Public Sentiment Analysis on the Corruption Issue of the Whoosh High-Speed Train Project Using TF-IDF and IndoBERT

David Christian Nathaniel
Department of Computer Science
Padjadjaran University
Sumedang, West Java, 45363 Indonesia
david23002@mail.unpad.ac.id

Hafizh Fadhl Muhammad
Department of Computer Science
Padjadjaran University
Sumedang, West Java, 45363 Indonesia
hafizh23006@mail.unpad.ac.id

Farhan Zia Rizky
Department of Computer Science
Padjadjaran University
Sumedang, West Java, 45363 Indonesia
farhan23012@mail.unpad.ac.id

Gideon Tamba
Department of Computer Science
Padjadjaran University
Sumedang, West Java, 45363 Indonesia
gideon23003@mail.unpad.ac.id

***Abstract*—The Jakarta-Bandung High-Speed Rail project, also known as "Whoosh," has become a hot topic of discussion among Indonesians, particularly regarding the alleged corruption issue widely discussed on YouTube. This study aims to analyze public sentiment on the issue to understand public opinion trends in a dataset with an imbalanced class distribution. The dataset consists of approximately 900 comments collected through scraping techniques from various related YouTube videos. The research methodology includes automatic labeling using the manually validated Gemini API, text preprocessing, and sentiment classification. This study compares the performance of a modern Natural Language Processing (NLP) model, namely IndoBERT (indobenchmark/indobert-base-p2), with a traditional baseline model using TF-IDF (Logistic Regression, Support Vector Machine, Multinomial Naive Bayes) feature extraction. The experimental results are expected to determine the effectiveness of the IndoBERT fine-tuning method compared to classical methods in handling limited and imbalanced Indonesian text data.**

Keywords—Sentiment Analysis; Imbalance Dataset; IndoBERT; Whoosh High Speed Train; NLP; YouTube

I. Introduction

Social media, particularly YouTube, has now transformed into one of the main platforms for the public to voice their opinions, criticisms, and aspirations regarding national issues. One topic that has garnered significant public attention is the construction of the Jakarta-Bandung High-Speed Railway, or *'Whoosh'*. Amidst the infrastructure advancements it offers, this project has not been without controversy, including allegations of corruption that have sparked widespread debate in the comments sections of news and opinion videos. Understanding public sentiment on this issue is crucial for gauging the level of public trust in the transparency of government projects.

However, analysing public opinion manually is a challenge in itself, given the large volume of unstructured data. In this study, there were approximately 1,000 unique comments that needed to be analysed after undergoing a data cleaning process. In addition, another challenge faced is the imbalance in the number of comments between positive, negative, and neutral sentiments, which often makes it difficult for classification models to accurately predict minority classes. Therefore, the application of Text Mining and Natural Language Processing (NLP) techniques is necessary to extract information efficiently.

Previous studies have applied machine learning methods to sentiment analysis. Idris and Mustofa [1] successfully analysed sentiment towards the use of the Shopee application using the *Support Vector Machine (SVM)* algorithm with an accuracy rate of 98%. Meanwhile, another study by Hapsari and Indriyanti [2] applied the *Random Forest* algorithm to analyse sentiment on digital wallet applications, with the highest accuracy reaching 89.02% in the LinkAja case study.

Although traditional methods show good performance, these approaches often have limitations in capturing complex semantic contexts in natural language, especially in non-standard Indonesian on social media. To address this gap, this study proposes the use of a modern NLP approach using the *IndoBERT* model, which will be compared with the classic TF-IDF-based approach. This study aims to build a robust sentiment analysis system and evaluate whether transformer-based models can provide better performance than baseline models (*Logistic Regression, SVM, and Multinomial Naive Bayes*) on unbalanced datasets.

II. Literature Review

This literature review aims to provide the theoretical foundation supporting sentiment analysis research on Indonesian-language text data. The discussion covers the basic concepts of sentiment analysis, characteristics of social media data, TF–IDF-based text representation methods, and the use of modern transformer-based Natural Language Processing models such as IndoBERT.

In addition, this section discusses the class imbalance problem that commonly occurs in sentiment analysis of public issues and its impact on classification model performance. Through a review of previous studies, this literature review helps position the present work and explains the rationale for comparing classical machine learning models and transformer-based models in this study.

