Journal of Information Systems Engineering and Management
2025, 10(9s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Integrating LSTM and Hybrid Methods for Automatic Balinese Script Transcription

Made Sudarma [1*], Ni Wayan Sri Ariyani[2], I Putu Agus Eka Darma Udayana [3], Putu Gede Surya Cipta Nugraha [4]

[1]Department of Engineering Science, Faculty of Engineering, Udayana University, Denpasar, Indonesia*
[2]Department of Engineering Science, Faculty of Engineering, Udayana University, Denpasar, Indonesia
[3]Departement of Informatics, Faculty of Technology and Informatics, Institute Business and Technology Indonesia, Denpasar, Indonesia
[4]Departement of Informatics, Faculty of Technology and Informatics, Institute Business and Technology Indonesia, Denpasar, Indonesia
***Corresponding author**: msudarma@unud.ac.id

ARTICLE INFO

Received: 01 Nov 2024

Revised: 20 Dec 2024

Accepted: 04 Jan 2025

ABSTRACT

The preservation of Balinese script through digitalization faces challenges, particularly in recognizing words with multiple meanings that require different transliterations depending on context. This study developed an artificial intelligence system based on Long Short-Term Memory (LSTM) to detect the meaning of specific words in Balinese sentences and provide accurate recommendations for Balinese script transliteration. Testing showed that the LSTM model achieved an accuracy of 71% in identifying word meanings in new sentences. To enhance accuracy, this research integrated a hybrid method combining Jaro-Winkler and Damerau-Levenshtein as a preprocessing step. This combination successfully increased system accuracy to 94.58%, surpassing the previous approach, which reached 94.3% without LSTM. This integration demonstrates that the hybrid method effectively corrects spelling errors and reduces ambiguities before deep learning processing. These results represent a significant step in preserving Balinese culture through script digitalization, with further potential for development through context-based processing.

Keywords: Transliteration Balinese Script, LSTM, Damerau-Levenshtein, Jaro Winkler, Digital Cultural Preservation

INTRODUCTION

Lontar is one of Bali's cultural heritages, playing a crucial role as a source of literature and knowledge still utilized by Balinese society in daily life [1]. During the Hindu-Buddhist era, lontar not only served as a source of literature but also as a writing medium, predominantly in the Kawi Balinese language [2]. However, over time, the use of lontar in Bali has declined drastically due to the community's limited ability to understand and write in Balinese script [3]. This condition poses a serious challenge to the preservation of lontar, relegating it to merely historical evidence of Bali's cultural civilization. In February 2023, reports from Balinesia.id and observations in Gianyar Regency revealed that many lontar manuscripts have suffered damage, both physically and in their content [4]. Even the collection of lontar at the Bali Provincial Cultural Office shows significant damage, highlighting the urgent need for preservation efforts. The damage is not limited to physical deterioration. For instance, in Batuan Village, Sukawati, lontar manuscripts have become dirty and decayed due to inadequate storage [5]. Lontar, which should be stored in dry places and protected from sunlight, was found in damp areas, leading to mold growth on the surface . This makes the text on the lontar difficult to read, with some parts requiring restoration. In Tenganan Pegringsingan Village, Karangasem, lontar containing important texts on customs and village history were found folded and torn. This damage resulted from using lontar without adequate maintenance, such as lack of coating or routine care [6]. Consequently, many lontar have lost portions of their text, making them difficult to restore. In the Klungkung area, some lontar stored in residents' homes have turned brown and become brittle [7]. This is due to prolonged exposure to kitchen smoke, which settles over time. This exposure causes the lontar to become fragile and prone to breaking upon touch. At the Bali Provincial Cultural Office, some lontar have been translated into books written in Latin script. However, as the original lontar made from palm leaves continues to deteriorate, digital documentation is also critically needed [8]. Digitalization becomes a crucial solution for preserving and reconstructing the contents of lontar by leveraging modern technological advancements. Although various Balinese script transliteration systems have been developed, the results remain inaccurate due to challenges in understanding linguistic context [9]. One of the main challenges is the existence of homonyms—words that have the same spelling but different meanings. For example, the word "asta" in Balinese can mean eight, bone, is, or hand, depending on the context [10]. Traditional rule-based systems or conventional approaches cannot capture these contextual patterns, resulting in less accurate transliterations [11]. Additionally, the sentence structure in lontar is often complex, utilizing intricate Kawi language syntax, which complicates the transliteration process. Existing systems lack the flexibility to handle these variations, necessitating more sophisticated approaches to analyze the linguistic data. To address these challenges, this study proposes the use of Long Short-Term Memory (LSTM), a part of deep learning, as the main solution [12]. LSTM is capable of understanding relationships between words in a sentence through long-term dependency analysis, allowing the system to detect word meanings based on context. By leveraging word embeddings and sequential training mechanisms, LSTM can capture complex linguistic patterns and provide more accurate recommendations for Balinese script transliteration [13]. In its final implementation, LSTM will be combined with hybrid methods such as Jaro-Winkler and Damerau-Levenshtein to build a system that significantly enhances accuracy [14], [15]. This integration not only improves system performance but also opens new opportunities for cultural preservation through more reliable and advanced digitalization of the Balinese script.

METHODS

This study aims to improve the accuracy of transliterating Latin text into Balinese script by integrating the Long Short-Term Memory (LSTM) method from deep learning with a hybrid approach utilizing Jaro-Winkler Distance and Damerau-Levenshtein Distance [16], [17], [18]. The research process is carried out through several stages designed to address challenges in Balinese script transliteration, including the management of homonyms, capitalization, and variations in sentence structure. Below is a detailed explanation of the research stages.

