International Journal of Advances in Intelligent Informatics
Vol. 7, No. 1, March 2021, pp. 12-25

ISSN 2442-6571

Cross-domain sentiment analysis model on Indonesian YouTube comment

Agus Sasmito Aribowo [a,1,*], Halizah Basiron [b,2], Noor Fazilla Abd Yusof [b,3], Siti Khomsah [c,4]

[a] Informatics Department, Universitas Pembangunan Nasional "Veteran" Yogyakarta, Jl. SWK 104 Sleman, 55283 Yogyakarta, Indonesia
[b] Centre for Advanced Computing Technology, Universiti Teknikal Malaysia Melaka, 76100 Melaka, Malaysia
[c] Data Science Department, Insitut Teknologi Telkom Purwokerto, Jl. D.I.Panjaitan No. 128, 53147 Purwokerto, Indonesia
[1] sasmito.skom@upnyk.ac.id; [2] halizah@utem.edu.my; [3] elle@utem.edu.my; [4] siti@ittp-pwt.ac.id
* corresponding author

ARTICLE INFO

Article history
Received September 17, 2020
Revised March 22, 2021
Accepted March 23, 2021
Available online March 31, 2021

Keywords
Cross-domain
Sentiment analysis
Tree-based ensemble ML
Stop word
Slang word

ABSTRACT

A cross-domain sentiment analysis (CDSA) study in the Indonesian language and tree-based ensemble machine learning is quite interesting. CDSA is useful to support the labeling process of cross-domain sentiment and reduce any dependence on the experts; however, the mechanism in the opinion unstructured by stop word, language expressions, and Indonesian slang words is unidentified yet. This study aimed to obtain the best model of CDSA for the opinion in Indonesia language that commonly is full of stop words and slang words in the Indonesian dialect. This study was purposely to observe the benefits of the stop words cleaning and slang words conversion in CDSA in the Indonesian language form. It was also to find out which machine learning method is suitable for this model. This study started by crawling five datasets of the comments on YouTube from 5 different domains. The dataset was copied into two groups: the dataset group without any process of stop word cleaning and slang word conversion and the dataset group to stop word cleaning and slang word conversion. CDSA model was built for each dataset group and then tested using two types of tree-based ensemble machine learning, i.e., Random Forest (RF) and Extra Tree (ET) classifier, and tested using three types of non-ensemble machine learning, including Naïve Bayes (NB), SVM, and Decision Tree (DT) as the comparison. Then, It can be suggested that the accuracy of CDSA in Indonesia Language increased if it still removed the stop words and converted the slang words. The best classifier model was built using tree-based ensemble machine learning, particularly ET, as in this study, the ET model could achieve the highest accuracy by 91.19%. This model is expected to be the CDSA technique alternative in the Indonesian language.

1. Introduction

Sentiment analysis will classify a group of sentences whether their polarity is positive or negative. The opinion texts are obtained from the microblog posting made by the users on social media. The users reveal their opinion about a topic from formal to informal language. The cross-domain sentiment analysis (CDSA) refers to an application of domain adaptation in which the classification is trained in one domain (called as source domain) to classify other domains (called as target domain). CDSA helps to implement the sentiment information learned in the source domain to a certain target domain. There is a feature transfer from a source domain to a target domain [1].

CDSA in the English language has been continuously learned, such as CDSA using the data from Amazon (books, DVD, electronics, or kitchen) [2] and the data from the Internet Movie Databases

https://doi.org/10.26555/ijain.v7i1.554

(IMDB) [3]. Other studies used sentiment of 140 corpora and dataset of semEval [4]. Other studies related to CDSA were the drug data taken from two sites, i.e., drugs.com and druglib.com. The research results showed that the transfer learning approach could be used to exploit the similarity in all domains [5]. In recent years, a similar study about CDSA was implementing the effect of CDSA using two datasets of the tweet and one review dataset. This study used three classifications, namely Naive Bayes, Multinomial Naive Bayes, and Support Vector Machine, through 18 experiments by varying the training dataset and classifier model to evaluate the model being built. The best-resulted model used the tweet for the classification of Multinomial Naive Bayes [4]. Another study used SVM to test the model by using the feature of word embedding and the combination of the word embedding feature and raw features [6]. Further study was the cross-domain using SVM model for two experiments: single source to multiple target domains and multiple sources to a single target (MSST). The best performance was obtained using the regulation of MSST with an accuracy of 85.05% [7]. Other studies included the increase of preprocessing, feature extraction, and the use of the ensemble approach. The experiment used the dataset of Amazon review benchmarks, and the model tested as CDSA was CRD-SentEnse with and without a noun, CRD-SentEnse-var with and without a noun, and semi-controlled CRD-SentEnse with and without nouns like learning machine in the ensemble model, SVM, and Logistic Regression. The feature extraction used FastText word embeddings aggregated with a mean (in The CRD-SentEnse Approach Input) and FastText word embeddings aggregated by TFIDF (in The CRD-SentEnse-var Approach Input) [8]. Another study also used the review of Amazon books and proposed the multi-layer convolutional neural network (CNN)-based learning transfer method, and it resulted in a good performance. To solve the problem that words that occur in the train (source) domain might not appear in the test (target) domain, It can use sentiment-sensitive thesaurus from source domains (labeled data) and both source and target domains (unlabeled data) [9]. There are paper explores the effectiveness of several feature vectors in CDSA [10]. Studies above commonly used the available labeled dataset with the research focus to develop the model and select the machine learning method. The widely used machine learning methods are Naïve Bayes and SVM. There are various text preprocessing methods, such as tokenizing, stemming, POS tagging, and lemmatizing. In general, all models resulted in better accuracy compared to ones from the previous studies. However, the studies above are referred to as CDSA in the English language. Thus, we only obtained knowledge about the preprocessing method, feature extraction, machine learning, and cross-domain strategy.

