Kinetik: Game Technology, Information System, Computer Network, Computing, Electronics, and Control
Journal homepage: http://kinetik.umm.ac.id
ISSN: 2503-2267
Vol. 9, No. 2, May 2024, Pp.119-128

Multi-label classification of Indonesian qur'an translation using long short-term memory model

Ismail Akbar*[1], Muhammad Faisal*[1], Totok Chamidy[1]
Master Program in Informatics, Maulana Malik Ibrahim State Islamic University Malang, Indonesia[1]

Article Info

Keywords:
LSTM, Bi-LSTM, Word Embedding, Qur'an, Translation Classification

Article history:
Received: November 17, 2023
Accepted: March 28, 2024
Published: May 31, 2024

Cite:
I. Akbar, M. Faisal, and T. Chamidy, "Multi-Label Classification of Indonesian Qur'an Translation using Long Short-Term Memory Model", KINETIK, vol. 9, no. 2, May 2024. Retrieved from https://kinetik.umm.ac.id/index.php/kinetik/article/view/1901

*Corresponding author.
Ismail Akbar
Muhammad Faisal
E-mail address:
ismaelakbar12@gmail.com
mfaisal@ti.uin-malang.ac.id

Abstract

Studying the Quran is an integral act of worship in Islam, necessitating a nuanced comprehension of its verses to ease learning and referencing. Recognizing the diverse thematic elements within each verse, this research pioneers in applying Deep Learning techniques, specifically Long Short-Term Memory (LSTM) and Bidirectional LSTM (Bi-LSTM), coupled with Word Embedding methods like Word2Vec and FastText, to refine the multi-label classification of the Quran's translations into Indonesian. Targeting core thematic categories such as Tawheed, Worship, Akhlaq, and History, the study aims to elevate classification accuracy, thereby enhancing the textual understanding and educational utility of the Quran's teachings. The employment of Bi-LSTM in conjunction with FastText and meticulous hyperparameter optimization has yielded promising results, achieving an accuracy of 71.63%, precision of 64.06%, recall of 63.60%, and a hamming loss of 36.17%. These outcomes represent a significant advancement in the computational analysis of religious texts, offering novel insights into the complex domain of Quranic studies. Furthermore, the research accentuates the critical role of selecting suitable word embedding techniques and the necessity of precise parameter adjustments to amplify model performance, thereby contributing to the broader field of religious text analysis and understanding. Through such computational approaches, this study not only fosters a deeper appreciation of the Quran's multifaceted teachings but also sets a new precedent for the interdisciplinary integration of Islamic studies and artificial intelligence.

1. Introduction

Recent advancements in multi-label classification [1], such as its application in bioinformatics [2] for gene function prediction and in image recognition [3] for object identification, highlight the technique's potential across diverse domains. By extending these innovative methodologies to the analysis of Indonesian Quran translations, this study contributes to the evolving landscape of computational linguistics in religious text analysis. The application of Long Short-Term Memory (LSTM) [4] and Bidirectional LSTM (Bi-LSTM) [5] models, especially in the context of Indonesian translations of the Quran, represents a novel frontier in this research domain. Despite the growing application of computational methods in religious text analysis, the nuanced nature of Quranic verse classification, especially in Indonesian, poses unique challenges [6]. Current methods have yet to fully address the intricate interplay of semantic and syntactic nuances in multi-label classification of such texts.

The act of studying and interpreting the Quran is a practice that is encouraged in Islam, allowing individuals to understand and adhere to the principles of Islam accurately and firmly. A categorized understanding of its verses can facilitate deeper study and easier identification of related teachings. According to Hasyim in the research [7] the teachings contained in the Qur'an primarily cover Tawhid, Worship, Social Transactions, Morals, History, and Sharia. While the classification of Quranic verses has seen some application of computational methods, the nuanced nature of multi-label classification, particularly in the context of Indonesian translations, remains a significant challenge. Existing methodologies have struggled to accurately categorize verses that encompass multiple themes, resulting in a need for more sophisticated approaches. This research addresses the critical issue of enhancing multi-label classification accuracy for Indonesian Quran translations, a pivotal step toward deepening our understanding and interpretation of this sacred text.

Categorizing verses of the Qur'an can facilitate the search for verses as desired. Just like in searching for the meaning of the translation of the Qur'an [8]. What's interesting about the classification of the verses of the Qur'an is that each verse can belong to one or more different classes, and many words with the same meaning are often found in the translated text of the Qur'an [9]. This fact shows that the classification of verses in the Qur'an differs from the overall classification, where only all sources or documents are classified. Such classification case can be referred to as multi-label classification.

Multi-label classification is applied to the classification of Bukhari Hadith translations in Indonesian using the K-Nearest Neighbor (K-NN) model [10] - [11] in testing the K-Nearest Neighbor (K-NN) model for the classification of Quran translations by improving feature selection with Information Gain to reduce noise caused by irrelevant features to obtain higher accuracy results. The accuracy, precision, and recall rates obtained were 64.10%, 65%, and 62.68% respectively. Moreover, the multi-label classification of Quran translations using the Support Vector Machine (SVM) algorithm was able to increase the accuracy by adding the number of test data. The results of these tests showed that the Support Vector Machine (SVM) outperformed other algorithms by achieving an accuracy of 70% [12].

Text classification using a deep learning baseline was conducted by [13] using LSTM as the classification algorithm. The optimal result was achieved by LSTM with 256 neurons and 10 epochs, obtaining an accuracy of 86.23%. In another variant of LSTM, the use of Bi-directional Long Short-Term Memory (Bi-LSTM) combined with the Word2Vec CBOW architecture, Hierarchical Softmax, dimension 200, Bi-LSTM with a dropout of 0.5, and a learning rate of 0.0001, reached the highest accuracy of 96.86% [14].

From the explanation above, this study aims to fill the knowledge gap by applying and comparing two word embedding techniques, Word2Vec and FastText, to improve accuracy in the classification of Al-Qur'an translation texts in Indonesian. The choice of these techniques is based on their proven ability to detect semantic and syntactic relationships between words, which is essential for the contextual analysis of the content of the Al-Qur'an. This research introduces a novel integration of LSTM and Bi-LSTM models with advanced word embedding techniques, specifically tailored for the Indonesian Quran translation. Unlike previous studies, this approach is designed to capture the unique linguistic and thematic complexity of the Quran, offering a pioneering solution in the field of religious text analysis. By innovatively applying these models to Indonesian Quran translations, we address the specific challenges posed by the text's multifaceted themes and linguistic nuances, thereby advancing the field of computational religious text analysis.

