บทความวิจัย

การจำแนกข้อความโดยใช้การเรียนรู้ของเครื่องสำหรับหนังสือราชการไทย

ปกรณ์ สันตกิจ พงษ์พร พันธ์เพ็ง ปรีชา โพธิ์แพง และ เยาวลักษณ์ งามแสนโรจน์*
สาขาวิชาเทคโนโลยีสารสนเทศ คณะวิทยาศาสตร์ มหาวิทยาลัยราชภัฏลำปาง

* ผู้นิพนธ์ประสานงาน โทรศัพท์ 08 1289 6133 อีเมล: ayaowalak@hotmail.com

DOI: 10.14416/j.kmutnb.2024.05.03
รับเมื่อ 2 กันยายน 2565 แก้ไขเมื่อ 3 กุมภาพันธ์ 2566 ตอบรับเมื่อ 23 กุมภาพันธ์ 2566 เผยแพร่ออนไลน์ 8 พฤษภาคม 2567

บทคัดย่อ

บทความนี้มีวัตถุประสงค์เพื่อกำหนดรูปแบบการจำแนกประเภทข้อความที่เหมาะสมที่สุดสำหรับการจัดประเภทข้อความหลายชั้นในโดเมนเอกสารทางราชการภาษาไทย ในการทดลองได้ทำการศึกษา โดยการสร้างตัวแยกประเภทข้อความโดยใช้ WangchanBERTa ซึ่งเป็นโมเดลภาษาไทยแบบฝึกล่วงหน้าร่วมกับตัวแบบดั้งเดิมที่เป็นที่นิยมและเปรียบเทียบประสิทธิภาพ โมเดลจำแนกประเภททั้งหมดได้รับการปรับแต่งให้เหมาะสม และทำการฝึกฝนชุดข้อมูลองค์กร ซึ่งได้ประเมินจากเมตริกการประเมิน 4 แบบ ได้แก่ค่า Accuracy, Precision, Recall และ F1-score. ผลการทดลองแสดงให้เห็นว่าแบบจำลอง WangchanBERTa มีความแม่นยำสูงถึง 76% ซึ่งประสิทธิภาพดีกว่าแบบจำลองพื้นฐานอื่น ๆ และสามารถนำมาประยุกต์ใช้สำหรับหน่วยงานราชการไทย ในการจำแนกประเภทของหนังสือราชการไทยได้

คำสำคัญ: BERT WangchanBERTa การจำแนกข้อความ การเรียนรู้เชิงลึก เอกสารราชการไทย

การอ้างอิงบทความ: ปกรณ์ สันตกิจ, พงษ์พร พันธ์เพ็ง, ปรีชา โพธิ์แพง และ เยาวลักษณ์ งามแสนโรจน์, “การจำแนกข้อความโดยใช้การเรียนรู้ของเครื่องสำหรับหนังสือราชการไทย,” *วารสารวิชาการพระจอมเกล้าพระนครเหนือ*, ปีที่ 34, ฉบับที่ 4, หน้า 1–12, เลขที่บทความ 244-176235, ต.ค.–ธ.ค. 2567.

Research Article

Text Classification Using Machine Learning for Thai Official Letters

Pakorn Santakij, Pongporn Punpeng, Preecha Phopaeng and Yaowalak Ngamsanroaj*
Department of Information Technology, Faculty of Science, Lampang Rajabhat University, Lampang Thailand

* Corresponding Author, Tel. 08 1289 6133, E–mail: ayaowalak@hotmail.com

DOI: 10.14416/j.kmutnb.2024.05.03
Received 2 September 2022; Revised 3 February 2023; Accepted 23 February 2023; Published online: 8 May 2024

Abstract

This article aims to determine the most suitable text classification model for creating a multi-class Text classification in the Thai official letter domain. An experimental study was conducted by creating text classifiers using WangchanBERTa, a Pre-trained Thai Language Model, along with other popular traditional ones and comparing their performance. All classifiers were fine-tuning and trained on the organization dataset. They were evaluated by four evaluation metrics: accuracy, precision, recall, and F1-scores. The experiment results showed that the WangchanBERTa model outperforms the baseline models with the highest accuracy of 76%. It can also be applied for Thai government organizations to classify types of Thai official letters.