As revealed from the site of databoks.katadata.co.id. 88% of netizens in Indonesia accessed YouTube in 2020 [11]. The use of the Indonesian language in social media was placed in the third rank in the world [12]. YouTube is a social media-based video with the highest number of users. The Analysis Sentiment from the comments on Tube Indonesia is interesting to be studied as there are a number of particular language features in terms of dialect, grammar, and non-formal use in the opinions on YouTube. The study of CDSA in the Indonesian language is still found rare. However, there have been some researches on sentiment analysis (SA) in the Indonesian language, including the use of word embedding with CNN (convolutional neural network) method to sentiment analysis (SA) in Indonesia language with an accuracy of only 76.2% [13], SA in comments of YouTube using SVM with the accuracy of 84% [14]. SA using the tree decision, obtained the accuracy by 76,06% with the emoticons and slang words dictionaries [15]. The study on the use of SVM for the YouTube comments for the cyberbullying classification had an accuracy by only 79.412% [16]. Other research using SVM with linear kernel function had the accuracy by only 62.76% [17]. Another study developed the combination of K-Nearest Neighbor and Levenshtein Distance. This combination method obtained the accuracy by 65.625% [18]. A similar study about Multinomial Naïve Bayes has given the mean F1-Score of 91,4% using the preprocessing and feature selection, and the combination of Naive Bayes and SVM resulted in the accuracy by 91% [19], but it has not been tested in the data from some domains. Similar research used Naive Bayes and reached the accuracy by only 81% onYouTube Movie Trailer [20]. Our last study is about the use of comments on YouTube to categorize the fanaticism of sentiment using the tree-based machine learning and it obtained the accuracy by 91.8% using the Random Forest [21]. This study is also not done in the data from various domains yet. Based on the literature review above, most previous researches on Sentiment Analysis in Indonesia used Naive Bayes and SVM as machine learning.

SVM commonly provides high accuracy [22]. The preprocessing methods used are tokenizing, stemming, clear stop word, lemmatization, and POS tagging. Our previous research explained that removing stopwords and converting slang words could increase accuracy by 3.5% [23].

The challenges in CDSA include the differences in the meaning of words across domains. This problem occurs as there are some words in one domain – not in other domains. Another problem is related to the existence of slang words. In this research dataset, it has been found out that approximately 60% of opinions contained slang words, while it is only from 27% to 28% as the stopwords. The existence of stopwords and slang words in the Indonesian language needs to be managed in CDSA. Annotation is another problem. The sentiment analysis is supervised machine learning requiring the annotated data training. Annotation is the polarity of the positive and negative sentiment, and giving the annotation or label in the data text is something simple. Giving the annotation must concern the meaning of each word in a sentence, either in standard language or in a non-formal one. If there are positive and negative elements in a sentence, it is necessary to determine its dominant polarity. Even for objectivity, it requires a number of experts to give the polarity level in the same texts. Thus, the annotation process is costly [24], [25], and time-consuming [26]. The research to develop the machine giving the automatic annotation in the text opinion in a different domain is certainly interesting to be studied considering that, if it is successful, it can be more efficient and can reduce the dependence on the experts. The questions formulated in this study include how to build the CDSA model that can give high accuracy by answering any existing challenges. Another question is if CDSA in the Indonesian language can be used for the labeling process to substitute the experts.

This study aimed to obtain the best model of CDSA for the opinion in Indonesia language that commonly is full with stopwords and slang words in the Indonesian dialect. The main objective of this research is to observe the performance of CDSA by using the public opinions from the comments in YouTube videos in the form of the Indonesian language. It also aimed to develop the model of negative or positive sentiment labeling in Indonesian society's opinion using the cross-domain concept. Another objective is to evaluate the process of language feature transfer from the source domain to the target domain, such as the benefits in removing the stop words and converting the slang words in the preprocessing phase in CDSA.

The novelty of this research is a new model of CDSA for the opinions in the form of Indonesia language with a target accuracy of more than 90%. The new model will combine some preprocessing methods to search for the best machine learning. We did 200 experiments using 40 models of CDSA and 2 tree-based ensemble machine learnings such as Random Forest (RF) and Extra Tree (ET), and 3 algorithms of machine learning such as Support Vector Machines (SVM), Multinomial Naıve Bayes (NB), and Decision Tree (DT) as the comparison. Five different datasets were used in this experiment in which in each of our experiments, we tested the benefit of clear stopword and converted the slang word di CDSA. The dataset in this research was obtained from the comments in YouTube videos in the Indonesian language from 5 different domains, as presented in Table 1.

This research is structured as follows. Part II presents the methodology we used, including the information about the dataset, classification machine, and experiments. Part III presents the results of the experiments and statistical analysis. Part IV finally presents the conclusion and any possibilities for the development of further research.

2. Method

Our research has followed the standard steps of the CDSA process. It began with a preliminary process, i.e., data crawling and data labeling. Data crawling refers to the process of obtaining the opinion dataset from YouTube. Data labeling is the process of annotating positive-negative sentiments by experts. Fig. 1 shows the CDSA process steps.

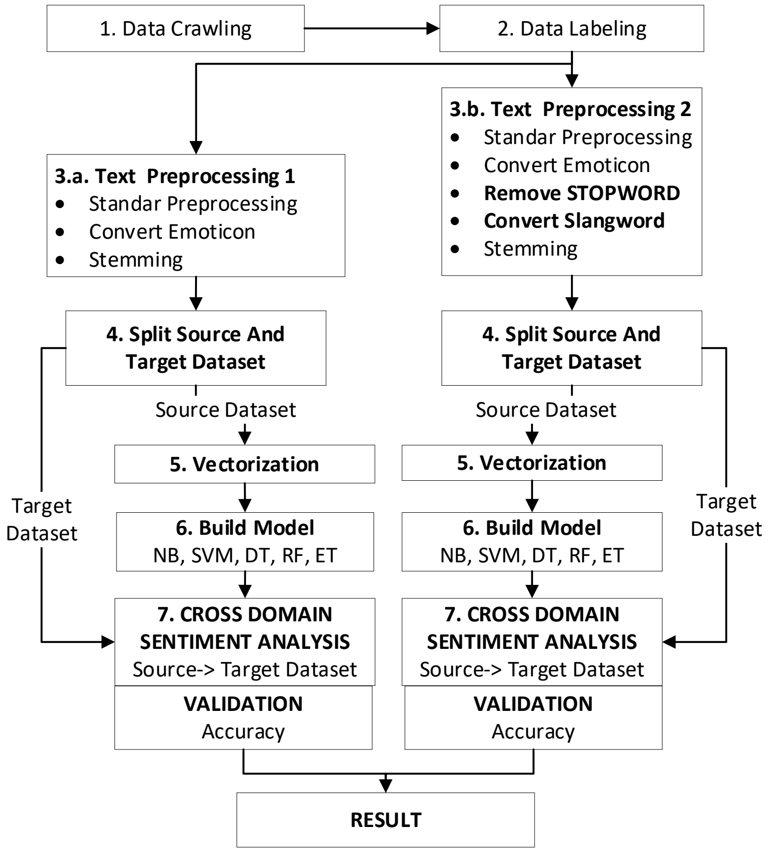


Fig. 1. Research Steps

2.1. Data Crawling (Step 1)

The dataset for this study was obtained from Indonesian-language YouTube comments from 5 different domains. Comments were crawled from 1 February 2019 to 31 May 2019. The comment selection process started by searching YouTube videos based on the keywords of the five topics.

