Vietnamese News Articles Classification Using Neural Networks

To Nguyen Phuoc Vinh and Ha Hoang Kha
Ho Chi Minh City University of Technology (HCMUT), Ho Chi Minh City, Vietnam
Vietnam National University Ho Chi Minh City, Ho Chi Minh City, Vietnam
Email: {1870325, hhkha}@hcmut.edu.vn

***Abstract*—In this paper, a new benchmark Vietnamese online news article dataset for a multi–label task is introduced. The dataset is collected from well-known Vietnamese news websites and, then, it is assigned into 30 topics comparable to the way that editors label their articles. This leads our dataset to be very suitable for training practical applications in Vietnamese text classification. Furthermore, we modify the original pipeline of Vietnamese text classification by cutting the dimension of feature vectors based on the term frequency across the whole corpus which has been combined in the term frequency-inverse document frequency weighting step, instead of applying feature selection algorithms after extracting a huge dimension term frequency–inverse document frequency feature vector. Although this makes the computational complexity of method decreased, input feature vectors are weak due to removal of feature selection steps. Thus, utilizing the powerful neural network models for classification helps the efficiency be still as good as the original method, even it is slightly better.**

***Index Terms*—term frequency-inverse document frequency, neural network model, text classification, Vietnamese online news article dataset**

I. Introduction

Text Classification (TC) targets to categorize a text document into a set of pre-labeled documents. In other words, given a set of m text documents $D=(d_1,d_2,...,d_m)$ and a set of k categories $C=(c_1,c_2,...,c_k)$, a text classifier maps a document d_i correctly into an appropriate category c_j. Nowadays, TC is widely deployed to be one of the most important sub-parts of natural language processing systems [1]. Conventionally, methods such K-Nearest-Neighbor (KNN) [2], Naïve Bayes (NB) [3], Support Vector Machine (SVM) [4] have been adopted for TC. Recently the Machine Learning (ML) based approaches for TC have been attracted due to their superior performance. Because of development of powerful hardware and advanced ML algorithms, there are various practical applications using TC such as hate-speech detection [5], articles auto-tagging [6], chatbot [7].

Manuscript received January 5, 2021; revised August 25, 2021.

Three main parts of topic classification are comprised of binary class, multi-class, and multi-label problems. Based on their names, each problem can be explained briefly as follows. Multi-class classifiers aim to label an object (document) into its correctly class from a set of categories $C=(c_1,c_2,...,c_k)$ $k>2$, while a binary-class problem is a special case of multi-class one with the minimum number of categories ($k=2$) [8]. On the other hand, in multi-label tasks each object can be assigned into many categories.

Related works: Along with many of ML problems, TC has been extremely attractive. Thus, a lot of researches have been conducted to build feature extractions as well as efficient classifiers for TC tasks during recent decades [9]-[12]. In these methods, SVMs and Neural Networks (NNs) have been considered as two most choices in classification problems, particularly in TC. For feature extractions, there have also been many feature schemes which are investigated to represent text objects into robust features for TC such as mutual information [13], information gain [14], Chi-square [15], odds ratio [16], distinguishing feature selector [17].

News article classification is one of main applications of TC. Commonly, each news article has to be titled by editors. That may be an effortful work because they have to categorize hundreds, even thousands of news a day. Therefore, auto-tagging news articles helps editors' works reduce significantly. Since human's language on over the world is very diverse, various news article classification researches have been performed on different languages. For example, the authors in [18] have presented the combination of Term Frequency-Inverse Document Frequency (TF-IDF) feature extraction method and multinomial NB classifiers are the best choice for news article classification in Indonesian language among other methods. Alternatively, the reference [19] has surveyed the efficiency of KNN, NB, and SVM on Chinese news articles based on TF-IDF features. In [20], SVMs have been experimentally proved as the top performance in Arabic language.

Although auto-categorizing news articles in Vietnamese are very important today, there is the small number of researches which address this problem. In [21], the authors have proposed two models for Vietnamese TC. A Bag-of-Words (BoW) based approach which uses the BoW method to build a dictionary has been

doi: 10.12720/jait.12.4.363-369

introduced. This approach employs the TF-IDF to represent feature vectors followed by feature selection algorithms. Then, SVM is chosen as a classifier to investigate the efficiency of those feature selection methods. The second one is called an N-gram model and N-gram model based classifier. Instead of using BoW, this method utilizes the N-gram model to create dictionary. A statistical N-gram model based classifier is then deployed to categorize the objects. Van *et al.* [22] have implemented the key words extraction method to extract vector representation after generating dictionary based on the BoW method. For the classification phase, a 6-layer NN with tanh activation functions is utilized. In [23], different classifiers have been surveyed with TF-IDF weighting vectors as the inputs. In addition, the authors have proposed their benchmark Vietnamese news dataset named VNNews-01.