A. Sentiment Analysis

Sentiment analysis is a text mining task that focuses on grouping opinions in text into specific classes (e.g., positive, neutral, and negative). In social media comments such as those on YouTube, opinions tend to exhibit high linguistic variability (slang, abbreviations, character repetitions, and mixed punctuation), which requires adequate preprocessing so that the features learned by the model better represent sentiment.

B. Text Mining and General Text Processing Stages

Text classification is a supervised learning process that maps text documents to class labels. In general, a sentiment classification pipeline consists of data collection, labeling, preprocessing, text representation, model training, evaluation, and result interpretation. In the context of public opinion, classification results are not only used to achieve high accuracy but also to understand public perceptions of a particular issue.

C. Indonesian Text Preprocessing

Preprocessing aims to reduce noise and improve text consistency. Common steps include text cleaning, case folding, normalization of non-standard words, tokenization, stopword removal, and stemming. In Indonesian, stemming is important because affixed words can take many forms, which affects feature matching. Indonesian stemming approaches have been widely discussed in prior studies and are often key components in information retrieval and text classification tasks.

D. Text Representation Using TF–IDF

TF–IDF (Term Frequency–Inverse Document Frequency) is a document representation method that assigns weights to terms based on their frequency in a document (TF) and their rarity across the entire corpus (IDF). The key intuition is that terms that frequently appear in a particular document but rarely appear in others tend to be more informative for distinguishing classes. TF–IDF is widely used as a strong baseline for text classification because it is simple, fast, and effective for high-dimensional, sparse data.

E. Text Representation Using TF–IDF

After documents are represented as TF–IDF vectors, several machine learning algorithms are commonly used for text classification, including:

1. *Logistic Regression (LR)***:** A linear model that maps feature combinations to class probabilities. LR is often used as a baseline because it is stable, efficient, and well-suited for high-dimensional sparse data.
2. *Linear Support Vector Machine (Linear SVM):* Linear SVM seeks an optimal separating hyperplane by maximizing the margin between classes. In text classification tasks, SVM is widely regarded as a strong and robust method because it can effectively leverage a large number of relevant features in high-dimensional feature spaces.
3. *Multinomial Naive Bayes (MNB):* MNB models word occurrences as multinomial events and is popular for document classification. Comparative studies of Naive Bayes models for text show that the multinomial approach often performs competitively on many text corpora.

F. Transformer, BERT, and Fine-Tuning

Transformer is an architecture that relies on self-attention mechanisms to learn relationships among tokens without recurrent dependencies, making it efficient and effective for sequence modeling.

BERT (Bidirectional Encoder Representations from Transformers) is a Transformer-based model that is pre-trained to obtain bidirectional language representations (leveraging both left and right context) and can be fine-tuned for various NLP tasks, including sentiment classification, by adding a simple output layer.

G. IndoBERT for Indonesian

For Indonesian, the availability of Indonesian-specific pre-trained models is important so that the model better aligns with local vocabulary, morphology, and language usage. IndoNLU introduces resources and benchmarks for Indonesian NLU and provides pre-trained models (IndoBERT) that can be fine-tuned for downstream tasks such as sentiment classification.

H. LLM-Assisted Data Labeling

In supervised learning, label quality strongly determines model performance. Manual labeling on large-scale comment data is typically costly and time-consuming, so *LLM-assisted labeling* can be used to accelerate initial annotation. Studies evaluating LLMs as annotators indicate that LLMs can help the labeling process, but quality control is still necessary (e.g., sample validation, clear labeling guidelines, and error checking) to maintain label consistency.

I. Class Imbalance in Sentiment Classification

Sentiment data in social media are often imbalanced; for example, neutral/negative classes may be far more dominant than positive. Class imbalance can bias models toward the majority class and make metrics such as accuracy appear high even when performance on minority classes is poor. The imbalanced learning literature discusses various strategies such as resampling, class weighting, and selecting appropriate evaluation metrics.

