FUDMA Journal of Sciences (FJS)
ISSN online: 2616-1370
ISSN print: 2645 - 2944
Vol. 10 No. 10, May, 2026, pp 219 – 225
DOI: https://doi.org/10.33003/fjs-2026-1010-5074

Enhancing Hausa Words Lemmatization through Feature Engineering

***[1]Adamu Muhammad, [2]Rasheed Abubakar Rasheed and [1]Muhammed Yusuf Muhammed**

[1]Department of Computer Science, Faculty of Computing and Information Technology (FCIT), Bayero University, Kano State, Nigeria.
[2]School of Computer Science, Taylor`s University Lakeside Campus, Subang Jaya, Malaysia.

*Corresponding authors' email: am.makarfi88@gmail.com

ABSTRACT

Hausa is spoken by over 50 million people across Africa, yet it remains critically under-resourced in natural language processing (NLP), particularly for lemmatization. The language's rich morphology characterized by agglutination, internal vowel alternation, and extensive affixation poses significant challenges for existing rule-based and conventional machine learning approaches. This study addresses this gap by developing and evaluating supervised machine learning models for Hausa word lemmatization. We constructed a manually annotated dataset comprising 4,530 unique word-lemma pairs extracted from diverse media sources, achieving a high Inter-Annotator Agreement of 91.10%. Two baseline algorithms, Support Vector Machine (SVM) and Random Forest, were trained and optimized using GridSearchCV on an 80/20 train-test split. The study introduces an enhanced feature engineering framework that integrates phonological attributes, morphological markers, syllable counts, gemination flags, and extended character n-grams (1-5) alongside traditional surface-level features. Experimental results demonstrate that the Random Forest classifier consistently outperforms SVM. When paired with the enhanced feature set, Random Forest achieved the highest performance metrics, recording an accuracy of 64.02% and a weighted F1-score of 0.6134. Feature importance analysis further confirms that linguistically informed attributes significantly improve model generalization and prediction accuracy. These findings underscore the critical role of domain specific feature design in overcoming data scarcity and linguistic complexity. The curated dataset and optimized modeling framework provide a foundational resource to advance downstream Hausa NLP applications, including information retrieval, machine translation, and computational linguistics.

Keywords: Hausa NLP, Lemmatization, Machine Learning, Random Forest, Support Vector Machine, Feature Engineering

INTRODUCTION

Machine learning (ML) enables computers to learn from data and adapt without explicit programming (Singh & Kaur, 2020; Chatterjee, 2021), driving significant advancements in Natural Language Processing (NLP) (Zakari et al., 2021). A critical Natural Language Processing task is lemmatization, which reduces inflected words to their canonical dictionary forms to facilitate information retrieval, semantic analysis, and machine translation (Nuţu, 2021; Alhakim & Abbas, 2018; Akhmetov, Pak, et al., 2020). While Natural Language Processing tools have advanced rapidly for high-resource languages, Hausa a widely spoken Chadic language with over 50 million speakers across Africa (Tukur et al., 2020; Abdulmumin et al., 2022) remains significantly under-resourced (Adeyemi et al., 2021).

The primary challenge in Hausa Natural Language Processing is the language's rich and complex morphology, characterized by agglutination, internal vowel alternation, cliticization, and excessive affixation (Akhmetov, Krassovitsky, et al., 2020; Kanerva et al., 2021). Traditional rule-based lemmatizers are labor-intensive, difficult to scale, and fail to generalize due to these morphological complexities (Ibrahim et al., 2018). Furthermore, existing computational methods often rely on simplistic feature sets that cannot adequately model complex morphological phenomena such as gemination and reduplication (Kanerva et al., 2021). Although recent studies have introduced classical Machine Learning models for Hausa lemmatization, they are limited by inadequate preprocessing and basic feature engineering (Muhammad & Rasheed, 2025). This highlights a critical gap and the urgent need for robust, data-driven approaches utilizing linguistically informed feature representations.

To address these critical gaps, this research aims to curate a manually annotated Hausa word-lemma dataset and develop supervised Machine Learning lemmatization models; specifically Support Vector Machine (SVM) and Random Forest (RF) that leverage enhanced, linguistically informed feature sets. . By curating a manually annotated dataset and engineering an "Enhanced Feature Set" that incorporates linguistically informed attributes such as vowel ratios, syllable counts, and extended character n-grams, this research aims to provide a robust, scalable, and accurate foundation for Hausa Natural Language Processing applications, thereby mitigating the digital language divide.

