Journal of Soft Computing Exploration

Homepage: htttps://shmpublisher.com/index.php/joscex

p-ISSN: 2746-7686
e-ISSN: 2746-0991

.

Optimizing Seq2Seq LSTM for regional-to-national language translation on a web platform

Dwi Intan Af'idah[1], Ardi Susanto[2], Masurah Mohamad[3], Lathifah Alfat[4]
[1,2]Department of Informatics Engineering, Harapan Bersama Polythecnic, Indonesia
[3]College of Computing, Informatics and Mathematics, Universiti Teknologi MARA Perak Branch, Tapah Campus, Malaysia
[4]Department of Informatics, Universitas Pembangunan Jaya, South Tangerang, Indonesia

Article Info

Article history:

Received March 17, 2025
Revised April 9, 2025
Accepted April 95, 2025

Keywords:

Seq2Seq LSTM
Machine translation
Low-resource languages,
Hyperparameter optimization,
Tegalan-to-indonesian

ABSTRACT

Machine translation for low-resource languages remains a significant challenge due to the lack of parallel corpora and optimized model configurations. This study developed and optimized a Seq2Seq Long Short-Term Memory (LSTM) model for Tegalan-to-Indonesian translation. A manually curated parallel corpus was constructed to train and evaluate the model. Various hyperparameter configurations were systematically tested, with the best-performing model achieving a BLEU score of 11.7381 using a dropout rate of 0.5, batch size of 64, learning rate of 0.01, and 70 training epochs. The results demonstrated that higher dropout rates, smaller batch sizes, and longer training durations enhanced model generalization and translation accuracy. The optimized model was deployed into a web-based application using Streamlit, ensuring accessibility for real-time translation. The findings highlighted the importance of hyperparameter tuning in neural machine translation for low-resource languages. Future research should explore Transformer-based architectures, larger datasets, and reinforcement learning techniques to further enhance translation quality and generalization.

Corresponding Author:

Dwi Intan Af'idah,
Department of Electrical and Computer Engineering,
Harapan Bersama Polytechnic,
Mataram Street No.9, Pesurungan Lor, Tegal, Central Java, 52147, Indonesia.
Email: dwiintanafidah@poltektegal.ac.id
https://doi.org/10.52465/joscex.v6i1.561

1. INTRODUCTION

Machine translation systems are currently more accurate and efficient as a result of the progressive developments in Natural Language Processing (NLP). However, linguistic variances, a lack of comparable corpora, and unique cultural characteristics make translation from regional language to Indonesian difficult [1]. While most machine translation research focuses on high-resource languages, underrepresented languages such as Tegalan have received little attention. Our study addresses this gap by developing a specialized translation model using the Sequence-to-Sequence (Seq2Seq) Long Short-Term Memory (LSTM) architecture. Additionally, we integrate the model into a web-based platform to enhance accessibility for real-time translation.

The necessity of smooth cross-linguistic communication has prompted advancements in translation techniques. Although they set the groundwork for automation, early methods like statistical and Bayesian models had limitations in terms of accuracy and adaptability. A breakthrough was made with the emergence of Neural Machine Translation (NMT), which uses deep learning to enhance translation quality. Due to complicated linguistic traits and a lack of training data, systems such as Google's Neural Machine Translation (GNMT) perform poorly in low-resource languages but well in high-resource ones [2]. Despite progress in NMT, Tegalan and other regional languages are still not extensively studied. Regional languages are not sufficiently addressed by some studies on NMT, which mostly focus on national language. A few researchers have explored machine translation for low-resource languages. However, these efforts are still limited in scope and methodology [3]. Thus, exploring MT models for previously unstudied low-resource languages is essential.

Previous research has examined a variety of machine translation methodologies, from statistical models to advanced neural network-based techniques. Seq2Seq LSTM networks are essential for improving performance, and the transition from conventional techniques to NMT has resulted in notable improvements in translation accuracy. This is shown from previous research that compared various LSTM methods to find out the LSTM architecture with the best performance. According to this work [4], when analyzed using standardized measures including the BLEU score, precision, recall, and F1-score, enhanced Seq2Seq LSTM models obtain greater translation accuracy. The potential of Seq2Seq LSTMs in enhancing machine translation performance was demonstrated by a comparative investigation.

Further validation of the LSTM Seq2Seq model is evident in other translation tasks. A study on Spanish-English translation using the LSTM Seq2Seq-based NMT model demonstrated effectiveness in maintaining contextual relationships on long sentences. A dataset of 47 multilingual culinary recipes in Spanish-English and English-Spanish is used in this study. Seventy percent of the entire dataset is used for training, while the remaining portion is used for testing. The higher performance of LSTM Seq2Seq in Spanish-English translation has better that English-Spanish as indicated by their respective BLEU score results [5].

The next study describes a translation case that uses a low-resource language, namely Sundanese. Sundanese is a language spoken by an ethnic group in Indonesia. An encoder-decoder Seq2Seq LSTM model was employed in the study to translate Sundanese-Indonesian and Indonesian-Sundanese. This study concludes that translation accuracy is high with minimal loss values. This confirms the appropriateness of Seq2Seq LSTMs for low-resource languages. Furthermore, this study demonstrates that using greater epochs in Adam optimization improves stability and performance in both translation directions [5].

Moreover, recent developments in NMT, such as the Skip Convolutional Network and LSTM (SCN-LSTM) architecture, have outperformed conventional models. The SCN-LSTM model increased accuracy, enhanced translation quality, and reduced semantic ambiguity [6]. Another research paper [7] describes an NMT paradigm that combines the strengths of transformers and LSTM. The suggested methodology requires enhancing the quality of translation from English to Hindi by utilizing self-attention techniques and sequential modeling. This research project is examining the integration of paraphrase approaches inside NMT frameworks for English-to-Hindi translation.

