JTE HCMUTE
ISSN: 2615-9740

JOURNAL OF TECHNICAL EDUCATION SCIENCE
Ho Chi Minh City University of Technology and Education
Website: https://jte.edu.vn
Email: jte@hcmute.edu.vn

Sentiment Prediction Based on Deep Learning for Intelligent E-Learning Systems

Anh-Tu Phuong Nguyen, Duc-Khai Duong, Van-Dung Hoang*
Ho Chi Minh City University of Technology and Education, Vietnam

*Corresponding author. *Email:* dunghv@hcmute.edu.vn

ARTICLE INFO

Received:	27/09/2024
Revised:	04/11/2024
Accepted:	05/12/2024
Published:	28/08/2025

KEYWORDS

BERT model;
Feedback information;
Intelligent e-learning systems;
Machine learning;
RoBERTa;
Sentiment analysis.

ABSTRACT

Sentiment analysis on feedback information plays a crucial role in intelligent e-learning systems (IES). To enhance customer service quality through reviews, predicting sentiment polarity is crucial. This paper focuses on integrating sentiment analysis into an online programming course in intelligent e-learning systems to better understand the challenges faced by students. E-learning platforms, typically web-based, are vital for reflecting users' experiences, emotions, and purchase intentions. Recently, sentiment analysis systems have frequently been used on popular platforms like Facebook and Twitter, helping companies identify potential customers across different segments. Utilizing sentiment analysis assists in making informed product adjustments to meet user satisfaction. Our approach employs modern frameworks to build deep learning models, specifically leveraging the pre-trained RoBERTa model. We integrate this into a website application with a Django back-end, facilitating an easy implementation of RoBERTa into an API that communicates with Next.js, a cutting-edge framework for full-stack developers. This combination ensures optimal performance for the user experience and scalability of the project. Experimental results demonstrate that incorporating sentiment analysis into websites significantly enhances trust among users in e-learning systems.

Doi: https://doi.org/10.54644/jte.2025.1661

1. Introduction

Sentiment analysis is the process of using natural language processing techniques, text analysis, and machine learning to identify, extract, and classify opinions or emotions expressed in text. The goal of sentiment analysis is to better understand the attitudes, emotions, and viewpoints of users or customers toward a specific topic, product, service, or event. This process involves collecting data from various sources such as social media, product reviews, and customer feedback; preprocessing text data by cleaning and standardizing it; extracting features from text using methods like bag-of-words, TFIDF, or word embedding; and finally, classifying sentiment using machine learning models or dictionaries. Popular models include Naive Bayes, SVM, and neural network models like LSTM, CNN, or transformer models such as BERT. Sentiment analysis has several important applications including improving customer service, managing brand reputation, market research, measuring public reactions to political and social events, and predicting market trends in financial sectors. It is a powerful tool that helps organizations gain deeper insights into customer perceptions and attitudes, thereby informing strategic decisions and enhancing customer experience. Nowadays, modern e-commerce platforms should meet higher demands from both customers and businesses. There are some studies on applying sentiment analysis in e-commerce [1]–[3], showing practical and beneficial results with relatively high accuracy and efficiency. The application of sentiment analysis in e-commerce websites is crucial to gain a competitive edge.

When Fine-tuning RoBERTa for sentiment analysis, gradient clipping has many disadvantages despite being helpful in minimizing expanding gradients. By overly constraining the gradients, it might hinder the model's ability to learn and result in subpar performance since it fails to capture subtleties in language that are crucial for precise emotion prediction. In situations when immediate insights are required, this strategy may potentially slow down the convergence process, leading to longer training duration and greater computing costs. Furthermore, gradient clipping might disguise underlying training errors and obfuscate model interpretability, making it more difficult to identify and fix faults. All of these elements work together to compromise the effectiveness and dependability of sentiment analysis with RoBERTa.

We can use a few tactical techniques to improve the sentiment analysis Fine-tuning of RoBERTa and solve the gradient clipping drawbacks. Gradient norm scaling, which modifies the entire gradient vector to maintain a constant norm and preserves crucial directional information while avoiding excessive growth, is one efficient technique that may be substituted for hard clipping. Additionally, learning efficiency may be greatly increased by utilizing adaptive gradient techniques like AdamW. In order to help the model learn more efficiently without the need for restricted clipping, AdamW modifies the learning rates for each parameter depending on previous gradients. By combining these strategies with meticulous hyperparameter adjustment and training dynamics monitoring, RoBERTa's sentiment analysis performance and dependability may be improved, guaranteeing that it catches the complex patterns seen in natural language.

In this paper, we focus on developing a sentiment analysis model that leverages the state-of-the-art RoBERTa model specifically for the e-learning sector - an e-commerce website that sells courses, an area where such advanced models have been underutilized. This integration allows businesses to discern which courses are popular and best-selling, while also enabling course owners to evaluate the quality of their lectures based on student reviews. To construct the model, we trained it on a prepared dataset. Through the preprocessing stages, we opted for the RoBERTa pre-trained model to extract language in a manner conducive to machine learning and comprehension. During the training process, optimization techniques were employed to ensure optimal outcomes. Upon model completion, we will integrate it into the e-commerce website using modern libraries, frameworks, and tools, specifically leveraging Django for server-side development and Next.js for client-side implementation.