Related Work

Research on lemmatization has been conducted across various languages, employing different techniques. (Freihat et al, 2018) presented an optimization approach for Arabic lemmatization combining machine learning and dictionary lookup, achieving 98% accuracy. Similarly, Md. Kowsher et al. (2019) proposed a Bengali Information Retrieval System using lemmatization and TF-IDF, achieving 97.22% accuracy.

In the context of African and Asian languages, Akhmetov, Pak, et al. (2020) presented an open-source language-independent lemmatizer based on the Random Forest classification model, showing good potential for Asian and African languages. (Tukur et al., 2020) presented a novel POS model for tagging Hausa sentences using Hidden Markov Model (HMM), achieving an average accuracy of 76.795%. More recently, Islam et al., (2022) BaNeL, an encoder-decoder based Bangla neural lemmatizer, achieving 95.75% accuracy.

However, specific gaps remain for Hausa. (Abdi & Abdullahi, 2023) Proposed a lexicon and rule-based lemmatizer for Somali, highlighting the need for similar tools for Hausa. Table 1 summarizes key related works.

Table 1: Summary of Literature Reviewed

Author(s)	Language	Method	Accuracy/Performance
(Abdi & Abdullahi, 2023)	Somali	Rule-based	95.87%
Freihat et al. (2018)	Arabic	ML + Dictionary	98%
Md. Kowsher et al. (2019)	Bengali	Hobbs' Algorithm	97.22%
Akhmetov, Pak, et al. (2020)	Multi-language	Random Forest	0.8405 (Weighted Avg)
Tukur et al. (2020)	Hausa	HMM (POS Tagging)	76.795% (Avg)
Islam et al. (2022)	Bangla	Neural Encoder-Decoder	95.75%
Adamu & Rasheed (2025)	Hausa-Based Language	Basic text preprocessing (trigrams, word length, binary affix flags).	RF: 63.25% Acc, 60.26% F1 SVM: 56.73% Acc, 54.75% F1

Despite the reviewed advancements in lemmatization for various global languages, significant gaps remain in the context of Hausa NLP. First, there is an absence of publicly available, manually annotated Hausa word-lemma datasets specifically designed for lemmatization tasks. Second, there is insufficient investigation into the feature importance of Hausa morphology, such as character n-grams, reduplication detection, and gemination identification. Finally, there is a lack of comparative analysis between different feature engineering strategies for Hausa lemmatization. This study seeks to bridge these gaps by curating a novel dataset and engineering an enhanced, linguistically informed feature set to optimize classical machine learning models for Hausa word lemmatization.

MATERIALS AND METHODS

This study adopts a direct supervised approach to building word-lemma classification models. The methodology involves dataset organization, model training, and performance evaluation.

Data Collection and Annotation

To train robust models, a dataset of 4,530 unique inflected words_form with 4,127 unique lemmas was assembled. Sources included BBC Hausa (28 articles), VOA Hausa (32 articles), Leadership Hausa (12 articles), DW (16 articles), BUK FM (8 articles), and Facebook posts (17). These articles covered genres such as politics, health, sports, and culture from January to April 2025.

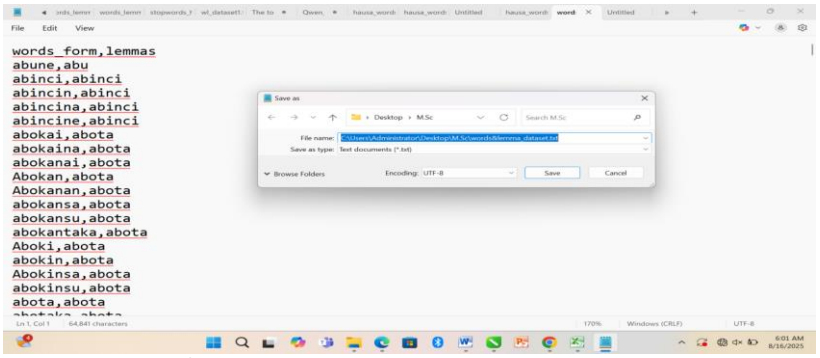

Figure 1: Sample of Dataset

Lemmas were provided by native speakers and validated by Hausa linguistics professionals. The annotation process involved preparation of guidelines, manual annotation using Microsoft Excel, and calculation of Inter-Annotator Agreement (IAA). The dataset achieved an IAA of 91.10%, indicating high reliability (Table 2).

