removal at application gateway side.

Hashtag and website link removal

To gain competitive markets, SEO has been the dominant tool, especially to improve backlinks and digital world penetration. In other words, professionals use different sophisticatedly designed tools such as Hashtag ("#") and URL weblinks to improve SEO or allied reachability. To achieve it, the individual or firms often use manual or bot techniques to drop contents with hashtag and weblink. However, in reality such hashtags and weblinks don't have significance or intend towards sentiment or opining analysis. Such outliers on the other hand might impact sanity of the sentiment features and hence overall learning potential. To alleviate aforesaid problems, our proposed ORDSAENet model performed hashtag and URL removal from the input text contents. To achieve it, we applied Python inbuilt functions that detect hashtag and URL contents and removes from the original data content. Once removing these outliers, it concatenates the words within the sentence to have linearity across text composition.

Fig 1. Word-Cloud Presentation for the Different Ratings

Emoji Removal

Similar to the hashtag components, the model digital world provides users an attractive and perceptually engaging tool called "Emoji" signifying sentiment or emotional aspects. The users often use Emoji to present their emotional aspects and state of mind; however, in digital processing such elements turn-out to be outlier due to the lack of any relatable significance with other text constituents. The non-text presentation of emoji (For instance, the happiness mood is stated by "😊"; however, the textual components of this emoji are ":" and ")", which as cumulative content (i.e., ":", ")") doesn't have any relationship with sentiment label or aspect. To ensure optimal feature extraction and learning, removing these emoji contents can be vital. ORDSAENet used NLTK (NLP) inbuilt function that removed emoji content(s) from each sentence.

Missing Value Removal

To alleviate any possibility of broken sentences, missing word or text constituents, which is unavoidable in real-world data ecosystem, our proposed ORDSAENet applied NLTK library that removed the content with incomplete communication or meaning.

Punctuation removal

In text data, users often use non-text elements or non-word elements (say, symbols or the punctuation marks like ex. “,”, “?”, “!”,”;”, “:”, etc.). These punctuation marks are used in phrases to enhance textual understanding and provide emphasis. Noticeably, as stated above such symbols don’t have the impact on sentiment polarity, yet, its presence in data might makes feature extraction and learning inferior. Moreover, it can make NLP solution more complex and ambiguous due to the lack of intrinsic feature continuity and learnable feature environment. Thus, employing a regular expression approach and rule-based method ORDSAENet model removed punctuations or non-word characters.

Numeric word removal

Sentiment polarity being reliant on constituting words (in each review) remains unimpacted due to the presence of numerical components. For instance, in text-based educational sentiment analysis tool, there can be the data elements like “90% marks”, “50000 Rs. fees”, etc. don’t have impact on polarity label. Therefore, to ensure feature optimality the proposed model removes numeric data elements. To achieve it, NLTK tools and rule-based methods are applied to remove numeric words.

Unicode Normalization

In the last few years, the development in competitive and user-friendly applications where the application(s) provides users the ability to write texts in the different fonts or languages. In this case, the likelihood of the text presence in the different languages, fonts (Times New Roman, Helvetica, etc.) or writing style (Ex. Italics, bold, etc.) can’t be ruled out. Such diversity can make NLP more complex and difficult to achieve expected performance. To alleviate such problems, our proposed ORDSAENet model performed Unicode normalization that transformed entire text into single font and hence achieved uniform data for further tokenization and allied semantic embedding.

Stopping word removal

Similar to the aforesaid non-text components, stopping words which states the English words which add no value to the sentence and hence makes embedding more complex. In this work, our proposed ORDSAENet model removed stop-words from the input text or review sentences without changing the real intend of the sentence. We applied NLTK corpus stop-words to remove existing stop-words from the input texts.

Lower case conversion

Typically, the machine learning algorithms are case-sensitive and therefore the differences amongst the case might make learning more complex during training. To alleviate this problem, ORDSAENet model performed lower case conversion by using Python’s NLTK libraries and allied lower-case conversion tool.

Lemmatization

This method transforms an extended word into its root form. It measures the real intended component of speech perfectly without losing the sense of a word in a phrase. In general, stemming and lemmatization differs, where the lemmatization method examines the context, which is then followed by transforming the extended word into the appropriate root word. Unlike lemmatization, stemming merely extended characters like “s”, “es” etc. are removed at the end of the word. Stemming is criticized due to its resulting compromises with the actual or intended meaning of the word in the review or feedback phrases. The illustration of the different lemmatization and stemming can be found as follows:

Hobbies → Lemmatization→ Hobby

Hobbies → Stemming→ Hobbi.

The lemmatization can be of decisive significance, and therefore this work performed lemmatization over the input text sequences, which was later processed for tokenization.

Tokenization

Once converting the raw inputs into structured and noise free text sequences, the input sentences were processed for tokenization that transformed input sentences into “tokens”. A token can be a phrase comprising a few words, a number, or any symbol that possesses all allied relevant information related to the data. Once obtaining the tokens from each input review sentences, it was processed for semantic word-embedding.

Semantic Word2Vec CBOW Embedding

Recalling the significance of the latent feature learning towards NLP problems [43][63], in this work the tokenized outputs were processed for the latent semantic feature extraction. Here, the key intend was to retain maximum possible local as well as contextual features to perform learning and classification. In majority of the state-of-arts the authors have directly passed tokenized data as input to the deep networks like LSTM that limits their ability to carry merely the local features. On the contrary, this study hypothesizes to train the model with both local as well as global features with latent details so as to make learning superior. ORDSAENet initially performs semantic embedding, which is then followed by deep feature extraction. Here, the motive was to retain the latent details from the input text data, which could be effectively processed for deep feature extraction by using Bi-LSTM which has emerged as a potential approach to retain maximum possible intrinsic features (including local as well as contextual features) from the input sequential data. Moreover, it addresses the problem of long-term dependency that strengthens ORDSAENet to achieve better learning and hence superior reliability. In the past, the different semantic embedding methods are proposed such as Word2Vec, Glove [64], TF-IDF etc.; however, the variants of Word2Vec like CBOW has performed superior. This is because of its low-dimensional intrinsic feature capability [67]. Our proposed ORDSAENet model used Word2Vec driven CBOW word-embedding method which converted input tokens into the equivalent word-level embedding matrix. It applies neuro-computing to learn the associations amongst the tokens and obtains text-features for each word in the input text-corpus, representing the review dataset. The proposed method applied a list of numbers, also called "vector" to represent each unique word in the input text. The potential to extract latent semantic information from text without human intervention makes it suitable for educational sentiment analysis tool. In function, the CBOW contains two sets of vectors (often called the word-embedding vectors), "Source-side" and "Target-side" vectors stating $v_w, v'_w \in \mathbb{R}^d$ for each input tokens (let, $w \in V$ be the tokenized data). Being a Gensim driven approach, a text window within the input text corpus embodies central token w_0 and thus generates allied context embedded vector $w_1, \ldots, w_C$. In this manner, it retrieved the CBOW loss as per the equation (1).

$$v_c = \frac{1}{C}\sum_{j=1}^{C} v_{w_j} \tag{1}$$

$$\mathcal{L} = -\log\sigma\left(v'_{w0}\,T_{v_c}\right) - \sum_{i=1}^{k}\log\sigma\left(-v'_{n_i}\,T_{v_c}\right) \tag{2}$$

In (2) $n_1, \ldots, n_k \in V$ represents the negative examples generated from the noise distribution P_n over the input token vector V. Here, $\mathcal{L}$ gradient is estimated with respect to the target value v'_{w0}, negative target value v'_{n_i} and average context source (v_c) by using following mathematical models.

$$\frac{\partial\mathcal{L}}{\partial v'_{w0}} = \left(\sigma\left(v'_{w0}\;T_{v_c}\right) - 1\right)v_c \tag{3}$$

$$\frac{\partial\mathcal{L}}{\partial v'_{n_i}} = \left(\sigma\left(v'_{n_i}\;T_{v_c}\right) - 1\right)v_c \tag{4}$$

$$\frac{\partial\mathcal{L}}{\partial v_c} = \left(\sigma\left(v'_{w0}\;T_{v_c}\right) - 1\right)v'_{w0} + \sum_{i=1}^{k}\left(\sigma\left(v'_{n_i}\;T_{v_c}\right) - 1\right)v'_{n_i} \tag{5}$$

Thus, applying the Chain-rule mechanism over the source context embedding, our proposed model retrieved the gradient of the predicted word vector, as given in equation (6).

$$\frac{\partial\mathcal{L}}{\partial v_{w_j}} = \frac{1}{C}\left[\left(\sigma\left(v'_{w0}\;T_{v_c}\right) - 1\right)v'_{w0} + \sum_{i=1}^{k}\left(\sigma\left(v'_{n_i}\;T_{v_c}\right) - 1\right)v'_{n_i}\right] \tag{6}$$

In ORDSAENet to alleviate any likelihood of improper context vector update, the context word normalization is performed over the extracted embedding vectors. To achieve it, the context window's width was randomly sampled in the defined range of 1 to C_{max} for each target word. Thus, ORDSAENet obtained word-embedding metrics for each input sentence.

