Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 10, Number 2, April 2026
DOI : https://doi.org/10.33395/sinkron.v10i2.15975
e-ISSN : 2541-2019
p-ISSN : 2541-044X

Improving Multi-Class Public Complaint Classification with LSTM, Word2Vec, and Random Oversampling

Azza Nimasari[1)*], Galuh Wilujeng Saraswati[2)], Erba Lutfina[3)]
[1,2,3)]Information System, Faculty of Computer Science, Universitas Dian Nuswantoro, Kediri, Indonesia
[1)]612202400157@mhs.dinus.ac.id, [2)]galuhwilujengs@dsn.dinus.ac.id, [3)]erba.lutfina@dns.dinus.ac.id

Submitted :Mar 6, 2026 | **Accepted** : April 1, 2026 | **Published** : April 2, 2026

Abstract: Digital transformation in the public sector encourages local governments to enhance service quality through online complaint management systems. However, the high volume of incoming complaints and significant data imbalance across 31 Organisasi Perangkat Daerah (OPD) pose challenges for efficient manual classification, often resulting in delays and misclassification. This study proposes an automated text classification model that integrates Long Short-Term Memory (LSTM), Word2Vec, and Random Oversampling (ROS), optimized using the Adam algorithm. The novelty of this research lies in the integration of sequential modeling and imbalance handling to address an extreme multi-class classification problem involving 31 OPD categories within a highly imbalanced dataset. The research stages include text preprocessing, word embedding construction using Word2Vec, data balancing through ROS, and model training using LSTM. Experimental results show that the proposed model achieves an accuracy of 0.72, with macro-average precision, recall, and F1-score of 0.67, 0.67, and 0.66, respectively. Considering the complexity of classifying 31 classes and the presence of severe data imbalance, the macro F1-score of 0.66 indicates that the model is reasonably effective in capturing classification patterns, although performance is not yet evenly distributed across all classes. Overall, the combination of LSTM, Word2Vec, and ROS demonstrates potential as a baseline approach for automating public complaint classification in complex multi-class scenarios. The proposed model can improve the accuracy and speed of complaint distribution to the appropriate OPD, thereby enhancing the efficiency and responsiveness of public service delivery compared to conventional manual methods.
Keywords: LSTM; Public Complaint; Random Oversampling; Text Classification; Word2Vec

INTRODUCTION

The digital transformation sweeping the public sector demands service delivery that is increasingly transparent, efficient, accurate, and accessible. This condition aligns with the rising number of internet users in Indonesia, which according to the Association of Indonesian Internet Service Providers (APJII) in 2024, has reached over 221 million people with a penetration rate of 79.5% (Agus Tri Haryanto, 2024). This increase in digital access and connectivity encourages the government to innovate in online public complaint management as part of efforts to improve public service quality and build public trust.

According to Undang-Undang Nomor 25 Tahun 2009 tentang Pelayanan Publik, the provision of public services must effectively meet the needs of citizens through various media, both electronic and conventional (Undang-Undang (UU) Nomor 25 Tahun 2009 Tentang Pelayanan Publik, 2009). Local governments now utilize various official channels such as email, media-social, and internet-based applications to convey public aspirations and complaints. However, in practice, obstacles remain, such as the high volume of complaints and lengthy validation processes, causing delays in information delivery and follow-up (Alkaff et al., 2021). Furthermore, constraints in time and human resources increase the risk of errors in the classification and distribution of complaints to the authorized OPD (Umam et al., 2025).

In a single day, administrators may receive approximately 70 complaints for classification, a number that can surge to 100–150 under specific conditions such as LPG shortages while the validation process is handled by only two administrators. Moreover, the distribution of complaints is inherently imbalanced, as certain OPD receive

*name of corresponding author

e-ISSN : 2541-2019
p-ISSN : 2541-044X

significantly more reports than others. This imbalance introduces additional difficulty in ensuring fair and accurate classification, particularly in multi-class scenarios involving many categories. Therefore, an automated text classification system capable of processing large volumes of unstructured data efficiently and accurately is critically needed. Text mining techniques have been shown to support such processes by enabling automatic extraction of meaningful information from textual data (Asrawi et al., 2023).

Previous studies have explored various approaches to complaint classification. Traditional methods, such as LDA combined with SVM, have achieved moderate performance 78% accuracy, but they rely on shallow feature representations that often fail to capture complex semantic relationships in text (Alkaff et al., 2021). More recent approaches utilize Deep Learning models, including Convolutional Neural Network (CNN), Recurrent Neural Network (RNN), and Long Short-Term Memory (LSTM), which are capable of learning contextual representations automatically. Among these, LSTM demonstrates superior performance due to its ability to model long-term dependencies in sequential data and mitigate the vanishing gradient problem (Perumal et al., 2024) However, existing studies exhibit several limitations. First, many works focus on binary or low-class classification problems, rather than multi-class routing tasks involving numerous target categories, such as OPD distribution. Second, while LSTM has shown strong performance, its application in high-class, real-world e-government datasets remains limited. Third, although data imbalance handling techniques such as Random Oversampling (ROS) and SMOTE have been widely studied, they are predominantly evaluated in simpler classification settings and rarely integrated with deep learning models in complex multi-class scenarios. Furthermore, prior studies often emphasize overall accuracy, with insufficient attention to macro-level evaluation metrics, which are more appropriate for imbalanced datasets.