Keywords: BERT, WangchanBERTa, Text Classification, Deep Learning, Thai Official Letter

Please cite this article as: P. Santakij, P. Punpeng, P. Phopaeng, and Y. Ngamsanroaj, "Text classification using machine learning for Thai official letters," *The Journal of KMUTNB*, vol. 34, no. 4, pp. 1–12, ID. 244-176235, Oct.–Dec. 2024 (in Thai).

1. Introduction

Text classification is a crucial aspect of natural language processing. It is a procedure designed to arrange or classify documents for easier administration, retrieval, or data analysis. Initially, the Rule-Based method was founded on human-defined language rules. The benefit is that the dataset is not necessary, but it is restricted to learning the structure of the human-defined language [1], which is problematic when processing non-syntax text and bigger volumes of data.

In the past decade, there has been a shift toward categorization applications that use machine learning methods. Machine learning is the process of creating applications that acquire and enhance their knowledge based on the data they encounter. This knowledge can then be reused as an experience, which increases the efficiency of working with new data that have never been seen [2]. Machine learning was statistic-based and could learn language patterns with greater flexibility and precision [3]. Classic Machine learning technics work well in document classification. A linear model (Logistic Regression; LR), Kernel (Support Vector Machines; SVM), distance functions (K Nearest Neighbors; KNN), or employing probability principles (Naive Bayes; NB) are all widespread and form strong results concerning the model as mentioned above. The preceding methods have been utilized in the official documentation. For instance, NB, KNN, and SVM have been employed to categorize the World Intellectual Property Organization (WIPO) dataset [4], [5].

In document classification, deep learning, a specialized subset of machine learning, was recently developed. This is because, when taught with big data, it is more accurate than traditional methods [6]. In addition, it decreases the human work required for feature engineering by automating the process and making the models transferable across different tasks. Pre-trained models are the new standard for high-performance NLP tasks and they are deep learning models that turn existing information into new models by training on large datasets with varying contexts. Bidirectional Encoder Representations from Transformers (BERT) is an NLP model designed to pre-train deep bidirectional representations from unlabeled text and it could be fine-tuned using labeled text for different NLP tasks [7]

The BERT-based model is used, such as, in patent classification using a large dataset [8]. To use the BERT-based model to train for building a model that is especially learning Thai text and has been presented WangchanBERTa by Thai researchers. WangchanBERTa is a Pre-trained Thai Language Model that uses the transformer architecture of BERT. This model was trained with a total data size of 78 GB of data collected from various domain domains such as social media posts, news articles, and other publicly available datasets and applied Thai-specific message processing rules. The results from performance testing found that WangchanBERTa outperforms baseline models [9]. In addition, it was founded that research WangchanBERTa was used to analyze Thai sentiments from Twitter in Sentiment Classification. In addition, it was established that research led WangchanBERTa to examine Thai sensation from Twitter in sentiment classification. The performance was compared to models generated from SVM and LR. The experiments demonstrate that the WangchanBERTa model outperforms both SVM and LR [10].

This study aims to develop machine-learning models for classifying Thai official letters for use in LPRU as an alternative to human labor. The following Research Questions are addressed. First, what is the performance of traditional models for the classification of multi-class text? As a comparison baseline, we established Multinomial NB, Linear SVM, and LR as baseline models and conducted experiments on the dataset derived from our organization's documents. Second, does WangchanBERTa outperform conventional models in the classification of multi-class text? To facilitate comparison, we conduct the experiments on the same dataset as the baseline models. Finally, does the imbalanced dataset impact the effectiveness of WangchanBERTa? Thus, we experimented with use the same dataset to train two distinct models.

The remainder of the paper is organized as follows. Section 2 outlines our study process by introducing our classification criteria, dataset, models, and Evaluation Metrics. The outcomes of our investigations are presented in Section 3, which will answer research questions. Conclusion and discussion of Section 4.

2. Materials and Methods

This research defines an experiment by preparing a classed and labeled dataset. Then, we create a classifier to classify the test dataset using WangchanBERTa, a pre-trained Thai Language Model and evaluate the classifier's efficiency.