SMOTE-ENN Resampling

In the real-time data system, the likelihood of class-imbalance can't be ruled out, especially over the dynamic data conditions. Therefore, training any machine learning model over skewed data or class-imbalanced data might cause frequent false positive or false negative performance. To alleviate it, ORDSAENet performed resampling over the semantic embedding metrics (6).

Unlike up-sampling or random sampling techniques, which are often criticized for its iterative hotspot creation (say, frequent imbalance), we used SMOTE-ENN that applied ML method to select the optimal set of synthetic samples which could leverage the difference of the majority and the minority class samples. The SMOTE generates synthetic samples representing highly correlated features, without affecting the original sample distribution across the dataset. SMOTE-ENN applied minority samples from the different classes as input to generate the synthetic samples, which were later processed with k-NN classifier with the Euclidean distance. It formed a vector between the current samples and the one from obtained k-neighbours. The obtained vector is then multiplied with a random number existing in between 0 to 1, which was appended to the original sample to obtain the final synthesized sample as output. Noticeably, in traditional SMOTE defining class-boundaries can be difficult as there can be synthetic minority samples undergoing cross-over or overlap with the majority class. The severity of such overlap can be high over the large non-linear features, and therefore, it can mis-label the synthetic samples. Training a model with miss-labelled data can cause false positive or false negative results. To alleviate it, ORDSAENet applied SMOTE-ENN algorithm that applies Edited Nearest Neighbour (ENN) in which the label of each synthetic sample is compared with respect to the vote of its k-NNs neighbours. In case it finds any inconsistency between the input sample and corresponding k-NNs, it drops that sample from the synthetic sample set, else it retains the same to append the original sample. The higher k enables stringent cleaning and hence appends original data with the optimal set of synthetic samples to alleviate class-imbalance problem. The pseudo-code for the SMOTE-ENN model is given as follows:

Pseudo Algorithm-1 SMOTE-ENN Resampling
Input: Semantic (Latent) Embedded Metrics

Step-1 Oversampling

1. Select a sample x_i arbitrarily from the minority class samples
2. Search for k-NN samples of x_i
3. Generate the synthetic sample p by randomly selecting one of the k-Nearest neighbours q, and connect p and q to create a line segment in the feature space
4. Assign the minority class label to the newly created synthetic sample
5. Generate successive synthetic samples as a convex combination of the two selected samples.

Step-2 Under sampling

6. Select the sample $x_i \in S$, where S being the total number of samples x_i from the minority class.
7. Search for the k-NN sample of x_i.
8. In case x_i has more neighbours from other class(es), then discard x_i.
9. Repeat the process 6-8 for all examples in the dataset.

Output: Balanced sentiment dataset

Now, once obtaining the CBOW embedding matrix from the input text sequences, the outputs were passed to the Bi-LSTM network for further feature extraction and learning. Noticeably, unlike traditional approaches including ML and deep driven solutions, Bi-LSTM performs encoding and decoding over the input text sequences (or sequential inputs). Its ability to perform self-embedding, feature extraction, learning and classification makes it robust towards real-time prediction or classification tasks, for example the at hand sentiment prediction problem. A brief of the proposed Bi-LSTM feature extraction and learning model is given in the subsequent section.

Bi-LSTM Learning.

Unlike classical deep networks such as CNN, recurrent neural network (RNN (LSTM)), which merely exploit local features from the input text sequences, our proposed ORDSAENet model applies Bi-LSTM RNN. Unlike classical LSTM model that can merely retrieve local deep features, Bi-LSTM can achieve both local as well contextual features for further learning and classification. Before discussing Bi-LSTM deep network, a brief of the LSTM network is given as follows:

LSTM Network

The LSTM model was initially designed to address the issue of vanishing effect in deep feature extraction and learning, especially in the RNN. The basic approach behind the implementation of LSTM network is to control the cell-states by applying a set of gate components, called input gate, forget gate and output gates. As illustrated in Fig. 2, in LSTM the forget gate f_t examines whether it needs to keep the information pertaining to the previous state (c_{t-1}) or forget it by assessing the values of the input (x_t) and the hidden state (h_{t-1}). The output values for this gate can be either 0 or 1. Similarly, the input gate (i_t) estimates the level of information related to the input text (x_t) and the hidden layer information (h_{t-1}) which has to be passed to update corresponding cell-state, to achieve output either as 0 or 1. The parameter c_t signifies the retrieved cell state by using mathematical formula on c_{t-1}, f_t and i_t. In this process, the information flow in between the current cell state to the hidden state is controlled by the output gate (O_t) that exists in either state 0 or 1. Let, at certain time t, let $x_t$be the input to the LSTM, and the earlier hidden state be h_{t-1}, with the descendent cell-state c_{t-1}. Moreover, consider that the current output of the hidden state and the current cell state be h_t and c_t, correspondingly. Now, in this configuration (Fig. 1), the different gate elements and their respective outputs can be derived as per (7-11).

$$f_t = sigmoid\left(W_{fx}x_t + W_{fh}h_{t-1} + b_f\right) \tag{7}$$

$$i_t = sigmoid(W_{ix}x_t + W_{ih}h_{t-1} + b_i) \tag{8}$$

$$c_t = c_{t-1} \odot f_t + i_t \odot tanh(W_{cx}x_t + W_{ch}h_{t-1} + b_c) \tag{9}$$

$$O_t = sigmoid(W_{ox}x_t + W_{ch}h_{t-1} + b_0) \tag{10}$$

$$h_t = O_t tanh(c_t) \tag{11}$$

In (7-11), $x_t \in R^n$ be the input vector, $W \in R^{v*n}$, $b \in R^v$, where the superscript variables n and v be the dimensions of the input vector and the words in input corpus, correspondingly.

Input Layer

The embedded vectors were passed to the input layer of the Bi-LSTM, as x_t. Consider, $w_1, w_2, w_3, \ldots, w_v$ be the total count of the unique words in each tokenized dictionary $D = d_1, d_2, d_3, \ldots, d_m$ and $i_1, i_2, i_3, \ldots, i_v$ pertains to the total unique indices, referring the natural numbers, where 1 and v states the first and the last data index or the vocabulary, respectively. In the proposed Bi-LSTM, the input vectors are inputted as sequential data with a definite length possessing unique indices. In the subsequent layer, it uses an embedding layer where each word-embedded element index, belonging to the unique text words in the data corpus, is converted to the equivalent real-valued feature vector. The retrieved real-valued feature vectors are stacked in the form of a matrix, often called the embedding matrix (12).

$$R = \begin{array}{cccc} r_{1,1} & r_{1,2} & \cdots & r_{1,n} \\ r_{2,1} & r_{2,2} & \cdots & r_{2,n} \\ r_{3,1} & r_{3,2} & \cdots & r_{3,n} \\ \vdots & \vdots & \ddots & \vdots \\ r_{v-1,1} & r_{v-1,2} & \cdots & r_{v-1,n} \\ r_{v,1} & r_{v,1} & \cdots & r_{v,n} \end{array} \tag{12}$$

In the derived embedding matrix (12), each row states a unique index representing a unique word in the vocabulary. The dimension of the embedding matrix be $v * d$, with v and d as the dataset size and the dimension of the dense vector, respectively. Here, d was assigned as 320 to retrieve embedding vector.