2. Research Method

Multi-label classification differs from single-label classification. Single-label classification categorizes text into one label category. Meanwhile, multi-label classification can group text into several categories or labels [15]. In this study, a variant of the LSTM algorithm consisting of Long Short-Term Memory (LSTM) and Bidirectional Long Short-Term Memory (Bi-LSTM) is used for the multi-label classification of the Indonesian translation of the Quran, which is represented in Figure 1.

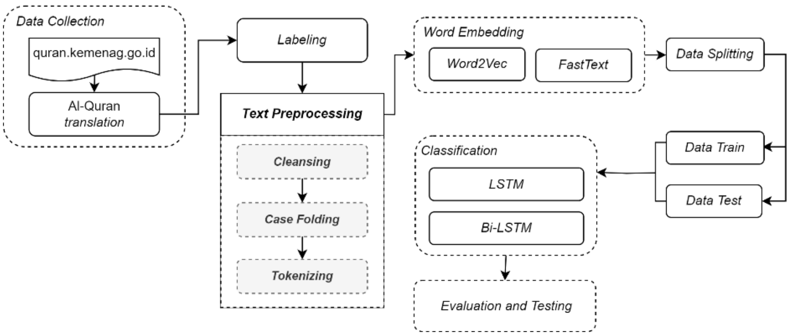


Figure 1. Research Process

2.1 Data Collection

In this study, multilabel text classification requires text as the object of study. The data used comes from the translation of the Quran in Indonesian by the Ministry of Religious Affairs of the Republic of Indonesia in 2022, available at https://quran.kemenag.go.id. To access it, navigate to the bottom of the main page and find the Quran Manuscript Authentication Commission, then direct to the download page and select "Translate the Quran". The file available for download is named "Al-Quran Translation_v161122.rar", which contains documents of each surah. To facilitate multilabel text classification, the document is then converted into a csv file format consisting of several surahs in the form of Table 1. After that, the dataset is distributed to experts for labeling.

2.2 Data Labeling

In the context of this research, data labeling was carried out manually by experts. There are two specialists involved: one is an expert in Qur'an and Hadith studies, while the other is an expert in Qur'an interpretation. They each independently annotated the identical dataset, and the labeling results from each expert are presented in Table 2 and Table 3.

Table 1. Results of Al-Qur'an Translation Dataset Collection

Surah	Verse	Language	Qur'an Translation
5	26	ID	Allah berfirman: "(Jika demikian), maka sesungguhnya negeri itu diharamkan atas mereka selama empat puluh tahun, …..
		EN	Allah says: "(If so), then indeed the land is forbidden to them for forty years,...
5	28	ID	Sungguh kalau kamu menggerakkan tanganmu kepadaku untuk membunuhku, aku sekali-kali tidak akan menggerakkan ….
		EN	Indeed, if you move your hand on me to kill me, I will never move...
			….
6	160	ID	Barangsiapa membawa amal yang baik, maka baginya (pahala) sepuluh kali lipat ….
		EN	Whoever brings good deeds, he will be (rewarded) ten times as much...

Table 2. Labeling Results from Qur'an and Hadith Studies Expert

Language	Qur'an Translation	Label			
		Tawheed	Worship	Morals	History
ID	Dan berikanlah kepada anak-anak yatim (yang sudah balig) harta mereka, jangan …	0	1	1	0
EN	And give to orphans (who have reached adulthood) their wealth, don't ...				

Table 3. Labeling Results from Al-Qur'an Interpretation Expert

Language	Qur'an Translation	Label			
		Tawheed	Worship	Morals	History
ID	Dan berikanlah kepada anak-anak yatim (yang sudah balig) harta mereka, jangan …	0	0	1	0
EN	And give to orphans (who have reached adulthood) their wealth, don't ...				

After the labeling process, the results are compared to determine the consistency between the two experts. In cases where there is a difference of opinion between experts, discussions are conducted to reach a mutual agreement on the precise classification for each data instance, where both experts must discuss their reasons. The discussion should focus on the predetermined annotation criteria and how each instance meets or does not meet these criteria. According to the definition of Tawhid Label, which describes texts that state the oneness of Allah Subhanahu Wa Ta'ala [16]. Second, Worship is a text that has a meaning of submission and obedience in performing and avoiding all prohibitions of Allah Subhanahu Wa Ta'ala [17]. Third, Morals describe texts that state the character and nature of a person that produces good and bad deeds [18]. Fourth, History describes texts containing stories or events from the past that narrate the history of previous nations to be learned by subsequent generations [19]. Thus, the labeling results can be exemplified in Table 4, where if a label is given a value of 1, then the translation falls under that category, and a value of 0 means the translation does not fall under the label.

Table 4. Labeling Results from Two Experts

Language	Qur'an Translation	Tawheed	Worship	Morals	History
ID	Dan berikanlah kepada anak-anak yatim (yang sudah balig) harta mereka, jangan …	0	1	1	0
EN	And give to orphans (who have reached adulthood) their wealth, don't ...				

2.3 Preprocessing

The main purpose of text preprocessing is to improve the data by making it more structured. The text preprocessing step is necessary to convert the sentiment dataset into structured text so that the data is ready to be processed to the next stage. In addition, this process can improve the performance and accuracy of classification models [20]. The text preprocessing process in the research consists of three stages, namely cleansing, the removal or cleaning of translation data containing numbers and delimeters, such as commas (,), quotation marks ("), full stops (.) and others, case folding, which is used to convert all characters into similar letters [21], and tokenizing, which is a step-by-step process to break sentences, paragraphs, or documents into parts called tokens [22].

2.4 Word Embedding

In simple terms, word embedding is a technique to convert text into vectors so that it can be processed by certain algorithms [23]. The result of word embedding can be used to describe the closeness of a word or document, but it must be understood that the closeness is contextually close depending on the training data used during word training [24]. Visually, Word Embedding can be represented that each word is represented by a dot in a certain area. This study has a testing scheme for word embedding with the determination of dimension size and epoch parameters presented in Table 5.