As shown in Table 1, Video comments were selected from the official YouTube channel of national news office or television station such as TV One News, MNC TV, CNN Indonesia, Kompas TV, and IDN Times. The video with the most comments and the most viewers were selected. Comments were downloaded with our application and stored in 5 different datasets. Table 1 presents the results of the obtained dataset.

Table 1. Dataset Crawled From YouTube Comment

Dataset Number	Domain	YouTube Channel	Opinion Number
1	Energy and Infrastructure	MNC TV, CNN Indonesia	2271
2	Ideology and Governance	Kompas TV, IDN Times	3113
3	Education and Health	Kompas TV, IDN Times	6087
4	Law and Human Rights	Kompas TV, IDN Times	7406
5	Economy and Social Welfare	TV One News, CNN Indonesia	19839
		TOTAL	38716

2.2. Data Labeling (Step 2)

In supervised learning, the label of datasets is given by the experts [28]. In this study the opinions in all datasets were manually labeled by several experienced annotators. Annotators set the polarity labels into positive and negative sentiments by observing the context of the whole comment sentences. The summary of labeling results is presented in Table 2.

Table 2. Information Datasets

Dataset Number	Number of Comments	Sentiment Polarity	
		Positive	*Negative*
1	2217	1351	866
2	3113	1878	1235
3	6087	3868	2219
4	7406	4345	3061
5	19839	10987	8852

The annotation result showed an imbalance in the number of positive and negative comments. Imbalanced data could affect the classification result. Oversampling or under-sampling can be used to balance the number of positive and negative samples to be equal [29]. The ensemble algorithm can be operated in the imbalance dataset, such as the Random Forest algorithm and the Extremely Randomize Tree (Extra Tree). Naive Bayes, SVM, and Decision Trees were also be tested in classifying the imbalance data.

2.3. Text Preprocessing (Step 3)

All datasets entered the preprocessing stage. This stage began with the data cleaning process, such as removing URLs, numbers, single characters, changing to lowercase, converting emoticons, removing non-alphabetic characters, and tokenizing. It was then followed by duplicating the dataset into two groups. In contrast to the second group, the first group dataset was processed to remove any stop words and remove slang words. Stemming was the last process by removing any affixes in each word. Preprocesssing tested on the first group dataset consisted of the following steps:

1) **Stop Words Removal.** If a word was found in our stop words dictionary, it would be deleted then. Deleting the subject or object was done as well. The examples of subjects were the names of political figures or names of institutions or names of objects having no sentimental elements such as "Jokowi", "Prabowo", "Maruf Amin", "Sandiaga", "01", "02", "cebong", "kampret". These words were found in all positive and negative sentences; hence they were not the feature of sentiment [30].

2) **Slang word Conversion.** If a character was found to be repeated in sequence, it would be changed into a single character. Slang words usually contain many repeated characters, for instance the word "okeee" that would be shortened to be "oke" (okay). "Siiiipp" became "sip," and "maantaaap" became "mantap" (steady). The next process was to remove any words containing only one character, such as a word consisting of one character, i.e., y, or t. Then the slang word was converted into the standard KBBI words like "elo" converted into "kamu" (you), "guwe" converted into "saya" (me), "pengen" converted into "ingin" (want), "laen" converted into "lain" (other), "jgn" converted into "jangan" (do not). In this conversion process, we compiled a slang word dictionary containing 5721 slang words in Indonesian.

2.4. Splitting the Source and Target Datasets (Step 4)

There were 5 datasets used in the CDSA, namely dataset 1, dataset 2, dataset 3, dataset 4, and dataset 5. CDSA process used one dataset to all other datasets, for example, dataset 1 used for the source dataset, and another dataset as the target dataset. In the next step, dataset 2 was used as the source dataset for another dataset, and the next step was dataset 3, 4, and 5.

2.5. Vectorization (Step 5)

The preprocessing phase changes the unstructured source dataset into sa emi-structured dataset, enabling the pattern to be extracted more easily. The input of vectorization is a semi-structured source dataset. The dominant feature from the source dataset in positive and negative sentiment groups was carried out using the Count-Vectorizer (CV) method. CV was used to convert a collection of sentences to a vector of terms counts. Every word in a sentiment group would be counted [31]. The dominant word in one type of sentiment became a member of the sentiment group.

2.6. Building the Model (Step 6)

Models were built by using a source dataset and several machine learning. There were five machine learning used to build a model. Three basic machine learning (NB, SVM, and DT) were chosen based upon the results of a literature review of previous studies, and two tree-based ensemble machine learning methods (RF and ET) were selected because it was believed to provide high accuracy as found in previous studies [21], [25].

Naive Bayes (NB), used in previous studies [19], [20], [21]. Naive Bayes is a method for modeling sentiment analysis that can produce high accuracy. The Bayes' rule is presented in (1).

$$P(c|X) = \frac{P(c|X)\,P(c)}{P(X)} \tag{1}$$

P(c|X) is a probability of value c to be true if value X is true. P(X|c) is a probability of value X is true if value c is true. P(c) is a probability value c is correct, and P(X) is a probability of value X to be true. The Bayes theorem is based on the statistics of probability and cost generated from the classification decision. NB is one of the simple implementations of the Bayes theorem.

Support Vector Machine (SVM). SVM is a popular technique for classification. This research used the linear kernel. This technique attempts to find the most optimum separation function (hyperplane) to separate any opinion data from different classes (positive and negative), or in this case, called binary classes. The illustration of a hyperplane in SVM can be seen in Fig. 2. SVM has the separation function separating Class 1 and Class 2 effectively and divided by a clear gap as wide as possible [3]. The question is how SVM finds the optimum hyperplane. The trick is to find the outermost data in the two classes on the border and find the optimum hyperplane considering the outer data.

