ICIC Express Letters
Volume **18**, Number **6**, June **2024**

ICIC International ©2024 ISSN 1881-803X
pp. **541–550**

MULTI-CLASS TEXT CLASSIFICATION ON KHMER NEWS ARTICLES USING DEEP LEARNING MODELS WITH OPTIMAL HYPERPARAMETERS

Raksmey Phann, Chitsutha Soomlek, Pongsathon Janyoi and Pusadee Seresangtakul*

Natural Language and Speech Processing Laboratory
College of Computing
Khon Kaen University
123 Village No. 16, Mittraphap Road, Nai-Muang, Muang District, Khon Kaen 40002, Thailand
raksmey.p@kkumail.com; { chitsutha; pongsathon }@kku.ac.th
*Corresponding author: pusadee@kku.ac.th

Received September 2023; accepted November 2023

Abstract. *Organizing news articles into different categories based on their content is essential for news organizations to make their content easier to find, analyze, and understand for their readers. However, manually categorizing news articles is a tedious and error-prone task, making automated approaches more appealing. This study focused on investigating the use of various models for performing multi-class text classification on Khmer news articles. The models used in this study included a deep neural network (DNN) model, a fastText supervised model, and a recurrent neural network (RNN) model with an attention mechanism, where the DNN and RNN models use the term frequency-inverse document frequency (TF-IDF) method to represent the Khmer news articles. However, there is limited availability of annotated datasets for NLP tasks in Khmer, which led the researchers to collect a dataset of Khmer news articles for this task. The news articles were preprocessed by removing spaces, numbers, English words, punctuation, and performing word segmentation. The models were then trained and evaluated on a dataset of Khmer news articles annotated with six different class labels: sports, politics, technology, social, health, and environment. The study found that the fastText supervised model with optimal hyperparameters was the best performing model for multi-class text classification on Khmer news articles, achieving an overall accuracy of 93.81% with high levels of precision, recall, and F1-score. This automated approach has the potential to significantly improve the efficiency of Khmer news content management by automating the categorization of large quantities of news articles.*
Keywords: fastText, Neural network, Text classification, TF-IDF

1. **Introduction.** By using deep learning models for multi-class text classification, it is possible to automatically categorize Khmer news articles into different categories, such as politics, social, technology, health, sports, or environment. Since the majority of text classification methods at present focus on classifying texts with a single label. This can make it easier for readers to find the articles that are most relevant to them, and it can also be used to help news organizations better understand and analyze the content they produce.

One approach to text classification is to use neural network model, which can learn complex patterns in the Khmer news articles and make accurate predictions based on these patterns [1]. Text classification on Khmer news articles is a challenging task due to the language's unique characteristics. However, machine learning techniques have advanced significantly, enabling accurate text classification on Khmer news articles using the fastText supervised model [2]. It uses a combination of word embeddings and a linear

DOI: 10.24507/icicel.18.06.541

classifier to classify text documents into predefined categories. The training of fastText is fast and precisely represents unseen words [3]. However, the text classification depends on many factors such as feature engineering and modeling. This research has made several important contributions to the field of natural language processing (NLP) for the Khmer language.

- Collect Khmer news articles for text classification and corpus construction.
- Investigate the use of various models for performing multi-class text classification on Khmer news articles. This provided valuable insights into the strengths and limitations of these techniques for Khmer text and help to inform the choice of technique for future research in this area.
- Explore on the difference text representation with different models and find that the fine-turning fastText model achieved the best performance.

The following sections of the paper are organized as follows. In Section 2, the background and related work are provided. In Section 3, we explain the research methodology, data preparation, and the explored models. Section 4 mentions the experiment setup and results, followed by a conclusion in Section 5.

2. **Literature Review.** News articles are an important source of information, entertainment, and education. Researchers have been using deep learning algorithms to classify news articles into predefined categories. Multi-class text classification algorithms are used to analyze the content of news articles and assign them to the appropriate category.

2.1. **Multi-class text classification.** The ability of machine learning algorithms to handle multi-class text classification problems has been demonstrated in various studies. One such study by [4] focused on text classification for the Khmer language using neural networks with word embedding. Their dataset consisted of 13,902 articles, of which 4,687 had a single label, and they achieved an impressive precision of 87.8% using an RNN model for multi-label classification.