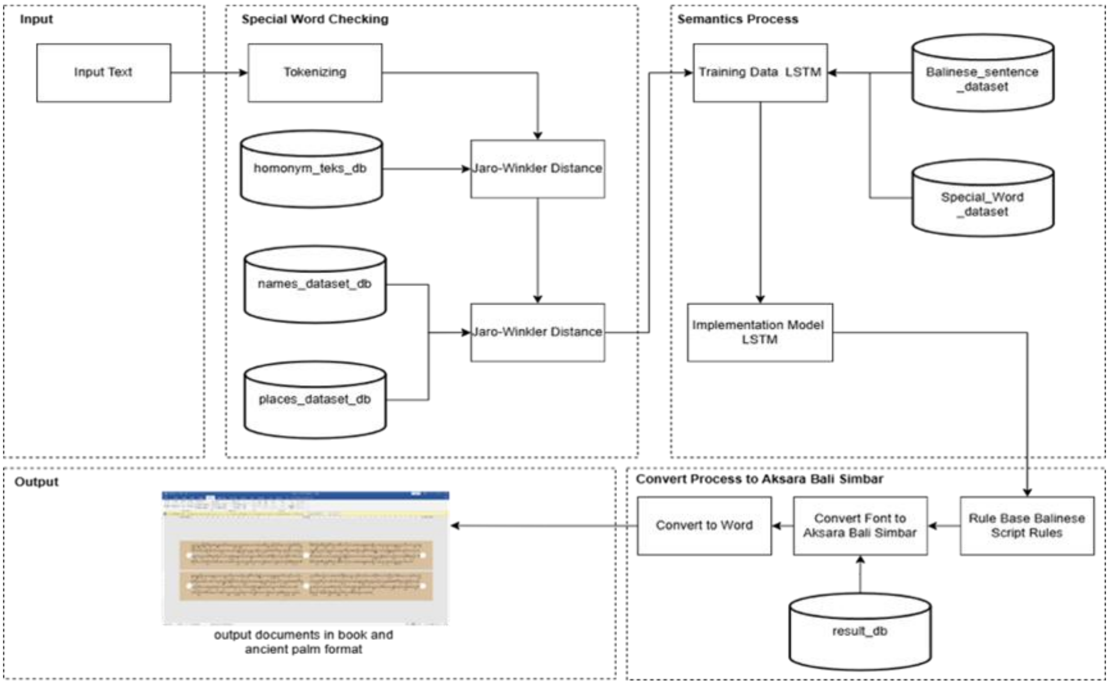


Figure : 1 system workflow

In Figure 1, the workflow explains how the system processes input text, starting with tokenization, which breaks the text into smaller units such as words or phrases. These tokens are then compared to several datasets, such as homonym_teks_db, names_dataset_db, and places_dataset_db, using the Jaro-Winkler Distance algorithm to measure string similarity and identify key elements like homonyms, names, or places. Additionally, the system employs an LSTM (Long Short-Term Memory) model trained with a dataset of Balinese sentences (Balinese_sentence_dataset) and a dataset of special words (Special_Word_dataset) to further process the identified tokens. The processed results are then converted into a more structured text format and translated into Balinese script using rule-based approaches (Rule-Based Balinese Script Rules) and the Aksara Bali Simbar font. Finally, the system's output is stored in a database (result_db) for further use or visualization. This system combines machine learning with rule-based methods to generate accurate transliteration in accordance with Balinese script grammar.

Data collection

This study uses the lontar manuscript "Babad Blahbatuh" as the primary data source. This manuscript consists of 200 lontar leaves containing various traditional literary texts in the Kawi Balinese language, rich in historical and linguistic content. An example of the lontar manuscript *Babad Blabatuh* can be seen in Figure 2.

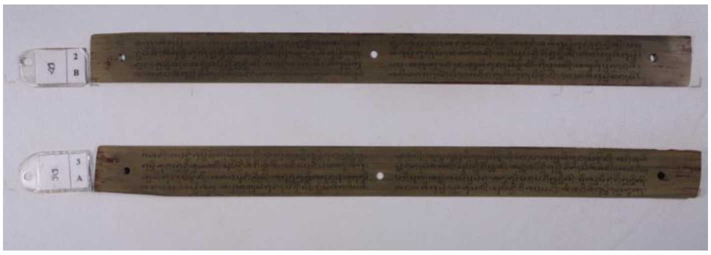


Figure 2: the babad blahbatuh lontar manuscript

In addition, this study focuses on the analysis of 151 specific words in the Balinese language [19], [20]. These words were selected due to their variations in meaning and writing forms in Balinese script, such as the word "*asta*," which can mean eight, bone, or hand, depending on the context. Examples of the usage of "*asta*" with different meanings in Balinese are:

1. *Tiang medue pianak asta diri* (*asta in this sentence means eight*).
2. *Ring usadha Bali, asta dados dasar kaperiksa* (*asta in this sentence means bone*).

In this study, each specific word is composed into 300 Balinese sentences reflecting various contexts of its usage, resulting in a total of 45,300 sentences in the research dataset.

Table 1: Examples of specific words in Balinese

No	Latin Text	Word Meaning	Balinese Script Writing
1	*asta*	eight	ᬅᬱ᭄ᬝ᭟
2	*asta*	is	ᬅᬲ᭄ᬢ᭟
3	*asta*	bone	ᬅᬲ᭄ᬝ᭟
4	*asta*	hand	ᬳᬲ᭄ᬢ᭟

The data collection process began with the identification of specific words from the lontar manuscript and other literary sources, reflecting the linguistic complexity of the Balinese language, including homonyms and specialized terms. Subsequently, Balinese sentences were manually composed to ensure a broad and relevant contextual variation. Transliterations from Latin text to Balinese script were manually performed by language experts to ensure accuracy and compliance with Balinese scriptwriting rules. After compilation, the data was verified to ensure quality and completeness, making it optimally usable for training the Long Short-Term Memory (LSTM) model and integrating the hybrid methods of Jaro-Winkler and Damerau-Levenshtein. This dataset forms a critical foundation for enhancing the system's ability to recognize the meanings of specific words and produce Balinese script transliterations that align with the context.

Data preprocessing

In the preprocessing stage, the dataset containing Balinese sentences is prepared for training the Long Short-Term Memory (LSTM) model. The process begins with tokenization, which involves breaking sentences into word units (tokens) to separate specific words from the sentence context [21]. Subsequently, each specific word in the dataset is labeled with its appropriate meaning based on the sentence context. This labeling process ensures that the model can correctly recognize the word's meaning, such as asta, which can mean eight, bone, or hand . Additionally, Bali Simbar

Unicode is assigned to specific words to reflect the differences in Balinese script writing based on their meanings . Assigning Unicode is crucial for helping the system understand the complex rules of Balinese scriptwriting. The preprocessing process also includes data normalization, such as removing irrelevant characters, adjusting grammar, and handling capitalization and punctuation to ensure data consistency and compatibility with the model [22]. To ensure the model understands the semantic relationships between words, each word is represented as numerical vector embeddings, enabling the analysis of contextual relationships in richer dimensions. After completing all the steps, the dataset is verified to ensure its quality and completeness, making it optimally usable for model training. This stage is crucial to ensure that the system can recognize words in Balinese script and produce accurate transliterations.

