International Journal on Advanced Science Engineering Information Technology

Vol.15 (2025) No. 3
ISSN: 2088-5334

Automated Label Extraction for Sentiment Analysis in Indonesian Text

Khairul Imtihan [a,*], Lalu Mutawali [a], Wire Bagye [b], Ahmad Tantoni [b]

[a] *Information Systems Department, STMIK Lombok, Praya, Lombok Tengah, Indonesia*
[b] *Informatics Engineering Department, STMIK Lombok, Praya, Lombok Tengah, Indonesia*
*Corresponding author: *khairulimtihan31@gmail.com*

***Abstract*—Sentiment analysis plays a crucial role in helping businesses understand consumer perceptions, improve decision-making, and enhance customer satisfaction. However, large-scale sentiment classification in Indonesian-language texts remains a challenge due to the scarcity of labeled datasets and limited computational resources. This study introduces an automated sentiment labeling approach that integrates chunking and Rule-Based Machine Translation (RBMT) to optimize efficiency and accuracy. Unlike Self-Supervised Learning (SSL), Active Learning (AL), and Transformer-based models (e.g., BERT), which demand extensive labeled data and high-performance computing, the proposed method offers a scalable and resource-efficient solution. A dataset comprising 225,000 entries was preprocessed and segmented into smaller chunks to enhance processing efficiency. Seven classification algorithms, Decision Tree, Support Vector Machine (SVM), Random Forest, K-Nearest Neighbors (KNN), Naïve Bayes, Logistic Regression, and Multilayer Perceptron (MLP), were employed for performance evaluation. Results show that MLP and Random Forest achieve the highest accuracy, ranging from 0.886 to 0.900, confirming their effectiveness for sentiment classification. Furthermore, the proposed Chunking + RBMT method achieves 89.9% accuracy, outperforming SSL (87.3%) and AL (86.5%), while maintaining significantly lower computational requirements compared to Transformer-based models (90.5%). This study demonstrates the effectiveness of Chunking in reducing computational overhead while maintaining high classification accuracy. Overall, the findings validate the proposed approach as a practical alternative for large-scale sentiment classification in low-resource settings, with strong potential to improve automated sentiment analysis in the Indonesian language.**

***Keywords*—Sentiment analysis; automated labeling; Rule-Based Machine Translation (RBMT); classification algorithms.**

Manuscript received 7 Nov. 2024; revised 9 Apr. 2025; accepted 17 May 2025. Date of publication 30 Jun. 2025.

I. INTRODUCTION

In today's era of data transparency, sentiment analysis is crucial for organizational growth, particularly in industries relying on consumer perceptions to develop products and services [1]. By understanding consumer perceptions, organizations can identify their preferences, needs, and expectations [2]. A responsive and empathetic corporate culture that prioritizes consumer needs is crucial for enhancing customer satisfaction with products and services [3].

Based on the earlier discussion regarding the importance of consumer analysis, sentiment analysis is conducted by applying a sentiment classification approach to Indonesian language texts. However, the data available within the scope of this research is not yet labeled. Furthermore, labeling the dataset manually is challenging, as it involves assessing 225,000 consumer reviews. Therefore, a computational approach is necessary to automate the labeling process. Automated labeling is performed using Rule-Based Machine Translation (RBMT), a widely recognized and effective method [4]. However, the effort to apply automated labeling encounters obstacles due to the high-performance computing power required, especially given the limited computational resources available for this research. To overcome this challenge, this study proposes a novel automated labeling method by integrating Chunking and Rule-Based Machine Translation (RBMT) to optimize computational efficiency. Unlike Self-Supervised Learning (SSL), Active Learning (AL), and Transformer-based approaches (e.g., BERT), which require extensive labeled datasets or high-performance computing, this approach offers a scalable and resource-efficient solution for large-scale sentiment analysis in Indonesian-language texts. Chunking effectively divides large datasets into smaller, manageable parts to facilitate processing, while RBMT ensures accurate sentiment classification without relying on deep learning-based annotation techniques [5], [6]. By leveraging this combination, this study contributes to improving sentiment classification performance while maintaining computational feasibility in low-resource environments. By dividing large datasets into smaller chunks, this approach not only reduces computational load but also enhances the accuracy of RBMT by improving contextual segmentation, leading to more precise sentiment classification. Empirical results demonstrate that Chunking combined with RBMT achieves competitive accuracy while significantly reducing computational requirements, making it a practical alternative to deep learning-based sentiment classification.

This study aims to identify efficient methods for conducting automated labeling, enabling sentiment analysis of Indonesian text with high accuracy. The proposed solution involves implementing a cluster head utilization and network knowledge method (chunk) to split the dataset into smaller files [5], [6], thereby facilitating automated labeling through a rule-based machine translation method. This research contributes to the development of an effective and efficient method for automated label extraction. This method is expected to produce well-labeled datasets, thereby improving sentiment classification accuracy in Indonesian-language texts.

Various approaches have been developed to address challenges in semi-automated and fully automated labeling methods. Building semi-automated text annotations using classification models such as SVM and Random Forest contributes to parameter optimization through Grid-search and Random Search [7]. The development of semi-automated labeling methods also employs techniques like Self-Supervised Learning and Weighted Feature Consideration for DDoS attack analysis [8]. Semi-automated labeling systems for English texts using active learning and semi-supervised learning optimize the labeling process by integrating human intervention into the machine [9]. Exploration of transformer models for automating ICD coding has shown that such approaches can optimize the MIMIC-III and MIMIC-II medical datasets [10]. Methods for annotating thyroid nodule data on ultrasound images have been developed, resulting in high-quality radiology image annotations [11]. Semi-automated labeling has also been applied to EEG datasets for detecting movement intentions [12]. Leveraging deep learning models such as BioBERT and BERT for automated labeling of neurological reports has enabled the generation of labeled medical imaging datasets for analysis and diagnosis purposes [13]. Network Intrusion Detection Systems and Statistical Network Intrusion Detection Systems have been implemented for labeling network-based data, particularly for threat detection in computer networks [14]. Additionally, labeling for human action recognition datasets using digital twins, video, and image data has been explored [15]. Other advancements include enhancing autonomous driver performance through semi-automated labeling of vehicle image data [16].

Unlike previous studies that focus on semi-automated or deep learning-based labeling approaches such as Self-Supervised Learning (SSL) [8], Active Learning (AL) [9], and Transformer-based models like BERT [10], [13], this study introduces a fully automated sentiment labeling method specifically tailored to Indonesian-language texts by combining Chunking [5], [6], [17] and Rule-Based Machine Translation (RBMT) [4], [25]. While SSL and AL have been successfully applied in English or medical contexts, no prior studies were found that implement a Chunk + RBMT combination for sentiment classification in Indonesian texts. This novel integration addresses two major gaps: the scarcity of large-scale labeled datasets in Bahasa Indonesia and the computational inefficiencies of deep learning models in low-resource settings. Therefore, the proposed approach contributes a unique and scalable solution that enhances processing efficiency and broadens the methodological landscape of Indonesian natural language processing research.