This study aims to address these gaps by proposing a text classification model that integrates Word2Vec for semantic feature representation, LSTM for sequential learning, and ROS for handling class imbalance, optimized using the ADAM algorithm. The model is evaluated on real-world public complaint data involving 31 OPD, representing a complex multi-class classification scenario.

The research provides an empirical evaluation of the combined use of Word2Vec, LSTM, and ROS in handling imbalanced, high-dimensional multi-class text classification problems. It also emphasizes the use of appropriate evaluation metrics to better reflect model performance across all classes. From a practical perspective, the proposed model offers a solution for automating the initial classification and routing of public complaints, thereby reducing administrative workload, minimizing human error, and accelerating response time in government service systems.

LITERATURE REVIEW

Text classification is a fundamental task in Natural Language Processing (NLP) that aims to automatically map text documents into specific categories. In the context of public complaints, this task has significant practical implications because the daily volume of incoming reports far exceeds the manual processing capacity of operators. This approach aims to extract useful information from a collection of documents, where the data used is typically unstructured or semi-structured text, involving specific tasks such as text clustering (Purba & Yadi, 2023).

Several studies have explored automated text-based public complaint classification. Alkaff et al. (2021) tested a combination of Latent Dirichlet Allocation (LDA) as a topic feature extraction method with Support Vector Machine (SVM) using report data from a complaint system covering four agencies. This model achieved an accuracy of 79.85% and an F1-score of 74.67% using a 70:30 split (Alkaff et al., 2021). Nevertheless, the LDA-SVM approach has fundamental limitations, LDA represents documents as latent topic distributions based on a bag-of-words model, making it unable to capture word order or contextual semantic relationships between words in a sentence. This becomes a serious issue when target classes have high contextual similarity, as is often the case with complaint reports directed to Organisasi Perangkat Daerah (OPD) with overlapping duties.

In news text classification, Widhiyasana et al. (2021) compared CNN, LSTM, and their combination (C-LSTM) for classifying Indonesian news texts into three categories. C-LSTM achieved an F1-score of 93.27%, surpassing CNN 89.85% and LSTM 90.87% (Widhiyasana et al., 2021). These findings indicate that an architecture combining the local feature extraction capabilities of CNN with the sequence modeling capabilities of LSTM can provide better results. However, that study only tested three classes with relatively balanced data distributions and did not explore multi-class scenarios with high distribution imbalances.

Long Short-Term Memory (LSTM) architecture is specifically designed to overcome the vanishing gradient problem that often hinders the performance of standard Recurrent Neural Networks (RNN) when processing long data sequences. The primary mechanism enabling this is the use of a cell state regulated by three main gates: the forget gate, input gate, and output gate (Perumal et al., 2024). The forget gate determines which past information is no longer relevant and should be removed from memory, while the input gate identifies and adds important new information to the cell state. Finally, the output gate determines which part of the memory will be passed as the hidden state to the next step. Through the integration of these three gates, LSTM is able to maintain information

*name of corresponding author

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 10, Number 2, April 2026
DOI : https://doi.org/10.33395/sinkron.v10i2.15975
e-ISSN : 2541-2019
p-ISSN : 2541-044X

gradients over a longer period, making it highly effective at capturing contextual dependencies between words in complex public complaint texts.

Perumal et al. (2024), in their research using the same dataset, showed that LSTM performed better than standard RNN and CNN, reaching an accuracy of 93.29% (Perumal et al., 2024). This superiority is attributed to the LSTM architecture's design, which is explicitly intended to capture long-term dependencies in sequential data—the main weakness of standard RNNs. These findings strengthen the argument for using LSTM as the classification backbone in this study, especially considering that public complaint reports tend to have significant sentence length variations and high levels of semantic ambiguity.

Research by Kusuma et al. (2023) showed that the use of SMOTE and ADASYN consistently improved the performance of SVM and Random Forest, yet actually decreased the accuracy of Naive Bayes (I Gusti Ngurah Ady Kusuma et al., 2023). This finding confirms that the interaction between balancing techniques and classification algorithms is specific and cannot always be assumed to have a positive impact. Alex et al. (2022) explained that data balancing techniques significantly increase the F1-score and sensitivity of deep learning models in recognizing patterns in imbalanced datasets (Alex et al., 2022). Wongvorachan et al. (2023) also confirmed that oversampling techniques provide more stable and optimal performance compared to models without imbalance handling (Wongvorachan et al., 2023). In this study, Random Oversampling (ROS) was chosen for its ability to strengthen class representation without losing information from the majority class, as well as its compatibility with the deep learning architecture used.