Another study by [5] introduced a novel approach to classifying news text using a customized deep learning model called DCLSTM-MLP. This model combined convolutional neural network (CNN), long short-term memory (LSTM), and multilayer perceptron (MLP) algorithms and utilized word vectors and discrete vectors to represent relationships between words and categories. As a result, it effectively solved problems with text length and feature extraction, outperforming other models in terms of accuracy, recall rate, and overall performance.

[6] also proposed a K-Nearest Neighbor classifier for identifying news item types in a multiclass dataset structured by Nurfikri [7]. They employed pre-processing techniques such as case folding, data cleaning, stop-word removal, and tokenization, and used feature TF-IDF with 10-fold cross-validation for their K-Nearest Neighbor classification model. They achieved a high level of precision of 87.48%.

While these studies have shown promise for text classification on news articles written in different languages, further research is necessary to explore the use of these models and compare their performance with feature extraction methods on such as TF-IDF and the continue bag of word (CBOW) with fastText model on Khmer news articles specifically.

2.2. **Hyperparameter optimization.** Several studies have been conducted in the past on hyperparameter optimization techniques, notably in the context of machine learning. [8, 9, 10] provided valuable insights into various hyperparameter optimization techniques, including grid search, random search, Bayesian optimization, evolutionary algorithms, and bandit-based methods. In particular, [8] showed that random search is a more efficient strategy for hyperparameter optimization compared to grid search and manual search, while [9] introduced a novel approach that combines the advantages of Bayesian optimization and bandit-based methods, offering superior performance and rapid convergence to optimal configurations.

Moreover, [10] provided a comprehensive introduction to hyperparameter optimization techniques and compared the performance of different optimization methods, with results showing that Bayesian optimization and evolutionary algorithms outperform grid search and random search in terms of efficiency and effectiveness.

Given the importance of hyperparameter optimization in developing effective machine learning models, it is relevant to apply these techniques to Khmer news clarification. By optimizing hyperparameters, it is possible to improve the accuracy and efficiency of natural language processing algorithms used for Khmer news clarification, leading to better performance and more accurate results.

3. **Research Methodology.** In this section, we demonstrate how to use multi-class text classification on Khmer news articles using deep learning models. Our approach is organized into six primary sections, including data collection, text preprocessing, feature extraction, hyperparameters optimization, a deep learning model and evaluation results section. Figure 1 illustrates an overview of the proposed methodology.

