A Comparative Study of Classical Machine Learning, Deep Learning, and Transformer Algorithms for Emotion Classification on Indonesian-Language Twitter Text

Jonathan Davin
Computer Science Department
School of Computer Science
Bina Nusantara University
Jakarta, Indonesia
jonathan.davin@binus.ac.id

Shane Anthony
Computer Science Department
School of Computer Science
Bina Nusantara University
Jakarta, Indonesia
shane.anthony@binus.ac.id

Bryan Dale
Computer Science Department
School of Computer Science
Bina Nusantara University
Jakarta, Indonesia
bryan.dale@binus.ac.id

Mohammad Faisal Riftiarrasyid
Computer Science Department
School of Computer Science
Bina Nusantara University
Jakarta, Indonesia
mohammad.riftiarrasyid001@binus.

***Abstract*—Emotion classification in Indonesian-language social media text faces two fundamental challenges, the noisy linguistic profile of Twitter text and severe class imbalance. To address these, this study compares three modeling paradigms on the EmoT dataset from IndoNLU, comprising 4,401 samples across five emotion classes: Classical Machine Learning (Complement Naive Bayes/CNB, SVM, Random Forest), a sequential Bidirectional Long Short-Term Memory (BiLSTM) model with FastText embeddings, and Pre-trained Transformers from both the general domain (IndoBERT) and the Twitter-specific domain (IndoBERTweet). Based on Macro Average F1-Score, IndoBERTweet achieves the best result at 0.76, ahead of IndoBERT (0.70), CNB (0.69), SVM (0.67), Random Forest (0.64), and BiLSTM (0.63). Three findings stand out. First, the proximity between the pre-training data domain and the downstream task's data distribution has a greater deterministic effect than model parameter size. Second, CNB proves to be a stronger baseline than BiLSTM under limited, imbalanced dataset conditions. Third, the superiority of IndoBERTweet over CNB is statistically significant, confirmed by McNemar's Test with p-value = 0.0066. The code for this experiment can be accesssed via the following github link: https://github.com/SauHin/emotion-classification.**

Keywords—Emotion classification, Classical machine learning, BiLSTM, Transformer, Indonesian Twitter.

I. INTRODUCTION

The growth of social media, particularly Twitter (X), has produced a large and varied source of textual data for opinion and emotion analysis. Automatically detecting emotion in Twitter text supports a range of applications, from brand monitoring and early detection of mental health crises to real-time analysis of public opinion. However, processing text from this platform has substantial linguistic problems for Natural Language Processing (NLP) systems.

Twitter users typically write in informal spelling, non-standard language (slang), and an ever-shifting set of abbreviations [1]. This unstructured and noisy text introduces considerable ambiguity into computational processing. Standard dictionaries struggle to capture such slang variations and abbreviations, and this gap often drives classification accuracy sharply down [2], [3]. The problem deepens further with sarcasm and implicit emotional expression, both of which depend entirely on sentence-level context to interpret correctly.

The dataset itself adds another layer of difficulty on top of these linguistic issues. This study uses the EmoT dataset from IndoNLU [4], a standard benchmark for Indonesian NLP evaluation. The dataset exhibits fairly severe class imbalance, where from its five emotion classes, Anger, Happy, and Sadness dominate the distribution, while Fear and Love sit as minority classes at roughly 14% each. Any model evaluated on this dataset has to contend with that imbalance directly.

Previous work has tended to evaluate these algorithms separately, comparing classical models on their own or Transformers on their own, which leaves no comprehensive comparison across all three modeling paradigms under the same dataset conditions [5], [6]. The relative effectiveness of domain-specific pre-training (i.e., Twitter-domain pre-training) versus general-domain pre-training for Indonesian emotion classification also remains underexplored, particularly with rigorous statistical significance testing. Therefore, our comparison covers Classical Machine Learning (CNB, SVM, Random Forest), a sequential Deep Learning model (BiLSTM with FastText), and Pre-trained Transformers (IndoBERT and IndoBERTweet) [7], [8].

