IEEE Access
Multidisciplinary ⁝ Rapid Review ⁝ Open Access Journal

Received 25 July 2023, accepted 22 August 2023, date of publication 28 August 2023, date of current version 1 September 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3308951

RESEARCH ARTICLE

SETAR: Stacking Ensemble Learning for Thai Sentiment Analysis Using RoBERTa and Hybrid Feature Representation

PREE THIENGBURANATHUM[1,3] **AND PHASIT CHAROENKWAN**[2,3]
[1]Department of Software Engineering, College of Arts Media and Technology, Chiang Mai University, Chiang Mai 50200, Thailand
[2]Department of Modern Management Information and Technology, College of Arts Media and Technology, Chiang Mai University, Chiang Mai 50200, Thailand
[3]Research Group of Modern Management Information and Technology, College of Arts, Media and Technology, Chiang Mai University, Chiang Mai 50200, Thailand

Corresponding author: Pree Thiengburanathum (pree.t@cmu.ac.th)

This work was supported in part by the Thai National Research Council under Grant 910250, in part by the KhonThai 4.0 Project under Grant 25307, and in part by the College of Arts, Media, and Technology at Chiang Mai University.

ABSTRACT Sentiment classification of social media posts is among the most challenging and time-consuming tasks for analysts. This is particularly true when applied to languages that employ scriptio continua, such as the Thai language, in which there are no spaces between written words and where there is no end of sentence punctuation. Thai is considered a scarce-resource language as few datasets are available to researchers. Although machine-learning (ML) and deep-learning (DL) algorithms can identify sentiment classification polarity, the performance of the existing classification models are still inadequate. This study proposes a novel stacking ensemble learning technique for identifying sentiment classification polarity in the Thai language, SETAR. Our stacking ensemble strategy utilized the pre-trained Thai language model (WangChanBERTa), based on a Robustly Optimized BERT Pretraining Approach (RoBERTa) architecture to form a feature vector. This feature was combined with three distinct feature vectors obtained from three well-known categories, namely Word2Vec, TF-IDF, and bag-of-words, as a new hybrid sentence representation. The base learners were trained using seven chosen complex heterogeneous ML algorithms, including support vector machine (SVM), random forest (RF), extremely randomized trees (ET), light gradient boosting machine (LGBM), multi-layer perceptron (MLP), partial least squares (PLS), and logistic regression (LR) to enable the development of the final meta-learners. The results revealed that our proposed stacking ensemble model outperformed the baseline models of all classification metrics among the training and test sets, as was determined by extensive benchmarking, carried out on the four datasets, which included our developed sentiment corpus that domain experts annotated.

INDEX TERMS Deep learning, ensemble learning, sentiment analysis, text classification, transfer learning.

I. INTRODUCTION

In the past decade, the amount of textual data that has been generated by social media has increased exponentially, as have the number of users. Social media has continuously evolved into an essential channel of communication between business owners and customers. Customers typically share their thoughts and opinions about their product-purchasing experiences in social media posts, and discussion forums, as well as in online articles and reports [1]. The abundance of user-generated content of textual data found on social media platforms, particularly textual data in product review e-commercials, has piqued the interest of researchers and business owners in sentiment discovery [1], [2]. However, the language used in social media is informal and generally written in slang, making pre-processing problematic, and identifying sentiment from a large amount of unstructured textual data is a complex challenge and considered a laborious

The associate editor coordinating the review of this manuscript and approving it for publication was Xiong Luo.

task for humans. Thus, the automation of such a process that can perform on par with a human identification capacity would be highly beneficial.

The computational predictive models used for various practical and intelligent applications typically utilize Natural Language Processing (NLP). Within this context, sentiment analysis (SA) has been developed and employed to automate the process [3]. SA has advanced by way of certain syntax-driving techniques, such as word-counting, to establish a genuine comprehension of human language. Recently, grammatical, contextual, and both semantic and syntactic constraints were taken into account to bridge the gap between human and computer interaction. The successful application of SA such as sentiment classification has gone well beyond listening to people's emotions with regard to the sale of products. The insight gained from effective SA could then be expanded to include the stock market within the financial domain. Furthermore, it could also be successfully applied as a social listening technique in politics or even to detect depression within a medical domain [4], [5], [6].

