ACLM Model: A CNN-LSTM and Machine Learning Approach for Analyzing Tourist Satisfaction to Improve Priority Tourism Services

Ulya Ilhami Arsyah[1,*], Mutiana Pratiwi[2], Harfebi Fryonanda[3], M. Khairul Anam[4], Munawir[5]

[1,3]*Department of Software Engineering Technology, Politeknik Negeri Padang, Jl. Kampus, Limau Manis, Padang, 25164, Indonesia*

[2]*Department of Information System, Universitas Putra Indonesia YPTK Padang, Jl. Raya Lubuk Begalung, Padang, 25145, Indonesia*

[4,5]*Department of Informatics, Universitas Samudra, Jl. Prof. Dr. Syarief Thayeb, Langsa, 24416, Indonesia*

(Received: March 15, 2025; Revised: June 10, 2025; Accepted: September 25, 2025; Available online: October 22, 2025)

Abstract

Tourist satisfaction is a key proxy for destination service quality, yet automatic sentiment analysis of online reviews still faces class imbalance, overfitting, and limited deployability. This study proposes ACLM, a hybrid sentiment classification pipeline that learns semantic and temporal features with a CNN-LSTM backbone and evaluates three classifier heads (Softmax, Logistic Regression, XGBoost) on a three-class corpus (neutral, satisfied, dissatisfied). The objective is to deliver an accurate and operational model for decision support in tourism services. The idea combines Word2Vec embeddings, a compact CNN for local patterns, an LSTM for sequence dependencies, and a training workflow with text cleaning, SMOTE based balancing, and regularization to curb overfitting; outputs are exposed through a simple Streamlit interface. Results show that CNN-LSTM with a Softmax head attains accuracy 0.89, macro precision 0.89, macro recall 0.84, and macro F1 0.86, outperforming Logistic Regression (accuracy 0.87, macro precision 0.84, macro recall 0.82, macro F1 0.82) and XGBoost (accuracy 0.85, macro precision 0.80, macro recall 0.82, macro F1 0.80). The findings indicate that deep sequence features paired with a simple Softmax head provide the best tradeoff between accuracy and stability for three-way sentiment classification. The contribution is a reusable, end to end blueprint from preprocessing and balanced training to quantitative evaluation and an inference GUI, and the novelty lies in testing interchangeable classifier heads on a single CNN-LSTM feature extractor while explicitly addressing data imbalance and deployment constraints. The GUI is implemented using the highest accuracy model, namely CNN-LSTM with Softmax.

Keywords: CNN-LSTM, Online Review, Machine Learning, Streamlit, Tourist Satisfaction

1. Introduction

Tourism is a strategic sector that plays a crucial role in national economic growth [1]. The Indonesian government has designated several destinations as Super Priority National Tourism Strategic Areas (KSPN), including Lake Toba, Borobudur, Mandalika, Likupang, and Labuan Bajo [2]. Efforts to improve the quality of tourism services in these destinations require data-driven decision-making, particularly concerning tourist perceptions and satisfaction.

With the advancement of technology, tourist reviews distributed through digital platforms such as Google Review, TripAdvisor, and social media have become an important source of information to evaluate the quality of services and facilities at tourist destinations. However, manual analysis of these reviews is limited in terms of efficiency and objectivity. Therefore, an analytical model is needed that can automatically process textual review data and provide accurate classifications regarding tourist satisfaction levels [3].

Previous studies have explored hybrid deep learning approaches such as CNN-LSTM. For example, [4] conducted sentiment analysis using CNN-LSTM combined with early stopping to prevent overfitting and achieve optimal performance, resulting in an accuracy of 85%. Another study by [5] using CNN-LSTM achieved 91% accuracy. Similarly, sentiment analysis on product reviews using CNN-LSTM yielded an accuracy of 85% [6]. A related study [7] also reported an 86% accuracy using CNN-LSTM. In addition to deep learning, several studies utilized machine

***Corresponding author: Ulya Ilhami Arsyah (ulya@pnp.ac.id)**
DOI: https://doi.org/10.47738/jads.v6i4.974

learning methods. For instance, [8] used the Naïve Bayes algorithm for sentiment analysis and achieved 92% accuracy, while another study employing SVM reached 93% accuracy [9]. Some research has even combined both approaches, using hybrid deep learning–machine learning models.

Although some studies have proposed hybrid models such as CNN-SVM [10] or LSTM-Adaboost [11], these approaches are still limited to basic integrations and have not optimized the full feature extraction pipeline of CNN-LSTM followed by classification using more robust machine learning models such as XGBoost or Logistic Regression. Furthermore, there is a lack of research applying this hybrid approach in the context of tourist satisfaction analysis at Indonesia's super-priority destinations, which face linguistic challenges, diverse user expressions, and a critical need to utilize review data for directly improving tourism service quality. This study specifically focuses on reviews written in Indonesian, making it highly relevant to the local context but also more challenging due to the variety of informal expressions and slang commonly used by reviewers. To ensure the dataset was ready for analysis, a series of preprocessing steps were applied, including text cleaning, case folding, tokenizing, stemming, and filtering, which helped standardize the input data and improve the effectiveness of the sentiment analysis model.

This study proposes the ACLM Model (Architecture of CNN-LSTM and Machine Learning) as a solution to address the limitations of previous research regarding the integration of deep learning and machine learning, specifically in the context of analyzing tourist satisfaction. The ACLM model leverages Word2Vec as an embedding technique to convert textual data into numerical representations for further processing. The CNN architecture is used to extract spatial features from the text [12], while LSTM captures sequential patterns and long-term context [13]. The outputs from CNN-LSTM are then classified using several machines learning algorithms, including Softmax, Logistic Regression, and XGBoost, to obtain optimal classification performance. Another advantage of this approach is the deployment of the best-performing model in a GUI-based application using Streamlit, enabling real-time automated testing on new review data. Thus, the analysis results are not only academic but also practical, providing a data-driven foundation for enhancing the service quality of Indonesia's super-priority tourist destinations in a more adaptive manner.

2. Literature Review

Research on sentiment analysis in the tourism sector has been widely discussed using both machine learning and deep learning algorithms. A study by [14] compared two deep learning algorithms, CNN and LSTM, for sentiment analysis on drug reviews. The results showed good F1-Score: CNN achieved 88.44% and LSTM achieved 88.82%. Another study employed a machine learning algorithm, Random Forest, to analyze tourist reviews of Phuket, achieving an accuracy of 79.70% [15]. Further research in Indonesia applied sentiment analysis to travel agents using a machine learning approach with NSS, yielding high accuracy ranging from 95.44% to 97.71%. However, despite the high accuracy, this model performed poorly in terms of precision, recall, and F1-score [16], indicating overfitting. Overfitting occurs when a machine learning model becomes too tailored to the training data, memorizing details and noise [17]. As a result, the model performs well on training data but poorly on new or test data, indicating a lack of generalization. Overfitting can often be identified through a large gap between training and testing accuracy or rising validation error [18].

Previous studies have also implemented hybrid algorithms for sentiment analysis using deep learning. For instance, [19] combined CNN and LSTM to analyze hotel reviews, achieving a reasonably good accuracy of 77%. To address overfitting, other studies introduced dropout in CNN-LSTM architectures [20], and some also implemented early stopping [4]. Another common problem in sentiment datasets is class imbalance. When data labels are imbalanced, machine learning models tend to favor the majority class [13]. This leads to higher overall accuracy but poor performance in predicting the minority class, as reflected in low precision, recall, or F1-score values [21]. Such issues are critical in real-world tourism applications, where misclassifications can lead to misleading decisions. Prior studies have addressed this problem using techniques such as SMOTE [22], SMOTE-ENN [23], ADASYN [24].