II. RELATED WORKS

A. Text Classification and Indonesian NLP Benchmarks

The standardization of NLP evaluation for Indonesian was significantly advanced by the release of IndoNLU by Wilie et al. (2020), which provided a set of standardized benchmarks, including the EmoT emotion classification dataset [4]. On EmoT and similar social media datasets, Classical Machine Learning algorithms such as Support Vector Machine (SVM) and Naive Bayes are commonly used as baselines, because of their well-established historical reliability [5]. Qolbu et al. (2025) showed that on Twitter sentiment analysis tasks with imbalanced data, NB and SVM consistently form baselines that more complex architectures struggle to beat [5].

TABLE I. EMOT DATASET DISTRIBUTION

Emotion Class	Train	Validasi	Test	Total
Anger	881	110	110	1.101
Happy	814	102	101	1.017
Sadness	798	99	100	997
Fear	519	65	65	649
Love	509	64	64	637
Total	**3.521**	**440**	**440**	**4.401**

B. *Limitations of Sequential Architectures on Limited Data*

Deep Learning architectures such as Long Short-Term Memory (LSTM) networks can model sequential dependencies between words, but their performance depends heavily on training data volume (*data-hungry*). Anggraini et. al. (2026) found that on small datasets, BiLSTM tends to overfit and struggles to learn meaningful sequential patterns, a consequence of its narrow representational variance [6]. The result fits the broader theoretical claim that recurrent architectures need massive data volumes to extract contextual features that generalize. Classical algorithms like Logistic Regression tend to be more computationally efficient and often outperform BiLSTM in exactly these limited-data settings [5], [6].

C. *Transformer Development and Domain-Specific Pre-training*

The Transformer architecture resolves many of LSTM's limitations through its Self-Attention mechanism, which processes all tokens in a sentence in parallel and assigns adaptive computational weight to semantically important words [5]. This bidirectional reading of context is essential for detecting implicit emotion, ambiguity, and sarcasm [1].

In Indonesia, Koto et al. introduced IndoBERT, pre-trained on formal text such as news articles and Wikipedia [7], and IndoBERTweet, pre-trained specifically on an Indonesian-language Twitter corpus [8]. Shaw et al. (2025) found that the proximity between the pre-training data distribution and the downstream fine-tuning data has a more deterministic effect on performance than model parameter size alone [9]. Koto et al. (2021) reinforce this point, showing that general-domain models tend to split slang vocabulary into sub-words that lose their semantic meaning [8].

III. METHODOLOGY

This study systematically compares the performance of three machine learning models: Classical Machine Learning, BiLSTM, and Pre-trained Transformer. Every stage of the experiment was designed to ensure a fair comparison across models.

A. *Classification Models*

We evaluate six different model based of the three modeling paradigms mentioned. Each model is selected to investigate how different algorithmic complexities handle the dataset. The descriptions of the selected models are detailed below:

1) *Complement Naive Bayes:* A modification of the Multinomial Naive Bayes algorithm specifically designed to address imbalanced datasets [10].
2) *Support Vector Machine:* Identifies the optimal decision hyperplane to classify words based on the maximum margin between emotion classes. It exhibits high robustness in handling high-dimensional data. [5]
3) *Random Forest:* An ensemble method that determines the final classification through a majority voting mechanism across a collection of decision trees. This algorithm is well-suited for evaluating non-linear classification approaches. [2]
4) *BiLSTM + FastText:* Processes text bidirectionally to capture sequential context, a capability that the three aforementioned classical algorithms lack [11].
5) *IndoBERT:* A Transformer-based model utilizing a self-attention mechanism, pre-trained on a formal Indonesian corpus to mitigate data starvation [1].
6) *IndoBERTweet:* A Transformer model pre-trained on Indonesian Twitter social media text, directly aligning with the characteristics of the dataset used in this study [8].

B. *Dataset Description*

The dataset used is Emotion Twitter (EmoT) from the IndoNLU corpus [4], which contains Indonesian-language tweets that manually annotated into five emotion classes: Anger, Happy, Sadness, Fear, and Love. The complete data distribution across each split is presented in Table I. The dataset exhibits significant class imbalance: Anger, Happy, and Sadness dominate the data proportion, while Fear and Love each contribute only around 14.7% of the total samples. The dataset can be accessed via the following GitHub link: https://github.com/IndoNLP/indonlu/tree/master/dataset/emot_emotion-twitter.