Inter Annotator Agreement

The IAA of the dataset was 91%, which is a high agreement level. This highlights that the annotation guidelines were well defined, which assisted annotators in distinguishing between various levels in the proposed dataset. In addition, this also showed that the annotators were well trained and had expertise in the relevant field. In the proposed dataset, there were 4,530 instances requiring agreement.

Table 2: Inter-Annotator Agreement of the Dataset

Data Source	Total Words	Inflected Words	Same Annotators	IAA (%)
BBC Hausa	6,230	1,220	1,022	83.77%
VOA Hausa	5,345	1,103	1,088	98.64%
Leadership	4,367	697	548	78.62%
Total	24,154	4,530	4,127	91.10%

Data Preprocessing

Text cleaning is a method to clean the text data and enable it to feed data to the model, it is the process of applying any sort of computation to shapeless raw data to change them into a format that can be handled more quickly and efficiently in another procedure (Bari et al., 2025). It ensures that morphological analyzers operate on consistent, normalized input, especially important given the orthographic and syntactic variability in real world Hausa text. Recent research emphasizes integrated pipelines that combine cleaning and

linguistic rules to improve NLP performance for low resource African languages. Key steps in text cleaning for Hausa include: Normalization of character (e.g., normalizing diacritic characters like; ‘ɓ’, ‘ɗ’, ‘ƙ’, ‘ŋ’, ‘š’, ‘z’), Remove non-Hausa characters and noise (removing or normalized emojis, hashtags, URLs, and mixed scripts of Arabic or English) to isolate pure hausa content, Spelling normalization (standardized the variation of writing such as “na gani” vs “nà gàni”)

Feature Extraction

In order to build a robust feature set, we use combine multiple feature types into a single feature matrix. When developing machine learning models for tasks like Hausa lemmatization, it's rare that a single type of feature will capture all the necessary patterns needed for accurate predictions. To improve model performance, we often combine multiple types of features into a single input representation. Combining multiple feature types means taking different kind of information from the input (specifically, character n-grams, word length, suffixes) and merging them into unified feature vector that the model can use to make predictions and character level encoding holds structural information of words enabling the model to correctly predict new unseen lemma if the type of the inflection is familiar (Islam et al., 2022). Some of the benefits of combining multiple features are not limited to; improved accuracy, better generalization, handling sparsity, and more robust to noise. The extract_features function in this study is responsible to translate each word into numerical features that the models can be able to understand. Two sets of features were design in order to translate words into numerical features (initial and advanced features).

Existing Features

These features represent a more traditional and general approach to word analysis and provides basic structural information and some common Hausa morphological markers that includes:

Word Length

(The total number of characters in the word)

Reduplication Flag (is_reduplicated)

a binary indicator if a word shows simple reduplication (e.g., the last two characters repeating the previous two, like 'keke').

Binary Prefix/Suffix Flags (has_prefix_X, has_suffix_X)

a predefined, limited set of prefixes and suffixes, this feature simply indicates whether the word starts or ends with one of them.

Trigrams (trigram_XYZ)

all contiguous 3-character sequences within the word that capture local character patterns.

Enhanced Features

This set expands significantly on the existing features, providing a much richer and more nuanced representation of the word's internal structure. It includes:

Vowel Ratio (vowel_ratio);

the proportion of vowels to the total length, this can be an indicator of word type or structure.

Syllable Count (syllables)

an estimated count of syllables, derived from vowel count. Syllable structure is a fundamental aspect of phonology and morphology.

Gemination Flag (geminated)

A binary indicator if a word contains any repeated adjacent characters (e.g., 'bb', 'ss', 'kk'). Gemination is a common phonological process in Hausa and often marks grammatical distinctions.

Character N-grams (1 to 5)

This is a significant expansion over just trigrams. It includes all contiguous character sequences from unigrams (single characters) up to 5-grams. This captures much broader and finer-grained character patterns, which are highly indicative of morphological structure.