2. Related Work

There are many studies on sentiment analysis and applying RoBERTa or Bert in deep learning models. We've reviewed and incorporated some of the studies' findings to optimize our model. Two studies developed Bert models using data from Twitter [4], [5]. The first focused on sentiment analysis while the latter focused on author profiling. Various preprocessing techniques were tested. The results were quite mixed, the sentiment analysis model performed best with an intelligent preprocessor but the author profiling model performed best without one. In the first study, the difference between using a preprocessor and not using is 3.5%

Furthermore, Bert pretrains deep bidirectional representation from text data by joint conditioning on both the left and right context in all layers. The Fine-tuning for Bert can be done easily with one additional output layer but the model performance for NLP tasks is still excellent.

In a study using similar data to ours, Muhammad Bilal's and Abdulwahab Ali Almazroi's research tested the effectiveness of the Fine-tuned Bert model compared to other classifiers regarding Online Customer Review Classification [6]. The experiment is to train various models, including Bert, to classify the helpful and unhelpful reviews from Yelp a popular website for making reservations. The finding was Bert models performed better than all other classifiers (KNN, NB, SVM) and the best-performed Bert model has a sequence length of 320. The authors stated that one of the limitations of the research was not using other variants of Bert, including RoBERTa. However, another study has compared the performance of different types of Bert models. There were nine models in total; these models were trained on a COVID-19 literature dataset to classify fake news and real news. Overall, BERT-base and BERT-large have the same highest performance (99.71%), followed by RoBERTa base (99.6%). In our experiment, we chose a RoBERTa base to handle sentiment analysis tasks.

Sentiment analysis is a powerful tool in the field of e-commerce, helping businesses better understand customer perceptions of their products and services. This has been practically demonstrated in research on measuring e-commerce service quality using sentiment analysis with the Naïve Bayes Classification method [1]. Moreover, major companies today leverage sentiment analysis to optimize their marketing strategies. For example, Alibaba and Nike use sentiment analysis to assess consumer feedback from reviews on their websites and social media platforms. This data-driven approach allows them to study market trends and fine-tune their marketing campaigns effectively. Therefore, in this study, we propose an e-commerce website for online courses that applies sentiment analysis to evaluate student feedback based on their reviews.

3. Research Approaches

A. Text tokenizing methodology

Text is tokenized into smaller pieces, and RoBERTa processes these simultaneously to understand word meanings in context, using an "attention" mechanism to focus on relevant parts. Words are represented as numerical vectors called embeddings [4]. Although some different techniques can be applied using the RoBERTa model for preprocessing, when no preprocessing technique is applied, it is also able to bring the best results in some special cases [5].

Since we use the RoBERTa model, BertTokenizer serves as an intelligent bridge, converting sentences and paragraphs into a format RoBERTa can process. It encodes text into tokens and adds special tokens like [CLS] and [SEP], optimizing text encoding for increased speed and efficiency. This process is illustrated in Fig. 1.

TensorDataset, from the torch library, creates datasets for PyTorch by organizing data into tensors that can be indexed and passed into a model. The encoding step converts tokenized text into numerical representations, creating "inputids" and "attention masks" to construct a model-ready dataset. Attention masks help the model focus on important words and ignore padding, enhancing learning efficiency [7]. These numerical representations, capturing both semantics and syntax, are crucial for the model to understand language complexities and interpret sentiment accurately.