Although various academics have investigated machine translation for low-resource languages, there has been little investigation into the optimization of Seq2Seq LSTM parameters for regional language translation. As a result, it is critical to investigate the Tegalan-Indonesian translation, which includes Tegalan as a low-resource language. In addition, it is also important to optimise the Seq2Seq LSTM using multiple hyperparameter combinations. This is required to determine which hyperparameter provides the highest performance. Furthermore, the existing literature rarely examines the practical deployment of web-based platforms. Consequently, this study is trying out an unusual approach by utilizing the most effective translation model on a web platform.

Our research is aimed at highlighting the significance of hyperparameter tuning in neural machine translation namely seq2seq LSTM for low-resource languages. To accomplish our goal, we begin by building a Tegalan-Indonesian corpus to use for modeling and assessing translation models. Using this corpus, a Seq2Seq LSTM model will be constructed, regarding hyperparameters fine-tuned for optimal performance. The model's accuracy will be evaluated using BLEU, precision, recall, and the F1-score. Finally, the best-performing model will be deployed on a web-based application by Streamlit, allowing for real-time translation.

Our research contributes to preserving local languages by enhancing computerized translation for under-resourced languages. The study improves the usability of Tegalan translation by optimizing a Seq2Seq LSTM model and implementing on a simple website. Moreover, this study demonstrates the potential of deep learning in extending NLP applications to regional dialects. Most importantly, this research shows that machine translation is essential to prevent endangered languages.

2. METHOD

This project uses a structured process to generate a Seq2Seq LSTM-based machine translation model for Tegalan-to-Indonesian translation. The research procedure, depicted in Figure 1, consists of five major steps. The five steps are simultaneous corpus construction, preprocessing, model development, evaluation, and deployment. Initially, a parallel corpus is produced by manually translating Tegalan texts into Indonesian. For improved model performance, the dataset is preprocessed using tokenization, text normalization, and vectorization. During the model developing phase, the Seq2Seq LSTM model is trained through hyperparameter adjustment, training, and testing. The trained model is then assessed using standard machine translation criteria to determine its effectiveness. Finally, the most effective model is integrated into a web-based application. Web-based apps require real-time translation and accessibility to end users. This systematic methodology allows it to be easier to construct an efficient and scalable translation system for the Tegalan language.

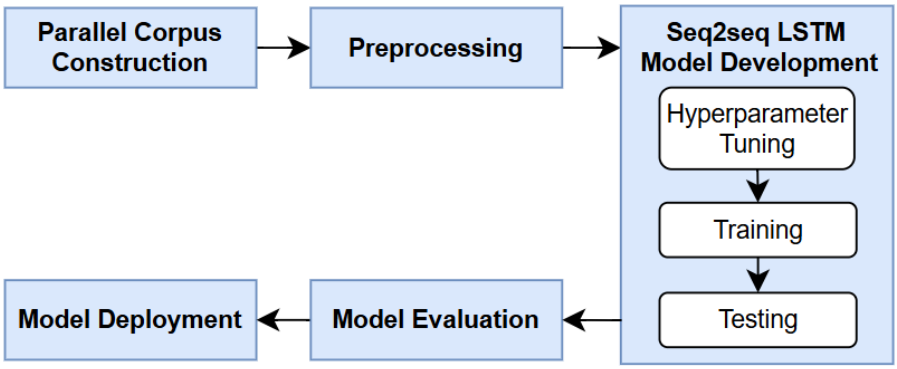

Figure 1. Workflow of Seq2Seq LSTM translation model

Parallel Corpus Contruction

Constructing a high-quality parallel corpus is a crucial phase in developing a machine translation model. Since Tegalan is a language with limited resources. Our study utilizes an existing dataset obtained by Harsh Jain's research, which is freely available on the Kaggle platform [8]. The collection contains English-Indonesian translation pairings. Following that, only the Indonesian text component is extracted and used to build a Tegalan-to-Indonesian parallel corpus. Native speakers and linguistic experts carry out manual translations to ensure that grammatical structures, cultural settings, and semantic meanings are preserved. The input of language specialists is essential in maintaining authenticity between Tegalan and Indonesian sentences while minimizing potential translation discrepancies.

In order to enhance the trustworthiness of the parallel corpus, rigorous validation was performed. Linguistic experts meticulously analyzed and validated the sentence alignment to ensure that each Tegalan sentence corresponded appropriately to Indonesian. The data was then organized into a systematic format. This was done for preprocessing and integrating the Seq2Seq LSTM model development workflow. The completed corpus is a valuable resource for training and assessing translation models. Finally, this will increase the performance of Tegalan-Indonesian translation machines and support low-resource language processing research.

Preprocessing

A systematized preprocessing workflow is adopted. This provides data consistency and quality before model training training [9], [10]. The method begins with reading and cleaning the Tegalan-Indonesian parallel corpus. A validation process follows, assuring accurate alignment between the two languages by selecting sentence pairings with the same number of lines. The cleaned text is then saved as structured Tegalan-Indonesian sentence pairs. The following step is to filter by removing punctuation marks. Filtering is also used to reduce data size and eliminate variances in terms with punctuation marks [11]. Following that, casefolding is the method employed to transform all characters into lowercase letters. This technique assures word form consistency and eliminates variances caused by letter sensitivity.