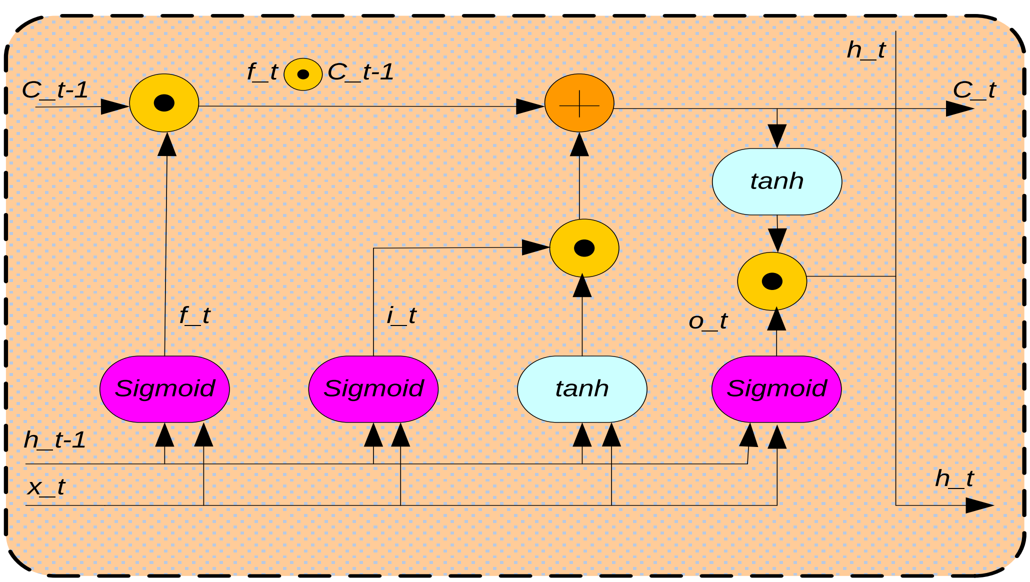

Fig 2. Functional Diagram of LSTM Network

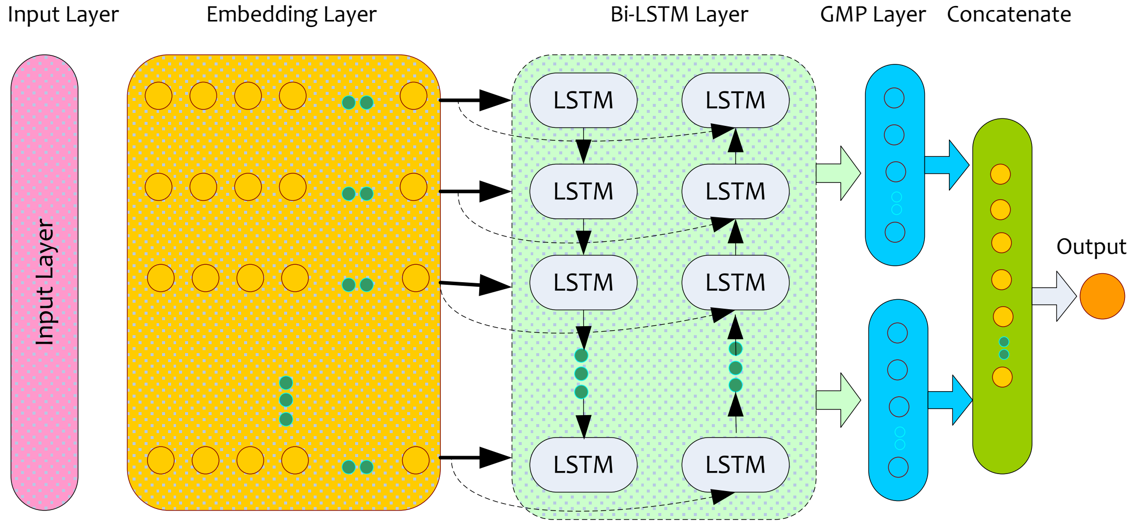

Fig 3. Bi-LSTM Layer

A brief of the Bi-LSTM with the Word2Vec embedding layer is given as follows:

Bi-LSTM Layer

Unlike traditional LSTM network, in which the information traverses in forward direction, the Bi-LSTM retains information flow in forward as well as backward directions. In Bi-LSTM, the state at time t depends not only on the information before t, but also on other sequential periods. To define the complete semantic information over the input review text (and allied embedding metrics), the use of subsequent information can be vital. Unlike traditional CNN and/or LSTM driven local feature extraction, ORDSAENet applies Bi-LSTM that exploits both the previous states as well as the subsequent information to achieve more semantically enriched feature or the contextual information for learning and classification. In this work, Bi-LSTM was designed with two LSTM networks, each capable of processing input vectors in both forward as well as backward direction (Fig. 3). In this work, our proposed ORDSAENet employed each LSTM for feature extraction in single direction (i.e., one for forward direction and another for backward direction). The forward LSTM processes input from left-to-right, and hence its hidden layer is defined as per (13). On the other hand, the backward LSTM processes information in right-to-left direction, and therefore the allied hidden layer information from backward LSTM is obtained as per the equations (14).

$$\vec{h}_t = LSTM(x_t, \vec{h}_{t-1}) \tag{13}$$

$$\overleftarrow{h}_t = LSTM(x_t, \overleftarrow{h}_{t+1}) \tag{14}$$

The outputs from Bi-LSTM network (13-14) are concatenated with the final contextual information, as given in equation (15).

$$h_{t,Bi-LSTM} = [\vec{h}_t, \overleftarrow{h}_t] \tag{15}$$

Unlike LSTM driven solution, which applies unidirectional feature propagation, ORDSAENet used bidirectional mechanism and hence it retrieved $\vec{h}_t$ and $\overleftarrow{h}_t$, independently. Now, the retrieved features were concatenated by using global pooling layer to perform further learning and classification. The finally extracted feature vector $h_{t,Bi-LSTM}$ (15) were passed to the feature optimization model by performing resampling, feature selection and normalization methods. In the proposed ORDSAENet model, the deployed Bi-LSTM network was applied with ADAM non-linear optimizer that in conjunction with the learning rate of 0.0001 and cascaded binary cross-entropy cost function performed multi-class classification and classified each input sentence as “Extremely Unsatisfied”, “Unsatisfied”, “Neutral”, “Satisfied” and “Extremely Satisfied”, and labelled them with the label “1”, “2”, “3”, “4” and “5”, respectively.

In addition to the proposed SMOTE-ENN resampled CBOW feature driven Bi-LSTM learning model, we designed other simulation environments as well, where the embedded matrix was processed for classification by using the different machine learning classifiers, including Naïve Bayes (NB), support vector machine (SVM), decision tree (DT), logarithmic regression (LR) and k-NN classifiers. Noticeably, in these ML classifiers, the resampled word-embedding outputs were passed as input to perform training and classification. A brief of these ML algorithms is given as follows:

ML Classification

To assess relative performance, we applied the different machine learning classifiers to perform classification. More specifically, we applied the following machine learning methods to perform classification.

- *Naïve Bayes (NB)*
- *Support Vector Machine (SVM)*
- *Decision Tree (DT)*
- *Enhanced k-NN, and*
- *Logistic Regression (LR).*

A brief of these ML algorithms is given as follows:

Gaussian Naïve Bayes (GNB) Classifiers

Naïve Bayes classifier is a well-known and most employed probabilistic classification algorithm that applies Bayes' rules to learn and classify input pattern(s). Being probabilistic in nature this algorithm is also called as the "autonomous feature model" applies an assumption that the associated features remain independent of each other and hence don't influence learning and classification outputs, decisively. It also assumes that the existence of a particular feature attribute within a class can't be connected to the availability of the other feature. Functionally, the Naïve Bayes algorithm allocates a value x with the class probability $\mathrm{e}^* = \mathrm{argmax}_\mathrm{d} \mathrm{P}(\mathrm{d}|\mathrm{x})$ in reference to the Bayes′ rule. Mathematically, it can be represented as per the equation (16).

$$P(d|x) = \frac{P(x|d)P(d)}{P(x)} \tag{16}$$

In (16), $P(d)$ states the probability of a class c, while the likelihood of the data element x is given by $P(d|x)$. The other parameter $P(x)$ states the prior probability of the predictor (17).

$$P(x|d) = \prod_{l=1}^{m} P(x_l|d) \tag{17}$$

We applied GNB classifier to classify each review sentence. In case of GNB, the proposed model hypothesizes that the samples follow the normal distribution. Here, GNB performed classification of each input review sentence as labelled them as "1", "2", "3", "4" and "5", for respective emotions like "Extremely Unsatisfied", "Unsatisfied", "Neutral", "Satisfied" and "Extremely Satisfied", respectively.

Support Vector Machine (SVM)

SVM presents a type of supervised machine learning method, usually applied for the pattern learning and classification. The ability to learn and classify input features by using hyper-plane makes SVM one of the most used machine learning algorithms for text and image classification problems. We used SVM to learn over the normalized features obtained from the data warehouse, where it functions as a non-probabilistic binary classifier algorithm. This method reduces the generalization error iteratively over the normalized features. The support vector was measured to present a training subset depicting the boundary conditions, also called the hyper-plane. We measured support vector to define the hyper-plane between the classes and applied (18) to classify each review sentence.

$$Y' = w * \phi(x) + b \tag{18}$$

In (18), $\phi(x)$ states non-linear transform function that focuses on assigning the suitable values for the weights w and the biases b to exhibit sentiment classification. We estimated the outputs $Y'$by minimizing a regression-risk parameter, given in (19).

$$R_{reg}(Y') = C * \sum_{i=0}^{l} \gamma(Y_i' - Y_i) + \frac{1}{2} * \|w\|^2 \tag{19}$$

In (19), C and γ signify the penalty factor and the cost-function, respectively. We calculated weights as per (20).

$$w = \sum_{j=1}^{l} (\alpha_j - \alpha_j^*) \phi(x_j) \tag{20}$$

In (20), α and α^*state the non-zero value, called the Lagrange relaxation. Thus, SVM classifies each input sentence as per (21).

$$Y' = \sum_{j=1}^{l} (\alpha_j - \alpha_j^*) \phi(x_j) * \phi(x) + b \\ = \sum_{j=1}^{l} (\alpha_j - \alpha_j^*) * K(x_j, x) + b \tag{21}$$

In (21), $K(x_j, x)$ states the kernel function, which we applied as the radial basis function (RBF) in this work.