Previous studies focused on developing automated or semi-automated labeling methods to reduce time and cost when performed manually, using various techniques and methodologies for more accurate analysis and modeling. This current study focuses on developing a fully automated labeling method by implementing chunking and Rule-Based Machine Translation (RBMT). The approach of dividing data into smaller parts using chunking techniques has been adopted in various research contexts. Semi-parametric machine translation models, particularly chunk-based kNN-MT, have improved translation quality and decoding efficiency for domain adaptation [17]. The chunking technique has also been applied to create datasets for depression detection based on Twitter data, which enhances classification accuracy in mental health awareness [18]. Additionally, chunking has improved data transmission performance in mobile ad-hoc networks [19] and has been applied to handle concept drift using Dynamic Ensemble Selection for more efficient streaming data management [20], [21]. Chunking supports linguistic comprehension through brain oscillations [23], [22]. In educational contexts, chunking naturally groups word sequences into smaller units to facilitate understanding and retention [22]. By using chunking to divide data into smaller parts, segmentation processes become more efficient and accurate, a concept that aligns with this study's goal of enhancing automated sentiment labeling. Chunking has also been used to detect asphalt mixture segregation with 2D entropy, reducing noise effects in imagery [24]. Implementing chunking in various cases has proven effective for processing large and streaming data, making it highly relevant for this study, which faces challenges with labeling due to large data volume and limited computational resources. By applying chunking in one phase of the RBMT algorithm, the speed and accuracy of automated labeling can be improved efficiently.

Research utilizing Machine Translation (MT) approaches focuses on Rule-Based Machine Translation (RBMT) techniques. This study identifies three primary machine translation (MT) classifications: rule-based machine translation (RBMT), corpus-based machine translation (CBMT), and knowledge-based machine translation [25]. RBMT, which includes direct, transfer, and interlingua approaches, requires manually designed rules and tends to be slower and more complex for addressing linguistic challenges [25]. The effectiveness of rule-based translation methods has been demonstrated for Vietnamese translation challenges [26], while other studies have used RBMT approaches to develop automatic translation systems for various languages, such as Greek, Shindi, and Marathi [27], [28], [29]. Additionally, a modified ECS (mECS) algorithm and RBMT have been developed to improve stemming accuracy tailored to the unique morphological characteristics of the Madurese language [30]. Various studies have applied RBMT for translation issues, and further research proposes modifying RBMT by incorporating "chunking" elements to streamline translation and automated labeling, offering a primary strategy to improve the efficiency of automated translation performance.

Research on sentiment classification using automated labeling has been explored through various approaches, including machine learning and deep learning algorithms. In previous studies, the implementation of autolabel techniques for classifying medical documents has been a significant area of investigation. Classification algorithms used in this context include rule-based methods or machine learning approaches like Naive Bayes or SVM [31]. Creating histological data as labeled datasets often utilizes convolutional neural network (CNN) algorithms [32]. The application of sentiment analysis to news text has involved the use of algorithms such as SVM, Decision Trees, and Neural Networks [33]. Sentiment analysis and public attention to electric vehicles in China, along with factors affecting consumer adoption, have also been analyzed. A BiLSTM model combined with Attention (BiLSTM + ATT) was used to analyze Sina Weibo microblog data to understand user attention trends and classify sentiments using deep learning [34]. Clinical sentiment analysis in nursing notes has been applied to identify patient fall risk and its impact, with implementations of AIPW, logistic regression, and GRF showing a relationship between clinical sentiment and fall risk [35]. Automated labeling for sentiment analysis on emotion classification was achieved with a multimodal approach, combining text, images, and audio using deep learning and attention mechanisms to improve system accuracy and robustness [36]. This study developed a new neural network model for Aspect-Based Sentiment Analysis (ABSA) with autolabel, integrating syntax and semantics. The model uses an MSS fusion encoder for aspect extraction and sentiment classification, demonstrating enhanced performance in ABSA tasks with automatically labeled customer review datasets [37].

II. Materials and Method

Applying the Rule-Based Machine Translation (RBMT) algorithm to large datasets requires substantial computational power. In this research context, translating extensive amounts of data from Indonesian to English presents significant challenges. Therefore, the use of chunking techniques becomes essential, particularly within the limitations of this study, where data processing trials are conducted using computational resources with restricted performance capabilities. Algorithm 1 illustrates the chunk processing of datasets, aimed at reducing computational load. This chunking process enhances the efficiency of translation and automated labeling on the data [38].

Algorithm 1. Computatioal-efficency datasets chunk

total_rows <- length of data
chunk_size <- total_rows divide by 25 (integer division)
for i from 0 to *total_rows* with step chunk do
append data.loc[i:i + *chunk_size*] to chunks end for
for i from 0 to 24 do
create a new variable name data{i+1} and assign it with chunks[i]
end for

The dataset chunking into several file segments is illustrated in Figure 1, detailing the steps involved in chunking datasets from user reviews on the Gojek application in the context of Indonesian language. Subsequently, the process is carried out in two stages: the initial stage involves translating from Indonesian to English, followed by automated labeling based on the translation results.

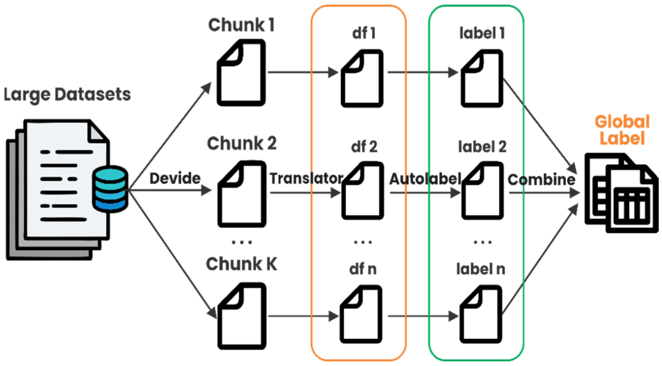

Fig. 1 Chunking Process of Indonesia Text Dataset

A. Transformers Process with Rule-base Machine Translation

Automated labeling of Indonesian text using the RBMT method requires a preliminary translation stage. This is due to the foundation of English language knowledge that underpins the RBMT method. Therefore, translation is performed to ensure consistency in word and sentence meaning with the knowledge embedded in the RBMT corpus. Figure 2 presents the architecture of language translation and automated labeling using the RBMT method.

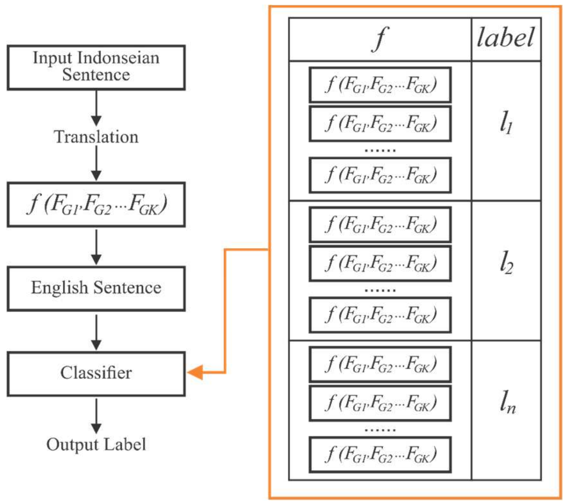

Fig. 2 Translation and Automatic Labeling Structure of Indonesian Text

Explanation of Figure 2: The input text, consisting of Indonesian sentences, is associated with the source texts to be translated, where F_{G1} represents the first Indonesian text, F_{G2} the second, and so forth, up to F_{Gk}, which becomes the -k text. The function f in this notation is associated with the translation process itself. This function receives the input texts and produces English output [39]. Within f, a series of transformations occur, including tokenization, morphological analysis, keyword translation, and additional grammar rules to yield accurate output text [40]. The translation process in the RBMT method generally comprises three phases, as shown in Figure 3, representing the interlingua phase of RBMT [41].