Despite these advancements, prior research still has several limitations. Some studies such as [15] and [20] only focused on comparing or combining CNN and LSTM models without systematically addressing overfitting. Others like [16] and [17] used traditional algorithms like Random Forest and NSS, which yielded high accuracy but failed in precision and recall, indicating overfitting and inability to handle class imbalance effectively.

This study differs by not only utilizing a hybrid CNN-LSTM architecture but also integrating dropout at multiple layers and implementing early stopping to systematically prevent overfitting. Moreover, this research adopts an ensemble approach at the output layer by comparing the performance of Softmax, Logistic Regression, and XGBoost, aiming to enhance both accuracy and model stability. For class imbalance, this study applies weighting or balancing strategies integrated within the training process, rather than relying solely on preprocessing techniques.

Additionally, the proposed pipeline starts with Word2Vec embedding, followed by feature extraction using CNN, temporal processing with LSTM, and classification using various output algorithms. Evaluation is performed comprehensively using accuracy, precision, recall, and F1-score. As a practical contribution, this research also offers a GUI to facilitate interactive sentiment classification for end-users. Thus, the study not only excels technically but also presents a more practical and user-friendly solution for real-world applications, particularly in supporting decision-making based on tourist feedback.

3. Methodology

Figure 1 illustrates the proposed system workflow in this study, which integrates a hybrid CNN-LSTM architecture with several optimization techniques for sentiment analysis. The process begins with dataset preprocessing using Word2Vec embedding, followed by feature extraction through CNN and sequential modeling using LSTM. To prevent overfitting, dropout is applied at several layers along with an early stopping mechanism. The network output is passed to a dense layer and then classified using multiple approaches such as Softmax, Logistic Regression, and XGBoost. Model evaluation is conducted using accuracy, precision, recall, and F1-score metrics. As a practical implementation, the model's results are also tested through a GUI to facilitate user interaction and enable direct system usage.

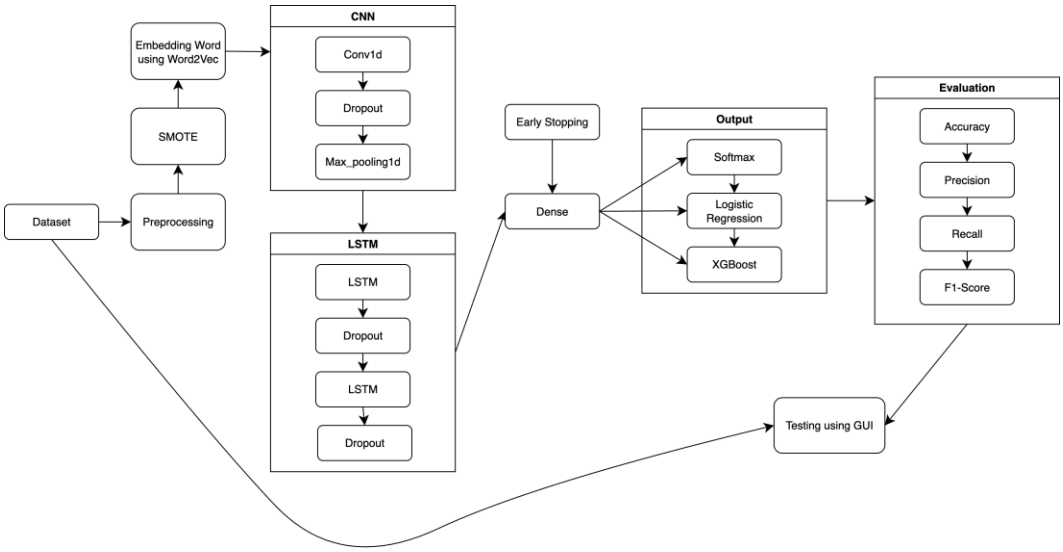


Figure 1. Methodology Flow

3.1. Dataset

This study utilizes a dataset of 4,982 user reviews obtained from the Google Review platform, specifically related to Indonesia's Super Priority Tourism Destinations such as Lake Toba, Borobudur, Mandalika, Likupang, and Labuan Bajo. These reviews reflect tourists' direct perceptions of service quality, facilities, and overall travel experiences, making them a rich and relevant source of data for analysis in the context of improving national tourism service quality. The dataset is distributed into three sentiment categories: Neutral (2,611 reviews), Satisfied (1,804 reviews), and Dissatisfied (567 reviews). This distribution highlights the predominance of neutral and positive feedback, while negative feedback is relatively limited. The data were systematically collected to cover various expressions of satisfaction and dissatisfaction, which were then used in the labeling process and sentiment analysis model training.

 ISSN 2723-6471

3.2. Preprocessing

Text preprocessing is a crucial step in Natural Language Processing (NLP) that aims to clean and prepare textual data before it is fed into a model [25]. The process begins with case folding, where all letters are converted to lowercase to prevent the model from distinguishing words based on capitalization [26]. Next, text cleaning is performed by removing irrelevant characters such as numbers, symbols, punctuation, URLs, and emojis [27]. Once the text is cleaned, tokenization is applied to split sentences into words or smaller units [28]. The resulting tokens are then filtered through stopword removal, which eliminates common words that carry little meaningful value for analysis, such as "and", "the", or "in" [29]. This is followed by stemming, which reduces words to their root forms so that different word variations with the same root are treated as the same by the model [30]. These preprocessing steps ensure the text data is clean, consistent, and ready for processing by machine learning or deep learning algorithms.

3.3. SMOTE

Synthetic Minority Over-sampling Technique (SMOTE) is a method used to address class imbalance in datasets, particularly when the number of samples in the minority class is significantly smaller than in the majority class [31]. This imbalance often causes models to become biased toward the dominant class while neglecting the minority class. SMOTE generates synthetic data for the minority class instead of simply duplicating existing samples. It works through interpolation by selecting a random point between an existing minority sample and one of its nearest neighbors, then creating a new synthetic sample at that point [32]. By increasing the diversity of the minority class, SMOTE enables models to better capture patterns from that class, thereby improving their generalization capability, especially in detecting minority cases. Figure 2 shows the initial dataset before data balancing was performed. Figure 2 shows the dataset before balancing using SMOTE.

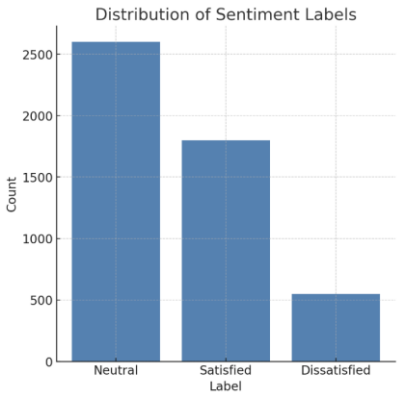


Figure 2. Dataset Before Balancing

Figure 2 shows the original distribution of sentiment labels before balancing. The dataset is dominated by the Neutral class with more than 2,500 samples, followed by the Satisfied class with around 1,800 samples, while the Dissatisfied class is the smallest, containing only about 550 samples. This imbalance indicates the necessity of applying a resampling technique to ensure the model can effectively learn from all sentiment categories. Figure 3 presents the distribution of sentiment labels after applying SMOTE. Each class: Neutral, Satisfied, and Dissatisfied has been balanced to contain 2,600 samples.

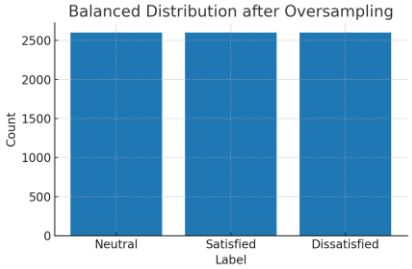


Figure 3. Dataset After Balancing with SMOTE

This balanced distribution ensures fair representation across all sentiment categories, reduces bias toward the majority class, and enhances the model's ability to classify minority sentiments more accurately. In this study, SMOTE was applied using the default parameter setting of k = 5 nearest neighbors, which determines how synthetic samples are generated through interpolation. This choice provides a good balance between diversity and stability of the synthetic data, helping the model generalize more effectively across classes.