2.1 Document Classification Criteria

In this section, we described the types of documents that were classified, and the criteria utilized to classify them. First, the document dataset used for this study is classified as Thai Official Letter, which operates inside the LPRU's official communications. According to the Office of the Prime Minister's Regulation on Correspondence, B.E. 1983 [11], the official letter can be classified into eight distinct types. The Performance Appraisal workload will be assigned to teachers and employees according to the Performance Handbook.

The classification criteria are based on the nature of the work specified in each order, which can be subdivided into reference types according to the workload criteria announcement of LPRU. The eight reference types are an advisor to individual projects/student research, professional experience supervisors, arts and culture work, academic services, advisors for student groups/clubs, specific orders, central work orders, and administrative work.

2.2 Dataset

We perform our experiments on the dataset as follows.

2.2.1 Data Source

The datasets are official letters-related organizational documents. There were 3,489 pieces obtained from the management information system of the Faculty of Science at Lampang Rajabhat University. All documents have been classified and labeled by the correspondent staff according to the workload requirements of the university. The datasets were cleaned by preprocessing and classified by humans into eight classes of orders:

(0) being an advisor to individual projects/student research, (1) professional experience supervisors, (2) arts and culture work, (3) academic services, (4) advisors for student groups/clubs, (5) specific orders, (6) central work orders, and (7) administrative orders the label is given 0-7. The datasets are organization documents based on the official letters domain. Which were collected from the management information system of the Faculty of Science, Lampang Rajabhat University consisting of 3,489 pieces. All documents have been classified and labeled by the correspondent staff according to the workload requirements of the university. The official documents will have the same template; namely, the beginning of the document is the description of the order, the middle of the document is the list of directors, and the end is part of the authorized signatories. In examining order kinds, the beginning of the document will be primarily considered without regard to the rest; therefore, the beginning of the document is utilized to generate the data set.

2.2.2 Data Exploration

A pre-classified dataset of 8 classes after preprocessing and Exploratory Data Analysis (EDA) was performed. It found that each class had a total number of documents not equal for a total of 3,489 messages, as shown in Figure 1.

2.2.3 Data Cleaning

Our data cleaning step was developed with Python® 3.9.12 and Pythainlp library [12]. The data preprocessing steps include symbol removal, number removal, English word removal, whitespace and tap removal, single character removal, stop-word removal, and checking spelling errors [13].

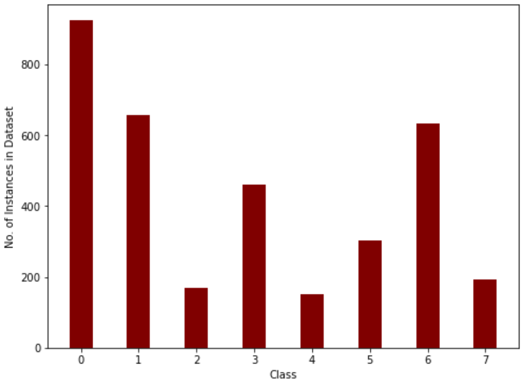

Figure 1: Class distribution of the dataset.

2.2.4 Train-Test Split Evaluation

The researchers divided the datasets for each class into two parts 80% training and 20% testing, as shown in Table 2.

Table 1: Number of datasets for training and testing in each class.

Label	Dataset		Label	Dataset	
0	train	739	4	train	122
	val	185		val	30
1	train	525	5	train	241
	val	131		val	61
2	train	136	6	train	507
	val	34		val	127
3	train	367	7	train	154
	val	92		val	38

Figure 1 and Table 1 show that the amount of data for each class varies significantly. The weighted-average F1 score was chosen to evaluate the classifier's performance.

2.2.5 Sampling method

Figure 1 was founded that our dataset is imbalanced. Therefore, we sampled for the experiment using Stratified sampling and Imbalanced

Dataset Sampling. Stratified sampling is a technique used to obtain samples representing the population. It reduces selection bias by dividing the population into homogeneous subcategories called strata and randomly sampling data from each stratum Stratify. By the way, sampling will preserve the proportion of the train and test datasets as in the original dataset [14]. Imbalanced Dataset Sampling is a technique to rebalance the class distributions when sampling from an imbalanced dataset. It oversamples minority classes and undersamples majority classes. In experimenting with Baseline models, we use Stratified sampling to ensure that each class is evenly distributed across our train/test splits. In addition, to the model training WangchanBERTa, we random sampling for the experiment with both methods.