Decision Tree (DT)

The DT algorithm is one of the most employed association rule mining algorithms used for text mining and data analysis tasks. It has evolved over years in the form of the algorithms such as CART, ID3, C4.5 and C5.0 methods. It has been used in varied ensemble-learning tasks. The DT algorithm starts at the root node, where it uses association rule that in conjunction with a split-condition divides the input data into the multiple branches. These branches are generated at each node of the tree. Subsequently, it applies information gain ratio (IGR) over each branch for pattern learning. Once splitting the input features into multiple branches, it generates other nodes that eventually branch-off other data. It resembles a tree with multiple branches with unit parent node having multiple children's nodes, also called left and right nodes.

Let, the left and the right child node be LC_d and RC_d, respectively. Consider x represents the input feature, while I be the noise value. Thus, with the available samples in P_d, LC_d and RC_d, it enhances information gain, repeatedly by using (22).

$$Information\ Gain\ (P_d x) = I(P_d) - \frac{LC_n}{P_n} I(L.C_d) - \frac{RC_n}{P_n} I(R.C_d) \tag{22}$$

In (22), I is measured by using varied methods like Gini-Index I_G, entropy I_H or the classification error I_E. Mathematically, it can be derived as per the equations (23-25), respectively.

$$I_H(n) = -\sum_{i=1}^{c} p(c|n) \log_2 p(c|n) \tag{23}$$

$$I_G(n) = 1 - \sum_{i=1}^{c} p(c|n)^2 \tag{24}$$

$$I_E(n) = 1 - max\{p(c|n)\} \tag{25}$$

In (23-25), c and n states the class(es) and the respective node n. The probability parameter is measured as the ratio of c to n. Thus, based on the predicted output, each review is annotated as "1", "2", "3", "4" and "5", for respective emotions like "Extremely Unsatisfied", "Unsatisfied", "Neutral", "Satisfied" and "Extremely Satisfied", respectively.

Logistic Regression

We applied regression over the input resampled CBOW semantic embedding matrix, where the data features were considered as the independent variable, while the sentiment class was considered as the dependent variable. We used equation (26) as the regression function to perform logistic regression-based classification.

$$logit[\pi(x)] = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots .... + \beta_m X_m \tag{26}$$

$$logit[\pi(x)] \tag{27}$$

In (27), x_i states the independent variable signifying the input semantic feature taken from the data warehouse, while $logit[\pi(x)]$ function presents the dependent variable. Here, $logit[\pi(x)]$ transformed the dichotomous results by applying $logit$ function and thus it resulted varied outputs of $\pi(x)$ in the range of 0 to 1 to $-\infty$ to$+\infty$. Here, the variable m states the independent variables, and the sentiment probability (28) is estimated as π, which is measured as per the equation (28).

$$\pi(x) = \frac{e^{\beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots .... + \beta_m X_m}}{1 + e^{\beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots .... + \beta_m X_m}} \tag{28}$$

Applying (28), the LOGR (logistic regression) algorithm classified each review sentence as positive, negative and neutral and labeled them as 1, -1 and 0, respectively.

Enhanced k-NN Classifier

k-NN is one of the most used classifiers that classifies unannotated patterns or data by assigning the class label of the most similar labelled pattern. Being simple in nature and computationally efficient, k-NN is one of the most used algorithms for data science (say, data mining and regression) and allied predictive problems. Its abilities enable it to be used for the binary classification tasks. Here, k-NN applied Euclidean distance to measure inter-data or inter-element distance using (29). In (29), p and q are projected for the comparison with the n features. Though, one can use Manhattan distance as well.

$$D(p,q) = \sqrt{(p_1 - q_1)^2 + (p_2 - q_2)^2 + (p_n - q_n)^2} \tag{31}$$

In k-NN algorithm, the performance largely relies on the value of k that decides the number of neighbors which can be selected to perform learning and classification. The selection of an optimal k enables optimal performance. The larger k reduces the impact of the variance caused by the arbitrary error and hence forces it to be employed with the smaller data size. And therefore, the force retaining balance between computational cost and performance (by choosing suitable k) relies on the ability to balance over-fitting and under-fitting. The classical methods have employed k-value as the square root of the number of the data elements in the training data; however, its efficiency over large- data, especially with non-linear patterns can't be guaranteed. In major state-of-arts the value of k is applied on the basis of the sample size by using cross-validation method; yet, at the cost of computational exhaustion and delay that limits their efficacy for large non-linear datasets. Unlike classical method, we used a k-Tree learning concept which helped to perform learning for the different k values for the different samples. While performing training the k-Tree method initially performs learning for the optimal value of k for the overall data samples by using sparse reconstruction method. Subsequently, it constructed a DT say, k-Tree by applying training samples and the learned values of k. While performing testing, it yields the k-value for each sample, which is followed by the k-NN classification with the optimal k-value. Thus, the proposed k-NN classifier model was classified each review sentence as "Extremely Unsatisfied", "Unsatisfied", "Neutral", "Satisfied" and "Extremely Satisfied", and annotated them with the "1", "2", "3", "4" and "5", labels respectively.

Thus, applying above discussed method the different simulations were made to assess respective performance. The simulation results with allied inferences are given in the subsequent sections.

VI. RESULTS AND DISCUSSION

This paper contributed a novel and robust outlier resilient semantic featured deep driven sentiment analysis model (ORDSAENet) was developed for educational sentiment analysis. Unlike classical NLP solutions, the ORDSAENet model emphasized on multiple aspects including data optimization, feature improvement and computation to achieve a reliable educational sentiment analysis and annotation. In sync with data heterogeneity and unstructured-ness over real-world data environment, ORDSAENet used varied pre-processing tasks, which helped improving data quality for further processing and learning. The pre-processed text sequences representing the student's reviews towards online courses, were tokenized by transforming each input review into corresponding set of tokens for each sentence. Subsequently, to retain intrinsic features, semantic features were obtained by transforming tokens into Word2Vec features. To implement Word2Vec CBOW method with the window size of five was taken into consideration. The semantic features were then processed for SMOTE-ENN for resampling that helped addressing class-imbalance problem. The resampled CBOW embedding matrix was then passed to the Bi-LSTM which extracted both local features as well as the global features (say, contextual features) to perform learning and classification. We implemented Bi-LSTM network with the ADAM non-linear optimizer. Moreover, the initial learning rate was fixed at 0.0001, while the batch size was defined as 32. The total number of iterations considered were 200, while to improve learning efficiency and hence classification accuracy five-fold cross-validation was performed. The different feature models and allied learning environment is given in the **Table 2.**

Table 2. Execution Models

Feature Model	Algorithm
Word2Vec CBOW+SMOTE-ENN	GNB SVM-RBF LOGR DT k-NN
Word2Vec CBOW+SMOTE-ENN+ BI-LSTM	Bi-LSTM (ADAM, Cascaded Cross Entropy)

The developed semantic analysis models (Table-II) were simulated over a central processing unit (CPU) armored with Intel Core i5 processor, with 8 GB memory and 3.2 GHz processing elements. The proposed model was developed by using Python

Notebook platform, where the algorithms were developed in Python language. More specifically, the Python libraries including NumPy, Keres, TensorFlow, Pandas, NLTK, etc. were taken into consideration. To alleviate the reliance on local processing components Google Collaboratory, often called Google CoLab was applied to simulate the model. In this work, the native processing setting was applied to perform simulation. To assess efficacy of the proposed ORDSAENet model, the statistical performance analysis was performed. To achieve it, the confusion metrics values were obtained in terms of true positive (TP), true negative (TN), false positive (FP) and False negative (FN). Thus, using these metrices the performance parameters were obtained as per **Table 3**.