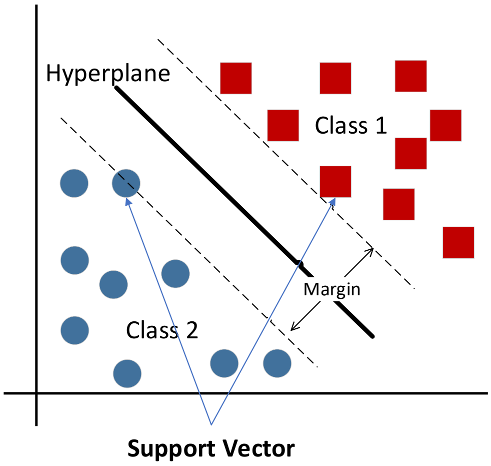


Fig. 2. Support Vector Machine

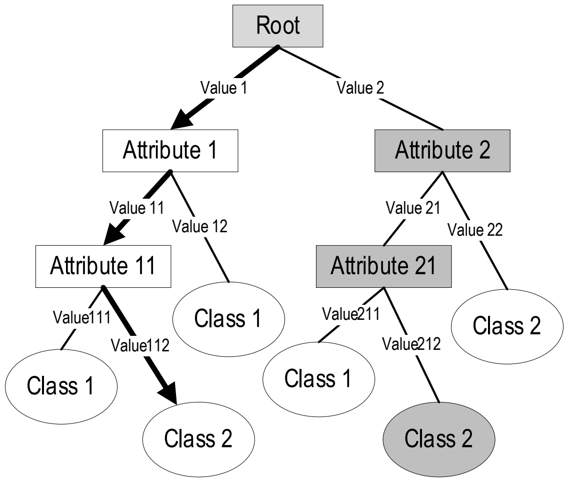
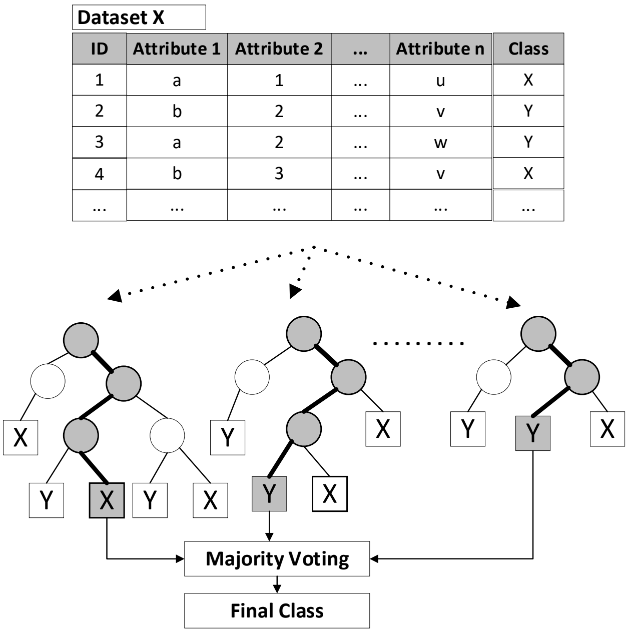


Fig. 3. Decision Tree

Decision Tree (DT). Decision Tree is an algorithm that will arrange the training data features into a tree structure for classification or decision making. In its structure, branches are the classification question, the edge is the answer of questions, and the leaf is the classification class. The first step in building a Decision Tree is to choose the main attribute as root, create a branch for each value, and then divide the data into the branches created using the entropy formula [32]. The process is repeated in all branches until all data in the branch become homogeneous class, as shown in Fig. 3. Here the arrows are the values of the attribute, and the leaves are the classes.

Random Forest (RF) is the development of a Decision Tree. It builds many trees in the same steps as a Decision Tree and splits nodes using the best split among a random subset of features selected at every node. RF reduces the risk of overfitting by building multiple trees and bootstrapping. An example of the process of making many trees is shown in Fig. 4. Each tree provides the classification results, and the final classification is the most classes produced from these trees (majority class).

Extra Tree (ET) Classifier. Extra Tree resembles a Random Forest in which the main process is to divide the dataset into clusters to build many trees and split nodes randomly. However, with two differences when compared to Random Forest that does not use bootstrap (sample without replacement), nodes are divided using random splits, not by the best split. In Extra Trees, randomness is not derived from bootstrap data but comes from the random separation of all observations. The final classification is the majority class resulted from these trees. Extra Trees is named for Extremely Randomized Trees.



Dataset X

ID	Attribute 1	Attribute 2	...	Attribute n	Class
1	a	1	...	u	X
2	b	2	...	v	Y
3	a	2	...	w	Y
4	b	3	...	v	X
...	...	...	...	...	...

Fig. 4. Random Forest

2.7. CDSA (Step 7)

Source dataset was used as training data, and machine learning method was used to create models. The target dataset here was data testing. The prediction results of the data testing were compared with the actual labels. This process was repeated. In the first round, dataset 1 was used for the source dataset for other datasets. Then in the second round, dataset 2 was used as the source dataset for other datasets, up to dataset 5, so that 20 CDSA models were formed. Each model would be tested for 5 types of machine learning, and then there was a cross-domain process 100 times for preprocessing using clear stop words and converting slang words, and 100 times for preprocessing without any clear stop words and without removing any slang words. Accuracy here was calculated per experiment. The accuracy test was based on the prediction of the number of positive sentiment target datasets classified into the positive sentiment class and the prediction of the negative sentiment target datasets classified into the negative sentiment class. The result of the model was being tested for its performance using the confusion matrix (Table 3). The confusion matrix compared the predicted results and the actual conditions of the machine learning model results. The annotator gave the actual class, and the model's classification result was the predicted class.

Table 3. Confusion Matrix

		Actual Class	
		Positive	*Negative*
Predicted Class	*Positive*	True Positive (TP)	False Positive (FP)
	Negative	False Negative (FN)	True Negative (TN)

From the confusion matrix, an accuracy value will be obtained. Accuracy is a measure of how many actual class values are the same as the predicted class, the number of true positive (TP), and true-negative (TN). Accuracy is calculated using (2).

$$Accuracy = \frac{(TP+TN)}{(TP+FP+TN+FN)} \tag{2}$$

3. Results and Discussion

The preprocessing stage showed some interesting facts, such as 60% of all comments in all datasets containing slang words. The unstructured data caused by stop words and slang words were found more compared to the structured ones. Compared to all the words in each dataset, the percentage of stop words was 26.58% to 28.02% of all words in each dataset. The number of slang words was 13.85% to 15.26% of all words in the dataset. Thus the experiment without involving a stop word could reduce the word features by approximately 26.58% to 28.02%. The experiment of converting slang words to formal words could add 13.85% to 15.26% word features. Facts about the dataset in the study are presented in Table 4.

Table 4. The Existence of Stop Words and Slang words in Datasets

No Dataset	Number of							
	Comments	*Comments With Slang word*		*Word*	*Stop word*		*Slang word*	
		Σ	*%*		*Σ*	*%*	*Σ*	*%*
1	2217	1340	60.44%	29482	8119	27.54%	4486	15.22%
2	3113	1898	60.97%	56258	15508	27.57%	7794	13.85%
3	6087	3641	59.82%	92832	24678	26.58%	14332	15.44%
4	7406	4492	60.65%	127116	35624	28.02%	18952	14.91%
5	19839	12838	64.71%	348482	97615	28.01%	53187	15.26%

This study examined two types of preprocessing, and five types of machine learning on 5 datasets. The CDSA experimental scenario was to measure the accuracy of sentiment analysis in all models and all machine learnings. We considered the benefits of removing stop words and converting slang words to be analyzed for higher accuracy. Hence, we conducted two experiments in which the first experiment was to make CDSA models with five machine learning datasets without any preprocessing stop word removal and slang word conversion. The second experiment was to make CDSA models using five machine learning datasets with clear stop words and converting the slang words.

