COMPILER
Vol. 14, No. 1, May 2025, pp.1-8
ISSN: 2252-2403 (Print), ISSN: 2549-2403 (Online)
Aaccredited Fourth Grade by Kemenristekdikti, Decree No:200/M/KPT/2020
DOI: 10.28989/compiler.v14i1.2965

Enhancing Sentiment and Emotion Classification with LSTM-Based Semi-Supervised Learning

Rochmat Husaini [1,*], Nur Heri Cahyana [2], Wisnalmawati [3], Tri Mardiana [4], Yuli Fauziah [5]
[1,2,5]Department of Informatics, Faculty of Industrial Technics, Universitas Pembangunan Nasional "Veteran' Yogyakarta, Indonesia
[3,4]Department of Management, Faculty of Economic and Business, Universitas Pembangunan Nasional "Veteran' Yogyakarta, Indonesia

Article Info

Article history:

Received February 18, 2025
Accepted April 24, 2025
Published June 13, 2025

Keywords:

Semi-supervised Learning
LSTM
Sentiment Analysis
Emotion Classification
Emotion Analysis

ABSTRACT

The evolution of sentiment analysis has increasingly relied on semi-supervised learning (SSL) models, particularly due to their efficiency in utilizing large amounts of unlabeled data. This study employed four Indonesian datasets—sentiment datasets, emotion dataset and hate speech dataset. The LSTM model was trained using labeled data and used to generate pseudo-labels for unlabeled data across three iterations. The performance of the pseudo-labels was evaluated using Random Forest, Logistic Regression, and Support Vector Machine (SVM). The LSTM model demonstrated varying effectiveness across different datasets. For the sentiment dataset, LSTM achieved an accuracy of 70.23%, slightly lower than Random Forest but higher than Logistic Regression and SVM. In the sentiment dataset, LSTM's accuracy was 86.12%, showing strong performance but slightly below Random Forest and Logistic Regression. The emotion dataset revealed similar performance across models, while the Hate Speech dataset saw LSTM perform well with an accuracy of 86.49%. The results indicate that while LSTM-based SSL can effectively generate pseudo-labels and enhance model performance. This study underscores the need for further research into optimizing pseudo-labeling techniques and exploring advanced NLP models to improve sentiment and emotion analysis in diverse languages.

Corresponding Author:

Rochmat Husaini,
Department of Informatics, Faculty of Industrial Technics,
Universitas Pembangunan Nasional "Veteran' Yogyakarta, Indonesia,
Jl. Padjajaran Jl. Ring Road Utara No.104, Ngropoh, Condongcatur, Kec. Depok, Kabupaten Sleman,
Daerah Istimewa Yogyakarta 55283.
Email: *husaini@upnyk.ac.id

1. INTRODUCTION

Semi-supervised learning (SSL) has gained significant attention in sentiment analysis due to its ability to leverage large volumes of unlabelled data, which are cheaper and easier to obtain than labelled data [1]. Within SSL frameworks, Long Short-Term Memory (LSTM) networks have shown promise for their capability to capture sequential and contextual patterns in text, making them well-suited for sentiment and emotion classification tasks [2].

This study evaluates an LSTM-based SSL model for generating pseudo-labels on Indonesian-language sentiment datasets. These pseudo-labels are then tested using three conventional machine learning algorithms—Random Forest, Logistic Regression, and Support Vector Machine (SVM)—to assess their effectiveness in enhancing classification performance.

SSL is especially valuable when labelled data is scarce or expensive to obtain. By learning from unlabelled data, SSL can improve model accuracy without full reliance on manual annotation [3]. Despite extensive research in English-language sentiment analysis, there remains limited exploration of SSL for

Indonesian, a language with complex morphology, informal expressions, and regional dialects [4]. Addressing these linguistic challenges requires tailored approaches. To fill this gap, we focus on four Indonesian datasets: two sentiment datasets (3-class), one emotion dataset (6-class), and one hate speech dataset (binary class), enabling the evaluation of LSTM's generalization in different scenarios. Although LSTM is widely used in NLP, its role in generating reliable pseudo-labels through SSL remains underexplored.

One of the main challenges in SSL is the risk of error propagation, where incorrect pseudo-labels degrade model performance over time . The quality of pseudo-labels is also influenced by the distribution and representativeness of the unlabelled data. To mitigate these issues, we use diverse datasets and optimize hyperparameters during training.