Contributions: In this paper, three types of tokenization methods including the BoW model, N-gram model, BoW+N-gram model are investigated for generating dictionaries. The vector representation based on the TF-IDF weighting of those dictionaries is generated at the next step. Finally, we analyze the impact of the number of hidden layers and activation functions in NNs on BoW model tokenization with TF-IDF feature extraction. Furthermore, our new benchmark Vietnamese online news article dataset constructed by crawling tens thousands of documents on well-known Vietnamese news websites is presented. Unlike the datasets have been introduced in earlier research [23] which are not deeply labeled into the final sub-categories on online news websites, each label of the document in our dataset is based more on editors' practical way.

The rest of this paper is structured as follows. The detailed structure of our new benchmark Vietnamese online news dataset and the pipeline of our work are presented in Section II. Section III then shows the investigation of our proposed model on some hyper-parameters as well as the comparation of our model with earlier models for the Vietnamese news classification task, followed by our conclusions and future works in Section IV.

II. Our Dataset and Classification

In this section, the details of building our benchmark dataset for Vietnamese online news classification task will be clearly presented. Furthermore, the pipeline of our method to classify news topics on that dataset is also mentioned.

A. Our Dataset

We develop a Vietnamese online news benchmark dataset collected by using the Python Scrapy framework to crawl the documents from 5 Vietnamese well-known online news websites such as: dantri.com.vn, thanhnien.vn, vnexpress.net, doisongsuckhoe.vn, qdnd.vn. Constructing a news dataset based on those news websites takes an advantage of categorizing the topic of each document precisely since they have been labeled by the experienced editors. This is very convenient and useful for classification tasks based on ML algorithms. The detail of our dataset is shown in Fig. 1.

For practical purposes, all labels have been retained as editors assigned the topics for their news articles. Moreover, note that VNNews-01 dataset mentioned in [23] may get a problem of duplication. For example, batdongsan (real estate) and kinhdoanh (business) are two of 25 categories in VNNews-01; however, on vnexpress.vn news website, batdongsan (real estate) is a sub-category of kinhdoanh (business).

Thus, we assign each news articles following their final sub-categories in order to not only prevent from that issue, but also make the labels similar to the way that editors assign those articles.

B. The Pipeline of Our Pethod

Preprocessing: After being fetched from news websites, the documents are put into preprocessing steps including: removing all numbers and special words as well as punctuations, transforming all words into lower-case, and stop-words removal. With removing stop-words process, a list of stop-words which contains mostly 2000 unnecessary words for Vietnamese topic classification tasks (e.g., "chẳng hạn như", "do đó", "vì vậy") is constructed. It is really imperative that eliminating these kinds of words make the computational complexity much reduce.

Tokenization: The preprocessed documents are tokenized by both the BoW model and N-gram model to create different dictionaries. With the BoW model, each word is considered as a term of dictionary, while the N-gram model uses the sequences of N adjacent words to set elements in the dictionary. In our work, the various dictionaries are built to find out a reasonable model which is the best one for Vietnamese TC. Namely, we introduce 7 dictionaries – one based on the BoW model, three dictionaries generated by three kinds of the N-gram model (bi-gram, 3-gram, and 4-gram), and the last three things are the combinations of BoW and these three N-gram dictionaries.

Term Frequency-inverse Document Frequency: TF-IDF which measures the influence of on a document in the corpus returns statistical weights [24]. The importance of that word increases comparably to how many times that word appears in the document and, then, the result is also offset by the frequency of the word in the whole corpus. Nowadays, there are an overwhelming majority of applications utilizing TF-IDF as a weighting factor in searches of information retrieval, text mining, and user modeling. According to the survey [25], 83% of text-based recommender systems in digital libraries use TF-IDF. Beyond to that, the comprehensive experiments [26], feature extraction using the TF-IDF statistical model illustrates the good performance for TC.