J. Model Performance Evaluation

Classification evaluation commonly uses a confusion matrix and derived metrics such as *precision, recall, and F1-score*. For multi-class and imbalanced settings, macro-F1 is often considered because it gives equal weight to each class and is therefore more sensitive to performance on minority classes. Systematic analyses of classification performance measures have been discussed in the literature and are widely used as references when selecting suitable metrics.

III. Method

This research was conducted through a series of systematic stages to ensure accurate sentiment analysis results. The research flow began with data collection, data labelling, text pre-processing, and model development and evaluation.

A. Research Workflow

This study follows the stages illustrated in **Figure. 1**. The process begins with problem identification and a literature review, followed by the collection of YouTube comments and sentiment labeling. The data are then processed through a preprocessing stage and organized into two modeling pipelines: (1) a baseline approach using TF–IDF feature extraction and classical classification algorithms (Logistic Regression, Linear SVM, and Multinomial Naive Bayes), and (2) a transformer-based approach using IndoBERT tokenization and fine-tuning. All models are subsequently evaluated using the same performance metrics, interpreted, and concluded with findings and recommendations.

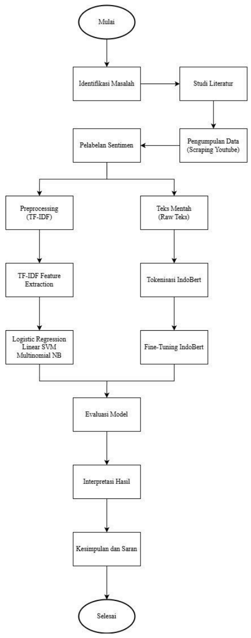


Fig. 1. Research workflow for sentiment analysis (TF–IDF baseline vs. IndoBERT).

B. Data Collection

The research data was taken from the comments section of YouTube videos discussing the alleged corruption in the Whoosh Fast Train project. Data collection was carried out using scraping techniques. After going through a deduplication process, the total dataset used amounted to approximately 900 comments.

C. Data Labeling

This study applied the *LLM-Assisted Labeling* method. The labelling process was carried out using the *Gemini API* to classify comments into three classes: Positive, Neutral, and Negative. Given that the distribution of data in the field tends to be dominated by certain sentiments (imbalanced dataset), the machine labelling results were revalidated to ensure the accuracy of the training data.

D. Preprocessing

This stage aims to clean the text data so that it is ready for processing. The stages include:

1. *Cleaning:* Removing noise elements such as URLs, emojis, numbers, and symbols.
2. *Case Folding:* Converting all letters to lowercase.
3. *Normalization:* Converting non-standard words (slang/abbreviations) into standard Indonesian.
4. *Tokenization:* Breaking sentences into word segments or tokens.
5. *Stopword Removal:* Removing common conjunctions that do not have significant sentiment meaning.
6. *Stemming:* Converting inflected words into their base form using the Sastrawi library.

E. Modeling

This study compares two main approaches:

1. *Model Baseline (TF-IDF):* Uses the *TF-IDF* (Term Frequency-Inverse Document Frequency) feature extraction method combined with classic classification algorithms, namely *Logistic Regression, Linear SVM, and Multinomial Naive Bayes.*
2. *Model Utama (IndoBERT):* Performs fine-tuning on the pre-trained *IndoBERT* model (indobenchmark/indobert-base-p2 variant). This model was chosen because it has been trained on a large Indonesian language corpus and is able to capture sentence context better.

F. Evaluation

The performance of each model was evaluated using a Confusion Matrix to see the details of correct and incorrect predictions. The quantitative metrics used included Accuracy to measure overall accuracy, as well as Precision, Recall, and F1-Score to measure the model's performance on each sentiment class more specifically, especially if there was class imbalance in the dataset.

IV. Results and Discussion

This section presents the results of the experiments and discussion of all stages of the research that has been carried out, starting from data labelling using the LLM-Assisted

Labelling approach, text preprocessing, exploratory data analysis (EDA), to the evaluation of the sentiment classification model's performance. The discussion focuses on data quality analysis, label distribution, and performance comparison between the classic *TF–IDF-based model* and the *IndoBERT* transformer model in handling unbalanced Indonesian sentiment data.