Models and Algorithms

The aforementioned machine learning algorithms were taken into consideration due to their classification ability, suitability as baseline models for lemmatization function for Hausa-based words, and supervised learning nature, which necessitates labeled data for learning during model training the models for effective predictions of words and its lemmas. Support vector machine and random forest are mostly effective for binary classification exercises. Rule-based approach is also considered to provide comparative insights into the performance of all three algorithms

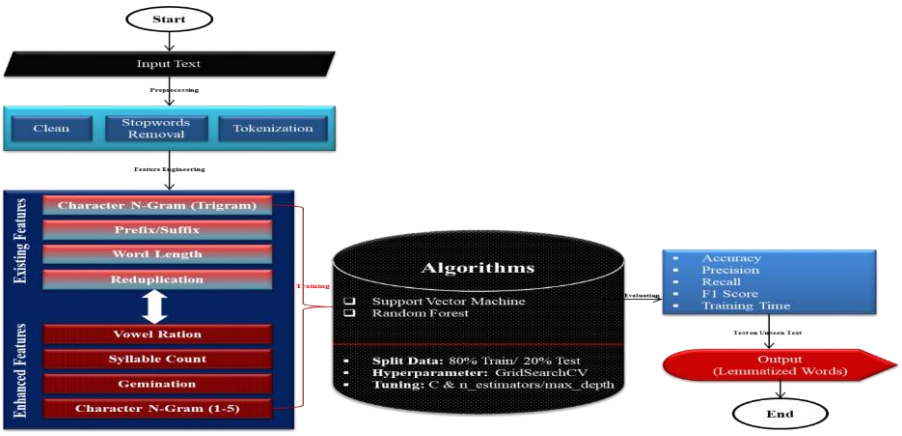

Figure 2: Model Architecture

Support Vector Machine (SVM): An SVM classifier is used to find an optimal hyperplane separating different classes of data (lemmas) in the feature space. An optimal hyperplane is the one that separates the data with the largest margin (the distance between the hyperplane and the closest data point in each class). Given the feature matrix X and the numerical label y, the SVC (Support Vector Classifier) tries to solve an optimization problem where the weights (w) and bias (b) are such that for every training data point xi, the quantity $w \cdot xi + b$ has a large margin from the separating hyperplane. The class of data point xi is determined by the sign of $w \cdot xi + b$. If a non-linear kernel (such as an RBF kernel) is used instead of the linear kernel, then SVM employs the "kernel trick." The kernel trick allows us to compute the dot product between data points in higher dimensions without actually mapping those data points into that higher-dimensional space (which may become computationally expensive); this calculation can be directly computed from the coordinates in the original feature space. As a result, the decision boundaries are non-linear in the original feature space. SVM is used with sklearn.svm.SVC (Support Vector Classifier) and a linear kernel. The model seeks to find an optimal separating hyperplane for different classes of data. The parameter C of the SVM model is tuned by means of GridSearchCV to balance the training error and generalization. The space searched for C was the array [0.1, 1, 10]. Works well on high-dimensional space. The linear kernel is employed and C is tuned using GridSearchCV.

Random Forest (RF) a robust ensemble method to avoid overfitting, the multiple trees are created while the training phase. A random subset of the training set (bootstrapping) is drawn, and only a random subset of the features are considered when making split at each node. This randomization helps to overcome overfitting and enhance generalization ability of the model. Finally the class output is majority voting on the prediction of individual tree, and for regression the output is mean. The RandomForestClassifier builds n estimators decision trees and for each tree: the bootstrap sample of training data is drawn (sampling with replacement), at each node of the tree, instead of using all features to find best split, only a random subset of features are used. Tree grown until stopping criteria reached (e.g. Max depth or minimum samples per leaf). Using Random Forest model uses sklearn.ensemble.

RandomForestClassifier, Random Forests are ensemble learning methods that construct a multitude of decision trees during training and output the class that is the mode of the classes (classification) or mean prediction (regression) of the individual trees. They are known for their robustness and ability to handle high dimensional data. For hyperparameter tuning, use GridSearchCV method to find the best nestimators and maxdepth. The values for nestimators are in [50, 100, 200], max depth is in [10, 15, 20, None] (None means the nodes are expanded until all leaves are pure or contain less than min samples split samples). Njobs =-1 parameter utilizes all available CPU cores to speed up the computation. Optimized for n_estimators and max _depth. Training set is 80% and testing set is 20% of data. Used 3-fold cross validation with GridSearchCV to optimize F1-weighted score.