C. *Data Preprocessing*

To accommodate differences in how each architecture operates, data preprocessing was split into two independent pipelines, each tailored to the representational requirements of its model group.

1) *Classical ML Pipeline:* Text was lowercased, then stripped of tags (‘[USERNAME]’ and ‘[URL]’), numbers and punctuation. Next, the slang words was normalized using Colloquial Indonesial Lexicon from Salsabila [3] that we modified by adding new slang based on the most frequent slang word from the dataset. We also applied a laugh normalization using regex to map laugh variation (e.g awkoakowaok will becom “haha”). Lastly, we applied stopwords removal using Sastrawi

TABLE II. CLASICAL MODEL'S HYPERPARAMETER

Hyperparaneter	**CNB**	**SVM**	**Random Forest**
Before RandomizedSearchCV	Default	kernel = 'linear' class_weight = 'balanced' random_state = 42	n_estimators = 100 class_weight = 'balanced' random_state = 42 n_jobs = -1
After RandomizedSearchCV	**alpha = 3.142880890840109 norm = True**	**C = 0.6251373574521749**	**max_depth = 30 min_samples_split = 7 n_estimators = 179**

StopWordRemoverFactory that have been added with frequent english stopword used in the dataset and stemming via the Sastrawi algorithm to reduce vocabulary dimensionality and standardize the morphological form of words.

2) *BiLSTM Pipeline:* The preprocessing steps were the same as the Classical ML Pipeline but without stopword removal and stemming, to preserve grammatical structure and full sentence-level context.
3) *Transformer Pipeline:* For IndoBERT, the text was lowercased and both tags were stripped out. While for IndoBERTweet specifically, the '[USERNAME]' and '[URL]' tags were not removed but normalized to '@USER' and 'HTTPURL', aligning with the model's pre-training data format [8].

D. Feature Extraction and Word Representation

Feature extraction was used to each model's architecture:

1) *TF-IDF (Classical ML):* A Term Frequency-Inverse Document Frequency matrix was used for its ability to balance a word's local frequency against its global importance across the corpus [12]. This representation is highly efficient for modeling discrete term distributions in probabilistic models [11].
2) *FastText Word Embeddings (BiLSTM):* BiLSTM used pre-trained 300-dimensional Indonesian FastText word vectors with frozen weights (trainable=False), chosen for their ability to handle Out-Of-Vocabulary (OOV) words by decomposing them into character n-grams, a property that lets the model synthesize meaningful vectors even for previously unseen slang or typographical errors [13]. Weights were frozen to prevent overfitting given the limited dataset size.
3) *Sub-word Tokenization (Transformer):* IndoBERT and IndoBERTweet relied on their respective built-in sub-word tokenizers, with a maximum sequence length of 43 tokens, matching the median tweet length in the dataset.

E. Classical Model Optimization

To ensure a fair comparison between classical and neural network models, all three Classical Machine Learning algorithms (CNB, SVM, and Random Forest) were first optimized through a hyperparameter space search using Randomized Search Cross-Validation. This method was chosen because it has been empirically shown to find optimal configurations more efficiently than exhaustive Grid Search in high-dimensional feature spaces such as TF-IDF matrices [14]. Table II sumarizes the results of hyperparamter tuning done on three classical model.