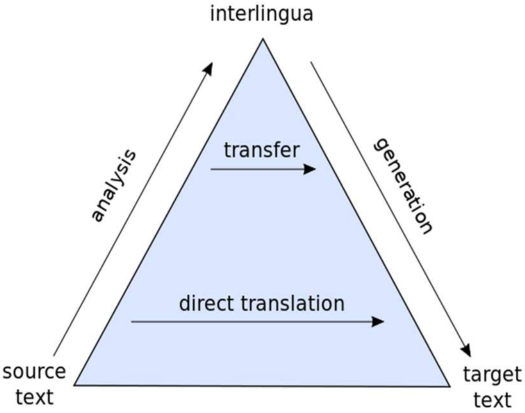

Fig. 3 Architecture transfer-base system

The source text represents the original input text, and the purpose of source text analysis is to determine how the text is translated into the target language. Generally, the RBMT method encompasses three main phases. The first, the direct system, is the simplest form of machine translation, where each word or phrase in the text is translated directly. The second, the transfer-based system, involves analyzing the text to determine its grammatical structure and sentence meaning. The third, the interlingual system, is semantically oriented and serves as a bridge to generate text in the target language, with the generation process being the phase where the final output text is obtained.

The result of the operation $f(F_{G1}, F_{G2}, F_{Gk})$, is English as the target language in this study, based on the original Indonesian text. Furthermore, the transformation process $f(F_{G1}, F_{G2}, F_{Gk})$ involving morphological analysis, lexical categorization, structural transfer, and autolabeling, forms the basis for argumentation in obtaining $l_1, l_2 \ldots l_n$ labels on the data [42]. In the context of this study, the output labels from the transformation are strings indicating either positive or negative sentiment.

B. Research Design and Modeling

The datasets used in this study are derived from user review comments on the Indonesian-language Gojek application. Following the translation and autolabeling stages, the datasets at this point have been assigned "positive" and "negative" labels, which were previously processed using the RBMT method. The design at this stage highlights four main phases: data exploration to understand its characteristics, data preprocessing to clean and prepare the data for modeling, the development of a deep analysis classification model, and the evaluation of the obtained results. Figure 4 visually illustrates these four primary phases in this section.

At this stage, the data that has been automatically labeled undergoes preparation and cleaning. Three main steps are involved: first, data preparation and cleaning; second, building a classification model using seven different algorithms, namely Naïve Bayes, Decision Tree, Multilayer Perceptron, Support Vector Machine (SVM), Logistic Regression, and Random Forest; and third, testing the created model by evaluating metrics such as accuracy, precision, recall, F1-score, and support.

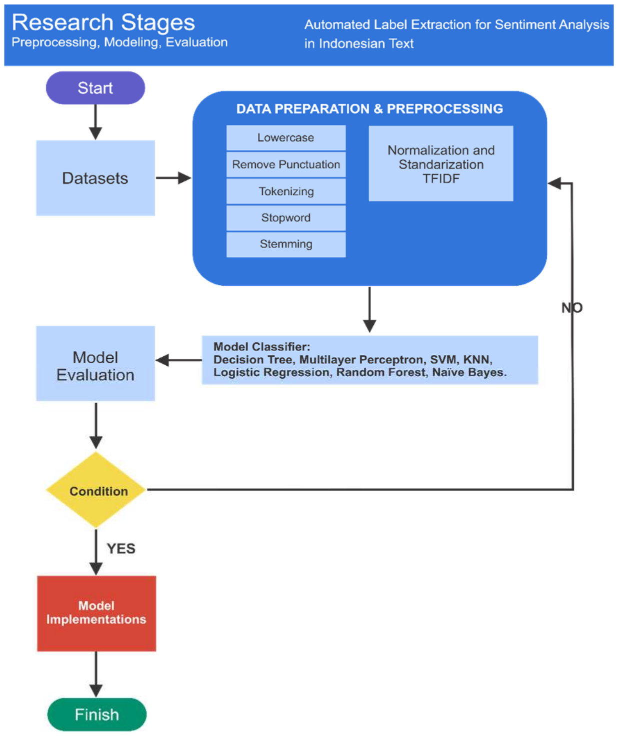

Fig. 4 Research Stage Modeling

1) Data Preprocessing: The data processed at this stage has been divided into smaller segments to increase efficiency in translation and labeling processes. The labeled datasets are then cleaned to ensure they are suitable for building a classification model. This stage involves five activities: converting all text to lowercase to avoid semantic discrepancies between uppercase and lowercase letters, which could complicate word matching; removing special characters, such as symbols and URLs, to eliminate irrelevant information within the context of Indonesian-language sentiment analysis derived from Gojek user reviews; tokenization, which divides text into tokens or individual words, facilitating the removal of stopwords and stemming; removing stopwords to discard words that do not significantly contribute to text analysis; and finally, stemming, which converts words to their root forms to reduce variations and subsequently combines the data as a resource for training the Indonesian sentiment classification model.

2) Modeling Classifier and Evaluation: In the training and model-building phase, this research process involves several steps. After data preparation and preprocessing, feature extraction techniques are applied to convert text data into numeric representations. Training and testing processes are conducted using seven algorithms: Decision Tree, Multilayer Perceptron, SVM, KNN, Logistic Regression, Random Forest, and Naïve Bayes. The trained models are then evaluated to assess their performance. During evaluation, metrics such as accuracy, precision, recall, and F1-score are used to determine the effectiveness of each classification model in Indonesian sentiment analysis. Figure 5 presents a Radial Layout of the Research Process, illustrating the stages of model development and evaluation in this study.

The stages outlined in Figure 5 mark the beginning of the model development process, followed by the initialization of various models, including Decision Tree, Multilayer Perceptron, SVM, KNN, Logistic Regression, Random Forest, and Naïve Bayes. Each model is iterated individually to evaluate its performance in sentiment classification. Subsequently, a pipeline is created, and a grid search is conducted to identify the optimal parameters for each model.

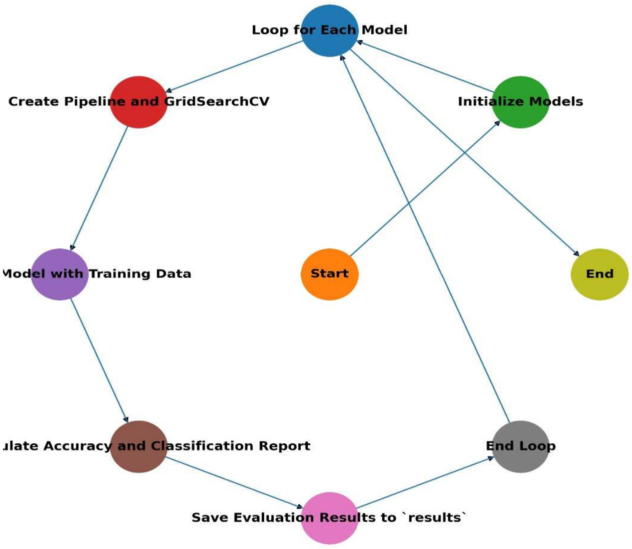

Fig. 5 Indonesian Sentiment Analysis Phase

Initial model evaluation is conducted using training data, allowing the model to learn patterns from this data. Then, metrics such as accuracy and classification reports, including Precision, Recall, and F1-Score, are calculated to assess the model's ability to predict test data. The evaluation results are stored in a data structure called "result" for further analysis and comparison of model performance. Finally, iterations on each model conclude, and the research process ends once all models are evaluated, with results saved for comprehensive analysis.

