Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 7, Number 4, October 2023 e-ISSN : 2541-2019
DOI : https://doi.org/10.33395/sinkron.v8i4.12989 p-ISSN : 2541-044X

Identification of 10 Regional Indonesian Languages Using Machine Learning

Azhar Baihaqi Nugraha[1)*], Ade Romadhony[2)]
[1,2,3)]School of Computing, Study Program of Informatics, Telkom University, Bandung, Indonesia
[1)]azharnugraha@student.telkomuniversity.ac.id, [2)]aderomadhony@telkomuniversity.ac.id

Submitted : Sep 10, 2023 | **Accepted** : Sep 11, 2023 | **Published** : Oct 1, 2023

Abstract: Language Identification plays a pivotal role in deciphering the rich tapestry of Indonesia's diverse regional languages, encompassing a wide spectrum of scripts, and spoken forms. Language Identification, an integral component of Natural Language Processing, is frequently addressed through Text Classification. In this study, we embark on the task of identifying 10 Indonesian languages, leveraging the NusaX dataset, with the overarching objective of contextual language determination. To achieve this, we harness a diverse array of machine learning techniques, including Support Vector Machine, Naïve Bayes Classifier, Decision Tree, Rocchio Classification, Logistic Regression, and Random Forest. We complement these methods with two distinct feature extraction approaches: N-gram and TF-IDF. This comprehensive approach enables us to construct robust models for language identification. Our findings unveil the strong efficacy of these models in discerning Indonesian languages, with the Naïve Bayes Classifier emerging as the frontrunner, achieving an impressive accuracy rate of 99.2% with TF-IDF and an even more remarkable 99.4% with N-Gram. To gain deeper insights, we delve into error analysis, revealing that misclassifications often stem from shared words across different languages. This research is underpinned by the necessity for a robust language identification model, underscoring its critical role within the complex linguistic landscape of Indonesian regional languages. These results hold great promise for applications in automated language processing and understanding within this diverse and multifaceted linguistic context.

Keywords: Decision Tree, Language Identification, Naïve Bayes Classifier, Support Vector Machine, Text Classification.

INTRODUCTION

Language serves as a structured mode of communication, comprising elements like words, phrases, clauses, and sentences, expressed both verbally and in written form (Wiratno and Santosa 2014). Indonesia, an archipelagic nation, is rich in tribes, provinces, and regions, resulting in a plethora of regional languages. Ranking second globally in regional linguistic diversity, Indonesia comes after Papua New Guinea (Tondo 2009). Local languages persist, particularly in remote areas, while even in major cities, some locals continue to converse in regional dialects. Nonetheless, many Indonesians find it challenging to discern the language spoken in conversations or during interactions. Despite the historical usage of regional languages, these dialects are integral to a dynamic cultural landscape susceptible to change, potentially leading to language shifts if not vigilantly monitored (Setyawan 2011).

Language identification is an important pre-processing step in many automated systems that operate using written text such as Text Classification. (TC). An illustrative case is within TC tasks, where language identification as a preprocessing measure demonstrates commendable effectiveness, evident in precision and recall values (Jauhiainen, Lindén, and Jauhiainen 2017). The inception of language

identification traces back to 1965, pioneered by the statistician Mustone, who trained computers to discern languages at the word level, distinguishing English, Swedish, and Finnish (Jauhiainen et al. 2019). To generate language identification, the TC method is required, which is one of the branches in Language Processing. (NLP). TC, as the name suggests, serves to categorize text, often employing distinct cues or regulations tailored for each classification (Ahmad 2018).

In this investigation, the researcher aims to recognize 10 Indonesian regional languages, namely Aceh, Bali, Banjar, Bugis, Madura, Minangkabau, Jawa, Ngaju, Sunda, and Toba Batak. This dataset originates from prior research known as NusaX (Winata et al. 2022). To carry out language identification, there are multiple approaches available for determining the language of these regions. Drawing on earlier studies in Text Classification (TC) for language identification (Jauhiainen et al. 2019), several methodologies will be employed, including Support Vector Machine (SVM), Naïve Bayes Classifier (NBC), Decision Tree (DT), Rocchio Classification (RC), Logistic Regression (LR), and Random Forest (RF). The development of the model also involved the use of two different features: n-gram and TF-IDF. The primary achievement of this study lies in creating a model capable of precisely identifying and categorising the languages spoken in this particular region. This holds significant significance within the realm of natural language processing (NLP) and text analysis, especially considering the wide array of languages employed throughout Indonesia.

LITERATURE REVIEW

In a preceding investigation, Tommi V et al. (2010) conducted an extensive analysis of N-gram character identification (Vatanen, Väyrynen, and Virpioja 2010). The research involved language identification in a concise passage from the Universal Declaration of Human Rights, encompassing 281 languages. N-gram features were employed as extraction elements, alongside identification models like Absolute Discounting, Katz, Kneser-Ney, Ranking method, Lidstone, and Laplace.