The second method used ImbalancedDatasetSampler, a PyTorch sampler, to compare the predictive performance of the WangchanBERTa-generated classifier

2.3 Text Classification Model

We will now introduce how we build traditional models that we are comparing with WangchanBERTa and the details of the BERT and WangchanBERTa models.

2.3.1 Traditional Machine Learning Models

The model used in the experiment considered from research that model performs well in the multi-class classification task [15]. Therefore, multinomial NB, Linear SVM, and Logistic regression were used as the baseline for the experiment with the following steps:

2.3.1.1 Text feature extraction methods

The three baselines have to learn from a set of features from the training data to produce output for the test data. In this study, We use Scikit-learn's CountVectorizer and Tfidftransformer to extract data features. First, we create a CountVectorizer to count the number of words in a collection of raw documents. Then, Tfidftransformer will use to compute the word counts, generate IDF values and then compute a set of TF-IDF scores.

2.3.1.2 Hyperparameter Tuning

The hyperparameters used for fine-tuning the models are represented in Tables 2–4. For each model, we tune hyperparameters using grid search. The grid search method is exhaustive to find the optimal hyperparameters of a model, which results in the most accurate predictions. Its main idea is to create a hyper-parameter grid and try all their combinations. Then, the method will represent the score for the model by considering which one is best [16]. The models are trained according to the list, and their performance is evaluated. The best sets of hyperparameters were selected as chosen models.

Table 2: Hyperparameter setting for Multinomial NB.

Hyperparameter	Tuned Range
alpha	[1e-2, 2e-2, 3e-2, 5e-2,]
fit_prior	[True, False]

Table 3: Hyperparameter setting for Linear SVM.

Hyperparameter	Tuned Range
alpha	[1e-4,1e-3,1e-2,1e-1]
penalty	["l2", "l1", "none"]
random_state	42
max_iter	5

Table 4: Hyperparameter setting for Logistic Regression.

Hyperparameter	Tuned Range
C	[1e-3, 1e-2, 1e-1, 1, 10, 1e2,1e3,1e4, 1e5]
penalty	['l1', 'l2']
fit_intercept	True
random_state	42
solver	'liblinear'
max_iter	100

The configuring hyperparameters refer to the results of text classification experiments for the Thai language dataset [13]

2.3.2 BERT

BERT's model architecture is a multi-layer bidirectional Transformer encoder [5] based on the original implementation described in the work of Ashish Vaswani et al. [17]. Regarding the BERT model, there are two steps in its framework, including pre-training and fine-tuning [5]. For pre-training, the model is trained on a large unlabeled corpus. For fine-tuning, the model is initialized with the pre-trained hyperparameter parameters, and all the parameters are fine-tuned using the labeled dataset for specific tasks.

2.3.3 WangchanBERTa

For the experiment, we employ WangchanBERTa by selecting wangchanberta-base-att-spm-uncased, a pre-trained deep learning language model trained with a 78.5 GB Thai data set that configures hyperparameters to cut subword-level tokenization using the SentencePiece library. Regarding hyperparameters setting for our specific tasks. Learning rates were obtained from the RoBERTa paper [18]. Sequence length is limited to 256 token sequences in their base configuration. Training data is fed to the model in batches of size 32 to prevent RAM overflow. For Warmup and Dropout, we set them as 0.1 as they are fine-tuned for multi-class sequence classification tasks[9]. Finally, the epoch number was set to 8 epochs. The hyperparameters are listed in Table 5.

Table 5: Hyperparameters setting for WangchanBERTa.

Hyperparameter	Tuned Range
Learning rate	[1e-5,2e-5,3e-4,5e-5,7e-4]
Sequence length	[64,128, 256]
Batch size	32
Epoch	8
Warmup	0.1
Dropout	0.1

2.4 Evaluation Metric

We use performance metrics to evaluate our models, including accuracy, precision, recall, and F1-score. There are popularly used to measure multi-class classification. They are defined as follows [19].