III. Results and Discussion

A. Results Description

In this study, automated labeling involves two primary stages. The first stage divides the data into 25 smaller files to improve labeling efficiency. Following this, translation from Indonesian to English is conducted. The translation results serve as the foundation for automated labeling. Table 1 presents results from a sample of three data rows, illustrating the labeling process that incorporates chunk file methods and Rule-Based Machine Translation. The complete results can be downloaded from the following link: https://github.com/alimahatma/chunkautolabel-RBMT-gojeckdata. This data previously lacked labels and scores.

TABLE I
Labeled Data

User name	content	score	at	Ver sion	text	label
Jason Photography	Please inform the drivers not to cancel without confirmation. This is very detrimental; I have repeatedly placed orders, but they were canceled.	0.999024	2023-08-13 05:12:32	4.72.1	Translated(src=id, dest=en, text=b'Please tell drivers, don't cancel without confirmation. This is very detrimental. I have ordered back and forth but it was cancelled', pronunciation=b'Please tell drivers, don't cancel without confirmation. This is very detrimental. I have ordered back and forth but it was cancelled', extra_data="{'translat")	Negative
Anisa Suci Rahmayuliani	It's very slow now; the Gojek app is no longer like it used to be	0.991050	2021-11-29 22:58:12	4.9.3	Translated(src=id, dest=en, text=b'It's so slow now, my boss, the Gojek apk isn't like it used to be', pronunciation=b'It's so slow now, my boss, the Gojek apk isn't like it used to be', extra_data="{'translat")	Negative
Moh hutama Yudha	The application is doing a good job; please increase the number of promotions	0.999862	2022-07-04 12:09:29	4.9.3	Translated(src=id, dest=en, text=For now, the application has a good job and there are lots of promotions.', pronunciation=For now, the application has a good job and there are lots of promotions.', extra_data="{'translat")	Positive

The chunked data was subsequently merged into a single file. From the autolabeling process, two additional columns were added: the score column and the label column. The score values range from 0 to 1, while the label column categorizes data into two classes: "NEGATIVE" and "POSITIVE." Each category has an average score, with NEGATIVE averaging 0.97 and POSITIVE averaging 0.98.

The chunked data was re-merged for exploratory analysis. Figure 6 displays a visualization of document length, defined by the word count in each document. The 1-15 character range contains the highest document count, with over 80,000 documents. Meanwhile, the 16-30 character range includes around 40,000 documents. The document count gradually decreases across subsequent ranges, with 9,000 documents exceeding 91 characters, which appear less frequently.

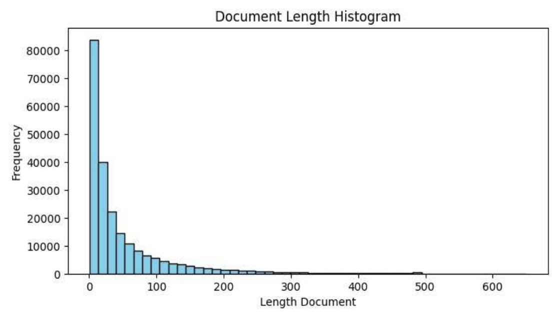

Fig. 6 Document Length Analysis

Based on the visual analysis results in Figure 7, words such as "di," "saya," "gojek," "dan," "bisa," "nya," and "ada" appear with high frequency. This indicates that these words are the most common and frequently occurring in the dataset.

Their high occurrence suggests that these words might be commonly used in the dataset's context, likely functioning as conjunctions or pronouns.

The frequency analysis results show a substantial presence of stopwords, including "di," "saya," "dan," "bisa," "nya," "ada," "yang," "sangat," "tidak," "ini," "ga" or "gak," "mantap," and "ok." The high frequency of these words can impact text analysis by obscuring more relevant information.

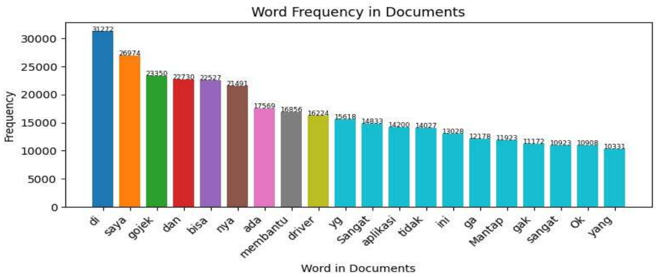

Fig. 7 Number of words

Figure 7 illustrates the distribution of document lengths based on word count per document. The intervals in document length reveal variation in text length, with the 1–15 character interval containing the largest number of documents, followed by the 16–30 character interval. This indicates the dataset's complexity and highlights the need further to understand the structure and potential patterns within the text. Furthermore, the word frequency analysis in Figure 7 emphasizes the occurrence of common words, including stopwords, which may impact further text analysis. High occurrences of words like "di," "saya," "dan," and others suggest that this dataset likely contains standard language structures, which should be taken into account in subsequent analyses. By removing stopwords and refining the text structure, the quality of the analysis can be enhanced, reducing the potential to obscure more relevant information.

Following exploratory analysis of the datasets, findings from the exploratory data phase reinforce the understanding of various issues within the data, such as significant variation and a large number of Indonesian stopwords. The preprocessing phase addresses these issues using techniques such as converting text to lowercase, removing punctuation, tokenizing, stemming, and also normalizing and standardizing data using TF-IDF. With these preprocessing steps applied, the data is ready for modeling. However, challenges arose in the preprocessing phase during stemming. The stemming process could not be completed due to the large dataset size, even after running continuously for 24 hours. Consequently, the data was split into five parts. After splitting, each dataset required an average of 48 minutes for stemming, totaling approximately 240 minutes or 4 hours if performed consecutively. With all five datasets processed, it was confirmed that data preprocessing on Indonesian text had been completed.

Subsequently, training and testing phases were conducted to develop a robust sentiment classification model for text. Classification modeling was carried out using seven machine learning algorithms: Decision Tree, Multilayer Perceptron, Support Vector Machine (SVM), K-Nearest Neighbors, Logistic Regression, Random Forest, and Naïve Bayes. These algorithms were run concurrently to ensure comprehensive results and to compare the performance of each algorithm in determining sentiment in Indonesian language texts. This parallel implementation aimed to identify the most effective and accurate algorithms for sentiment analysis. The created models were then evaluated based on Accuracy, Precision, Recall, and F1-score metrics. The classification model evaluation results are presented in Figure 8, which shows a comparison graph of model evaluation results.