3.4. Word2vec

In this study, the Continuous Bag of Words (CBOW) architecture of the Word2Vec model is used to transform tourist review texts into fixed-size numerical vector representations. CBOW works by predicting a target word based on its surrounding context words within a specified window size [33]. For example, if the target word is w_t, the context words are $\{w_{t-c}, \dots, w_{t-1}, w_{t+1}, \dots, w_{t+c}\}$, where c is the context window size. The objective of the CBOW model is to maximize the probability of the target word given its context, as expressed by the following equation:

$$\frac{1}{T}\sum_{t=1}^{T} \log P(w_t \mid w_{t-c}, \dots, w_{t-1}, w_{t+1}, \dots, w_{t+c}) \quad (1)$$

The conditional probability $P(w_t \mid context)$ is computed using the softmax function:

$$P(w_t \mid context) = \frac{\exp\left(v_{w_t}^{T} . h\right)}{\sum_{w=1}^{W} \exp\left(v_{w}^{T} . h\right)} \quad (2)$$

h is the average of the input word vectors from the context; v_{w_t} is the output vector of the target word w_t; W is the total vocabulary size.

CBOW was chosen in this study due to its computational efficiency and its ability to capture the semantic meaning of words based on their local context. This is particularly useful for analyzing tourist reviews, which often contain short but meaningful expressions of satisfaction or dissatisfaction. The word embeddings generated by CBOW are then used as input to the CNN-LSTM architecture in the proposed ACLM model, enhancing the model's ability to understand both semantic context and sequential word patterns in the review texts [34].

3.5. CNN-LSTM

The CNN-LSTM architecture combines CNN and LSTM networks to extract both spatial and sequential features from text data [4]. In this study, CNN is first used to detect local patterns such as n-gram features from the embedded word vectors (produced by Word2Vec-CBOW). These convolutional filters slide over the input matrix and generate feature maps that highlight important local dependencies. Mathematically, the convolution operation in 1D CNN can be expressed as:

$$f_i = \sigma\left(\sum_{j=0}^{k-1} w_j . x_{i+j} + b\right) \quad (3)$$

x is the input sequence (word embeddings); w_j is the weight of the filter of size k; b is the bias term; σ is the activation function (e.g., ReLU); f_i is the output of the convolution operation at position i.

The output from CNN is then passed to the LSTM layer, which captures long-term dependencies and the temporal sequence of the extracted features. LSTM is a type of Recurrent Neural Network (RNN) designed to overcome the vanishing gradient problem by introducing gating mechanisms: input gate i_t, forget gate f_t, and output gate o_t [35]. The LSTM operations can be described by the following equations:

$$\begin{aligned} f_t &= \sigma\left(W_f . [h_{t-1}, x_t] + b_f\right) \\ i_t &= \sigma(W_i . [h_{t-1}, x_t] + b_i) \\ \tilde{C}_t &= tanh(W_C . [h_{t-1}, x_t] + b_C) \end{aligned} \quad (4)$$

 ISSN 2723-6471

$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t)$$
$$o_t = \sigma(W_o.[h_{t-1}, x_t] + b_o)$$
$$h_t = o_t * \tanh(C_t)$$

x_t is the input at time t; h_t is the hidden state; C_t is the cell state; $W$and b are weight matrices and biases; σ is the sigmoid activation function; $tanh$ is the hyperbolic tangent function; * denotes element-wise multiplication.

By combining CNN and LSTM, the model benefits from both local feature detection (via CNN) and temporal sequence learning (via LSTM). This makes CNN-LSTM especially effective for analyzing textual data like user reviews, where the local phrase structures and word order both contribute to understanding sentiment or satisfaction levels.

3.6. Softmax

Softmax is an activation function commonly used in the output layer of neural networks for multi-class classification tasks [36]. It converts the raw output scores (logits) of the model into probabilities that sum to 1, allowing each value to be interpreted as the likelihood of a particular class. In the context of the CNN-LSTM architecture, after spatial and sequential features are extracted from the review text data, the final output is passed through the Softmax function to determine the probability of each sentiment class, such as "satisfied", "neutral", or "dissatisfied". Mathematically, the softmax function for the i -th class is defined as:

$$softmax(z_i) = \frac{e^{z_i}}{\sum_{j=i}^{K} e^{z_j}} \quad (5)$$

z_i is the raw score (logit) for class i; K is the total number of classes; e is the exponential function.

Softmax ensures that the output values are positive and normalized, making them interpretable as class probabilities. The class with the highest probability is then selected as the model's prediction, which is particularly useful in sentiment analysis tasks where the categories are mutually exclusive.

3.7. Logistic Regression

Logistic Regression is a widely used statistical model for binary and multi-class classification tasks. Unlike linear regression, which predicts continuous values, logistic regression is used to predict the probability that a given input belongs to a particular class [37]. In the context of this study, logistic regression is employed as one of the classification layers following the CNN-LSTM feature extraction, aiming to classify the sentiment or satisfaction level of tourist reviews. The model estimates the probability of a class using the logistic (sigmoid) function, which maps any real-valued number into the range [0, 1]. For binary classification, the predicted probability that an input x belongs to the positive class is calculated as:

$$P(y = 1|x) = \sigma(w^t x + b) = \frac{1}{1 + e^{-(w^T x + b)}} \quad (6)$$

x is the feature vector (e.g., output from CNN-LSTM); w is the weight vector; b is the bias term; σ is the sigmoid function.

In multi-class classification (e.g., "satisfied", "neutral", "dissatisfied"), logistic regression can be extended using the softmax function to predict the class with the highest probability. Logistic regression is favored for its simplicity, interpretability, and efficiency, especially when the relationship between features and the output class is approximately linear [38]. In this study, it serves as a lightweight yet effective classifier that complements the deep feature representations generated by CNN-LSTM.

3.8. XGBoost

Extreme Gradient Boosting (XGBoost) is an advanced implementation of gradient boosting algorithms designed for speed, performance, and scalability. It is widely used for classification and regression tasks due to its high predictive accuracy and ability to handle a wide range of data types and feature interactions [39]. In this study, XGBoost is applied as one of the classifiers following the CNN-LSTM feature extraction process, aiming to enhance sentiment

classification accuracy from tourist review texts. XGBoost builds an ensemble of decision trees in a sequential manner, where each new tree attempts to correct the errors made by the previous ones [40]. The model optimizes a regularized objective function, which consists of a loss function and a penalty term to prevent overfitting. The general objective function of XGBoost is:

$$\mathcal{L}(\phi) = \sum_{i=1}^{n} l\left(y_i, \hat{y}_i^{(t)}\right) + \sum_{k=1}^{t} \Omega(f_k) \tag{7}$$

l is a differentiable convex loss function (e.g., log loss for classification); $\hat{y}_i^{(t)}$ is the prediction of the i-th sample at interation t; f_k is the k-th decision tree; $\Omega(f_k) = \gamma T + \frac{1}{2}\lambda \sum_j w_j^2$ is the regularization term (where T is the number of leaves, w_j is the score on each leaf).

XGBoost uses second-order Taylor expansion to approximate the loss function, allowing for efficient optimization. It also includes built-in handling for missing values, parallel processing, and tree pruning strategies that improve performance over traditional gradient boosting methods. In the context of this research, XGBoost is chosen for its robustness and ability to capture complex non-linear relationships in the extracted features, making it a powerful alternative to traditional classifiers like Softmax or Logistic Regression.