3.1. The first experiment, calculating the accuracy of the CDSA model without stop word removal and slang word conversion.

This experiment was to determine the accuracy of CDSA from the source dataset to the target dataset without removing any stop words and converting any slang words. CDSA was done by making dataset 1 as the source dataset and datasets 2, 3, 4, and 5 as the target dataset. Then, the next step was dataset 2 as a source and dataset 1, 3, 4, and 5 as a target. The next steps were until all datasets have been tested as the source and target. The experiment was repeated for five types of machine learning and two kinds of preprocessing. Then, 20 models must be made. Each model was experimented with 5 kinds of machine learning (NB, SVM, DT, RF, and ET). The cross-domain process occurred in as many as 100 experiments. The results of the experiment are depicted in Table 5. The results of the first experiment in Table 5 showed that the best accuracy in order from small to large was initiated by Naïve Bayes with an accuracy of 77.06%, followed by a decision tree with an accuracy of 86.92% and SVM of 87.67%. Random Forest had 88.46% accuracy and 89.64% Extra Tree. The CDSA process reaching the accuracy target was the CDSA from datasets 4 and 5 to all datasets, if using Extra Tree. If using Random Forest and Decision Tree, only CDSA from source dataset 5 could reach the accuracy target.

Table 5. Test Results For The CDSA Model Without Remove Stopword and Slangword

No Model	Source Dataset → Target Dataset	Machine Learning				
		Non-Ensemble			*Ensemble*	
		NB	*SVM*	*DT*	*RF*	*ET*
1	1→2	73.50%	77.48%	80.89%	82.11%	83.75%
2	1→3	74.54%	77.76%	79.40%	82.54%	83.62%
3	1→4	74.91%	77.25%	80.70%	82.96%	84.57%
4	1→5	72.42%	75.83%	80.18%	81.98%	83.92%
5	2→1	68.74%	66.35%	80.38%	80.42%	81.91%
6	2→3	73.21%	69.92%	81.26%	81.60%	83.29%
7	2→4	72.47%	68.13%	81.60%	83.54%	84.49%
8	2→5	69.29%	64.49%	79.00%	80.83%	82.86%
9	3→1	71.00%	78.39%	80.11%	81.37%	83.45%
10	3→2	73.27%	79.70%	80.08%	80.95%	83.36%
11	3→4	75.07%	81.15%	81.00%	83.35%	85.35%
12	3→5	72.68%	79.60%	79.15%	81.81%	84.45%
13	4→1	71.00%	80.56%	82.00%	83.13%	85.34%
14	4→2	73.92%	82.11%	83.65%	84.07%	86.22%
15	4→3	74.67%	82.32%	82.65%	83.98%	85.15%
16	4→5	73.33%	83.17%	83.96%	85.70%	87.01%
17	5→1	71.81%	84.39%	85.70%	85.25%	86.42%
18	5→2	74.91%	84.36%	86.09%	85.22%	86.70%
19	5→3	76.38%	85.89%	85.44%	86.35%	87.45%
20	5→4	77.06%	87.67%	86.92%	88.46%	89.64%

a.

If using SVM, it was only CDSA from source dataset 5 to dataset 3 and dataset 4 reached the target accuracy. In conclusion, to obtain a good CDSA, it must use Extra Tree and dataset 5 as the source dataset.

Of all the models that reached the target, it was only 9 of the Extra Tree models out of 20 models (45%), Random Forest only reached 5 out of 20 models (25%), the Decision Tree only reached 4 of 20 models (20%), and SVM only reached 2 of 20 models (10%). Thus, further research is needed to increase accuracy by completing the preprocessing stage.

3.2. The second experiment: calculating the accuracy of the CDSA model with stop word removal and slang word conversion.

This experiment was to determine the accuracy of CDSA from the source dataset to the target dataset after both datasets were processed by removing any stop words and converting any slang words. As in the previous stage, CDSA was done by making dataset 1 as the source dataset and dataset 2 , 3, 4, and 5 as the target dataset. Then the next step was dataset 2 as a source and dataset 1, 3, 4, and 5 as a target. The next steps were conducted until all datasets have been tested as source and target. The experiment was repeated for five types of machine learning and two types of preprocessing. There were 20 models that should be generated, each of which experimented with 5 kinds of machine learning (NB, SVM, DT, RF, and ET). The cross-domain process occurred through 100 experiments, and the results of the experiment can be seen in Table 6. The results of experiment 2 in Table 6 showed that the best accuracy was sorted from small to large started with Naïve Bayes with a value of 79.54%, followed by SVM 89.35% and Decision Tree 90.18%. Then, it was continued with the Random Forest with 91.06% and Extra Tree with 91.91%. All types of CDSA reached the target accuracy when using Extra Tree. Random forest achieved only 17 out of 20 models (85%), 11 out of 20 models (55%) were achieved in Decision Tree, and 5 out of 20 (25%) in SVM. It can be concluded that to obtain a good CDSA, it must use the Extra Tree. It then can be concluded that removing stop words and converting slang words could improve the accuracy.

Table 6. Test Results For The CDSA Model With Remove Stop word and Slang word

No Model	Source Dataset → Target Dataset	Machine Learning				
		Non-Ensemble			*Ensemble*	
		NB	*SVM*	*DT*	*RF*	*ET*
1	1→2	74.01%	79.57%	82.46%	85.32%	87.09%
2	1→3	76.34%	79.99%	82.98%	84.95%	86.20%
3	1→4	75.24%	79.83%	84.11%	86.32%	87.56%
4	1→5	73.04%	78.62%	83.08%	85.07%	87.19%
5	2→1	69.33%	78.48%	83.81%	83.18%	85.43%
6	2→3	73.32%	81.62%	84.74%	85.26%	86.86%
7	2→4	72.20%	81.70%	84.39%	86.02%	87.24%
8	2→5	69.52%	80.19%	83.82%	84.71%	85.93%
9	3→1	71.81%	81.37%	83.76%	85.16%	87.19%
10	3→2	74.17%	82.94%	85.26%	86.86%	87.79%
11	3→4	75.86%	84.39%	85.69%	87.71%	88.90%
12	3→5	73.52%	83.19%	85.06%	86.73%	88.26%
13	4→1	72.80%	82.77%	86.74%	85.93%	87.05%
14	4→2	75.10%	84.81%	86.09%	87.47%	89.27%
15	4→3	76.52%	84.44%	85.81%	86.64%	87.65%
16	4→5	74.92%	86.26%	87.89%	88.60%	89.77%
17	5→1	74.79%	86.92%	89.13%	88.90%	89.58%
18	5→2	77.06%	86.67%	89.59%	88.37%	90.30%
19	5→3	78.87%	87.53%	89.09%	89.14%	89.96%
20	5→4	79.54%	89.35%	90.18%	91.06%	91.91%

A summary of the number of machine learning models that have reached the target is presented in Fig. 5. There were 20 models for each machine learning in each experiment.

