Jiang, B. et al. (Eds.) (2025). Proceedings of the 33rd International Conference on Computers in Education. Asia-Pacific Society for Computers in Education

Automated Multilingual Sentiment Analysis of Student Comments in Faculty Evaluations using Transformer-Based AI

Kent Levi BONIFACIO[a*], Jheanel ESTRADA[b], May Marie TALANDRON-FELIPE[c] , Gladys AYUNAR[d], Ferdinand Jr. BASCONES[e], Nathalie Joy Casildo[f] & Jinky MARCELO[g]

[a]*Student, Technological Institute of the Philippines – Manila Campus, Philippines*
[b] *Faculty, Technological Institute of the Philippines – Manila Campus, Philippines*
[c]*Faculty, University of Science and Technology of Southern Philippines Main Campus - Alubijid, Philippines*
[d,e,f,g] *Faculty, Central Mindanao University, Philippines*
*mklbonifacio@tip.edu.ph

Abstract: Open-ended student comments in faculty evaluations offer rich evidence for improving teaching, but the scale, subjectivity, and frequent code-mixing (English–Tagalog–Bisaya) make manual analysis slow and inconsistent. This research presents ClasSentiments, an automated, human-in-the-loop pipeline for multilingual sentiment analysis in higher education. Using a five-year dataset of 24,000 comments from a Philippine state university, we fine-tune and compare three transformer models—mBERT, Twitter-RoBERTa-base, and GPT-2—on a balanced training set (9,129 comments; 3,043/class). Twitter-RoBERTa achieves the best test performance (88% accuracy and the highest macro-F1) and generalizes on an expert-verified holdout (91.3% overall on 300 comments). We deploy the best model in a lightweight web application that provides per-comment labels, aggregate visualizations, and manual overrides with audit logs to preserve human judgment in high-stakes decisions. A formative usability study with staff, chairs, and instructors yields a SUS score of 87 (excellent). Contributions include (1) an empirical comparison of transformer architectures for short, code-mixed educational feedback; (2) a deployable analytics tool that integrates model outputs with human oversight; and (3) evidence of real-world readiness via expert validation and usability results. The approach aligns with learning analytics goals by turning qualitative student voice into timely and actionable insights for instructional improvement.

Keywords: Multilingual sentiment analysis, faculty evaluations, Transformer, Natural Language Processing (NLP), and Code-mixed text.

1. Introduction

Faculty evaluations are central for improving instructional quality in higher education. Institutions use this method to inform decisions about tenure, promotion, and professional development. Instructors use this as basis for actionable feedback on teaching effectiveness, course delivery, and student satisfaction. Beyond numeric ratings, open-ended student comments capture nuances that rating scales cannot, making them a valuable—yet underutilized—source of evidence for improvement.

Systematic analysis of large volumes of free-text comments is difficult. A single evaluation cycle can produce thousands of remarks that academic units still review by hand. This approach is slow, inconsistent, and prone to subjective interpretation, not to mention the consistency of evaluation among different reviewers. An additional challenge has been observed because of multilingual, code-mixed inputs (English–Tagalog–Bisaya). While off-the-shelf sentiment tools are available, often tuned for monolingual English which struggle to generalize to under-resourced languages and informal classroom discourse.

Improvements in NLP offer a practical alternative to this problem. Transformer models for instance, use self-attention to capture both local and global context (Vaswani et al., 2017). Pre-trained variants such as BERT and RoBERTa have set strong baselines for text classification (Devlin et al., 2019; Liu et al., 2019). When fine-tuned, these models perform

well on sentiment analysis, including applications to educational feedback (Fairooz & Hasan, 2023). This study presents ClasSentiments, an automated system for multilingual sentiment analysis of faculty-evaluation comments. We fine-tuned and compared three transformer models—BERT-base-multilingual-cased, Twitter-RoBERTa-base, and GPT-2—on a five-year dataset of student feedback written in English, Tagalog, and local Philippine languages (primarily Bisaya). We then integrated the best-performing model into a web application that delivers real-time analytics and administrative controls (e.g., manual overrides) to support institutional reporting and decision-making. With this, the research aims to contribute to the following domains.