Table 5. Word Embedding Testing Scheme

Word Embedding Models	Parameter	
	Dimensions Size	Epoch
Word2Vec	100	10, 20, 30
	200	10, 20, 30
	300	10, 20, 30
FastText	100	10, 20, 30
	200	10, 20, 30
	300	10, 20, 30

2.5 Word2Vec

The Word2Vec method was proposed by Mikolov in 2013, considering the corpus as input and the output as a vector. Word2Vec converts each word into a vector. The advantage of Word2Vec is that it can represent the contextual similarity of two words in the result vector [25]. Word2Vec has three parameters that affect the model learning process, namely architecture, evaluation method, and dimension. Each of the three parameters in Word2Vec has an influence on the performance of deep learning accuracy [26]. Word2Vec architecture consists of two types, namely Continuous Bag of Words (CBOW) and Skip-gram [27] which is outlined in Figure 2. Continuous Bag of Words (CBOW) is chosen as the architecture in Word2Vec because Continuous Bag of Words (CBOW) produces better performance and has slightly higher accuracy on frequently occurring words, and in this search dataset there are many frequently occurring words [14].

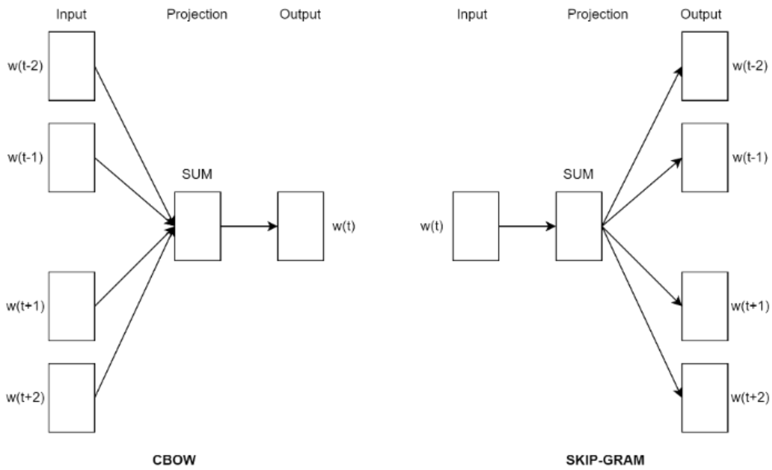


Figure 2. CBOW and Skip-Gram Architecture

2.6 FastText

FastText is a feature extraction method from real numbers with the concept of word embedding based on prediction [28]. Unlike word2vec, this algorithm has a hierarchical structure and is represented as a dense vector. One of the advantages of fasttext is that it can cope with imbalanced data distribution [29]. The difference of FastText is that the input consists of scharacter n-gram subwords, so that feature extraction can be done at the prefix and suffix level outside the training data or out-of-vocabulary (OOV) [30].

2.7 Data Splitting

The research dataset is divided into training data and test data. Training data is used as training data in testing the model, then test data is data used for testing, which has never been used or is not used as training data [31]. Each research model was tested with three test scenarios. In the first scenario, 70% of the data is used to train the model, while the remaining 30% is reserved for testing its performance. The second scenario adjusts this split to 80% for training and 20% for testing. Lastly, the third scenario allocates most of the data for training, with 90% used for this purpose and only 10% used for testing the model. This separation is critical to validate the model's ability to generalize to unseen data and to prevent overfitting.

2.8 Long Short-Term Memory

Long Short-Term Memory (LSTM) is one type of RNN (Recurrent Neural Network) architecture used in deep learning models. LSTM itself was built to solve the vanishing gradient problem in RNN (Recurrent Neural Network) where when processing long sequential data, the slope of the loss function drops exponentially [32]. The general architecture of LSTM can be seen in Figure 3. Figure 3 illustrates the standardization of the architecture of the LSTM method which is divided into four components namely, Input Gate (i) to control the input current entering the neuron, Forget Gate (f) to put the neuron into the current reset state, Output Gate (o) to control the effect of neuron activation on other neurons and memory cells (c) [33]. LSTM components can be calculated in Equation 1 input gate, Equation 2 Forget Gate, Equation 3 Output Gate, Equation 4 memory cell, Equation 5.

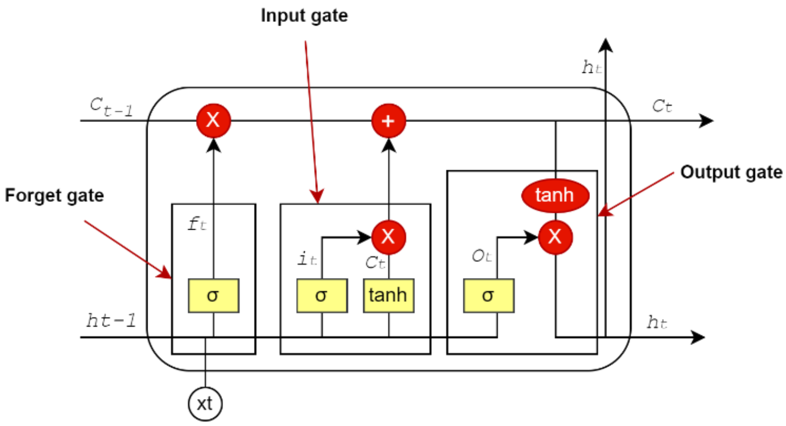


Figure 3. Architecture of Long Short-Term Memory

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

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

$$O_t = \sigma (W_o \, . \, [h_{t-1} \, . x_t] + b_0) \tag{3}$$

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

$$h_t = O_t * \tanh (C_t) \tag{5}$$

The LSTM model includes a cell state and three gates to decide how much information to store and discard. Input Gate (i_t) updates the cell state by changing the current input (x_t) and the previous hidden state (h_{t-1}) into a sigmoid function (σ) and converting it to the range 0 –1. f_t is the forget gate that determines the amount of memory to be erased, and O_t is the output gate that determines the next hidden state. The cell state is represented by C_t to store new state information by calculating the weighted sum of the previous cell state and the current information generated by the cell. h_t is the state of the hidden layer at time t by calculating this update history C_t using the function $\tanh$ [34].