In connection with other research, King and Dehdari, employing methods such as On notation, n-gram features, LIGA algorithm, whatlang programs, VariKN toolkit, and HeLI, undertook language identification (Jauhiainen et al. 2017). Their study revolved around evaluating six language identification techniques across 285 languages. The study findings underline the effectiveness of the employed method in achieving proficient language identification across a wide array of languages.

Text Classification

Text classification plays a crucial role across a wide array of applications, as it involves the task of categorizing documents into predefined groups or classes. This process typically follows pre-existing class labels, making it a supervised learning task (Kowsari et al. 2019). While text classification often relies on statistical analysis of word frequencies to create document models, language identification poses a unique challenge, especially for languages that don't rely on spaces to delineate word boundaries (Jauhiainen et al. 2019).

Language classification entails a series of essential steps, including preprocessing, feature extraction, modeling, and evaluation. Among these steps, the selection of an appropriate classification method stands out as paramount. Without a comprehensive grasp of each method's conceptual underpinnings, arriving at an effective classification model becomes a challenging endeavor.

Support Vector Machine

The inception of the Support Vector Machine (SVM) dates back to 1963, when Vapnik and Chervonenkis initially devised it. Subsequently, in the early 1990s, B.E. Boser and colleagues extended this original version to accommodate a nonlinear formulation (Kowsari et al. 2019). To extend the fundamental SVM model for multiclass classification, a straightforward approach involves employing a sequence of one-vs-rest classifications. In this strategy, the language of the test document is determined by the classification with the highest score.

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 7, Number 4, October 2023
DOI : https://doi.org/10.33395/sinkron.v8i4.12989
e-ISSN : 2541-2019
p-ISSN : 2541-044X

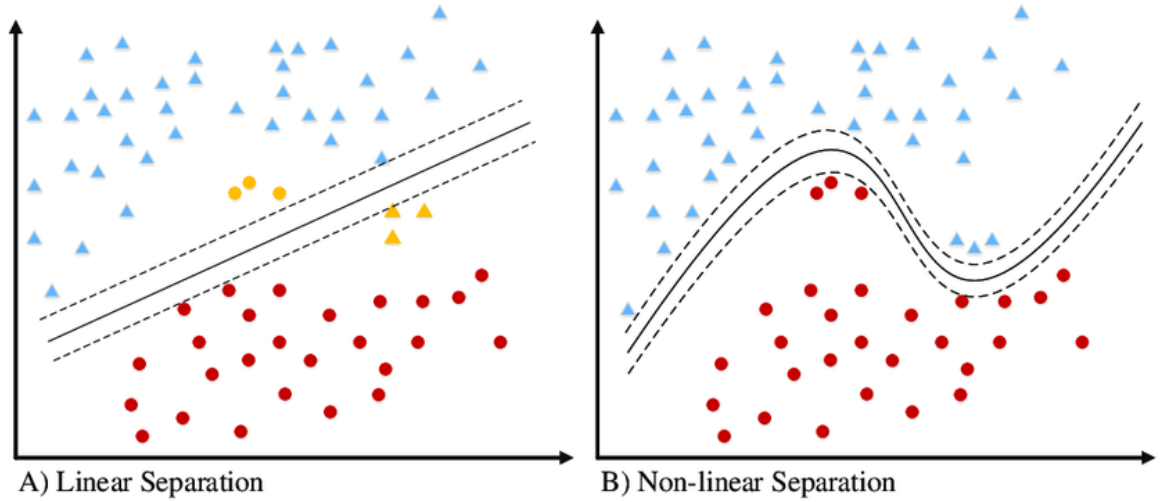

Fig. 1 Linear and Non-linear SVM

As depicted in Figure 1, Support Vector Machine (SVM) exhibits various iterations, including Linear and Non-Linear SVM. Class 1 is represented by red dots, class 2 by blue dots, and unclassified data is denoted by yellow dots.

Naïve Bayes Classifier

The Naïve Bayes Classifier (NBC) serves the purpose of estimating probabilities or likelihoods for predictive inference, drawing from past data, or facilitating categorization within a system (Tuhenay and Mailoa 2021). The fundamental formula for the Naïve Bayes Classifier is presented in equation (1).

$$P(H \mid X) = \frac{P(H|X)\, P(H)}{pii} \quad (1)$$

Where:

X = Unknown data class
H = Data X is a class-specific hypothesis
P(H | X) = Probability of the hypotheses H based on conditions X
P(H) = Probability of hypotheses H (prior)
P(X) = Probability of X

Decision Tree

Since 2001, Hakkinen and Tian have employed Decision Trees (DT) for language identification purposes. They utilized character- and context-based DT techniques, omitting word frequency information in their approach (Jauhiainen et al. 2019). The training process of a Decision Tree involves iteratively splitting nodes into child nodes based on optimization criteria derived from information theory. At each node, a feature is chosen to maximize the information gain for that node. The evaluation involves traversing the tree until only a single node remains. De Mantaras introduced a statistical model for feature selection in Decision Trees. Each training instance involves a positive (p) and negative (n) count.