- Compare several transformer models to identify a best multilingual student comment in faculty evaluations.
- Develop a web app that uses the identified best-performing model with useful dashboards for staff and instructors.

With this, the research aims to turn rich, qualitative student feedback into reliable, timely insights that inform teaching improvements and encourage actionable decisions. (Devlin et al., 2019; Liu et al., 2019; Vaswani et al., 2017).

Beyond polarity classification, prior work has approached open-ended faculty-evaluation comments through manual qualitative coding (with intercoder agreement) to surface actionable categories, topic/aspect extraction to group remarks by themes such as pacing or assessment clarity, and summarization/visualization systems that condense large comment sets for instructors and administrators (Hu, Zhang, Sathy, Panter, & Bansal, 2022). In the broader learning analytics literature, dashboards combine text with course outcomes and LMS traces to support reflective teaching and program review (Siemens & Long, 2011). Our contribution complements these strands by (a) handling multilingual, code-mixed inputs common in the Philippine context, (b) benchmarking modern transformers on short, informal comments, and (c) delivering a deployable, human-in-the-loop tool with overrides and audit trails for accountable use (Amershi et al., 2019; Shneiderman, 2020).

2. Methodology

The methodology comprised two phases. First is the development and fine-tuning transformer-based models for multilingual sentiment classification. The second phase is integrating the top-performing model into a web application - *ClasSentiments*. The output is a web application with real-time analysis and reporting powered by a trained model for multilingual comments. The methodology is structured to ensure effective model training, evaluation, and deployment for practical use in faculty evaluations.

2.1 Model Development

We fine-tuned three pre-trained transformer architectures—BERT-base-multilingual-cased (mBERT), Twitter-RoBERTa-base, and GPT-2. These are selected for their demonstrated sentiment classification performance and suitability for multilingual, short, and informal student comments. mBERT offers cross-lingual representations for English, Tagalog, and Bisaya (Devlin et al., 2019); Twitter-RoBERTa-base has strong results on conversational, short-form text (Liu et al., 2019); and GPT-2 provides a complementary decoder-only architecture that we adapt for supervised classification via a task-specific head (Radford et al., 2019). Publicly available considerations for all three models support reproducibility and deployment without licensing barriers (Devlin et al., 2019; Liu et al., 2019; Radford et al., 2019).

2.2 Data Collection and Preprocessing

2.2.1 Collection and Annotation

With approved ethics clearance, 24,000 open-ended student comments from faculty evaluations were collected over multiple academic years at a state university in the Philippines. All personally identifying information (PII) was removed prior to analysis to protect privacy. Only de-identified texts were retained for modeling. Comments span English, Tagalog, and Bisaya, reflecting frequent code-mixing typical of the local context.

Comments were labeled into Positive, Neutral, and Negative categories by a domain expert from the university (guidance counselor). A written guideline was made to ensure consistency and verifiability of the dataset.

2.2.2 Text Normalization

We applied language-agnostic cleaning to preserve linguistic cues important for sentiment while reducing noise – which includes trimming whitespace, removing control characters and obvious non-text strings (e.g., “1111111”), collapsing repeated punctuation, and lowercasing where appropriate. Single-character/tokens and nonsensical entries were discarded. Because Filipino student comments often contain code-switching and slang, we retained mixed-language tokens and consulted student volunteers to normalize common slang forms without altering semantic polarity. Retaining code-mixed structure is appropriate when using multilingual transformer encoders that learn cross-lingual sub-word representations (Devlin et al., 2019; Liu et al., 2019).

2.2.3 Class Imbalanced Dataset

Preliminary inspection of the dataset indicated class imbalance. The negative comments were underrepresented by thousands. We first experimented with SMOTE to synthetically expand minority classes (Chawla et al., 2002) and ADASYN to adaptively oversample more difficult minority examples (He et al., 2008). To mitigate lexical repetition artifacts that can arise when oversampling short texts, we followed with random undersampling of the majority instances. After balancing, the working dataset contained 9,129 comments (3,043 per class).

Rather than translating to a different language, we processed comments in the original languages so that fine-tuned multilingual encoders (e.g., mBERT, RoBERTa variants) could exploit subword sharing and contextual transfer across English–Tagalog–Bisaya (Devlin et al., 2019; Liu et al., 2019). This aligns with the deployment setting where mixed-language feedback is the norm and translation may introduce noise once deployed.