It has been determined that through the use of ML based models exhibited an adequate level of classification performance and outperformed the lexicon-based models [7]. Modern data-driven techniques, such DL along with state-of-the-art transformers, have demonstrated significant potential for sentiment classification [8], [9]. Recently, with ensemble learning techniques, a collection of learning models are generated either in parallel or sequentially, and wherein separate predictions are merged [10], have been employed in sentiment classification which demonstrates strong generalizability gains [11], [12]. Nonetheless, the number of studies that have employed ensemble learning in sentiment classification is still limited, especially when it comes to examination of the robustness of the model. In addition, a selection of computational models to establish base-learners, along with investigation of model performance when combined with available DL models or pre-trained language models in ensemble techniques, remain to be explored [13], [14].

With regard to sentiment classification in the Thai language, the same challenges must be confronted in the text of social media posts as with other languages, particularly in the development of quantity and quality corpus. To determine the appropriate degree of sentiment in a sentence, language experts are required. Concerning the examination of scarce-resource languages, such as Thai, only a few studies have been conducted, and there are a limited number of publicly available quality datasets. Only three datasets annotated by qualified annotators are publicly accessible and employable for bench-marking purposes in order to perform a Thai language sentiment classification task. These datasets include data that were obtained from Wisesight [15], Thai toxic tweets [16], and Thai tales [12]. The remaining datasets that have been appeared in studies, have been kept private and dataset details have not disclosed. Moreover, many of them contain a very small number of trainable examples, such as Thai sentence Wiki [17], which is comprised of 600 samples, and Thai depression dataset with only 944 samples [5].

To the best of our knowledge, there is a lack of comprehensive analysis of SA in the Thai language, such as an evaluation of text representations of machine learning (ML) algorithms. Only two research studies have done a limited comparison study [18] compared nine traditional ML models with term frequency-inverse document frequency (TF-IDF) and Word2Vec on a private dataset, without the results of an independent test dataset having been evaluated. Reference [12] only compared two DL models with three embedding methods (e.g., POS-Tagging, Sentinet, and Word2Vec). To construct a robust classifier capable of achieving this desired standard in the Thai language, it is necessary to address all three above-mentioned concerns. This involves developing a comprehensive Thai sentiment analysis dataset with language experts, conducting an extensive comparison of ML and DL models with well-known text representations, and exploring ensemble learning strategies to increase the robustness and accuracy of the sentiment classification model.

In an attempt to respond to the abovementioned challenges, a novel stacking ensemble learning model for Thai sentiment analysis, named SETAR (Stacking Ensemble learning for Thai sentiment Analysis using RoBERTa and hybrid feature Representation), is proposed to improve the classification performance of the sentiment classification task in Thai language. The main contributions of this research study can be summarized as follows.

1) SETAR is the first stacking ensemble model in sentiment classification in the Thai language and was constructed from seven selected complex heterogeneous ML algorithms to provide accuracy and robustness. The base-learners used a pre-trained Thai language model (WangchanBERTa) as the DL based feature. This feature was then integrated with three text representations included Word2Vec with average embedding, TF-IDF unigram, and bag-of-words (BOW) unigram to form a new hybrid text representation. This resulted in a new state-of-the-art model for a variety of Thai sentiment classification datasets.
2) Systematic bench-marking experiments were conducted. Eleven well-known ML algorithms that utilize five various types of text representations were targeted for investigation. The text representations include BOW (unigram, bigrams, and 1-2 grams), TF-IDF (unigram, bigrams, and 1-2 grams), Word2Vec with average and TF-IDF embedding vector, Dictionary-based constructed from a list of negative and positive words for both BOW and TF-IDF, and part-of-speech tagging (POS-tagging). We analyzed and investigated the performance of eleven ML algorithms, including SVM, MLP, RF, ET, PLS, LR, decision tree (DT), k-nearest neighbor (KNN), extreme gradient boosting (XGB), light gradient boosting machine (LGBM), and

naive bayes (NB) on both training and testing dataset while employing four state-of-the-art DL algorithms, including CNN, BiLSTM, BERT, and WangchanBERTa. All the models were then evaluated using four classification metrics, including accuracy rate (ACC), precision (PRE), recall (REC), and F1-score (F1), on our developed dataset, and three publicly available datasets.

3) We collected and annotated our own comprehensive sentence-level Thai sentiment analysis corpus. To the best of our knowledge, our dataset is the largest Thai sentiment social media dataset that has been annotated by Thai language experts. Moreover, our developed dataset is the first short-sentence dataset in Thai sentiment analysis.