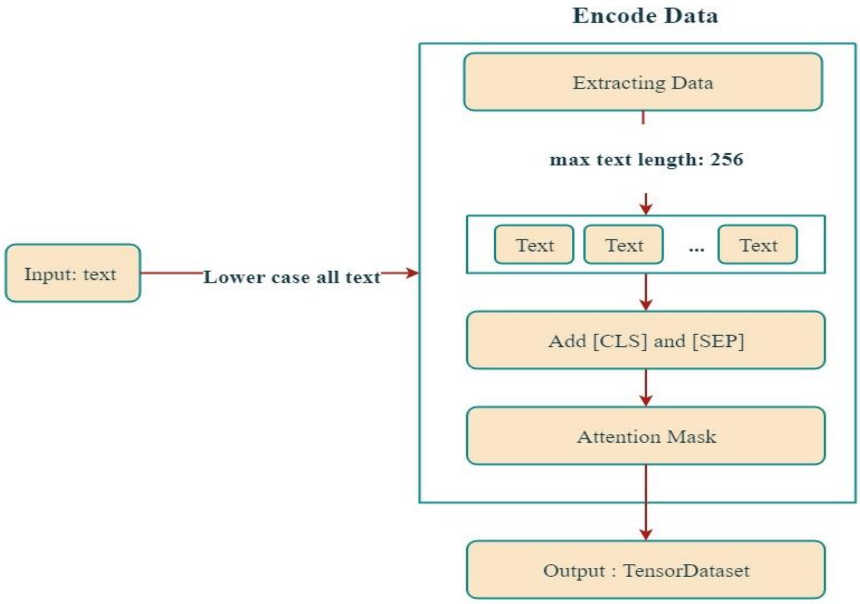

Figure 1. *Data preprocessing and tokenizing.*

B. Fine-tuning RoBERTa methodology

The process of Fine-tuning RoBERTa for sentiment analysis starts by leveraging the extensive linguistic knowledge the model has acquired during its pretraining phase. RoBERTa is built on the Transformer architecture, a powerful neural network architecture that enables the model to understand the relationships between words in a sentence, regardless of their distance. This is particularly important in sentiment analysis, where the meaning of a word can change depending on its surrounding context.

To adapt RoBERTa for the task of sentiment analysis, a classification layer is added on top of the model's architecture. This layer is responsible for predicting the sentiment of an input text segment, typically classifying it as positive, negative, or neutral. The Fine-tuning process involves retraining the entire model, including the new classification layer, on a labeled sentiment analysis dataset. The Fine-tuning processing is illustrated in Fig. 2, during which the model's weights are adjusted to optimize its sentiment prediction capability.

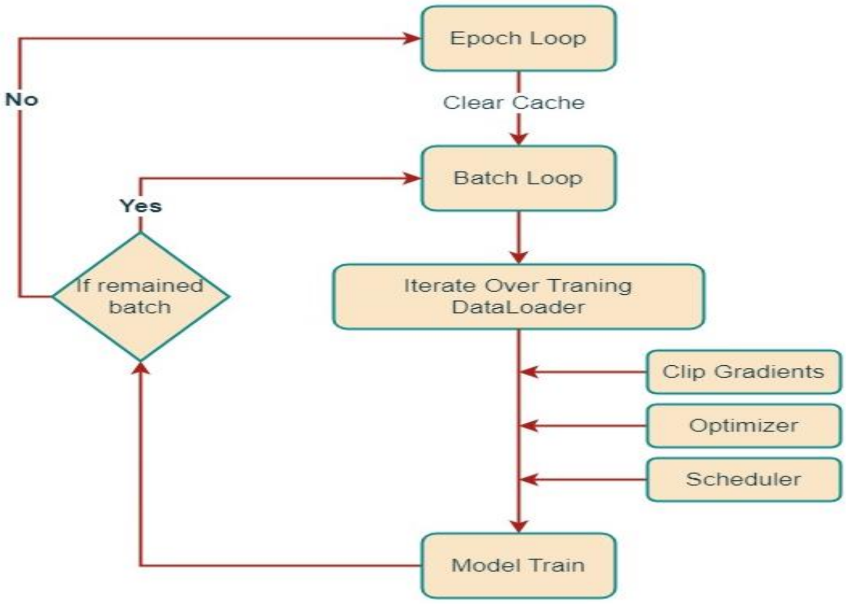

Figure 2. *Finetuning progress*

The training loop is a critical part of the Fine-tuning process. In each loop, a batch of data is fed into the model. The model then computes the sentiment predictions for each text segment in the batch and compares these predictions to the actual labels. The difference between the predictions and the actual labels is measured using a loss function. The goal of the training process is to minimize this loss function.

To minimize the loss function, optimization techniques such as Adam or SGD are used. These techniques calculate the gradient of the loss function with respect to each weight in the model and update the weights in a direction that reduces the loss function. This process is repeated over many training loops until the model achieves good performance on the training dataset. AdamW is a great optimizer that works well with Bert [8]–[10]

The model's effectiveness is judged based on how well it can predict the sentiment of a course feedback text. After the model receives and evaluates the input course feedback text, it will label the text as one of the following tones Angry, Disappointed, Neutral, Satisfied, or Happy.

We measured the robustness of the model using weighted f1-score, primally accuracy as it is a popular evaluation method, many other researchers have also applied this metric when evaluating Bert models [6], [11]. After the model analyzes the review text and predicts the tone of it, the prediction is marked as false positive (FP), true positive (TP), false negative (FN), or true negative (TN). The accuracy score is calculated using the total number of FP, TP, FN, and TN predictions to show how well the model can find out the correct emotions.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

4. Experiments and Evaluations

A. Dataset processing

Load Data to Analysis Coursera is a massive open online course (MOOC) provider and it hosts many courses from top universities in different subjects. The Course Reviews dataset on Coursera contains over 100 thousand reviews posted by students and participants. This dataset is available on Kaggle[1]. First, to fit with our purpose, we need to modify certain fields in the dataset. In this case, we

[1] https://www.kaggle.com/datasets/septa97/100k-courseras-course-reviews-dataset

will need only ID, Score, and Text. The score label is the rating of the review, ranging from 1 to 5. Moreover, we also need to handle some null values that are contained in the dataset to prevent unexpected errors. Table 1 depicts the overview of the dataset.

Training dataset processing About data splitting, the data sample is often divided into two sections, a training set for model training and a testing set for model evaluation.