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

$$\text{Precision} = \frac{TP}{TP + FP}$$

$$\text{Recall} = \frac{TP}{TP + FN}$$

$$\text{F1-score} = \frac{2\,Precision * Recall}{Precision + Recall}$$

The accuracy is the number of correctly predicted documents out of all documents. The precision represents the fraction of correctly predicted documents among all the indicated documents for a given class. At the same time, the recall represents the fraction of correctly predicted documents and all documents belonging to a given class. F1-score is the harmonic mean of the precision and recall. We used a weighted-average F-score because of using an imbalanced dataset in our experiment [20]. When each class's F1-score was obtained, the class's F1-score was weighted. The 'weight' essentially refers to the number of instances per class relative to the total number of cases [21].

All four metrics were obtained TP, TN, FP, and FN. Where TP is the number of positive documents correctly predicted as positive, TN is the number of negative documents correctly predicted as unfavorable. Moreover, FP is the number of negative documents incorrectly predicted as positive, and FN is the number of positive documents incorrectly predicted as unfavorable.

3. Results

This section will answer the three research questions we assigned in the introduction. We build classifiers of baseline models and WangchanBerta, where each classifier has a different hyperparameter, as in section 2 then evaluated the performance of each classifier using four performance metrics from the same dataset, running on Google Colab in a GPU runtime environment. The experimental results are as follows:

Table 6: Hyperparameter setting for baseline models.

Models	Optimal Values
MultinomialNB	alpha = 2e-2, fit_prior =True
Linear SVM	alpha = 1e-3, penalty = “l2”
Logistic regression	c=1e5, penalty = “l2”

3.1 Performance of Baseline Models

We train all classifiers on various hyperparameters to identify the ideal hyperparameter for our task. In Table 6, we report the results of tuning hyperparameters of each classifier with the grid search method. Table 7 details the ideal performance of baseline models at their best levels. Furthermore, the Multinomial NB earned 0.70 for accuracy, 0.71 for precision, 0.70 for recall, and 0.69 for F1-Performance, the lowest score among the Traditional models. Additionally, logistic regression achieved an accuracy of 0.73, precision of 0.74, recall of 0.73, and F1-Score of 0.73. Linear SVM earned 0.75 for accuracy, 0.75 for precision, 0.75 for recall, and 0.75 for F1-Score. Linear SVM earned an accuracy of 0.75, precision of 0.75, recall of 0.75, and F1-Score of 0.75. The Linear SVM model gives the best performance with the highest score on four measures.

Table 7: Performance of baseline models.

Model	A	P	R	F1
MultinomialNB	70	71	70	69
Linear SVM	**75**	**75**	**75**	**75**
Logistic regression	73	74	73	73

3.2 Performance of WangchanBerta Models

WangchanBerta Classifiers were trained with various learning rates and sequence lengths shown in Table 5, while performance was measured using a loss function and the weighted-average F1. This was checked after every training session. Figures 2 and 3 show that the training loss rates of the classifiers in groups (a), (b), and (d) gradually decrease after the number of training epochs increases. It has the

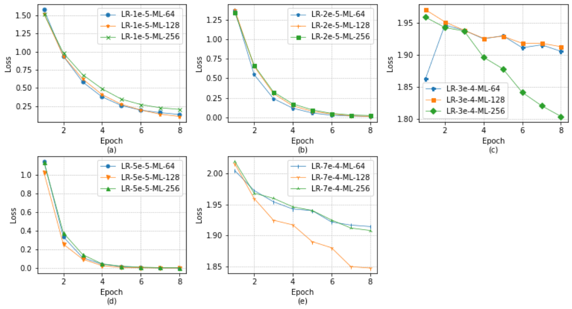

Figure 2: Training loss for WanchanBERTa classifiers.

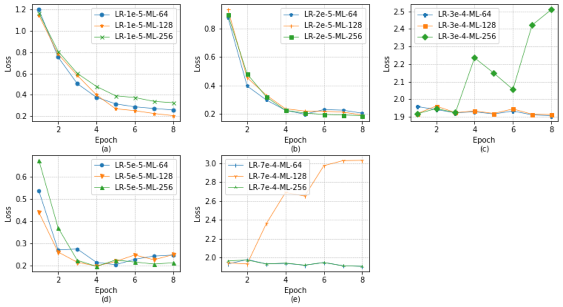

Figure 3: Validation loss of WanchanBERTa classifiers.

same decline when considered together with the validation loss rate. While the classifiers in groups (c) and (e) had a smaller decrease in training loss, it shows that learning is not so good. Compared to the validation loss rate, the validation loss in groups (c) and (e) is flattened out and rises, indicating that the classifier starts overfitting.