Rocchio Classification

In 1971, Rocchio introduced the renowned Rocchio algorithm, a significant technique for text classification utilizing vector space models (Andayani et al. 2019; Rocchio 1971). Employing a vector space model mandates the identification of class boundaries to facilitate appropriate classification. Rocchio's approach employs centroids as boundaries to impose constraints. The centroid of a given class, denoted as c, is computed as the mean value of all the vectors within that class. The computation of the centroid value is outlined in equation (2).

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 7, Number 4, October 2023 e-ISSN : 2541-2019
DOI : https://doi.org/10.33395/sinkron.v8i4.12989 p-ISSN : 2541-044X

$$\vec{\mu}(c) = \frac{1}{[D_c]} \sum_{d \in D_c} \vec{v}(d) \quad (2)$$

Where:

μ (c) = Class centroid C
D_c = Total Class C documents
$\underset{v}{\rightarrow}(d)$= Normalized document vector

Logistic Regression

Logistic Regression (LR) is a machine learning technique utilized for conducting classification tasks. The LR algorithm is particularly suitable for binary data processing, where the values are limited to 0 and 1, signifying membership in distinct categories (Hassan, Ahamed, and Ahmad 2022). Depending on the number of categories, Logistic Regression can be categorized in the following manner:

(i) Binomial: Involves two distinct potential values within the target variable: "0" or "1", which can signify outcomes like "loss" versus "win", "fail" versus "pass", "alive" versus "dead", and similar scenarios.
(ii) Multinomial: refers to target variables with three or more non-ordered types, such as "virus A" versus "virus B" versus "virus C", where the types lack quantitative significance.
(iii) Ordinal: Encompasses ordered categories in the target variable; for instance, rating scores could be categorized as "very good", "good", "bad", and "very bad". In this case, each category can be assigned a numerical value like 0, 1, 2, 3, or the reverse.

Random Forest

Random Forest (RF) is a computational technique involving the simultaneous utilization of multiple Decision Trees (DT). Within this algorithm, Decision Trees play a pivotal role (Shah et al. 2020). The construction of a Decision Tree forms a core principle in the creation of RF. Consequently, the random forest encompasses a collection of these Decision Tree algorithms, collectively employed to classify novel items by relying on input vectors. Each constructed Decision Tree is harnessed for classification purposes. The Random Forest process involves several sequential stages:

Step 1: From the training data, select a random K point.
Step 2: Raise a DT at point K.
Step 3: Before repeating steps 1 and 2, determine the number of NTree trees you want to build.
Step 4: Predict the y value by creating each of the NTree trees for a new data point and giving the average of all the predicted y values to the new data points.

Feature Extraction

Data is described through numerous attributes, which may be binary, categorical, or continuous in nature (Wang, Su, and Yu 2020). These attributes correspond to input variables or attributes, and determining a suitable data representation for effective measurement is vital. Informational text exhibits diverse data sizes and structures. The key consideration in feature extraction from text is the presence of structured data. Typically, raw, unprocessed data undergoes transformation into structured formats. The process of obtaining impactful attributes is referred to as feature extraction.

Feature Extraction (FE) addresses the challenge of identifying a concise and informative set of features. For classification and regression tasks, the prevalent and practical approach to data representation involves defining features as vectors. FE organizes data into a straightforward table, where each feature corresponds to a quantitative or qualitative measurement, referred to as "attributes" or "variables". This study centers around two features, N-Gram and TF-IDF, which are compared to enhance language identification performance.

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 7, Number 4, October 2023 e-ISSN : 2541-2019
DOI : https://doi.org/10.33395/sinkron.v8i4.12989 p-ISSN : 2541-044X

a) N-Gram

N-Gram involves sequences of n units, often single characters or strings separated by spaces (Zaman, Hariyanti, and Purwanti 2015). N-grams represent groups of N characters extracted from strings, with the introduction of empty markers at the string's beginning and end to establish start and end boundaries. To illustrate, if we add these markers to the string "TEXT," N-Gram spaces result as follows:

Unigram: T, E, X, T
Bigram: TE, EX, XT
Trigram: TEX, EXT
Quadgram: TEXT

This reveals that a string of size n yields n unigrams, n+1 bigrams, n+1 trigrams, and so forth. Leveraging N-Grams for word matching finds applications in recovering noise-affected ASCII inputs, interpreting postal codes, information retrieval, and diverse applications within Natural Language Processing (Zaman et al. 2015).