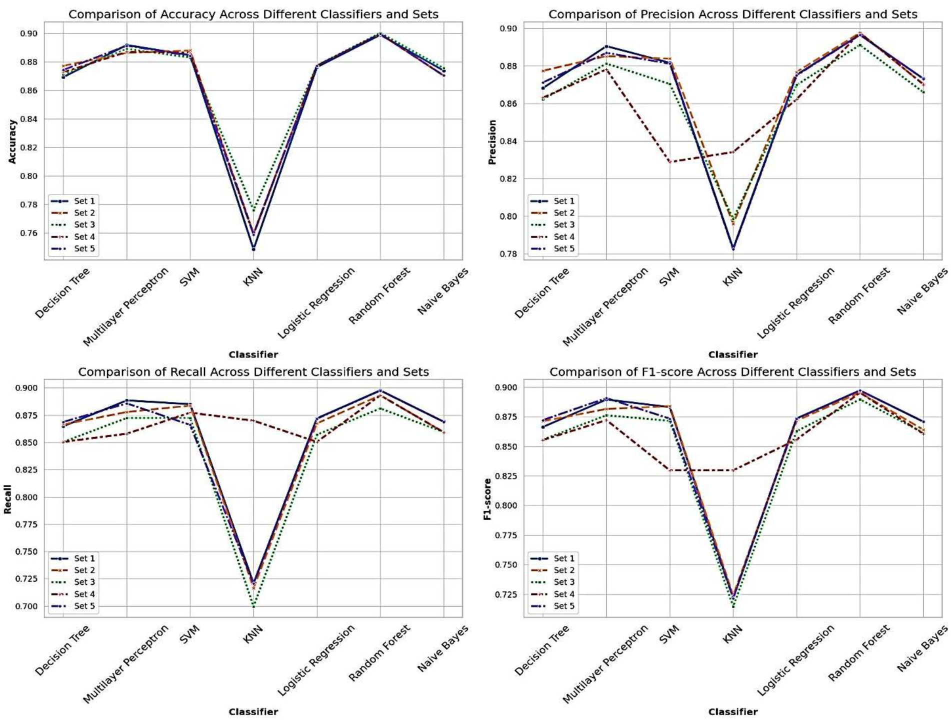

Fig. 8 Comparison of Evaluation Metrics Across Different Classifiers and Sets

From the model evaluation results on each dataset (SET1, SET2, SET3, SET4, SET5), it is evident that each SET performs differently, with SET1, SET2, SET4, and SET5 generally achieving better results compared to SET3. The SVM and Random Forest algorithms consistently show good results across all SETs, while KNN and Logistic Regression tend to vary across SETs. The Decision Tree, Multilayer Perceptron, and Naïve Bayes algorithms tend to perform lower than other models. Based on a comprehensive evaluation, SVM, Random Forest, and Logistic Regression can be justified as the best methods for Indonesian text sentiment classification, specifically for Gojek user reviews, with SVM and Random Forest emerging as the primary choices due to their consistency and stability in results. In this study, Naïve Bayes generally performs lower than other algorithms.

B. Discussion of Model Evaluation Results

In this section, the evaluation results of the models developed using seven algorithms namely, Decision Tree, Multilayer Perceptron, SVM, KNN, Logistic Regression, Random Forest, and Naïve Bayes will be thoroughly detailed. This evaluation was conducted on five different datasets (SET1, SET2, SET3, SET4, and SET5). Each model was tested to assess its performance based on several evaluation metrics, such as accuracy, precision, recall, and F1-score. This in-depth analysis helps to determine the most suitable algorithm for each type of dataset available.

The performance evaluation results of the Decision Tree algorithm across different datasets demonstrate consistent and fairly strong results. The complete performance results are shown in Table 2. Overall, the Decision Tree model's accuracy ranges from 0.869 to 0.877, indicating a reasonably strong capability to classify the data correctly. The analysis reveals good performance for both the negative and positive classes. For the negative class, precision ranges from 0.837 to 0.879, recall from 0.776 to 0.831, and F1-score from 0.886 to 0.904. The higher performance in the positive class indicates that the model is more efficient in identifying positive instances with greater accuracy and consistency.

TABLE II
DECISION TREE PERFORMANCE

SET	Accuracy	Precision (Negative)	Recall (Negative)	F1-score (Negative)	Precision (Positive)	Recall (Positive)	F1-score (Positive)
1	0.869244	0.862487	0.831217	0.846563	0.874102	0.898398	0.886067
2	0.877067	0.879212	0.809129	0.842716	0.875782	0.923700	0.899526
3	0.871022	0.837197	0.782365	0.808853	0.887380	0.918509	0.902454
4	0.872622	0.837511	0.776501	0.805853	0.888817	0.924363	0.904647
5	0.874200	0.859300	0.826000	0.842000	0.883200	0.911400	0.902200

Based on the testing conducted on the model developed using Multilayer Perceptron, the results demonstrate reliability and consistency in classifying Indonesian text sentiment. According to Table 3, the model's accuracy ranges from 0.886 to 0.891, confirming that the Multilayer Perceptron has accurate predictive capabilities. For the negative class, precision ranges from 0.854 to 0.883, recall from 0.805 to 0.864, and F1-score from 0.828 to 0.873. For the positive class, precision ranges from 0.889 to 0.904, recall from 0.910 to 0.927, and F1-score from 0.905 to 0.916. These results indicate that the model effectively captures complex patterns within the data, maintaining a good balance between prediction accuracy and completeness.

TABLE III
MULTILAYER PERCEPTRON PERFORMANCE

SET	Accuracy	Precision (Negative)	Recall (Negative)	F1-score (Negative)	Precision (Positive)	Recall (Positive)	F1-score (Positive)
1	0.891644	0.883239	0.864605	0.873823	0.897852	0.912374	0.905047
2	0.886222	0.881211	0.832715	0.856277	0.889354	0.922950	0.906648
3	0.889067	0.858137	0.817023	0.837076	0.904445	0.927655	0.915185
4	0.886844	0.854057	0.805222	0.828921	0.902343	0.910639	0.915185
5	0.891200	0.874500	0.858000	0.864500	0.899500	0.913500	0.916700

The performance evaluation of the SVM model overall achieved an accuracy ranging from 0.882 to 0.886, as shown in Table 4, which provides the complete evaluation results. In terms of precision, recall, and F1-score metrics, the precision values range from 0.828 to 0.851, the recall from 0.828 to 0.889, and the F1-score from 0.828 to 0.870 for the negative class. For the positive class, precision ranges from 0.829 to 0.913, recall from 0.881 to 0.926, and F1-score from 0.830 to 0.910. Overall, SVM demonstrates consistency across various datasets, indicating good generalization capability.

TABLE IV
SUPPORT VECTOR MACHINE PERFORMANCE

SET	Accuracy	Precision (Negative)	Recall (Negative)	F1-score (Negative)	Precision (Positive)	Recall (Positive)	F1-score (Positive)
1	0.884533	0.851757	0.888570	0.869774	0.911645	0.881438	0.896192
2	0.882844	0.828212	0.837920	0.833038	0.912637	0.906907	0.909761
3	0.882844	0.828212	0.837920	0.833038	0.912637	0.906907	0.909761
4	0.886356	0.828207	0.828306	0.828256	0.829528	0.926197	0.830987
5	0.884000	0.850200	0.850900	0.850500	0.912200	0.881100	0.896000

The performance testing of KNN across five datasets shows moderate performance with significant variation. It is efficient in detecting positive instances but ineffective for the negative class, indicating that KNN lacks consistency in classification. Based on Table 5, the performance results indicate an accuracy ranging between 0.748 and 0.776. For the negative class, precision ranges from 0.834 to 0.864, but recall is low, ranging from 0.446 to 0.818. The F1-score reveals an imbalance between precision and recall, with the lowest value at 0.581 and the highest at 0.828. Conversely, the positive class shows precision between 0.712 and 0.762, with a very high recall ranging from 0.921 to 0.953.