From the discussion above, three gaps have been identified that haven't been addressed simultaneously by previous studies. First, no research has tested public complaint classification on a scale of more than five OPD classes, whereas real conditions in many local governments involve dozens of OPD with overlapping duties. Second, studies using deep learning for Indonesian text classification generally work on relatively balanced data and a small number of classes, thus failing to provide a performance overview under conditions of extreme imbalance with many multi-classes. Third, the combination of LSTM with data balancing techniques for multi-OPD public complaint classification has not been explored, even though each component has proven effective in different domains and scales.

Therefore, this study proposes an LSTM-based classification approach combined with Word2Vec as a contextual semantic representation and Random Oversampling as an imbalance handling strategy, targeting 31 OPDs as the destination classes. This combination is designed to address these three gaps: LSTM's ability to capture long-term contextual dependencies is expected to overcome semantic ambiguity between closely related OPD classes, while ROS ensures the model remains unbiased toward the dominant OPD classes in the dataset.

Table 1. Summary of Related Works

Author(s) & Year	Method	Dataset	Class	Result	Limitations
Alkaff et al. (2021)	LDA + SVM	Public Complaints	4 Classess	Accuracy 79.85%,	Limited classes, data imbalance ignored
I Gusti Ngurah Ady Kusuma et al. (2023)	NBC, SVM, TF + SMOTE/ADASYN	Public Complaints	4 Classes	Accuracy 82.32% (RF + sampling)	Traditional methods; limited multi-class scenario
Widhiyasana et al. (2021)	CNN, LSTM, C-LSTM	Indonesian News	3 Classses	F1-score 93.27% (C-LSTM)	Balanced dataset; not in complaint domain

METHOD

All stages of this research are conducted within the Google Colab cloud-based computing environment using the Python programming language. Data processing and analysis are performed using pandas and NumPy, while text preprocessing utilizes Sastrawi and NLTK. Word representations are constructed through Gensim using Word2Vec. The data splitting process and class imbalance handling are executed with Scikit-learn and Imbalanced-learn, specifically employing the Random Oversampling technique. Furthermore, the development and training of the deep learning model are carried out using TensorFlow and Keras, which provide essential components such as Embedding, LSTM, Dense, and Dropout to build the text classification architecture. This research begins with a request for public complaint data from the relevant institution. The data have been classified based on the OPD responsible for handling each complaint. Figure 1 illustrates the implementation workflow of the model developed in this study.

*name of corresponding author

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 10, Number 2, April 2026
DOI : https://doi.org/10.33395/sinkron.v10i2.15975
e-ISSN : 2541-2019
p-ISSN : 2541-044X

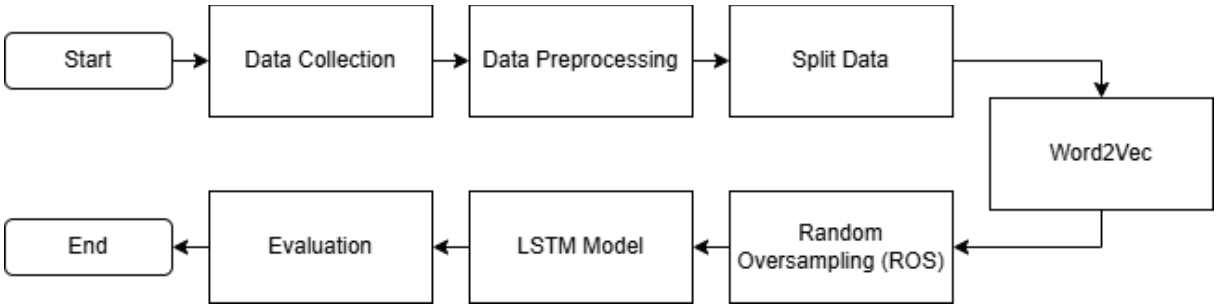

Fig. 1 Research Stages

Data Collection

The dataset used in this study was obtained through an official data request to the relevant institution. The data were provided in CSV format, containing a collection of public complaints exported from multiple complaint channels. The dataset consists of two features: message and OPD, where OPD serves as the target variable representing the department responsible for handling each complaint. In total, 3,567 complaint records were collected. However, the distribution of data across OPD classes is imbalanced.

Data Preprocessing

The preprocessing stage is an important step to transform text into a format suitable for processing by the model (Idris et al., 2025). This process consists of several stages, including data cleaning, case folding, tokenization, stopword removal, stemming, and label encoding.

Data cleaning is performed to improve the accuracy of the analysis results. This process aims to remove irrelevant characters, duplicate data, noise, and other unnecessary information from the dataset (Utami et al., 2023). A total of 319 duplicate records were identified and removed, reducing the dataset to 3,248 records. In this stage, noise such as numbers, punctuation marks, URLs, and unnecessary symbols was also eliminated.