b) TF-IDF

TF-IDF combines the weights of Term Frequency (TF) and Inverse Document Frequency (IDF) through multiplication. TF assigns a higher weight to commonly used words in sentences, while IDF diminishes the weight for words occurring frequently across sentences. TF-IDF represents the weight assigned to a word based on its Term Frequency (TF) and the inverse Document Frequency (DF). This can be understood through equations 1 and 2:

$$TF - IDF(w,d) = TF(w,d) \times IDF(W) \quad (3)$$

Where:

$TF - IDF$ = the weight of a word in the entire document
W = a word
d = a document
$TF(w,d)$ = the frequency word w occurrence in a document

$$IDF\ (w) = \log(\frac{N}{DF(w)}) \quad (4)$$

Where:

$IDF(w)$ = DF's inverse from the word 'w'
N = total of documents
$DF(w)$ = total of documents that has the word 'w'

The sentence's length inherently influences word weights, necessitating an assessment of sentence length disparity. To address this, every sentence's feature vector is normalized to a standard length unit. This normalization process yields a sentence feature vector suitable for classification input.

Error Analysis

Error Analysis (EA) serves as an assessment aimed at pinpointing errors or recognizing error-inducing patterns within models or analytical methodologies. In the study by Van Aken (2018), Error Analysis was performed on the class exhibiting the highest number of predictive errors (van Aken et al. 2018). The manual analysis executed within this study involves a thorough examination of the text, context, and attributes that could potentially underlie the occurrence of the errors.

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 7, Number 4, October 2023 e-ISSN : 2541-2019
DOI : https://doi.org/10.33395/sinkron.v8i4.12989 p-ISSN : 2541-044X

METHOD

The primary objective of this concluding research task is to employ classification techniques to discern various Indonesian regional languages and to scrutinize the employed models and features. The study employs textual data encompassing 10 distinct Indonesian regional languages. Multiple methods will be assessed, each subject to performance evaluation. A depiction of the anticipated system framework is outlined in Figure 2.

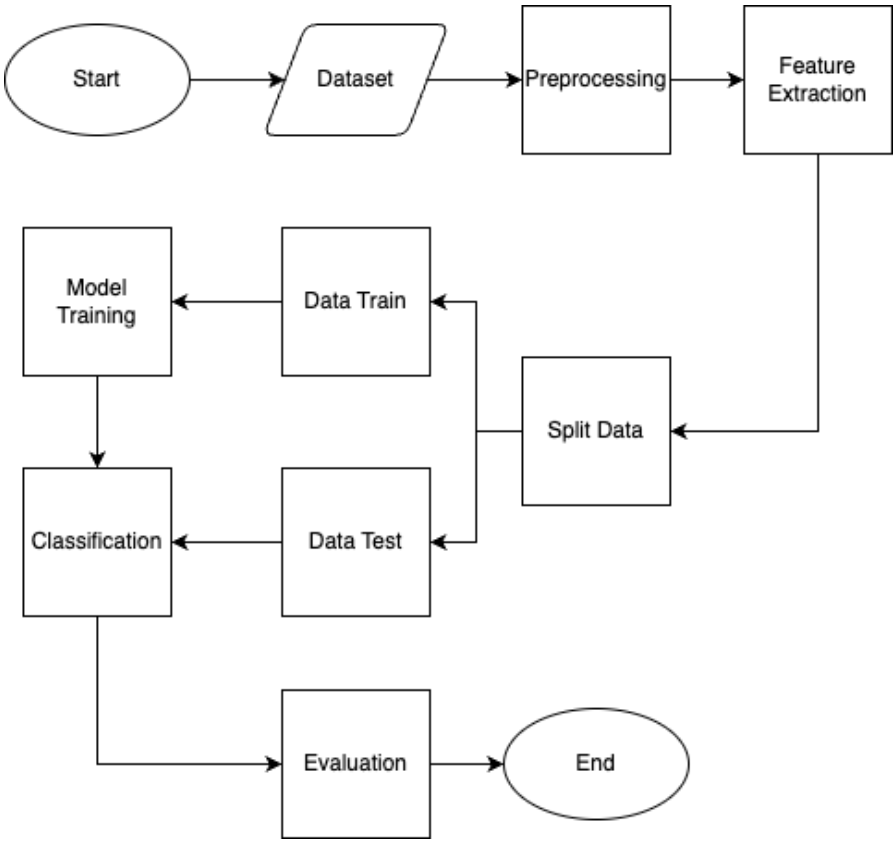

Fig. 2 System Planning Flowchart

Dataset

The dataset utilized in this investigation comprises texts in ten Indonesian regional languages, namely Aceh, Bali, Banjar, Bugis, Madura, Minangkabau, Java, Ngaju, Sunda, and Toba Batak. The dataset was sourced from the NusaX research project (Winata et al. 2022).