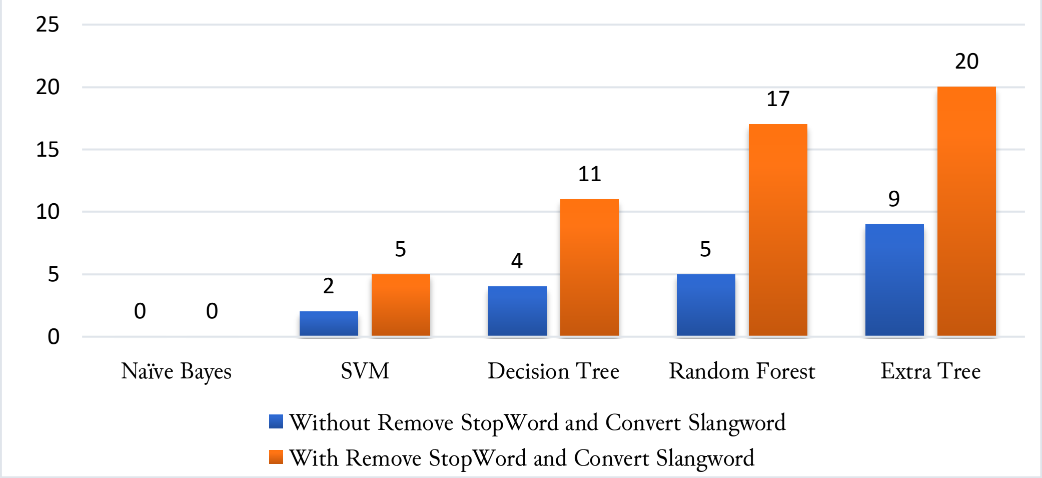


Fig. 5. Number of Machine Learning Models Reaching the Performance Targets

The effect of removing the stop words and converting slang word on CDSA were shown by calculating the difference between the accuracy as shown in Table 6 and the accuracy in Table 5. The results show that removing stop words and converting slang words could increase the accuracy by 15.70% in the SVM machine learning model (Table 7).

Table 7. Increased Accuracy of Model Without Remove Stop word and Convert Slang word to Model With Remove Stop word and Convert Slang Word

No Model	Source Dataset → Target Dataset	Machine Learning				
		Non-Ensemble			*Ensemble*	
		NB	*SVM*	*DT*	*RF*	*ET*
1	1→2	0.51%	2.09%	1.57%	3.21%	3.34%
2	1→3	1.81%	2.23%	3.58%	2.41%	2.58%
3	1→4	0.32%	2.58%	3.40%	3.36%	3.00%
4	1→5	0.63%	2.80%	2.91%	3.08%	3.27%
5	2→1	0.59%	12.13%	3.43%	2.75%	3.52%
6	2→3	0.11%	11.70%	3.48%	3.66%	3.56%
7	2→4	-0.27%	13.57%	2.80%	2.48%	2.75%
8	2→5	0.23%	15.70%	4.82%	3.88%	3.07%
9	3→1	0.81%	2.98%	3.65%	3.79%	3.74%
10	3→2	0.90%	3.24%	5.17%	5.91%	4.43%
11	3→4	0.78%	3.24%	4.69%	4.36%	3.55%
12	3→5	0.84%	3.59%	5.92%	4.91%	3.81%
13	4→1	1.80%	2.21%	4.74%	2.80%	1.71%
14	4→2	1.19%	2.70%	2.44%	3.41%	3.05%
15	4→3	1.86%	2.12%	3.15%	2.66%	2.50%
16	4→5	1.59%	3.09%	3.94%	2.89%	2.76%
17	5→1	2.98%	2.53%	3.43%	3.65%	3.16%
18	5→2	2.15%	2.31%	3.50%	3.15%	3.60%
19	5→3	2.50%	1.64%	3.65%	2.79%	2.51%
20	5→4	2.48%	1.67%	3.27%	2.61%	2.27%

Table 7 shows that machine learning models rather than SVM could increase the accuracy up to 2.98% for Naïve Bayes, 5.92% for the Decision Tree, by 5.91% for Random Forest, and by 4.43% for the Extra Tree. The SVM method mainly required better preprocessing, especially in dataset 2 to another dataset (Fig. 6).