Table 3. Performance Parameters

Parameter	Mathematical Expression	Definition
Accuracy	$\frac{(TN+TP)}{(TN+FN+FP+TP)}$	Signifies the proportion of class that were classified correctly over the total number of classes 9say reviews labelled correctly for their original class) inspected out of all modules.
Precision	$\frac{TP}{(TP+FP)}$	States the degree to which the repeated measurements under unchanged conditions show the same results.
Recall	$\frac{TP}{(TP+FN)}$	It indicates how many of the relevant items are to be identified.
F-Score	$2.(Recall.Precision)/(Recall + Precision)$	It combines the precision and recall numeric value to give a single score, which is defined as the harmonic mean of the recall and precision.

Since, in this work the different machine learning classifiers have been applied to perform educational sentiment annotations, and therefore we compared the performance with the different machine learning classifier including Naïve Bayes, SVM, DT, LR and k-NN. The overall performance characterization is done in terms of the intra-model assessment and the inter-model characterization, where the first examines the efficacy over the different machine learning algorithms. On the contrary, in inter-model assessment the performance by ORDSAENet is compared with other educational sentiment analysis approaches.

Intra-Model Characterization

In this section, the relative performance with the different machine learning classifiers including Naïve Bayes, SVM, k-NN, LR and DT is discussed. Here, the predominant motive is to assess the efficacy with the different machine classifier so as to identify the optimal classifier platform to achieve optimal sentiment annotation performance. The simulation results obtained are given in Table IV. As depicted in Table IV, the average sentiment annotation (classification) accuracy obtained by the GNB classifier is 92.31%, while SVM, k-NN, LR and DT algorithms exhibited (over the CBOW+SMOTE-ENN feature) the classification accuracy of 94.24%, 95.37%, 95.21% and 95.27%, respectively. On the contrary, our proposed ORDSAENet model achieved the average (over fivefold cross validation) sentiment classification accuracy of 95.87%, which is higher than other state-of-arts.

Though, the k-NN, DT and LR methods too has exhibited the average accuracy of 95.37%, 95.37% and 95.21%, correspondingly. Precision and recall being the efficacy over the different changed input patterns reveal that the proposed sentiment analysis solution (i.e., ORDSAENet) exhibits the highest precision and recall of 95.26% and 95.06%, respectively. The precision and recall by LR regression model were found to be 95.08% and 94.93%, correspondingly. The k-NN algorithm too exhibited the precision and recall of 94.43% and 95.71%, correspondingly.

Similarly, SVM with radial basis function (RBF) kernel exhibited the precision and recall of 94.51% and 95.63%, respectively. On the contrary, Naïve Bayes with Gaussian kernel function exhibited the precision of 94.31%, and recall of 95.05%. The F-Measure performance, which is derived by using precision and recall revealed that the Naïve Bayes model exhibited F-Measure of 94.67%. Similarly, the SVM, k-NN, LR and DT algorithms exhibited the F-Measure of 95.06%, 95.06%, 94.96% and 95.06%, correspondingly. Our proposed ORDSAENet model exhibited the highest F-Measure of 95.15%, signifying its superiority over the other state-of-arts.

Table 4. Intra-Model Assessment

Model	Accuracy (%)	Precision (%)	Recall (%)	F-Measure (%)
Naïve Bayes	92.31	94.31	95.05	94.67
SVM	94.24	94.51	95.63	95.06
k-NN	95.37	94.43	95.71	95.06
LR	95.21	95.08	94.93	94.96
DT	95.27	94.43	95.71	95.06
ORDSAENet	95.87	95.26	95.06	95.15

In sync with overall proposed pre-processing, tokenization, Word2Vec embedding, SMOTE-ENN resampling, Bi-LSTM learning model with ADAM optimizer and cross-entropy loss function exhibited the superior performance with the highest accuracy (95.87%), precision (95.26%), recall (95.06%) and F-Measure 95.15%. Taking this as contributed best performance we have assessed relative performance with the other state-of-arts. The relative performance with the existing educational sentiment analysis model is given in the subsequent section.

Inter-Model Characterization

This section discusses the relative efficacy of the proposed ORDSAENet educational sentiment analysis model over state-of-arts related to the educational sentiment analysis and classification. The authors [47] performed educational sentiment analysis and classification over the YouTube educational videos. They applied the different machine learning classifiers including RF, LR, NB and SVM algorithms. The depth assessment revealed that their proposed model with SVM classifier exhibited the highest sentiment classification accuracy (i.e., for the binary sentiment positive or negative annotation) of 92.82%, which is almost 3.05% lower than our proposed ORDSAENet model. Here, the role of Bi-LSTM along with the SMOTE-ENN resampled Word2Vec (CBOW) semantic feature can be hypothesized to have led better performance in ORDSAENet. The authors in [61] applied deep learning to perform educational sentiment analysis by exploiting student's feedback data. The authors applied CNN deep network which extracted local features from 181 undergraduate students pertaining to the attendance, grades etc. and labelled the as 1 and -1 for positive and negative, respectively.

They claimed to have average sentiment annotation performance with the F-Measure of 0.74. On the contrary, the same CNN with lecture related feedback in [63] exhibited the accuracy, precision, recall and F-Measure of 87.95%, 87%, 87% and 81%, respectively. The authors [64] used LSTM with multi-head attention layer armored with the Glove and Cove embedding layer to perform sentiment learning and classification. In other words, they applied Glove and Cove embedding layer to generate the embedding metrics from input students' feedback, which was fed as input to the LSTM for further extraction and learning. Noticeably, unlike [64], in ORDSAENet we applied Word2Vec with CBOW word-embedding that generated a low-dimensional embedding feature, which was later processed for SMOTE-ENN resampling followed by Bi-LSTM-based learning and classification (rather LSTM in [64]).

Here, the key motive was to extract both local as well as contextual feature, which can't be done by merely applying LSTM model. Moreover, the use of Bi-LSTM in ORDSAENet model addresses the problem of long-term dependency, which makes it more suitable over real-world data ecosystems. It makes ORDSAENet superior than the state-of-art [64] Similar to our proposed sentiment classification problem, the authors [64] to annotated each review sentence as positive, negative and neutral review. Though, the authors [64] simulated their model over Vietnamese students feedback corpus (UIT-VSFC) containing 16,175 review sentences. Unlike the state-of-arts [64], we trained over almost 1.07 lakh qualitative and quantitative reviews obtained towards online courses. The authors in [64] assessed performance with four different feature models including LSTM, LSTM+ attention (LSTM+ATT), Multi-head attention (MULTIHEAD ATT) and Fusion of the different feature model (Say, FUSION) [64]. The average accuracy, precision, recall and F-Measure by the different model are given in **Table V.**

The authors exploited features from the input review sentences, where the depth assessment with the RF classifier resulted in accuracy of 98.01% [57] and 96.99% [58]. The authors in [56] designed an ensemble learning approach encompassing LR, SVM and NB algorithms. The maximum voting ensemble concept trained over student's feedback data resulted in the accuracy of 90.32%, F-Measure 93.80% and recall of 90.86%. Despite this result, our proposed model seems to outperform the state-of-art [56]. In [65] BERT transform method was applied to learn over the student's text feedback data. More specifically, the authors proposed shallow BERT model which could perform on average 89.79% sentiment classification accuracy. To ensure higher reliability, the authors [67] performed two-fold cross validation to learn and classify sentiment classes in student's feedback related to the qualitative and quantitative feedback for post graduate study. The depth performance characterization revealed that the use of RF ensemble classifier can yield the accuracy and precision of 98.87% and 97.71%, correspondingly. Undeniably, this approach performed quantifiably better than the proposed methods; yet the approach employed failed to address the lack of long-term dependency, class-imbalance and local minima and convergence, where the authors itself [67] agreed that the quantitative rank provided by the students and faculties vary and possess class-imbalance problem. On the contrary, the proposed ORDSAENet addressed almost all computational complexity, challenges and feature constraints. It affirms superiority of the proposed model over the state-of-art [67].