Development of the LSTM Model

The Long Short-Term Memory (LSTM) model developed in this study is designed to predict the meaning of specific words within the context of Balinese sentences. The model incorporates several key layers, including an embedding layer, a bidirectional LSTM layer, and a dense layer [23]. The dataset used consists of Balinese sentences containing specific words, with each word labeled with its meaning.

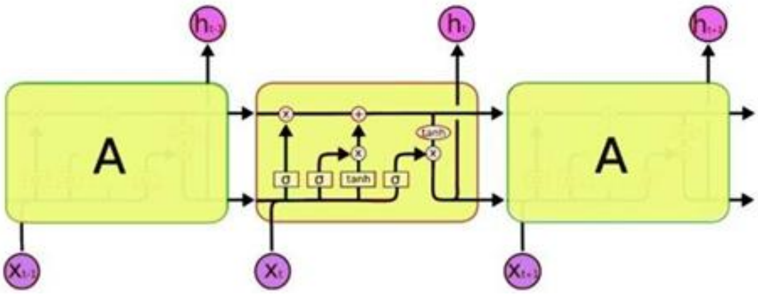


Figure 3: LSTM method process flow

In Figure 3, it illustrates the LSTM network architecture applied in this research. With this architecture, it can handle sequential data with long-term dependencies, such as text or time-series signals, more effectively. LSTM enables the model to "remember" relevant information from previous time steps, even when the sequence is very long. The explanation of the layers can be described as follows:

Input gate (i_t) functions to select and filter new information that is important to update in the memory. Meanwhile, (σ) is the sigmoid activation function, (W_i) is the weight for the input gate and candidate cell state, (h_{t-1}) is the previous hidden state, (x_t) is the current input, and (b_i) bias untuk input gate. is the bias for the input gate. The mathematical equation for calculating the input gate is as follows:

$$i_t = \sigma(W_i[h_{t-1}, x_t] + b_i) \tag{1}$$

Forget gate (f_t) plays a role in determining how much information from the previous memory needs to be forgotten. The forget gate allows the model to filter out irrelevant information from the previous step. Using similar notation, (W_f) represents the weight of the forget gate, and (b_f) is the bias. The equation is:

$$f_t = \sigma(W_f[h_{t-1}, x_t] + b_f) \tag{2}$$

Next, the output gate (o_t) is responsible for determining how much information from the cell memory will be output as the hidden state. The output gate controls the proportion of information retrieved from the cell memory. The mathematical equation is:

$$o_t = \sigma(W_o[h_{t-1}, x_t] + b_o) \tag{3}$$

The candidate cell memory $(\tilde{C}_t)$ represents new information to be added to the cell memory. This memory is calculated using the hyperbolic tangent activation function to constrain its value between -1 and 1. The equation is:

$$\tilde{C}_t = tanh(W_c[h_{t-1}, x_t] + b_c) \tag{4}$$

The cell memory (C_t) is updated by combining new information obtained from the input gate with the old memory retained by the forget gate. The equation is:

$$C_t = (i_t \times \tilde{C}_t) + (f_t \times C_{t\,-\,1}) \tag{5}$$

Finally, the hidden state (h_t) is generated from the current cell memory combined with the output gate. This hidden state is used as the output representation at the current time step. The equation is:

$$h_t = o_t \times tanh(C_t) \tag{6}$$

Hybrid Method Integration

This study applies a hybrid method to enhance the accuracy of transliterating Latin text into Balinese script by combining the Jaro-Winkler Distance and Damerau-Levenshtein Distance algorithms. Jaro-Winkler Distance is used to measure the similarity between strings by considering the number of matching characters, transpositions, and string lengths [24]. This algorithm assigns additional weight to the similarity at the beginning of the string (prefix), which is particularly relevant for detecting homonyms in the Balinese language [25]. The similarity score ranges from 0 to 1, where 0 indicates very low or no similarity, and 1 represents very high similarity or identical strings.

$$dj = \frac{1}{3}\left(\frac{m}{|s1|} + \frac{m}{|s2|} + \frac{m-t}{m}\right) \tag{7}$$

This calculation consists of three main components: first, the ratio of the number of matching characters to the length of the first string; second, the ratio of the number of matching characters to the length of the second string; and third, the result of subtracting the number of matching characters by the number of transpositions, which is then divided by the number of matching characters. The average value of these three components provides the Jaro distance, where a higher value indicates a greater similarity between the two strings. After calculating the Jaro distance, the next step is to determine the Jaro-Winkler distance using the following formula:

$$d_w = d_j + \left(l * p(1 - d_j)\right) \tag{8}$$

This parameter aims to enhance the accuracy of Jaro distance calculations by adjusting the similarity found at the beginning of the string. This adjustment is considered significant as it can indicate a more meaningful overall similarity score. Additionally, the scaling factor amplifies the influence of prefix length on the final similarity score. Meanwhile, Damerau-Levenshtein Distance is used to calculate the edit distance between two strings by considering four types of operations: substitution, insertion, deletion, and transposition.

$$d(i,j) = min\begin{cases} d(i-1,j) + 1\ (deletion) \\ d(i,j-1) + 1\ (insertion) \\ d(i-1,j-1) + (a_i \neq b_j)(subtitution) \end{cases} \tag{9}$$

Additionaly, $if\ i > 1, j > 1, a_i = b_{j-1}, and\ a_{i-1} = b_j$, we also consider transposition: $d(i,j) = min(d(i,j), d(i-2,j-2) + 1)$