Previous studies on SSL for sentiment analysis show varying results. For example, research on Chinese datasets (COAE2014 and COAE2015) reported accuracies up to 0.79[5] , while a hotel review dataset reached 0.841 [6]. Another study combining CNN and word embeddings achieved an F1 score of 89% on Algerian and Arabizi datasets [7]. The AraSenCorpus framework improved accuracy from 80.37% to 87.4% on the SemEval 2017 dataset [8]. Meanwhile, other research found Random Forest to be effective in sentiment and emotion analysis [9], and methods based on polarity scores or CNNs yielded high accuracy in Turkish sentiment analysis [10]. These findings affirm the potential of SSL and the importance of selecting appropriate algorithms and data representations tailored to language and task-specific contexts.

Thus, this study aims to test the ability of LSTM to generate accurate pseudo-labels and evaluate its effectiveness in various sentiment and emotion contexts. We hope that this research will make a significant contribution to the literature, particularly in the context of sentiment analysis using Indonesian-language datasets, which remains underexplored. Therefore, this study focuses on evaluating the performance of an SSL approach using an LSTM model for sentiment and emotion classification on Indonesian-language datasets. By doing so, it aims to bridge the research gap and assess whether pseudo-labeling strategies can effectively improve model accuracy in linguistically complex environments.

2. RESEARCH METHOD

2.1. Dataset Description

This study employs four Indonesian language datasets to evaluate the performance of a semi-supervised learning model based on Long Short-Term Memory (LSTM):

1. Sentiment Dataset from https://github.com/ridife/dataset-idsa, already researched in [11][12] : Consists of three sentiment classes: positive, negative, and neutral. It is used to train and test the model's ability to classify general opinions. The dataset contains a mix of formal and informal Indonesian expressions, which can hinder consistent feature representation. Additionally, the presence of context-dependent neutral statements makes classification non-trivial.
2. Sentiment Dataset from https://github.com/IndoNLP/indonlu/tree/master/dataset/smsa_doc-sentiment-prosa, already used in [13] and [14] : Contains three sentiment classes (positive, negative, and neutral) and is used to test the model on more general sentiment analysis tasks. Sentences in this dataset are often short, contextually ambiguous, and written in everyday Indonesian, which lacks standardized grammar or punctuation. This increases the difficulty for models to capture sentiment polarity accurately.
3. Emotion Dataset from https://github.com/IndoNLP/indonlu/tree/master/dataset/emot_emotion-twitter. Already used in [13] and [14] : comprises six classes representing different emotions (e.g., anger, sadness, joy, surprise, fear, disgust). This dataset aims to evaluate the model's performance in detecting more complex emotional variations in text. Emotion classification is inherently more complex due to overlapping semantics among emotional states. Moreover, the use of sarcasm, slang, emojis, and code-mixing (Indonesian English) in tweets further complicates accurate emotion identification.
4. Hate Speech Dataset from https://github.com/okkyibrohim/id-multi-label-hate-speech-and-abusive-language-detection, already used in [15]: Contains two classes, "hate" (hate speech) and "non-hate." This dataset is employed to assess the model's ability to detect offensive or inappropriate content. Detecting hate speech is particularly difficult due to the implicit nature of offensive content, cultural and contextual dependencies, and the subtle difference between criticism and hate. The presence of multiple overlapping labels in the original version of the dataset (multi-label setting) adds another layer of complexity for binary classification.

2.2 Data Preprocessing

Data preprocessing is a crucial step in preparing raw text data for effective sentiment analysis. This process involves cleaning and transforming the data to enhance the model's ability to learn meaningful patterns [16]. The preprocessing steps applied in this study include:

1. Removal of URLs, punctuation, numbers, and special characters: This step cleans the text by removing elements that are irrelevant for sentiment analysis, such as hyperlinks, punctuation marks, numbers, and special characters.
2. Tokenization: This process splits the text into individual words (tokens) to facilitate easier processing by the machine learning models.
3. Padding: To ensure compatibility with the LSTM model, all token sequences are standardized to the same length through padding.
4. Stopword Removal: Common words (e.g., "that," "and," "in") that do not significantly contribute to the sentiment of the text are removed to reduce noise and improve model performance.

It is important to note that stemming, which involves reducing words to their base or root form, is not applied in this study. This decision is made to preserve the context and meaning of the words, which is particularly crucial for sentiment analysis tasks where subtle differences in word forms can significantly affect the results.