Tokenization, coding, and padding proceed after text normalization or case folding. These procedures transform textual data into a structured format appropriate for deep learning models. The tokenization function is implemented using Keras' Tokenizer module. The tokenization process outputs two separate tokenizers for

the Tegalan and Indonesian text components. These tokenizers map words to unique indices while respecting the vocabulary and structure of each language. Sequence encoding is used, which converts phrases into numerical representations [12][13]. For handling sentence lengths, padding is used by adding zeros to shorter sequences. Padding also guarantees that the input dimensions are uniform throughout the training procedure. This preparation procedure increases the dataset's quality, enabling the Seq2Seq LSTM model to effectively learn linguistic patterns [10].

After preprocessing, the dataset was separated into two parts: 80% for use in training and 20% for use in the testing phase. This fraction was chosen to ensure that the model had enough data to train from while also providing a trustworthy set for testing generalization ability. Using a set random seed ensures that data splitting remains consistent over numerous runs. Although no specific validation set is defined, the model training process is monitored using loss values to reduce the potential risk of overfitting. This data partitioning method offers a balanced framework for constructing and evaluating translation models in consistent and reproducible situations.

Model Development

This stage comprises training the translation model with the Long Short-Term Memory (LSTM) algorithm [14]. The Seq2Seq architecture comprises an encoder and a decoder for designing our model. The encoder converts the input sequence (Tegalan sentences) into a fixed-dimensional state vector, and the decoder uses this state vector as input to construct the output sequence (Indonesian sentences). Figure 2 depicts the Sequence-to-Sequence Learning (Seq2Seq) approach in our study.

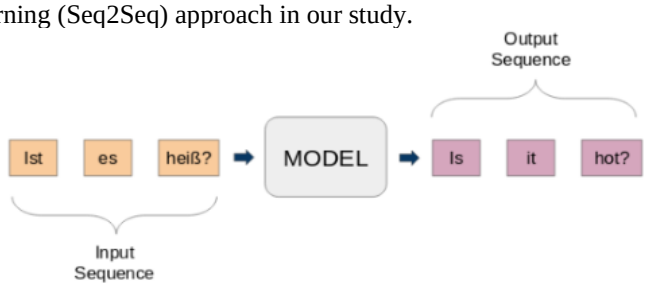

Figure 2. Seq2Seq method

Our model is composed of multiple layers, including an input layer, an embedding layer, an LSTM layer, and an output layer. The input layer gets tokenized and padded sequences. The embedding layer translates integer sequences to dense vector representations. The encoder and decoder both use LSTM layers to record temporal dependencies. A dense layer that employs the softmax activation function to generate a probability distribution across the target vocabulary [15]. The LSTM units are defined by the following equations:

$$f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] + b_f\right) \quad (1)$$

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \quad (2)$$

$$\widetilde{C_t} = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C) \quad (3)$$

$$C_t = f_t * C_{t-1} + i_t * \widetilde{C_t} \quad (4)$$

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \quad (5)$$

$$h_t = o_t * \tanh(C_t) \quad (6)$$

The f_t is the forget gate, i_t is the input gate, $\widetilde{C_t}$ is the candidate cell state, C_t is the cell state, o_t is the output gate, and h_t is the hidden state. W and b are the weight matrices and biases, and σ and tanh are the sigmoid and hyperbolic tangent activation functions, respectively. The model is trained using a preprocessed parallel corpus. The training process involves several iterations (epochs) to optimize the model weights. The loss function used is categorical cross-entropy, which is suitable for multi-class classification problems:

$$\text{Loss} = -\sum (y_i \log(\hat{y}_i)) \quad (7)$$

The y_i is the actual label and $\hat{y}_i$ is the predicted probability for class i. The Adam optimizer is employed to update the model weights, combining the advantages of AdaGrad and RMSProp. The optimization process is governed by the following equations:

$$m_t = \beta_1 m_{t-1} + (1 - \beta_1) g_t \quad (8)$$

$$v_t = \beta_2 v_{t-1} + (1 - \beta_2) g_t^2 \quad (9)$$

$$\widehat{m_t} = \frac{m_t}{1-\beta_1^t} \tag{10}$$

$$\widehat{v_t} = \frac{v_t}{1-\beta_2^t} \tag{11}$$

$$\theta_t = \theta_{t-1} - \frac{\alpha \widehat{m_t}}{\sqrt{\widehat{v_t}} + \epsilon} \tag{12}$$

The m_t and v_t are estimates of the first moment (mean) and second moment (uncentered variance) of the gradients, β_1 and β_2 are the decay rates for these estimates, α is the learning rate, g_t is the gradient at time step t ,and ϵ is a small constant for numerical stability.

The parameterization procedure entails determining model parameters such as batch size, learning rate, dropout, and epoch. The batch size sets the amount of training samples utilized in a single iteration. Meanwhile, the learning rate determines the step size utilized by the optimizer to update the model weights. Furthermore, the dropout function inhibits overfitting by making the network learn redundant representations. In addition, there is an epoch that facilitates the model to learn patterns by modifying its weights to reduce errors [16][17]. This optimization step is crucial for achieving high translation accuracy and improving the model's ability [18][19].

The building of the Seq2Seq LSTM model is systematic, beginning with data segmentation. Data segmentation is required to ensure the best training and evaluation methodologies. The dataset is separated into two sets, including a testing set and a training set. The remaining 20% of the data is saved for assessment, while the other 80% is used for the training process. Training this data enables the model to learn linguistic patterns efficiently. At the same time, the testing data serves as an impartial source for performance evaluation. Following that, each round of hyperparameter tuning will include training and testing until the model's performance value is demonstrated.