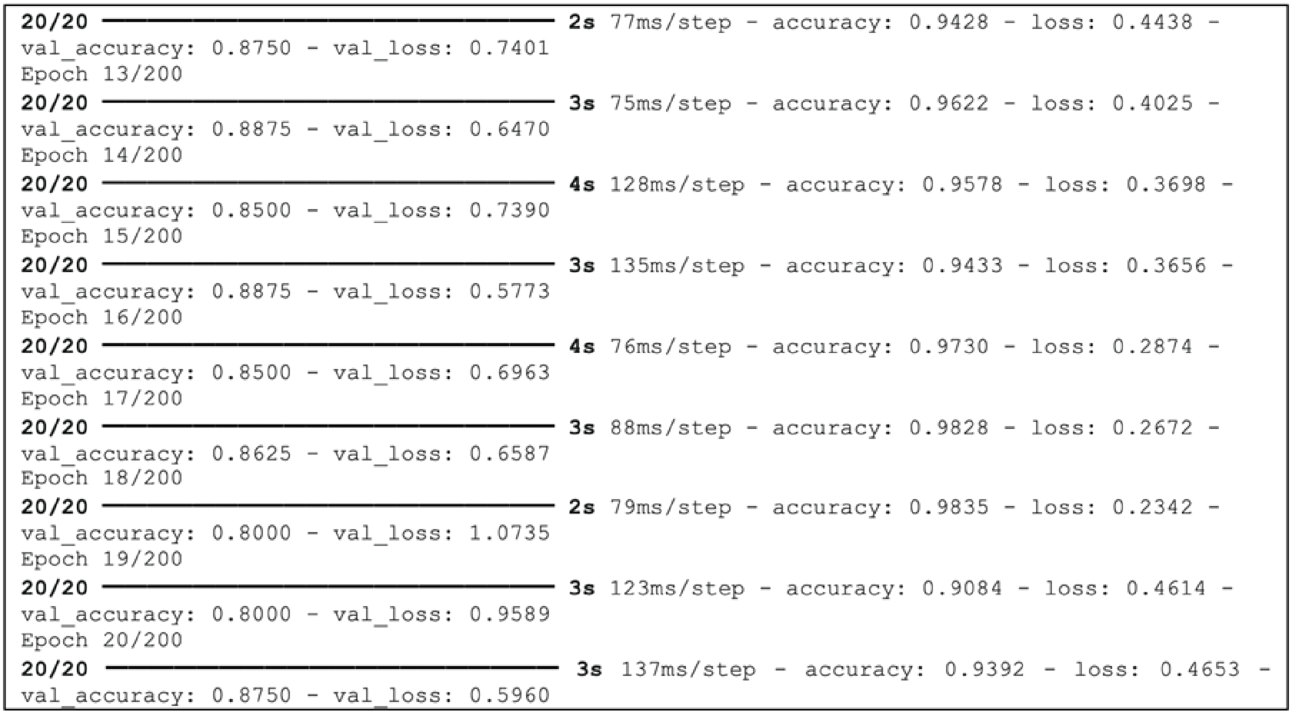
This algorithm helps correct spelling errors, particularly in Latin text, which often experiences character rearrangements or typographical errors. In its implementation, Jaro-Winkler Distance is first used to identify candidate strings with a high similarity score. The results of this process are then passed to Damerau-Levenshtein Distance to correct errors and ensure that the generated text adheres to Balinese script rules [26]

RESULTS AND DISCUSSION

This section presents the results and evaluation of the Balinese script transliteration system, which was developed by combining the Long Short-Term Memory (LSTM) method with a hybrid approach. The discussion focuses on the system's implementation, accuracy testing, and a comparative analysis of the performance of the methods utilized. Additionally, the section explores the impact of the study's findings on the preservation of the Balinese script and highlights its potential for future advancements.

Implementation of LSTM to Recognize Word Meaning in Balinese Sentences

The process begins with tokenizing the text and converting it into numeric representations using a tokenizer. The data sequence length is then adjusted to the maximum length (max_len) by applying a padding method to ensure uniform input dimensions. The model architecture includes an embedding layer with an output dimension of 256, which represents words as vectors that the model can interpret. Two bidirectional LSTM layers, with 128 and 64 units respectively, are utilized to capture both long-term and short-term dependency relationships in the data. These layers are equipped with a 20% dropout rate to mitigate overfitting risks and an L2 regularizer kernel to enhance training stability. This is followed by a dense layer with 128 units and a ReLU activation function to process the output of the LSTM layers. An additional 50% dropout is applied before the final dense layer, which employs a softmax activation function to generate classification probabilities. The model is trained using the Adam optimizer with a learning rate of 0.001 and a categorical cross-entropy loss function. During the training process, each of the 151 special words was presented in 300 unique sentences, resulting in a total of 45,300 sentences used for training the model.