Each subsection in this section is arranged sequentially to show the relationship between the research stages. The data labelling results are analysed first to understand the initial characteristics of the dataset, followed by an evaluation of the preprocessing and EDA results to ensure data quality before modelling. Next, the performance of the baseline model and IndoBERT is compared using relevant evaluation metrics, specifically accuracy and macro-F1, to provide a fair picture of the model's performance on each sentiment class.

The discussion in this section also highlights the limitations that arise due to the imbalance in label distribution and its implications for classification results. Thus, this section not only presents numerical results, but also provides interpretations and critical analyses that support the research conclusions.

A. LLM-Assisted Data Labeling (Gemini)

Sentiment labelling was carried out using the LLM-Assisted Labelling approach with the Gemini model to classify public opinion into three classes: negative, neutral, and positive. The labelling process was made resilient to disruptions through batch processing (25 comments per batch), inter-batch delays (60 seconds), a retry mechanism when rate limits were reached (120 seconds), and checkpoints so that the process could be resumed without having to start over from the beginning.

The labelling results produced a total of 1,000 labelled comments with a highly unbalanced class distribution, namely negative (646), neutral (320), and positive (34). The labels were then converted to numerical values for the modelling stage (negative=0, neutral=1, positive=2). This imbalance was a major factor affecting the model's performance, particularly in its ability to detect the positive class, which was much smaller in number.

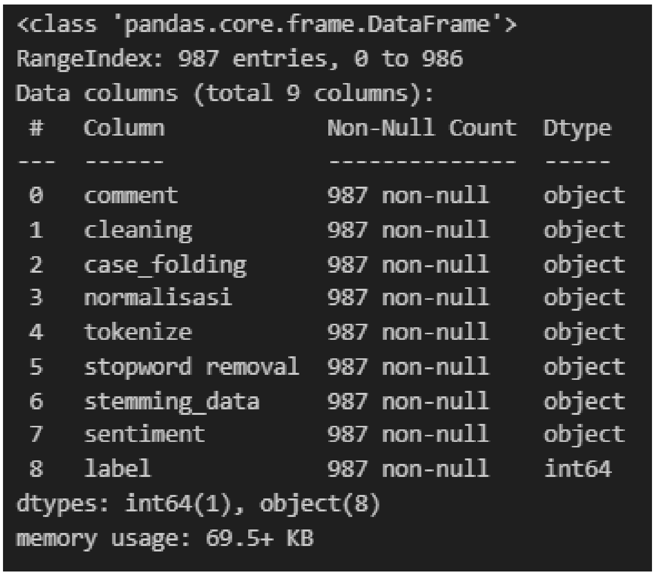
TABLE I
COMPARISON OF DATA DISTRIBUTION AFTER LABELLING AND PREPROCESSING

Dataset Stage	**Total**	**Negative**	**Neutral**	**Positive**
After labelling (Gemini)	1000	646	320	34
After Preprocessing (Final)	987	645	309	33

B. Text Preprocessing Results

The preprocessing stage aims to reduce noise and standardise text before EDA and modelling. The pipeline used includes: cleaning (removing URLs, symbols, numbers, etc.), case folding, dictionary-based normalisation of non-standard words, tokenisation, stopword removal (NLTK Indonesia), and stemming (Sastrawi). The final output is stored in the *stemming_data* column (for classical models) and ensures that empty data in the preprocessing results column is deleted.

After removing *NaN* data from stemming_data, the amount of data became 987 comments. The final dataset has 9 columns, and the numeric label type is stored as int64.



```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 987 entries, 0 to 986
Data columns (total 9 columns):
 #   Column           Non-Null Count  Dtype
---  ------           --------------  -----
 0   comment          987 non-null    object
 1   cleaning         987 non-null    object
 2   case_folding     987 non-null    object
 3   normalisasi      987 non-null    object
 4   tokenize         987 non-null    object
 5   stopword removal 987 non-null    object
 6   stemming_data    987 non-null    object
 7   sentiment        987 non-null    object
 8   label            987 non-null    int64
dtypes: int64(1), object(8)
memory usage: 69.5+ KB
```

Fig. 2. Information on the structure of the dataset columns after preprocessing.