Table 1. *Dataset Overview*

Reviews	Reviews by course
100,038 unique values	123,243 unique values

Many researchers have proposed a ratio of 70/30 or 80/20 (training/testing data) for producing datasets in landslide susceptibility problems [12], [13]. Regarding studies on estimating the course review using ML algorithms, previous works mainly used ratios of 75/25, and 80/20 (training/testing) for generating datasets [14], [15]. In general, the training set size plays an important role in the prediction ability of the machine learning models [16]. For this work, we decided to split the data into train and test sets with a ratio of 85/15.

Data preprocessing is crucial before training a model, as different strategies yield different results. Since we use the RoBERTa model, BertTokenizer serves as an intelligent bridge, converting sentences and paragraphs into a format RoBERTa can process. It encodes text into tokens and adds special tokens like [CLS] and [SEP], optimizing text encoding for increased speed and efficiency. TensorDataset, from the torch library, creates datasets for PyTorch by organizing data into tensors that can be indexed and passed into a model. The encoding step converts tokenized text into numerical representations, creating “inputids” and “attention masks” to construct a model-ready dataset. Attention masks help the model focus on important words and ignore padding, enhancing learning efficiency [7]. These numerical representations, capturing both semantics and syntax, are crucial for the model to understand language complexities and interpret sentiment accurately.

B. Experimental setting

The original dataset was highly imbalanced, with a predominance of “happy” and “satisfied” labels. To avoid bias in the model towards the more frequent emotion classes, a balancing procedure was implemented. Firstly, we determined the minimum sample size for each class. Subsequently, for classes with more samples than the target size, random down-sampling was performed without replacement to achieve the data distribution in Fig. 3.

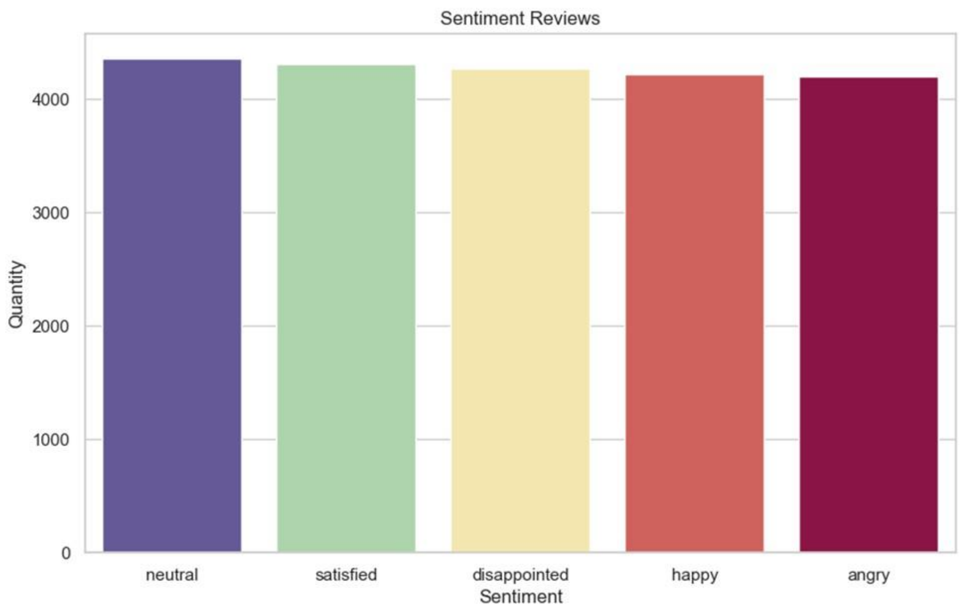

Figure 3. *Dataset labels overview*

JTE HCMUTE
ISSN: 2615-9740
JOURNAL OF TECHNICAL EDUCATION SCIENCE
Ho Chi Minh City University of Technology and Education
Website: https://jte.edu.vn
Email: jte@hcmute.edu.vn

Next, we split the data into training and test sets with a ratio of 85/15 while maintaining the class distribution during the data split. This ensures a fair distribution of data between the training and testing sets while maintaining a sufficiently large training set for the model.

We applied the RoBERTa model for sequence classification. For the data processing and construction process, the configured hyperparameter values for the transformers are shown in Table 2.

Table 2. *Hyperparameter values for Transformers*

Batch size	Epochs	Learning rate	Epsilon value	Optimizer
17	77	2e-5	1e-8	AdamW

The base model we performed tuning is uncased RoBERTa. We used PyTorch (a popular framework for building deep learning models) to simplify the tuning process and maximize the computational resources by utilizing the computer's GPU. We selected AdamW as the optimizer since it improves convergence by incorporating weight decay to prevent overfitting and it works well with Bert [8]–[10]. To adjust the learning rate during training, we applied the get_linear_schedule_with_warmup scheduler, which linearly increases the learning rate during a warmup phase and then decays it linearly for the rest of the training. This helps avoid instabilities at the start of training and improves generalization by gradually reducing the learning rate. The CrossEntropyLoss function was selected as it is a popular loss function for classification tasks, measuring the error between predicted class probabilities and actual labels. To boost performance and reduce memory consumption, we employed GradScaler for mixed precision training, which not only speeds up training but also prevents numerical instability by scaling gradients dynamically. Additionally, we set the gradient accumulation step to 1, meaning the optimizer updates the model parameters after every batch for faster training time. Table 3 shows the parameter values for the tuning process.