2.3 Model Training

We used three (3) fine-tuned transformer models: mBERT (BERT-base-multilingual-cased), Twitter-RoBERTa-base, and GPT-2. This is to span a multilingual encoder (mBERT); an encoder pretrained on short, conversational text (RoBERTa); and a decoder-only architecture adapted with a classification head (GPT-2) (Devlin et al., 2019; Liu et al., 2019; Radford et al., 2019).

2.3.1 Tokenization

A stratified 80/20 train–test split and reserved 10% of the training set as a validation set for early stopping and model selection was used. Labels followed a three-class scheme −1 (Negative), 0 (Neutral), 1 (Positive). Each model applied its native tokenizer (WordPiece for mBERT, byte-pair encoding for RoBERTa, and GPT-2). The tokenization preserved code-mixed text (English–Tagalog–Bisaya) and set max sequence length = 128 to match short student comments while controlling memory usage (Devlin et al., 2019; Liu et al., 2019).

2.3.2 Optimization and hyperparameters

We optimized with AdamW (decoupled weight decay) using weight decay = 0.01 (Loshchilov & Hutter, 2019). Learning rates were 2×10^{-5} for mBERT and Twitter-RoBERTa and 1×10^{-5} for GPT-2; a linear warmup (10% of total steps) and linear decay scheduler

stabilized training (Devlin et al., 2019). We trained for 10 epochs, with batch size = 32 (reduced as needed on memory-limited runs), gradient clipping = 1.0, and early stopping with patience = 2 based on validation macro-F1.

2.3.3 Training details

We did not apply class weights because the dataset produced a balanced dataset (3,043 per class). Balancing was conducted before splitting, and the test set was held out from any oversampling/undersampling to avoid leakage.

For mBERT and RoBERTa, we attached a standard linear classification head; for GPT-2, we added a task-specific linear layer and fine-tuned end-to-end (Radford et al., 2019). We selected the best checkpoint by validation macro-F1, saved it, and used it for downstream evaluation and deployment.

Experiments were made on Google Colaboratory. We fixed random seed(s) for the tokenizer, data loader, and model initialization. Core libraries included PyTorch and Hugging Face Transformers (version details recorded), ensuring replicability on commodity cloud hardware.

2.3.4 Model Evaluation

We report accuracy, precision, recall, and F1-score, including macro-F1 (and weighted-F1, where noted), and present confusion matrices for class-wise error analysis. To enhance robustness, we conducted multiple runs with fixed random seed(s) and, where applicable, report mean ± standard deviation (Kohavi, 1995).

2.4 Software Development

The best-performing model was integrated via a lightweight Flask API backed by PyTorch/Hugging Face. The UI (HTML/CSS/JavaScript) reads de-identified comments from a MySQL view, applies the same tokenization/truncation used in training to avoid train–serve skew (Devlin et al., 2019; Liu et al., 2019), and returns batched predictions for course/term analyses. Each comment is shown with a Positive/Neutral/Negative label and can be manually overridden by reviewers; overrides are flagged in the interface and logged for transparency and post-hoc error analysis. This preserves human judgment in a high-stakes, institutional workflow and follows human-AI guidance emphasizing oversight and accountability (Amershi et al., 2019; Shneiderman, 2020).

A Dashboard summarizes the sentiment distributions per faculty/course/term and support PDF export (two templates: student-focused and administrative) for inclusion in routine reports. Inputs are de-identified before inference; the system records override events (who/when/what) to support audit and continuous model improvement.

In a formative study with 30 participants, the interface achieved a System Usability Scale (SUS) score of 87, typically interpreted as excellent, indicating low learning overhead for staff, chairs, and instructors (Brooke, 1996). The production service loads the validated checkpoint used in evaluation to ensure consistency and supports batched inference to keep end-to-end latency low during bulk processing.