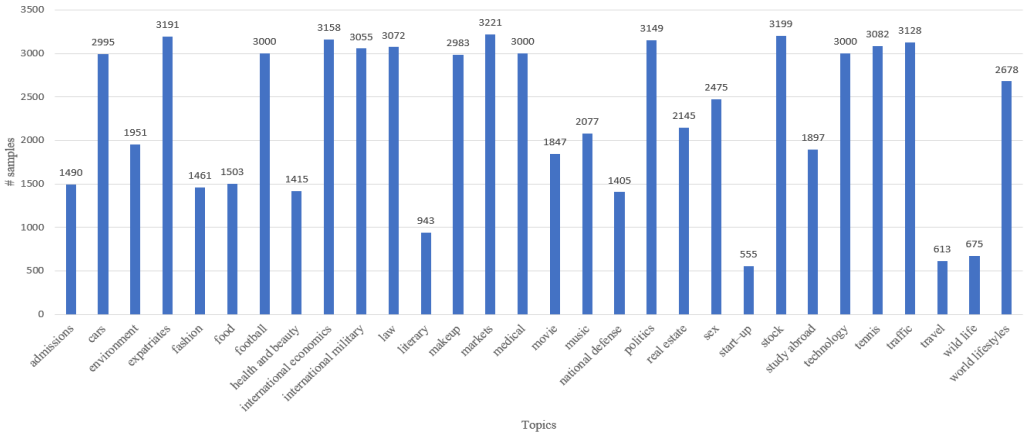


Figure 1. The detail of our Vietnamese online news article dataset.

The first term - TF reflecting the normalization of the dominant characteristic of a word w on the whole document d is defined as:

$$tf_{w,d} = \frac{\text{Number of } w \text{ in } d}{\text{Total number of words in } d}. \quad (1)$$

In fact, the more frequently a specific word presents in the document, the more discriminating power that word contains.

The second term - IDF is proposed to scaling the $tf_{w,d}$ weight of a word with the regularity of that word in the whole corpus. The value of IDF term of a word w is defined as follows:

$$idf_w = \log \frac{N}{df_w + 1} \quad (2)$$

where N is the total number of documents in the corpus; df_w determined by counting all documents which contain the word w is the document frequency. Hence, the more widely a word is used in the corpus, the less the idf_w value is.

Finally, the TF-IDF weighting which is the combination of the TF term and IDF term to produce a composite weight for each word in documents is computed as:

$$tf\text{-}ifd_{w,d} = tf_{w,d} \times idf_w \quad (3)$$

In other words, if a word w rarely appears in documents, but it is often used in a specific document d, then that word has much more discriminating power than others in d. That is, the $tf\text{-}idf_{w,d}$ value of w in d is greater than others' one.

The TF-IDF transformation outcomes a huge dimensional vector for each document. To be more efficiently used, these feature vectors are usually then applied to well-known dimensionality reduction techniques like Principal Component Analysis (PCA) [27], Orthogonal Centroid Feature Selection (OCFS) [28], etc. In this paper, each text document is TF-IDF-transformed into a 10000-dimensional vector. Instead of using feature selection algorithms, the features are ordered by their TF values across the whole dataset, K top maximum TF value features are chosen to reduce the dimension of these feature vectors. Because the TF value of features have been already computed from TF-IDF transformation step, this way to reduce the dimension of feature vectors is much simplifier than most of current dimensionality reduction techniques.

C. Neural Network Model

The architecture of our classifier, as shown in Fig. 2, comprises of the input layer, the stack of L hidden layers whose combinations are an affine transformation followed by the batch normalization and activation function, the dropout layer and the output layer.

Input layer: The processed text documents are tokenized into a list of N-sequence words called the N-gram. The TF-IDF model computes the weight of elements in this list and returns vectors in which the *i-th* row contains a TF-IDF value of the *i-th* N-sequence word in the list. At that time, each news article is represented by a TF-IDF weighting vector, therefore, the input layer of NNs has the number of input nodes equal to the size of TF-IDF vectors. We also denote a minibatch of m samples where each sample is a TF-IDF weighting vector.

The stack of hidden layers: The first part of a hidden layer is an affine function which linearly transforms the output of the previous layer as follows:

$$\mathbf{Z}^{[l]} = \mathbf{\Theta}^{[l]}\mathbf{G}^{[l-1]} + \mathbf{b}^{[l]}\ ;\ l = 1, 2, \ldots, L \quad (4)$$

where $\mathbf{\Theta}^{[l]} \in \mathbb{R}^{n_l \times n_{l-1}}$ and $\mathbf{b}^{[l]} \in \mathbb{R}^{n_l \times 1}$ are affine weights and biases of the *l–th* hidden layer, respectively.

Batch normalization is used to accelerate the convergence of NNs and make an impact a little bit on overfitting problems in training time. For each minibatch, given $\mathbf{z}^{(i)} \in \mathbb{R}^{n_l \times 1}$ in $\mathbf{Z}^{[l]}$, its normalized value is computed as follows:

$$\tilde{\mathbf{z}}^{(i)} = \boldsymbol{\gamma} \odot \frac{\mathbf{z}^{(i)} - \hat{\boldsymbol{\mu}}_{\mathbf{Z}^{[l]}}}{\hat{\boldsymbol{\sigma}}_{\mathbf{Z}^{[l]}}} + \boldsymbol{\beta}\ ;\ i = 1, 2, \ldots, m \quad (5)$$

where $\boldsymbol{\gamma}$ and $\boldsymbol{\beta}$ are learnable parameters that would be learned during training time; while $\hat{\boldsymbol{\mu}}_{\mathbf{Z}^{[l]}}$ and $\hat{\boldsymbol{\sigma}}_{\mathbf{Z}^{[l]}}$ are the estimated mean and variance based on the statistics of the current minibatch $\mathbf{Z}^{[l]}$ as the following expressions:

$$\hat{\boldsymbol{\mu}}_{\mathbf{Z}^{[l]}} = \frac{1}{m}\sum_{i=1}^{m} \mathbf{z}^{(i)} \quad (6)$$