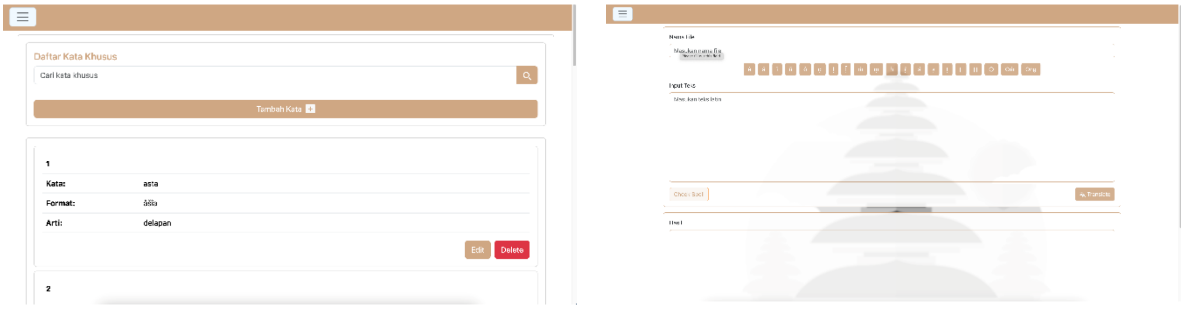
```
20/20 ━━━━━━━━━━━━━━━━━━━━ 2s 77ms/step - accuracy: 0.9428 - loss: 0.4438 -
val_accuracy: 0.8750 - val_loss: 0.7401
Epoch 13/200
20/20 ━━━━━━━━━━━━━━━━━━━━ 3s 75ms/step - accuracy: 0.9622 - loss: 0.4025 -
val_accuracy: 0.8875 - val_loss: 0.6470
Epoch 14/200
20/20 ━━━━━━━━━━━━━━━━━━━━ 4s 128ms/step - accuracy: 0.9578 - loss: 0.3698 -
val_accuracy: 0.8500 - val_loss: 0.7390
Epoch 15/200
20/20 ━━━━━━━━━━━━━━━━━━━━ 3s 135ms/step - accuracy: 0.9433 - loss: 0.3656 -
val_accuracy: 0.8875 - val_loss: 0.5773
Epoch 16/200
20/20 ━━━━━━━━━━━━━━━━━━━━ 4s 76ms/step - accuracy: 0.9730 - loss: 0.2874 -
val_accuracy: 0.8500 - val_loss: 0.6963
Epoch 17/200
20/20 ━━━━━━━━━━━━━━━━━━━━ 3s 88ms/step - accuracy: 0.9828 - loss: 0.2672 -
val_accuracy: 0.8625 - val_loss: 0.6587
Epoch 18/200
20/20 ━━━━━━━━━━━━━━━━━━━━ 2s 79ms/step - accuracy: 0.9835 - loss: 0.2342 -
val_accuracy: 0.8000 - val_loss: 1.0735
Epoch 19/200
20/20 ━━━━━━━━━━━━━━━━━━━━ 3s 123ms/step - accuracy: 0.9084 - loss: 0.4614 -
val_accuracy: 0.8000 - val_loss: 0.9589
Epoch 20/200
20/20 ━━━━━━━━━━━━━━━━━━━━ 3s 137ms/step - accuracy: 0.9392 - loss: 0.4653 -
val_accuracy: 0.8750 - val_loss: 0.5960
```

Figure 4: Results of training the LSTM method to detect the meaning of special words.

The following are the test results for detecting the meaning of special words in Balinese sentences. In this test, each special word, totaling 151 words, was used to create 50 sentences. In the next step, the best LSTM model was used to test the data. Based on the test results, this model achieved an accuracy of 70%.

Implementation

In the implementation of the Balinese script transliteration system, the main steps include data preprocessing, training the Long Short-Term Memory (LSTM) model, and integrating the hybrid methods of Jaro-Winkler Distance and Damerau-Levenshtein Distance. The results of the implementation show that the combination of these methods is effective in handling the linguistic complexity of the Balinese language, especially in recognizing homonyms.



Daftar Kata Khusus

Cari kata khusus

Tambah Kata

1

Kata: asta

Format: ắśta

Arti: delapan

Edit Delete

2

Nama File

Input Teks

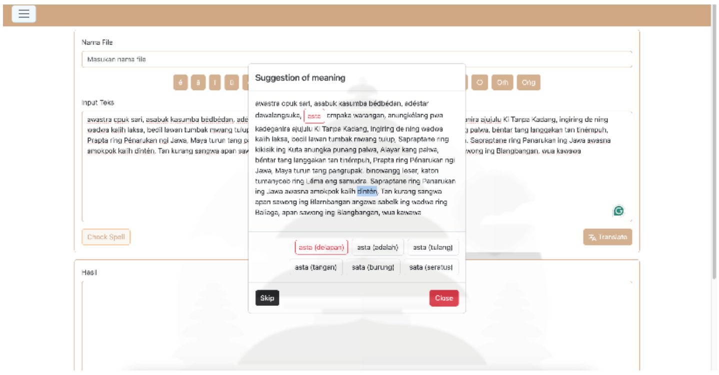
Check Spell

Translate

Hasil

(a) (b)

Figure 5: Transliteration system interface (a) list of special words and (b) form for transliteration process.



Nama File

Masukan nama file

Input Teks

Suggestion of meaning

awastra cpuk sari, asabuk kasumba bédbédan, adéstar dawalangsuka, asta cmpaka warangan, anungkélang pwa kadeganira ajujulu Ki Tanpa Kadang, ingiring de ning wadwa kalih laksa, becil lawan tumbak mwang tulup, Sapraptane ring kikisik ing Kuta anungka punang palwa, Alayar kang palwa, béntar tang langgakan tan tinémpuh, Prapta ring Pénarukan ngi Jawa, Maya turun tang pangrupak. binowangg lesar, katon tumanyceb ring Léma eng samudra. Sapraptane ring Panarukan ing Jawa awasna amokpok kalih dintén. Tan kurang sangwa apan sawong ing Blambangan angawa sabelk ing wadwa ring Baliaga, apan sawong ing Blangbangan, wus kawawa

asta (delapan) asta (adalah) asta (tulang)

asta (tangan) sata (burung) sata (seratus)

Skip Close

Check Spell

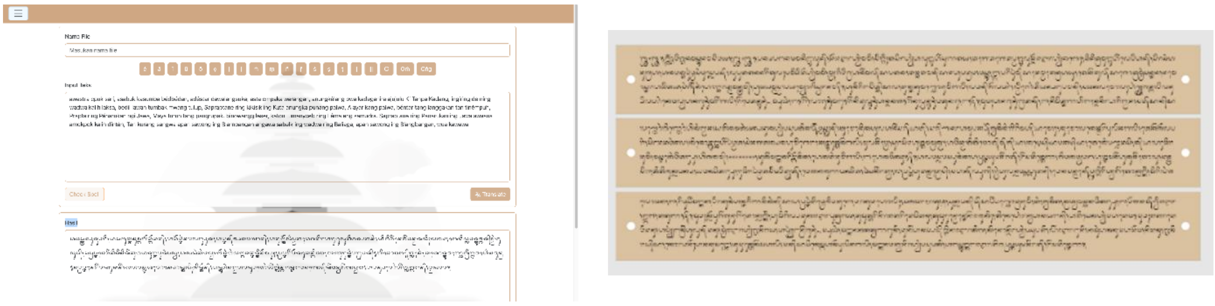
Translate

Hasil

Figure 6: The process of matching words with Jaro-Winkler Distance and Damerau-Levenshtein Distance, as well as providing recommendations for the meaning of words in sentences using LSTM.