A modeling approach is carried out based on each hyperparameter combination. The Seq2Seq LSTM architecture is trained by passing preprocessed data onto it. Furthermore, the encoder-decoder system learns the relationship between Tegalan and Indonesian sentence pairings. In out experiment study, the Adam Optimizer Algorithm improves model performance during training. Following training, the model is assessed on the testing dataset. The BLEU score will be reported based on the performance evaluation results from the testing process. The BLEU score is calculated to determine the quality of the translation. Furthermore, the BLEU results provide information about the model's effectiveness.

Model Evaluation

The evaluation stage determines which model has the best performance based on the BLEU score. The BLEU score is utilized in this study since it is a well-known metric for evaluating machine translation quality. The BLEU score additionally measures the similarity between the model's translations. This allows the BLEU score to give a quantitative evaluation of translation correctness [20]. Several hyperparameter configurations are evaluated, and the model with the greatest BLEU score is chosen for use. This evaluation approach ensures that the final model performs optimally in translating Tegalan to Indonesian.

The Bilingual Evaluation Understudy (BLEU) score is a commonly used statistic for assessing the quality of machine translation outputs. This statistic is based on how closely the candidate translation matches the reference translation in terms of n-grams. The BLEU score is computed using the subsequent equation:

$$\text{BLEU} = BP.\exp\left(\sum_{n=1}^{N} w_n \log P_n\right) \tag{13}$$

In formula 13, *BP* represents the brevity penalty. P_n denotes the modified n-gram precision for n-grams with size n_1. While w_n is the weight attributed to each n-gram level (usually uniform, e.g., $w_n = \frac{1}{N}$). The brevity penalty defines *BP=1* if the candidate length *c* is more than the reference length *r.* Besides, if BP is not equal to one then $\text{BP} = \exp\left(1 - \frac{r}{c}\right)$. This approach assures that translation outputs are not only precise,

but also sufficiently long. This formulation establishes a BLEU score as a credible metric for evaluating machine translation quality.

Model Deployment

The final step is to integrate the best-performing Seq2Seq LSTM model into a web-based platform. We provide a web-based platform for real-time translation of Tegalan to Indonesian. This implementation is built with Streamlit. Streamlit is employed, resulting in a lightweight and interactive platform for developing machine learning applications. Furthermore, Streamlite's user interface is simple to use interface [21]. Users of the platform are able to input Tegalan sentences and obtain exact Indonesian translations. This implementation ensures that a larger audience has access to it while remaining computationally efficient. Then, depending on the web implementation, the translation system becomes more practical. Furthermore, the platform that we developed enables people to interact with the model without the requirement for technical expertise.

3. RESULTS AND DISCUSSIONS

Result of Dataset Preparation

Our research produces a parallel corpus of Tegalan and Indoenesia languages. This corpus is derived from two distinct text files, "tegalan.txt" and "indonesia.txt". Each file includes the relevant sentence pair. To assure data quality and consistency, a preprocessing flow is used on this corpus. First, the data set is read, and any blank lines are eliminated. Following that, we build a single corpus. Next, all punctuation marks are deleted from both language columns using a technique that reduces noise and focuses on lexical data. To ensure consistency, all text is changed to lowercase. This lowercase procedure guarantees that there is no case sensitivity.

The corpus was converted into tokens using the Keras Tokenizer. This tokenization procedure generates two unique tokenizers for Indonesian and Tegalan. As a result, these tokenizers facilitate it being easier to translate words into numerical representations, which is an important step in model input. Padding is required to handle variable phrase lengths while maintaining input uniformity for the model. Sequences were padded with zeros to ensure that all sequences had the same length. Table 1 offers an overview of the dataset following the preprocessing steps.

Table 1. Sample Dataset After Preprocessing

Original Tegal Sentence	Original Indonesian Sentence	Preprocessed Tegal Sentence	Preprocessed Indonesian Sentence	Padded Tegal Sequence	Padded Indonesian Sequence
Nyong lagi mangan sega. *(I am eating rice.)*	Saya sedang makan nasi. *(I am eating rice.)*	nyong lagi mangan sega	saya sedang makan nasi	[2, 3, 4, 5, 0, 0, 0, 0]	[6, 7, 8, 9, 0, 0, 0, 0]
Kowen pan ngendi? *(Where are you going?)*	Kamu mau pergi ke mana? *(Where are you going?)*	kowen pan ngendi	kamu mau pergi ke mana	[10, 11, 12, 0, 0, 0, 0, 0]	[13, 14, 15, 16, 17, 0, 0, 0]
Delengna kuwe! *(Look at that!)*	Lihat itu! *(Look at that!)*	delengna kuwe	lihat itu	[18, 19, 0, 0, 0, 0, 0, 0]	[20, 21, 0, 0, 0, 0, 0, 0]

Following preprocessing, we investigated the sentence length distribution in the Indonesian and Tegalan languages. This was done in order to better understand the corpus's properties. Sentence lengths were calculated by grouping words into separate groups. Then, we put them into a Pandas Data Frame to allow for statistical analysis. Figure 3 presents a Histogram illustrating the distribution of sentence lengths in both languages. This histogram provides a comparative overview and highlights any potential issues or biases in the dataset. This sentence length analysis can be used to identify the proper sequence length for padding, which may influence the neural machine translation model design.