The organization of this paper is as follows. Section II reviews related research works. Section III discusses the datasets, text representations, and all the baseline models that were used in this study. Section III also presents the proposed novel stacking-ensemble in detail. The experimental results, and the evaluation analysis are discussed in the end of this section. The last section offers a tentative conclusion and outlines future research work. Link to the source code of the experiment can be found at https://github.com/preenet/SETAR.

II. BACKGROUND AND RELATED STUDIES

A. MACHINE-LEARNING AND DEEP-LEARNING TECHNIQUES IN SENTIMENT ANALYSIS

Traditional and diverse ML algorithms have been broadly applied in sentiment analysis tasks by many researchers. For example, [2] compared three ML algorithms, including SVM, RF, and NB, on the Amazon product review dataset, and POS-tagging was used as a sentence representation. Another recently conducted research study investigated the efficacy of five different ML algorithms, namely RF, LR, SVM, NB, and DT in identifying the sentiments present in tweets about COVID-19 [19].

Textual datasets which typically contain a large number of samples and features are being employed. To achieve better model performance for such a dataset, DL-based models have the potential to extract better features than traditional ML-based models, has recently gained a lot of attention. Convolutional neural network (CNN), recurrent neural network (RNN), long short-term memory (LSTM), and gated recurrent units (GRU) architectures have been studied, as has been evidenced by the existing range of published works [19], [20], [21]. CNN [22] has performed admirably with image data. Recently, it has begun to be successfully applied in the field of NLP, as well as in the sentence classification task. The CNN model has provided a pooling layer which is a key feature that can be used to reduce the size of the text features.

To handle the gradient explosion and disappearances in RNN. LSTM [23] presents the input, output gates, and cell unit. BiLSTM is an extension of LSTM, and has a new mechanism, in which both forward and backward input sequences x are learned with regards to time t can be defined as $x = x\{x_1, x_2, \ldots, x_t\}$. The forward of input for LSTM from the left can be defined as $\vec{h}_t = f\left(x, \vec{h}_{t-1}\right)$. For the opposite direction is $\overleftarrow{h}_t = f\left(x_t, \overleftarrow{h}_{t-1}\right)$. This mechanism improves the network's information flow. GRU is an extension of LSTM that is simpler and faster to train. Recently in SA, DL techniques that operate in conjunction with the convolution CNN, LSTM, BiLSTM, and GRU have been investigated to produce an even more accurate model, as can be seen from the work [24], [25], [26]. Word embedding, involving Word2Vec, fastText, and Glove, was successfully applied in sentiment classification studies, in order to capture sufficient sentiment information [26], [27].

A number of research studies have also been conducted that combine DL-based models (CNN-LSTM-GRU) and ML-based models. For example, [29] applied BiGRU as a feature extractor and then used conditional random field (CRF) as a head classifier. The model was found to outperform the conventional neural network (NN) based models. Reference [30] created a hybrid NN of CNN-BiLSTM, where CNN was used as a feature extraction mechanism. Lately, researchers have started to modify NN architecture in order to increase classification performance. For instance, [31] utilized a Glove embedding vector with multi-layers of BiLSTM. This model resulted in a 3% improvement in performance when compared with previous research studies.

Even though LSTM-based models have shown remarkable success across several fields, the models still have some limitations. These include the significant lengthier amount of time required to train the model when compared to CNN and GRU, as well as a certain amount of data-loss that occurs when more training is required. Pre-training of Deep Bi-direction Transformer [32] can be used in sentiment classification task. For instance, a pretrained language model (BERT) was applied to SA task in aspect-based and sentence level, with a fine-tuning method [9], [33]. The transformer based model is the current state-of-the-art model in sentiment analysis, such as in analyses that employ the pre-trained language models BERT, RoBERTa [34], GPT-2 [35].

B. ENSEMBLE TECHNIQUE IN SENTIMENTAL ANALYSIS

Bagging [36], also known as bootstrap aggregation, samples the population for training utilizing random sampling with replacement (i.e., cases can be taken several times for the sample, and they are not ignored from the dataset once selected). The remaining non-selected samples were allocated to the validation set. To determine the final prediction, either majority vote or plurality voting was performed. Boosting [37], the idea behind the boosting approach is to construct a strong learner from a sequence of weak learners. Boosting technique operates by successively training a group of classifiers sequentially and combining them for prediction. The latter learner strengthens and concentrates more on the errors of the earlier learners. Both ensemble learning techniques have been investigated for use in SA tasks, mainly to enhance predictive accuracy. There are a few available of ensemble learning studies that appear in the SA literature. Reference [13] compared three traditional ensemble strategies: bagging, boosting, and random subspace, employing five traditional ML algorithms as base-learners [13]. Random subspace with SVM provided the best accuracy (ACC) on four benchmark datasets.