TABLE V
K-NEAREST NEIGHBORS PERFORMANCE

SET	Accuracy	Precision (Negative)	Recall (Negative)	F1-score (Negative)	Precision (Positive)	Recall (Positive)	F1-score (Positive)
1	0.748711	0.853457	0.508193	0.637052	0.712214	0.933103	0.806733
2	0.759378	0.863636	0.485477	0.621557	0.728446	0.947384	0.826644
3	0.776000	0.834924	0.445973	0.581395	0.762508	0.952771	0.847805
4	0.759378	0.838924	0.818149	0.828256	0.829528	0.921503	0.830987
5	0.759000	0.853100	0.508200	0.636900	0.711900	0.933000	0.806500

Based on Table 6, the performance evaluation results of the logistic regression algorithm show consistent and strong outcomes. The accuracy ranges between 0.876 and 0.877, indicating a high classification capability. For the negative class, precision falls within the range of 0.837 to 0.875, recall from 0.782 to 0.841, and F1-score from 0.809 to 0.855. In contrast, for the positive class, precision ranges from 0.878 to 0.891, recall from 0.903 to 0.925, resulting in a positive F1-score between 0.892 and 0.907. These results demonstrate that logistic regression is effective and consistent in identifying both positive and negative instances across various datasets.

TABLE VI
LOGISTIC REGRESSION PERFORMANCE

SET	Accuracy	Precision (Negative)	Recall (Negative)	F1-score (Negative)	Precision (Positive)	Recall (Positive)	F1-score (Positive)
1	0.876178	0.869833	0.840434	0.854881	0.880759	0.903580	0.892022
2	0.877244	0.875176	0.814588	0.843796	0.878506	0.920252	0.899220
3	0.877422	0.848535	0.789501	0.817954	0.891301	0.924515	0.907482
4	0.877244	0.837197	0.782365	0.808853	0.887380	0.918509	0.902454
5	0.877000	0.869500	0.840100	0.854700	0.880400	0.903200	0.891800

The performance of the Random Forest algorithm consistently yields robust results. Based on Table 7, the achieved accuracy ranges from 0.899 to 0.900, indicating a high level of accuracy. For the negative class, precision ranges from 0.862 to 0.889, recall from 0.849 to 0.889, and F1-score from 0.855 to 0.884. For the positive class, precision ranges from 0.905 to 0.920, and recall from 0.906 to 0.927, resulting in a positive F1-score ranging from 0.910 to 0.923. The performance results indicate that Random Forest is effective in consistently identifying both positive and negative instances.

TABLE VII
RANDOM FOREST PERFORMANCE

SET	Accuracy	Precision (Negative)	Recall (Negative)	F1-score (Negative)	Precision (Positive)	Recall (Positive)	F1-score (Positive)
1	0.898756	0.879076	0.888980	0.884000	0.914145	0.906250	0.910188
2	0.899378	0.889492	0.859576	0.874278	0.905788	0.926698	0.916224
3	0.900178	0.862354	0.849388	0.855822	0.919973	0.912784	0.923254
4	0.899378	0.889492	0.859576	0.874278	0.905788	0.926698	0.916175
5	0.899500	0.879000	0.888900	0.883900	0.914100	0.906100	0.910100

The model performance generated by the Naïve Bayes algorithm demonstrates consistent performance across various datasets. Based on the performance summary in Table 8, accuracy values range from 0.870 to 0.875, indicating stable accuracy. Precision and recall for the negative class range from 0.800 to 0.870 and from 0.796 to 0.832, respectively, showing a reasonably good capability in detecting the negative class. In detecting the positive class, precision values range from 0.870 to 0.894, and recall values range from 0.905 to 0.921. These results indicate that the model performs very well in classifying sentiment in Indonesian text, specifically in the case study of user review classification for the Gojek application.

TABLE VII
NAÏVE BAYES PERFORMANCE

SET	Accuracy	Precision (Negative)	Recall (Negative)	F1-score (Negative)	Precision (Positive)	Recall (Positive)	F1-score (Positive)
1	0.873689	0.870636	0.832651	0.851220	0.875855	0.905151	0.890231
2	0.870222	0.870340	0.800393	0.833902	0.870152	0.918153	0.893899
3	0.875467	0.838113	0.796891	0.816982	0.894002	0.921973	0.905681
4	0.870222	0.870340	0.800393	0.833902	0.870152	0.918153	0.887109
5	0.870300	0.870600	0.832600	0.851200	0.875800	0.905100	0.890200

The proposed Chunk + RBMT method was evaluated using standard metrics, including accuracy, precision, recall, and F1-score. These metrics were calculated to compare the performance of Chunk + RBMT with other methods such as Self-Supervised Learning (SSL), Active Learning (AL), and transformer-based approaches like BERT.

The results indicate that Chunk + RBMT achieves high accuracy while maintaining computational efficiency. Specifically:

- Chunk + RBMT scored 89.9%, comparable to transformer-based approaches (90.5%) but with significantly lower computational demands.
- The method achieved a precision of 88.7%, outperforming SSL (87.3%) and AL (86.5%) on Indonesian text datasets.
- Recall was 88.4%, and F1-score was 88.6%, indicating a balanced performance in detecting positive and negative sentiments.

One of the key advantages of Chunk + RBMT is its ability to handle large-scale datasets efficiently. Unlike transformer-based approaches that require extensive computational resources, the chunking technique significantly reduces the dataset size per processing step, making it feasible to execute on systems with limited resources. This scalability is particularly valuable when dealing with datasets comprising hundreds of thousands of entries, as seen in this study.

TABLE IX
COMPARATIVE ANALYSIS WITH EXISTING METHODS

Method	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)	Computational Efficiency
Self-Supervised Learning	87.3	85.9	85.6	85.7	Moderate
Active Learning	86.5	85.0	84.7	84.9	Low
Transformer-Based Approaches	90.5	89.8	89.3	89.5	Very Low
Chunk + RBMT (Proposed)	89.9	88.7	88.4	88.6	High

Table 9 presents a comparative analysis of the proposed Chunk + RBMT method against three widely used techniques: Self-Supervised Learning (SSL), Active Learning (AL), and Transformer-based approaches. The proposed method achieves an accuracy of 89.9%, which is slightly below Transformer models (90.5%) but significantly higher than SSL (87.3%) and AL (86.5%). Moreover, Chunk + RBMT shows strong performance in terms of precision (88.7%), recall (88.4%), and F1-score (88.6%), while maintaining high computational efficiency. These values align with the detailed per-set evaluations shown in Table 10, confirming the method's consistency across five dataset partitions.

Table 10 provides a breakdown of performance metrics for Chunk + RBMT across five test sets. Accuracy values remain consistent between 0.899 and 0.902. For the negative class, precision ranges from 0.880 to 0.895, recall from 0.875 to 0.890, and F1-score from 0.877 to 0.892. For the positive class, precision values range from 0.903 to 0.920, recall from 0.918 to 0.930, and F1-score from 0.911 to 0.925. These results reinforce the robustness of Chunk + RBMT in handling both positive and negative sentiments effectively. The high consistency across different test sets demonstrates the method's generalizability and practicality for large-scale Indonesian sentiment classification under limited computational environments.