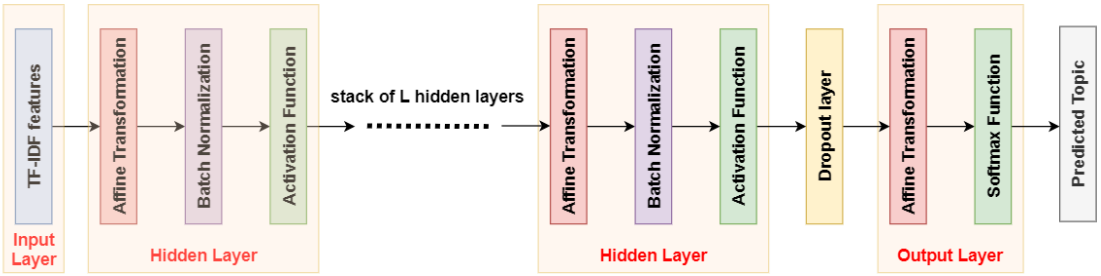


Figure 2. The proposed neural networks architecture.

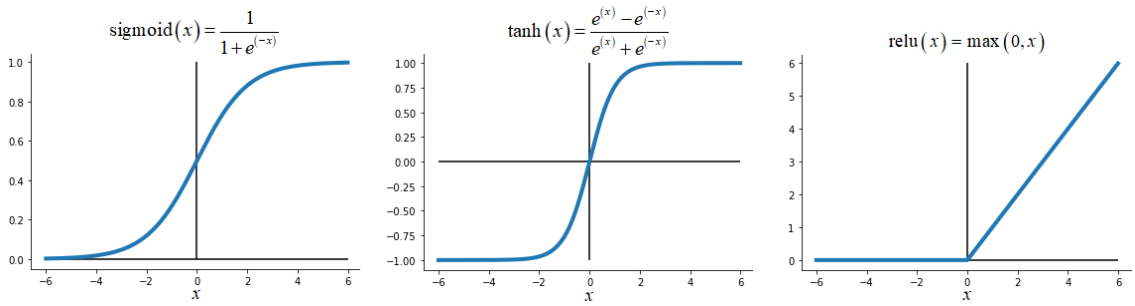


Figure 3. Some commonly-used activation functions.

$$\hat{\boldsymbol{\sigma}}_{\mathbf{Z}^{[l]}} = \sqrt{\frac{1}{m}\sum_{i=1}^{m}\left(\mathbf{z}^{(i)} - \hat{\boldsymbol{\mu}}_{\mathbf{Z}^{[l]}}\right)^2 + \varepsilon} \tag{7}$$

Note that ε is a small value used to avoid attempt at division by zero. The standardized matrix $\tilde{\mathbf{Z}}^{[l]}$ is the output after batch normalizing $\mathbf{Z}^{[l]}$ and it is described as:

$$\tilde{\mathbf{Z}}^{[l]} = \begin{bmatrix} \vdots & \vdots & \vdots & \vdots \\ \tilde{\mathbf{z}}^{(1)} & \tilde{\mathbf{z}}^{(2)} & \cdots & \tilde{\mathbf{z}}^{(m)} \\ \vdots & \vdots & \vdots & \vdots \end{bmatrix} \tag{8}$$

The last function called activation function is used to non–linearly transform each element in $\tilde{\mathbf{Z}}^{[l]}$. Some commonly–used activation functions are illustrated in Fig. 3. In this paper, the two most recommended activation functions $\mathbf{G}^{[l]} = \text{relu}\left(\tilde{\mathbf{Z}}^{[l]}\right)$, and $\mathbf{G}^{[l]} = \tanh\left(\tilde{\mathbf{Z}}^{[l]}\right)$ are studied.

Dropout layer: Along with regularization, dropout [29] is an effective technique to prevent overfitting. The principal of the dropout layer is to randomly remove a number of nodes with probability P in its previous layer on each training iteration. This leads learning progress not to concentrate merely on some specific input features of that layer, but all of these inputs are weighted fairly. However, unlike batch normalization, applying too much dropout makes the performance of our networks shrink. Thus, in our work, the dropout layer is only deployed for the *L–th* hidden layer followed by the output layer. This decision is to make our networks overcome overfitting problems, yet it still keeps a good performance.