Table 5. Inter-Model Characterization

Feature Model	Accuracy (%)	Precision (%)	Recall (%)	F-Measure (%)
[47]	92.82	-	-	-
[52]	-	-	-	94.00
[56]	90.32	-	90.86	93.80
[57]	98.01	-	-	99.43
[58]	96.99	-	-	88.72
CNN [61]	-	-	-	74.00
LSTM [63]	87.95	87.00	87.00	81.00
LSTM [64]	83.92	65.13	70.13	63.56
LSTM+ATT [64]	85.28	67.38	74.85	66.27
MULTIHEAD ATT [64]	88.00	71.49	82.93	71.33
FUSION [64]	91.06	75.60	87.51	76.90
S-BERT [65]	89.79	-	-	-
[67]	98.87	97.71	-	-
LSTM + Glove [68]	95.80	-	-	-
Multi-layered Aspect2Labels (A2L) [69]	91.30	-	-	-
Variable Stop-words Filtering (VSF)+ Global Feature Selection Scheme (VGFSS) [69]	97.00 (SVM) 93.00 (ANN)	-	-	-
Bi-LSTM [70]	-	-	-	93.34
ORDSAENet	95.87	95.26	95.06	95.15

In [68] The authors applied LSTM over Glove embedding metrics, where the highest accuracy obtained was 95.80%. Unlike [68], our proposed ORDSAENet model exhibited the accuracy of 95.87%, which is higher than the state-of-art [68]. Moreover, the ability to inculcate Word2Vec embedding in conjunction with SMOTE-ENN resampling and Bi-LSTM feature extraction and learning strengthened ORDSAENet to address varied challenges including data heterogeneity, unstructured-ness, convergence and class-imbalance. The authors [71] used Aspect2Labels with VGFSS mechanism that in conjunction with SVM and ANN classifiers exhibited the accuracy of 97% and 93%, respectively. Yet, the computational complexity and lack of ability to learn contextual features limits its efficacy towards a scalable and reliable educational sentiment annotation solution. To address aforesaid problem of the lack of contextual details and lack of long-term dependency, the authors [70] applied Bi-LSTM deep network [72] that exhibited the F-Measure of 93.34%. In comparison of the state-of-art [73], our proposed ORDSAENet achieved the F-Measure of 95.15%. The results reveal that the proposed ORDSAENet outperforms other state-of-arts.

In sync with the simulation results and allied inferences it can be confirmed that the use of different pre-processing functions be effective to address data heterogeneity and unstructured-ness problem of the real-time educational review datasets for computational efficient sentiment annotation solution. And therefore, the aforesaid pre-processing solutions can ensure a robust sentiment annotation model and therefore the answer for the RQ1 is found affirmative. In sync with [68][70], the response of the Word2Vec embedding with SMOTE-ENN resampling followed by Bi-LSTM feature extraction and learning can be effective towards educational sentiment analysis. Thus, the answers for RQ2 and RQ3 too are found affirmative. The other state-of-arts as well such as [68][70] too revealed that the semantic embedding method can yield better feature environment towards educational sentiment annotation solution. The answer for the research question RQ3 is found positive. Unlike any state-of-arts existing methods, which failed addressing class-imbalance problem, our proposed ORDSAENet model in conjunction with cross-correlation feature selection and Min-Max normalization yielded real-time educational sentiment analysis solution (RQ4). Eventually, the amalgamation of the different proposed approaches (RQ1-RQ4) can provide an optimal solution towards sentiment annotation tool (RQ5).

VII. CONCLUSION

This paper proposed an outlier resilient semantic featured deep driven sentiment analysis model (ORDSAENet) for educational sentiment analysis. As the name indicates, the ability to drop noise elements as well as the outlier created irrelevant text contents including the one with exceedingly large review text, URL, hashtag, missing terms etc. The overall pre-processing tasks including missing value removal, Unicode normalization, Emoji removal, Website link removal, removal of the words with numeric values, punctuations removal, lower case conversion, stop-word removal, lemmatization and tokenization converted raw text reviews into corresponding tokens, for further feature extraction and learning. To ensure optimal features towards accurate sentiment annotation, ORDSAENet applied Word2Vec CBOW embedding followed by SMOTE-ENN resampling, where the later alleviated any likelihood of class-imbalance, while the earlier (i.e., CBOW) provided low-dimensional but feature (information) rich data for further processing. The resampled word-embedding matrix was processed for Bi-LSTM feature extraction and learning that ensured training over the local as well as contextual features from each review sentence. Moreover, it helped addressing long-term dependency problem as well. The learning and classification by using categorical cross-entropy loss function with ADAM optimizer exhibited that the proposed ORDSAENet model can achieve the sentiment annotation accuracy of 95.87%, precision 95.26%, recall 95.06% and F-Measure of 95.15%. The depth assessment revealed that the training with Bi-LSTM over the SMOTE-ENN resampled CBOW embeddings performed superior over other ML-based solutions. It infers that the use of Bi-LSTM network retrieved both local as well as contextual features from the input sentences, which helped achieving better learning and hence higher accuracy. On the contrary, the ML-driven solutions merely exploited the local word-embedding information to train and classify and therefore underwent lower accuracy than the proposed ORDSAENet educational sentiment analysis solution. The relative performance comparison with other state-of-arts too confirmed that the proposed ORDSAENet model outperforms major approaches, where its robustness to address multiple real-time challenge make it more productive and scalable.

Undeniably, the proposed ORDSAENet model performed superior towards educational sentiment classification; yet, it lacked the ability to reduce insignificant feature before learning, which could have improved its computational efficacy towards real-world problem. Moreover, the results confirm that the ML-driven solutions too perform nearly same (though quantifiably lower) as ORDSAENet over SMOTE-ENN resampled CBOW embedding features. In future, the efforts can be made to apply other word-embedding solutions such as Glove, TF-IDF, Word2Vec n-Skip Gram (SKG), etc. so as to identify the best feature model to yield computationally efficient sentiment analysis solution. In addition, the other resampling methods like SMOTE-boundary line, SMOTE can be assessed for their relative efficacy. Feature selection methods too can be applied to achieve a computationally efficient solution. In the last few years, ensemble learning methods have performed superior over the classical base classifiers, and therefore in future other ML algorithms including ensemble learning classifiers can be examined for their relative efficacy. The assessment of the aforesaid solution can help achieving an optimal computing solution for educational sentiment analysis. It can be considered as the future scope for further study.

Data Availability

The Data used to support the findings of this study will be shared upon request.

Conflicts of Interests

The author(s) declare(s) that they have no conflicts of interest.

Funding

No funding was received to assist with the preparation of this manuscript.

Ethics Approval and Consent to Participate

The research has consent for Ethical Approval and Consent to participate.

Competing Interests

There are no competing interests.

Reference

[1]. M. Bansal, S. Verma, K. Vig, and K. Kakran, “Opinion Mining from Student Feedback Data Using Supervised Learning Algorithms,” Lecture Notes in Networks and Systems, pp. 411–418, 2022, doi: 10.1007/978-3-031-12413-6_32.

[2]. Ligthart, C. Catal, and B. Tekinerdogan, “Systematic reviews in sentiment analysis: a tertiary study,” Artificial Intelligence Review, vol. 54, no. 7, pp. 4997–5053, Mar. 2021, doi: 10.1007/s10462-021-09973-3.

[3]. Liu, “Sentiment Analysis,” Jun. 2015, doi: 10.1017/cbo9781139084789.

[4]. E. Cambria, “Affective Computing and Sentiment Analysis,” IEEE Intelligent Systems, vol. 31, no. 2, pp. 102–107, Mar. 2016, doi: 10.1109/mis.2016.31.