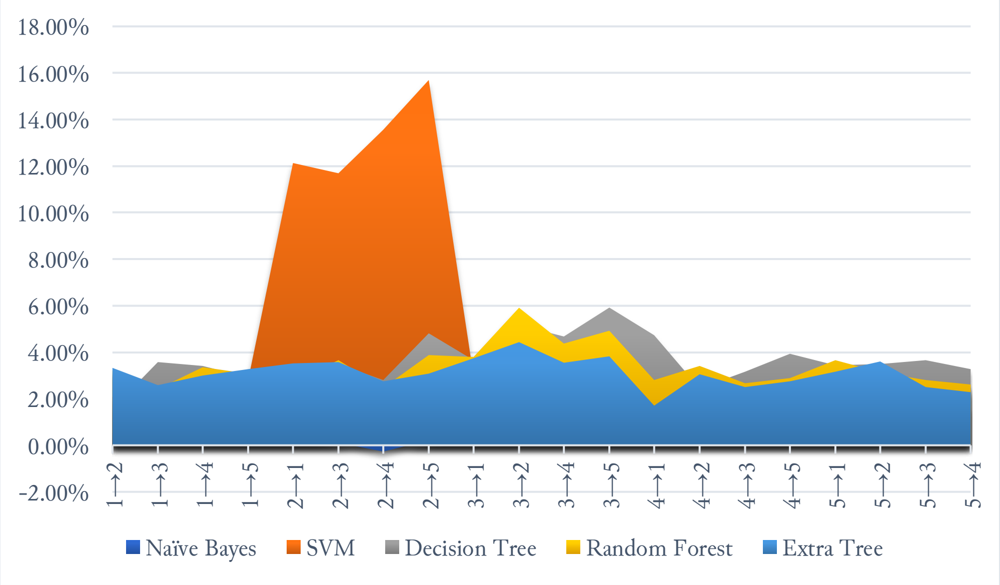


Fig. 6. Increased Model Accuracy for Each Machine Learning

Based on the experiment, the number of records in the dataset was found not to affect the accuracy of CDSA. For example, dataset 1 had fewer records compared to dataset 2. Dataset 1 was used as the source of dataset 2, and its accuracy was higher than dataset 2 used as a source for dataset 1. However, this did not apply to the relationship between dataset 1 and dataset 5, where dataset 5 was found better as a source for dataset 1. The existence of a stop word and slang word could increase the noise from a dataset. Stop words existed in all types of classes and could not be used as a feature. So it has been proven that removing the stop words in CDSA was better than keeping it. Slang words cannot be deleted but must be converted to normal words, and this process requires a conversion dictionary. We have compiled a slang word dictionary and considered the number of collections to be good because they have been proven to increase accuracy. The next analysis was a comparison of machine learning models. This study indicated that the tree-based ensemble machine learning (RF and ET) method was better than other methods (NB, SVM, and DT). The Extra Tree method was found as the best method, followed by the Random Forest. The next best method was the Decision Tree, followed by SVM. So, this research has succeeded in finding a machine learning model for CDSA in Indonesian, using tree-based ensemble machine learning, especially the Extra Tree method with a maximum accuracy of 91.91% on the test dataset. The accuracy of this CDSA was found higher than the results of sentiment analysis research in Indonesian such as research [13] with an accuracy of only 76.2%, and sentiment analysis on YouTube comment using SVM with an accuracy of 84% [14]. Sentiment analysis using a decision tree with emoticon and slang dictionary obtained 76.06% accuracy [15]. SVM on YouTube comment obtained 79.412% [16], and SVM with linear kernel function obtained the accuracy of 62.76% [17]. Meanwhile, the Naive Bayes and SVM combination resulted in an accuracy of 91% [19], and previous research about categorizing sentiment fanaticism using Random Forest resulted in an accuracy of 91.8% [21].

4. Conclusion

The main aim of this research is to develop the CDSA model and test its performance with a number of the experimental dataset. This research also observes the benefit of removing any stop words and converting the slang words in CDSA. The accuracy of CDSA performance in the Indonesian language would be better by conducting the preprocessing, removing the stop words, and converting the slang words. The classification model was built using ensemble-based machine learning. In this research, the best model was found by building it with the Extra Tree Classifier. Then, to obtain the maximum results for labeling the cross-domain, it could be done by selecting a clean dataset that rich in features as the source dataset. Thus, the best model to conduct CDSA is by using the training data that rich in features, implementing the complete preprocessing by removing any stop words and converting the slang words, and using the tree-based ensemble machine learning. This model is expected to be a good alternative for CDSA in the Indonesian language. Future research for CDSA in the Indonesian language should search other dataset sources such as Twitter, Facebook, and Instagram. It also can be using the topic from other domains. The research could be addressed to implement the model using deep learning combined with the feature extraction using the word embedding, unigram, bigram even trigram for the higher.

Acknowledgment

The authors would like to thank the Computational Intelligence and Technologies laboratory (CIT Lab) research group, the Center of Advanced Computing Technology (C-ACT), Fakulti Teknologi Maklumat dan Komunikasi, Universiti Teknikal Malaysia Melaka (UTeM) and Informatics Engineering Department, Universitas Pembangunan Nasional "Veteran" Yogyakarta Indonesia for their incredible support for this research.

Declarations

Author contribution. All authors have equally contributed to this article. As well, all authors have read and approved the final version of the article.

Funding statement. None of the authors has received any funding from any institution or source.

Conflict of interest. The authors declare no conflict of interest.

Additional information. No additional information is available for this paper.

References

[1] T. Al-Moslmi, N. Omar, S. Abdullah, and M. Albared, “Approaches to Cross-Domain Sentiment Analysis: A Systematic Literature Review,” *IEEE Access*, vol. 5, no. c, pp. 16173–16192, 2017, doi: 10.1109/ACCESS.2017.2690342.

[2] J. S. Deshmukh and A. K. Tripathy, “Entropy Based Classifier for Cross-Domain Opinion Mining,” *Appl. Comput. Informatics*, vol. 14, no. 1, pp. 55–64, 2018, doi: 10.1016/j.aci.2017.03.001.

[3] A. A. Aziz, A. Starkey, and M. C. Bannerman, “Evaluating Cross Domain Sentiment Analysis using Supervised Machine Learning Techniques,” in *2017 Intelligent Systems Conference, IntelliSys 2017*, 2017, no. September, pp. 689–696, doi: 10.1109/IntelliSys.2017.8324369.

[4] B. Heredia, T. M. Khoshgoftaar, J. Prusa, and M. Crawford, “Cross-Domain Sentiment Analysis: An Empirical Investigation,” in *2016 IEEE 17th International Conference on Information Reuse and Integration*, 2016, pp. 160–165, doi: 10.1109/IRI.2016.28.

[5] F. Gräßer, H. Malberg, S. Kallumadi, and S. Zaunseder, “Aspect-Based Sentiment Analysis of Drug Reviews Applying Cross-Domain and Cross-Data Learning,” in *2018 International Digital Health Conference - ACM International Conference Proceeding Series*, 2018, vol. 2018-April, pp. 121–125, doi: 10.1145/3194658.3194677.

[6] N. X. Bach, V. T. Hai, and T. M. Phuong, “Cross-Domain Sentiment Classification With Word Embeddings and Canonical Correlation Analysis,” in *Proceedings of the Seventh Symposium on Information and Communication Technology, SoICT 2016*, 2016, vol. 08-09-Dece, pp. 159–166, doi: 10.1145/3011077.3011104.

[7] F. H. Khan, U. Qamar, and S. Bashir, “Enhanced Cross-Domain Sentiment Classification Utilizing a Multi-Source Transfer Learning Approach,” *Soft Comput.*, vol. 23, no. 14, pp. 5431–5442, 2018, doi: 10.1007/s00500-018-3187-9.

[8] K. Katsarou and D. S. Shekhawat, “CRD-Sentense: Cross-Domain Sentiment Analysis Using An Ensemble Model,” *11th Int. Conf. Manag. Digit. Ecosyst. MEDES 2019*, no. November, pp. 88–94, 2019, doi: 10.1145/3297662.3365808.

[9] D. Bollegala, D. Weir, and J. Carroll, “Cross-Domain Sentiment Classification using a Sentiment Sensitive Thesaurus,” *IEEE Trans. Knowl. Data Eng.*, vol. 25, no. 8, pp. 1719–1731, 2013, doi: 10.1109/TKDE.2012.103.

[10] R. Suharshala, K. Anoop, and V. L. Lajish, “Cross-Domain Sentiment Analysis on Social Media Interactions using Senti-Lexicon based Hybrid Features,” *Proc. 3rd Int. Conf. Inven. Comput. Technol. ICICT 2018*, pp. 772–777, 2018, doi: 10.1109/ICICT43934.2018.9034272.