We further investigated the overfitting factor of the BERT model dropout [21]. Therefore, further experiments were performed for the classifiers in groups (c) and (e) to reduce overfitting by choosing a classifier. LR-7e-4-ML-128 and LR-3e-4-ML-256 re-train by gradually adding a dropout rate from 10% by default to 50%. After training, it was found that the overfitting rate of classifiers (f) and (g) decreased from the original, as shown in Figure 5. However, the performance evaluation results not increased much

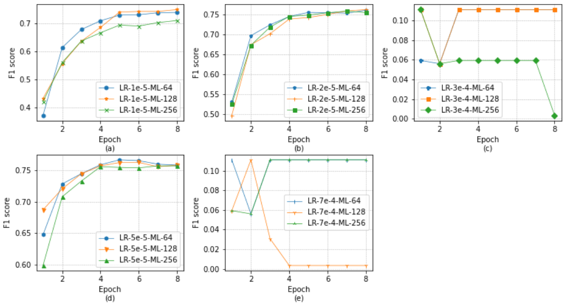

Figure 4: Performance of WanchanBERTa classifiers.

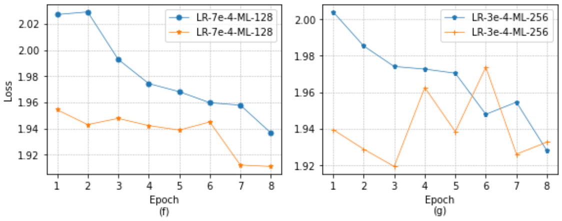

Figure 5: Training and validation loss of classifiers on adding dropout rate to 50%.

from the original. Therefore, only the classifiers in groups (a), (b), and (d) were taken into consideration to select the one with the highest performance scores for comparison with the baseline classifiers.

Figure 6 compares the training loss and validation loss of the best f1-score classifier, as depicted in Figure 4. The loss value decreased significantly during the first four epochs. It stabilized at epoch eight, with the training loss value being less than the validation loss value, indicating a tendency for the classifier to predict more accurately [22].

To answer this topic, we take all the classifiers generated from WangchanBERTa and test their performance with evaluation metrics, respectively accuracy, precision, recall, and F1 score, then select the highest-rated classifier against the best classifiers from the baseline models.

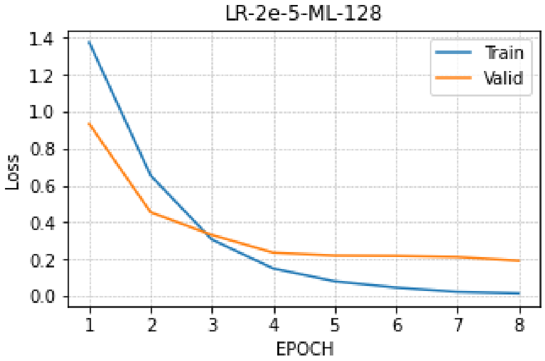

Figure 6: Training and validation loss for LR-2e-5-M-128.

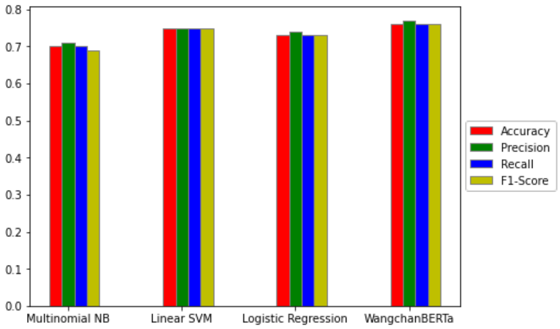

Figure 7: Comparison of Multinomial NB, Linear SVM, Logistic Regression, and WangchanBERTa.