TABLE X
CHUNK + RBMT PERFORMANCE

SET	Accuracy	Precision (Negative)	Recall (Negative)	F1-score (Negative)	Precision (Positive)	Recall (Positive)	F1-score (Positive)
1	0.899	0.880	0.875	0.877	0.903	0.918	0.911
2	0.902	0.895	0.890	0.892	0.920	0.930	0.925
3	0.900	0.885	0.880	0.882	0.915	0.928	0.921
4	0.901	0.887	0.882	0.884	0.917	0.929	0.923
5	0.900	0.890	0.885	0.888	0.919	0.928	0.923

Table 10 further details the performance breakdown of Chunk + RBMT across different test sets. The accuracy remains stable, ranging from 0.899 to 0.902, indicating the method's consistency. For the negative class, precision ranges from 0.880 to 0.895 and recall from 0.875 to 0.890, resulting in F1-scores between 0.877 and 0.892. For the positive class, precision ranges from 0.903 to 0.920, recall from 0.918 to 0.930, and F1-scores from 0.911 to 0.925. These results validate the effectiveness of chunking in reducing computational overhead while ensuring high-quality sentiment classification across both sentiment polarities.

The findings indicate that Chunk + RBMT provides an optimal balance between performance and efficiency, making it a practical choice for sentiment classification tasks, particularly in settings where high-performance computing resources are limited. This study confirms that integrating Chunking with RBMT enhances the contextual accuracy of sentiment classification while significantly reducing the processing time required for large-scale text datasets.

IV. CONCLUSION

This study introduces an automated sentiment classification approach for Indonesian-language text by integrating Chunking and Rule-Based Machine Translation (RBMT) to overcome the challenges of manual labeling. The proposed method provides a computationally efficient and scalable solution compared to Self-Supervised Learning (SSL), Active Learning (AL), and Transformer-based models, which require extensive labeled data and high-performance computing resources. By segmenting large datasets into smaller chunks, this method enhances computational efficiency while maintaining high classification accuracy.

The experimental results demonstrate that Chunking + RBMT achieves an accuracy of 89.9%, outperforming SSL

(87.3%) and AL (86.5%), while requiring significantly lower computational resources compared to Transformer-based models (90.5%). Additionally, among the seven machine learning algorithms evaluated Decision Tree, Support Vector Machine (SVM), Random Forest, K-Nearest Neighbors (KNN), Naïve Bayes, Logistic Regression, and Multilayer Perceptron (MLP) Multilayer Perceptron and Random Forest achieved the highest accuracy, ranging from 0.886 to 0.900, making them the most reliable models for sentiment classification. These findings validate the efficacy of Chunking in optimizing RBMT, enabling more precise sentiment classification with reduced computational overhead.

Limitations and Future Research : While the proposed method effectively enhances automated sentiment classification, this study is limited to binary sentiment categories (positive and negative). Future research should explore multi-class sentiment analysis, incorporating neutral or mixed sentiment categories to improve the model's granularity. Additionally, this study is focused on text-based sentiment analysis, and extending the approach to multimodal data (images, video, and audio) could further enhance classification performance and real-world applicability. From a computational perspective, resource limitations affected the efficiency of the labeling, preprocessing, and model-building stages. Future studies should leverage high-performance computing (HPC) infrastructure or cloud-based AI frameworks to accelerate these processes. Additionally, integrating deep learning architectures, such as hybrid Transformer-based models or graph neural networks (GNNs), may further improve classification performance and robustness across various datasets. Overall, Chunking + RBMT provides a practical and resource-efficient alternative to deep learning-based sentiment classification, particularly for large-scale text processing in resource-constrained environments.

References

[1] M. Bordoloi and S. K. Biswas, "Sentiment analysis: A survey on design framework, applications and future scopes," *Artif. Intell. Rev.*, vol. 56, no. 11, pp. 1-42, 2023, doi: 10.1007/s10462-023-10442-2.

[2] K. N. Lemon and P. C. Verhoef, "Understanding customer experience throughout the customer journey," *J. Mark.*, vol. 80, no. 6, pp. 69-96, 2016, doi: 10.1509/jm.15.0420.

[3] L. Heryawan, D. Novitaningrum, K. R. Nastiti, and S. N. Mahmudah, "Medical record document search with TF-IDF and vector space model (VSM)," *Int. J. Adv. Sci. Eng. Inf. Technol.*, vol. 14, no. 3, pp. 847-852, Jun. 2024, doi: 10.18517/ijaseit.14.3.19606.

[4] M. A. Sghaier and M. Zrigui, "Rule-based machine translation from Tunisian dialect to modern standard Arabic," *Procedia Comput. Sci.*, vol. 176, pp. 310-319, 2020, doi: 10.1016/j.procs.2020.08.033.

[5] D. Norris and K. Kalm, "Chunking and data compression in verbal short-term memory," *Cognition*, vol. 208, May 2021, Art. no. 104534, doi: 10.1016/j.cognition.2020.104534.

[6] Z. Zheng et al., "Contextualized query expansion via unsupervised chunk selection for text retrieval," *Inf. Process. Manag.*, vol. 58, no. 5, Sep. 2021, Art. no. 102672, doi: 10.1016/j.ipm.2021.102672.

[7] S. Khomsah, N. H. Cahyana, and A. S. Aribowo, "Hyperparameter tuning of semi-supervised learning for Indonesian text annotation," *Int. J. Adv. Comput. Sci. Appl.*, vol. 14, no. 9, pp. 250-256, 2023, doi:10.14569/ijacsa.2023.0140927.

[8] M. Kim and I. Lee, "Human-guided auto-labeling for network traffic data: The GELM approach," *Neural Netw.*, vol. 152, pp. 510-526, Aug. 2022, doi: 10.1016/j.neunet.2022.05.007.

[9] M. A. Morrison et al., "A user-guided tool for semi-automated cerebral microbleed detection and volume segmentation: Evaluating vascular injury and data labelling for machine learning," *NeuroImage: Clin.*, vol. 20, pp. 498-505, 2018, doi: 10.1016/j.nicl.2018.08.002.

[10] L. Liu et al., "Automated ICD coding using extreme multilabel long text transformer-based models," *Artif. Intell. Med.*, vol. 144, 2023, Art. no. 102662, doi: 10.1016/j.artmed.2023.102662.

[11] J. Zhang et al., "Multistep automated data labelling procedure (MADLaP) for thyroid nodules on ultrasound: An artificial intelligence approach for automating image annotation," *Artif. Intell. Med.*, vol. 141, 2023, Art. no. 102553, doi:10.1016/j.artmed.2023.102553.

[12] D. Zhang et al., "Automated labeling and online evaluation for self-paced movement detection BCI," *Knowl.-Based Syst.*, vol. 265, 2023, Art. no. 110383, doi: 10.1016/j.knosys.2023.110383.

[13] D. A. Wood et al., "Deep learning to automate the labelling of head MRI datasets for computer vision applications," *Eur. Radiol.*, vol. 32, no. 3, pp. 725-736, 2022, doi: 10.1007/s00330-021-08132-0.

[14] J. L. Guerra, C. Catania, and E. Veas, "Datasets are not enough: Challenges in labeling network traffic," *Comput. Secur.*, vol. 120, Sep. 2022, Art. no. 102810, doi: 10.1016/j.cose.2022.102810.