2.3. Semi-Supervised Learning Model with LSTM

The semi-supervised learning model based on LSTM is trained using labelled data and then applied to generate pseudo-labels from unlabeled data. These pseudo-labels are model predictions with certain probabilities and are used to enrich the labelled dataset in subsequent iterations. The process is as follows:

1. Initial Training: The LSTM model is trained on a subset of labelled data.
2. Pseudo-Labelling: The trained model predicts labels on the unlabelled data.
3. Incorporation of Pseudo-Labels: Only predictions with high probability (above a certain threshold) are added to the labelled dataset for the next iteration.
4. Re-Training Iterations: The model is re-trained with the updated labelled dataset until no more unlabelled data meet the criteria for addition.
5. Pseudo Label Evaluation
6. The pseudo-labels generated by the LSTM model are tested using three kinds of machine learning models:
   a. Random Forest: An ensemble model that constructs multiple decision trees to achieve more accurate predictions and prevent overfitting.
   b. Logistic Regression: A simple and fast classification model used as a baseline to compare the performance of pseudo-labels.
   c. Support Vector Machine (SVM): A model that seeks the best hyperplane to separate classes in the data, often used in classification tasks with strong performance on high-dimensional data.

2.4. Pseudocode: SSL with LSTM

The following pseudocode outlines the implementation of Semi-Supervised Learning (SSL) utilizing Long Short-Term Memory (LSTM) networks. SSL is a machine learning technique that leverages unlabeled data alongside labelled data to enhance model performance. The pseudocode is shown in Figure 1.

```
Set best_units = 64 (optimal number of LSTM units)
Set best_learning_rate = 0.001 (optimal learning rate)
Define Function to Create LSTM Model:

Loop for 3 Semi-Supervised Learning Iterations:

FOR iteration i IN range(3):
Train model using labeled data (`X_train`, `y_train`) with
100 epochs, batch size 32
Predict labels for unlabeled data (`X_unlabeled`)

Identify confident data indices:
`confident_indices`←indices where pred - 0.5 > 0.4
`confident_labels` ← convert to binary labels (0 or 1)

Append newly labeled data to training data:
`X_train`←add`X_unlabeled[confident_indices]`to `X_train`
`y_train` ← add `confident_labels` to `y_train`

Remove labeled data from unlabeled data:
`X_unlabeled`←delete `X_unlabeled[confident_indices]`

Evaluate model on validation data (X_val, y_val)
Print final accuracy and total time spent.
```

Figure 1. Pseudocode of The SSL Model

The pseudocode (Figure 1) illustrates the process of employing an LSTM model for SSL across three iterations. In each iteration, the model is trained using labeled data and subsequently used to predict labels for unlabeled data. Confident data indices, where predictions exceed a predefined threshold, are identified and converted into binary labels. These pseudo-labeled data are then appended to the training set, and the labeled data are removed from the unlabeled set to prevent re-use. After each iteration, the model is evaluated on validation data to assess its performance. This iterative process is repeated for three cycles to incrementally optimize the model. Ultimately, the final model accuracy and total training time are reported. This approach ensures that the model progressively utilizes an increasing amount of unlabeled data, thereby enhancing accuracy and reliability in sentiment classification or other relevant tasks.

Based on Figure 1, the parameters used in the LSTM-based semi-supervised learning model were selected to balance performance and computational efficiency. The number of LSTM units was set to 64, as this configuration is sufficient to capture sequential patterns in text data without causing overfitting or excessive computational demand. A learning rate of 0.001 was chosen for its stability and effectiveness when used with optimizers such as Adam, enabling reliable convergence without gradient explosions. Training was conducted over 100 epochs with a batch size of 32, a combination commonly used in natural language processing tasks for its stability and computational practicality. The model performed three semi-supervised learning iterations to gradually expand the labeled dataset while minimizing the risk of error propagation from incorrect pseudo-labels. Only predictions with high confidence (probability greater than 0.9 or less than 0.1) were used as pseudo-labels, ensuring that the model incorporated only the most reliable unlabeled data into subsequent training rounds.

2.5. Evaluation and Validation Methods

To evaluate the model's performance, a stratified k-fold cross-validation method is employed:
Stratified k-Fold Cross-Validation: The dataset is split into k balanced subsets, each with the same class proportions. The model is trained on k-1 subsets and tested on the remaining subset. This process is repeated k times so that each subset serves as the test data once.