Table 1. Dataset NusaX

Label	Training	Testing	Validation
Acehnese	500	400	100
Banjarnese	500	400	100
Madurese	500	400	100
Ngaju	500	400	100
Sundanese	500	400	100
Balinese	500	400	100
Buginese	500	400	100
Javanese	500	400	100
Minangkabau	500	400	100
Toba_batak	500	400	100
TOTAL	**5500**	**4400**	**1100**

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 7, Number 4, October 2023 e-ISSN : 2541-2019
DOI : https://doi.org/10.33395/sinkron.v8i4.12989 p-ISSN : 2541-044X

Preprocessing

After obtaining the dataset, the subsequent phase involves initial processing, or preprocessing, aimed at rectifying any issues encountered during data manipulation. In this research, the preprocessing method employed is data cleansing. This particular step is illustrated in the flowchart depicted in Figure 3, as outlined below:

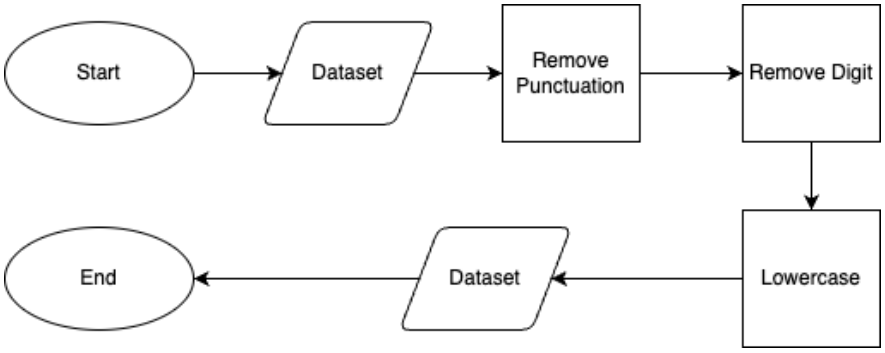

Fig. 3 Preprocessing Flowchart

a) *Remove Punctuation*
The "Remove Punctuation" process involves eliminating punctuation marks that hold no relevance in feature extraction. The punctuation marks to be removed during this process encompass: '!"#$%&'()*+,-./:;<=>?@[]^_`{ |}~'.

b) *Remove Digit*
The "Remove Digit" procedure entails the elimination of numerical characters. This action is employed within the domain of language identification research due to the consistent typing conventions shared across various languages. Consequently, numeric characters have no impact on the process of language identification.

c) *Lowercase*
Lowercase involves converting uppercase letters into lowercase ones. This procedure aims to mitigate discrepancies in word representation caused by capitalization variations, such as 'I' and 'i'.

Feature Extraction Step

Feature extraction entails converting textual data into numerical formats, enabling comprehension and processing by classification models. The feature extraction methods employed in this research encompass TF-IDF and N-Gram.

Classification Algorithm

The classification task is executed employing SVM algorithms, Naïve Bayes Classifiers, Decision Trees, Rocchio Classifications, Logistic Regressions, and Random Forest. These algorithms are chosen in combination with N-Gram and TF-IDF features to identify compact and informative feature sets. The algorithms will undergo training using the provided training data, utilizing default parameters from the following parameter list:

a) *SVM*
The SVM was trained using the following parameters: C: 1.0; kernel: 'rbf'; degree: 3; gamma: 'scale'; coef0: 0.0; shrinking: True; probability: False; tol: 1E-3; cache_size: 200; class_weight: none; verbose: False; max_iter: 1; decision_function_shape: 'ovr'; break_ties: False; random_state: None.

b) *Naïve Bayes Classifier*
The Naïve Bayes Classifier was trained using these parameters: alpha: 1.0; force_alpha: False; fit_prior: True; class_prior: None.

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 7, Number 4, October 2023 e-ISSN : 2541-2019
DOI : https://doi.org/10.33395/sinkron.v8i4.12989 p-ISSN : 2541-044X

c) *Decision Tree*
The Decision Tree was trained using the following parameters: criterion: 'gini'; splitter: 'best'; max_depth: None; min_samples_split: 2; min_samples_leaf: 1; min_weight_fraction_leaf: 0.0; max_features: None; random_state: None; max_leaf_nodes: None; min_impurity_decrease: 0.0; min_impurity_split: None; class_weight: None; presort: 'deprecated'; ccp_alpha: 0.0.

d) Rocchio Classification
The Rocchio Classification was trained using these parameters: metric: 'euclidean'; shrink_threshold: None.