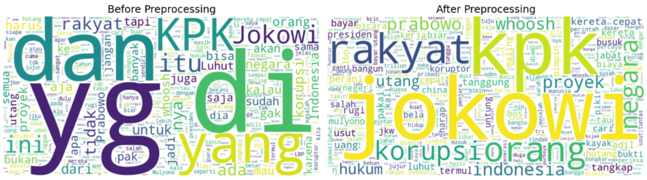


Fig. 3. Comparison of word clouds before and after preprocessing.

C. Exploratory Data Analysis Result (EDA)

EDA on the preprocessed dataset shows good data quality:

- *Dataset size* : (987, 9)
- *Missing values* : 0 (all main columns are complete)
- *Duplication* : 0 (both comment duplication and entire column duplication)

The sentiment distribution in the final dataset is negative (645), neutral (309), and positive (33). This pattern confirms that the dataset remains imbalanced even after cleaning, so model evaluation cannot rely solely on accuracy; metrics such as macro-F1 are needed to reflect performance in the minority class.

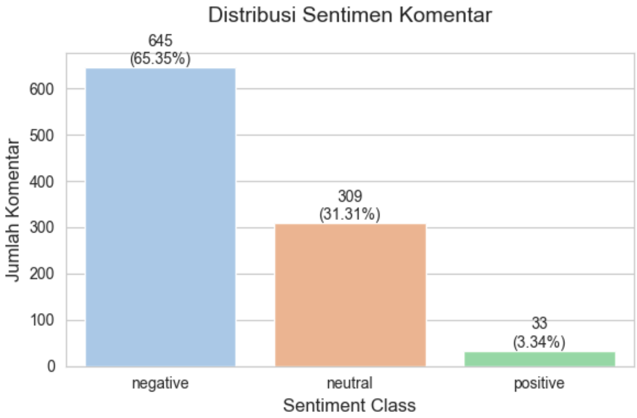


Fig. 4. Distribution of sentiment labels after preprocessing.

D. Baseline Models Result (TF–IDF + Classic Classification)

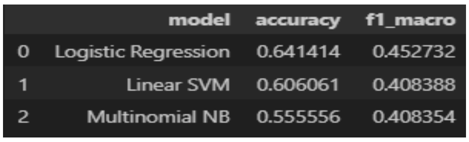
The baseline experiment uses stemming_data with stratified data division: train=789 and test=198. Feature representation uses TF–IDF with the following configuration: ngram_range=(1,2), min_df=3, max_features =10000, and sublinear_tf=True. To reduce the impact of class imbalance on the training data, RandomOverSampler was used so that the train distribution became balanced ({0:516, 1:516, 2:516}).

Three models were tested : *Logistic Regression, Linear SVM, and Multinomial Naive Bayes*. The test set results showed:

- *Logistic Regression* achieved accuracy = 0.6414 and macro-F1 = 0.4527 (the best).
- *Linear SVM* achieved accuracy = 0.6061 and macro-F1 = 0.4084.
- *Multinomial Naive Bayes* achieved accuracy = 0.5556 and macro-F1 = 0.4084.

The best model (Logistic Regression) still shows weaknesses in the positive class (only 7 support tests), with F1-positive=0.10, while the performance of the majority negative class is much better (F1-negative=0.76). This is consistent with the condition of the data, which has very few positive examples.

In terms of interpretability, the top word analysis shows reasonable class signals: strong negative classes in words such as *"bayar", "usut", "tangkap", "kpk", "koruptor"*, while positive classes appear more often in words such as *"manfaat" dan "dukung"*. These findings reinforce that the model captures topic patterns and language tones relevant to the issue.



	model	accuracy	f1_macro
0	Logistic Regression	0.641414	0.452732
1	Linear SVM	0.606061	0.408388
2	Multinomial NB	0.555556	0.408354