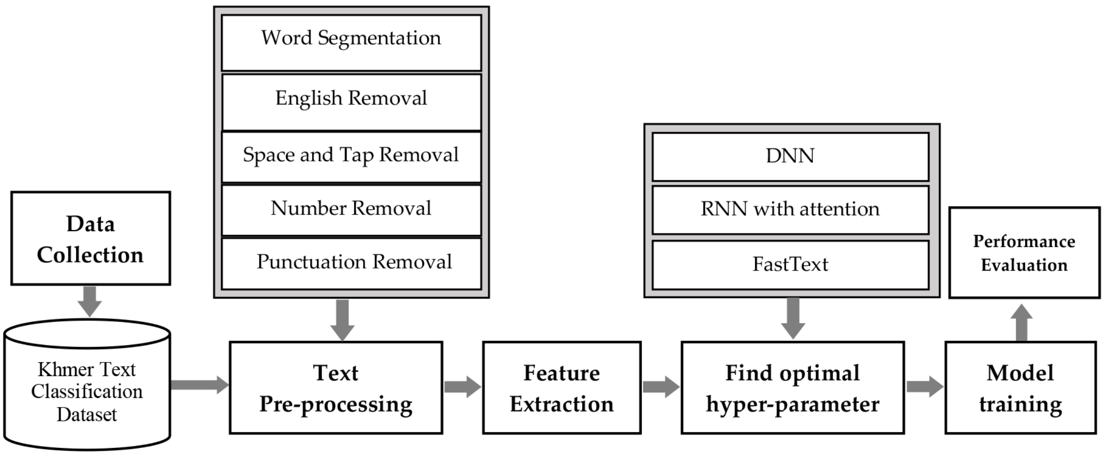

Figure 1. An overview of the proposed methodology

3.1. **Data collection.** In order to build a balanced dataset for the Khmer language, Khmer news articles were collected from three different sources: Sabay news website (news.sabay.com.kh), Radio Free Asia website (www.rfa.org), and VOA websites (khmer.voanews.com). We scraped 15,205 Khmer news articles with six news categories. Figure 2 displays the number of news articles for each category. The data was gathered from several websites. Social and political contents were collected from VOA website while entertainment and health-related contents were obtained from Radio Free Asia website. Technology and sports categories were collected from Sabay news website. To scrape the title and content of Khmer news articles from news websites, we used the BeautifulSoup library, which is a popular Python® library for parsing HTML and XML documents.

3.2. **Data pre-processing.** Data pre-processing is an important step in the analysis and modeling of datasets, as it can help to eliminate outliers and nonsensical content that could negatively impact the validity of the analysis [11]. In the case of Khmer news articles, data pre-processing can be particularly important, as the dataset may contain spaces, numbers, English words, and punctuation that can interfere with the accuracy of Khmer word tokenization. To address these issues, we implemented a data pre-processing

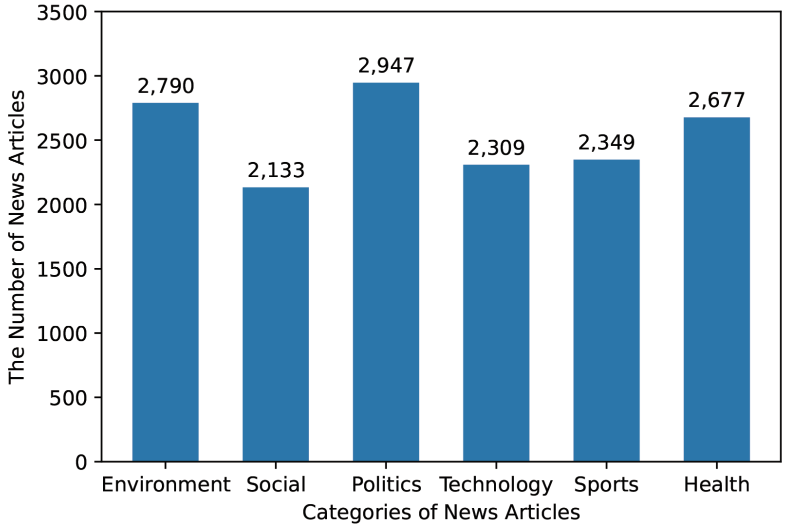

FIGURE 2. The number of news articles for each category

pipeline that consists of five steps. Firstly, we used regular expressions to eliminate both Khmer and English numbers from the Khmer news articles. This helps to ensure that the tokenization process is not affected by the presence of numerical data. Secondly, we utilized Python® programming's "replace" function to convert all spaces to blanks. This helps to eliminate any extraneous spaces that might be present in the data and could interfere with the tokenization process. Thirdly, to eliminate all English words from the Khmer news articles, we applied a regular expression. This helps to ensure that only Khmer words are included in the tokenization process. Fourthly, we again utilized Python® programming's "replace" function to convert all Khmer and English punctuation to blanks. This helps to eliminate any punctuation that might be present in the data and could interfere with the tokenization process. Finally, conditional random fields [12] was used to tokenize Khmer words. Table 1 shows samples of Khmer news articles. Table 2 shows example of data pre-processing.

TABLE 1. Sample of Khmer news articles

Title	Text	Category
តិចនិកសំខាន់៥ការពារលុយក្នុងកាត មិនឱ្យបាត់បង់ដោយមិនដឹងខ្លួន... (Here are 5 important techniques to protect your money from being lost unknowingly on...)	មើលកម្មវិធីឱ្យបានច្បាស់លាស់មុនចុចប្រើប្រាស់ ឬមុនចុចអនុញ្ញាតឱ្យកម្មវិធីទាំង នោះអាចកាត់លុយបាន ... (Take a closer look at the software before using it, or before clicking, as it may deduct money...)	Technology
ដីស្ត្រីរងគ្រោះដោយគ្រូពេទ្យវះកាត់ ដាច់ពោះវៀនសម្រេចប្ដឹង មន្ទីរសំរាកព្យាបាលបង្កបញ្ហានោះ... (The woman whose land was affected by the intestinal...)	ប្ដីស្ត្រីជាកម្មកររោងចក្រដែលត្រូវបានគ្រូពេទ្យវះកាត់ដាច់ ពោះវៀនសម្រេចប្ដឹងមន្ទីរសំរាកព្យាបាលឯកជន ដើម្បីរកយុត្តិធម៌។... (The husband and wife of a factory worker who was sued by an intestinal surgeon have decided to...)	Health