Output layer: The output layer comprises of affine transformations and softmax activation functions as follows:

$$\mathbf{O} = \text{softmax}\left(\boldsymbol{\Theta}_o\mathbf{G}^{[L]} + \mathbf{b}_o\right) \tag{9}$$

The dimension of each sample in the minibatch output matrix should be the number of categories. As mentioned, our dataset has been labeled into 30 categories and, thus, $\mathbf{O} \in \mathbb{R}^{30\times m}$ which results in $\boldsymbol{\Theta}_o \in \mathbb{R}^{30\times n_L}$, and $\mathbf{b}_o \in \mathbb{R}^{30\times 1}$.

Training process: The categorical cross–entropy loss has been utilized to training the networks. Given the pair of predicted-ground true labels, we denote $\mathbf{o}^{(i)} \in \mathbb{R}^{30\times 1}$ and one-hot vector $\mathbf{y}^{(i)} \in \mathbb{R}^{30\times 1}$ corresponding to predicted label of the *i–th* sample in $\mathbf{O}$ and its ground true label, respectively. The categorical cross–entropy loss expanded by regularization for each mini batch is expressed as:

$$J = -\frac{1}{m}\left(\sum_{i=1}^{m}\sum_{c=1}^{30}\left(\mathbf{y}_c^{(i)}\log\left(\mathbf{o}_c^{(i)}\right)\right) + \frac{\lambda}{2}\left(\sum_{l=1}^{L}\left\|\boldsymbol{\Theta}^{[l]}\right\|_F^2 + \left\|\boldsymbol{\Theta}_o\right\|_F^2\right)\right) \tag{10}$$

where the second term in is the Frobenius regularization terms with

$$\left\|\boldsymbol{\Theta}^{[l]}\right\|_F^2 = \sum_{i=1}^{n_l}\sum_{j=1}^{n_{l-1}}\left(\boldsymbol{\Theta}_{ij}^{[l]}\right)^2 \tag{11}$$

and

$$\left\|\boldsymbol{\Theta}_o\right\|_F^2 = \sum_{i=1}^{30}\sum_{j=1}^{n_L}\left(\boldsymbol{\Theta}_{oij}\right)^2 \tag{12}$$

here, λ is the hyper–parameter which leads the tradeoff of model performance between training time and test time.

The number of learnable parameters is shown in Table I.

TABLE I. NUMBER OF LEARNING PARAMETERS

Functions	No. of weights	No. bias
Affine function	$\sum_{l=1}^{L}\left(n_l \times n_{l-1}\right) + 30\times n_L$	$\sum_{l=1}^{L} n_l + 30$
Batch normalization	$\sum_{l=1}^{L} n_l$	$\sum_{l=1}^{L} n_l$

The dataset is given in Fig. 1 in which the total number of samples is 68363. Our dataset has been divided into training, validation, and test set that contain 75%, 12.5%, 12.5% samples of dataset, respectively. The NN models have been trained through 100 epochs with the mini batch size of 128. Adam optimizer is also used to train the networks with the initial learning rate is 3×10^{-4}, the first and second exponential decay rate factors are 0.9 and 0.999, respectively.

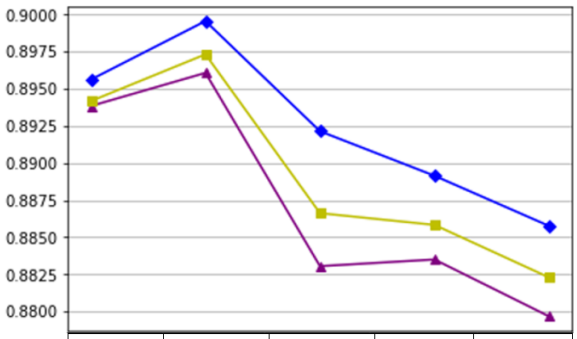
III. EXPERIMENTS

For evaluation, we choose the F_1 score to measure the efficiency of different models on our benchmark Vietnamese news dataset. The F_1 score depends on *recall* - the ratio of the number of documents assigned correctly to a category by our classifier to the total number of documents labeled to that category in our corpus, and *precision* - the percentage of the correctly-assigned documents by the machine over both correctly-assigned documents and falsely–assigned documents in a category. The expression of the F_1 score is described below:

$$F_1 = \frac{2\times recall \times precision}{recall + precision} \quad (13)$$

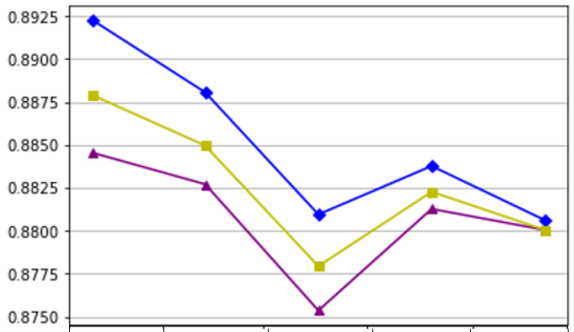
For the BoW input feature, K = 2500 tokens are selected after computing the TF-IDF feature vector. Furthermore, we also choose the probability of the dropout layer as P = 0.3, and λ = 0.015 for the Frobenius regularization term.