The next step is case folding, which converts all text into lowercase format. After that, the tokenization process splits sentences into individual word units called tokens. The resulting tokens are then compared with a stopword list to remove words that do not carry significant meaning in the text analysis process (Khairani et al., 2024).

The subsequent step is stemming, which transforms each word that has passed the stopword removal process into its base form by removing all affixes. This process is performed using the Sastrawi library, which is specifically designed to facilitate Indonesian language text processing and to prepare textual data for further analytical stages (Cahyani & Saraswati, 2023).

Split Data

The data that had undergone the preprocessing stage were subsequently divided into training data and testing data. The training data were used to train the model so that it could learn patterns from the available dataset, while the testing data were used to evaluate the model's performance on previously unseen data (Maulana et al., 2023). In this study, the dataset was divided using a ratio of 70% for training data, 10% for validation data and 20% for testing data.

Word2Vec

Word2Vec is a word embedding technique used to represent words in public complaint data as vectors with a specific embedding size, enabling similarity and relational operations between words. The result of word embedding is a vector representation in which words with similar meanings have similar numerical values (Amalia et al., 2022).

In this study, Word2Vec was built using X_train as the input data. The model was trained with the parameters vector_size = 100, window = 5, min_count = 2, and the Skip-gram architecture (sg = 1). After the training process was completed, a word_index dictionary was created to map each word in the vocabulary to a numerical index. This mapping is later used to convert text into sequence form for the modeling stage.

Random Oversampling (ROS)

Random Oversampling (ROS) is a method used to balance data in machine learning. Handling imbalanced data is necessary to ensure that the applied method can produce high classification accuracy. ROS works by randomly duplicating samples from the minority class. The ROS method has shown strong performance under imbalanced data conditions because it can improve the accuracy and recall of classification models without sacrificing the majority class (Wongvorachan et al., 2023).

The dataset was partitioned into training, validation, and testing sets with a 70:10:20 ratio using stratified

*name of corresponding author

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 10, Number 2, April 2026
DOI : https://doi.org/10.33395/sinkron.v10i2.15975
e-ISSN : 2541-2019
p-ISSN : 2541-044X

sampling to maintain the class proportions across all subsets. This process resulted in 2,273 training samples, 325 validation samples, and 650 testing samples. This approach ensures that minority classes remain represented in data test, allowing the evaluation to be derived from original, unaltered data. However, the initial distribution of data train exhibited a significant imbalance, with a majority to minority class ratio of 24:1, which poses a risk of the model being biased toward classes with larger data volumes.

To address this issue, Random Oversampling (ROS) was applied exclusively to data train after the partitioning process to prevent data leakage, a condition where duplicate samples enter the test set, rendering the evaluation invalid. ROS balanced the distribution by duplicating samples from the minority classes until each class reached 288 samples. Consequently, the total data train increased from 2,273 to 8,928, while the test set remained constant at 650 samples. With this balanced distribution, the model can learn more equitably across all classes without compromising the validity of the evaluation.

Long Short Term Memory (LSTM)

Long Short-Term Memory (LSTM) is an extension of the Recurrent Neural Network (RNN) designed to overcome the limitations of traditional RNNs in learning long-term dependencies in sequential data. LSTM can retain important information within long data sequences through its internal memory mechanism, making it more effective than standard RNN models.

In practice, the LSTM architecture can be developed with various layer configurations such as single-layer, double-layer, or triple-layer architectures to achieve optimal classification performance (Fajar Abdillah & Kusnawi, 2023).

The LSTM architecture consists of three main components: input gate, forget gate, and output gate, which regulate the flow of information within the network.The input gate determines which new information will be stored in the memory cell.

$$i_t = \sigma(W_i \cdot [h_{t-1}, Xt] + b_i) \quad (1)$$

i_t = input gate
σ = sigmoid activation function
W_i = weight of the input gate
h_{t-1} = previous hidden state
Xt = input value
b_f = bias of the input gate

The candidate gate generates candidate values that will potentially update the cell state.

$$C_t = tanh(W_c \cdot [h_{t-1}, Xt] + b_c) \quad (2)$$

C_t = candidate gate
W_c = weight of the candidate gate
b_c = bias of the candidate gate
tanh = hyperbolic tangent activation function

The Cell State (c_t) updates the previous memory cell with new information obtained from the gates.

$$c_t = (i_t * C_t + f_t * c_{t-1}) \quad (3)$$

c_t = cell gate
i_t = input gate
C_t = candidate gate
f_t = forget gate
c_{t-1} = previous cell state

The forget gate determines which information should be retained or discarded from the memory cell.

$$f_t = \sigma(W_f \cdot [h_{t-1}, Xt] + b_f) \quad (4)$$

f_t = forget gate
W_f = bobot forget gate
b_f = bias forget gate

The output gate determines which part of the memory cell will be produced as output.