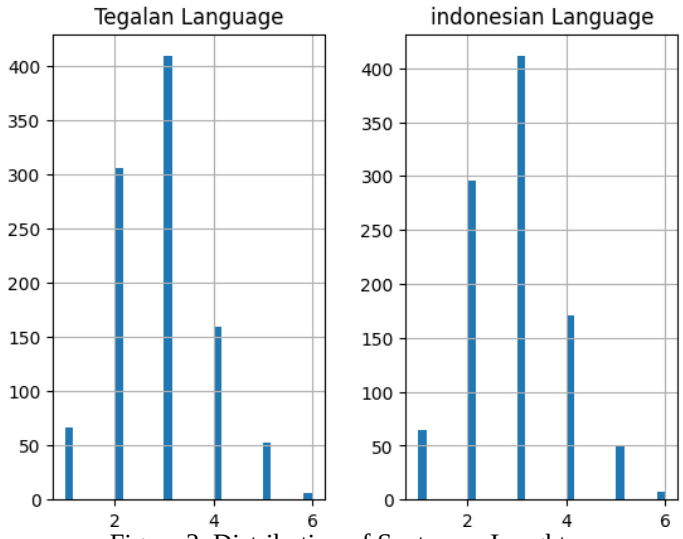

Figure 3. Distribution of Sentences Lenghts

Result of Model Evaluation

The Seq2Seq LSTM model is evaluated by adjusting a number of hyperparameters to enhance translation performance. We employed four key hyperparameters: dropout rate, batch size, learning rate, and number of epochs. The dropout rate controls the inactivation of neurons during training to minimize overfitting. The dropout values were set to 0.2, 0.3, and 0.5. The batch size is the second parameter, and it specifies how many training data are processed before the model's weights are updated. We tried with batch sizes of 64, 128, and 256. The next parameter is the learning rate. The learning rate is a key aspect in determining the step size in weight updating. We tested learning rates of 0.001, 0.005, and 0.01. Lastly, the number of epochs determines the frequency which the model processes the complete dataset. We used three epochs: 30, 50, and 70. There were 81 distinct possibilities formed when all available hyperparameter setups were taken into account. These hyperparameter adjustments were investigated systematically to discover the most successful configuration using the BLEU Score.

Figure 4 demonstrates the distribution of BLEU scores for various hyperparameter setups. The box plots show the median, interquartile range, and probable outliers, revealing how each hyperparameter affects translation performance. The dropout rate varies significantly depending on BLEU scores. Higher dropout values lead to a greater range of BLEU scores. The batch size shows moderate fluctuation. Smaller batch sizes were used to produce more consistent BLEU results. Meanwhile, the learning rate has a significant impact on the BLEU score distribution. Larger learning rates resulted in a wider range of scores. Similarly, the number of epochs influences performance consistency. Epochs with longer training cycles might result in performance improvements but with increased variability.

The analysis of Figure 4 demonstrates that hyperparameter selection has a considerable impact on translation accuracy. This is evident in the range of BLEU scores across configurations. The learning rate appears as a key component. Lower numbers (e.g., 0.001) yield more stable results, whereas higher rates (e.g., 0.01) result in increased transmission. At the same time, a 0.2 dropout rate appears to strike a balance between generalizability and performance. Moreover, higher dropout values lead to larger volatility. Besides that, smaller batch sizes (e.g., 64) result in more consistent BLEU results. This shows that using a big batch size may not be appropriate for this dataset. Furthermore, training for more epochs (50 or 70) often improves performance, but the benefits decline beyond a certain point. These findings serve as a foundation for determining the most effective hyperparameter configuration for improving the resilience and accuracy of the Tegalan-to-Indonesian translation model.

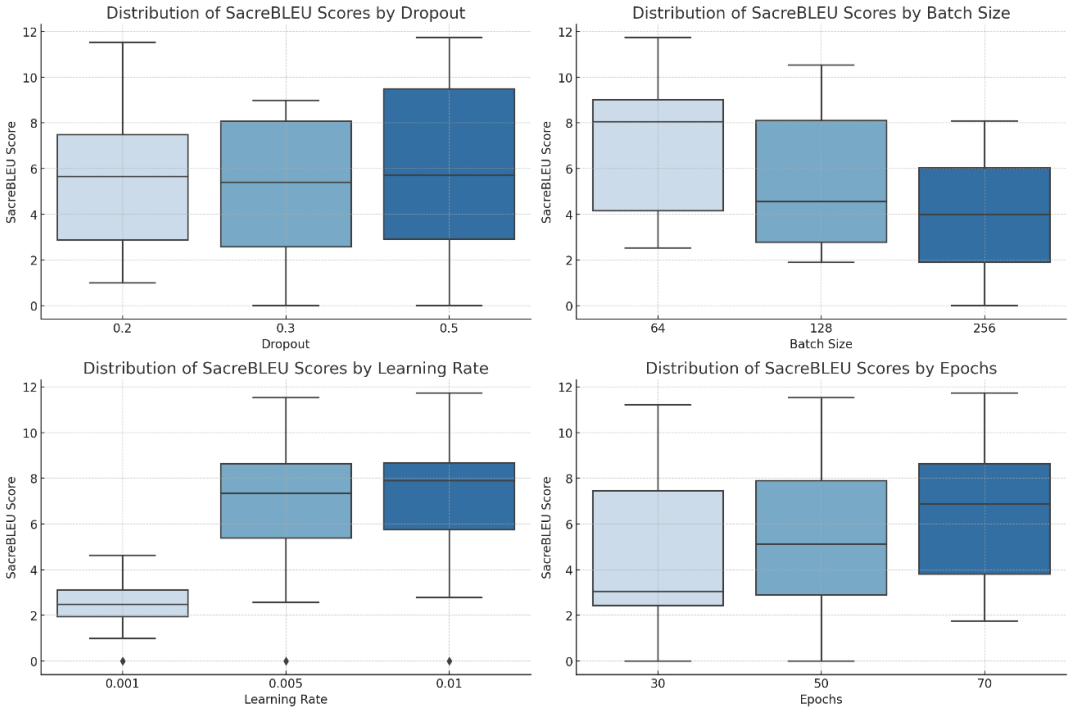

Figure 4. Distribution of scare BLEU score regarding parameters

Figure 5 depicts a line chart that shows the trend of BLEU scores across various hyperparameter setups. The observed variation implies that hyperparameter selection has a considerable impact on translation performance. This is clear because certain configurations produce much greater results than others. Among all evaluated setups, the following ratio hyperparameters produced the highest BLEU score of 11.7381: dropout 0.5, batch size 64, learning ratio 0.01, and 70 epochs. This configuration demonstrates that a higher dropout percentage, paired with a smaller batch size and more training epochs, leads to enhanced model performance. These findings emphasize the need of careful hyperparameter adjustment in neural machine translation. This emphasizes the need of carefully selecting configuration parameters to optimize translation accuracy and model stability.