[5]. B. Liu, "Sentiment Analysis and Opinion Mining," Synthesis Lectures on Human Language Technologies, vol. 5, no. 1, pp. 1–167, May 2012, doi: 10.2200/s00416ed1v01y201204hlt016.
[6]. X. Tao, Y. Li, and R. Nayak, "A knowledge retrieval model using ontology mining and user profiling," Integrated Computer-Aided Engineering, vol. 15, no. 4, pp. 313–329, Jul. 2008, doi: 10.3233/ica-2008-15404.
[7]. X. Tao, T. B. Shaik, N. Higgins, R. Gururajan, and X. Zhou, "Remote Patient Monitoring Using Radio Frequency Identification (RFID) Technology and Machine Learning for Early Detection of Suicidal Behaviour in Mental Health Facilities," Sensors, vol. 21, no. 3, p. 776, Jan. 2021, doi: 10.3390/s21030776.
[8]. X. Tao, X. Zhou, J. Zhang, and J. Yong, "Sentiment Analysis for Depression Detection on Social Networks," Lecture Notes in Computer Science, pp. 807–810, 2016, doi: 10.1007/978-3-319-49586-6_59.
[9]. X. Zhou, X. Tao, J. Yong, and Z. Yang, "Sentiment analysis on tweets for social events," Proceedings of the 2013 IEEE 17th International Conference on Computer Supported Cooperative Work in Design (CSCWD), Jun. 2013, doi: 10.1109/cscwd.2013.6581022.
[10]. T. Shaik et al., "FedStack: Personalized activity monitoring using stacked federated learning," Knowledge-Based Systems, vol. 257, p. 109929, Dec. 2022, doi: 10.1016/j.knosys.2022.109929.
[11]. T. Shaik et al., "A Review of the Trends and Challenges in Adopting Natural Language Processing Methods for Education Feedback Analysis," IEEE Access, vol. 10, pp. 56720–56739, 2022, doi: 10.1109/access.2022.3177752.
[12]. T. Shaik, X. Tao, C. Dann, H. Xie, Y. Li, and L. Galligan, "Sentiment analysis and opinion mining on educational data: A survey," Natural Language Processing Journal, vol. 2, p. 100003, Mar. 2023, doi: 10.1016/j.nlp.2022.100003.
[13]. V. K. Yeruva, M. Chandrashekar, Y. Lee, J. Rydberg-Cox, V. Blanton, and N. A. Oyler, "Interpretation of Sentiment Analysis with Human-in-the-Loop," 2020 IEEE International Conference on Big Data (Big Data), Dec. 2020, doi: 10.1109/bigdata50022.2020.9378221.
[14]. M. Elfeky, T. S. Y. Masadeh, and M. Y. H. Elbyaly, "Advance organizers in flipped classroom via e-learning management system and the promotion of integrated science process skills," Thinking Skills and Creativity, vol. 35, p. 100622, Mar. 2020, doi: 10.1016/j.tsc.2019.100622.
[15]. L. McKinney, A. B. Burridge, M. M. Lee, G. V. Bourdeau, and M. Miller-Waters, "Incentivizing Full-Time Enrollment at Community Colleges: What Influences Students' Decision to Take More Courses?," Community College Review, vol. 50, no. 2, pp. 144–170, Feb. 2022, doi: 10.1177/00915521211061416.
[16]. H. Zhao, Z. Liu, X. Yao, and Q. Yang, "A machine learning-based sentiment analysis of online product reviews with a novel term weighting and feature selection approach," Information Processing & Management, vol. 58, no. 5, p. 102656, Sep. 2021, doi: 10.1016/j.ipm.2021.102656.
[17]. Y. Zhang, J. Wang, and X. Zhang, "Conciseness is better: Recurrent attention LSTM model for document-level sentiment analysis," Neurocomputing, vol. 462, pp. 101–112, Oct. 2021, doi: 10.1016/j.neucom.2021.07.072.
[18]. Z. Liu, C. Yang, S. Rüdian, S. Liu, L. Zhao, and T. Wang, "Temporal emotion-aspect modeling for discovering what students are concerned about in online course forums," Interactive Learning Environments, vol. 27, no. 5–6, pp. 598–627, Apr. 2019, doi: 10.1080/10494820.2019.1610449.
[19]. J. J. Zhu, Y.-C. Chang, C.-H. Ku, S. Y. Li, and C.-J. Chen, "Online critical review classification in response strategy and service provider rating: Algorithms from heuristic processing, sentiment analysis to deep learning," Journal of Business Research, vol. 129, pp. 860–877, May 2021, doi: 10.1016/j.jbusres.2020.11.007.
[20]. F. A. Acheampong, H. Nunoo-Mensah, and W. Chen, "Transformer models for text-based emotion detection: a review of BERT-based approaches," Artificial Intelligence Review, vol. 54, no. 8, pp. 5789–5829, Feb. 2021, doi: 10.1007/s10462-021-09958-2.
[21]. V. Kuleto et al., "Exploring Opportunities and Challenges of Artificial Intelligence and Machine Learning in Higher Education Institutions," Sustainability, vol. 13, no. 18, p. 10424, Sep. 2021, doi: 10.3390/su131810424.
[22]. F. S. Dolianiti, D. Iakovakis, S. B. Dias, S. Hadjileontiadou, J. A. Diniz, and L. Hadjileontiadis, "Sentiment Analysis Techniques and Applications in Education: A Survey," Technology and Innovation in Learning, Teaching and Education, pp. 412–427, 2019, doi: 10.1007/978-3-030-20954-4_31.
[23]. R. Catelli, S. Pelosi, and M. Esposito, "Lexicon-Based vs. Bert-Based Sentiment Analysis: A Comparative Study in Italian," Electronics, vol. 11, no. 3, p. 374, Jan. 2022, doi: 10.3390/electronics11030374.
[24]. V. Hatzivassiloglou and K. R. McKeown, "Predicting the semantic orientation of adjectives," Proceedings of the 35th annual meeting on Association for Computational Linguistics -, 1997, doi: 10.3115/976909.979640.
[25]. Taboada, M., Gillies, M.A., McFetridge, Sentiment classification techniques for tracking literary reputation. In: LREC Workshop: To-Wards Computational Models of Literary Analysis. Citeseer, pp. 36–43. P., 2006.
[26]. Benamara, F., Cesarano, C., Picariello, A., Recupero, D.R., Subrahmanian, V.S., Sentiment analysis: Adjectives and adverbs are better than adjectives alone. ICWSM 7, 203–206.
[27]. Vermeij, M., The orientation of user opinions through adverbs, verbs and nouns. In: 3rd Twente Student Conference on IT, Enschede June. Citeseer. 2005.
[28]. Neviarouskaya, H. Prendinger, and M. Ishizuka, "Compositionality Principle in Recognition of Fine-Grained Emotions from Text," Proceedings of the International AAAI Conference on Web and Social Media, vol. 3, no. 1, pp. 278–281, Mar. 2009, doi: 10.1609/icwsm.v3i1.13987.
[29]. Baccianella, S., Esuli, A., Sebastiani, F., Sentiwordnet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining, 2010.
[30]. B. Maegaard, K. Choukri, N. Calzolari, and J. Odijk, "ELRA – European Language Resources Association-Background, Recent Developments and Future Perspectives," Language Resources and Evaluation, vol. 39, no. 1, pp. 9–23, Feb. 2005, doi: 10.1007/s10579-005-2692-5.
[31]. T. Wilson et al., "OpinionFinder," Proceedings of HLT/EMNLP on Interactive Demonstrations -, 2005, doi: 10.3115/1225733.1225751.
[32]. D. Boullier and A. Lohard, "Opinion mining et Sentiment analysis : résumé," Opinion mining et Sentiment analysis, 2012, doi: 10.4000/books.oep.223.
[33]. T. Wilson, J. Wiebe, and P. Hoffmann, "Recognizing contextual polarity in phrase-level sentiment analysis," Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing - HLT '05, 2005, doi: 10.3115/1220575.1220619.
[34]. Nielsen, F.Å, A new anew: Evaluation of a word list for sentiment analysis in microblogs. arXiv preprint arXiv:1103.2903, 2011.
[35]. Neviarouskaya, H. Prendinger, and M. Ishizuka, "SentiFul: Generating a reliable lexicon for sentiment analysis," 2009 3rd International Conference on Affective Computing and Intelligent Interaction and Workshops, Sep. 2009, doi: 10.1109/acii.2009.5349575.
[36]. Hutto and E. Gilbert, "VADER: A Parsimonious Rule-Based Model for Sentiment Analysis of Social Media Text," Proceedings of the International AAAI Conference on Web and Social Media, vol. 8, no. 1, pp. 216–225, May 2014, doi: 10.1609/icwsm.v8i1.14550.
[37]. "RAND HRS Longitudinal File 2018 (V2) Documentation: Includes 1992-2018 (Final Release)," 2022, doi: 10.7249/tla2097-1.