3.9. Early Stopping

Early stopping is a regularization technique used in training machine learning models, particularly neural networks, to prevent overfitting [41]. It works by halting the training process early when the model's performance on the validation data begins to decline, even if the accuracy on the training data continues to improve. During training, the model is periodically evaluated on validation data. If no significant improvement is observed over a defined number of epochs, such as a decrease in validation loss the training is automatically stopped. This approach ensures that the resulting model maintains optimal performance without overfitting to the training data. Early stopping is highly beneficial for avoiding unnecessarily long training durations and for promoting good generalization to new, unseen data [42].

3.10. Evaluation

To assess the performance of the proposed CNN-LSTM and Machine Learning hybrid models, this study employs a set of standard classification evaluation metrics, including Accuracy, Precision, Recall, and F1-Score. These metrics provide a comprehensive understanding of the model's ability to correctly classify tourist sentiment based on textual reviews [43]. Accuracy measures the overall correctness of the model's predictions, while precision reflects the proportion of true positive predictions among all positive predictions. Recall evaluates the model's ability to identify all relevant instances, and F1-Score serves as a harmonic mean of precision and recall, offering a balanced metric especially useful in the presence of class imbalance.

The evaluation process involves splitting the dataset into training and testing subsets, followed by the application of the trained models to unseen data. Confusion matrices and classification reports are used to visualize and interpret the results for each class label (e.g., satisfied, dissatisfied). Additionally, the study compares the performance of different classifiers (Softmax, Logistic Regression, and XGBoost) applied after CNN-LSTM feature extraction, to determine the most effective approach. This evaluation strategy ensures that the selected model not only performs well on the training data but also generalizes effectively to new, real-world review data.

3.11. GUI with Streamlit

To enhance usability and support real-time sentiment analysis, this study integrates the best-performing model into a GUI using Streamlit, an open-source Python framework designed for building interactive web applications for data science and machine learning. Streamlit enables rapid deployment of models with minimal front-end coding, allowing users such as tourism stakeholders or decision-makers to input new review texts and instantly receive sentiment predictions (e.g., satisfied, neutral, dissatisfied) based on the trained hybrid CNN-LSTM + Machine Learning model.

The GUI accepts raw textual input from users, processes it through the same preprocessing and embedding pipeline used during training (including tokenization and Word2Vec-CBOW embedding), and feeds it into the saved model for

 ISSN 2723-6471

inference. The output is then displayed in a clear and user-friendly format, including the predicted sentiment and associated probability scores. This implementation transforms the research model into a practical decision-support tool that can be used in real-world tourism service evaluation and policy-making, particularly in Super Priority Destinations in Indonesia.

4. Results and Discussions

4.1. Result

To support the sentiment classification process of tourist reviews, the model used in this study was designed by integrating several interconnected neural network layers. The architecture leverages the strength of CNN in extracting spatial features and LSTM networks in capturing sequential patterns from textual data. The details of each layer in the model architecture are presented in table 1.

Table 1. Model Architecture

Layer (type)	Output Shape	Information
Input_layer	(none, 35)	The input layer receives data in the form of a token sequence with a length of 35 words (sequence length = 35).
Embedding	(none, 35, 100)	The word embedding layer transforms each token into a vector of 100 dimensions. The total number of parameters is 471,100, derived from the vocabulary size × embedding dimension.
Conv1d	(none, 31, 256)	A 1D convolutional layer with 256 filters. The output length becomes 31 due to the use of valid (non-padded) convolution.
dropout	(none, 31, 256)	A dropout layer that helps prevent overfitting by randomly deactivating certain neurons. This layer has no trainable parameters.
Max_pooling1d	(none, 15, 256)	A max pooling layer that reduces the sequence length by selecting the maximum value over a fixed window, resulting in a sequence length of 15.
lstm	(none, 15, 128)	The first LSTM layer with 128 units, processes sequential data output from the CNN layer.
Dropout_1	(none, 15, 128)	A dropout layer applied after the first LSTM for regularization.
Lstm_1	(none, 128)	The second LSTM layer, which returns only the final output of the sequence (return_sequences=False).
Dropout_2	(None, 128)	An additional dropout layer applied before the dense layer.
Dense	(None, 64)	A fully connected (dense) layer with 64 units to refine the feature representation learned by the LSTM.
Dense_1	(None, 3)	The output layer with 3 units (corresponding to the classes: satisfied, neutral, dissatisfied) using a Softmax activation function.

Statistical Parameters: Total Parameters: 936,511 → Represents the total number of trainable weights. Trainable Parameters: 936,511 → All parameters in the model are trainable. Non-trainable Parameters: 0 → No parts of the model are frozen (no frozen layers).

This model employs a hybrid CNN-LSTM approach, where the CNN captures local features and the LSTM captures temporal sequences in the text data. The architecture is followed by several dropout and dense layers to prevent overfitting and to classify the input into three categories. It is well-balanced for text analysis tasks such as tourist review classification, offering a moderate level of complexity that is suitable for deployment in GUI-based applications like Streamlit. Figure 4 illustrates the model training process using early stopping and the softmax activation function.

 ISSN 2723-6471


```
Epoch 1/60
141/141 ━━━━━━━━━━━━━━━━━━━━ 21s 108ms/step - accuracy: 0.5325 - loss: 0.9563 - val_accuracy: 0.7655 - val_loss: 0.6477
Epoch 2/60
141/141 ━━━━━━━━━━━━━━━━━━━━ 14s 101ms/step - accuracy: 0.8012 - loss: 0.5368 - val_accuracy: 0.8417 - val_loss: 0.4617
Epoch 3/60
141/141 ━━━━━━━━━━━━━━━━━━━━ 15s 104ms/step - accuracy: 0.8804 - loss: 0.3233 - val_accuracy: 0.8858 - val_loss: 0.3384
Epoch 4/60
141/141 ━━━━━━━━━━━━━━━━━━━━ 20s 100ms/step - accuracy: 0.9226 - loss: 0.2225 - val_accuracy: 0.8697 - val_loss: 0.3584
Epoch 5/60
141/141 ━━━━━━━━━━━━━━━━━━━━ 22s 109ms/step - accuracy: 0.9319 - loss: 0.1940 - val_accuracy: 0.8838 - val_loss: 0.3523
Epoch 6/60
141/141 ━━━━━━━━━━━━━━━━━━━━ 20s 102ms/step - accuracy: 0.9507 - loss: 0.1427 - val_accuracy: 0.8517 - val_loss: 0.4055
CPU times: user 2min 12s, sys: 6.03 s, total: 2min 18s
Wall time: 1min 57s
```

Figure 4. Training Model using Softmax

The training process for the CNN-LSTM model was set with a maximum of 60 epochs. However, it was terminated early at the 6th epoch using the Early Stopping technique. This technique aims to prevent overfitting by monitoring the validation loss and halting training when no significant improvement is observed. At the start of training (epoch 1), the model achieved a training accuracy of 53.25% and a validation accuracy of 76.65%, with a relatively high loss value. As training progressed, the accuracy improved consistently, and by epoch 6, the model reached a training accuracy of 95.07% and a validation accuracy of 85.17%, with the validation loss reduced to 0.4095. These results indicate that the model had reached an optimal point in terms of accuracy and generalization to the validation data. The use of Early Stopping enabled a more efficient and effective training process, preventing overtraining and conserving computational resources—particularly important when working with limited data and complex model architectures. Figure 5 presents the testing results of the CNN-LSTM model with the Softmax activation function.

	precision	recall	f1-score	support
0	0.89	0.92	0.91	259
1	0.88	0.89	0.88	185
2	0.89	0.71	0.79	55
accuracy			0.89	499
macro avg	0.89	0.84	0.86	499
weighted avg	0.89	0.89	0.88	499

Figure 5. CNN-LSTM Classification Report Result with Softmax

Figure 5 presents the classification report results of the CNN-LSTM model using the Softmax activation function in the output layer. The evaluation was performed across three sentiment classes: class 0 (satisfied), class 1 (neutral), and class 2 (dissatisfied). The model achieved strong performance for class 0, with a precision of 0.89, recall of 0.92, and an F1-score of 0.91, indicating high capability in detecting positive sentiment. Similarly, for class 1 (neutral), the model yielded a precision of 0.88, recall of 0.89, and an F1-score of 0.88, showing reliable consistency in identifying neutral feedback.