3. Results

3.1 Model Performance

We evaluated three transformer models—mBERT, Twitter-RoBERTa, and GPT-2—on the balanced test set using accuracy, precision, recall, and F1-score (macro and weighted). Shown in Table 1, Twitter-RoBERTa achieved the best overall performance (accuracy ≈ 88%), followed by GPT-2 (≈ 87%) and mBERT (≈ 84%). Macro-F1 patterns mirrored accuracy: Twitter-RoBERTa > GPT-2 > mBERT. Validation curves indicated earlier convergence and lower validation loss for Twitter-RoBERTa, suggesting better generalization on short, informal, and code-mixed comments. Typical failure cases for mBERT involved Positive ↔ Neutral confusion; GPT-2 showed slight instability on Positive class boundaries; Twitter-RoBERTa reduced both patterns.

Table 1. *Overall performance*

	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)	Best Validation Accuracy (%)	Best Epoch
BERT-base-multilingual	84	84	84	84	84.17	5
Twitter-RoBERTa-base	**88**	**88**	**87**	**88**	**87.35**	**5**
GPT-2	87	87	87	87	86.58	7

3.2 Error Analysis - Confusion Matrices

Confusion matrices highlight systematic confusions and class balance after preprocessing. Twitter-RoBERTa made fewer Neutral-to-Positive / Positive-to-Neutral and Negative-to-Neutral / Neutral-to-Negative errors than mBERT and GPT-2. This is consistent with its higher macro-F1 score as shown in Figure 1. Common missed predictions involved short, sarcastic “positives” and polite negatives with hedging language.

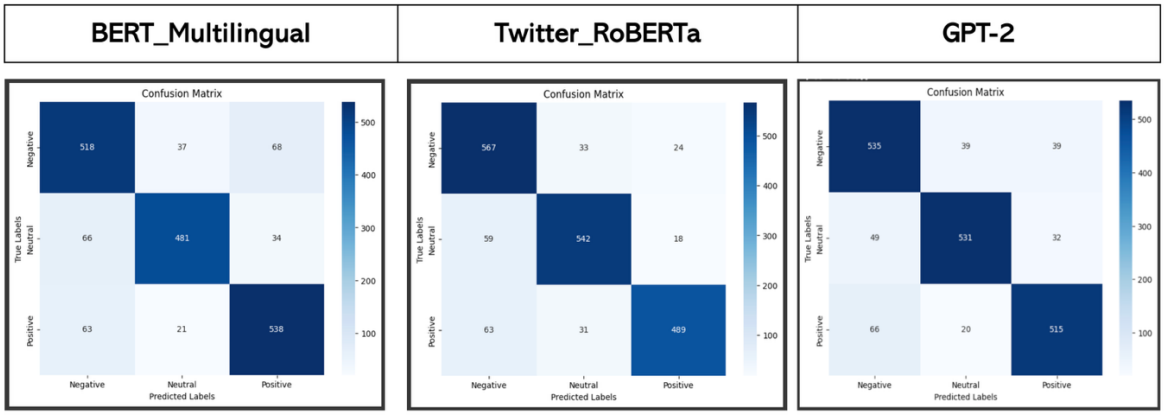

Figure 1. Confusion matrices by model: mBERT, Twitter-RoBERTa, and GPT-2

Twitter-RoBERTa outperformed mBERT and GPT-2 because its pretraining domain and training match our data - short, informal, sometimes noisy student remarks. RoBERTa’s optimized objectives and data scale (dynamic masking, larger corpora) bolster robustness on conversational text. At the same time, byte-pair encoding helps represent code-mixed tokens without heavy vocabulary penalties (Liu et al., 2019). This manifests as fewer Neutral-to-Positive/ Positive-to-Neutral and Negative-to-Neutral confusions precisely where hedging and politeness create ambiguity. We expect similar gains in deployments with brief, multi-lingual or code-mixed comments; for longer formal narratives or many non-Latin scripts, a multilingual encoder such as XLM-R may be competitive.

3.3 External Validation on Expert-Verified Set

The trained models were evaluated on a new set of expert-verified comments with 100 comments per class. Like the training results, the Twitter-RoBERTa performed best with an overall ≈ 91.3% as presented in Table 2. The result shows strong accuracy for all three classes. Following is GPT-2 and mBERT with the lowest scores among the three.