Fig. 5. Model Evaluation Results Information

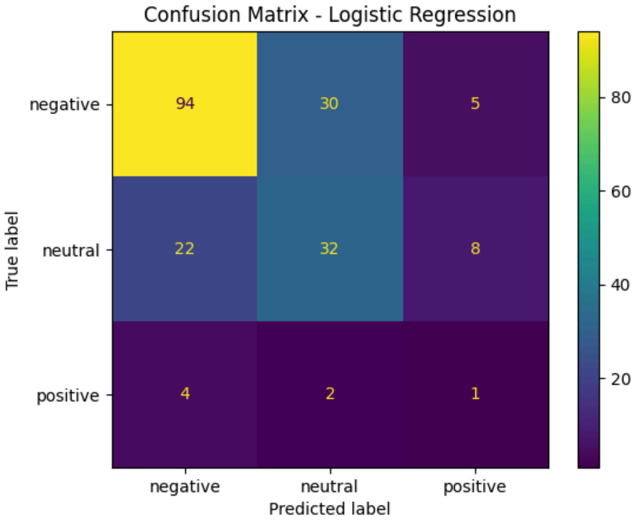
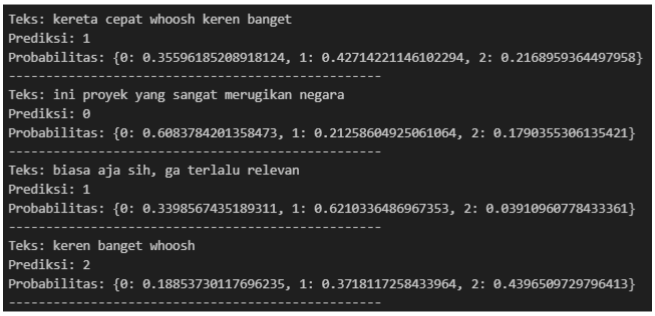


Fig. 6. Confusion matrix of the Logistic Regression model on the test data.



```
Teks: kereta cepat whoosh keren banget
Prediksi: 1
Probabilitas: {0: 0.35596185208918124, 1: 0.42714221146102294, 2: 0.2168959364497958}
-----------------------------------------------
Teks: ini proyek yang sangat merugikan negara
Prediksi: 0
Probabilitas: {0: 0.6083784201358473, 1: 0.21258604925061064, 2: 0.1790355306135421}
-----------------------------------------------
Teks: biasa aja sih, ga terlalu relevan
Prediksi: 1
Probabilitas: {0: 0.3398567435189311, 1: 0.6210336486967353, 2: 0.03910960778433361}
-----------------------------------------------
Teks: keren banget whoosh
Prediksi: 2
Probabilitas: {0: 0.18853730117696235, 1: 0.3718117258433964, 2: 0.4396509729796413}
-----------------------------------------------
```

Fig. 7. Testing results of TF–IDF-based models.

E. IndoBERT Fine-Tuning Result

The transformer approach uses indobenchmark/indobert-base-p2 with train/validation/test data split = (789, 99, 99). Tokenisation is performed with max_length=128, padding=max_length, and truncation=True. The fine-tuning process was run for 4 epochs with a learning rate of 2e-5, batch size of 8, and weight decay of 0.01.

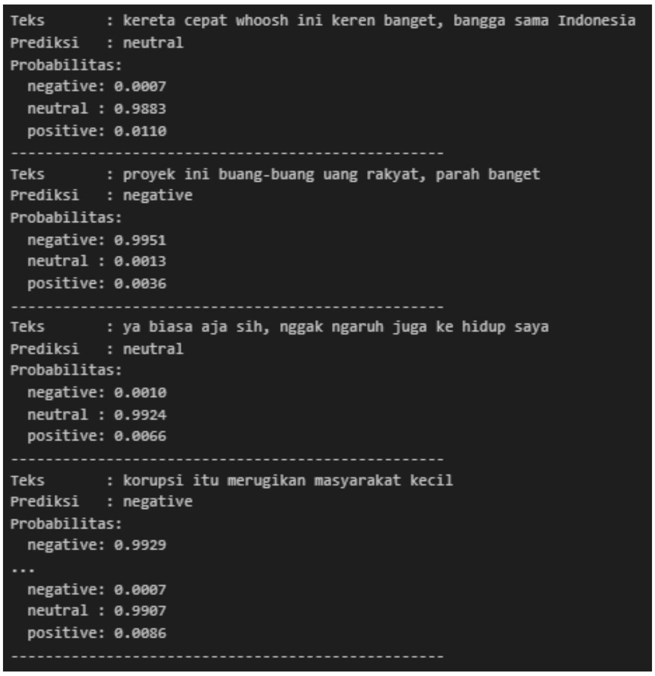
Evaluation on the test set showed:

- Accuracy = 0.6566
- Macro-F1 = 0.4012

By class, IndoBERT performed well on the majority negative class (F1=0.77) and reasonably well on the neutral class (F1=0.43), but failed to predict the positive class (F1=0.00, test support only 3). This explains why IndoBERT's accuracy is relatively high (due to the dominance of negative classes), but its macro-F1 is lower than the best baseline: the model 'collapses' on the minority class.