Confusion Matrix: A matrix that represents the performance of the classification model by showing the number of true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN). The confusion matrix is used to calculate accuracy as formula (1) [17] .

$$Accuracy = (TP + TN) / (TP + TN + FP + FN) \quad (1)$$

The k-fold cross-validation method provides a comprehensive evaluation of the model's performance by minimizing the bias due to data splitting. Stratified k-fold ensures that each fold has balanced class representation, which is crucial for imbalanced datasets [18]. Evaluation using the confusion matrix and accuracy metrics allows for a thorough assessment of how well the model can classify data overall and handle minority classes that are often difficult to predict.

This approach is expected to provide deep insights into the effectiveness of the LSTM-based semi-supervised model in generating effective pseudo-labels and enhancing model performance in various sentiment classification scenarios using Indonesian language datasets.

3. RESULTS AND DISCUSSIONS

The LSTM model is used to generate pseudo labels for the unlabeled data, with accuracy evaluated on the validation set. In this study, LSTM is chosen due to its capability to capture temporal and contextual relationships in textual data, which is crucial for sentiment and emotion analysis. The pseudo labels generated by the LSTM model are subsequently tested using three traditional machine learning algorithms: Random Forest, Logistic Regression, and Support Vector Machine (SVM). Before testing, the pseudo-labeled data is converted into vector representations using the TF-IDF (Term Frequency-Inverse Document Frequency) technique to capture the essential features of the text. The evaluation results present a comparison of the accuracy and performance of each algorithm on the pseudo-labeled data, providing insights into the effectiveness of integrating semi-supervised learning models with different machine learning algorithms.

Performance comparison analysis includes a discussion of the evaluation results and performance comparison between the deep learning model (LSTM) and traditional machine learning algorithms (Random Forest, Logistic Regression, and SVM). The aim is to understand the strengths and weaknesses of each approach in the context of sentiment and emotion data labelling and to determine the most effective model for various types of test data.

3.1. Result

For each test dataset, the data is divided into 20% as labeled training data and 80% as unlabeled data. Specifically, the labeled training data is further split, with 50% used for model training and the remaining 50% reserved for validation. This division is designed to maximize the use of labeled data in training the semi-supervised learning model while maintaining a sufficient validation set to evaluate model performance during the pseudo-labeling process. The result of evaluation dataset shown in Table 1.

Table 1. Result of evaluation dataset

Dataset	Class Number	LR	RF	SVM	LSTM
Sentiment Dataset 1	3	0.6968	0.6340	0.7045	0.7023
Sentiment Dataset 2	3	0.8794	0.8361	0.8802	0.8612
Emotion Dataset	6	0.6969	0.6747	0.6996	0.6953
Hate Speech	2	0.8514	0.8011	0.8697	0.8649

3.2. Discussions

The evaluation results (Table 1) of the pseudo-labeled datasets generated using the LSTM-based semi-supervised learning (SSL) model show varied performances across different datasets and machine learning algorithms, revealing key insights into the effectiveness of the SSL approach for sentiment and emotion classification.

For the Sentiment dataset 1 (3 classes: positive, negative, neutral), LSTM achieved an accuracy of 0.7023, which is slightly lower than Random Forest (0.7045) but higher than Logistic Regression (0.6968) and SVM (0.6340). This suggests that while LSTM effectively captures temporal and contextual information, Random Forest performs slightly better for this dataset. The lower performance of Logistic Regression and SVM may be due to their limited ability to handle complex text relationships.

In the Sentiment dataset 2 (3 classes), LSTM obtained an accuracy of 0.8612, which is slightly lower than Random Forest (0.8802) and Logistic Regression (0.8794), but higher than SVM (0.8361). These results indicate that traditional machine learning algorithms, particularly Random Forest, can achieve competitive or superior performance for this dataset, potentially due to its simpler feature relationships.

The Emotion dataset (6 classes) presented a more complex challenge. Here, LSTM achieved an accuracy of 0.6953, comparable to Logistic Regression (0.6969) and Random Forest (0.6996), while SVM performed slightly lower at 0.6747. This suggests that the LSTM model performs similarly to traditional models in nuanced emotion classification tasks, with no single model significantly outperforming others due to data variability.