Recently, the Stacking ensemble appeared in the SA task. Stack ensemble introduced by [38], is a supervised learning method comprised of learners that typically consist of two levels including base learners and meta learners models. Stacking ensemble generates a final prediction from the meta-learner by combining the predictions of the base-learners that trained from the same set of a dataset. For instance, [14] demonstrated various type voting stacking ensemble strategies. The authors employed six traditional ML algorithms along with CNN and LSTM as base-learners, with BOW text representation, and employed LR as only one option for meta learner. The results revealed a 5.5% improvement in terms of ACC on three product reviews and one banking survey dataset.

Lately, a research study was conducted that employed ensemble learning with a language model. Currently, there is a fusion between language model, DL, and traditional ML to improve classification performance. For example, the authors [39] proposed a stacking ensemble of DL models, involving RNN, LSTM, and GRU for the Arabic language. Three meta-learners were investigated including LR, RF and SVM. The results showed that Stacking-LR provided the highest ACC of the three cases. In a recent study [28], the authors combine linguistic-based analysis with ensemble learning techniques to analyze Tamil text found in YouTube comments. The text representations were generated by employing a combination of data stemming and the utilization of the MuRIL pre-trained transformer developed by Google. These representations were then used in an ensemble-learning approach, where the majority vote of different ML was employed to make predictions.

C. METHODS IN THAI SENTIMENT ANALYSIS

It is commonly recognized that the Thai language has specific challenges with regard to performing basic NLP tasks, such as word and sentence segmentation. At present, a number of these obstacles have been overcome as current advancements in word tokenization has reached a reasonable degree of prediction. However, sentiment classification can be enhanced for improvement in terms of the model's robustness, generalizability, and practicality.

Aside from the limitations of the established datasets developed for the Thai language that we have mentioned; a few studies have attempted to provide a comparison of ML and DL based models. For instance, [40] combined a genetic algorithm with four traditional ML algorithms that consist of SVM, DT, NM, and KNN on private Thai online product reviews, and collected 4k samples from agoda.com and booking.com. The results revealed that SVM with GA achieved the highest predictive performance in terms of ACC at a level of 0.88. Reference [18] developed a private hotel review dataset from agoda.com and booking.com that was comprised of 16k of labeled samples. The authors made comparisons of the two feature extraction methods: TF-IDF and word embedding with nine traditional ML algorithms, including SVM, BNB, DT, LR, RF, stochastic gradient decent (SGD), ridge regression (RR), passive aggressive (PA), and AdaBoost (ADA). The findings of the experiments indicate that SVM employing the Delta TF-IDF approach was the most effective, with an ACC of 0.89 on training set.

Previously, there were two pre-trained language models that supported the Thai language including mBERT [32] and XLMR [41]. Both models were trained on multi-language and only includes the Thai wiki dataset. Simultaneously, WangchanBERTa [42], a Thai language model was constructed using the RoBERTa model architecture, a robustly optimized form of the BERT [34]. WangchanBERTa employed SentencePience [43] as tokenizer, and was trained with fewer steps than its original, used dynamic masking and was trained on longer sequences through the use of a combination of existing training corpuses. The resulting data was investigated for sentiment analysis and other related tasks.

Recently, [12] proposed a hybrid of deep-learning models with a voting ensemble strategy in pursuit of Thai sentiment classification performance. Different combinations of CNN and BiLSTM were applied across three text representations, involving Word2Vec, POS-Tagging, and Sentic-Tagging with different voting mechanisms, such as soft voting and concatenation. Their experimental results revealed that a combination of BiLSTM and CNN with soft voting achieved a decent level of classification performance on three different Thai language corpuses. In terms of the performance of the Wisesight and Thai children stories datasets, the proposed combination of these features, along with the applied ensemble techniques, yielded F1 of 0.55, and 0.72 for the testing dataset, respectively.