F. Deep Learning and Transformer Configuration

We used the following configuration for the Deep Learning and Transformer model:

1) BiLSTM + FastText: The architecture consists of an Embedding layer (300-d, frozen), a SpatialDropout1D layer (rate=0.3), a Bidirectional LSTM layer with 64 units, a Dropout layer (rate=0.3), and a final Dense layer with softmax activation across the five classes. The model was compiled with the Adam optimizer ($lr=1\times10^{-3}$) and sparse categorical crossentropy loss. Input sequence length was fixed at 43 tokens, with post-padding and pre-truncation.
2) IndoBERT and IndoBERTweet (Custom Classification Head): Both models adopt the configuration from Shaw et al. [9], replacing the standard [CLS] token extraction with Global Average Pooling (GAP) over the final hidden layer. GAP averages the representations of all valid tokens into a single document vector that captures information from the full sentence, a method shown to minimize information loss [9]. The resulting GAP vector is then passed to a Custom Classification Head, an MLP arranged as Dense(64, ReLU), Dropout(0.05), Dense(16, ReLU), Dense(5). Training used AdamW ($lr=3\times10^{-5}$, weight_decay=0.005) with a batch size of 64.

G. Deep Learning Training and Class Imbalance Handling

For the BiLSTM models, class imbalance was addressed through class weight adjustment, computed automatically via scikit-learn's balanced scheme. The resulting weights penalize misclassification of the minority classes (Fear: 1.36, Love: 1.38) more heavily than the majority classes (Anger: 0.80, Happy: 0.87, Sadness: 0.88). As an implicit ablation experiment, IndoBERTweet was deliberately left without class weights, isolating the pure contribution of domain-specific pre-training.

To prevent overfitting, Early Stopping was applied to all neural network models with a patience of 5 epochs to make sure the training stopped automatically and the best weights were restored once validation loss stopped improving to prevent overfitting [15]. BiLSTM converged optimally at epoch 23 of a maximum 30, IndoBERT at epoch 4 of a maximum 25, and IndoBERTweet at epoch 1 of a maximum 25.

H. Evaluation Metrics and Statistical Significance Testing

The primary evaluation metric is the Macro Average F1-Score, which weights every class equally regardless of sample size. This keeps prediction performance on the minority classes (Fear and Love) from being masked by majority-class dominance, as can happen with ordinary accuracy.

TABLE III. MODELS PERFROMANCE COMPARASION

Model	Anger	Fear	Happy	Love	Sadness	Macro F1
CNB	0.73	0.72	0.65	0.75	0.61	0.69
SVM	0.69	0.75	0.62	0.75	0.52	0.67
Random Forest	0.65	0.72	0.61	0.71	0.49	0.64
BiLSTM + FastText	0.69	0.77	0.60	0.68	0.40	0.63
IndoBERT	0.75	0.75	0.69	0.73	0.59	0.70
IndoBERTweet	**0.81**	**0.78**	**0.74**	**0.80**	**0.68**	**0.76**

To confirm that performance differences between models are not simply mathematical coincidence (random noise), McNemar's Test was applied to the best-performing pair of models. This 2×2 contingency test suits the comparison of two classifiers on the same test set well, since it focuses on the number of discordant predictions [15].

IV. RESULTS AND DISCUSSION

A. *Overall Performance Comparasion*

The evaluation results for all models on the test set (440 samples) are presented in Table III. Ranked by Macro F1-Score, performance runs from highest to lowest as follows: IndoBERTweet (0.76) > IndoBERT (0.70) > CNB (0.69) > SVM (0.67) > Random Forest (0.64) > BiLSTM (0.63).

Table III shows IndoBERTweet leading not just in the aggregate score but across all five emotion classes individually, without exception. That consistency suggests the model's advantage is systemic rather than the product of one unusually strong class masking weaknesses elsewhere.

B. *Confusion Matrix Analysis*

The confusion matrices on Fig.I. show a consistent error patterns across all models. Sadness is consistently the hardest class to get right: CNB misclassified 25 Sadness samples as Anger, BiLSTM misclassified 22 in the same direction, and even IndoBERT still mistook 17 Sadness samples for Anger. This points to substantial semantic overlap between anger and sadness in Indonesian-language Twitter text, where both emotions tend to surface through similar word choices.

IndoBERTweet's biggest gain shows up in the Sadness class where its correct predictions rose to 68 of 100 samples (Recall = 0.68), and the clearest evidence that Twitter-specific pre-trained vocabulary helps the model pick up on the informal-language cues that separate sadness from anger. For Fear class, CNB (Precision=0.91) has the highest precision but weak Recall (0.60), while IndoBERTweet balances both (Precision=0.75, Recall=0.82). That balance suggests the Transformer model is better at catching Fear samples even in ambiguous context.