For the Hate Speech dataset (2 classes: hate, non-hate), LSTM achieved an accuracy of 0.8649, close to Random Forest (0.8697) and significantly higher than Logistic Regression (0.8514) and SVM (0.8011). This indicates that LSTM is effective in identifying hate speech, which involves subtle contextual cues. Random Forest's high performance also suggests that ensemble methods can be robust in binary classification tasks with simpler decision boundaries.

The LSTM model demonstrates strong performance across various datasets, especially for tasks requiring complex temporal and contextual understanding, such as emotion classification and hate speech detection. However, its performance is not always superior to traditional algorithms, which suggests the model choice should be aligned with the dataset's characteristics and classification requirements.

Traditional machine learning algorithms, particularly Random Forest, often perform competitively with LSTM on simpler sentiment classification tasks, suggesting that simpler models might be preferable due to their lower computational cost and interpretability.

Performance variability across datasets highlights the importance of dataset characteristics in model selection. Complex datasets with more classes may benefit from LSTM's nuanced capabilities, whereas simpler datasets might not require such complexity.

The SSL approach with LSTM effectively generates pseudo labels that enhance the training of other machine learning algorithms, demonstrating the value of pseudo labels in expanding labelled data and improving model performance.

In conclusion, while the semi-supervised LSTM model is a robust tool for generating pseudo labels in sentiment and emotion analysis, integrating it with traditional algorithms should be considered based on dataset characteristics and specific tasks. Further research could focus on optimizing pseudo-labeling techniques and enhancing feature representations to improve overall model performance.

Future research should focus on enhancing the pseudo-labelling process by incorporating uncertainty-aware strategies, such as entropy-based selection or Monte Carlo dropout, to improve the quality of pseudo-labels used in training. Additionally, integrating pre-trained language models (PLMs) like IndoBERT or IndoGPT within a semi-supervised framework could significantly boost performance, especially for context-sensitive tasks such as emotion and hate speech detection. Applying text data augmentation methods—such as synonym replacement or back-translation—may further enrich the training data and enhance model generalization. Ensemble learning approaches combining pseudo-labelled data from multiple models, as well as empirical investigations into varying proportions of labelled versus unlabelled data, would provide deeper insights into the scalability and robustness of semi-supervised learning. Lastly, including error analysis and model interpretability techniques (e.g., SHAP, LIME) alongside domain-specific evaluations in low-resource settings could ensure broader applicability and a better understanding of model behaviour in real-world scenarios.

4. CONCLUSION

This study evaluated the effectiveness of a semi-supervised learning (SSL) approach using an LSTM model for generating pseudo-labels in sentiment and emotion analysis tasks across multiple Indonesian-language datasets. The findings suggest that the LSTM-based SSL model is a robust tool for generating pseudo-labels, effectively utilizing a small amount of labelled data along with a large amount of unlabeled data to enhance the overall model performance. The results demonstrated that the pseudo-labels generated by the LSTM model can achieve competitive performance when tested with traditional machine learning algorithms such as Random Forest, Logistic Regression, and Support Vector Machine (SVM). Future research should focus on refining pseudo-labeling techniques, such as dynamically adjusting confidence thresholds or using ensemble methods to minimize error propagation. Integrating advanced NLP models like transformers could improve context understanding and label accuracy. Expanding the approach to other languages and domains and comparing SSL methods with various deep learning models on larger datasets, would provide broader insights. Additionally, applying and testing the model in real-world scenarios could offer practical feedback and enhance the robustness of sentiment analysis tools.

ACKNOWLEDGEMENTS

This work was supported by a research grant from the Universitas Pembangunan Nasional "Veteran" Yogyakarta**.**

REFERENCES

[1] V. L. Shan Lee, K. H. Gan, T. P. Tan, and R. Abdullah, "Semi-supervised learning for sentiment classification using small number of labeled data," *Procedia Computer Science*, vol. 161, pp. 577–584, 2019. [Online]. Available: https://doi.org/10.1016/j.procs.2019.11.159

[2] P. Sudhir and V. D. Suresh, "Comparative study of various approaches, applications and classifiers for sentiment analysis," *Global Transitions Proceedings*, vol. 2, no. 2, pp. 205–211, 2021. [Online]. Available: https://doi.org/10.1016/j.gltp.2021.08.004

[3] A. S. Aribowo, H. Basiron, and N. F. A. Yusof, "Semi-supervised learning for sentiment classification with ensemble multi-classifier approach," *International Journal of Advances in Intelligent Informatics*, vol. 8, no. 3, pp. 349–361, 2022.