Hyperparameter Tuning

Even though recent studies show that linear SVM with optimized C (e.g., C = 1.0 or 10.0) consistently outperforms other configurations in sentiment analysis and spam detection tasks. In order to improve the accuracy of the models for this research, instead of training the models with default parameters, GridSearchCV is used. GridSearchCV systematically tries out different combinations of hyperparameters (like C and kernel for SVM, and n_es timators and max_depth for Random Forest) defined in the param_grids. It then performs a cross-validation on each parameter combination (that splits the training set into smaller folds) to get the model's performance. The purpose here is to find the hyper-parameters which leads to the optimal performance (in this case measured using the f1_weighted score) on the training data. After finding the best hyper-parameters, the GridSearchCV trains the model on the complete training set with the optimal hyper-parameters. In NLP, GridSearchCV or random search combined with cross validation is typically used to find optimal *C* and kernel settings. The parameter C controls the trade-off between maximizing margin and minimizing training errors. Small C value is more forgiving of the training error while aiming for large margin; large C value imposes harsher penalties for training error, hence may result in smaller margin but fewer errors during training. N estimators (number of trees) and max depth (maximum tree depth) affect the size and complexity of the forest. More trees and deeper trees can potentially capture more complex patterns but also increase the risk of overfitting and computation time of the models. While the relative performance between feature sets and models is clear, hyperparameter tuning played a crucial role in maximizing the potential of each model by finding the C and n_estimators/ max_depth values that resulted in the best F1-scores.

Evaluation Metrics

Models were evaluated using Accuracy, Precision, Recall, F1-Score (weighted), and Training Time.

RESULTS AND DISCUSSION

A core finding of this research is the significant impact of feature engineering on model performance. The transition from "Existing Features" to "Enhanced Features" yielded consistent improvements across both SVM and Random Forest models. SVM Improve the F1-Score increased from 0.5475 (Existing) to 0.6069 (Enhanced), a gain of approximately 5.94 percentage points and Random Forest Improve the F1-Score increased from 0.6026 (Existing) to 0.6134 (Enhanced). The enhanced features incorporated linguistically informed attributes such as vowel ratio, syllable count, gemination flags, and extended character n-grams (1-5 grams). When the importance of each feature was evaluated, word length was the leading feature in the previously designed features but phonological and morphological features (vowel ratio, prefix count, last char) had much higher importance value in the proposed feature set. So the proposed feature set clearly demonstrates the importance of exploiting the structural and phonological information of Hausa morphology for better lemmatization.

The comparative analysis between Support Vector Machines (SVM) and Random Forest (RF) demonstrated the superiority of ensemble methods for this task. Random Forest (64.02%) outperformed SVM (62.69%) when using enhanced features in term of accuracy and efficiency in computations (training time) compared to SVM, particularly when handling the high-dimensional feature space created by the enhanced features and also Random Forest showed greater robustness to noise and overfitting, likely due to its ensemble nature which aggregates predictions from multiple decision trees.

The SVM model, while effective in high-dimensional spaces, required more stringent hyperparameter tuning (specifically the C parameter) and exhibited longer training times. The optimal C value shifted from 10 (Existing) to 1 (Enhanced),

suggesting that with richer features, the model required less regularization to generalize well.

Model Performance

The performance of SVM and Random Forest using both feature sets is presented in Table 3.

Table 3: Overall Model Comparison

Method	Accuracy	Precision	Recall	F1-Score	Time (s)
Existing Features + SVM	0.5673	0.5585	0.5673	0.5475	306.48
Existing Features + RF	0.6325	0.6041	0.6325	0.6026	196.76
Enhanced Features + SVM	0.6269	0.6139	0.6269	0.6069	334.71
Enhanced Features + RF	0.6402	0.6144	0.6402	0.6134	336.14

The Random Forest model consistently produced better results than SVM on both feature sets. On the enhanced feature set with the RF model, the best F1-Score was 0.6134. Transferring from the Existing to the Enhanced feature set provided a steady gain for both: F1-Score improved by roughly 5.94% for SVM and 1.08% for RF. Even though the additional features made training time higher by increasing the dimensionality, the improvement made it worth it.