TABLE II
RESULTS OF THE INDOBERT FINE-TUNING MODEL EVALUATION

Model	**Accuracy**	**Macro-F1**	**F1-Negative**	**F1-Neutral**	**F1-Positive**
IndoBERT fine-tuning	0.6566	0.4012	0.77	0.43	0.00



```
Teks       : kereta cepat whoosh ini keren banget, bangga sama Indonesia
Prediksi   : neutral
Probabilitas:
  negative: 0.0007
  neutral : 0.9883
  positive: 0.0110
-------------------------------------------------
Teks       : proyek ini buang-buang uang rakyat, parah banget
Prediksi   : negative
Probabilitas:
  negative: 0.9951
  neutral : 0.0013
  positive: 0.0036
-------------------------------------------------
Teks       : ya biasa aja sih, nggak ngaruh juga ke hidup saya
Prediksi   : neutral
Probabilitas:
  negative: 0.0010
  neutral : 0.9924
  positive: 0.0066
-------------------------------------------------
Teks       : korupsi itu merugikan masyarakat kecil
Prediksi   : negative
Probabilitas:
  negative: 0.9929
...
  negative: 0.0007
  neutral : 0.9907
  positive: 0.0086
-------------------------------------------------
```

Fig. 8. Testing results of IndoBERT fine-tuning.

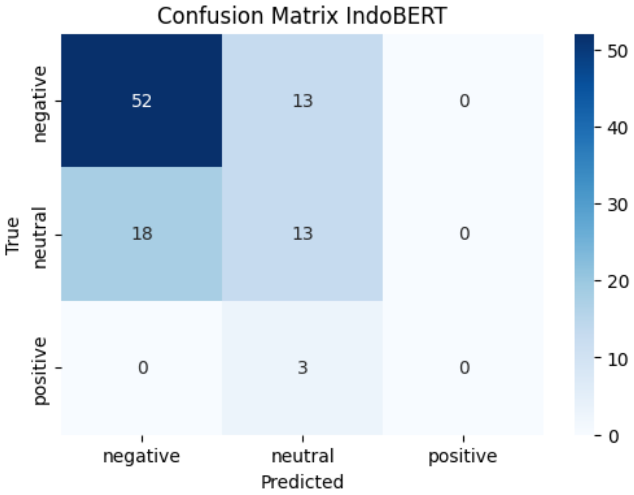


Fig 9. Confusion Matrix of IndoBERT fine-tuning

F. *IndoBERT Qualitative Prediction Analysis*

Manual prediction tests show a tendency for IndoBERT to choose neutral for sentences with a general tone of praise that do not explicitly affirm/deny the issue of corruption. For example, the text 'this whoosh high-speed train is really cool, proud of Indonesia' is predicted to be neutral with a probability of neutral=0.9883 and positive=0.0110. Conversely, accusatory/detrimental texts show a very high probability of being negative (e.g., negative ≈ 0.9951). This pattern is consistent with quantitative findings: the model is very confident in the majority class and tends to avoid the positive class due to weak training signals.

G. *Comparative Discussion and Implications*

The main comparison shows:

1. *Baseline Logistic Regression* provides the highest macro-F1 (0.4527), resulting in a more balanced distribution between classes in the current data conditions.
2. *IndoBERT* is slightly superior in accuracy (0.6566), but inferior in macro-F1 (0.4012) because it is unable to recognise positive classes.
3. Label imbalance (very few positives) is a dominant limitation and causes accuracy-based evaluation to be potentially misleading.

As an implication, for applications that demand fairness in performance across classes (particularly in detecting support/denial that is classified as positive), data improvement or advanced training strategies are required (e.g., adding positive data/active learning, class re-weighting, or loss techniques that are more sensitive to minor classes).