e) *Logistic Regression*
The Logistic Regression was trained using the following parameters: penalty: 'l2'; dual: False; tol: 0.0001; C: 1.0; fit_intercept: True; intercept_scaling: 1; class_weight: None; random_state: None; solver: 'lbfgs'; max_iter: 100; multi_class: 'auto'; verbose: 0; warm_start: False; n_jobs: None; l1_ratio: None.

f) *Random Forest*
The Random Forest was trained using the following parameters: n_estimators: 100; criterion: 'gini'; max_depth: None; min_samples_split: 2; min_samples_leaf: 1; min_weight_fraction_leaf: 0.0; max_features: 'auto'; max_leaf_nodes: None; min_impurity_decrease: 0.0; min_impurity_split: None; bootstrap: True; oob_score: False; n_jobs: None; random_state: None; verbose: 0; warm_start: False; class_weight: None; ccp_alpha: 0.0; max_samples: None.

Evaluation

In this research, the assessment stage employed the Performance Evaluation Measure (PEM), utilizing the confusion matrix (CM), with the primary metric being Accuracy. The outcomes of the modeling process are characterized by four terms: True Positive (TP) represents the total of correctly predicted positive data; True Negative (TN) indicates the total of accurately predicted negative data; False Positive (FP) corresponds to the count of negative data incorrectly predicted as positive; and False Negative (FN) denotes the number of positive data mistakenly predicted as negative. The utilization of the confusion matrix is exemplified in Table 2.

Table 2. Confusion Matrix

Confussion Matrix		**Actual Value**	
		Positive	**Negative**
Prediction Value	Positive	TP	FP
	Negative	FN	TN

a) Accuracy

$$A = \frac{TP + TN}{TP + TN + FP + FN} \quad (5)$$

b) Precision

$$P = \frac{TP}{TP + FP} \quad (6)$$

c) Recall

$$R = \frac{TP}{TP + FN} \quad (7)$$

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 7, Number 4, October 2023 e-ISSN : 2541-2019
DOI : https://doi.org/10.33395/sinkron.v8i4.12989 p-ISSN : 2541-044X

d) F1Score

$$F1 = 2 \times \frac{(precision \times recall)}{(precision + recall)} \quad (8)$$

Following the employment of the confusion matrix for evaluation, the next step involves the comparison of the employed models and features. In this investigation, two features are considered, namely N-Gram and TF-IDF, alongside six models: Support Vector Machine (SVM), Naïve Bayes Classifier (NBC), Decision Tree (DT), Rocchio Classification (RC), Logistic Regression (LR), and Random Forest (RF).

RESULT

Employing a dataset comprising 5500 sentences, with 500 sentences per language for each of the 10 Indonesian regional languages, as training data proves highly effective for training language classification models. Utilizing 4400 sentences as test data, with 400 sentences for each language, yields notably satisfactory outcomes. The average accuracy and F1-Score classifications stand at 0.966 and 0.967, respectively, when employing TF-IDF features. Similarly, when using N-Gram features, the average accuracy and F1-Score classifications are 0.947 and 0.95. Comprehensive performance details of the classification models and features are provided in Table 3 and Table 4.

Table 3. Performance TF-IDF + Model

TF-IDF		
Model	**Accuracy**	**F1-Score**
NBC	**0.992**	**0.993**
SVC	0.988	0.989
DT	0.878	0.881
RC	0.983	0.983
LR	0.989	0.989
RF	0.967	0.968

As indicated in Table 3, the study encompassed six models, namely NBC, SVC, DT, RC, LR, and RF, utilizing TF-IDF feature extraction. Notably, NBC exhibited the highest and nearly perfect F1-Scores (0.992, 0.993), while SVC displayed commendable accuracy (0.988, 0.989). DT showcased accurate results (0.878, 0.881), RC demonstrated precision (0.983, 0.983), LR presented accuracies (0.989, 0.989), and RF exhibited accuracies (0.967, 0.968).

Table 4. Performance N-Gram + Model

N-Gram		
Model	**Accuracy**	**F1-Score**
NBC	**0.994**	**0.995**
SVC	0.972	0.972
DT	0.894	0.897
RC	0.874	0.885
LR	0.982	0.983
RF	0.968	0.968

As outlined in Table 4, the evaluation involved the same six models using N-gram feature extraction. NBC, again, emerged with the highest and almost flawless F1-Scores (0.994, 0.995), while SVC achieved solid accuracy (0.972, 0.972). DT yielded favorable precision (0.894, 0.897), RC attained accuracy (0.874, 0.885), LR maintained accuracies (0.982, 0.983), and RF maintained accuracies (0.968,

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 7, Number 4, October 2023
DOI : https://doi.org/10.33395/sinkron.v8i4.12989
e-ISSN : 2541-2019
p-ISSN : 2541-044X

0.968). The comparative performance of the six models using TF-IDF and N-gram features is visually depicted in Figure 2.