Table 3. *Tuning model's parameters*

Base model	Optimizer	Scheduler	Lost function	Scaler	Gradient accumulation step
Uncased RoBERTa	AdamW	get_linear_schedule_with_warmup	CrossEntropyLoss	GradScaler	1

C. Experimental results

Sentiment analysis results: In our experiment, we trained the models on 20k and 60 datasets while maintaining a balanced distribution between the data labels in order to compare the models' performance on datasets with different sizes. Additionally, the models were trained for 77 epochs and the following results are from the last 7 epochs. Generally, the model trained on the 60k dataset performed better with lower training and validation loss and a higher F1 score. The training losses of the model trained on the 20k and 60k datasets decrease over epochs are presented in Fig. 4. They are 0.37 and 0.24 after the first epoch, and 0.06 and 0.04 after the seventh epoch respectively. Similarly, the validation losses show the same trends. In Fig. 5, the validation losses on the 20k and 60k datasets start at 0.67 and 0.47 and decrease to 0.35 and 0.28 respectively on the seventh epoch. Finally, the F1 scores of the 20k and 60k datasets fluctuate around 0.885 and 0.930, respectively, as shown in Fig. 6.

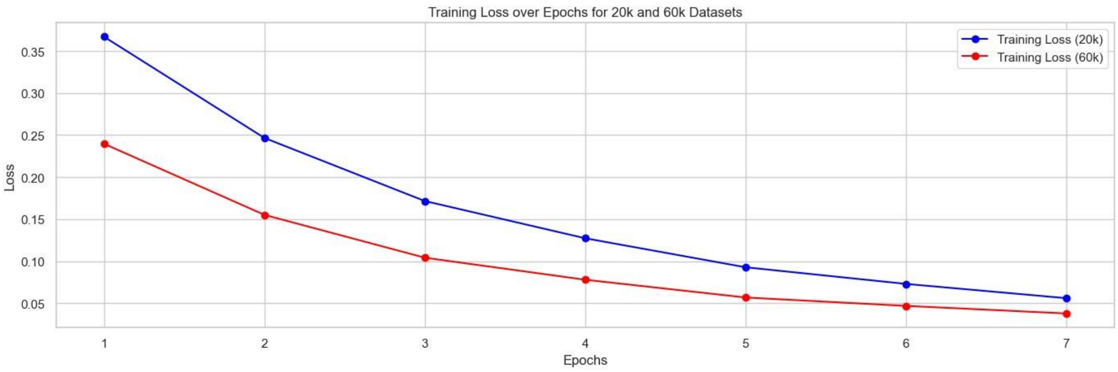

Figure 4. *Training loss over epochs for 20k and 60k datasets*

Additionally, we compare the model's accuracy in classifying different emotions across the two dataset sizes (20k and 60k). Fig. 7 shows it is clear that the model predicted all 5 emotion classes best when trained with the 60k dataset. The accuracy differences between the 2 dataset sizes range from 0.72% to 10.42%, with the most significant improvements being in the "Satisfied" and "Disappointed" classes.