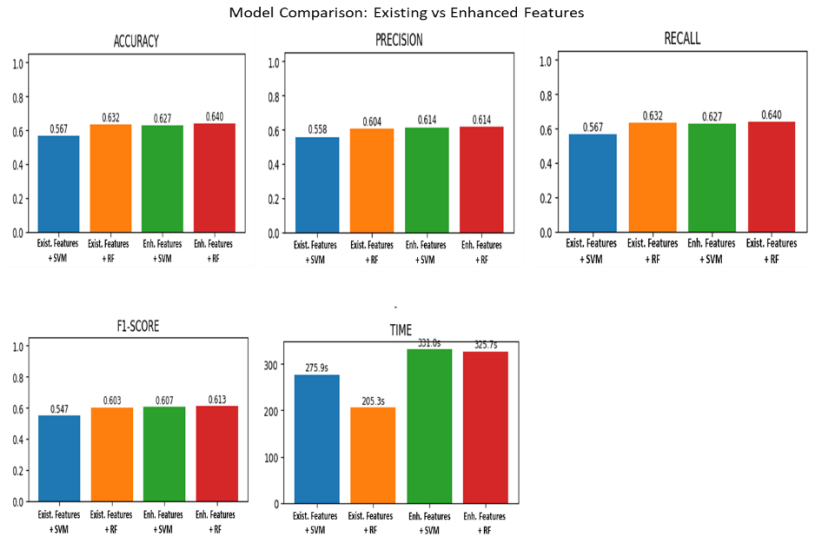

Figure 3: Models/ Features Comparison

Figure 3 provides a comprehensive visual comparison of the performance metrics across all four model configurations (Existing and Enhanced Features for both SVM and Random Forest). Each bar chart in the figure represents a specific metric (Accuracy, Precision, Recall, F1-Score, and Time), allowing for a direct side-by-side assessment of how each approach performs. As illustrated in Figure 3, the transition to Enhanced Features yielded visible improvements across all metrics for both algorithms. Furthermore, the figure clearly demonstrates the dominance of the Random Forest models, which consistently show taller bars in the performance metrics compared to SVM, while also highlighting the computational trade-off in the Training Time graph.

Feature Importance

Feature importance analysis revealed distinct patterns between the two sets. In the Enhanced Features model, linguistic features dominated, prioritizing phonological and morphological attributes (vowel ratio, syllables) and boundary characters (last_char, first_char). In the Existing Features model, word length had ~4x higher importance, relying more on surface-level patterns like trigrams and binary affix flags. This confirms that capturing structural and phonological nuances is crucial for accurate Hausa lemmatization.

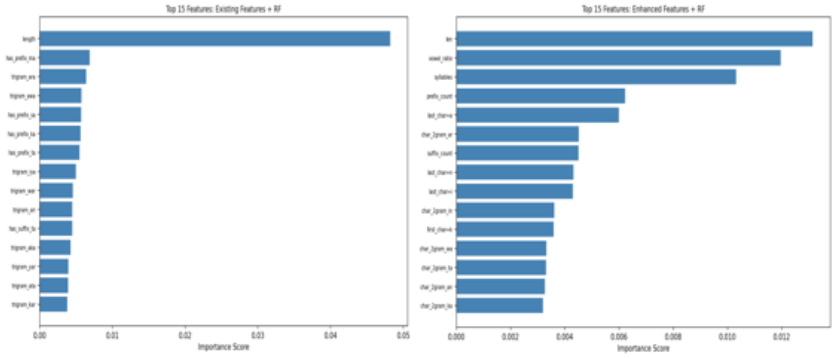

Figure 4: Top 15 Existing and Enhanced Random Forest Features

Figure 4 presents a visual comparison of the top 15 feature importance scores derived from the Random Forest models trained on both the Existing and Enhanced feature sets. These charts illustrate how the model prioritizes different linguistic attributes when classifying Hausa words.

As depicted in Figure 4, the two feature sets reveal distinct patterns. For the Enhanced Features (right chart), linguistic features dominate; the model prioritizes phonological and morphological attributes such as vowel_ratio and syllables. Additionally, character-level patterns (e.g., char_2gram_ar, char_2gram_in) and boundary characters (last_char = a, first_char = k) are highly ranked, along with affix counts (prefix_count, suffix_count).