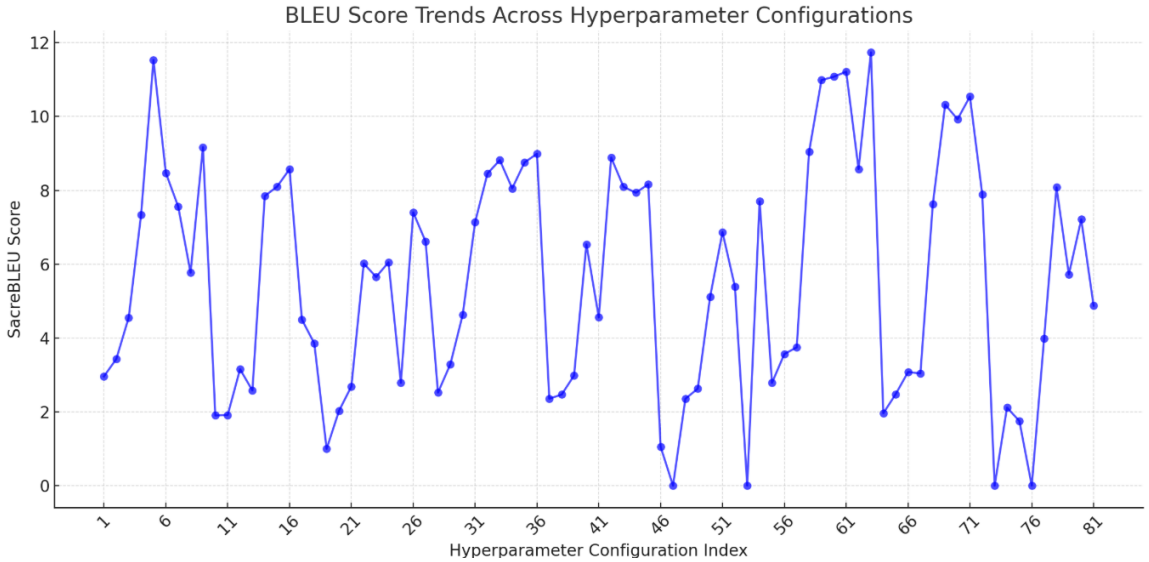

Figure 5. BLEU score trends across hyperparameter configurations

Result of Model Deployment

The top-performing Seq2Seq LSTM model had a BLEU score of 11.7381. This paradigm is used in a web-based program called Streamlit to provide real-time Tegalan-to-Indonesian translation. As illustrated in Figure 6, the web interface enables users to enter Tegalan sentences and receive quick translations into Indonesian. Our implementation on a web-based platform improves accessibility by incorporating the optimized model into

an interactive environment. This platform enables efficient and accurate translations without requiring significant computational resources. This implementation improves the model's practical applicability while also helping to preserve low-resource languages.