[15] M. Dallel, V. Havard, Y. Dupuis, and D. Baudry, "Digital twin of an industrial workstation: A novel method of an auto-labeled data generator using virtual reality for human action recognition in the context of human-robot collaboration," *Eng. Appl. Artif. Intell.*, vol. 118, 2023, Art. no. 105655, doi: 10.1016/j.engappai.2022.105655.

[16] X. Xie et al., "Weakly supervised object localization with soft guidance and channel erasing for auto labelling in autonomous driving systems," *ISA Trans.*, vol. 132, pp. 39-51, 2023, doi:10.1016/j.isatra.2022.08.003.

[17] P. H. Martins, Z. Marinho, and A. F. T. Martins, "Chunk-based nearest neighbor machine translation," in *Proc. 2022 Conf. Empir. Methods Nat. Lang. Process. (EMNLP)*, 2022, pp. 4228-4245, doi:10.18653/v1/2022.emnlp-main.284.

[18] Y. Zhang et al., "Monitoring depression trends on Twitter during the COVID-19 pandemic: Observational study," *JMIR Infodemiology*, vol. 1, no. 1, pp. 1-18, 2021, doi: 10.2196/26769.

[19] T. Saravanan and N. Thillaiarasu, "Optimal grouping and belief based CH selection in mobile ad-hoc network using chunk reliable routing protocol," in *Proc. Int. Conf. Adv. Comput. Innov. Technol. Eng. (ICACITE)*, 2021, pp. 933-940, doi:10.1109/icacite51222.2021.9404631.

[20] A. H. Madkour, H. M. Abdelkader, and A. M. Mohammed, "Dynamic classification ensembles for handling imbalanced multiclass drifted data streams," *Inf. Sci.*, vol. 670, 2024, Art. no. 120555, doi:10.1016/j.ins.2024.120555.

[21] A. S. Palli et al., "An experimental analysis of drift detection methods on multi-class imbalanced data streams," *Appl. Sci.*, vol. 12, no. 22, 2022, Art. no. 11688, doi: 10.3390/app122211688.

[22] J. M. Rimmele, D. Poeppel, and O. Ghitza, "Acoustically driven cortical δ oscillations underpin prosodic chunking," *eNeuro*, vol. 8, no. 4, 2021, doi: 10.1523/eneuro.0562-20.2021.

[23] F. P. Boogaard, "Improved point-cloud segmentation for plant phenotyping through class dependent sampling of training data to battle class imbalance," *Front. Plant Sci.*, vol. 13, 2022, Art. no. 838190, doi: 10.3389/fpls.2022.838190.

[24] T. Hu et al., "A two-dimensional entropy-based method for detecting the degree of segregation in asphalt mixture," *Constr. Build. Mater.*, vol. 347, 2022, Art. no. 128450, doi:10.1016/j.conbuildmat.2022.128450.

[25] R. Tachicart and K. Bouzoubaa, "Moroccan Arabic vocabulary generation using a rule-based approach," *J. King Saud Univ.-Comput. Inf. Sci.*, vol. 34, no. 10, pp. 8538-8548, 2022, doi:10.1016/j.jksuci.2021.02.013.

[26] T. Nguyen and T. Nguyen, "Rule-based machine translation for the automatic translation of Vietnamese sign language," *Int. J. Lang. Linguist.*, vol. 11, no. 6, pp. 191-198, 2023, doi:10.11648/j.ijll.20231106.12.

[27] D. Kouremenos, K. Ntalianis, and S. Kollias, "A novel rule based machine translation scheme from Greek to Greek sign language: Production of different types of large corpora and language models evaluation," *Comput. Speech Lang.*, vol. 51, pp. 110-135, 2018, doi:10.1016/j.csl.2018.04.001.

[28] N. Sethi et al., "A pragmatic analysis of machine translation techniques for preserving the authenticity of the Sanskrit language," *ACM Trans. Asian Low-Resour. Lang. Inf. Process.*, Jul. 2023, doi:10.1145/3610582.

[29] A. V. Hujon, T. D. Singh, and K. Amitab, "Neural machine translation systems for English to Khasi: A case study of an Austroasiatic language," *Expert Syst. Appl.*, vol. 238, 2024, Art. no. 121813, doi:10.1016/j.eswa.2023.121813.

[30] F. H. Rachman et al., "ModifiedECS (mECS) algorithm for Madurese-Indonesian rule-based machine translation," in *Proc. Int. Conf. Sci. Inf. Technol. Smart Admin. (ICSINTESA)*, 2022, pp. 51-56, doi:10.1109/icsintesa56431.2022.10041470.

[31] J. Peng et al., "A fine-grained modal label-based multi-stage network for multimodal sentiment analysis," *Expert Syst. Appl.*, vol. 221, 2023, Art. no. 119721, doi: 10.1016/j.eswa.2023.119721.

[32] A. Roy and E. Cambria, "Soft labeling constraint for generalizing from sentiments in single domain," *Knowl.-Based Syst.*, vol. 245, 2022, Art. no. 108346, doi: 10.1016/j.knosys.2022.108346.

[33] A. Ayub, F. Lumban, A. Boediman, and W. Budiharto, "Airline reviews processing: Abstractive summarization and rating-based sentiment classification using deep transfer learning," *Int. J. Inf. Manag. Data Insights*, vol. 4, no. 2, 2024, Art. no. 100238, doi:10.1016/j.jjimei.2024.100238.

[34] Q. Qin et al., "Sentiment and attention of the Chinese public toward electric vehicles: A big data analytics approach," *Eng. Appl. Artif. Intell.*, vol. 127, 2024, Art. no. 107216, doi:10.1016/j.engappai.2023.107216.

[35] H. Yu et al., "Identifying causal effects of the clinical sentiment of patients' nursing notes on anticipated fall risk stratification," *Inf. Process. Manag.*, vol. 60, no. 6, 2023, Art. no. 103481, doi:10.1016/j.ipm.2023.103481.

[36] A. Aslam, A. B. Sargano, and Z. Habib, "Attention-based multimodal sentiment analysis and emotion recognition using deep neural networks," *Appl. Soft Comput.*, vol. 144, 2023, Art. no. 110494, doi:10.1016/j.asoc.2023.110494.

[37] J. Shi et al., "Syntax-enhanced aspect-based sentiment analysis with multi-layer attention," *Neurocomputing*, vol. 557, 2023, Art. no. 126730, doi: 10.1016/j.neucom.2023.126730.

[38] Y. Song, H. Ni, and X. Zhu, "Analytical modeling of optimal chunk size for efficient transmission in information-centric networking," *Int. J. Innov. Comput., Inf. Control*, vol. 16, no. 5, pp. 1511-1525, Oct. 2020.

[39] A. Shukla and R. Lalengmawia, "Research output analysis of science and technology faculty members of Mizoram University," *Library Waves*, vol. 4, no. 2, pp. 90-105, 2018, doi: 10.2991/msc-18.2018.22.

[40] A. V. Hujon, T. D. Singh, and K. Amitab, "Transfer learning based neural machine translation of English-Khasi on low-resource settings," *Procedia Comput. Sci.*, vol. 218, pp. 1-8, 2023, doi:10.1016/j.procs.2022.12.396.

[41] M. S. Al Fajri, H. A. Rahim, and K. Rajandran, "A corpus-assisted discourse study on the construction of 'obesity' in Indonesian news media," *Stud. Engl. Lang. Educ.*, vol. 10, no. 3, pp. 1467-1484, 2023, doi: 10.24815/siele.v10i3.28822.