Similarly, for the Existing Features (left chart), the length feature has approximately 4x higher importance than in the enhanced model. The model relies heavily on surface-level patterns, utilizing multiple binary affix flags (e.g., has_prefix_ma, has_prefix_sa) and trigram patterns (e.g., trigram_ara, trigram_awa). This comparison confirms that the enhanced feature set provides a more linguistically informed approach, moving beyond simple character sequences to capture the phonological and morphological properties that drive word formation in Hausa.

Unknown Text Prediction

The trained models were tested on unseen Hausa text. Example predictions are shown in Table 4.

Table 4: Predicting Lemmas on Unknown Text

Words_form	SVM Prediction	Random Forest Prediction
Bayan	baya	baya
Doguwar	dogo	dogo
Tattaunawa	tattauna	tattauna
Jahar	jaha	jaha
Kammala	kammalawa	kammalawa

Random Forest demonstrated better consistency in handling morphological variations in unseen data.

Discussion

The study successfully curated a specialized dataset of 4,530 unique Hausa word-lemma pairs, addressing the critical gap of unavailable annotated resources for Hausa NLP. The dataset achieved an Inter-Annotator Agreement (IAA) of 91.10%, indicating high reliability and consistency in the ground truth labels. This high agreement level validates the annotation guidelines and the expertise of the native speakers and linguists involved. The diversity of sources (BBC, VOA, Leadership, DW, BUK FM, and Facebook) ensures that the model is exposed to various genres including politics, health, sports, and culture, enhancing its generalizability.

A core finding of this research is the significant impact of feature engineering on model performance. The transition from "Existing Features" to "Enhanced Features" yielded consistent improvements across both SVM and Random Forest models. SVM Improve the F1-Score increased from 0.5475 (Existing) to 0.6069 (Enhanced), a gain of approximately 5.94 percentage points and Random Forest Improve the F1-Score increased from 0.6026 (Existing) to 0.6134 (Enhanced). The enhanced features incorporated linguistically informed attributes such as vowel ratio, syllable count, gemination flags, and extended character n-grams (1-5 grams). Feature importance analysis revealed that while word length remained a dominant factor in existing features, the enhanced model prioritized phonological and morphological features (e.g., vowel_ratio, prefix_count, last_char). This confirms that capturing the structural and phonological nuances of Hausa morphology is crucial for accurate lemmatization.

The comparative analysis between Support Vector Machines (SVM) and Random Forest (RF) demonstrated the superiority of ensemble methods for this task. Random Forest (64.02%) outperformed SVM (62.69%) when using enhanced features in term of accuracy and efficiency in computations (training time) compared to SVM, particularly when handling the high-dimensional feature space created by the enhanced features and also Random Forest showed greater robustness to noise and overfitting, likely due to its ensemble nature which aggregates predictions from multiple decision trees.

The SVM model, while effective in high-dimensional spaces, required more stringent hyperparameter tuning (specifically the C parameter) and exhibited longer training times. The optimal C value shifted from 10 (Existing) to 1 (Enhanced), suggesting that with richer features, the model required less regularization to generalize well.

The feature importance analysis provided valuable linguistic insights into Hausa word structure. Prefix and suffix counts were identified as top features, confirming the agglutinative nature of Hausa where affixes significantly alter word forms, the importance of first_char and last_char features highlights the significance of word boundaries in morphological decomposition and the has_hausa_chars feature helped distinguish native Hausa words from loanwords, aiding in specific phonological processing.

The successful implementation of these models demonstrates that supervised machine learning is a viable approach for low-resource, morphologically rich languages like Hausa. The developed lemmatizer can serve as a preprocessing tool for downstream NLP tasks such as Information Retrieval (IR), Machine Translation (MT), and Sentiment Analysis. By reducing morphological variations to base forms, search engines and text analysis tools can achieve higher precision and recall when processing Hausa text. In conclusion, this research validates the hypothesis that enhanced, linguistically-informed feature sets improve machine learning-based lemmatization for Hausa. Random Forest emerged as the optimal algorithm, balancing performance and efficiency. The study provides a foundational framework and dataset for future Hausa NLP research, contributing to the reduction of the digital language divides for African languages.