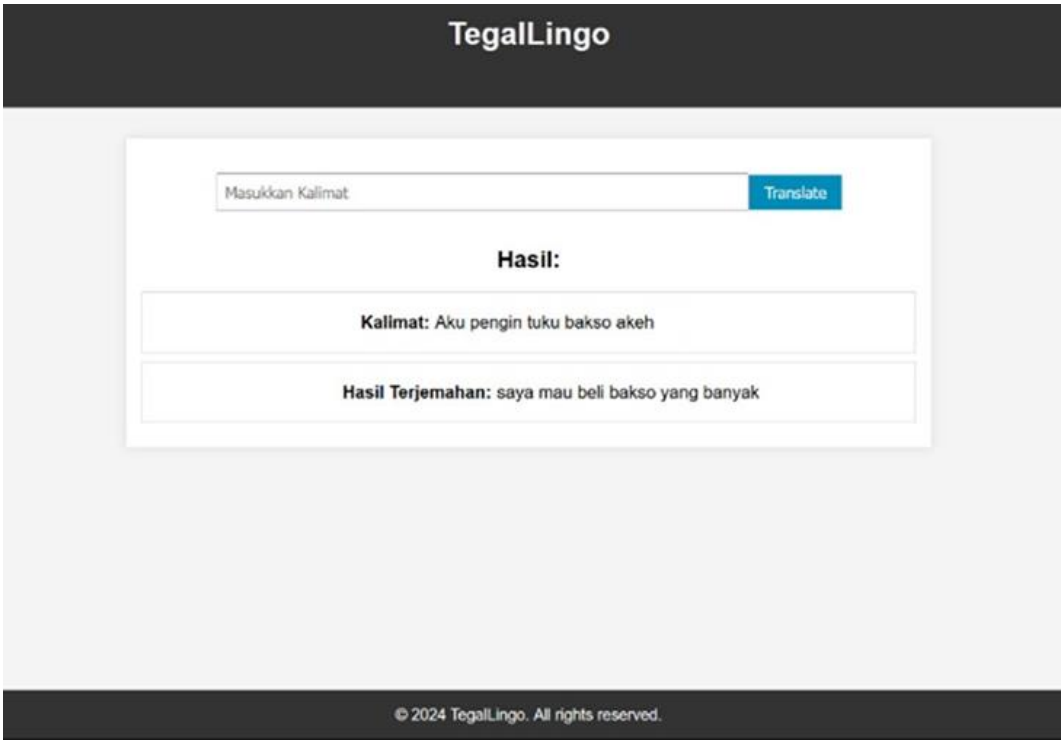

Figure 6. Web platform of tegalan-to-indonesian language

Discussion

The results of this study show that hyperparameter adjustment has a substantial impact on the performance of the Tegalan to Indonesian translation model based on Seq2Seq LSTM. The maximum BLEU score of 11.7381 was obtained with the following configurations: a dropout rate of 0.5, a batch size of 64, a learning rate of 0.01, and 70 epochs.

When compared to earlier studies on machine translation for low-resource languages, our findings show competitive performance in terms of translation accuracy and system deployment. Table 2 displays that the proposed Seq2Seq LSTM model for Tegalan-Indonesian translation outperformed earlier LSTM-based techniques applied to languages such as Sundanese and Javanese, which got BLEU scores of 10.28 [5] and 9.80 [22]. In contrast to previous studies, which relied mostly on lower dropout rates and shorter training epochs, our work used a greater dropout rate and a longer training duration, resulting in improved generalization and translation quality. Furthermore, unlike earlier studies, which did not discuss implementation in practical systems, our current work distributed the optimized model through a web platform. This platform supports real-time translation and improves accessibility. These results indicate that careful hyperparameter tweaking and deployment integration can considerably increase the effectiveness and usability of neural machine translation systems for regional languages.

Table 2. Comparison of BLEU scores in language translation studies

Study	Language Pair	Model Architecture	BLEU Score	Deployment
Our Study	Tegalan-Indonesia	seq2seq LSTM	11.74	Web Platform (Streamlit)
Ramadhan et al., 2022 [5]	English-Sundanese	LSTM	10.28	Not specified
Hidayattullah et al., 2020 [22]	Javanese Dialects	CNN-BiLTSM	9.80	Not specified

.

Our study's findings emphasize the necessity of hyperparameter adjustment in neural machine translation models, particularly for low-resource languages. The higher performance of chosen hyperparameter configurations implies that fine-tuning the dropout rate, learning rate, and training time might have a considerable impact on translation quality. The practical application of our concept on a web-based platform emphasizes the research's practical usefulness. This demonstrates the feasibility of implementing real-time translation systems for underrepresented languages. Furthermore, our findings highlight the importance of high-quality parallel corpora, such as manual translation methods, for ensuring adequate language preservation.

One of the study's main features is its methodical approach to hyperparameter optimization, which allowed us to identify an optimal configuration for Tegalan-to-Indonesian translation. Furthermore, integrating the model into a Streamlit-based web application makes it more accessible to a wider range of users. However, the study had certain drawbacks. The short parallel corpus may have limited the model's ability to generalize across Tegalan's numerous language variances. Furthermore, while Seq2Seq LSTM networks performed well in this setting, the use of Transformer-based models could provide even greater translation accuracy. Future research should look into larger datasets, different deep learning architectures, and reinforcement learning techniques to further improve machine translation for low-resource languages.

4. CONCLUSION

Our study successfully constructed and optimized a Seq2Seq LST model for Tegalan-to-Indonesian machine translation, revealing that hyperparameter adjustment is critical for enhancing translation performance. The greatest BLEU score of 11.7381 was obtained with the ideal setup of 0.5 dropout rate, 64 batch size, 0.01 learning rate, and 70 epochs. Our analysis confirms that correctly chosen hyperparameters considerably improve translation accuracy. The deployment of the best-performing model into a web-based application further validated its practical usability, making real-time translation accessible to a wider audience and supporting the broader goal of low-resource language preservation.

Future studies should explore larger parallel corpora, Transformer-based architectures, and reinforcement learning techniques to further refine translation performance. The integration of advanced deep learning methods, including attention mechanisms and transfer learning, may enhance translation quality while addressing existing challenges in low-resource language processing. Additionally, expanding the application scope of the developed model to other regional dialects could contribute to broader linguistic inclusivity in machine translation research.

ACKNOWLEDGEMENTS

The authors would like to express their gratitude to Harapan Bersama Polytechnic for providing financial support for this research. The funding played a crucial role in facilitating data collection, model development, and the deployment of the translation system. The authors also extend their appreciation to all contributors who provided valuable insights and expertise throughout the study.

CREDIT AUTHORSHIP CONTRIBUTION STATEMENT

Author1: Conceptualization, Methodology, Model Development. **Author2**: Project administration, Software, Writting – original draft. **Author3**: Writing – review & editing. **Author4**: Validation

DECLARATION OF COMPETING INTERESTS

The authors declare that this research was conducted independently, with no financial or personal relationships that could influence the findings. This study was funded by Harapan Bersama Polytechnic, which had no role in the study design, data collection, analysis, or publication

DATA AVAILABILITY

The data used in this study will be made available upon reasonable request. Interested researchers may contact the corresponding author for access to the dataset.

REFERENCES

[1] K. Jiang and X. Lu, "Natural Language Processing and Its Applications in Machine Translation: A Diachronic Review," *Proc. 2020 IEEE 3rd Int. Conf. Safe Prod. Informatiz. IICSPI 2020*, pp. 210–214, 2020, doi: 10.1109/IICSPI51290.2020.9332458.