3.3. **Feature extraction.** Feature extraction is the process of identifying and extracting the most relevant and informative features from a dataset. In the context of text classification, feature extraction involves identifying the most important words or phrases in

Table 2. Example of data pre-processing

Step	The output from each step of text pre-processing
Raw text	មើលកម្មវិធីឱ្យបានច្បាស់លាស់មុនចុចប្រើប្រាស់ឬមុនចុចអនុញ្ញាតឱ្យកម្មវិធីទាំងនោះ អាចកាត់លុយ អ្នក ចាត់ការ ក្លឹប Manchester United លោក Erik Ten Hag បាន ទម្លាយ ប្រាប់ អ្នក កាសែត កាល ពី ថ្ងៃ អាទិត្យ ថា ខ្សែ ប្រយុទ្ធ ព័រទុយហ្គាល់ កីឡាករ Cristiano Ronaldo ជា ជម្រើស ទី៤ នៃ បេក្ខភាព ប្រធាន ក្រុម ។ ពត៌មានចេញផ្សាយនៅ ឆ្នាំ 2010។ តើអ្នកបានអានឬនៅ ?
Space and tap removal	អ្នកចាត់ការក្លឹបManchesterUnitedលោកErikTenHagបានទម្លាយប្រាប់អ្នកកាសែត កាលពីថ្ងៃអាទិត្យថាខ្សែប្រយុទ្ធព័រទុយហ្គាល់កីឡាករCristianoRonaldoជាជម្រើសទី៤ នៃបេក្ខភាពប្រធានក្រុម។ ពត៌មានចេញផ្សាយនៅឆ្នាំ2010។តើអ្នកបានអានឬនៅ?
Number removal	អ្នកចាត់ការក្លឹបManchesterUnitedលោកErikTenHagបានទម្លាយប្រាប់អ្នកកាសែត កាលពីថ្ងៃអាទិត្យថាខ្សែប្រយុទ្ធព័រទុយហ្គាល់កីឡាករCristianoRonaldoជាជម្រើសទី នៃបេក្ខភាពប្រធានក្រុម។ ពត៌មានចេញផ្សាយនៅឆ្នាំ។តើអ្នកបានអានឬនៅ?
Punctuation removal	អ្នកចាត់ការក្លឹបManchesterUnitedលោកErikTenHagបានទម្លាយប្រាប់អ្នកកាសែត កាលពីថ្ងៃអាទិត្យថាខ្សែប្រយុទ្ធព័រទុយហ្គាល់កីឡាករCristianoRonaldo ជាជម្រើសទីនៃបេក្ខភាពប្រធានក្រុមពត៌មានចេញ ផ្សាយនៅឆ្នាំតើអ្នកបានអានឬនៅ
English removal	អ្នកចាត់ការក្លឹបលោកបានទម្លាយប្រាប់អ្នកកាសែតកាលពីថ្ងៃអាទិត្យថាខ្សែ ប្រយុទ្ធព័រទុយហ្គាល់កីឡាករជាជម្រើសទីនៃបេក្ខភាពប្រធានក្រុមពត៌មានចេញផ្សាយ នៅឆ្នាំតើអ្នកបានអានឬនៅ
Word segmentation	អ្នកចាត់ការ ក្លឹប លោក បាន ទម្លាយ ប្រាប់ អ្នកកាសែត កាល ពី ថ្ងៃអាទិត្យ ថា ខ្សែ ប្រយុទ្ធព័រ ទុយហ្គាល់ កីឡាករ ជា ជម្រើស ទី នៃ បេក្ខភាព ប្រធាន ក្រុម ពត៌មាន ចេញ ផ្សាយនៅ ឆ្នាំ តើ អ្នក បាន អាន ឬ នៅ