Figure 7 shows the accuracy, precision, recall, and F1 score, a weighted average of the best classifiers of Multinomial NB, Linear SVM, Logistic Regression, and WanchanBERTa.

WangchanBERTa scored accuracy 0.76, precision 0.77, recall 0.76, and F1-Score 0.76 as the best score. In conclusion, WangchanBERTa outperforms the traditional models on the task of multi-class text classification for this experiment.

3.3 Impact of the Imbalanced Dataset on the Performance of WangchanBERTa

We experimented using a single classifier to train two datasets using different sampling methods. This is a separate case of stratified sampling, where the training dataset is distributed across all classes. Still, the dataset remains unbalanced and imbalanced dataset sampling using Imbalanced-DatasetSampler [23], which gives a more balanced data set. Then train the classifier and compare the performance.

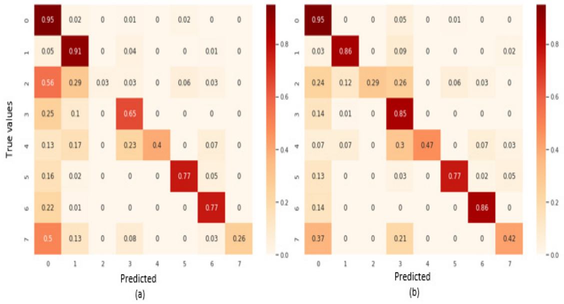

Figure 8: Confusion matrix for the LR-2e-5-ML-128 predictions.

Figure 8 shows the prediction result of the classifier "LR-2e-5-ML-128" where (a) it is the result of the classifier trained with Stratified sampling and (b) it is the same classifier trained with stratified sampling. Randomized dataset with Imbalanced Dataset sampling.

In the confusion matrix (a), we see that the minority classes [2, 4, 7], which have fewer samples [136,122, 154] respectively, are, indeed, having significantly fewer scores [0.03, 0.40, 0.26] as compared to the classes with the higher number of samples like [0,1,5,6]. When training with Imbalanced Dataset sampling, the result (b) was better, with minority classes 2, 4, and 7 having correct prediction rates must increase from (a) [0.29, 0.47, 0.42], indicating that the imbalanced dataset affects the performance of wangchanBERTa

4. Discussion and Conclusion

In this study, we provided the wangchanBERTa pre-trained model and the traditional NLP models to develop classifiers, and we refined it for multi-class text classification in order to classify our organization's documents according to pre-defined categories. We have introduced four different models where we have shown their performances and found that the BERT-based model outperforms the others. Although implementing wangchanBERTa, they were found to perform better than baseline models but were limited in learning unbalanced datasets. Moreover, to get better results based on such limitations without increasing the amount of data for training, we can improve the model's efficiency by rebalancing imbalanced dataset techniques. Furthermore, to get better results based on such limitations without increasing the amount of data for training, we can improve the model's efficiency by rebalancing imbalanced dataset techniques. For future work, we would like to optimize the model's learning by searching for other sampling methods that may improve the model's performance. Furthermore, the model can be enhanced by employing the Active Learning technique that enables models to learn better with fewer data and utilizing the data set selection process for training the model to determine if it can be optimized for the model in our task.

5. Acknowledgements

The experimental dataset was provided by the inventor of the Wangchanberta model and the Faculty of Science at Lampang Rajabhat University.

References

[1] A. K. H. Tung, “Rule-based Classification,” *Encyclopedia of Database Systems*, Springer, Boston, MA, 2009, pp. 2459-2462

[2] A. Rana, (2018, Oct.). Journey From Machine Learning to Deep Learning. Towards Data Science. [Online]. Available:https://towardsdatascience.com/journey-from-machine-learning-to-deep-learning-8a807e8f3c1c

[3] M. Marcus, “New trends in natural language processing: Statistical natural language processing,” *in Proceedings of the National Academy of Sciences 92.22*, 1995, pp. 10052–10059.

[4] C. J. Fall, A. Törcsvári, P. Fiévet, and G. Karetka, “Automated categorization of German-language patent documents,” *Expert Systems with Applications*, vol. 26 no. 2, pp. 269–277, 2004.