III. MATERIALS AND METHODS

A. BENCHMARK DATASETS

This study involved the use of four benchmark datasets. Three of them are from social texts, and one is from a written textbook. All the datasets used in this experiment have been categorized at the sentence level and were manually annotated by humans. These datasets present different sentiment classification problems involving binary and multi-class classification. The datasets are represented in various domains, encompassing product reviews, food, autos, mobile phones, and even short sentences from published textbooks.

TABLE 1. Summary description of the thai sentiment datasets.

Datasets	Domain	#Sample	Avg. sentence length	#Unique words	# Classes
The 40 Thai Children Stories (TT)	written book	1,964	84.4	1,950	3
Thai Toxic Tweet (TX)	social media	2,094	103.4	2,094	2
Wisesight (WS)	social media	26,737	107	26,383	4
KhonThai (KT) (ours)	social media	60,081	52.6	59,836	3

This ensures that our proposed model can be generalized to a diverse range of textual sources. Table 1 presents the domain and characteristics of each dataset used in this study.

Khonthai (our dataset): Due to the scarcity of available Thai datasets and the need to evaluate the generalizability of our own proposed model, we developed a most comprehensive Thai sentiment dataset. The textual information was collected from 4k posts on Pantip.com, a Thai online public discussion forum, between late 2019 and mid-2020. The online product posts contain opinions related to cosmetics, food, dietary supplements, and skin-care items. With regard to data cleaning, sentences exceeding 1,000 words and duplicated sentences were removed. In addition, sentences that were not written in Thai (less than 50 percent Thai in a sentence) were filtered out using a language detection library [46] as were sentences that contained any personal information.

To handle excessively long comments, we performed sentence tokenization using a Conditional Random Field (CRF) model constructed from four distinct datasets established in our previous research work [47].

Regarding the data annotation process, three Thai linguists from the Faculty of Humanities, and the Department of Thai language, Chiang Mai University, were responsible for the annotation of the sentiment. Before the annotation process started, all annotators were provided with explicit annotation guidelines that specified which symbol to use for each class target (e.g., neu, pos, and neg). Pilot annotation of 100 samples were given to each annotator before they can begin the full-scale annotation. The final class target was determined by majority/hard voting calculated by mode, such that a comment that has the most votes from the annotators for the sentiment polarity will be the class target. In addition, during the annotation phase, we hold a weekly meeting to ensure that any ambiguities are clarified.

Presently, our developed dataset is the largest Thai sentiment analysis dataset to have been manually annotated by experts. Finally, the dataset contains 60,081 samples comprised of three class targets, 9,661 negative, 38,054 neutral, and 12,366 positive sentences, as shown in Table 1.

Wisesight [15]: This corpus is made up of Thai social media posts that were posted from 2016 to 2019 and which concerned restaurants, hotels, drinks, and cars. The corpus contains 26,737 examples that were manually labeled by three annotators as 6,823 negative, 14,561 neutral, 4,778 positive, and 575 questions. The data is downloadable from https://huggingface.co/datasets/wisesight_sentiment

Thai toxic tweet [45]: This corpus is comprised of 2,104 tweets and consists of two class targets; 1,291 toxic and 813 non-toxic tweets. The authors collected posts from Thai users of Tweeter API which appeared from January to December, 2017 and which were manually annotated by three hired annotators in order to classify positive and negative tweets. The data can be downloaded from https://huggingface.co/datasets/thai_toxicity_tweet

The 40 Thai Children Stories [44]: This dataset contains 1,964 Thai sentences derived from 40 children's stories. Three annotators categorized the data as follows: 451 negative, 940 neutral, and 573 positive sentences. Downloadable data may be obtained from https://github.com/dsmlr/40-Thai-Children-Stories

All the above-mentioned datasets were pre-processed by administering the follows steps:

- All the URLs and emoticons were replaced by special tags.
- Word tokenization was performed using a maximal matching algorithm that employed a dictionary-based approach and Thai character cluster method obtained from PythaiNLP [48]
- Words that contained less than two syllables were removed.
- Repetition of words were replaced by the original word.

B. TEXT REPRESENTATION

The objective of sentiment sentence extraction is to identify the most informative and condensed set of features. The performance of the ML classifier is dependent upon how the features are represented, making the selection of a suitable text representation an essential step [29] In this study, we investigated and utilized five distinct text representations. In the following sub-sections, each method will be described in greater depth.