the text that will be used to make predictions about the class label of the text. Term frequency-inverse document frequency (TF-IDF) is a feature extraction method, which is based on the idea that words that are more frequent in a document but less frequent in other documents are more important for classifying the document. To use TF-IDF for multi-class text classification on Khmer news articles, we tokenized Khmer news articles to create a list of words that can be used to calculate the TF-IDF values. The term frequency (TF) of a word is the number of times it appears in a document, divided by the total number of words in the document. This can be expressed as

$$TF(t,d) = \frac{Number\ of\ occurrences\ of\ t\ in\ d}{Total\ number\ of\ words\ in\ d} \tag{1}$$

where t stands for the word, and d for each document. The inverse document frequency (IDF) of a word is the logarithm of the total number of documents in the dataset divided by the number of documents that contain the word. This can be expressed as

$$IDF(t) \ = \log\left(\frac{N}{Number\ of\ documents\ containing\ t}\right) \tag{2}$$

where N is the total number of documents in the dataset. The TF-IDF value of a word is calculated by multiplying the term frequency and inverse document frequency of the word. This can be expressed as

$$TF\text{-}IDF(t,d) = TF(t,d) * IDF(t) \tag{3}$$

Using the TF-IDF values of the words, we created a feature matrix where each row represents a Khmer news article, and each column represents a Khmer word. The value in each cell of the matrix is the TF-IDF value of the word for the corresponding Khmer news article. The feature matrix is used as the input for training the model.

3.4. **fastText supervised model.** The fastText supervised model is a machine learning algorithm that is specifically designed for text classification [3]. It can handle large datasets in a short amount of time, making it a useful tool for text classification tasks. The fastText is able to effectively handle long text in our Khmer news dataset. When working with long text inputs, fastText split the input into multiple chunks and process them separately. The final prediction is obtained by averaging the predictions for each chunk. This allows fastText to scale to long texts without a significant increase in computation time. The main architecture used in fastText is the continuous bag-of-words (CBOW) architecture, which predicts a word from its surrounding context. The CBOW employs a neural network architecture. The input layer takes in a sequence of words (the context) and converts them into word embeddings, which are dense numerical representations of the words [13] as shown in Figure 3.

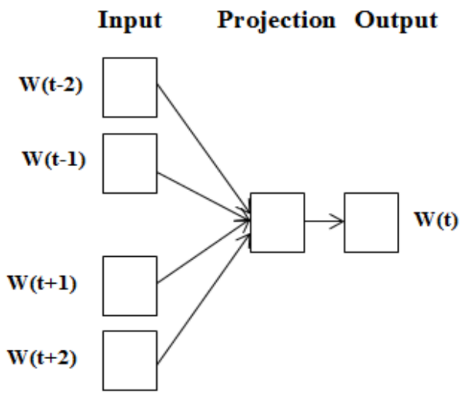

Figure 3. Overview of CBOW architecture

In fastText, n-gram features are also used to represent words in the text as vectors of their constituent n-grams. The fastText uses a sliding window approach to extract n-grams from text, meaning that it considers all possible n-grams within a specified window size.

3.5. **Deep neural network model.** A deep neural network (DNN) is a type of deep learning models that is composed of multiple layers of artificial neurons [14], which are inspired by the structure and function of neurons in the human brain. The DNN model can be used for a variety of tasks, including multi-class text classification on Khmer news articles. The architecture of a DNN refers to the structure and organization of the layers and neurons within the network. We implemented a DNN for text classification on Khmer news articles. There are several factors to consider achieving good performance such as the input layer, the number of hidden layers and hidden units. The loss function and optimizer also influence the performance of the model. This work uses tanh activation function for the hidden layers and the softmax activation function for the output layer. The categorical-cross entropy was taken as loss function.

3.6. **Recurrent neural network model with attention mechanisms.** RNN is a type of artificial neural networks that is particularly well-suited for processing sequential data [15], such as time series or natural language. Unlike traditional feedforward neural networks, which process input data in a single pass through the network, RNNs have feedback connections that allow information to be passed from one step of the sequence to the next. In recent years, attention mechanisms have emerged as a powerful tool for improving the performance of RNN models. The basic idea behind attention is to allow the model to selectively focus on certain parts of the input sequence, based on their relevance to the task at hand. This allows the model to effectively handle long-term dependencies in the input sequence, without relying solely on the hidden state of the RNN. One of the earliest and most well-known applications of attention in RNNs is the encoder-decoder architecture for machine translation. In this architecture, an RNN-based encoder processes the input sequence and produces a fixed-length vector representation, which is then used by a decoder RNN to generate the output sequence. However, instead of using the final hidden state of the encoder as the input to the decoder, attention mechanisms are used to dynamically select different parts of the input sequence as needed.