Table 2. *Result of the expert-verified dataset with 100 comments per class*

	BERT-base-multilingual (%)	Twitter-RoBERTa-base (%)	GPT-2 (%)

Positive	81	**92**	90
Neutral	77	**92**	91
Negative	80	90	**91**
Overall	79.33	**91.33**	90.67

3.4 Decision Support Outcomes

Case reviews with staff and department chairs showed that aggregate sentiment dashboards and per-comment labels surfaced recurring issues (e.g., pacing, assessment clarity) faster than manual reading. The override log feature strengthened the accountability of the system's results. The reviewers corrected edge cases and flagged items to prompt follow-up conversations with faculty. Figure 2 shows the screenshots of the web application.

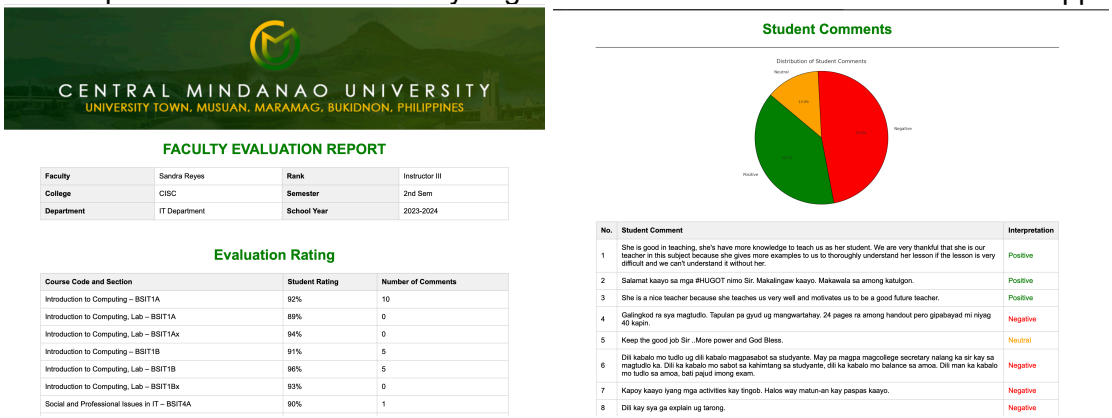

Figure 2. Developed web application screenshots

3.5 Usability Evaluation (SUS)

A usability study with 30 participants representing expected user roles consisting of academic staff, department chairs, and instructors resulted to a System Usability Scale (SUS) score of 87. This is interpreted as excellent (Brooke, 1996). Participants cited the clarity of sentiment summaries, easy PDF export, and the override workflow as key strengths. Table 3 shows SUS by role: academic staff (n=12, mean 86.0, SD 7.2, 95% CI 81.9–90.1), department chairs (n=8, 88.0, 6.8, 83.3–92.7), and instructors (n=10, 87.0, 7.5, 82.4–91.6). Overall SUS is 86.9≈87 (SD 7.0, 95% CI 84.4–89.4), with an excellent interpretation for every group.

Table 3. *System Testing using SUS result*

	Respondents	SUS mean	SD	95% CI	Interpretation
Academic staff	12	86.0	7.2	81.9 – 90.1	Excellent
Department chair	8	88.0	6.8	83.3 – 92.7	Excellent
Instructor	10	87.0	7.5	82.4 – 91.6	Excellent
Overall	**30**	**86.9 ≈ 87**	**7.0**	**84.4 – 89.4**	**Excellent**

3.6 Ablation of the study

We quantify how key design choices affect performance on short, code-mixed comments. Unless noted, ablations start from the Twitter-RoBERTa baseline used in production - balanced training set (3,043/class), max sequence length 128, AdamW with linear warmup/decay, early stopping on validation macro-F1, and preservation of code-mixed tokens (English–Tagalog–Bisaya) (Devlin et al., 2019; Liu et al., 2019; Loshchilov & Hutter, 2019). With Baseline (B0), Twitter-RoBERTa @ seq len 128, balanced data on test metrics of Acc 88.0, Macro-F1 88.0 as presented in Table 1.