$$o_t = \sigma(W_o \cdot [h_{t-1}, Xt] + b_o) \quad (5)$$

*name of corresponding author

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 10, Number 2, April 2026
DOI : https://doi.org/10.33395/sinkron.v10i2.15975
e-ISSN : 2541-2019
p-ISSN : 2541-044X

$$h_t = o_t * tanh\,(c_t) \quad (6)$$

o_t = output gate
W_o = weight of the output gate
b_o = bias of the output gate
h_t = hidden state
c_t = cell gate

The classification model is built using a single-layer LSTM architecture designed to capture sequential representations of complaint text data. The first layer is an embedding layer with a dimension of 100, initialized using pre-trained Word2Vec Skip-gram weights. The trainable parameter is enabled to allow the embedding representation to be fine-tuned during training, enabling the model to capture semantic context more relevant to the complaint domain. Additionally, the mask_zero=True parameter is used to ensure that padding tokens do not affect computations in subsequent layers. The second layer is an LSTM with 64 units, equipped with a dropout rate of 0.4 on input-to-hidden connections and a recurrent dropout of 0.4 on hidden-to-hidden connections to reduce the risk of overfitting. The output layer uses a fully connected (Dense) layer with the number of units corresponding to the number of OPD classes and a softmax activation function to produce a probability distribution over all classes.

The model is compiled using the Adam optimizer. The loss function used is sparse categorical crossentropy, as class labels are represented as integer values. Training is performed with a batch size of 32 for up to 30 epochs. To prevent overfitting and improve training efficiency, an early stopping mechanism is applied by monitoring the validation loss with a patience of 5 epochs. Additionally, ModelCheckpoint is used to save the best model weights during training. All experiments are conducted with a random seed set to 42 to ensure reproducibility of results.

Model Evaluation

A confusion matrix is an evaluation method used to assess the performance of a classification model by comparing the predicted results with the actual classes. This method consists of four main components: True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN), which represent correct and incorrect predictions for each class. From these components, several evaluation metrics such as accuracy, precision, and recall can be calculated to analyze model performance in greater detail (Yudhistira et al., 2025).

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \quad (7)$$

$$\text{Precision} = \frac{TP}{TP + FP} \quad (8)$$

$$\text{Recall} = \frac{TP}{TP + FN} \quad (9)$$

RESULT

This section presents the results of the study gradually according to the research workflow discussed previously. The model was developed using the Python programming language with the support of relevant libraries.

Preprocessing Results

The total number of public complaint records obtained through an official request was 3,567 data entries with 31 OPD labels. The dataset contained 319 duplicate records, which were removed during preprocessing, resulting in a final dataset of 3,248 records. Table 2 presents a sample of the data used in this study.

Table 2. Dataset

Messages	OPD
dalam upaya mitigasi bencana mohon bantuannya untuk pembenahan jalan di desa bugo sampai petok ada jalan berlobang. karena sering terjadi kecelakaan. tks	Dinas Pekerjaan Umum dan Penataan Ruang
lampu Trafic light Perempatan Djimboen Dalam keadaan Mati semenjak 3hr yg lalu, hal ini menimbulkan kemacetan, Mohon Bantuan untuk pembenahan nya, untuk menghindari terjadinya Kecelakaan. tks	Dinas Perhubungan

After the data cleaning process, several stages are performed: case folding, tokenization, stopwords removal, and stemming. Table 3 presents the comparison of the data before and after preprocessing. Subsequently, label encoding is applied to the target variable, as shown in Table 4.

*name of corresponding author

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 10, Number 2, April 2026
DOI : https://doi.org/10.33395/sinkron.v10i2.15975
e-ISSN : 2541-2019
p-ISSN : 2541-044X

Table 3. Preprocessing Result

Before	After
dalam upaya mitigasi bencana mohon bantuannya untuk pembenahan jalan di desa bugo sampai petok ada jalan berlobang. karena sering terjadi kecelakaan. tks	upaya mitigasi bencana bantu benah jalan desa bugo tok jalan lubang celaka
lampu Trafic light Perempatan Djimboen Dalam keadaan Mati semenjak 3hr yg lalu, hal ini menimbulkan kemacetan, Mohon Bantuan untuk pembenahan nya, untuk menghindari terjadinya Kecelakaan. tks	lampu trafic light empat djimboen mati semenjak honor timbul macet bantu benah hindar celaka

Table 4. Label Data

OPD	Label
Badan Kepegawaian Daerah	0
Badan Kesatuan Bangsa dan Politik	1
Badan Penanggulangan Bencana Daerah	2
…	…

Modeling Results

The application of ROS to the training data has a noticeable impact on model performance, particularly in addressing class imbalance. Figures 2 and 3 illustrate the class distribution before and after the application of ROS. Table 5 presents a comparison of the performance between the LSTM + Word2Vec model without and with ROS. In general, the use of ROS improves performance across nearly all evaluation metrics. Accuracy increases from 0.69 to 0.72, while macro recall shows a significant improvement from 0.57 to 0.67. The macro F1-score also rises from 0.58 to 0.66. On the other hand, macro precision remains stable at 0.67 for both models. This indicates that the performance improvement after applying ROS is primarily driven by the model's enhanced ability to correctly identify more classes (recall), rather than by increased prediction precision. In other words, the model becomes more inclusive in classifying data, particularly for minority classes that were previously often overlooked.