[5] D. Tikk, G. Biró, and J. D. Yang, “Experiment with a hierarchical text categorization method on WIPO patent collections,” i*n Applied Research in Uncertainty Modeling and Analysis.* Springer, Boston, MA, 2005. pp. 283–302.

[6] S. Minaee, N. Kalchbrenner, E. Cambria, N. Nikzad, M. Chenaghlu, and J. Gao, “Deep learning-based text classification: a comprehensive review,” *ACM computing surveys (CSUR)*, vol. 54, no. 3, pp. 1–40, 2021.

[7] J. Devlin, M. W. Chang, K. Lee, and K. Toutanova, “Bert: Pre-training of deep bidirectional transformers for language understanding,” arXiv:1810.04805, 2018.

[8] J. S. Lee, and J. Hsiang, “Patent classification by fine-tuning BERT language model,” *World Patent Information*, vol. 61, Art. no. 101965, 2020.

[9] L. Lowphansirikul, C. Polpanumas, N. Jantrakulchai, and S. Nutanong, “WangchanBerta: Pretraining transformer-based Thai language models,” arXiv:2101.09635, 2021.

[10] W. Meeprasert and E. Rattagan, “Voice of customer analysis on twitter for Shopee Thailand,” *Journal of information systems in Business JISB*, vol. 7, no. 3, pp. 6–18, 2021 (in Thai)

[11] National Archives of Thailand. Regulations of the Prime Minister's Office on Correspondence B.E. 2526 and No. 2 B.E. 2548. [Online]. (in Thai). Available: http://bit.ly/3JF9rQK

[12] W. Phatthiyaphaibun, K. Chaovavanich, C. Polpanumas, A. Suriyawongkul, L. Lowphansirikul, and P. Chormai. (2020, June). PyThaiNLP/pythainlp: PyThaiNLP 2.2.0 (v2.2.0). *Zenodo*. [Online]. Available: https://doi.org/10.5281/zenodo.3906484

[13] N. Khamphakdee and P. Seresangtakul, “Sentiment analysis for thai language in hotel domain using machine learning algorithms,” *Acta Informatica Pragensia*, vol. 10, no. 2, pp. 155–171, 2021.

[14] M. Merrillees and L. Du, “stratified sampling for extreme Multi-label data,” In Advances in Knowledge Discovery and Data Mining: 25th Pacific-Asia Conference, PAKDD 2021, Virtual Event, May 11–14, 2021, pp. 334–345.

[15] W. Arshad, M. Ali, M. M. Ali, A. Javed, and S. Hussain, “Multi-class text classification: Model comparison and selection,” in *2021 International Conference on Electrical, Communication, and Computer Engineering (ICECCE)*, 2021, pp. 1–5.

[16] M. M. Ramadhan, I. S. Sitanggang, F. R. Nasution, and A. Ghifari, “Parameter tuning in random forest based on grid search method for gender classification based on voice frequency,” *DEStech transactions on computer science and engineering*, vol. 10, pp. 625–629, 2017.

[17] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, “Attention is All You Need,” *Advances in Neural Information Processing Systems*, Curran Associates, Long Beach, CA, USA, pp. 2–11, 2017.

[18] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, “RoBERTa: A Robustly Optimized BERT Pretraining Approach,” arXiv:1907.11692, 2019.

[19] M. Grandini, E. Bagli, and G. Visani, “Metrics for multi-class classification: an overview,” arXiv:2008.05756, 2020.

[20] G. Menardi, and N. Torelli, “Training and assessing classification rules with imbalanced data,” *Data mining and knowledge discovery*, vol. 28, pp. 92–122, 2014.

[21] S. El Anigri, M. M. Himmi, and A. Mahmoudi, “How BERT's dropout Fine-tuning affects text classification?,”in *Proceedings Business Intelligence: 6th International Conference, CBI 2021,* Beni Mellal, Morocco, 2021, pp. 130–139.

[22] A. Abdulwahab, H. Attya, and Y. H. Ali, “Documents classification based on deep learning,” *International Journal of Scientific & Technology Research,* vol. 9, no. 2, pp. 62–66. 2020.

[23] Torchsampler0.1.2. (2022, May) Imbalanced Dataset Sampler. [Online]. Available: https://pypi.org/project/torchsampler/