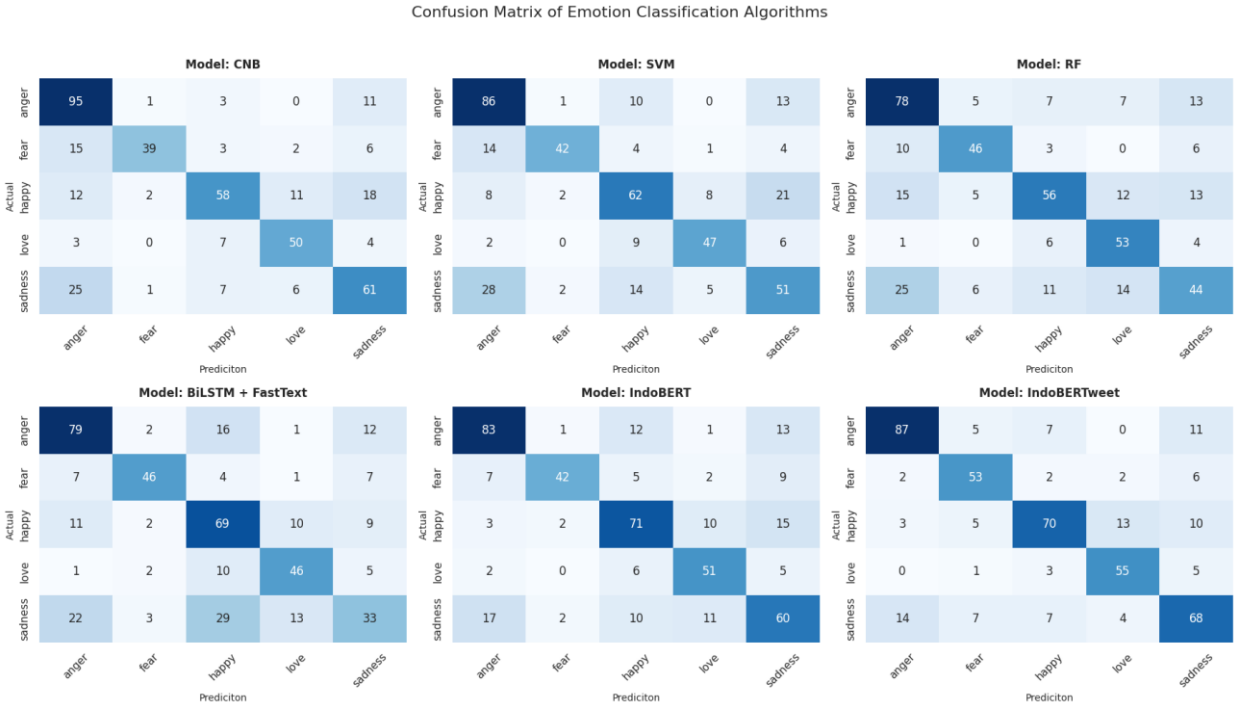

Fig. 1. Confusion Matrices of Emotion Classification Algorithms

C. The Advantage of Domain-Specific Pre-training

IndoBERTweet has fewer parameters than IndoBERT (~110.6 million versus ~123.5 million) [9], yet it still outperforms IndoBERT. This happened because IndoBERT was trained on formal Wikipedia and news text, is inherently limited in how well it maps social media semantics, since its vocabulary distribution diverges sharply from the fine-tuning data. Koto et al. (2021) found that general-domain tokenizers tend to break slang words into meaningless sub-words [8], so the resulting vector representations lose semantic information that matters for emotion detection.

What stands out most is that IndoBERTweet beats IndoBERT consistently across every class. That points to vocabulary initialization having a more deterministic effect on minority-class discrimination than algorithmic fixes [9].

D. Classical Baselines beat Deep Learning

CNB (Macro F1 = 0.69) outperforming BiLSTM (0.63) is itself a notable result. BiLSTM falls short of this baseline because recurrent architectures are inherently data-hungry, where without massive data volumes, the model struggles to learn sequential patterns that generalize, and overfitting follows. BiLSTM's Sadness performance proved this point clearly with Recall = 0.33, even below every classical algorithm tested [6].