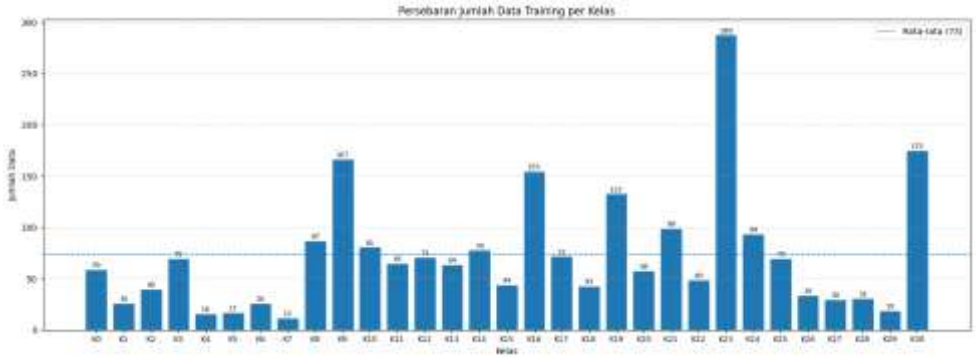

Fig. 2 Data Train without ROS

*name of corresponding author

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 10, Number 2, April 2026
DOI : https://doi.org/10.33395/sinkron.v10i2.15975
e-ISSN : 2541-2019
p-ISSN : 2541-044X

Fig. 3 Data Train with ROS

Table 5. Model Classification Result

Method	Macro			Accuracy
	Precision	Recall	F1-Score	
LSTM + Word2Vec	0.67	0.57	0.58	0.69
LSTM + Word2Vec +ROS	0.67	0.67	0.66	0.72

A more detailed per-class analysis reveals considerable variation in performance. Several classes, such as the Dinas Kependudukan dan Pencatatan Sipil, Dinas Ketahanan Pangan dan Peternakan, and Perusahaan Daerah Air Minum (PDAM), demonstrate strong performance with F1-scores above 0.85. This is influenced by a combination of sufficient data availability and more distinctive textual characteristics. In contrast, classes such as the Dinas Komunikasi dan Informatika and Dinas Pariwisata exhibit lower performance, indicating challenges in distinguishing contextual differences between classes as well as limitations in data availability.

Table 6. Classification Report

Label	OPD	Precision	Recall	F1-Score	Support
0	Badan Kepegawaian Daerah	0.57	0.47	0.52	17
1	Badan Kesatuan Bangsa dan Politik	0.67	0.57	0.62	7
2	Badan Penanggulangan Bencana Daerah	0.78	0.64	0.7	11
3	Badan Pendapatan Daerah	0.74	0.85	0.79	20
4	Badan Pengelolaan Keuangan dan Aset Daerah	0.5	0.6	0.55	5
5	Badan Perencanaan Pembangunan Daerah	0.5	0.4	0.44	5
6	Bagian Kesejahteraan	0.6	0.43	0.5	7
7	Bagian Organisasi Sekretariat Daerah	0.67	0.67	0.67	3
8	Bagian Perekonomian	0.88	0.6	0.71	25
9	Dinas Kependudukan dan Pencatatan Sipil	0.85	0.96	0.9	48
10	Dinas Kesehatan	0.8	0.67	0.73	24
11	Dinas Ketahanan Pangan dan Peternakan	0.78	1	0.88	18
12	Dinas Komunikasi dan Informatika	0.33	0.2	0.25	20
13	Dinas Koperasi dan Usaha Mikro	0.7	0.89	0.78	18
14	Dinas Lingkungan Hidup	0.67	0.73	0.7	22
15	Dinas Pariwisata	0.5	0.31	0.38	13
16	Dinas Pekerjaan Umum dan Penataan Ruang	0.76	0.91	0.83	45
17	Dinas Pemberdayaan Masyarakat dan Pemerintahan Desa	0.58	0.75	0.65	20
18	Dinas Penanaman Modal dan PTSP	0.53	0.75	0.62	12
19	Dinas Pendidikan	0.68	0.68	0.68	38
20	Dinas Perdagangan	0.63	0.75	0.69	16