2.9 Bi-directional Long Short-Term Memory

Bidirectional Long-Short Term Memory (Bi-LSTM) is a neural network architecture consisting of two LSTM layers, namely forward LSTM to model the previous context and backward LSTM to model the next context [35]. Bi-LSTM connects two hidden layers coming from opposite directions with the same output. Using this form of deep generative learning, layers of neurons can simultaneously acquire information about past and future states [36]. Figure 4 shows the architecture of the Bi-LSTM. The input is given from $t - 1$ to time $t - n$ for the previous layer and for the next layer, the input is given in the opposite direction from $t - n$ to t – 1. The forward LSTM operates from start to finish according to the order of the data. At each time step, information from the previous step is integrated into the current step processing. Each LSTM cell has three gates, namely input gate, forget gate, and output gate, which allow the organization and storage of long-term information. The Forward LSTM equation is presented in Equation 6.

$$\vec{h}_t = \sigma\,(W_{x\vec{h}}\,.\,x_t + W_{\vec{h}\vec{h}}\,.\,\vec{h}_{t-1} + b_{\vec{h}}) \tag{6}$$

Where, $\vec{h}_t$ the output produced by the LSTM cell for a given time step. σ the activation function applied to the sum on the right. $W_{x\vec{h}}$ the product of the matrix between the weights $W_{x\vec{h}}$ and the input vector x_t. $W_{\vec{h}\vec{h}}$ the result of the matrix multiplication between the weights $W_{\vec{h}\vec{h}}$ and the output vector from the previous time step $\vec{h}_{t-1}$. $b_{\vec{h}}$ is the bias.

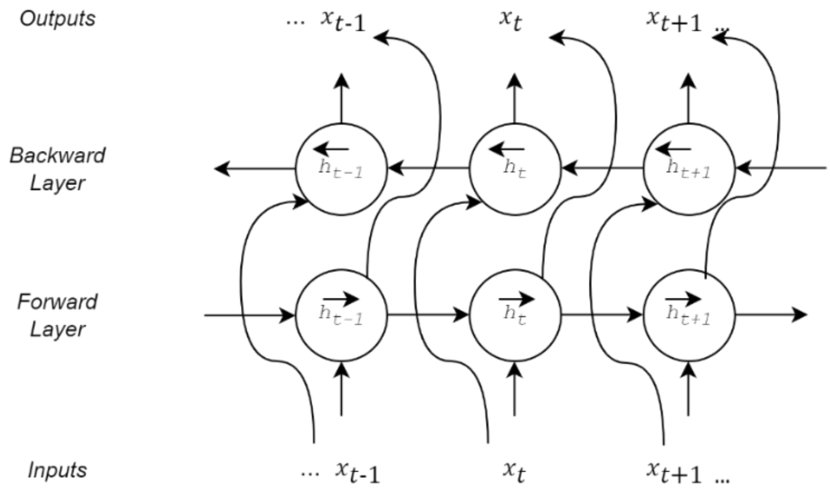


Figure 4. Bi-directional Long Short-Term Memory Architecture

Whereas the backward LSTM operates from start to finish following the same data sequence. Like the forward LSTM, the backward LSTM also has a similar structure with three gates, making it possible to understand the context of the backward time step. Equation 7 is the equation of the backward LSTM.

$$\overleftarrow{h}_t = \sigma\,(W_{x\overleftarrow{h}}\,.\,x_t + W_{\overleftarrow{h}\overleftarrow{h}}\,.\,\overleftarrow{h}_{t+1} + b_{\overleftarrow{h}}) \tag{7}$$

Where, $\overleftarrow{h}_t$ the output (hidden state) at time step t of the backward moving LSTM. σ the activation function applied to the linear sum of the input and the previous state. $W_{x\overleftarrow{h}}$ the result of the weight matrix that connects the input x_t to $\overleftarrow{h}_t$. $W_{\overleftarrow{h}\overleftarrow{h}}$ the weight matrix connecting $\overleftarrow{h}_t$ at the previous time step t + 1 to $\overleftarrow{h}_t$ at time step *t*. $b_{\overleftarrow{h}}$ is the bias.

The outputs of the Bi-LSTM, both forward and backward, are combined at each time step. This means that the Bi-LSTM layer produces an output vector yt, where each element is calculated using Equation 8.

$$y_t = W_{\vec{h}y}\vec{h}_t + W_{\overleftarrow{h}y}\overleftarrow{h}_t + b_y \tag{8}$$

Where y_t output at time step t. $W_{\vec{h}y}$ weights connecting the hidden representation of the previous time step to the output y_t. $\vec{h}_t$ hidden representation at the previous time step. $W_{\overleftarrow{h}y}$ the weight that connects the hidden representation of the next time step to the output y_t. $\overleftarrow{h}_t$ The hidden representation at the next time step. b_y bias for the output.

3. Results and Discussion

The data used in the study comes from the translation of the Qur'an by the Ministry of Religious Affairs of the Republic of Indonesia, selecting chapters from juz 4 - 8, including Surah An-Nisa', Surah Al-Maidah, and Surah Al-An'am [7], with a total of 461 Qur'anic verse translations. Subsequently, labeling was done manually by expert scholars with labels of Tawhid, Worship, Morals, and History assigned to each Qur'anic verse translation as presented in Table 4. Thus, the label distribution can be reviewed in Figure 5 with a total of 324 labels for Tawhid, 205 labels for Worship, 265 for Morals, and 141 for History.