However, for class 2 (dissatisfied), although the precision was still high at 0.89, the recall dropped to 0.71, resulting in an F1-score of 0.79. This indicates a relative difficulty in detecting dissatisfied sentiments. The lower recall may not only be influenced by the smaller number of samples in class 2 (only 55) compared to classes 0 and 1, but also by other factors such as vocabulary sparsity, sentiment ambiguity in user expressions, or the inherent effects of class imbalance. Overall, the model obtained an accuracy of 88.96%, with a balanced macro-average F1-score of 0.86. The next evaluation was carried out using graphical plots, as presented in figure 6.

 ISSN 2723-6471

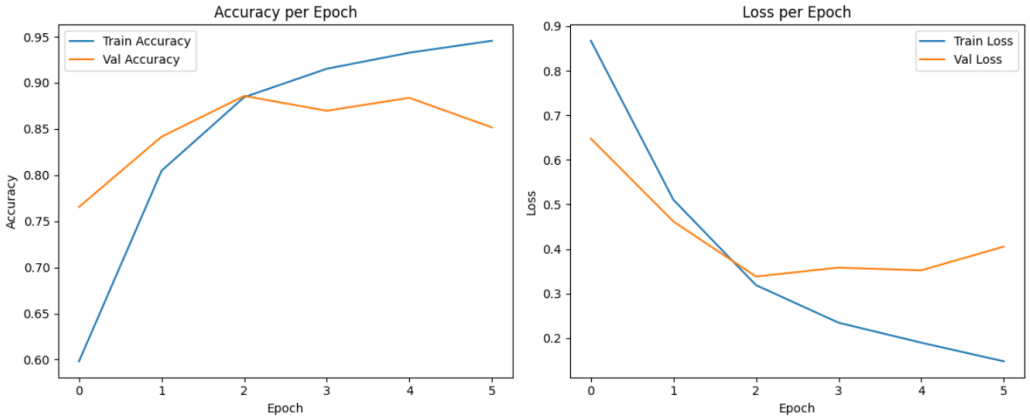
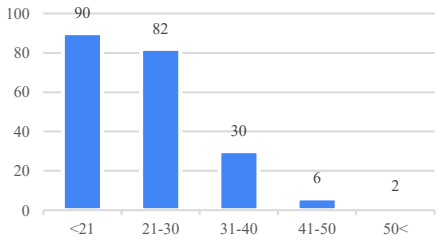


Figure 6. Accuracy and Loss Plots of CNN-LSTM + Softmax

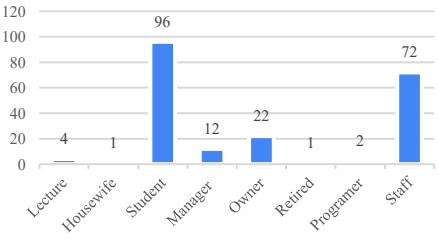
Figure 6 illustrates the accuracy and loss curves of the CNN-LSTM model with Softmax during the training process. The accuracy plot shows a consistent increase in both training and validation data, with validation accuracy reaching approximately 0.90 by the fourth epoch and remaining stable until the end of training. This confirms that the model was able to effectively learn patterns while maintaining strong performance on validation data. Meanwhile, the loss plot demonstrates a significant decline from the first to the fifth epoch for both training and validation datasets. This trend indicates that the model effectively reduced prediction errors as the number of epochs increased, with the loss curves continuing to decrease in a stable manner. Overall, the results in figure 6 confirm that the CNN-LSTM with Softmax achieved high accuracy and strong stability, while also demonstrating robust generalization ability on the tested data. The subsequent evaluation was conducted using alternative output classifiers, namely XGBoost and Logistic Regression.



XGBoost:

	precision	recall	f1-score	support
0	0.88	0.88	0.88	259
1	0.87	0.84	0.85	185
2	0.66	0.75	0.70	55
accuracy			0.85	499
macro avg	0.80	0.82	0.81	499
weighted avg	0.85	0.85	0.85	499

(a)



Logistic Regression:

	precision	recall	f1-score	support
0	0.89	0.90	0.89	259
1	0.88	0.87	0.88	185
2	0.74	0.76	0.75	55
accuracy			0.87	499
macro avg	0.84	0.84	0.84	499
weighted avg	0.87	0.87	0.87	499

(b)

Figure 7. Testing with Other Outputs

In Figure 7a, the CNN-LSTM combined with XGBoost achieved an accuracy of 85%, with performance metrics (precision, recall, F1-score) showing greater variation among classes, particularly with class 2 scoring a recall of only 0.56, which pulled the overall macro average F1-score down to 0.84. This reflects the classifier's less stable performance, especially for underrepresented classes.

In contrast, Figure 7b shows that pairing CNN-LSTM with Logistic Regression as the output classifier improved the model's accuracy to 87%. The evaluation metrics were more consistent across all sentiment classes, resulting in a macro average F1-score of 0.84 and a weighted average of 0.87. Although slightly lower than the Softmax approach, Logistic Regression offered a more balanced classification than XGBoost.

The relatively lower performance of CNN-LSTM + XGBoost can be attributed to the incompatibility between tree-based boosting methods and sequential representations generated by LSTM. While XGBoost excels at handling tabular features, it may fail to fully capture contextual dependencies preserved in sequence embeddings. In addition, the fixed representation passed from LSTM to XGBoost may lose temporal information, leading to weaker generalization on minority classes such as "dissatisfied".

In summary, among the three configurations, the CNN-LSTM + Softmax model not only achieved the highest accuracy (89%) but also demonstrated balanced performance across classes, making it the most effective and reliable choice for sentiment classification in this study. This reinforces its suitability for real-world applications, particularly when implemented in GUI-based systems for evaluating tourist reviews or service quality in the tourism sector. Figure 8 illustrates the implementation of this model in a user interface using Streamlit for real-time sentiment classification.

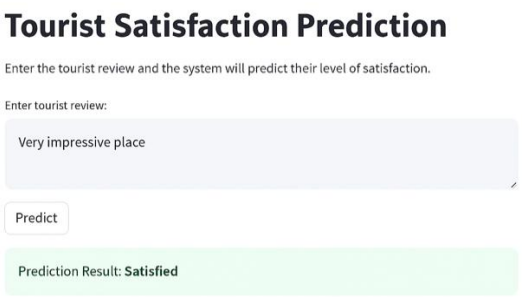


Figure 8. Testing with GUI using Streamlit

Figure 8 displays the results of the tourist satisfaction prediction system tested through a GUI built using the Streamlit framework. In this test, the model used is a CNN-LSTM with the Softmax activation function. The interface is designed to be simple and intuitive, allowing users to input a tourist review in the provided text box and then click the "Predict" button to view the system's prediction results.

In the displayed example, the input "Very impressive place" is processed by the system, and the prediction result is automatically shown in a light green box labeled "Prediction Result: Satisfied." This GUI implementation demonstrates that the model can be practically integrated into a web-based application, enabling non-technical users to evaluate tourist perceptions in real time.

4.2. Discussion

This study proposes a hybrid approach by integrating a CNN-LSTM architecture with several optimization techniques such as SMOTE and Early Stopping for classifying tourist review sentiments. As illustrated in figure 1, the process begins with text preprocessing, followed by class balancing using SMOTE and word embedding through the Word2Vec model with a CBOW architecture. The resulting vector representations are processed through CNN to extract spatial features and then passed to the LSTM layer to capture temporal sequences. The application of dropout in several layers, combined with the Early Stopping mechanism, effectively prevented overfitting during training. Training was automatically stopped at the 6th epoch when no further improvement was observed on the validation set. Evaluation was carried out using accuracy, precision, recall, and F1-score metrics.