1) BAG OF WORDS (BOW) [49]

In a matrix, a sentence is represented by a vector whose dimensions are determined by a set of comments that are denoted by $C = \{c_1, c_2, \ldots, c_n\}$, while the set of features is denoted by $F = \{f_1, f_2, \ldots f_n\}$ as a presentation of the feature that quantifies the frequency of the simple terms present in each comment. This is known as a bag of words.

2) TERM FREQUENCY-INVERSE DOCUMENT FREQUENCY (TF-IDF) [50]

To compare the similarity of the terms, term frequency (TF) and (document frequency) can be used. TF presents the occurrence of unique words in a comment while DF indicates how many times that word appears in the corpus. Accordingly, a log scale is used for normalization purpose. Equation (1) is defined as the calculation of TF-IDF.

$$TF - IDF = TF_{d,t} \cdot log\left(\frac{|D|}{df_t}\right) \quad (1)$$

For BOW and TF-IDF, both unigram (i.e., set of unique word) and bigram (i.e., combination of two terms) and combination of 1 gram and 2 grams were utilized in the experiment.

3) PART OF SPEECH TAGGING (POS-TAG)

Once the sentences were tokenized, every word in each of the sentences was tagged using the Orchid corpus based on universal dependencies [51]. The dataset was established to classify words based on 17 common part-of-speech classes, for example, noun (NOUN), verb (VERB), pronoun (PRON), adjective (ADJ), etc. Moreover, we created an additional custom token for any emoticons present in the comments. With regards to our POS tagging scheme, we flattened the list of words, so that each word was immediately followed by its tag. For example, [WORD_1, TAG_1, WORD_2, TAG_2,..., WORD_n, TAG_n].

4) DICTIONARY-BASED EXTRACTION METHOD

This extraction method guarantees the training speed and effectiveness of the model [52], but the development of a custom dictionary of sentiment words, particularly those in the Thai language, can be time-consuming. In this method, we merged two existing previously compiled lists of 512 positive and 1,313 negative Thai words as features. We then fit and transformed each sentence using BOW and TF-IDF extraction methods.

5) WORD EMBEDDING

Word embedding [53] is a prevalent text representation technique that is used in machine learning and recently has been applied to a sentence classification task [21], [54]. As can be seen from previous works, word embedding is particularly effective when applied to DL models. Importantly, it produces smaller feature spaces than the BOW and TF-IDF methods. However, word embedding also approximates the meaning or context of each word in every sentence. This would include both semantic and syntactic data. Accordingly, each word is defined by a point in the embedding space, and these points are learned and relocated based on the surrounding words. Word embedding is typically deployed as the first layer, or embedding layer, for NN-based models. In this study, we used 300-dimensional Word2Vec [50] for ML based models.

6) BASE LINE MODELS

To evaluate our proposed model, baseline models need to be developed. Eleven ML algorithms and four DL algorithms including CNN, BiLSTM, BERT, and RoBERTa were investigated.

a: CONVENTIONAL MACHINE-LEARNING BASED

Eleven well-known ML algorithms were investigated to evaluate their performance. These included a combination of diverse traditional and complex classifiers comprised of Support Vector Machine (SVM), Decision Tree (DT), Random Forest (RF), XGBoost (XGB) Extra trees (ET), Light Gradient Boosting (LGBM), Logistic Regression (LR), K-nearest neighbor (1NN), Multi-layer Perception (MLP), Naive Bayes (NB), and Partial Least Squares (PLS).

b: DEEP-LEARNING BASED

CNN: The CNN architecture employed in this study adapted from [21] consisted of five layers: an embedding layer with a non-pretrained word vector W of size $v \times d$ where v, $d = 300$. The embedding vector is connected to a 1D convolutional layer with the number of features equal to 100 and filter region sizes comprises of 3,4, and 5 respectively. Followed by one of max pooling the 1D layer is performed on each filter. The subsequent layers involve a merged layer, and a dropout layer as the output. ADADELTA [55] was used as a optimizer for this model.

c: BiLSTM

The first layer of the BiLSTM that we implemented consists of a non-pretrained embedding layer connected with two bidirectional LSTM layers that can parse the sentence in both directions. The architecture of the model is quite similar to the CNN including the embedding size, except that the model doesn't have the max pooling layer. For each LSTM layer, we merged the outputs from forward and backward cells as shown in Eq. (2). With regards to the hyper-parameters for BiLSTM in this experiment, we utilized the same configuration as the previously mentioned CNN, except for the optimizer and its learning rate values, in which we used the Adam optimizer [56].