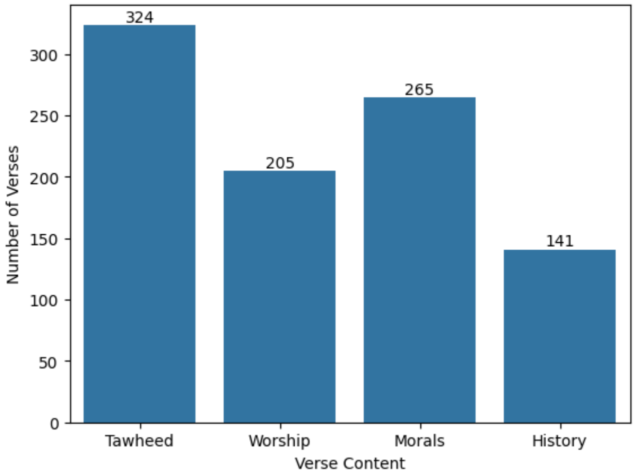


Figure 5. Distribution of Qur'an Translation Labels

Table 6. Preprocessing Results

Original Text	After Preprocessing
Dan mereka (orang-orang munafik) mengatakan: "(Kewajiban kami hanyalah) taat". Tetapi apabila mereka telah pergi dari sisimu, sebahagian dari mereka mengatur siasat di malam hari (mengambil keputusan) lain dari yang telah mereka katakan tadi. Allah menulis siasat yang mereka atur di malam hari itu, maka berpalinglah kamu dari mereka dan tawakallah kepada Allah. Cukuplah Allah menjadi Pelindung.	dan mereka orangorang munafik mengatakan kewajiban kami hanyalah taat tetapi apabila mereka telah pergi dari sisimu sebahagian dari mereka mengatur siasat di malam hari mengambil keputusan lain dari yang telah mereka katakan tadi allah menulis siasat yang mereka atur di malam hari itu maka berpalinglah kamu dari mereka dan tawakallah kepada allah cukuplah allah menjadi pelindung

Every model using LSTM algorithm variants must have the same sequence length, thus a process of adding empty tokens at the beginning of the data is necessary to make the total token length of each data the same [37]. Figure 6 shows a histogram producing the largest token length of about 20 to 50 tokens, while the smallest token length is about 125 to 150 tokens. Therefore, the value of max_seq_length in the LSTM and Bi-LSTM models is set to 150 tokens.

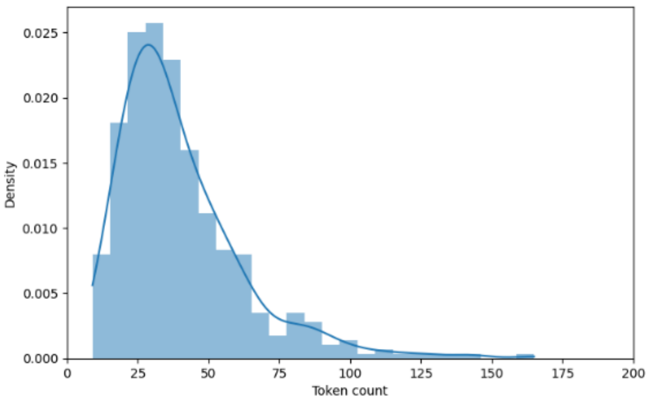


Figure 6. Distribution of Word Tokens in the Data

After obtaining the sequence length, the next step of determining the best model is carried out by using a training scheme on a translation dataset. The training scheme conducted in this study achieved quite good results with 70% of the data for training and 30% for testing. Based on this data distribution, training with the LSTM model resulted in a superior Bi-LSTM algorithm, as shown in Table 7, with an accuracy of 49.76%, precision of 62.23%, recall of 56.37%, and a hamming loss of 35.14%.

Table 7. Model Evaluation LSTM and Bi-LSTM

Algorithm	Accuracy	Precision	Recall	Hamming Loss
LSTM	47,70%	63,00%	63,00%	37,35%
Bi-LSTM	49,76%	62,23%	56,37%	35,14%

The results in Table 7 are the testing of LSTM and Bi-LSTM algorithms that have not used the word embedding technique and are still far from perfect. This is the basic capital to continue the next implementation using word embedding. The testing of LSTM and Bi-LSTM algorithms using word embedding technique has a testing scheme listed in Table 5. The results of the LSTM algorithm testing are presented in Figure 7, where LSTM + FastText embedding with a dimension size of 300 and epoch 20 obtained the highest accuracy results from other schemes, reaching 56.00%. Meanwhile, the testing results of the Bi-LSTM + Word2vec Embedding algorithm achieved the best result of 56.56% from embedding size 300 and epoch 30, this is shown in Figure 8.