First of all, the number of hidden layers in networks is surveyed with the ReLU activation functions. The results in Fig. 4 depict that the network with three hidden layers with 256 nodes in each layer gives the best prediction on test dataset. Meanwhile, Fig. 5 illustrates that the stack of two hidden layers with tanh activation functions performs prediction better than others. Moreover, ReLU functions are also considered as the reliable choice for multi-label task on our dataset.



	L=2	L=3	L=4	L=5	L=6
precise	0.89561	0.89955	0.89213	0.88913	0.88575
recall	0.89383	0.89606	0.88304	0.88349	0.87966
F1	0.89416	0.89731	0.88661	0.88582	0.88228

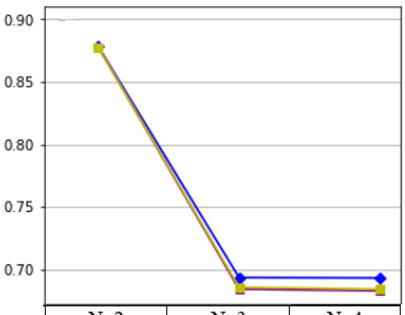
Figure 4. Evaluation of the ReLU activation functions on different number of hidden layers.



	L=2	L=3	L=4	L=5	L=6
precise	0.89228	0.88806	0.88095	0.88377	0.88062
recall	0.88455	0.88271	0.87536	0.88127	0.88005
F1	0.88791	0.88496	0.87794	0.88227	0.88005

Figure 5. Evaluation of tanh activation functions on different number of layers.

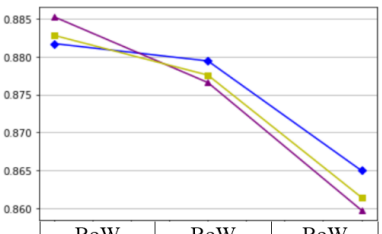
The results of our investigation with different types of the N-gram with K = 2500 tokens for each dictionary are depicted in Fig. 6. In this study, we use the best model with the stack of 3 hidden layers applying the ReLU activation functions to evaluate the impact of N-gram order on the multi-classification task.



	N=2	N=3	N=4
precise	0.87837	0.69351	0.69315
recall	0.87808	0.68420	0.68281
F1	0.87775	0.68576	0.68417

Figure 6. Evaluation on different N-gram orders.

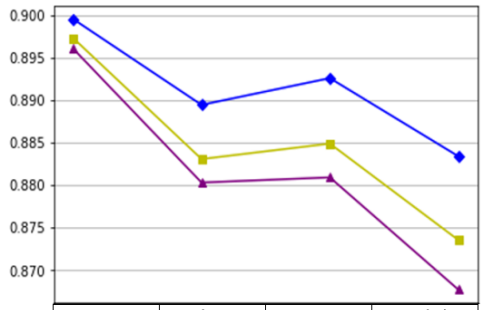
Fig. 7 shows the results of combining the BoW and N-gram model for generating dictionary. In this experiment, each vector representation has K = 5000 which comprises of 2500 TF-IDF weights of the BoW dictionary and others come from the N-gram model dictionary.



	BoW + bi-gram	BoW + 3-gram	BoW + 4-gram
precise	0.88171	0.87943	0.86498
recall	0.88523	0.87658	0.85976
F1	0.88279	0.87756	0.86145

Figure 7. Evaluation on the combination of the BoW and N-gram models.

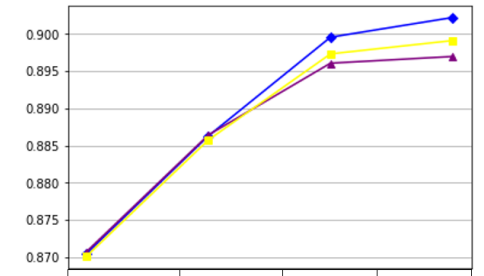
We also implement methods in [21] using OCFS to reduce the massive dimension of feature, so as to figure that the NN model slightly outperforms other classifiers although feature selection phase has been cut off the classification task. The results are shown in Fig. 8.