The evaluation results demonstrate that the CNN-LSTM model with Softmax activation achieved the best performance compared to other approaches such as CNN-LSTM + Logistic Regression and CNN-LSTM + XGBoost. The CNN-LSTM + Softmax model not only yielded the highest accuracy but also provided consistent performance across the three sentiment classes. The strength of Softmax in generating proportional probability distributions makes it well-suited for multi-class classification tasks like tourist sentiment analysis. Additionally, the use of SMOTE during

training contributed significantly to improving the model's ability to represent minority classes, helping the model to learn more balanced patterns. This played a key role in enhancing overall classification performance.

To ensure the model's usability in real-world applications, it was integrated into a GUI using Streamlit. The testing results showed that users can easily input tourist review texts and obtain sentiment predictions instantly. Thus, the developed model offers not only strong technical accuracy but also high practical value for implementation in tourism service evaluation systems. Thus, this approach not only produces an accurate classification model but also offers high practical value for real-world applications in the tourism sector. The next step is to conduct a comparison with previous studies, as shown in table 2.

Table 2. Comparison with Previous Research

Researcher	Model	Class Imbalance Handling Methods	Accuracy
[44]	CNN+Adaboost	SMOTE	86%
[45]	CNN+SVM	Does not employ	87%
[46]	BiLSTM+SVM	Does not employ	86%
[47]	GRU-SVM	Does not employ	82%
[48]	CNN-LSTM-Machine Learning	Does not employ	RF: 84%, LR: 85%, SVM: 85%, NB: 85%, KNN: 84%
This Research	CNN-LSTM+Softmax	SMOTE	89%
This Research	CNN-LSTM+XGBoost	SMOTE	85%
This Research	CNN-LSTM+Logistic Regression	SMOTE	87%

Table 2 presents a comparison of several hybrid deep learning and machine learning models for sentiment analysis, focusing on the use of class imbalance handling methods and overall accuracy. Previous studies that did not employ imbalance handling methods, such as CNN+SVM [45], BiLSTM+SVM [46], GRU-SVM [47], and CNN-LSTM combined with traditional machine learning classifiers [48], achieved accuracy ranging between 82% and 87%. In contrast, models that incorporated SMOTE, such as CNN+Adaboost [44] and the proposed CNN-LSTM variations in this research, demonstrated stronger performance, with the CNN-LSTM+Softmax achieving the highest accuracy of 89%. These results highlight the effectiveness of integrating SMOTE in addressing class imbalance and improving model performance compared to approaches that do not utilize such techniques.

The integration of CNN, LSTM, SMOTE, and dropout provides significant contributions to sentiment analysis in the tourism sector. CNN effectively extracts local patterns from tourist reviews, such as key words and phrases that indicate satisfaction or dissatisfaction, while LSTM captures long-term contextual dependencies to better understand the overall sentiment expressed in longer texts. SMOTE balances the dataset by generating synthetic samples for the underrepresented "dissatisfied" class, ensuring fairer classification and enabling the model to better detect negative feedback that is critical for improving tourism services. Meanwhile, dropout prevents overfitting and strengthens the model's generalization so it can perform reliably on unseen reviews from diverse destinations. Together, these components led to high accuracy (89%) and consistent performance across sentiment classes, demonstrating their value in enhancing service quality analysis within the tourism industry.

In addition to its relevance in the tourism sector, this approach also has the potential to be applied in other domains that rely on user reviews, such as education, e-commerce, banking, and public services. For instance, in the education sector, the model can help analyze student feedback on the quality of learning; in e-commerce, it can be used to identify customer satisfaction or complaints regarding products; while in public services, it can be employed to assess community responses to government policies or service delivery. This demonstrates that the combination of CNN, LSTM, SMOTE, and dropout is not only beneficial for tourism but also possesses strong generalization capabilities across multiple fields.

5. Conclusion

This study presents a hybrid deep learning and machine learning model, ACLM, for classifying tourist satisfaction based on online review texts. By combining the strengths of CNN in spatial feature extraction and LSTM in capturing temporal dependencies, the model effectively learns from text data. The best performance was achieved using the Softmax classifier, with 89% accuracy and 86% F1-score, proving to be superior compared to other approaches including Logistic Regression and XGBoost.

Furthermore, the implementation of the model in a user-friendly GUI using Streamlit demonstrates its real-world applicability for non-technical users. This practical aspect reinforces the model's potential for use in tourism management systems, particularly in Indonesia's super-priority destinations. At the same time, it is important to note that the study employed the CBOW model for word embeddings without utilizing Skip-gram, and the dataset was restricted to the tourism sector. As such, further testing on datasets from other domains is still required to fully evaluate the model's generalizability.

For future work, several directions can be pursued to address the current study's limitations and further enhance model performance. First, alternative embedding methods such as Skip-gram, FastText, or transformer-based embeddings could be explored to overcome the limitations of using only CBOW, particularly in handling rare or domain-specific terms. Second, the dataset, which in this study is limited to the tourism sector, should be expanded and tested on reviews from other domains such as education, e-commerce, or public services to evaluate the model's generalizability. Third, usability testing of the developed GUI with real users' needs to be conducted to assess its effectiveness, efficiency, and user satisfaction in practical applications. In addition, incorporating attention mechanisms or transformer architectures may improve the model's ability to capture contextual nuances in review texts, while domain adaptation techniques and multilingual models could strengthen its robustness across different languages and cultural contexts. Finally, advanced data augmentation methods and approaches for handling class imbalance, such as GANs or cost-sensitive learning, remain promising directions for improving classification performance.

6. Declarations

6.1. Author Contributions

Conceptualization: U.I.A., and M.K.A.; Methodology: H.F.; Software: M.K.A.; Validation: M.P., and M.; Formal Analysis: H.F., M., and M.P.; Resources: M.K.A.; Data Curation: U.I.A.; Writing Original Draft Preparation: H.F., M., and M.P.; Writing Review and Editing: M.K.A., U.I.A., and H.F.; Visualization: M.P.; All authors have read and agreed to the published version of the manuscript.

6.2. Data Availability Statement

The data presented in this study are available on request from the corresponding author.

6.3. Funding

This work was supported by the Ministry of Higher Education, Science, and Technology, Directorate General of Research and Development, Republic of Indonesia under Research Grant Contract Number 725/PL9.15/AL.04/2025.

6.4. Institutional Review Board Statement

Not applicable.

6.5. Informed Consent Statement

Not applicable.

6.6. Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] S. Ma and L. Zhang, "Enhancing Tourists' Satisfaction: Leveraging Artificial Intelligence in the Tourism Sector," *Pacific International Journal*, vol. 7, no. 3, pp. 89–98, Jun. 2024, doi: 10.55014/pij.v7i3.624.

[2] B. A. Utari, S. Arsallia, M. A. Ramdani, F. Rahmafitria, P. F. Belgiawan, P. Dirgahayani, R. A. Nasution, "Tourist destination choice on five priority destinations of Indonesia during health crisis," *Journal of Destination Marketing and Management*, vol. 32, no. 1, pp. 1–12, Jun. 2024, doi: 10.1016/j.jdmm.2024.100880.

[3] S. Z. Pranida and A. Kurniawardhani, "Sentiment Analysis of Expedition Customer Satisfaction using BiGRU and BiLSTM," *Indonesian Journal of Artificial Intelligence and Data Mining*, vol. 5, no. 1, pp. 44–53, Jun. 2022, doi: 10.24014/ijaidm.v5i1.17361.

[4] M. K. Anam, S. Defit, Haviluddin, L. Efrizoni, and M. B. Firdaus, "Early Stopping on CNN-LSTM Development to Improve Classification Performance," *Journal of Applied Data Sciences*, vol. 5, no. 3, pp. 1175–1188, 2024, doi: 10.47738/jads.v5i3.312.