$$h = concat\left(\vec{h}, \overleftarrow{h}\right) \quad (2)$$

d: BERT

The original pre-trained BERT-base from Google excluded Thai language as one of 103 languages, due to the difficulties that were associated with word segmentation. However, in this study, in order to develop a base line model, we fine-tuned the latest BERT-base-multi-language model [32], comprised of 104 languages that had previously trained with Thai Wikipedia data published in 2018. With regard to the model architecture, the first layer, known as the BERT layer, consisted of 12 layers with pre-trained weights that functioned as an embedding layer. This was followed by a classifier head as a dense layer with 32 fixed hidden nodes using ReLu activation function and SoftMax that served the last layer.

TABLE 2. Statistics of train, validation, and test dataset split of each dataset split of each dataset.

	Train	Validation	Test	Total
TT	1,178	393	393	1,964
TX	840	280	280	1,400
WS	16,042	5,347	5,348	26,737
KT	36,048	12,016	12,017	60,081

e: WangchanBERTa

For each dataset, we utilized the pre-trained Thai language model as a new feature representation, namely wanchabeta-base-att-spm-uncased. The architecture of WangchanBERTa shares the same similarity with RoBERTa, consists of 12-layer, 768-hidden size, and 12-attention heads.

7) EVALUATION METRICS

The following four measurements that were related to classification performance were selected and employed to evaluate the performance of the models. The first is the classification referred to as accuracy. With regards to imbalanced datasets, a measurement of accuracy alone was insufficient; therefore, measurements of precision, recall, and F1-score macro-averaged were utilized. These metrics were defined using Eq. (3)-(6), respectively.

$$ACC = \frac{TP + TN}{TP + TN + FP + FN} \tag{3}$$

$$Pre = \frac{TP}{TP + FP} \tag{4}$$

$$Rec = \frac{TP}{TP + FN} \tag{5}$$

$$F1 = 2 \times \frac{Pre \times Rec}{Pre + Rec} \tag{6}$$

Accordingly, the true positive, true negative, false positive, and false negative terms are represented by the letters TP, TN, FP, and FN, respectively.

C. CONSTRUCTION OF SETAR

This study proposed a novel stacking ensemble strategy consisting of five main stages, as denoted in Fig. 1. In the first two stages, (1) for each dataset, (2) the preprocessed datasets were split into training, validation, and test sets, at a ratio of 60:20:20, respectively. Table 2 describes the number of samples regarding split strategy of each benchmark dataset. TT represents the Thai children stories dataset, TX represents the Toxic Thai tweet dataset, WS represents the Wisesight dataset, and KT represents our proposed Khonthai dataset. A 10-repeated random stratified sampling ($k = 10$) was employed to avoid any bias that occurred from random sampling. To ensure that the experimental results can be reproduced, random state seeds were fixed the prescribed range [0,9] when we performed random hold-out sampling.

Moreover, for all the non-deterministic related models, random seeds were set to 0. Third, (3) each ML-based model consisted of 10 different feature extraction types from five conventional text representations, including BOW with 1-gram, 1-2-grams, 2-grams, and TF-IDF 1-gram, 1-2-grams, and 2-grams weights. DICT-based and POS-TAG were generated with 1 gram. Average embedding and TF-IDF embedding were carried out to generate a Word2Vec embedding layer for the ML-based models. Next, features were transformed, and then scaled using maximum absolute scaling as has been defined by (7), before fitting to ML models.

TABLE 3. Search details of the hyper parameters of all models.

Model	Parameters	Range
SVM	penalty cost	[1-32] in log2 steps
RF	n_estimators	[20, 50, 100, 200]
MLP	hidden layer sizes	[20, 50, 100, 200]
ET	n_estimators	[20, 50, 100, 200]
XGBoost	n_estimators	[20, 50, 100, 200]
LightGBM	n_estimators	[20, 50, 100, 200]
NB	default	default
1NN	n_neighbors	1
DT	default	default
LR	C	[0.001:100:10]
PLS	default	default
CNN	learning rate, dropout, epochs, and batch size	[0.25], [0.1:0.5:0.1], [20:50:10]
BiLSTM	learning rate, dropout, epochs, and batch size	[1e-4], [0.1:0.5:0.1], [20:50:10]
BERT	learning rate	[2e-5, 5e-5]
WangchanBERTa	learning rate	[1e-5, 2e-5]