[38]. Tzacheva and A. Easwaran, “Emotion Detection and Opinion Mining from Student Comments for Teaching Innovation Assessment,” International Journal of Education (IJE), vol. 09, no. 02, pp. 21–32, Jun. 2021, doi: 10.5121/ije2021.9203.
[39]. Preethi P, Yogapriya J, Nivetha L and R.Madanachitran, “Transfer Driven Ensemble Learning Approach using ROI Pooling CNN For Enhanced Breast Cancer Diagnosis, Journal of Machine and Computing, vol.3, no.3, pp. 297-311, July 2023. doi: 10.53759/7669/jmc202303026.
[40]. Ortony, A., Turner, T.J.,. What’s basic about basic emotions? Psychol. Rev. 97 (3), 315, 1990.
[41]. X. Zhu, S. Kiritchenko, and S. Mohammad, “NRC-Canada-2014: Recent Improvements in the Sentiment Analysis of Tweets,” Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014), 2014, doi: 10.3115/v1/s14-2077.
[42]. M. L. Maher, C. Latulipe, H. Lipford, and A. Rorrer, “Flipped Classroom Strategies for CS Education,” Proceedings of the 46th ACM Technical Symposium on Computer Science Education, Feb. 2015, doi: 10.1145/2676723.2677252.
[43]. C. Latulipe, N. B. Long, and C. E. Seminario, “Structuring Flipped Classes with Lightweight Teams and Gamification,” Proceedings of the 46th ACM Technical Symposium on Computer Science Education, Feb. 2015, doi: 10.1145/2676723.2677240.
[44]. S. MacNeil, C. Latulipe, B. Long, and A. Yadav, “Exploring Lightweight Teams in a Distributed Learning Environment,” Proceedings of the 47th ACM Technical Symposium on Computing Science Education, Feb. 2016, doi: 10.1145/2839509.2844577.
[45]. J. Melba Rosalind and S. Suguna, “Predicting Students’ Satisfaction Towards Online Courses Using Aspect-Based Sentiment Analysis,” IFIP Advances in Information and Communication Technology, pp. 20–35, 2022, doi: 10.1007/978-3-031-11633-9_3.
[46]. F. Alqasemi, A. Abdelwahab, and H. Abdelkader, “Constructing automatic domain-specific sentiment lexicon using KNN search via terms discrimination vectors,” International Journal of Computers and Applications, vol. 41, no. 2, pp. 129–139, Dec. 2017, doi: 10.1080/1206212x.2017.1409477.
[47]. U. Ö. Osmanoğlu, O. N. Atak, K. Çağlar, H. Kayhan, And T. Can, “Sentiment Analysis for Distance Education Course Materials: A Machine Learning Approach,” Journal of Educational Technology and Online Learning, vol. 3, no. 1, pp. 31–48, Jan. 2020, doi: 10.31681/jetol.663733.
[48]. H. H. Lwin, S. Oo, K. Z. Ye, K. Kyaw Lin, W. P. Aung, and P. Paing Ko, “Feedback Analysis in Outcome Base Education Using Machine Learning,” 2020 17th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON), Jun. 2020, doi: 10.1109/ecti-con49241.2020.9158328.
[49]. R. Faizi, “A Sentiment-Based Approach to Predict Learners’ Perceptions Towards YouTube Educational Videos,” Lecture Notes in Networks and Systems, pp. 549–556, 2022, doi: 10.1007/978-3-030-96299-9_52.
[50]. Kaur, C., Boush, M.S.A., Hassen, S.M., Hakami, W.A., Ab-dalraheem, M.H.O., Galam, N.M., Hadi, N.A., Hadi, N.A., Benjeed, A.O.S., 2022. Incorporating sentimental analysis into development of a hybrid classification model. Int. J. Health Sci. 6 (S1), 1709–1720.
[51]. F. S. Dolianiti, D. Iakovakis, S. B. Dias, S. Hadjileontiadou, J. A. Diniz, and L. Hadjileontiadis, “Sentiment Analysis Techniques and Applications in Education: A Survey,” Technology and Innovation in Learning, Teaching and Education, pp. 412–427, 2019, doi: 10.1007/978-3-030-20954-4_31.
[52]. M. Ahmad, S. Aftab, M. Salman, and N. Hameed, “Sentiment Analysis using SVM: A Systematic Literature Review,” International Journal of Advanced Computer Science and Applications, vol. 9, no. 2, 2018, doi: 10.14569/ijacsa.2018.090226.
[53]. M. Sivakumar and U. S. Reddy, “Aspect based sentiment analysis of students opinion using machine learning techniques,” 2017 International Conference on Inventive Computing and Informatics (ICICI), Nov. 2017, doi: 10.1109/icici.2017.8365231.
[54]. N. Nikolić, O. Grljević, and A. Kovačević, “Aspect-based sentiment analysis of reviews in the domain of higher education,” The Electronic Library, vol. 38, no. 1, pp. 44–64, Feb. 2020, doi: 10.1108/el-06-2019-0140.
[55]. Yang, R., Machine learning and deep learning for sentiment analysis over students’ reviews: An overview study, 2021.
[56]. J. Ding, H. Sun, X. Wang, and X. Liu, “Entity-level sentiment analysis of issue comments,” Proceedings of the 3rd International Workshop on Emotion Awareness in Software Engineering, Jun. 2018, doi: 10.1145/3194932.3194935.
[57]. H. Li and W. Lu, “Learning Latent Sentiment Scopes for Entity-Level Sentiment Analysis,” Proceedings of the AAAI Conference on Artificial Intelligence, vol. 31, no. 1, Feb. 2017, doi: 10.1609/aaai.v31i1.11016.
[58]. Kalaiselvi Balaraman and S.P.Angelin Claret, “Hybrid Resnet and Bidirectional LSTM-Based Deep Learning Model for Cardiovascular Disease Detection Using PPG Signals, Journal of Machine and Computing, vol.3, no.3, pp. 351-359, July 2023. doi: 10.53759/7669/jmc202303030.
[59]. M. Edalati, A. S. Imran, Z. Kastrati, and S. M. Daudpota, “The Potential of Machine Learning Algorithms for Sentiment Classification of Students’ Feedback on MOOC,” Intelligent Systems and Applications, pp. 11–22, Aug. 2021, doi: 10.1007/978-3-030-82199-9_2.
[60]. D. Wehbe, A. Alhammadi, H. Almaskari, K. Alsereidi, and H. Ismail, “UAE e-Learning Sentiment Analysis Framework,” ArabWIC 2021: The 7th Annual International Conference on Arab Women in Computing in Conjunction with the 2nd Forum of Women in Research, Sharjah, UAE, Aug. 2021, doi: 10.1145/3485557.3485570.
[61]. Deepa Devasenapathy, Vidhya K, Anna Alphy, Finney Daniel Shadrach, Kathirvelu M and Jayaraj Velusamy, “Kidney Impairment Prediction Due to Diabetes Using Extended Ensemble Learning Machine Algorithm, Journal of Machine and Computing, vol.3, no.3, pp. 312-325, July 2023. doi: 10.53759/7669/jmc202303027.
[62]. N. Dehbozorgi and D. P. Mohandoss, “Aspect-Based Emotion Analysis on Speech for Predicting Performance in Collaborative Learning,” 2021 IEEE Frontiers in Education Conference (FIE), Oct. 2021, doi: 10.1109/fie49875.2021.9637330.
[63]. Z. Kastrati, A. S. Imran, and A. Kurti, “Weakly Supervised Framework for Aspect-Based Sentiment Analysis on Students’ Reviews of MOOCs,” IEEE Access, vol. 8, pp. 106799–106810, 2020, doi: 10.1109/access.2020.3000739.
[64]. L. C. Yu et al., “Improving early prediction of academic failure using sentiment analysis on self-evaluated comments,” Journal of Computer Assisted Learning, vol. 34, no. 4, pp. 358–365, Mar. 2018, doi: 10.1111/jcal.12247.
[65]. Lang, P., Behavioral treatment and bio-behavioral assessment: Computer applications. Technol. Mental Health Care Deliv. Syst, 119–137, 1980.
[66]. E. Sutoyo, A. Almaarif, and I. T. R. Yanto, “Sentiment Analysis of Student Evaluations of Teaching Using Deep Learning Approach,” Lecture Notes in Networks and Systems, pp. 272–281, 2021, doi: 10.1007/978-3-030-80216-5_20.
[67]. Arulkumar V, Vinod D, Devipriya A, Chemmalar Selvi G, Surendran S and Mohammad Arif, “Monitoring and Recognition of Heart Health using Heartbeat Classification with Deep Learning and IoT, Journal of Machine and Computing, vol.3, no.3, pp. 326-339, July 2023. doi: 10.53759/7669/jmc202303028.
[68]. Y. Dyulicheva, “Learning Analytics in MOOCs as an Instrument for Measuring Math Anxiety,” Voprosy Obrazovaniya / Educational Studies Moscow, no. 4, pp. 243–265, Dec. 2021, doi: 10.17323/1814-9545-2021-4-243-265.

[69]. X. Li, H. Zhang, Y. Ouyang, X. Zhang, and W. Rong, "A Shallow BERT-CNN Model for Sentiment Analysis on MOOCs Comments," 2019 IEEE International Conference on Engineering, Technology and Education (TALE), Dec. 2019, doi: 10.1109/tale48000.2019.9225993.
[70]. D. Pramod, S. Vijayakumar Bharathi, and R. Raman, "Faculty Effectiveness Prediction using Machine Learning and Text Analytics," 2022 IEEE Technology and Engineering Management Conference (TEMSCON EUROPE), Apr. 2022, doi: 10.1109/temsconeurope54743.2022.9801997.
[71]. ONAN, "Sentiment analysis on massive open online course evaluations: A text mining and deep learning approach," Computer Applications in Engineering Education, vol. 29, no. 3, pp. 572–589, May 2020, doi: 10.1002/cae.22253.
[72]. S. Hussain et al., "Aspect2Labels: A novelistic decision support system for higher educational institutions by using multi-layer topic modelling approach," Expert Systems with Applications, vol. 209, p. 118119, Dec. 2022, doi: 10.1016/j.eswa.2022.118119.
[73]. P. K. Singh and S. Paul, "Deep Learning Approach for Negation Handling in Sentiment Analysis," IEEE Access, vol. 9, pp. 102579–102592, 2021, doi: 10.1109/access.2021.3095412.