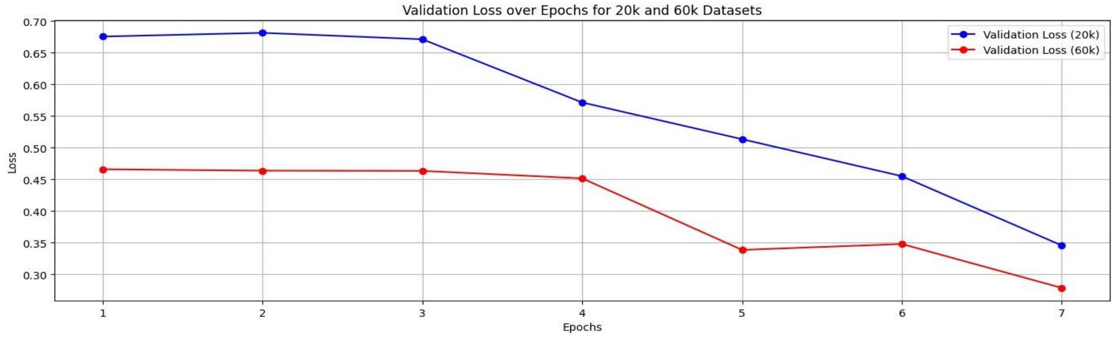

Figure 5. *Validation loss over epochs for 20k and 60k datasets*

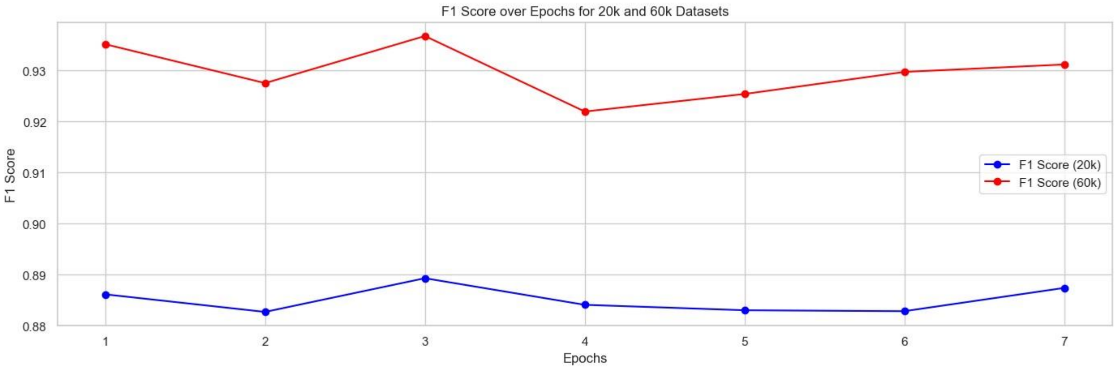

Figure 6. *F1 Score over epochs for 20k and 60k datasets*

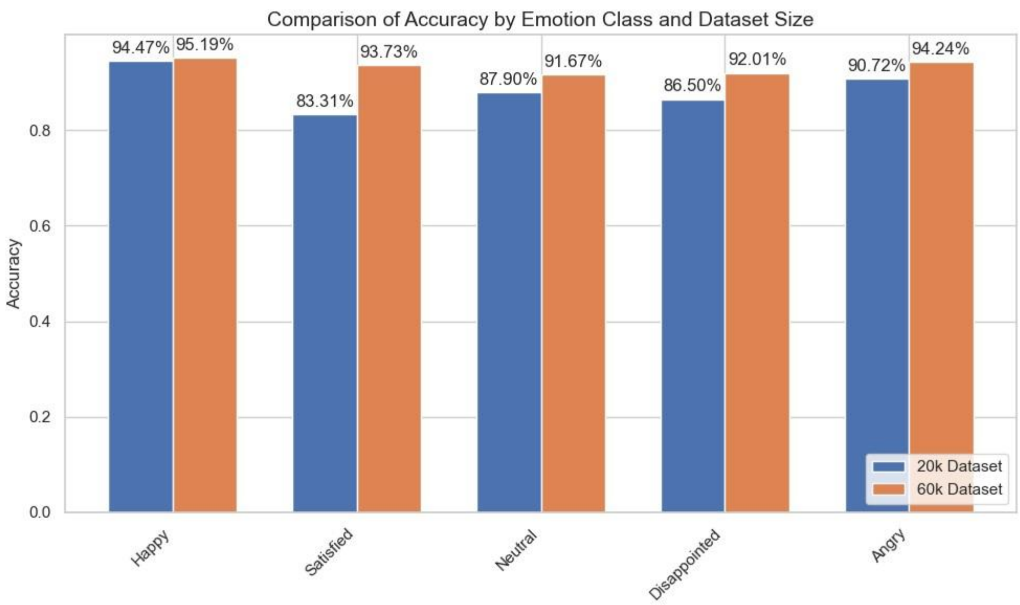

Figure 7. *Accuracy comparison of emotion class and dataset size*

Proposed website results: With the trained model successfully integrated into the e-commerce website's courses section, as depicted, comments and reviews about the courses will serve as inputs for sentiment analysis. Initially, the output results were categorized into 5 levels corresponding to different degrees, but to better match our application's goals, we adjusted it to 3 levels: ratings of 4-5 are positive, 1-2 are negative, and 3 is neutral. Grouping all emotions in 3 levels helps simplify the analysis and reduce confusion since our application's main goal is to detect positive and negative reviews. These new levels will be visualized through charts, allowing course administrators to assess the quality of user experience effectively. Fig. 8 illustrates some examples of our application's interface for administering students' reviews using the proposed models.

D. Application

In a previous study [17], we created a website that enables users to learn programming languages through available courses and incorporates AST (Abstract Syntax Tree) for code evaluation purposes. To leverage the achievements from this research, we have integrated the model obtained from the Fine-tuning process into the website. Fig. 8 shows our new website architecture. This allows us to analyze the sentiments of learners based on their course reviews. Some examples of the user comments and sentiment results are illustrated in Fig.8.