[5] S. Mallick, P. Awasthi, A. K. Yadav, B. Sri G, R, Porwal, N. Bhatia, J. Kataria, "Deep Hybrid CNN-LSTM Framework For Advanced Social Media Sentiment Analysis In Data-Driven Marketing Analytic," *Int J Environ Sci*, vol. 11, no. 6, pp. 169–180, 2025, doi: 10.64252/82p5f018.

[6] N. El Koufi, Y. M. Missah, and A. Belangour, "A Hybrid CNN-LSTM Based Natural Language Processing Model for Sentiment Analysis of Customer Product Reviews: A Case Study from Ghana," *Journal of Hunan University Natural Sciences*, vol. 51, no. 8, pp. 59–71, 2024, doi: 10.55463/issn.1674-2974.51.8.5.

[7] A. N. H. Zaied and G. Soliman, "Arabic Sentiment Analysis using Deep Learning and Machine Learning approaches," *Journal of Computing and Communication*, vol. 3, no. 2, pp. 10–22, 2024, doi: 10.21608/jocc.2024.380113.

[8] M. F. Madjid, D. E. Ratnawati, and B. Rahayudi, "Sentiment Analysis on App Reviews Using Support Vector Machine and Naïve Bayes Classification," *Sinkron*, vol. 8, no. 1, pp. 556–562, Feb. 2023, doi: 10.33395/sinkron.v8i1.12161.

[9] M. Sakhdiah, A. Salma, D. Permana, and D. Fitria, "Sentiment Analysis Using Support Vector Machine (SVM) of ChatGPT Application Users in Play Store," *UNP Journal of Statistics and Data Science*, vol. 2, no. 2, pp. 151–158, May 2024, doi: 10.24036/ujsds/vol2-iss2/158.

[10] A. E. Minarno, M. Fadhlan, Y. Munarko, and R. Chandranegara, "Classification of Dermoscopic Images Using CNN-SVM," *International of Dermoscopic Images Using CNN-SVM*, vol. 3, no. 2, pp. 606–612, 2024, doi: 10.62527/joiv.8.2.2153.

[11] G. A. Busari and D. H. Lim, "Crude oil price prediction: A comparison between AdaBoost-LSTM and AdaBoost-GRU for improving forecasting performance," *Comput Chem Eng*, vol. 155, no. 1, pp. 1–9, Dec. 2021, doi: 10.1016/j.compchemeng.2021.107513.

[12] Z. Yang, W. Li, F. Yuan, H. Zhi, M. Guo, B. Xin, Z. Gao "Hybrid CNN-BiLSTM-MHSA Model for Accurate Fault Diagnosis of Rotor Motor Bearings," *Mathematics*, vol. 13, no. 334, pp. 1–28, Jan. 2025, doi: 10.3390/math13030334.

[13] M. K. Anam, L. L. Van FC, H. Hamdani, R. Rahmaddeni, J. Junadhi, M. B. Firdaus, I. Syahputra, Y. Irawan "Sara Detection on Social Media Using Deep Learning Algorithm Development," *Journal of Applied Engineering and Technological Science*, vol. 6, no. 1, pp. 225–237, Dec. 2024, doi: 10.37385/jaets.v6i1.5390.

[14] C. Colón-Ruiz and I. Segura-Bedmar, "Comparing deep learning architectures for sentiment analysis on drug reviews," *J Biomed Inform*, vol. 110, no. 1, pp. 1–11, Oct. 2020, doi: 10.1016/j.jbi.2020.103539.

[15] N. Leelawat, S. Jariyapongpaiboon, A. Promjun, S. Boonyarak, K. Saengtabtim, A. Laosunthara, A. K. Yudha, J. Tang, "Twitter data sentiment analysis of tourism in Thailand during the COVID-19 pandemic using machine learning," *Heliyon*, vol. 8, no. 10, pp. 1–11, Oct. 2022, doi: 10.1016/j.heliyon.2022.e10894.

[16] A. D. Poernomo and S. Suharjito, "Indonesian online travel agent sentiment analysis using machine learning methods," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 14, no. 1, pp. 113–117, Apr. 2019, doi: 10.11591/ijeecs.v14.i1.pp113-117.

[17] C. Xu, P. Coen-Pirani, and X. Jiang, "Empirical Study of Overfitting in Deep Learning for Predicting Breast Cancer Metastasis," *Cancers (Basel)*, vol. 15, no. 7, pp. 1–18, Apr. 2023, doi: 10.3390/cancers15071969.

[18] Hartono and E. Ongko, "Avoiding Overfitting dan Overlapping in Handling Class Imbalanced Using Hybrid Approach with Smoothed Bootstrap Resampling and Feature Selection," *International Journal on Informatics Visualization*, vol. 2, no. 6, pp. 343–348, 2022, doi: 10.30630/joiv.6.2.985.

[19] Y. A. Singgalen, "A Hybrid CNN-LSTM Model with SMOTE for Enhanced Sentiment Analysis of Hotel Reviews," *Technology and Science (BITS)*, vol. 6, no. 3, pp. 1363–1373, 2024, doi: 10.47065/bits.v6i3.6301.

[20] A. Alasmari, N. Farooqi, and Y. Alotaibi, "Sentiment analysis of pilgrims using CNN-LSTM deep learning approach," *PeerJ Comput Sci*, vol. 10, no.1, pp. 1–47, 2024, doi: 10.7717/PEERJ-CS.2584.

[21] T. Miftahushudur, H. M. Sahin, B. Grieve, and H. Yin, "A Survey of Methods for Addressing Imbalance Data Problems in Agriculture Applications," *Remote Sens (Basel)*, vol. 17, no. 3, pp. 1–31, Feb. 2025, doi: 10.3390/rs17030454.

[22] N. Matondang and N. Surantha, "Effects of oversampling SMOTE in the classification of hypertensive dataset," *Advances in Science, Technology and Engineering Systems*, vol. 5, no. 4, pp. 432–437, 2020, doi: 10.25046/AJ050451.

[23] M. A. Latief, L. R. Nabila, W. Miftakhurrahman, S. Ma'rufatullah, and H. Tantyoko, "Handling Imbalance Data using Hybrid Sampling SMOTE-ENN in Lung Cancer Classification," *International Journal of Engineering and Computer Science Applications (IJECSA)*, vol. 3, no. 1, pp. 11–18, Feb. 2024, doi: 10.30812/ijecsa.v3i1.3758.

[24] D. Datta, P. K. Mallick, A. V. N. Reddy, M. A. Mohammed, M. M. Jaber, A. S. Alghawli, M. A. A. Al-qaness "A Hybrid Classification of Imbalanced Hyperspectral Images Using ADASYN and Enhanced Deep Subsampled Multi-Grained Cascaded Forest," *Remote Sens (Basel)*, vol. 14, no. 19, pp. 1-12, Oct. 2022, doi: 10.3390/rs14194853.

[25] R. J. Dhanal and V. R. Ghorpade, "Aspect-based sentiment-analysis using topic modelling and machine-learning," *International Journal of Electrical and Computer Engineering*, vol. 14, no. 6, pp. 6689–6698, Dec. 2024, doi: 10.11591/ijece.v14i6.pp6689-6698.

[26] S. Agarwal, "Deep Learning-based Sentiment Analysis: Establishing Customer Dimension as the Lifeblood of Business Management," *Global Business Review*, vol. 23, no. 1, pp. 119–136, Feb. 2022, doi: 10.1177/0972150919845160.