*name of corresponding author

e-ISSN : 2541-2019
p-ISSN : 2541-044X

21	Dinas Perhubungan	0.67	0.55	0.6	29
22	Dinas Pertanian dan Perkebunan	0.86	0.86	0.86	14
23	Dinas Perumahan dan Kawasan Permukiman	0.87	0.7	0.77	83
24	Dinas Sosial	0.73	0.7	0.72	27
25	Dinas Tenaga Kerja	0.68	0.75	0.71	20
26	Inspektorat	0.43	0.6	0.5	10
27	Perusahaan Daerah Air Minum (PDAM)	0.88	0.88	0.88	8
28	RSKK Pare	0.71	0.56	0.62	9
29	RSUD Simpang Lima Gumul	0.43	0.5	0.46	6
30	Satuan Polisi Pamong Praja	0.77	0.86	0.81	50
Macro average		0.67	0.67	0.66	650
Weighted average		0.72	0.72	0.71	650

DISCUSSIONS

The implementation of ROS is able to improve the overall model performance, particularly in its ability to recognize minority classes. The increase in macro recall from 0.57 to 0.67 confirms that the main limitation of the model without ROS lies in its low sensitivity toward classes with limited data. In the context of multi-class classification involving 31 classes, achieving an accuracy of 0.72 can be considered reasonably good, given the complexity of the task, which involves numerous categories with potentially high semantic similarity.

High performance in certain classes is influenced not only by the amount of data (support) but also by the specificity of the vocabulary used in the complaints. Classes with more specific tasks tend to be easier for the model to learn due to clearer semantic boundaries. In contrast, lower performance in some classes is not only caused by limited data but also by significant semantic overlap between classes. Although ROS improves performance, it has several limitations. ROS only duplicates samples from minority classes without introducing new variations, meaning the model still learns from limited patterns. This may lead to overfitting and does not fully resolve the imbalance problem. Furthermore, when the original data in minority classes is extremely limited, duplication alone is insufficient to build strong representations. Compared to other methods such as SMOTE or class weighting, ROS is relatively simple but less capable of capturing more complex data distributions. For instance, SMOTE generates more diverse synthetic samples, while class weighting allows the model to give greater attention to minority classes without duplicating data. In addition, limitations of the LSTM model also play an important role. LSTM combined with Word2Vec produces static representations, which are less effective in capturing complex contextual information. Transformer-based models such as IndoBERT offer advantages in understanding deeper context and have the potential to achieve better performance, especially in cases with high semantic overlap. Compared to previous studies and other deep learning approaches, these results are consistent in demonstrating that handling data imbalance is a key factor in improving text classification performance.

CONCLUSION

This study demonstrates that the application of Random Oversampling (ROS) in the LSTM + Word2Vec model is able to improve classification performance, particularly in handling data imbalance. This is reflected in the increase of the macro F1-score from 0.58 to 0.66 and the macro recall from 0.57 to 0.67, indicating that the model becomes more capable of recognizing minority classes.

However, the model's performance is still not evenly distributed across all classes, especially for those with very limited data and high semantic similarity. Therefore, the obtained results are not yet fully optimal, although they can still be considered reasonably good given the complexity of the classification task involving 31 classes.

Overall, the combination of LSTM, Word2Vec and ROS approach can be regarded as a promising baseline for imbalanced multi-class public complaint classification systems. Future research is recommended to combine more advanced imbalance handling techniques with contextual embedding-based models such as IndoBERT to achieve more comprehensive performance improvements across all classes.

REFERENCES

Agus Tri Haryanto. (2024, January 31). *APJII: Jumlah Pengguna Internet Indonesia Tembus 221 Juta Orang*. Https://Inet.Detik.Com/Cyberlife/d-7169749/Apjii-Jumlah-Pengguna-Internet-Indonesia-Tembus-221-Juta-Orang.

*name of corresponding author

Alex, S. A., Jhanjhi, N. Z., Humayun, M., Ibrahim, A. O., & Abulfaraj, A. W. (2022). Deep LSTM Model for Diabetes Prediction with Class Balancing by SMOTE. *Electronics (Switzerland)*, *11*(17). https://doi.org/10.3390/electronics11172737

Alkaff, M., Baskara, A. R., & Maulani, I. (2021). Klasifikasi Laporan Keluhan Pelayanan Publik Berdasarkan Instansi Menggunakan Metode LDA-SVM. *Jurnal Teknologi Informasi Dan Ilmu Komputer*, *8*(6), 1265–1276. https://doi.org/10.25126/jtiik.2021863768

Amalia, J., Pakpahan, J., Pakpahan, M., Panjaitan, Y., Informatika dan Teknik Elektro, F., & Teknologi Del, I. (2022). Model Klasifikasi Berita Palsu Menggunakan Bidirectional LSTM Dan Word2Vec Sebagai Vektorisasi. *Jurnal Teknik Informatika Dan Sistem Informasi*, *9*(4). https://doi.org/https://doi.org/10.35957/jatisi.v9i4.1332

Asrawi, H., Utami, E., & Yaqin, A. (2023). LSTM and Bidirectional GRU Comparison for Text Classification. *Sinkron*, *8*(4), 2264–2274. https://doi.org/10.33395/sinkron.v8i4.12899