Freezing the FastText weights (trainable=False) to guard against overfitting compounded the problem. FastText can synthesize vectors for OOV words through character n-grams [13], but frozen weights cannot adapt, which limited the model's ability to adjust its semantic representations to the slang-heavy vocabulary distribution in the EmoT dataset.

E. Statistical Significance

To make sure that IndoBERTweet outperform CNB is not just random variation in the test sample, McNemar's Test was applied to both models' prediction vectors on the same test set. The test produced a Chi-Square statistic of 7.3772 with a p-value of 0.0066, well below the standard significance threshold of $\alpha = 0.05$, which meant the null hypothesis (H0: both models have equal error rates) is decisively rejected. The performance gap between IndoBERTweet and CNB is real and statistically significant, not a mathematical coincidence [15].

V. Limitations

This study has several limitations. First, freezing the FastText weights (trainable=False) prevented overfitting effectively, but it also kept BiLSTM from adapting its semantic representations to new slang vocabulary appearing in EmoT. A trainable=True configuration paired with more aggressive regularization might have produced different results. Second, the relatively small dataset (3,521 training samples) limited how far deeper, train-from-scratch Deep Learning architectures could be explored, so the comparison here may understate BiLSTM's actual potential. Third, evaluation was limited to a single dataset (EmoT) in a single language (Indonesian); generalizing these findings to other datasets or languages will require further validation.

VI. Conclusions

This study has presented a comprehensive comparative analysis of Classical Machine Learning, BiLSTM, and Transformer models (IndoBERT and IndoBERTweet) on emotion classification for Indonesian-language Twitter text, where noise and class imbalance make the task genuinely difficult. There are three main findings emerge from this work.

First, IndoBERTweet stands out as the most adaptive model, reaching a Macro F1-Score of 0.76 and leading across all five emotion classes; McNemar's Test confirms this advantage is statistically significant ($\chi^2 = 7.38$, $p = 0.0066$). Second, how closely the pre-training data domain matches the fine-tuning data distribution has a far more deterministic effect on classification accuracy than model parameter size or algorithmic fixes. Third, under small, imbalanced dataset conditions, CNB outperforms BiLSTM as a baseline, confirming that sequential Deep Learning architectures need substantially more data than efficiently designed probabilistic models do before they can pull ahead.

CNB is the more sensible choice where computation and data are limited. Where top performance on social media text matters most, IndoBERTweet is the better fit. Future work might explore data augmentation to handle the dataset's small size.

Appendix

1) Code of the experiment and app: https://github.com/SauHin/emotion-classification
2) IndoBERTweet tuned model: https://drive.google.com/drive/folders/1VTyIFin-PXVaVihwbKjefcThcHsjoUi1?usp=sharing.
3) Presentation slides: https://docs.google.com/presentation/d/17bTvMr_fN-HA2WinnnhCtukm4bVKW4ozCZNlyuf1Q2E/edit?usp=sharing
4) App and Presentation demo video: https://www.youtube.com/watch?v=1jC9Udfpm98

References

[1] H. F. Karim and A. P. Wibowo, "Kinerja Metode Fine-Tuning IndoBERT untuk Klasifikasi Emosi Multi-Kelas pada Teks Informal Bahasa Indonesia," Bulletin of Computer Science Research, vol. 6, no. 1, pp. 63-74, Dec. 2025, doi: 10.47065/bulletincsr.v6i1.850.

[2] V. A. Fitri, R. Andreswari, and M. A. Hasibuan, "Sentiment Analysis of Social Media Twitter with Case of Anti-LGBT Campaign in Indonesia using Naïve Bayes, Decision Tree, and Random Forest Algorithm," Procedia Computer Science, vol. 161, pp. 765-772, 2019, doi: 10.1016/j.procs.2019.11.181.