3.7. **Hyperparameters optimization.** Hyperparameter optimization is a crucial step in building high-performing machine learning models, and there are several methods available to explore the space of possible hyperparameters. This research used the Bayesian optimization. It is a probabilistic approach that uses a surrogate model to approximate the objective function [16], which in our case was the accuracy of the models on the Khmer language data. The surrogate model is updated after each iteration using the acquired data, which includes both the objective function evaluations and the corresponding hyperparameter values. The updated surrogate model is then used to decide the next hyperparameter configuration to evaluate, using an acquisition function that balances exploration and exploitation. In this way, Bayesian optimization is able to efficiently search the hyperparameter space while gradually converging to the optimal hyperparameters. To implement Bayesian optimization, we used the "BayesianOptimization" object from the "bayes_opt" library in Python. We first defined the parameter space by specifying the range of values for each hyperparameter and any constraints that were placed on their values. We then called the "maximize" method to run the optimization, which involved iteratively evaluating the models with different hyperparameter values and updating the surrogate model to guide the search for optimal hyperparameters. One important aspect of Bayesian optimization is the choice of the surrogate model. In our research, we used a Gaussian process as the surrogate model, which is a popular choice for Bayesian optimization due to its flexibility and ability to capture complex relationships between hyperparameters and the objective function.

4. **Experiment and Result.** We trained models on a dataset of 15,205 Khmer news articles using 10-fold cross-validation. We performed 10-fold cross-validation on the dataset for all three models, where the models were trained on nine folds and evaluated on the remaining fold. We utilized a variety of metrics in Scikit-Learn framework [17] to evaluate model's performance, such as accuracy, precision, and recall which can be calculated using the following formulas:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{4}$$

$$Precision = \frac{TP}{TP + FP} \tag{5}$$

$$Recall = \frac{TP}{TP + FN} \tag{6}$$

$$F1\text{-}score = \frac{2 * (Precision * Recall)}{Precision + Recall} \tag{7}$$

where FP represents the count of false positives, TP stands for true positives, TN indicates true negatives, and FN signifies false negatives.

4.1. **Bayesian optimization.** We start by defining the search space for the hyperparameters to optimize. We used a BayesOpt library [18] to find the optimal hyperparameters for each model. The Bayesian Optimization library then samples hyperparameters from this search space to train and evaluate the models on the validation set. Based on the evaluation metrics, the Bayesian Optimization library updates its probabilistic model of the search space and generates a new set of hyperparameters to sample. This process is repeated for a set number of iterations, for our experiment, 50. At the end of the optimization process, the library returns the set of hyperparameters that resulted in the best validation accuracy for each model as shown in Table 3.

Table 3. Optimal hyperparameter of each model

Model	Optimal hyperparameters
DNN	learning rate = 0.0003, batch size = 198, dropout rate = 0.1 with 6 layers.
fastText	epoch = 49, lr = 0.1, dim = 98, word_ngrams = 3, loss = softmax, min_count = 2, bucket = 1900000, thread = 4
RNN with attention mechanism	Dense units = 128, Attention mechanism = Dot product, Activation function = Softmax, Optimizer = Adam, Maximum sequence length = 200

4.2. **Deep neural network model.** We designed the DNN model with three four layers and used the tanh activation function. To train the model, we used the Adam optimization algorithm, which is known to be a highly effective optimizer for deep learning models. We set the learning rate to 0.0003 and the batch size to 198. The learning rate was chosen based on empirical testing to achieve the best performance on the task, and the batch size was chosen to optimize training time while still ensuring that the model would generalize well to unseen data. During training, we used the cross-entropy loss function, which is commonly used for multi-class classification tasks.