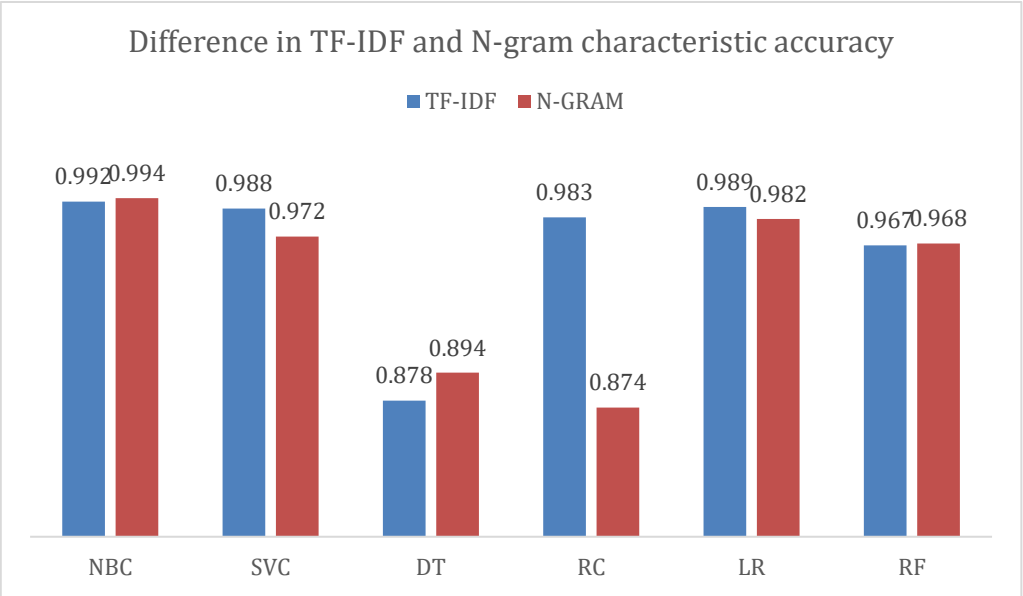

Fig.4 The differences in accuracy of six models with the use of TF-IDF and N-Gram features

DISCUSSIONS

In the examination of the test outcomes, an Error Analysis (EA) was conducted on the test results. The purpose of this analysis was to uncover the reasons behind instances where the classification model failed to correctly assign labels. EA focused on assessing the model with the highest accuracy, the Naïve Bayes Classifier (NBC) with TF-IDF extraction. From the NBC model's test results, a total of 28 test data points out of 4,400 were inaccurately classified. Instances of test data that were misclassified are detailed in Table 5.

Tabel 5. Example 4 Data From 28 Test Data That Failed to Classify

Sentences	Label	Prediction
badah parahne jne tiyan nelpon sing jemake twe...	Balinese	Javanese
palakunyo jan disarahan ka pak polisi kubua id…	Minangkabau	Madurese
ka gadeh hape xiaomi redmi note warna meuh nam...	Acehnese	Sundanese
di pilkada polri katunden nincapin pengamanan ...	Balinese	Toba_batak

Error Analysis (EA) was used on all texts where prediction results were different from the original label. It was based on what was learned from misclassified test data. For instance, consider the sentence ‘badah parahne jne tiyan nelpon sing jemake tweet sing balase email sing balase pesan sing balase', originally labeled as 'Balinese', yet misclassified as 'Javanese'. The EA outcomes disclosed that certain words within the sentence were encountered 1262 times in the 'Javanese' labeled dataset, whereas the same words appeared only 345 times in the 'Balinese' labeled dataset. The text contained several words that were frequently observed in the 'Javanese' language, including 'sing' which appeared 1256 times. The EA findings indicated that the classification model faltered due to the presence of identical words exhibiting a more dominant distribution in different language labels. The strength of this study lies in the nearly perfect language identification scores achieved across the 10 tested languages. However, a limitation of this research lies in the existence of languages with highly similar words, leading to predictive errors in the model.

For future research, it's suggested to expand the list of languages for Indonesian regional identification. Moreover, enhancing feature extraction and exploring alternative machine learning approaches like unsupervised or semi-supervised methods could be valuable. Comprehensive investigations should delve deeper into the proposed techniques and potential feature variations.