Figure 5 illustrates the interface of the transliteration system, which is divided into two main components. The first component (a) displays a list of special words that are critical to the transliteration process, enabling users to review or select specific entries that might require particular attention during transliteration. The second component (b) provides a form designed to facilitate the transliteration process. This form allows users to input text and seamlessly convert it into the desired script format through a user-friendly interface. Meanwhile, Figure 6 demonstrates the process of matching words and providing contextual recommendations. The system employs the Jaro-Winkler Distance and Damerau-Levenshtein Distance algorithms to compare words, identify similarities, and correct potential errors. This ensures a more accurate matching process. Additionally, the system integrates an LSTM model to provide recommendations for the meaning of words within sentences, enhancing the accuracy and contextual relevance of the transliteration output. This combined approach ensures both technical precision and contextual understanding in the transliteration process.

The most important aspect of this system is how the Jaro-Winkler Distance and Damerau-Levenshtein Distance methods work together to detect words that potentially belong to special word categories. Once such a word is identified, the LSTM method determines the correct meaning based on the sentence structure surrounding the word. When the LSTM method identifies the correct meaning, the system selects the appropriate Balinese script writing style to be transcribed in the form of lontar. Essentially, the same word with different meanings will have distinct writing styles, making it crucial to accurately understand the meaning of special words within the sentence.



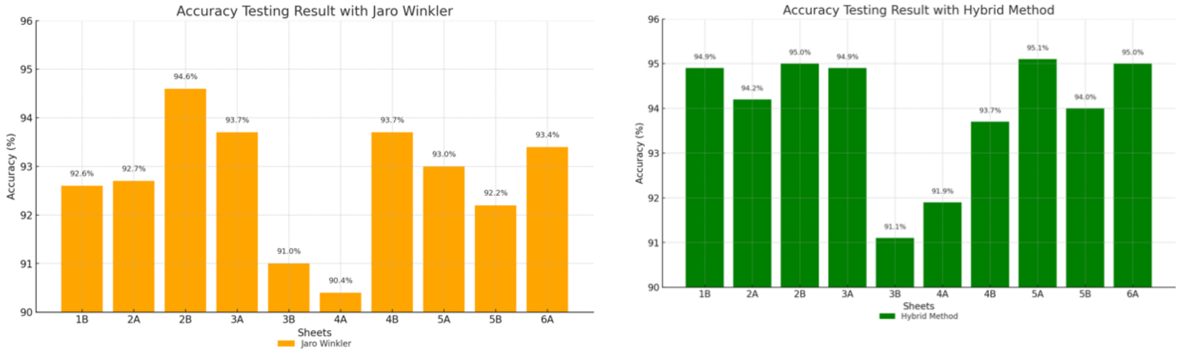
(a) (b)

Figure 7: The results of the transliteration process (a) Balinese script is contained in the system and (b) Balinese script has been saved in lontar format.

Figure 7 shows the results of the transliteration process in two distinct outputs. Part (a) displays the Balinese script generated by the system, showcasing the transliteration results directly within the application's interface. This allows users to verify and review the output before proceeding further. Part (b) highlights the Balinese script that has been saved in *lontar* format, preserving the script in a traditional style suitable for archival or cultural documentation purposes. This two-stage output ensures both digital usability and traditional preservation of the transliteration results. In the final stage of this application, the transliteration results will be produced in the form of Balinese script which can be saved as the original lontar form, but stored in digital form on a Microsoft Word worksheet.

Accuracy Testing with Hybrid Method

In previous research, the researchers implemented a transliteration system that only utilized the Jaro-Winkler Distance method and the hybrid method (an integration of the Jaro-Winkler Distance with the Damerau-Levenshtein Distance).



(a) (b)

Figure 8: Results of measuring the accuracy of the (a) Jaro Winkler Distance method and (b) Hybrid Integration Method of Jaro Winkler Distance and Damerau-Levenshtein Distance

Figure 8 presents the results of measuring the accuracy of different methods used in the transliteration system. Part (a) illustrates the accuracy achieved using the Jaro-Winkler Distance method, which produces an average accuracy of 92.7%. This method, while effective, encounters limitations in addressing variations in sentence structure and complex linguistic contexts. Part (b) shows the accuracy of the Hybrid Integration Method, which combines the Jaro-Winkler Distance and Damerau-Levenshtein Distance algorithms. This hybrid approach improves accuracy to 93.9% by effectively correcting spelling errors and detecting candidate strings with a high degree of similarity. The comparison highlights the benefits of integrating complementary algorithms to enhance the system's performance.

Accuracy Testing with LSTM with Hybrid Method

This section will explain the results of the accuracy test of the implementation of the LSTM model that has been tested to be combined with the hybrid method in the Balinese script transliteration process.

Table 2: Results of accuracy testing LSTM with hybrid method

No	Lontar Code	Total Characters	Correct	Wrong	Accuracy
1	1B	436	413	23	94.72%
2	2A	450	424	26	94.22%
3	2B	426	408	18	95.77%
4	3A	398	376	20	94.47%
5	3B	440	420	20	95.45%
6	4A	420	395	25	94.05%
7	4B	445	420	25	94.38%
8	5A	416	390	26	93.75%
9	5B	439	415	24	94.53%
10	6A	505	477	28	94.46%
Total					94.58%

Based on Table 2, the results of tests conducted on 10 Babad Blahbatuh palm leaf manuscripts, the combination of these methods produced an accuracy of 94.54%, this result indicates that the combination of these methods is superior to previous methods.

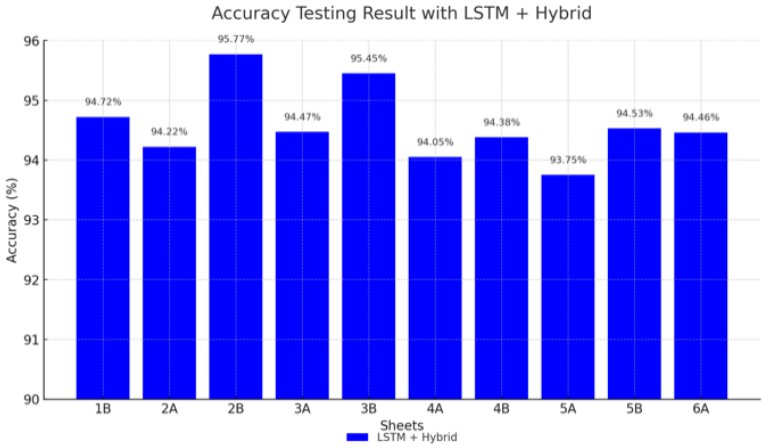


Figure 9: Results of accuracy testing LSTM with the hybrid method

In general, the combination of these methods demonstrates consistent accuracy above 94% in most lontar manuscripts. With performance nearing 95%, the LSTM + Hybrid method shows great potential for implementation in the Balinese script digitization system, supporting the preservation of cultural heritage.