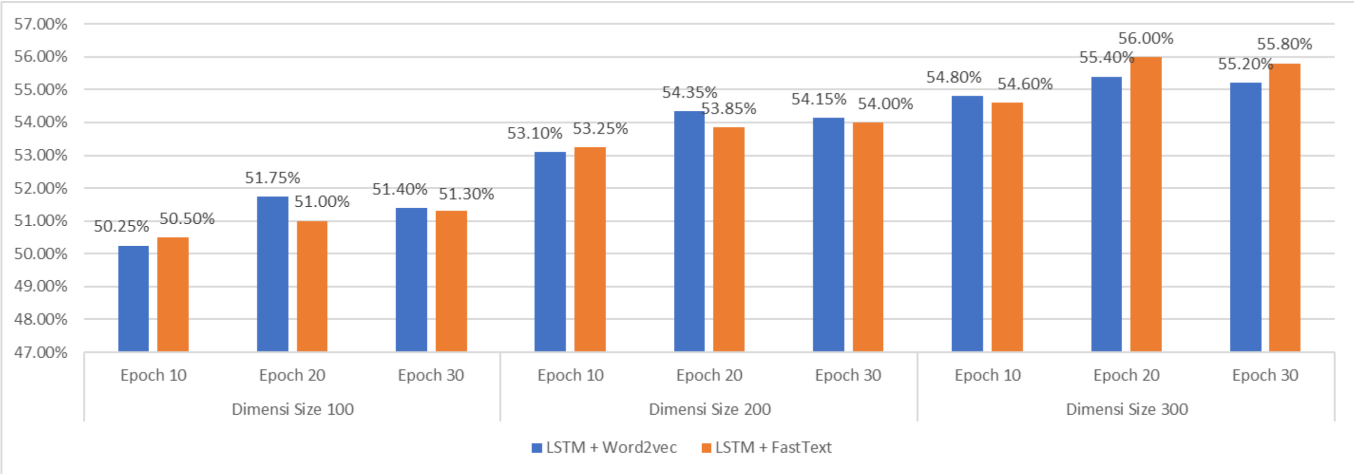


Figure 7. LSTM Results + Word Embedding

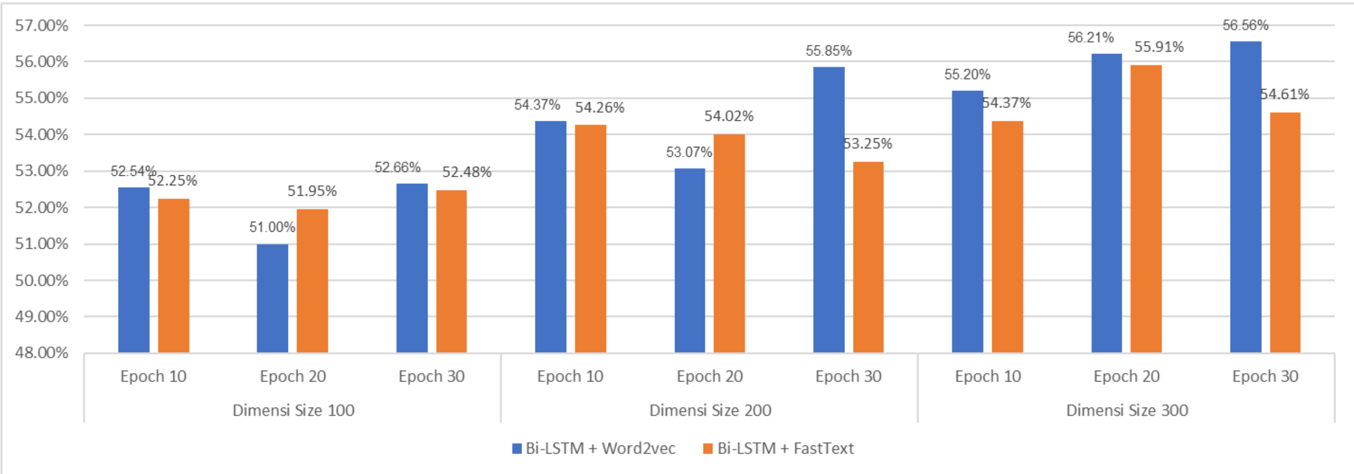
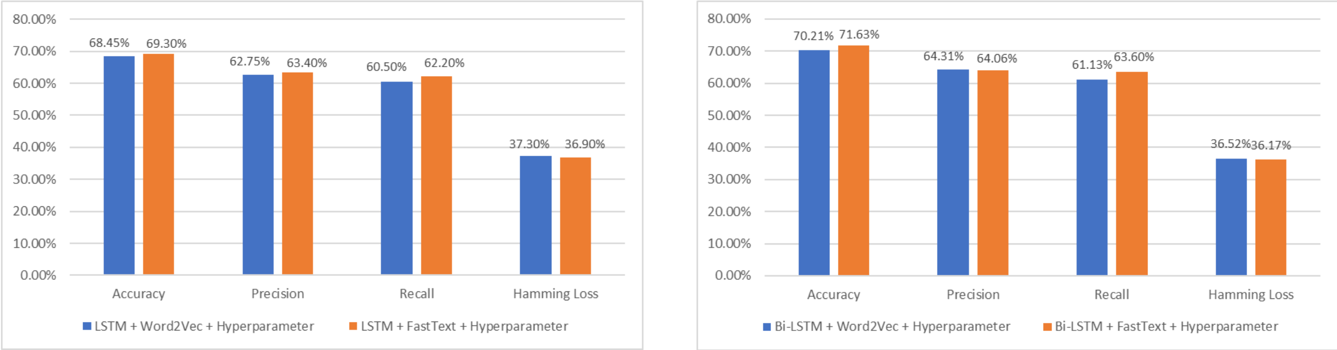


Figure 8. Bi-LSTM Results + Word Embedding

The results from word embedding are still not as expected, and the results of the Indonesian translation classification of the Qur'an are still unable to surpass those from research [7]. Therefore, this study implements hyperparameters. Hyperparameter tuning is based on the best word embedding results for each model. Parameters set with hyperparameters include epochs with a range of 10-100, batch size in the range of 16 – 128, and learning rate in this study ranges from 0.00001 (1e-5) to 0.1 (1e-1). By producing the best parameters of 96 epochs, batch size of 107, and learning rate of 0.09285776391334735.



(a) *(b)*

Figure 9. Test results with Hyperparameters (a) LSTM, (b) Bi-LSTM

Based on Figure 9 (a), the testing of LSTM + FastText + hyperparameters achieved the best result with an accuracy of 69.30%, whereas Figure 9 (b) shows that Bi-LSTM + FastText + hyperparameters had the best result among several tests with an accuracy value of 71.63%. This can be concluded that this research outperforms the research conducted previously by [7] although the accuracy results are not significantly higher. In another analysis, as illustrated in Figure 9, Bi-LSTM combined with Word2Vec reached an accuracy of 70.21% and a precision of 64.31%, while its recall was slightly lower at 61.13%. On the other hand, when FastText was used, there was a slight increase in overall accuracy to 71.63% and yield to 63.60%, but precision slightly decreased to 64.06%. This trade-off in precision can be associated with FastText's ability to generate better embeddings for rare or out-of-vocabulary words due to its subword approach. While this characteristic of FastText improves the model's ability to correctly identify more relevant instances (thus higher recall), it can also increase the likelihood of false positives, which may slightly lower precision. The fundamental difference between these two embedding methods underscores the importance of considering specific use case scenarios when selecting the appropriate technique. As shown in the optimization efforts, the combination of Bi-LSTM with FastText and hyperparameter tuning emerged as the superior method, offering a balance between accurately identifying relevant instances and minimizing false classifications, as evidenced by its performance metrics.

4. Conclusion

Multi-label classification of the Quran translation in Indonesian using Long Short-Term Memory (LSTM) and Bidirectional LSTM (Bi-LSTM) models, focusing on the surahs An-Nisa', Al-Maidah, and Al-An'am. Results show that the combination of Bi-LSTM with FastText and hyperparameter tuning yields the best performance, achieving an accuracy of 71.63%. The strength of the proposed method lies in the use of the Bi-LSTM model combined with FastText for multi-label Quran translation classification, which shows a significant accuracy improvement. This approach takes advantage of Bi-LSTM's ability to understand text context better compared to the standard LSTM model. However, this method also has weaknesses, including dependency on a large volume of data for effective training and difficulty in handling semantic ambiguity in religious texts. Future research is directed towards the development of further word embedding techniques, model architecture adjustments, and data pre-processing techniques to overcome these weaknesses and improve performance in religious text classification.