	Neural Network	Linear SVM (OCFS)	RBF-SVM (OCFS)	Logistic Regression (OCFS)
precise	0.89955	0.88945	0.89260	0.88341
recall	0.89606	0.88030	0.88091	0.86771
F1	0.89731	0.88307	0.88671	0.87355

Figure 8. Comparation of the neural network model with other classifiers.

Fig. 9 illustrates the performance on different sizes of TF-IDF feature vectors with K = 500, K = 1000, K = 2500, and the full-length K = 10000 dimensional vectors, respectively. With K = 2500 features, the neural network model still keeps the performance close to one with full-length feature vectors. However, when the size of feature vectors is dramatically reduced, namely with K = 500 and K = 1000, the performance of neural network models also gets a considerable decline on text classification task.



	K=500	K=1000	K=2500	K=10000
precise	0.87043	0.88628	0.89955	0.90220
recall	0.87069	0.88641	0.89606	0.89696
F1	0.87007	0.88574	0.89731	0.89910

Figure 9. Comparation of the performance of neural network models on different sizes of feature vectors.

IV. Conclusion

In this paper, we have introduced a new Vietnamese online news dataset which is separated strictly its topics. Also, we study the performance of the combination of TF-IDF features and classifier using neural network models without using feature selection algorithms. The result figures of neural network models should get the prospective in Vietnamese news articles classification tasks.

Our dataset could be extended with more classes in the future's work. Moreover, we would investigate some methods to enhance the features before putting them into our classifier.

Conflict of Interest

The authors declare that there is no conflict of interest regarding the publication of this article.

Author Contributions

T. N. P. Vinh conducted the research and the numerical simulations; T. N. P. Vinh and H. H. Kha analyzed the data; T. N. P. Vinh wrote the paper and H. H. Kha edited the paper; all authors had approved the final version.

Acknowledgement

This research is funded by Ho Chi Minh City University of Technology (HCMUT), VNU-HCM, under grant number BK-SDH-2021-1870325".

References

[1] Q. Li, H. Peng, J. Li, C. Xia, R. Yang, L. Sun, P. S. Yu, and L. He, "A survey on text classification: From shallow to deep learning," arXiv preprint arXiv:2008.00364, 2020.

[2] T. M. Cover and P. E. Hart, "Nearest neighbor pattern classification," *IEEE Trans. Inform. Theory*, vol. 13, no. 1, pp. 21-27, Jan. 1967.

[3] M. E. Maron, "Automatic indexing: An experimental inquiry," *J. ACM*, vol. 8, no. 3, pp. 404-417, 1961.

[4] T. Joachims, "Text categorization with support vector machines: Learning with many relevant features," in *Proc. European Conf. on Machine Learning*, Springer, Berlin, Heidelberg, 1998, pp. 137-142.

[5] N. Djuric, J. Zhou, R. Morris, M. Grbovic, V. Radosavljevic, and N. Bhamidipati, "Hate speech detection with comment embeddings," in *Proc. the 24th Int.Conf. on World Wide Web*, Florence, Italy, 2015, pp, 29-30.

[6] P. Mukalov, O. Zelinskyi, R. Levkovych, P. Tarnavskyi, A. Pylyp, and N. Shakhovska, "Development of system for auto-tagging articles, based on neural network," in *Proc. the 3rd Int. Conf. on Computational Linguistics and Intelligent Systems*, Kharkiv, Ukraine, 2019, pp. 106-115.

[7] K. Kuksenok and A. Martyniv, "Evaluation and improvement of chatbot text classification data quality using plausible negative examples," arXiv preprint arXiv:1906.01910, 2019.

[8] G. Tsoumakas and I. Katakis, "Multi-label classification: An overview," *Int. Journal Data Warehous*, vol. 3, no. 3, pp. 1-13, 2009.

[9] X. Zhang, J. Zhao, and Y. LeCun, "Character-Level convolutional networks for text classification," *Advances in Neural Information Processing Systems*, pp. 649-657, 2015.

[10] J. Howard and S. Ruder, "Universal language model fine-tuning for text classification," arXiv preprint arXiv:1801.06146, 2018.

[11] S. Lai, L. Xu, K. Liu, and J. Zhao, "Recurrent convolutional neural networks for text classification," in *Proc. Twenty-ninth AAAI Conf. on Artificial Intelligence*, Austin Texas, USA, 2015.

[12] G. Forman, "An extensive empirical study of feature selection metrics for text classification," *Journal of Machine Learning Research*, no. 3, pp. 1289-1305, Mar. 2003.

[13] H. Taira and M. Haruno, "Feature selection in svm text categorization," in *Proc. 11th Conf. on IAAI*, 1999, pp. 480-486.