[3] N. A. Salsabila, Y. A. Winatmoko, A. A. Septiandri, and A. Jamal, "Colloquial Indonesian Lexicon," in *2018 International Conference on Asian Language Processing (IALP)*, 2018, pp. 226-229, doi: 10.1109/ialp.2018.8629151.

[4] B. Wilie *et al.*, "IndoNLU: Benchmark and Resources for Evaluating Indonesian Natural Language Understanding," in *Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing*, 2020, pp. 843-857, doi: 10.18653/v1/2020.aacl-main.85.

[5] A. Anas Qolbu, N. Fitriyati, and N. Inayah, "Performa Naïve Bayes, SVM, dan IndoBERT pada Analisis Sentimen Twitter IndiHome dengan Strategi Penanganan Data Tidak Seimbang," *Jurnal Fourier*, vol. 14, no. 1, pp. 29–44, Oct. 2025, doi: 10.14421/fourier.2025.141.29-44.

[6] V. Anggraini *et al.*, "A Comparative Analysis of Machine Learning and Deep Learning Models for Tweet Sentiment Classification: A Case Study on the Sentiment140 Dataset," *arXiv preprint arXiv:2605.04888*, May 2026, doi: 10.48550/arXiv.2605.04888.

[7] F. Koto, A. Rahimi, J. H. Lau, and T. Baldwin, "IndoLEM and IndoBERT: A Benchmark Dataset and Pre-trained Language Model for Indonesian NLP," in *Proceedings of the 28th International Conference on Computational Linguistics*, 2020, pp. 757-770, doi: 10.18653/v1/2020.coling-main.66.

[8] F. Koto, J. H. Lau, and T. Baldwin, "IndoBERTweet: A Pretrained Language Model for Indonesian Twitter with Effective Domain-Specific Vocabulary Initialization," in *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, 2021, pp. 10660–10668, doi: 10.18653/v1/2021.emnlp-main.833.

[9] C. Shaw, P. LaCasse, and L. Champagne, "Exploring emotion classification of indonesian tweets using large scale transfer learning via IndoBERT," *Social Network Analysis and Mining*, vol. 15, no. 1, Art. no. 22, Mar. 2025, doi: 10.1007/s13278-025-01439-6.

[10] C. H. Yutika, Adiwijaya, and S. A. Faraby, "Analisis Sentimen Berbasis Aspek pada *Review* Female Daily Menggunakan TF-IDF dan Naïve Bayes," *Jurnal Media Informatika Budidarma*, vol. 5, no. 2, pp. 422-430, Apr. 2021, doi: 10.30865/mib.v5i2.2845.

[11] P. A. Wardhana and C. I. Ratnasari, "A Comparative Study of Naïve Bayes, SVM, and BiLSTM for Indonesian Tweet Gender Classification," *bit-Tech*, vol. 8, no. 2, pp. 2869–2879, Dec. 2025, doi: 10.32877/bt.v8i2.3405.

[12] T. Syafrudin, A. Hermawan, and D. Avianto, "Analisis Kinerja Support Vector Machine dan Naïve Bayes Classifier dalam Klasifikasi Sentimen dan Emosi pada Umpan Balik Mahasiswa terhadap Kinerja Dosen," Jurnal Ilmiah Digital of Information Technology (DIGIT), vol. 15, no. 2, pp. 34-45, 2025, doi: 10.51920/jd.v15i2.442.

[13] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov, "Enriching Word Vectors with Subword Information," *Transactions of the Association for Computational Linguistics*, vol. 5, pp. 135-146, Jun. 2017, doi: 10.1162/tacl_a_00051.

[14] J. Bergstra and Y. Bengio, "Random search for hyper-parameter optimization," *Journal of machine learning research.*, vol. 13, no. 1, pp. 281–305, Feb. 2012. Available: http://www.jmlr.org/papers/volume13/bergstra12a/bergstra12a.pdf [Accessed: May 26, 2026].

[15] T. G. Dietterich, "Approximate Statistical Tests for Comparing Supervised Classification Learning Algorithms," *Neural Computation*, vol. 10, no. 7, pp. 1895-1923, Oct. 1998, doi: 10.1162/089976698300017197.