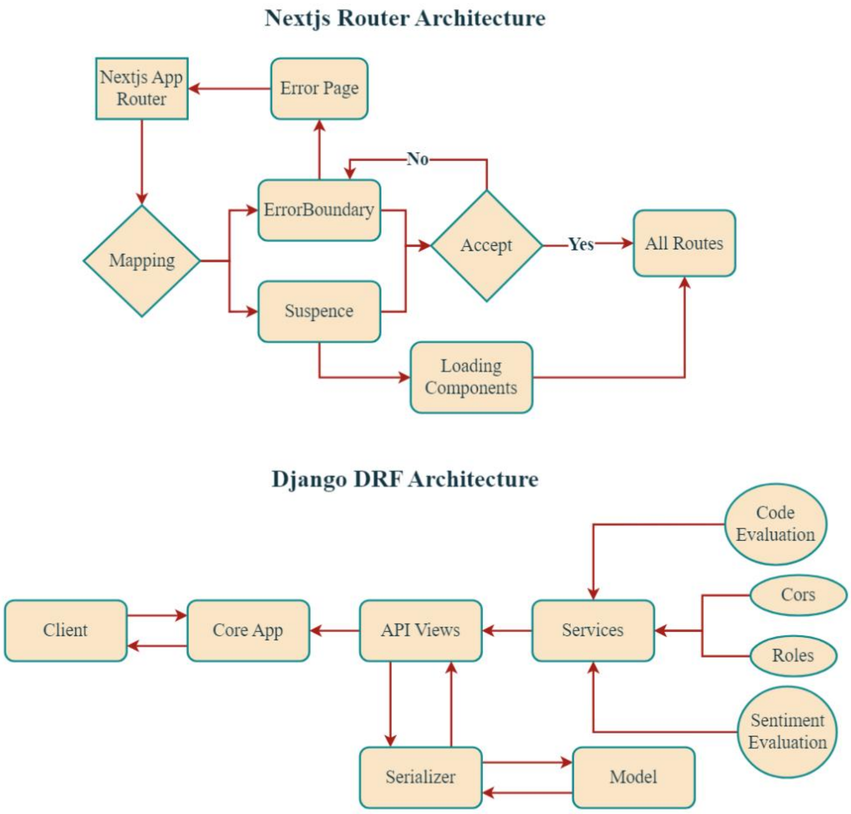

Figure 8. *Website architecture*

Integrating an NLP model into a programming learning website to analyze student sentiments based on their reviews offers significant benefits. Firstly, it improves the quality of the courses by collecting accurate feedback and identifying areas that need improvement or are highly appreciated. This also allows for customizing course content to better meet the needs and preferences of the students. Secondly, enhancing the user experience through instant feedback and personalized learning makes students feel cared for and supported, thus increasing their satisfaction and engagement. Thirdly, the system can early detect students with negative sentiments to intervene promptly, preventing them from dropping out or losing motivation. Fourthly, analyzing sentiment trends over time provides detailed data and insights to support management and product development, helping make strategic decisions based on well-founded analyses. Lastly, it enhances the reputation and quality of the service by generating more positive feedback and building a better brand image as you actively listen and improve based on student feedback.

5. Conclusions

In this research, we have developed a sentiment analysis system for an e-commerce education website, allowing the platform to better understand the emotions of the students' course reviews. Our approach employed RoBERTa, a robustly optimized BERT pretraining model. Furthermore, the balanced dataset and the use of advanced data processing techniques contributed significantly to the model's performance. However, there are still limitations in our research. A key one is the lack of diverse experimentations - we did not explore training with different models, data sizes, or datasets. The system's performance maybe influenced by the specific pre-training and fine-tuning processes of RoBERTa or the potential biases in the training dataset.

In the future, we aim to address the aforementioned limitations and improve the robustness and effectiveness of the model. A key focus will be on experimenting with different models, configurations, datasets and data sizes to expand the system's ability to handle a broader range of emotions. To achieve this, we plan to experiment with other state-of-the-art models with diverse configurations and incorporate data from various sources, including social media platforms, academic papers, and other relevant contexts, allowing the model to capture a wider array of emotional expressions and subtleties. Additionally, we are receptive to the latest advancements in the fields of natural language processing and sentiment analysis. We aim to continually refine and improve our system by integrating new insights and techniques from related studies.

Conflict of Interest

The authors declare no conflict of interest.

REFERENCES

[1] P. K. Sari, A. Alamsyah, and S. Wibowo, "Measuring e-commerce service quality from online customer review using sentiment analysis," *J. Phys.: Conf. Ser.*, vol. 971, p. 012053, 2018, doi: 10.1088/1742-6596/971/1/012053.

[2] L. Yang, Y. Li, J. Wang, and R. S. Sherratt, "Sentiment analysis for e-commerce product reviews in Chinese based on sentiment lexicon and deep learning," *IEEE Access*, vol. 8, pp. 23522–30, 2022, doi: 10.1109/ACCESS.2020.2969854.

[3] M. Demircan, A. Seller, F. Abut, and M. F. Akay, "Developing Turkish sentiment analysis models using machine learning and e-commerce data," *Int. J. Cogn. Comput. Eng.*, vol. 2, pp. 202–207, 2021.

[4] L. de Bruin, "Exploring the sentiment analysis performance of BERT models on domain-specific Twitter data when combined with an intelligent pre-processor," Bachelor thesis, Radboud University, Netherlands, 2022.