CONCLUSION

The concluding objective involved identifying Indonesian regional languages. This language identification process employed six classification models: Support Vector Machine (SVM), Naïve Bayes Classifier (NBC), Decision Tree (DT), Rocchio Classification (RC), Logistic Regression (LR), and Random Forest (RF), along with two feature extraction techniques: TF-IDF and N-Gram. The study also conducted a comparative analysis of the collective performance of the two feature extraction methods and the six classification models. The dataset used for classifying Indonesian regional languages was sourced from NusaX research, encompassing 10 Indonesian languages. The classification process utilized confusion matrices to gauge model accuracy, enabling model comparisons. Language identification yielded notably high accuracy scores, averaging 96% for each model with TF-IDF features and 94% for each model with N-Gram features. Based on the findings, the most proficient model for identifying the 10 Indonesian regional languages was NBC, achieving an accuracy of 0.992 with TF-IDF and 0.994 with N-Gram.

The results of the Error Analysis (EA) testing conducted on this final task highlighted that the classification model's failure to predict labels stemmed from word similarities across multiple languages and the prevalence of dominant words in different languages. Consequently, the classification model's performance was limited.

REFERENCES

Setyawan, Aan. 2011. "International_Proceeding_UNDIP_July__2,_2011_-_Aan_Setyawan." *Language Maintenance and Shift*.

van Aken, Betty, Julian Risch, Ralf Krestel, and Alexander Löser. 2018. "Challenges for Toxic Comment Classification: An In-Depth Error Analysis."

Andayani, U., D. Arisandi, Misbah Hasugian, M. F. Syahputra, and B. Siregar. 2019. "The English Language Scientific Literature Classification Based on Abstract Using Rocchio Algorithm." in *Journal of Physics: Conference Series*. Vol. 1235. Institute of Physics Publishing.

Ahmad, Arif. 2018. *PENDETEKSI BAHASA DAERAH PADA TWITTER DENGAN MACHINE LEARNING*.

Hassan, Sayar Ul, Jameel Ahamed, and Khaleel Ahmad. 2022. "Analytics of Machine Learning-Based Algorithms for Text Classification." *Sustainable Operations and Computers* 3:238–48. doi: 10.1016/J.SUSOC.2022.03.001.

Jauhiainen, Tommi, Krister Lindén, and Heidi Jauhiainen. 2017. *Evaluation of Language Identification Methods Using 285 Languages*.

Jauhiainen, Tommi, Marco Lui, Marcos Zampieri, Timothy Baldwin, and Krister Lindén. 2019. *Automatic Language Identification in Texts: A Survey*. Vol. 65.

Kowsari, Kamran, Kiana Jafari Meimandi, Mojtaba Heidarysafa, Sanjana Mendu, Laura Barnes, and Donald Brown. 2019. "Text Classification Algorithms: A Survey." *Information (Switzerland)* 10(4).

Rocchio, J. J. 1971. "Relevance Feedback in Information Retrieval." *The Smart Retrieval System - Experiments in Automatic Document Processing* 313–23.

Shah, Kanish, Henil Patel, Devanshi Sanghvi, and Manan Shah. 2020. "A Comparative Analysis of Logistic Regression, Random Forest and KNN Models for the Text Classification." *Augmented Human Research 2020 5:1* 5(1):1–16. doi: 10.1007/S41133-020-00032-0.

Tondo, Fanny Henry. 2009. *KEPUNAHAN BAHASA-BAHASA DAERAH: FAKTOR PENYEBAB DAN IMPLIKASI ETNOLINGUISTIS 1*. Vol. 11.

Tuhenay, Deglorians, and Evangs Mailoa. 2021. "PERBANDINGAN KLASIFIKASI BAHASA MENGGUNAKAN METODE NAÏVE BAYES CLASSIFIER (NBC) DAN SUPPORT VECTOR

MACHINE (SVM).” *Jurnal Informatika Dan Komputer) Akreditasi KEMENRISTEKDIKTI* 4(2). doi: 10.33387/jiko.

Vatanen, Tommi, Jaakko, J Väyrynen, and Sami Virpioja. 2010. *Language Identification of Short Text Segments with N-Gram Models*.

Wang, Dongyang, Junli Su, and Hongbin Yu. 2020. “Feature Extraction and Analysis of Natural Language Processing for Deep Learning English Language.” *IEEE Access* 8:46335–45. doi: 10.1109/ACCESS.2020.2974101.

Winata, Genta Indra, Alham Fikri Aji, Samuel Cahyawijaya, Rahmad Mahendra, Fajri Koto, Ade Romadhony, Kemal Kurniawan, David Moeljadi, Radityo Eko Prasojo, Pascale Fung, Timothy Baldwin, Jey, Han Lau, Rico Sennrich, and Sebastian Ruder. 2022. “NusaX: Multilingual Parallel Sentiment Dataset for 10 Indonesian Local Languages.”

Wiratno, Tri, and Riyadi Santosa. 2014. *Bahasa, Fungsi Bahasa, Dan Konteks Sosial*.

Zaman, Badrus, Eva Hariyanti, and Endah Purwanti. 2015. *Sistem Deteksi Bahasa Pada Dokumen Menggunakan N-Gram*.