Comparison of Results Method

In this section, the researchers explain the comparison of methods used in the developed system. Based on the comparative accuracy data of the three methods—Jaro-Winkler, Hybrid (a combination of Jaro-Winkler and Damerau-Levenshtein), and LSTM + Hybrid—the LSTM + Hybrid method demonstrates the best performance, achieving the highest average accuracy of 94.58%.

Table 3: Comparative results of accuracy testing each method

No	Lontar Code	Accuracy		
		Jaro Winkler Distance	Hybrid (Jaro + Damerau-Levenshtein Distance)	LSTM+Hybird
1	1B	92.6%	94.9%	94.72%
2	2A	92.7%	94.2%	94.22%
3	2B	94.6%	95.0%	95.77%
4	3A	93.7%	94.9%	94.47%
5	3B	91.0%	91.1%	95.45%
6	4A	90.4%	91.9%	94.05%
7	4B	93.7%	93.7%	94.38%
8	5A	93.0%	95.1%	93.75%
9	5B	92.2%	94.0%	94.53%
10	6A	93.4%	95.0%	94.46%
Total		92.7%	93.9%	94.58%

Based on table 3, The combination of LSTM + Hybrid methods consistently delivers better results compared to other methods across most sheets. The Hybrid method ranks second with an average accuracy of 93.9%, while the Jaro-Winkler method has the lowest average accuracy at 92.7%.

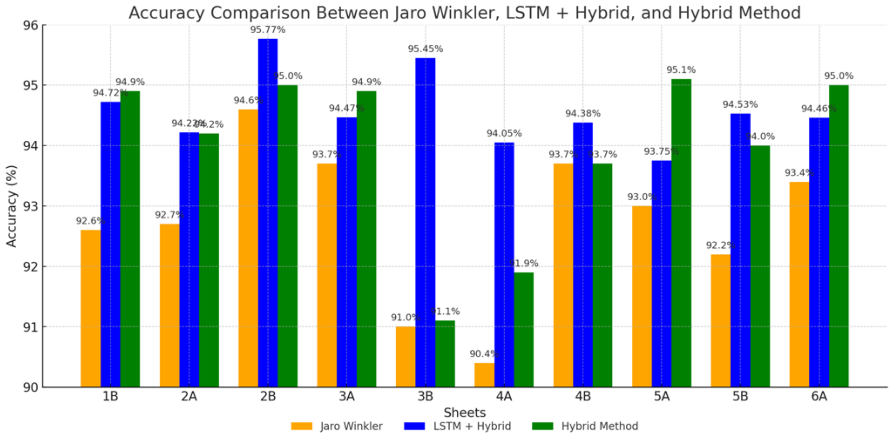


Figure 10: Comparative results of accuracy testing each method

In some manuscripts, such as 3B, the LSTM + Hybrid method demonstrates significantly superior performance, achieving an accuracy of 95.45%, much higher than the Hybrid method (91.1%) and Jaro-Winkler (91.0%). Similarly, manuscript 2B exhibits the best performance across all three methods, with the highest accuracy achieved by the LSTM + Hybrid method at 95.77%. In contrast, for manuscripts such as 5A, the Hybrid method provides the highest accuracy at 95.1%, slightly outperforming the LSTM + Hybrid method, which achieves 93.75%. This method is not only more consistent but also more accurate, making it highly promising for implementation in the lontar digitization process.

CONCLUSION

The conclusion of this research is that the integration of the Long Short-Term Memory (LSTM) method with a hybrid approach (Jaro-Winkler and Damerau-Levenshtein algorithms) has successfully addressed the challenges of Balinese script transliteration by focusing on the complexity of the linguistic context, including homonyms, and achieving an accuracy of 94.58%. The hybrid method plays a crucial role in the preprocessing stage, particularly in providing suggestions for specific words, optimizing the input for the LSTM model. The success of this system opens opportunities for further development, especially in improving transliteration results by considering the diversity of writing rules. Furthermore, the developed approach has the potential to be applied to other regional scripts, supporting the broader preservation and digitization of cultural heritage.

REFRENCES

[1] I. Putu Agus Eka Darma Udayana, M. Sudarma, I. Nyoman Satya Kumara, G. Program, and S. Denpasar, "Balinese Latin Text Becomes Aksara Bali Using Rule Base Method," 2017. [Online]. Available: http://indusedu.org

[2] S. Chakravarthy, "Tokenization for Natural Language Processing | by Srinivas Chakravarthy | Towards Data Science," 2020. [Online]. Available: https://towardsdatascience.com/tokenization-for-natural-language-processing-a179a891bad4

[3] A. A. A. P. Wiratni, "Sentiment Analysis Menggunakan Rule based Method Pada Online Movies Review," 2022.

[4] C. Pramartha, I. B. Ary, I. Iswara, I. Komang, and A. Mogi, "Digital Humanities: Community Participation in the Balinese Language Digital Dictionary," 2020. [Online]. Available: www.basabali.org

[5] C. Pramartha, I. B. Ary Indra Iswara, H. Suputra, and I. B. G. Dwidasmara, "Digital Humanities: Prototype Development for Balinese Script," 2021, pp. 205–214. doi: 10.1007/978-3-030-73043-7_17.

[6] Y. Chaabi and F. Ataa Allah, "Amazigh spell checker using Damerau-Levenshtein algorithm and N-gram," *Journal of King Saud University - Computer and Information Sciences*, vol. 34, no. 8, Part B, pp. 6116–6124, 2022, doi: https://doi.org/10.1016/j.jksuci.2021.07.015.