4.3. **Recurrent neural network model with attention mechanism.** The RNN model with attention is built using Keras. The embedded sequences are then passed through a bidirectional LSTM layer, which can capture both forward and backward dependencies in the text data. The output of the LSTM layer is then used to compute attention weights, which determine the importance of each time step in the sequence. The attention weights are computed as the dot product between the LSTM output and the dense features, followed by a softmax activation. The attention weights are then used to compute a weighted sum of the LSTM output, which is concatenated with the dense features and passed through a final dense layer with softmax activation to output the predicted class probabilities. During training, the model is compiled with the Adam optimizer and trained on the training data for a fixed number of epochs, with a specified batch size. The performance of the model is evaluated using k-fold cross-validation, where the data is split into 10 folds and the model is trained and evaluated on each fold.

4.4. **The fastText model.** The fastText library was used to train a supervised learning model for Khmer news classification. We used the optimal hyperparameters to train our fastText model. This included a learning rate of 0.1, which determines the step size for updating the model's parameters during training. We also set the dimensionality of the word vectors to 98, which is the number of dimensions used to represent each word in the text. Finally, we used a bucket size of 1,900,000, which determines how many unique word types are used in the model's vocabulary. Figure 4 shows evaluation results of the RNN with attention and DNN model using TF-IDF and the fastText supervised model with optimal hyperparameters.

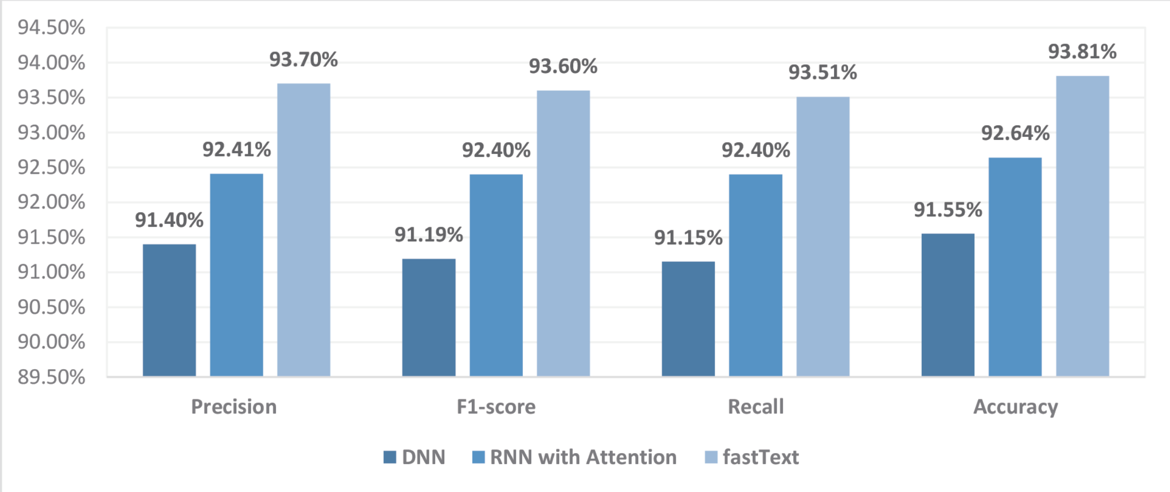

FIGURE 4. Performance evaluation of each model

As a result, we discovered that the fastText supervised model, with optimal hyperparameters, outperformed the DNN and RNN models using TF-IDF with optimal hyperparameters in all metrics. It achieved an accuracy of 93.81%, while the DNN and RNN models achieved 91.55% and 92.64%, respectively.

5. **Conclusions and Discussion.** After evaluating various models, we determined that the fastText supervised model with optimal hyperparameters outperformed the DNN and RNN models that used TF-IDF hyperparameters in all metrics, achieving an accuracy of 93.81%. Subsequently, we discovered that the combination of DNN and RNN models with TF-IDF feature representation method had certain limitations when compared to fastText. fastText is better equipped to handle out-of-vocabulary words by representing sub-word units in addition to whole words, which allows it to capture some of the meaning of unknown words. fastText uses a vector representation of words that allows it to capture more nuanced semantic information. fastText, on the other hand, reduces the dimensionality of the feature representation by representing words as vectors with fewer dimensions. Lastly, the TF-IDF feature representation may not adequately represent long documents because its representations are based on individual words and do not capture contextual information between words in a document. A text can be related to with multiple categories [19]. In the future, we aim to improve and classify news articles into multiple categories, and we intend to collect more Khmer news articles to improve the generalizability of the model and increase its performance on unseen Khmer news articles. Moreover, we plan to integrate the model into a larger system or application, such as a news recommendation engine or a text classification tool.