We also examined training without rebalancing and with alternative oversampling strategies to see the difference. Table 4 shows that class balance strategy materially affects macro-F1 on short, code-mixed comments. Training on the originally skewed data reduces macro-F1 to 83.9% (Acc 86.2%) because minority-class recall drops. SMOTE only lifts macro-F1 to 87.3% and ADASYN only to 87.0%, but both introduce repetition/noise that is typical of

short texts that are oversampled. The best trade-off comes from the combined moderate oversampling + light under sampling (the baseline of this research) that achieves 88.0% macro-F1 (Acc 88.0%) by boosting minority recall. Notably, the held-out test set remained untouched, so improvements reflect training-time handling of the imbalance dataset.

Table 4. *Class balancing result*

Training data setting	Accuracy (%)	Macro-F1 (%)	Δ Macro-F1 vs. B0
No balancing (original skew)	86.2	83.9	−4.1
SMOTE only	87.6	87.3	−0.7
ADASYN only	87.4	87	−1.0

3.7 Discussions

The model from the Twitter-RoBERTa performed best on short, code-mixed student comments. This outperforms mBERT and GPT-2. It reached about 88% accuracy on the balanced test split and about 91% on a separate expert-verified set. In practice, it produced fewer borderline errors, avoiding cases labeled “positive” when the language was merely polite or hedged. This pattern is consistent as expected from transformer encoders trained on conversational text (Devlin et al., 2019; Liu et al., 2019).

The web application incorporates the model into a usable feature, including features to further improve the class validation through user intervention – this keeps the department in control and supports accountability (Amershi et al., 2019; Shneiderman, 2020). The SUS score of 87 suggests the application is usable. This means that the cost of bringing the tool into routine evaluation cycles is low. Thus, easy to use (Brooke, 1996). Framed as learning analytics, this is less about replacing judgment and more about shortening the path from raw comments to useful conversation.

For teachers, per-comment labels and trend charts surface recurring issues (e.g., pacing, assessment clarity). This supports timely course adjustments. Department staff and chairs use cohort-level distributions to identify courses needing attention, while QA/administration exports structured PDFs for program review and accreditation. The override workflow sustains trust reviewers correct edge cases (sarcasm, mixed pragmatics) with changes recorded in an audit log for accountability (Amershi et al., 2019; Shneiderman, 2020).

To replicate this system, we recommend (1) governance first—de-identify data and document consent; (2) preserve code-mixing rather than translating, and pick encoders tuned for short; (3) balance training data and report macro-F1 with confusion matrices; (4) ship a human-in-the-loop UI with visible overrides, confidence display, and brief rationales; and (5) operationalize the best checkpoint by validation macro-F1. A compact SUS study (Brooke, 1996) helps refine defaults (filters, exports) before wider rollout.

4. Conclusion

ClasSentiments helps multilingual, code-mixed student feedback in faculty reviews. After tuning three transformer models, Twitter-RoBERTa consistently beat mBERT and GPT-2. In our main test split and on an expert-validated set, showing the strongest accuracy and macro-F1 for short, informal English–Tagalog–Bisaya comments. Integrated into a human-centered web application, the system supports institutional workflows through per-comment labels, aggregate visualizations, manual overrides with audit logs, and exportable reports. A formative usability study (SUS = 87) indicates excellent perceived usability and low adoption friction for academic staff and instructors (Brooke, 1996).

Our results suggest that modern transformer encoders, when fine-tuned with careful preprocessing and evaluation, can translate qualitative student comments into actionable learning analytics for teaching improvement (Devlin et al., 2019; Liu et al., 2019). With equal importance, humans in the loop feature through visible overrides is included, which is critical for accountable use in high-stakes settings (Amershi et al., 2019; Shneiderman, 2020) and future model improvement.

In summary, matching model pretraining to comment style (short, informal, code-mixed) and embedding the model in a human-in-the-loop workflow yields reliable, actionable summaries at scale. Next steps include expanding languages and institutions, adding aspect-based and emotion signals, improving confidence communication and lightweight explanations, benchmarking against stronger multilingual baselines (e.g., XLM-R), and running longitudinal studies to see whether these analytics meaningfully change teaching practice and student outcomes.

Acknowledgements

This research would not have been possible without the support and the contributions of the following students: Jianne Merijo E. Mengote, Ritchel M. Naquinez, and Arram T. Pamisa.