[7] C. Zhao and S. Sahni, "Linear Space String Correction Algorithm Using The Damerau-Levenshtein Distance," in *2018 IEEE 8th International Conference on Computational Advances in Bio and Medical Sciences (ICCABS)*, 2020, p. 1. doi: 10.1109/ICCABS.2018.8541927.

[8] M. A. Yulianto and N. Nurhasanah, "The Hybrid of Jaro-Winkler and Rabin-Karp Algorithm in Detecting Indonesian Text Similarity," *Jurnal Online Informatika*, vol. 6, no. 1, p. 88, Jun. 2021, doi: 10.15575/join.v6i1.640.

[9] N. P. Sutramiani, N. Suciati, and D. Siahaan, "Transfer Learning on Balinese Character Recognition of Lontar Manuscript Using MobileNet," in *2020 4th International Conference on Informatics and Computational Sciences (ICICoS)*, 2020, pp. 1–5. doi: 10.1109/ICICoS51170.2020.9299030.

[10] N. P. Sutramiani, N. Suciati, and D. Siahaan, "MAT-AGCA: Multi Augmentation Technique on small dataset for Balinese character recognition using Convolutional Neural Network," *ICT Express*, vol. 7, no. 4, pp. 521–529, 2021, doi: https://doi.org/10.1016/j.icte.2021.04.005.

[11] G. Indrawan, I. G. Aris Gunadi, M. Santo Gitakarma, and I. K. Paramarta, "Latin to Balinese Script Transliteration: Lessons Learned from the Computer-based Implementation," in *Proceedings of the 2021 4th International Conference on Software Engineering and Information Management*, in ICSIM '21. New York, NY, USA: Association for Computing Machinery, 2021, pp. 171–175. doi: 10.1145/3451471.3451499.

[12] D. Chopra, N. Joshi, and I. Mathur, "A Review on Machine Translation in Indian Languages," 2018. [Online]. Available: www.etasr.com

[13] M. Sudarma, I. Nyoman Satya Kumara, I. Putu Agus Eka Darma Udayana, G. Program, and S. Denpasar, "Balinese Latin Text Becomes Aksara Bali Using Rule Base Method I Putu Agus Eka Darma Udayana Balinese Latin Text Becomes Aksara Bali Using Rule Base Method," 2017. [Online]. Available: http://indusedu.org

[14] Q. Yang, "Research on E-commerce Customer Satisfaction Evaluation Method Based on PSO-LSTM and Text Mining," *3C Empresa. Investigación y pensamiento crítico*, vol. 12, no. 01, pp. 51–66, Mar. 2023, doi: 10.17993/3cemp.2023.120151.51-66.

[15] S. Gopali, F. Abri, S. Siami-Namini, and A. S. Namin, "A Comparison of TCN and LSTM Models in Detecting Anomalies in Time Series Data," in *2021 IEEE International Conference on Big Data (Big Data)*, 2021, pp. 2415–2420. doi: 10.1109/BigData52589.2021.9671488.

[16] A. Khattak, Z. Mehak, H. Ahmad, M. U. Asghar, M. Z. Asghar, and A. Khan, "Customer churn prediction using composite deep learning technique," *Sci Rep*, vol. 13, no. 1, Dec. 2023, doi: 10.1038/s41598-023-44396-w.

[17] Z. Zahraa and D. Stablichenkova, “Design of Long Short Term Memory Based Deep Learning Model for Customer Churn Prediction in Business Intelligence,” *International Journal of Advances in Applied Computational Intelligence*, vol. 5, no. 1, pp. 56–64, 2024, doi: 10.54216/IJAACI.050105.

[18] F. Zhao, B. Dong, H. Pan, and A. Shi, “A Mining Algorithm to Improve LSTM for Predicting Customer Churn in Railway Freight Traffic,” *Studies in Informatics and Control*, vol. 32, no. 2, pp. 25–38, 2023, doi: 10.24846/v32i2y202303.

[19] D. Singh and B. Singh, “Feature wise normalization: An effective way of normalizing data,” *Pattern Recognit*, vol. 122, p. 108307, 2022, doi: https://doi.org/10.1016/j.patcog.2021.108307.

[20] K. Manaf, S. W. Pitara, B. Subaeki, R. Gunawan, Rodiah, and Bakhtiar, “Comparison of Carp Rabin Algorithm and Jaro-Winkler Distance to Determine The Equality of Sunda Languages,” in *2019 IEEE 13th International Conference on Telecommunication Systems, Services, and Applications (TSSA)*, 2019, pp. 77–81. doi: 10.1109/TSSA48701.2019.8985470.

[21] R. Friedman, “Tokenization in the Theory of Knowledge,” *Encyclopedia*, vol. 3, no. 1, pp. 380–386, Mar. 2023, doi: 10.3390/encyclopedia3010024.

[22] Y. Fan, C. Arora, and C. Treude, “Stop Words for Processing Software Engineering Documents: Do they Matter?,” in *2023 IEEE/ACM 2nd International Workshop on Natural Language-Based Software Engineering (NLBSE)*, 2023, pp. 40–47. doi: 10.1109/NLBSE59153.2023.00016.

[23] X. Wen and W. Li, “Time Series Prediction Based on LSTM-Attention-LSTM Model,” *IEEE Access*, vol. 11, pp. 48322–48331, 2023, doi: 10.1109/ACCESS.2023.3276628.

[24] U. Pujianto, A. P. Wibawa, and R. Ulfah, “Dictionary Distribution Based on Number of Characters for Damerau-Levenshtein Distance Spell Checker Optimization,” in *2020 6th International Conference on Science in Information Technology (ICSITech)*, 2020, pp. 180–183. doi: 10.1109/ICSITech49800.2020.9392059.

[25] D. A. Carma Citrawati and I. G. G. P. Arsa Putra, “Rescuing balinese manuscripts (Lontar) with balinese Wikisource: creating metadata, cataloging and digitising,” *New Review of Hypermedia and Multimedia*, vol. 30, no. 3–4, pp. 223–237, 2024, doi: 10.1080/13614568.2024.2345182.

[26] I. Ketut Merta, I. Made, A. Pering, and G. A. Kumudawati, “The Role of Human Resources, Competencies And Training In Nawacita Mediation In Improving The Performance of Balinese Lontar Craftsmen (IKM) In The District Karangasem Bali Province,” 2024.