[5] E. Alzahrani and L. Jololian, "How different text-preprocessing techniques using the BERT model affect the gender profiling of authors," *arXiv*, vol. 10, p. 48550, 2021, doi: arXiv:2109.13890.

[6] M. Bilal and A. A. Almazroi, "Effectiveness of fine-tuned BERT model in the classification of helpful and unhelpful online customer reviews," *Electron. Commer. Res.*, vol. 23, pp. 2737–57, 2023, doi: 10.1007/s10660-022-09560-w.

[7] A. Vaswani, N. Shazeer, N. Parmar, *et al.*, "Attention is all you need," in *Adv. Neural Inf. Process. Syst. 30 (NIPS 2017)*, U. von Luxburg *et al.*, Eds. Long Beach, CA, USA: NIPS Foundation, 2017, pp. 5998–6008.

[8] H. N. Tran and U. Kruschwitz, "UR-IW-HNT at GermEval 2021: an ensembling strategy with multiple BERT models," *arXiv*, vol. 10, p. 48550, 2021, doi: 2110.02042.

[9] E. Hassan, T. Abd El-Hafeez, and M. Y. Shams, "Optimizing classification of diseases through language model analysis of symptoms," *Sci. Rep.*, vol. 14, p. 1507, 2024, doi: 10.1038/s41598-024-51615-5.

[10] W. Yu, B. T. Boenninghoff, and D. Kolossa, "BERT-based ironic authors profiling," *CEUR Workshop Proc.*, vol. 3180, pp. 2720–2733, 2022.

[11] R. Qasim, W. H. Bangyal, M. A. Alqarni, and A. A. Almazroi, "A fine-tuned BERT-based transfer learning approach for text classification," *J. Healthc. Eng.*, Jan. 7, 2022, doi: 10.1155/2022/3498123.

[12] D. T. Bui, B. Pradhan, O. Lofman, I. Revhaug, and O. B. Dick, "Landslide susceptibility mapping at Hoa Binh province (Vietnam) using an adaptive neuro-fuzzy inference system and GIS," *Comput. Geosci.*, vol. 45, pp. 199–211, 2012, doi: 10.1016/j.cageo.2011.10.031.

[13] K. Taalab, T. Cheng, and Y. Zhang, "Mapping landslide susceptibility and types using Random Forest," *Big Earth Data*, vol. 2, pp. 159–178, 2018, doi: 10.1080/20964471.2018.1472392.

[14] A. Hasan, S. Moin, A. Karim, and S. Shamshirband, "Machine learning-based sentiment analysis for Twitter accounts," *Math. Comput. Appl.*, vol. 23, p. 11, 2018, doi: 10.3390/mca23010011.

[15] S. Vijayaraghavan and D. Basu, "Sentiment analysis in drug reviews using supervised machine learning algorithms," *arXiv*, 2020, doi: arXiv:2003.11643.

[16] C. Qi, A. Fourie, Q. Chen, and Q. Zhang, "A strength prediction model using artificial intelligence for recycling waste tailings as cemented paste backfill," *J. Clean. Prod.*, vol. 183, pp. 566–578, 2018, doi: 10.1016/j.jclepro.2018.02.154.

[17] A. T. P. Nguyen, V. D. Hoang, *et al.*, "Development of Code Evaluation System based on Abstract Syntax Tree," *J. Tech. Educ. Sci.*, vol. 19, pp. 15–24, 2024, doi: 10.54644/jte.2024.1514.

Anh-Tu Phuong Nguyen is currently studying at the Ho Chi Minh City University of Technology and Education, Vietnam, majoring in Information Technology. Email: 21110105@student.hcmute.edu.vn. ORCID: https://orcid.org/0009-0005-9352-9353

JTE
HCMUTE
ISSN: 2615-9740

JOURNAL OF TECHNICAL EDUCATION SCIENCE
Ho Chi Minh City University of Technology and Education
Website: https://jte.edu.vn
Email: jte@hcmute.edu.vn

Duc-Khai Duong is currently studying at the Ho Chi Minh City University of Technology and Education, Vietnam, majoring in Information Technology. Email: 21110775@student.hcmute.edu.vn. ORCID: https://orcid.org/0009-0009-5456-6839

Van-Dung Hoang received his Ph.D. degree from the University of Ulsan, South Korea, in 2014. He was associated and joined as a visiting researcher with the Intelligence Systems Laboratory, University of Ulsan, in 2015. He joined the Robotics Laboratory on Artificial Intelligence, Telecom SudParis as a postdoctoral fellow, in 2016. He has been serving as an associate professor in computer science at, the Faculty of Information Technology, Ho Chi Minh City University of Technology and Education, Vietnam. He has published numerous research articles in ISI, Scopus indexed, and high-impact factor journals. He has been actively participating as a member of the societies such as IEEE, IEEE Computer, and ICROS. His research interests include a wide area, which focuses on pattern recognition, machine learning, medical image processing, computer vision application, vision-based robotics, and ambient intelligence. Email: dunghv@hcmute.edu.vn. ORCID: https://orcid.org/0000-0001-7554-1707