CONCLUSION

This research successfully addresses a critical gap in Hausa NLP by developing and evaluating machine learning models for lemmatization. This work indicates that a supervised, data-driven machine learning approach may work for a low-resource and morphologically complex language such as Hausa. Changing from the features under "Existing Features" to "Enhanced Features" worked well, as it gave the model slightly more detailed features on which to base predictions. Random Forest emerged as the superior performer, balancing performance and efficiency. The successful implementation

of feature engineering confirms that linguistic patterns in Hausa can be captured numerically and leveraged by ML classifiers.

REFERENCES

Abdi, A., & Abdullahi, M. (2023). Lexicon and rule-based word lemmatization approach for Somali language. *Somali Language Journal*, *3*(1), 12-25.

Abdulmumin, I., Dash, S. R., Dawud, M. A., Parida, S., Muhammad, S. H., Sa'id Ahmad, I., Panda, S., Bojar, O., Galadanci, B. S., & Bello, B. S. (2022). Hausa Visual Genome: A Dataset for Multi-Modal English to Hausa Machine Translation. *2022 Language Resources and Evaluation Conference, LREC 2022, June*, 6471–6479.

Adamu, M., & Rasheed, A. R. (2025). Machine learning models for Hausa-based language (words) lemmatization. *FUDMA Journal of Sciences (FJS)*, *9*(12), 352-357.

Adeyemi, O., et al. (2021). Low-Resource Languages in NLP. *Journal of African AI*.

Akhmetov, I., Pak, A., Ualiyeva, I., & Gelbukh, A. (2020). Highly language-independent word lemmatization using a machine-learning classifier. *Computacion y Sistemas*, 24(3), 1353–1364.

Alhakim, A., & Abbas, M. (2018). Towards an Optimal Solution to Lemmatization in Arabic. *Procedia Computer Science*, 142, 132–140.

Bari, M. A., Umar, H. A., Bello, B. S., & Ahmed, I. S. (2025). *A Rule-Based Model for Stemming Hausa Words*. 51. https://doi.org/10.3390/engproc2025087051

Chatterjee, I. (2021). *Machine Learning and Its Application : A Quick Guide for Beginners* (Issue December). https://doi.org/10.2174/9781681089409121010l

Freihat, A. A., Abbas, M., Bella, G., & Giunchiglia, F. (2018). Towards an Optimal Solution to Lemmatization in Arabic. *Procedia Computer Science*, 142, 132–140.

Islam, M. A., et al. (2022). BaNeL: an encoder-decoder based Bangla neural lemmatizer. *SN Applied Sciences*, 4(5).

Kanerva, J., Ginter, F., & Salakoski, T. (2021). Universal Lemmatizer: A sequence-To-sequence model for lemmatizing Universal Dependencies treebanks. *Natural Language Engineering*, *27*(5), 545–574. https://doi.org/10.1017/S1351324920000224

Khyani, D., Siddhartha, B. S., Niveditha, N. M., & Divya, B. M. (2021). *An interpretation of lemmatization and stemming in natural language processing. Journal of University of Shanghai for Science and Technology*. *22*(10), 350–357.

Md. Kowsher, Hossen, I., & Ahmed, S. (2019). Bengali Information Retrieval System (BIRS). *International Journal on Natural Language Computing*, 8(5), 1–12.

Muthee, M. G., Mutua Makau, & Omamo Amos. (2022). Review of Techniques for Morphological Analysis in Natural Language Processing. *African Journal of Science, Technology and Social Sciences*, *1*(2), 93–103. https://doi.org/10.58506/ajstss.v1i2.11

Nuțu, M. (2021). Deep Learning Approach for Automatic Romanian Lemmatization. *Procedia Computer Science*, 192, 49–58.

Singh, J., & Kaur, R. (2020). Machine learning and its applications: A Review.

Tukur, A., Umar, K., & SA, A. (2020). Parts-of-Speech Tagging of Hausa-Based Texts Using Hidden Markov Model. *Journal of Computer Science*, 6(2), 303–313.

Yelvita, F. S. (2022). No Titleהכי לנגד שבאמת מה את לראות קשה העינים. *הארץ*, *6*(8.5.2017), 2003–2005.

Zakari, R. Y., Lawal, Z. K., & Abdulmumin, I. (2021). A Systematic Literature Review of Hausa Natural Language Processing. *International Journal of Computer and Information Technology*, 10(4), 4–11.