[11] D. H. Jayani, “Orang Indonesia Habiskan Hampir 8 Jam untuk Berinternet,” *26 February 2020*. 2020., Available at: katadata.co.id

[12] Gatra.com, “Pemakaian Bahasa Indonesia Termasuk Terbesar di Medsos,” *Gatra.com*, 2019, Available at: gatra.com

[13] J. Savigny and A. Purwarianti, “Emotion classification on Youtube comments using word embedding,” in *International Conference on Advanced Informatics: Concepts, Theory and Applications*, 2017, pp. 1–5, doi: 10.1109/ICAICTA.2017.8090986.

[14] F. I. Tanesab, I. Sembiring, and H. D. Purnomo, “Sentiment analysis model based on Youtube comment using support vector machine,” *Int. J. Comput. Sci. Softw. Eng.*, vol. 6, no. 8, pp. 180–185, 2017. Available at: Google Scholar

[15] A. G. Prasad, S. Sanjana, S. M. Bhat, and B. S. Harish, “Sentiment Analysis For Sarcasm Detection on Streaming Short Text Data,” in *2017 2nd International Conference on Knowledge Engineering and Applications, ICKEA 2017*, 2017, no. 2009, pp. 1–5, doi: 10.1109/ICKEA.2017.8169892.

[16] M. Andriansyah *et al.*, “Cyberbullying comment classification on Indonesian selebgram using support vector machine method,” in *The 2nd International Conference on Informatics and Computing*, 2018, vol. 2018-Janua, pp. 1–5, doi: 10.1109/IAC.2017.8280617.

[17] E. Rinaldi and A. Musdholifah, "FVEC-SVM for opinion mining on Indonesian comments of youtube video," *Proc. 2017 Int. Conf. Data Softw. Eng. ICoDSE 2017*, vol. 2018-Janua, pp. 1–5, 2018, doi: 10.1109/ICODSE.2017.8285860.

[18] N. Anggraini and M. J. Tursina, "Sentiment analysis of school zoning system on Youtube social media using the K-nearest neighbor with levenshtein distance algorithm," in *7th International Conference on Cyber and IT Service Management*, 2019, no. May, pp. 1–4, doi: 10.1109/CITSM47753.2019.8965407.

[19] A. N. Muhammad, S. Bukhori, and P. Pandunata, "Sentiment analysis of positive and negative of YouTube comments using naïve bayes-support vector machine (NBSVM) classifier," in *International Conference on Computer Science, Information Technology, and Electrical Engineering*, 2019, vol. 1, pp. 199–205, doi: 10.1109/ICOMITEE.2019.8920923.

[20] R. Novendri, A. S. Callista, D. N. Pratama, and C. E. Puspita, "Sentiment analysis of YouTube movie trailer comments using naïve bayes," *Bull. Comput. Sci. Electr. Eng.*, vol. 1, no. 1, pp. 26–32, 2020, doi: 10.25008/bcsee.v1i1.5.

[21] A. S. Aribowo, H. Basiron, N. S. Herman, and S. Khomsah, "Fanaticism Category Generation Using Tree-based Machine Learning Method," *J. Phys. Conf. Ser.*, vol. 1501, no. 1, 2020, doi: 10.1088/1742-6596/1501/1/012021.

[22] N. Sultana and M. M. Islam, "Meta classifier-based ensemble learning for sentiment classification," in *Proceedings of International Joint Conference on Computational Intelligence, e, Algorithms for Intelligent Systems*, 2020, vol. 669, pp. 1–481, doi: 10.1007/978-981-13-7564-4.

[23] S. Khomsah and A. S. Aribowo, "Model text-preprocessing komentar Youtube dalam bahasa Indonesia," *Rekayasa Sist. dan Teknol. Informasi, RESTI*, vol. 4, no. 4, pp. 648–654, 2020, doi: 10.29207/resti.v4i4.2035

[24] T. F. Abidin, M. Hasanuddin, and V. Mutiawani, "N-grams based features for Indonesian tweets classification problems," *Proc. - 2017 Int. Conf. Electr. Eng. Informatics Adv. Knowledge, Res. Technol. Humanit. ICELTICs 2017*, vol. 2018-Janua, no. ICELTICs, pp. 307–310, 2017, doi: 10.1109/ICELTICS.2017.8253287.

[25] Y. Hao, T. Mu, R. Hong, M. Wang, X. Liu, and J. Y. Goulermas, "Cross-Domain Sentiment Encoding through Stochastic Word Embedding," *IEEE Trans. Knowl. Data Eng.*, vol. 32, no. 10, pp. 1909–1922, 2020, doi: 10.1109/TKDE.2019.2913379.

[26] B. ZHANG, X. XU1, M. YANG, X. CHEN, and Y. YE, "Cross-domain Sentiment Classification by Capsule Network with Semantic Rules," *IEEE Access*, vol. 6, pp. 58284–58294, 2018, doi: 10.1109/ACCESS.2018.2874623.

[27] Naveen Bindra and Manu Sood, "Detecting DDoS Attacks Using Machine Learning Techniques and Contemporary Intrusion Detection Dataset," *Autom. Control Comput. Sci.*, vol. 53, no. 5, pp. 419–428, Sep. 2019, doi: 10.3103/S0146411619050043

[28] L. B. Shyamasundar and P. Jhansi Rani, "A multiple-layer machine learning architecture for improved accuracy in sentiment analysis," *Comput. J.*, vol. 63, no. 3, pp. 395–409, 2019, doi: 10.1093/comjnl/bxz038.

[29] N. Cahyana, S. Khomsah, and A. S. Aribowo, "Improving imbalanced dataset classification using oversampling and gradient boosting," *5th Int. Conf. Sci. Inf. Technol. Embrac. Ind. 4.0 Towar. Innov. Cyber Phys. Syst. ICSITech*, pp. 217–222, 2019, doi: 10.1109/ICSITech46713.2019.8987499.

[30] A. S. Aribowo, H. Basiron, N. S. Herman, and S. Khomsah, "An evaluation of preprocessing steps and tree-based ensemble machine learning for analysing sentiment on Indonesian youtube comments," *Int. J. Adv. Trends Comput. Sci. Eng.*, vol. 9, no. 5, pp. 7078–7086, 2020, doi: 10.30534/ijatcse/2020/29952020.

[31] S. Kaur, P. Kumar, and P. Kumaraguru, "Automating fake news detection system using multi-level voting model," *Soft Comput.*, vol. 24, no. 12, pp. 9049–9069, 2020, doi: 10.1007/s00500-019-04436-y.

[32] A. K. Mohamad, M. Jayakrishnan, and N. H. Nawi, "Employ twitter data to perform sentiment analysis in the Malay language," *Int. J. Adv. Trends Comput. Sci. Eng.*, vol. 9, no. 2, pp. 1404–1412, 2020, doi: 10.30534/ijatcse/2020/76922020.