[4] T. N. Fatyanosa and F. A. Bachtiar, "Classification method comparison on Indonesian social media sentiment analysis," in *Proc. 2017 Int. Conf. Sustainable Information Engineering and Technology (SIET)*, 2018, pp. 310–315.

[5] Y. Li, Y. Lv, S. Wang, J. Liang, J. Li, and X. Li, "Cooperative hybrid semi-supervised learning for text sentiment classification," *Symmetry*, vol. 11, no. 2, pp. 1–17, 2019.

[6] D. A. K. Khotimah and R. Sarno, "Sentiment analysis of hotel aspect using probabilistic latent semantic analysis, word embedding and LSTM," *International Journal of Intelligent Engineering and Systems*, vol. 12, no. 4, pp. 275–290, 2019.

[7] I. Guellil, F. Azouaou, and F. Chiclana, "ArAutoSenti: automatic annotation and new tendencies for sentiment classification of Arabic messages," *Social Network Analysis and Mining*, vol. 10, no. 1, 2020. [Online]. Available: https://doi.org/10.1007/s13278-020-00688-x

[8] A. Al-Laith, M. Shahbaz, H. F. Alaskar, and A. Rehmat, "AraSenCorpus: A semi-supervised approach for sentiment annotation of a large Arabic text corpus," *Applied Sciences (Switzerland)*, vol. 11, no. 5, pp. 1–19, 2021.

[9] Y. Fauziah, S. Saifullah, and A. S. Aribowo, "Design text mining for anxiety detection using machine learning based-on social media data during COVID-19 pandemic," *Proc. LPPM UPN "Veteran" Yogyakarta Conf. Series 2020 – Engineering and Science Series*, vol. 1, no. 1, pp. 253–261, 2020.

[10] C. R. Aydin and T. Güngör, "Sentiment analysis in Turkish: Supervised, semi-supervised, and unsupervised techniques," *Natural Language Engineering*, vol. 27, no. 4, pp. 455–483, 2021.

[11] W. Maharani, “Sentiment analysis during Jakarta flood for emergency responses and situational awareness in disaster management using BERT,” in *Proc. 2020 8th Int. Conf. Information and Communication Technology (ICoICT)*, 2020.

[12] S. Khomsah, N. H. Cahyana, and A. S. Aribowo, “Hyperparameter tuning of semi-supervised learning for Indonesian text annotation,” *International Journal of Advanced Computer Science and Applications*, vol. 14, no. 9, pp. 250–256, 2023.

[13] H. Jayadianti, W. Kaswidjanti, A. Tri, and S. Saifullah, “Sentiment analysis of Indonesian reviews using fine-tuning IndoBERT and R-CNN,” *ILKOM Jurnal Ilmiah*, vol. 14, no. 3, pp. 348–354, 2022.

[14] H. Ahmadian, T. F. Abidin, H. Riza, and K. Muchtar, “Hybrid models for emotion classification and sentiment analysis in Indonesian language,” *Applied Computational Intelligence and Soft Computing*, vol. 2024, 2024.

[15] M. O. Ibrohim and I. Budi, “Multi-label hate speech and abusive language detection in Indonesian Twitter,” in *Proc. Third Workshop on Abusive Language Online*, 2019, pp. 46–57.

[16] A. S. Aribowo, H. Basiron, N. S. Herman, and S. Khomsah, “An evaluation of preprocessing steps and tree-based ensemble machine learning for analysing sentiment on Indonesian YouTube comments,” *International Journal of Advanced Trends in Computer Science and Engineering*, vol. 9, no. 5, pp. 7078–7086, 2020. [Online]. Available: ttps://www.scopus.com/inward/record.uri?eid=2-s2.085092659939&doi=10.30534%2Fijatcse%2F2020%2F29952020&partnerID=40&md5=92529b57f447b0e2b2c06d43c90bbdc7

[17] S. Khomsah and A. S. Aribowo, “Model semi-supervised learning menggunakan logistic regression untuk anotasi sentimen,” *Open Access Ledger*, vol. 1, no. 4, pp. 171–178, 2022.

[18] W. Wijiyanto, A. I. Pradana, S. Sopingi, and V. Atina, “Teknik K-fold cross validation untuk mengevaluasi kinerja mahasiswa,” *Jurnal Algoritma*, vol. 21, no. 1, pp. 239–248, 2024.