References

[1] S. Wu, Y. Chen, Z. Li, J. Li, F. Zhao, and X. Su, “Towards multi-label classification: Next step of machine learning for microbiome research,” *Computational and Structural Biotechnology Journal*, vol. 19. Elsevier B.V., pp. 2742–2749, Jan. 01, 2021. https://doi.org/10.1016/j.csbj.2021.04.054

[2] Y. Guo, F. L. Chung, G. Li, and L. Zhang, “Multi-Label Bioinformatics Data Classification with Ensemble Embedded Feature Selection,” *IEEE Access*, vol. 7, pp. 103863–103875, 2019. https://doi.org/10.1109/ACCESS.2019.2931035

[3] Z. M. Chen, X. S. Wei, P. Wang, and Y. Guo, “Multi-label image recognition with graph convolutional networks,” in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, IEEE Computer Society, Jun. 2019, pp. 5172–5181. https://doi.org/10.1109/CVPR.2019.00532

[4] H. K. Maragheh, F. S. Gharehchopogh, K. Majidzadeh, and A. B. Sangar, “A New Hybrid Based on Long Short-Term Memory Network with Spotted Hyena Optimization Algorithm for Multi-Label Text Classification,” *Mathematics*, vol. 10, no. 3, Feb. 2022. https://doi.org/10.3390/math10030488

[5] G. Liu and J. Guo, “Bidirectional LSTM with attention mechanism and convolutional layer for text classification,” *Neurocomputing*, vol. 337, pp. 325–338, Apr. 2019. https://doi.org/10.1016/j.neucom.2019.01.078

[6] F. Amalia and M. Arif Bijaksana, “Semantic Model Evaluation Dataset For Indonesian In Al-Qur’an Vocabulary: Similarity And Relatedness,” 2020. https://doi.org/10/25047/jtit.v7i1.137

[7] M. Fauzan, H. Junaedi, and E. Setyati, “KLASIFIKASI AL – QUR’AN TERJEMAHAN BAHASA INDONESIA DENGAN MENGGUNAKAN ALGORITMA SUPPORT VECTOR MACHINE (SVM),” *KONVERGENSI*, vol. 18, no. 2, pp. 42–49, Dec. 2022. https://doi.org/10.30996/konv.v18i1.6912

[8] M. Robani and A. Widodo, “Algoritma K-Means Clustering Untuk Pengelompokan Ayat Al Quran Pada Terjemahan Bahasa Indonesia,” *Jurnal Sistem Informasi Bisnis*, vol. 6, no. 2, p. 164, 2016,. https://doi.org/10.21456/vol6iss2pp164-176