[14] H. Uğuz, "A two-stage feature selection method for text categorization by using information gain, principal component analysis and genetic algorithm," *Knowledge-Based Systems*, vol. 24, no. 7, pp. 1024-1032, 2011.

[15] M. Rogati and Y. Yang, "High-Performing feature selection for text classification," in *Proc. the Eleventh Int. Conf. on Information and Knowledge Management*, New York, NY, USA, 2002, pp. 659-661.
[16] D. Mladeni´c, "Feature subset selection in text-learning," in *Proc. European Conf. on Machine Learning*, Berlin, 1998, pp. 95-100.
[17] A. K. Uysal and S. Gunal, "A novel probabilistic feature selection method for text classification," *Knowledge-Based Systems*, vol. 36, pp. 226-235, 2012.
[18] R. Wongso, F. A. Luwinda, B. C. Trisnajaya, and O. Rusli, "News article text classification in Indonesian language," *Proc. Computer Science*, vol. 116, pp. 137-143, 2017.
[19] F. Miao, P. Zhang, L. Jin, and H. Wu, "Chinese news text classification based on machine learning algorithm," in *Proc. 10th Int. Conf. on Intelligent Human-Machine Systems and Cybernetics*, 2018, vol. 2, pp. 48-51.
[20] L. A. Qadi, H. E. Rifai, S. Obaid, and A. Elnagar, "Arabic text classification of news articles using classical supervised classifiers," in *Proc. 2nd Int. Conf. on New Trends in Computing Sciences*, 2019, pp. 1-6.
[21] V. C. D. Hoang, D. Dinh, N. L. Nguyen, and H. Q. Ngo, "A comparative study on Vietnamese text classification methods," in *Proc. IEEE Int. Conf. on Research, Innovation and Vision for the Future*, 2007, pp. 267-273.
[22] T. P. Van and T. M. Thanh, "Vietnamese news classification based on bow with keywords extraction and neural network," in *Proc. 21st Asia Pacific Symposium on Intelligent and Evolutionary Systems*, 2017, pp. 43-48.
[23] H. Duong and V. T. Hoang, "A survey on the multiple classifier for new benchmark dataset of Vietnamese news classification," in *Proc. 11th Int. Conf. on Knowledge and Smart Technology*, 2019, pp. 23-28.
[24] Z. Yun-tao, G. Ling, and W. Yong-cheng, "An improved TF-IDF approach for text classification," *Journal of Zhejiang University-Science A6*, no. 1, pp. 49-55, 2005.
[25] J. Beel, B. Gipp, S. Langer, and C. Breitinger, "Research-Paper recommender systems: A literature survey," *Int. Journal on Digital Libraries*, vol. 17, no. 4, pp. 305-338, 2016.
[26] X. Deng, Y. Li, J. Weng, and J. Zhang, "Feature selection for text classification: A review," *Multimedia Tools and Applications*, vol. 78, no. 3, pp. 3797-3816, 2019.
[27] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *The Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1929-1958, 2014.
[28] A. Maćkiewicz and W. Ratajczak, "Principal Components Analysis (PCA)," *Computers & Geosciences*, vol. 19, no. 3, pp. 303-342, 1993.
[29] J. Yan, N. Liu, B. Zhang, S. Yan, Z. Chen, Q. Cheng, W. Fan, and W. Ma, "OCFS: Optimal orthogonal centroid feature selection for text categorization," in *Proc. the 28th Annual International ACM SIGIR Conf. on Research and Development in Information Retrieval*, 2005, pp. 122-129.

To Nguyen Phuoc Vinh received the B.Eng. degree in Electronics and Telecommunications Engineering from Ho Chi Minh City University of Technology, in 2018. His research interests are the areas of image and signal processing, applications of machine learning and deep learning.

Ha Hoang Kha received the B.Eng. and M.Eng. degrees from Ho Chi Minh City University of Technology, in 2000 and 2003, respectively, and the Ph.D. degree from the University of New South Wales, Sydney, Australia, in 2009, all in Electrical Engineering and Telecommunications. From 2000 to 2004, he was a research and teaching assistant with the Department of Electrical and Electronics Engineering, Ho Chi Minh City University of Technology. He was a visiting research fellow at the School of Electrical Engineering and Telecommunications, the University of New South Wales, Australia, from 2009 to 2011. He was a postdoctoral research fellow at the Faculty of Engineering and Information Technology, University of Technology Sydney, Australia from 2011 to 2013. He is currently a lecturer at the Faculty of Electrical and Electronics Engineering, Ho Chi Minh City University of Technology, Vietnam. His research interests are in digital signal processing and wireless communications, with a recent emphasis on convex optimization and machine learning techniques in signal processing for wireless communications.