References

Amershi, S., Weld, D. S., Vorvoreanu, M., Fourney, A., Nushi, B., Collisson, P.,Horvitz, E. (2019). Guidelines for human–AI interaction. Proceedings of CHI 2019, 1–13. https://doi.org/10.1145/3290605.3300233

Brooke, J. (1996). SUS: A “quick and dirty” usability scale. In P. W. Jordan, B. Thomas, I. L. McClelland, & B. A. Weerdmeester (Eds.), Usability evaluation in industry (pp. 189–194). Taylor & Francis.

Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: Synthetic minority over-sampling technique. Journal of Artificial Intelligence Research, 16, 321–357. https://doi.org/10.1613/jair.953

Cohen, J. (1960). A coefficient of agreement for nominal scales. Educational and Psychological Measurement, 20(1), 37–46.

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. NAACL-HLT 2019, 4171–4186.

Deshpande, H., Khuntia, J., & Gupta, P. (2025). Elevating educational insights: Sentiment analysis of faculty feedback. Advances in Continuous and Discrete Models, 2025(89). https://doi.org/10.1186/s13662-025-03933-9

Fairooz, M. S., & Hasan, M. T. (2023). Improving sentiment analysis in online education: A transformer approach [Preprint]. Research Square.

He, H., Bai, Y., Garcia, E. A., & Li, S. (2008). ADASYN: Adaptive synthetic sampling for imbalanced learning. 2008 IEEE World Congress on Computational Intelligence (IJCNN), 1322–1328. https://doi.org/10.1109/IJCNN.2008.4633969

Hu, Y., Zhang, S., Sathy, V., Panter, A. T., & Bansal, M. (2022). SETSum: Summarization and visualization of student evaluations of teaching. NAACL 2022 (Demonstrations), 63–70.

Kastrati, Z., Dalipi, F., Imran, A. S., Pireva Nuci, K., & Wani, M. A. (2021). Sentiment analysis of students’ feedback: A systematic mapping study. Applied Sciences, 11(9), 3986. https://doi.org/10.3390/app11093986

Kingma, D. P., & Ba, J. (2015). Adam: A method for stochastic optimization. ICLR 2015 (arXiv:1412.6980).

Kohavi, R. (1995). A study of cross-validation and bootstrap for accuracy estimation and model selection. IJCAI-95, 1137–1145.

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D.,Stoyanov, V. (2019). RoBERTa: A robustly optimized BERT pretraining approach. arXiv:1907.11692.

Loshchilov, I., & Hutter, F. (2019). Decoupled weight decay regularization. ICLR 2019.

Maceda, L. L., Satuito, A. A., & Abisado, M. B. (2023). Sentiment analysis of code-mixed social media data on Philippine UAQTE using fine-tuned mBERT model. International Journal of Advanced Computer Science and Applications, 14(7). https://doi.org/10.14569/IJACSA.2023.0140777

Peña-Torres, J. A. (2024). Towards an improved teaching practice using sentiment analysis in student evaluation. Ingeniería y Competitividad, 26(2), e-21013759. https://doi.org/10.25100/iyc.v26i2.13759

Pilicita-Garrido, A., & Barra, E. (2025). Sentiment analysis with transformers applied to education: Systematic review. International Journal of Interactive Multimedia and Artificial Intelligence, 9(2), 71–83. https://doi.org/10.9781/ijimai.2025.02.008

Pilicita, A., & Barra, E. (2025). LLMs in education: Evaluating GPT and BERT models in student comment classification. Multimodal Technologies and Interaction, 9(5), 44. https://doi.org/10.3390/mti9050044

Radford, A., Wu, J., Amodei, D., & Sutskever, I. (2019). Language models are unsupervised multitask learners. OpenAI.
Shneiderman, B. (2020). Human-Centered AI: Reliable, safe & trustworthy. International Journal of Human–Computer Interaction, 36(6), 495–504. https://doi.org/10.1080/10447318.2020.1741118
Singla, S., & Ramachandra, N. (2020). Comparative analysis of transformer-based pre-trained NLP models. International Journal of Computer Sciences and Engineering, 8(11), 40–44.