Acknowledgment. This work is supported by the Royal Scholarship under Her Royal Highness Princess Maha Chakri Sirindhorn Education Project to the Kingdom of Cambodia and the Computer Science Department, College of Computing, Khon Kaen University, Khon Kaen, Thailand.

REFERENCES

[1] P. Prasanna and D. Rao, Text classification using artificial neural networks, *International Journal of Engineering and Technology*, vol.7, pp.603-606, 2018.
[2] T. Yao, Z. Zhai and B. Gao, Text classification model based on fastText, *2020 IEEE International Conference on Artificial Intelligence and Information Systems (ICAIIS)*, pp.154-157, 2020.
[3] P. Bojanowski, E. Grave, A. Joulin and T. Mikolov, Enriching word vectors with subword information, *Transactions of the Association for Computational Linguistics*, vol.5, pp.135-146, 2017.

[4] R. Buoy, N. Taing and S. Chenda, Khmer text classification using word embedding and neural networks, *arXiv Preprint*, arXiv: 2112.06748, 2021.
[5] M. Zhang, Applications of deep learning in news text classification, *Scientific Programming*, vol.2021, pp.1-9, 2021.
[6] U. Pujianto, H. Rosyid and M. Anam, Classification of news articles for learning using the K-Nearest Neighbor algorithm, *2021 7th International Conference on Education and Technology (ICET)*, pp.256-260, 2021.
[7] F. S. Nurfikri, M. S. Mubarok and Adiwijaya, News topic classification using mutual information and Bayesian network, *2018 6th International Conference on Information and Communication Technology (ICoICT)*, pp.162-166, 2018.
[8] J. Bergstra and Y. Bengio, Random search for hyper-parameter optimization, *Journal of Machine Learning Research*, vol.13, 2012.
[9] S. Falkner, A. Klein and F. Hutter, BOHB: Robust and efficient hyperparameter optimization at scale, *International Conference on Machine Learning*, pp.1437-1446, 2018.
[10] L. Yang and A. Shami, On hyperparameter optimization of machine learning algorithms: Theory and practice, *Neurocomputing*, vol.415, pp.295-316, 2020.
[11] S. Symeonidis, D. Effrosynidis and A. Arampatzis, A comparative evaluation of pre-processing techniques and their interactions for Twitter sentiment analysis, *Expert Systems with Applications*, vol.110, pp.298-310, 2018.
[12] V. Chea, Y. Thu, C. Ding, M. Utiyama, A. Finch and E. Sumita, Khmer word segmentation using conditional random fields, *Khmer Natural Language Processing*, pp.62-69, 2015.
[13] T. Mikolov, Q. Le and I. Sutskever, Exploiting similarities among languages for machine translation, *arXiv Preprint*, arXiv: 1309.4168, 2013.
[14] J.-T. Chien, Chapter 7 – Deep neural network, *Source Separation and Machine Learning*, pp.259-320, 2019.
[15] D. P. Mandic and J. A. Chambers, *Recurrent Neural Networks for Prediction: Learning Algorithms, Architectures and Stability*, John Wiley & Sons, Ltd., 2001.
[16] J. Snoek, H. Larochelle and R. Adams, Practical Bayesian optimization of machine learning algorithms, *Advances in Neural Information Processing Systems*, vol.25, 2012.
[17] L. Buitinck, G. Louppe, M. Blondel et al., API design for machine learning software: Experiences from the scikit-learn project, *arXiv Preprint*, arXiv: 1309.0238, 2013.
[18] R. Martinez-Cantin, BayesOpt: A Bayesian optimization library for nonlinear optimization, experimental design and bandits, *J. Mach. Learn. Res.*, vol.15, pp.3735-3739, 2014.
[19] M. H. Nguyen, An extension of label-oriented approach for multi-labels text classification, *International Journal of Innovative Computing, Information and Control*, vol.18, no.4, pp.1265-1274, 2022.