V. CONCLUSION

This study examines sentiment analysis of public opinion on the issue of alleged corruption in the Whoosh High-Speed Rail project using the LLM-Assisted Labeling approach, a classic TF–IDF-based classification model, and the IndoBERT transformer model. Data labelling was performed automatically using the Gemini model, which enabled efficient and contextual large-scale annotation of the issue under study.

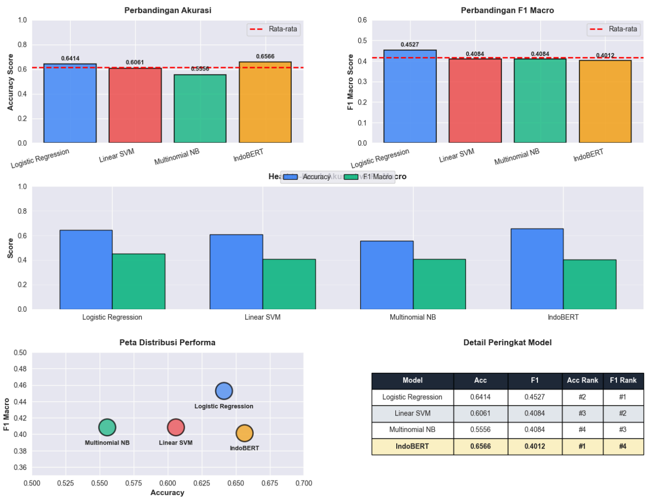
The labelling results show that the sentiment distribution is highly unbalanced, with a dominance of the negative class, followed by neutral, and a very limited number of positive. This imbalance has been proven to have a significant effect on the performance of classification models, particularly in detecting minor classes. The preprocessing and exploratory data analysis stages ensure good data quality, without duplication or empty values, making the dataset suitable for further modelling.

In the baseline experiment using TF–IDF, Logistic Regression produced the best performance with a macro-F1 of 0.4527, indicating a better balance of performance between classes compared to other classical models. Meanwhile, the IndoBERT model produced the highest accuracy (0.6566), but had a lower macro-F1 (0.4012) due to its failure to predict positive classes in the test data. These findings confirm that in highly imbalanced data conditions, the macro-F1 metric is more representative than relying solely on accuracy.

Overall, this study shows that:

1. LLM-Assisted Labeling can be a practical and scalable solution for labelling Indonesian sentiment data
2. Classic TF–IDF-based models are still competitive and even more stable than transformer models on datasets with extreme label distributions, and
3. Data quality and balance play a more crucial role than model complexity.

As future developments, this research can be improved by increasing the amount of data in the minority class through active learning, data augmentation, or adjusting the loss function such as focal loss in transformer models. Additionally, cross-validation-based evaluation and manual validation on data subsets can improve the reliability of labelling and classification results.



Model	Acc	F1	Acc Rank	F1 Rank
Logistic Regression	0.6414	0.4527	#2	#1
Linear SVM	0.6061	0.4084	#3	#2
Multinomial NB	0.5556	0.4084	#4	#3
IndoBERT	**0.6566**	**0.4012**	**#1**	**#4**

Fig. 10. Performance comparison of all evaluated models.

References

[1] I. S. K. Idris dan Y. A. Mustofa, "Analisis Sentimen Terhadap Penggunaan Aplikasi Shopee Mengunakan Algoritma Support Vector Machine (SVM)," *Jambura Journal of Electrical and Electronics Engineering (JJEEE)*, vol. 5, no. 1, pp. 32-35, Jan. 2023.

[2] N. A. Hapsari dan A. D. Indriyanti, "Analisis Sentimen pada Aplikasi Dompet Digital Menggunakan Algoritma Random Forest," *JEISBI: Journal of Emerging Information Systems and Business Intelligence*, vol. 4, no. 3, pp. 186-192, 2023.

[3] F. Koto, A. Rahimi, J. H. Lau, dan 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 (COLING)*, 2020, pp. 757–770.

[4] B. Wilie dkk., "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.

[5] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "*BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*," in *Proc. NAACL-HLT*, 2019, pp. 4171–4186.