[27] M. K. Anam, T. P. Lestari, H. Yenni, T. Nasution, and M. B. Firdaus, "Enhancement of Machine Learning Algorithm in Fine-grained Sentiment Analysis Using the Ensemble," *ECTI Transactions on Computer and Information Technology (ECTI-CIT)*, vol. 19, no. 2, pp. 159–167, Mar. 2025, doi: 10.37936/ecti-cit.2025192.257815.

[28] M. K. Anam, M. Munawir, L. Efrizoni, N. Fadillah, W. Agustin, I. Syahputra, T. P. Lestari, M. B. Firdaus, L. Lathifah, A. K. Sari, "Improved Performance of Hybrid GRU-BiLSTM for Detection Emotion on Twitter Dataset," *Journal of Applied Data Sciences*, vol. 6, no. 1, pp. 354–365, Jan. 2025, doi: 10.47738/jads.v6i1.459.

[29] M. K. Anam, P. P. Pratama, R. A. Malik, K. Karfindo, T. A. Putra, Y. Elva, R. A. Mahessya, M. B. Firdaus, I. Ikhsan, C. R. Gunawan, "Enhancing the Performance of Machine Learning Algorithm for Intent Sentiment Analysis on Village Fund Topic," *Journal of Applied Data Sciences*, vol. 6, no. 2, pp. 1102–1115, 2025, doi: 10.47738/jads.v6i2.637.

[30] Z. Abidin, A. Junaidi, and Wamiliana, "Text Stemming and Lemmatization of Regional Languages in Indonesia: A Systematic Literature Review," *Journal of Information Systems Engineering and Business Intelligence*, vol. 10, no. 2, pp. 217–231, Jun. 2024, doi: 10.20473/jisebi.10.2.217-231.

[31] S. Priyadarshinee and M. Panda, "Cardiac Disease Prediction Using SMOTE and Machine Learning Classifiers," *Journal of Pharmaceutical Negative Results* , vol. 13, no. 8, pp. 856–862, 2022, doi: 10.47750/pnr.2022.13.S08.108.

[32] G. Husain, D. Nasef, R. Jose, J. Mayer, M. Bekbolatova, T. Devine, M. Toma, "SMOTE vs. SMOTEENN: A Study on the Performance of Resampling Algorithms for Addressing Class Imbalance in Regression Models," *Algorithms*, vol. 18, no. 1, pp. 1–16, Jan. 2025, doi: 10.3390/a18010037.

[33] A. Fellah, A. Zahaf, and A. Elçi, "Semantic Similarity Measure Using a Combination of Word2Vec and WordNet Models," *Indonesian Journal of Electrical Engineering and Informatics*, vol. 12, no. 2, pp. 455–464, Jun. 2024, doi: 10.52549/ijeei.v12i2.5114.

[34] V. Yadav, P. Verma, and V. Katiyar, "Enhancing sentiment analysis in Hindi for E-commerce companies: a CNN-LSTM approach with CBoW and TF-IDF word embedding models," *International Journal of Information Technology (Singapore)*, vol. 2023, no. 1, pp. 1–16, 2023, doi: 10.1007/s41870-023-01596-x.

[35] P. K. Jain, V. Saravanan, and R. Pamula, "A Hybrid CNN-LSTM: A Deep Learning Approach for Consumer Sentiment Analysis Using Qualitative User-Generated Contents," *ACM Transactions on Asian and Low-Resource Language Information Processing*, vol. 20, no. 5, pp. 1–15, 2021, doi: 10.1145/3457206.

[36] Q. Zhu, Z. He, T. Zhang, and W. Cui, "Improving classification performance of softmax loss function based on scalable batch-normalization," *Applied Sciences (Switzerland)*, vol. 10, no. 1, pp. 1–8, Apr. 2020, doi: 10.3390/APP10082950.

[37] A. Desfiandi and B. Soewito, "Student Graduation Time Prediction Using Logistic Regression, Decision Tree, Support Vector, And Adaboost Ensemble Learning," *International Journal of Information System and Computer Science) IJISCS*, vol. 7, no. 3, pp. 195–199, 2023, doi: 10.56327/ijiscs.v7i2.1579.

[38] Herianto, B. Kurniawan, Z. H. Hartomi, Y. Irawan, and M. K. Anam, "Machine Learning Algorithm Optimization using Stacking Technique for Graduation Prediction," *Journal of Applied Data Sciences*, vol. 5, no. 3, pp. 1272–1285, Sep. 2024, doi: 10.47738/jads.v5i3.316.

[39] D. Tarwidi, S. R. Pudjaprasetya, D. Adytia, and M. Apri, "An optimized XGBoost-based machine learning method for predicting wave run-up on a sloping beach," *MethodsX*, vol. 10, no. 1, pp. 1–12, Aug. 2023, doi: 10.1145/2939672.2939785.

[40] T. Sugihartono, B. Wijaya, Marini, A. F. Alkayes, and H. A. Anugrah, "Optimizing Stunting Detection through SMOTE and Machine Learning: a Comparative Study of XGBoost, Random Forest, SVM, and k-NN," *Journal of Applied Data Sciences*, vol. 6, no. 1, pp. 667–682, Jan. 2025, doi: 10.47738/jads.v6i1.494.

[41] C. W. Bartlett, J. Bossenbroek, Y. Ueyama, P. McCallinhart, O. A. Peters, D. A. Santillan, M. K. Santillan, A. J. Trask, W. C. Ray, "Invasive or More Direct Measurements Can Provide an Objective Early-Stopping Ceiling for Training Deep Neural Networks on Non-invasive or Less-Direct Biomedical Data," *SN Comput Sci*, vol. 4, no. 2, pp. 1–12, Mar. 2023, doi: 10.1007/s42979-022-01553-8.

[42] M. Vilares Ferro, Y. Doval Mosquera, F. J. Ribadas Pena, and V. M. Darriba Bilbao, "Early stopping by correlating online indicators in neural networks," *Neural Networks*, vol. 159, no. 1, pp. 109–124, Feb. 2023, doi: 10.1016/j.neunet.2022.11.035.

[43] K. L. Tan, C. P. Lee, and K. M. Lim, "A Survey of Sentiment Analysis: Approaches, Datasets, and Future Research," *Applied Sciences (Switzerland)*, vol. 13, no. 7, pp. 1–21, Apr. 2023, doi: 10.3390/app13074550.

[44] S. Vora and Dr. R. G. Mehta, "Hcnnxgboost: A Hybrid Cnn-Xgboost Approach for Effective Emotion Detection in Textual Data," *International Journal of Innovative Technology and Exploring Engineering*, vol. 13, no. 10, pp. 12–17, Sep. 2024, doi: 10.35940/ijitee.J9959.13100924.

[45] Raviya K and M. Vennila S, "Aspect-Based Sentiment Analysis Using Hybrid CNN-SVM with Particle Swarm Optimization for Domain Independent Datasets," *International Journal of Emerging Trends in Engineering Research*, vol. 8, no. 10, pp. 7014–7023, Oct. 2020, doi: 10.30534/ijeter/2020/628102020.

[46] F. Liman, C. Carsten, S. Sufinata, S. Irviantina, and S. Winardi, "Implementation of BiLSTM-SVM Algorithm to Detect Fake News on Text-Based Media," *Jurnal CoreIT*, vol. 9, no. 2, pp. 27–34, Jan. 2024, doi: 10.24014/coreit.v9i2.18982.

[47] I. Evelyn Ezhilarasi and J. C. Clement, "GRU-SVM Based Threat Detection in Cognitive Radio Network," *Sensors*, vol. 23, no. 3, pp. 1–17, Feb. 2023, doi: 10.3390/s23031326.

[48] L. Khan, A. Amjad, K. M. Afaq, and H. T. Chang, "Deep Sentiment Analysis Using CNN-LSTM Architecture of English and Roman Urdu Text Shared in Social Media," *Applied Sciences (Switzerland)*, vol. 12, no. 5, pp. 1–18, Mar. 2022, doi: 10.3390/app12052694.