Cahyani, S. N., & Saraswati, G. W. (2023). IMPLEMENTATION OF SUPPORT VECTOR MACHINE METHOD IN CLASSIFYING SCHOOL LIBRARY BOOKS WITH COMBINATION OF TF-IDF AND WORD2VEC. *Jurnal Teknik Informatika (Jutif)*, *4*(6), 1555–1566. https://doi.org/10.52436/1.jutif.2023.4.6.1536

Fajar Abdillah, M., & Kusnawi, K. (2023). Comparative Analysis of Long Short-Term Memory Architecture for Text Classification. *ILKOM Jurnal Ilmiah*, *15*(3), 455–464. https://doi.org/10.33096/ilkom.v15i3.1906.455-464

I Gusti Ngurah Ady Kusuma, I Made Pradipta, I Made Ari Santosa, & I Komang Dharmendra. (2023). PENANGANAN KETIDAKSEIMBANGAN DATA PADA KLASIFIKASI PENGADUAN MASYARAKAT. *Jurnal Teknologi Informasi Dan Komputer*, *9*(5). https://doi.org/10.36002/jutik.v9i5.2643

Idris, M., Rifai, A., & Tania, K. D. (2025). Sentiment Analysis of Tokopedia App Reviews using Machine Learning and Word Embeddings. *Sinkron*, *9*(1), 210–219. https://doi.org/10.33395/sinkron.v9i1.14278

Khairani, U., Mutiawani, V., & Ahmadian, H. (2024). Pengaruh Tahapan Preprocessing Terhadap Model Indobert Dan Indobertweet Untuk Mendeteksi Emosi Pada Komentar Akun Berita Instagram. *Jurnal Teknologi Informasi Dan Ilmu Komputer*, *11*(4), 887–894. https://doi.org/10.25126/jtiik.1148315

Maulana, A. R., Wijoyo, S. H., & Mursityo, Y. T. (2023). Analisis Sentimen Kebijakan Penerapan Kurikulum Merdeka Sekolah Dasar dan Sekolah Menengah pada Media Sosial Twitter dengan Menggunakan Metode Word Embedding dan Long Short Term Memory Networks (LSTM). *Jurnal Teknologi Informasi Dan Ilmu Komputer*, *10*(3), 523–530. https://doi.org/10.25126/jtiik.2023106977

Perumal, T., Mustapha, N., Mohamed, R., & Shiri, F. M. (2024). A Comprehensive Overview and Comparative Analysis on Deep Learning Models. *Journal on Artificial Intelligence*, *6*(1), 301–360. https://doi.org/10.32604/jai.2024.054314

Purba, M., & Yadi, Y. (2023). Implementation Opinion Mining For Extraction Of Opinion Learning In University. *SinkrOn*, *8*(2), 694–699. https://doi.org/10.33395/sinkron.v8i2.11994

Umam, A. K., Alzami, F., Sani, R. R., Rohmani, A., Prabowo, D. P., Pergiwati, D., Megantara, R. A., & Iswahyudi, I. (2025). Enhancing Entity Extraction in E-Government Complaint Data using LDA-Assisted NER. *Sinkron*, *9*(4), 1878–1888. https://doi.org/10.33395/sinkron.v9i4.15292

Undang-Undang (UU) Nomor 25 Tahun 2009 Tentang Pelayanan Publik, Pub. L. 25, Lembaran Negara Republik Indonesia (2009).

Utami, S., Lhaksmana, K. M., & Sibaroni, Y. (2023). Deep Learning and Imbalance Handling on Movie Review Sentiment Analysis. *SinkrOn*, *8*(3), 1894–1907. https://doi.org/10.33395/sinkron.v8i3.12770

Widhiyasana, Y., Semiawan, T., Gibran, I., Mudzakir, A., & Noor, M. R. (2021). Penerapan Convolutional Long Short-Term Memory untuk Klasifikasi Teks Berita Bahasa Indonesia (Convolutional Long Short-Term Memory Implementation for Indonesian News Classification). In *Jurnal Nasional Teknik Elektro dan Teknologi Informasi* | (Vol. 10, Number 4).

Wongvorachan, T., He, S., & Bulut, O. (2023). A Comparison of Undersampling, Oversampling, and SMOTE Methods for Dealing with Imbalanced Classification in Educational Data Mining. *Information (Switzerland)*, *14*(1). https://doi.org/10.3390/info14010054

Yudhistira, D., Saraswati, G. W., & Lutfina, E. (2025). ANALISIS SENTIMEN OPINI MASYARAKAT INDONESIA TERHADAP KASUS CYBERBULLYING DI MEDIA SOSIAL X (TWITTER) MENGGUNAKAN ALGORITMA SUPPORT VECTOR MACHINE. *Information System and Emerging Technology Journal*, *6*(2), 50131. https://doi.org/https://doi.org/10.23887/insert.v6i2.95243

*name of corresponding author