[9] M. Irfan, W. Uriawan, N. Lukman, O. Kurahman, and W. Darmalaksana, “The Qur’anic Classification Uses Algorithm C4.5,” 2020. http://dx.doi.org/10.4108/eai.2-10-2018.2295558
[10] A. Hanafi, A. Adiwijaya, and W. Astuti, “Klasifikasi Multi Label pada Hadis Bukhari Terjemahan Bahasa Indonesia Menggunakan Mutual Information dan k-Nearest Neighbor,” *Jurnal Sisfokom (Sistem Informasi dan Komputer)*, vol. 9, no. 3, pp. 357–364, 2020. http://dx.doi.org/10.32736/sisfokom.v9i3.980
[11] T. H. Putrisanni, A. Adiwijaya, and S. Al Faraby, “Klasifikasi Ayat Al-Quran Terjemahan Bahasa Inggris Menggunakan K-Nearest Neighbor (Knn) Dan Information Gain,” *KOMIK (Konferensi Nasional Teknologi Informasi dan Komputer)*, vol. 3, no. 1, pp. 362–369, 2019. https://doi.org/10.30865/komik.v3i1.1614
[12] A. Abdullahi, N. A. Samsudin, M. H. A. Rahim, S. K. A. Khalid, and R. Efendi, “Multi-label classification approach for Quranic verses labeling,” *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 24, no. 1, pp. 484–490, 2021. http://doi.org/10.11591/ijeecs.v24.i1.pp484-490
[13] B. A. H. Kholifatullah and A. Prihanto, “Penerapan Metode Long Short Term Memory Untuk Klasifikasi Pada Hate Speech,” *Journal of Informatics and Computer Science (JINACS)*, vol. 04, pp. 292–297, Jan. 2023. https://doi.org/10.26740/jinacs.v4n03.p292-297
[14] D. I. Af’idah, D. Dairoh, S. F. Handayani, and R. W. Pratiwi, “Pengaruh Parameter Word2Vec terhadap Performa Deep Learning pada Klasifikasi Sentimen,” *Jurnal Informatika: Jurnal Pengembangan IT*, vol. 6, no. 3, pp. 156–161, 2021. https://dx.doi.org/10.30591/jpit.v6i3.3016
[15] R. B. Afrianto and L. Y. Kurniawati, “Kategorisasi Dokumen Teks Secara Multi Label Menggunakan Fuzzy C-Means Dan K-Nearest Neighbors Pada Artikel Berbahasa Indonesia,” *JUTI: Jurnal Ilmiah Teknologi Informasi*, vol. 11, no. 1, p. 23, 2013. https://dx.doi.org/10.12962/j24068535.v11i1.a17
[16] S. Sayyida, “Ayat-Ayat Tauhid Terhadap Budaya Pemeliharaan Keris Di Jawa (Studi Kasus Buku Mt Arifin),” *Journal of Qur’an and Hadith Studies*, vol. 6, no. 1, pp. 24–52, 2019. https://doi.org/10.15408/quhas.v6i1.13403
[17] A. Kallang, “Konteks Ibadah Menurut Al-Quran,” *Al-Din: Jurnal Dakwah dan Sosial Keagamaan*, vol. 4, no. 2, pp. 1–13, 2018. http://dx.doi.org/10.35673/ajdsk.v4i2.630
[18] M. Murharyana, I. I. Al Ayyubi, and R. Rohmatulloh, “Pendidikan Akhlak Anak Kepada Orang Tua Dalam Perspektif Al-Quran,” *Piwulang: Jurnal Pendidikan Agama Islam*, vol. 5, no. 2, pp. 175–191, 2023. http://dx.doi.org/10.32478/piwulang.v5i2.1515
[19] J. Mirdad and S. Rahmat, “Sejarah Dalam Perspektif Islam,” *El -Hekam*, vol. 6, no. 1, p. 9, 2021. https://doi.org/10.31958/jeh.v6i1.3303
[20] S. Bessou and R. Aberkane, “Subjective Sentiment Analysis for Arabic Newswire Comments,” *Journal of Digital Information Management*, vol. 17, no. 5, p. 289, 2019. https://doi.org/10.48550/arXiv.1911.03776
[21] A. A. Firdaus, A. Yudhana, and I. Riadi, “Public Opinion Analysis of Presidential Candidate Using Naïve Bayes Method,” *Kinetik: Game Technology, Information System, Computer Network, Computing, Electronics, and Control*, vol. 4, no. 2, 2023. https://doi.org/10.22219/kinetik.v8i2.1686
[22] E. Y. Sari, A. D. Wierfi, and A. Setyanto, “Sentiment Analysis of Customer Satisfaction on Transportation Network Company Using Naive Bayes Classifier,” *2019 International Conference on Computer Engineering, Network, and Intelligent Multimedia, CENIM 2019 - Proceeding*, vol. 2019-Novem, 2019. https://doi.org/10.1109/CENIM48368.2019.8973262
[23] H. A. Almuzaini and A. M. Azmi, “Impact of Stemming and Word Embedding on Deep Learning-Based Arabic Text Categorization,” *IEEE Access*, vol. 8, pp. 127913–127928, 2020. https://doi.org/10.1109/ACCESS.2020.3009217
[24] Y. D. Prabowo, T. L. Marselino, and M. Suryawiguna, “Pembentukan Vector Space Model Bahasa Indonesia Menggunakan Metode Word to Vector,” *Jurnal Buana Informatika*, vol. 10, no. 1, p. 29, 2019. https://doi.org/10.24002/jbi.v10i1.2053
[25] E. H. Mohamed and W. H. El-Behaidy, “An Ensemble Multi-label Themes-Based Classification for Holy Qur’an Verses Using Word2Vec Embedding,” *Arabian Journal for Science and Engineering*, vol. 46, no. 4, pp. 3519–3529, 2021. https://doi.org/10.1007/s13369-020-05184-0
[26] J. Hermanto, “Klasifikasi Teks Humor Bahasa Indonesia Memanfaatkan SVM,” *Journal of Information System,Graphics, Hospitality and Technology*, vol. 3, no. 01, pp. 39–48, 2021. https://doi.org/10.37823/insight.v3i01.118
[27] L. Xiao, G. Wang, and Y. Zuo, “Research on Patent Text Classification Based on Word2Vec and LSTM,” *Proceedings - 2018 11th International Symposium on Computational Intelligence and Design, ISCID 2018*, vol. 1, pp. 71–74, 2018. https://doi.org/10.1109/ISCID.2018.00023
[28] R. P. Hastuti, V. Riona, and M. Hardiyanti, “Content Retrieval dengan Fasttext Word Embedding pada Learning Management System Olimpiade,” *Journal of Internet and Software Engineering*, vol. 4, no. 1, pp. 18–22, 2023. https://doi.org/10.22146/jise.v4i1.6766
[29] A. Nurdin, B. Anggo Seno Aji, A. Bustamin, and Z. Abidin, “Perbandingan Kinerja Word Embedding Word2Vec, Glove, Dan Fasttext Pada Klasifikasi Teks,” *Jurnal Tekno Kompak*, vol. 14, no. 2, p. 74, 2020. https://doi.org/10.33365/jtk.v14i2.732
[30] Y. Taher, A. Moussaoui, and F. Moussaoui, “Automatic Fake News Detection based on Deep Learning, FastText and News Title,” *International Journal of Advanced Computer Science and Applications*, vol. 13, no. 1, pp. 146–158, 2022. https://dx.doi.org/10.14569/IJACSA.2022.0130118
[31] A. Saifudin, “Metode Data Mining Untuk Seleksi Calon Mahasiswa,” vol. 10, no. 1, pp. 25–36, 2018, doi: https://dx.doi.org/10.24853/jurtek.10.1.25-36.
[32] M. Kim and K.-H. Kang, “Comparison of Neural Network Techniques for Text Data Analysis,” *International Journal of Advanced Culture Technology*, vol. 8, no. 2, pp. 231–238, 2020. https://doi.org/10.17703/IJACT.2020.8.2.231
[33] X. H. Le, H. V. Ho, G. Lee, and S. Jung, “Application of Long Short-Term Memory (LSTM) neural network for flood forecasting,” *Water (Switzerland)*, vol. 11, no. 7, 2019. http://dx.doi.org/10.3390/w11071387
[34] M. T. Vu, A. Jardani, N. Massei, and M. Fournier, “Reconstruction of missing groundwater level data by using Long Short-Term Memory (LSTM) deep neural network,” *Journal of Hydrology*, vol. 597, 2021. https://doi.org/10.1016/j.jhydrol.2020.125776
[35] A. M. Ertugrul and P. Karagoz, “Movie Genre Classification from Plot Summaries Using Bidirectional LSTM,” *Proceedings - 12th IEEE International Conference on Semantic Computing, ICSC 2018*, vol. 2018-Janua, pp. 248–251, 2018. https://doi.org/10.1109/ICSC.2018.00043
[36] A. R. Isnain, A. Sihabuddin, and Y. Suyanto, “Bidirectional Long Short Term Memory Method and Word2vec Extraction Approach for Hate Speech Detection,” *IJCCS (Indonesian Journal of Computing and Cybernetics Systems)*, vol. 14, no. 2, p. 169, 2020. https://doi.org/10.22146/ijccs.51743
[37] A. A. Ningrum, I. Syarif, A. I. Gunawan, E. Satriyanto, and R. Muchtar, “Algoritma Deep Learning-Lstm Untuk Memprediksi Umur Deep Learning-Lstm Algorithm for Power Transformer Lifetime,” *Jurnal Teknologi Informasi dan Ilmu Komputer*, vol. 8, no. 3, pp. 539–548, 2021. https://doi.org/10.25126/jtiik.202184587