[2] M. D. Okpor, "Machine Translation Approaches: Issues and Challenges," *IJCSI Int. J. Comput. Sci.*, vol. 11, no. 5, pp. 159–165, 2014, doi: 10.13140/RG.2.2.12055.38561.

[3] A. F. Aji *et al.*, "One Country, 700+ Languages: NLP Challenges for Underrepresented Languages and Dialects in Indonesia," *Proc. Annu. Meet. Assoc. Comput. Linguist.*, vol. 1, pp. 7226–7249, 2022, doi: 10.18653/v1/2022.acl-long.500.
[4] M. Ramaiah, D. Datta, C. Vanmathi, and R. Agarwal, "Study of neural machine translation with long short-term memory techniques," *Deep Learn. Res. Appl. Nat. Lang. Process.*, no. January, pp. 65–88, 2022, doi: 10.4018/978-1-6684-6001-6.ch005.
[5] T. I. Ramadhan, N. G. Ramadhan, and A. Supriatman, "Implementation of Neural Machine Translation for English-Sundanese Language using Long Short Term Memory (LSTM)," *Build. Informatics, Technol. Sci.*, vol. 4, no. 3, pp. 1438–1446, 2022, doi: 10.47065/bits.v4i3.2614.
[6] B. Ren, "The use of machine translation algorithm based on residual and LSTM neural network in translation teaching," *PLoS One*, vol. 15, no. 11 November, pp. 1–16, 2020, doi: 10.1371/journal.pone.0240663.
[7] S. Sharma, "A Transformer based approach using LSTM and Paraphrase reference to Translate English Text into Hindi," 2023.
[8] H. Jain, "Machine Translation Using Seq2Seq Modelling," 2021.
[9] Gemma Team *et al.*, "Gemma: Open Models Based on Gemini Research and Technology," 2024.
[10] P. Bhuvaneshwari and A. N. Rao, "A comparative study on various pre-processing techniques and deep learning algorithms for text classification," *Int. J. Cloud Comput.*, vol. 11, no. 1, pp. 61–78, 2022, doi: 10.1504/IJCC.2022.121076.
[11] S. J. Mielke *et al.*, "Between words and characters: A Brief History of Open-Vocabulary Modeling and Tokenization in NLP," 2021.
[12] M. A. Oladipupo, P. C. Obuzor, B. J. Bamgbade, K. M. Olagunju, A. E. Adeniyi, and S. A. Ajagbe, "An Automated Python Script for Data Cleaning and Labeling using Machine Learning Technique," *Inform.*, vol. 47, no. 6, pp. 219–232, 2023, doi: 10.31449/inf.v47i6.4474.
[13] K. Imamura and M. Utiyama, "An Empirical Study of Multilingual Vocabulary for Neural Machine Translation Models," *WAT 2024 - 11th Work. Asian Transl. Proc. Work.*, no. Wat, pp. 22–35, 2024.
[14] Y. Yang, "Application of LSTM Neural Network Technology Embedded in English Intelligent Translation," *Comput. Intell. Neurosci.*, vol. 2022, 2022, doi: 10.1155/2022/1085577.
[15] G. Tiwari, A. Sharma, A. Sahotra, and R. Kapoor, "English-Hindi Neural Machine Translation-LSTM Seq2Seq and ConvS2S," in *2020 International Conference on Communication and Signal Processing (ICCSP)*, IEEE, Jul. 2020, pp. 871–875. doi: 10.1109/ICCSP48568.2020.9182117.
[16] H. Wardhana, I. M. Yadi Dharma, K. Marzuki, and I. Syarif Hidayatullah, "Implementation of Neural Machine Translation in Translating from Indonesian to Sasak Language," *MATRIK J. Manajemen, Tek. Inform. dan Rekayasa Komput.*, vol. 23, no. 2, pp. 465–476, 2024, doi: 10.30812/matrik.v23i2.3465.
[17] Y. Cui, S. Wang, and J. Li, "LSTM neural reordering feature for statistical machine translation," *2016 Conf. North Am. Chapter Assoc. Comput. Linguist. Hum. Lang. Technol. NAACL HLT 2016 - Proc. Conf.*, pp. 977–982, 2016, doi: 10.18653/v1/n16-1112.
[18] B. H. Shekar and G. Dagnew, "Grid search-based hyperparameter tuning and classification of microarray cancer data," *2019 2nd Int. Conf. Adv. Comput. Commun. Paradig. ICACCP 2019*, pp. 1–8, 2019, doi: 10.1109/ICACCP.2019.8882943.
[19] D. Puspitaningrum, "A Study of English-Indonesian Neural Machine Translation with Attention (Seq2Seq, ConvSeq2Seq, RNN, and MHA): A Comparative Study of NMT on English-Indonesian," *ACM Int. Conf. Proceeding Ser.*, pp. 271–280, 2021, doi: 10.1145/3479645.3479703.
[20] S. Lee *et al.*, "A Survey on Evaluation Metrics for Machine Translation," *Mathematics*, vol. 11, no. 4, pp. 1–22, 2023, doi: 10.3390/math11041006.
[21] S. Studer *et al.*, "Towards CRISP-ML(Q): A Machine Learning Process Model with Quality Assurance Methodology," *Mach. Learn. Knowl. Extr.*, vol. 3, no. 2, pp. 392–413, 2021, doi: 10.3390/make3020020.
[22] A. F. Hidayatullah, S. Cahyaningtyas, and R. D. Pamungkas, "Attention-based CNN-BiLSTM for Dialect Identification on Javanese Text," *Kinet. Game Technol. Inf. Syst. Comput. Network, Comput. Electron. Control*, vol. 4, pp. 317–324, 2020, doi: 10.22219/kinetik.v5i4.1121.

.