Accordingly, (4) we trained 114 baseline models (10 feature extractions × 11 ML-based + 4 DL-based). Due to our computational limitations, particularly for our KT corpus which comprised 60k sentences, we omitted terms that appeared in fewer than 20 sentences when trained with BOW and TF-IDF. Intel Extension for Scikit-learn was patched together with the API of scikit-learn to minimize the SVRs training time. An iterative search was employed to find optimal hyper-parameters for each ML algorithm. We also trained four DL algorithms involving CNN, BiLSTM, and BERT with non-pretrained word embedding. Bayesian optimization [57] was employed for the deep learning models. The search range of the hyper-parameters are presented in Table 3. The highest validation classification accuracy was used to select the optimal model. We then combined the training and the validation sets (i.e., 80 percent of the sample of the datasets.) as representative of a new training set, and the testing set was used to evaluate the model for true performance.

$$X_{scaled} = \frac{x}{\max(|X|)} \tag{7}$$

With regard to the base-learner selection in our stacking ensemble, we selected top ML models that performed well on the four benchmark datasets. Due to the class imbalance in the four datasets, the baseline model that produced the highest F1 on the training datasets was regarded as the model with the best performance. For each base learner, we applied stratified 5-folds cross-validation ($k = 5$) to obtain predicted probabilities. The hyperparameters were

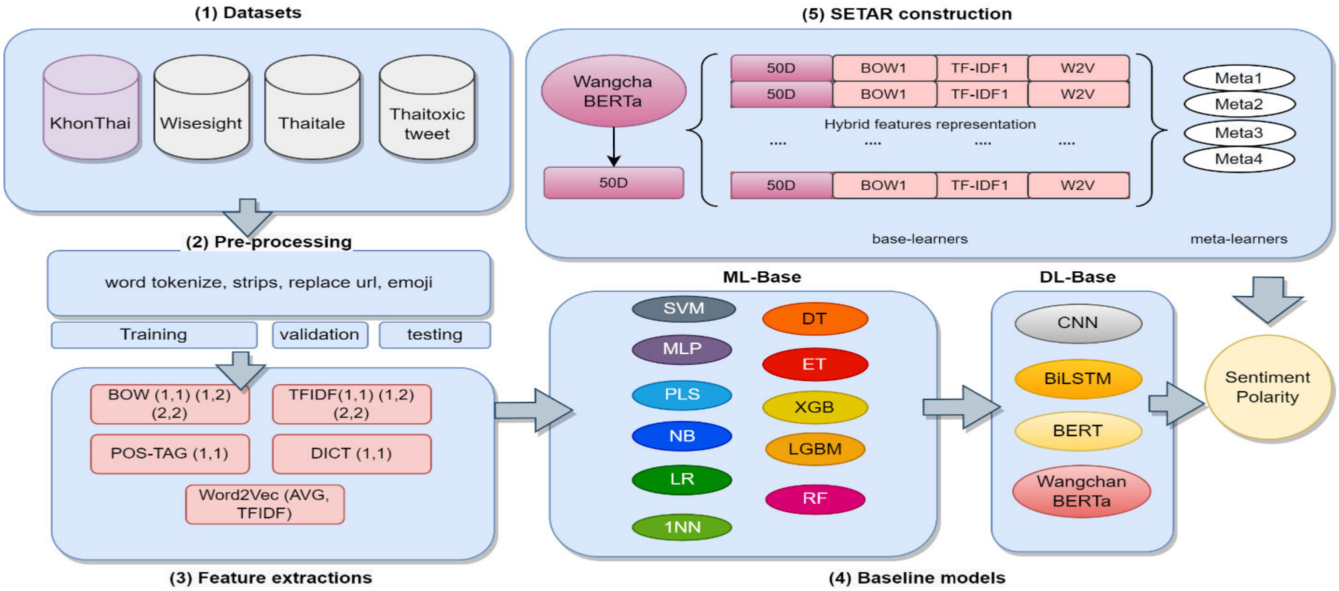

FIGURE 1. The proposed stacking ensemble framework for predicting sentiment polarity in Thai language involving five steps (1) datasets acquisition (2) text pre-processing, (3) text representations generation (4) baseline models involving MLs and DLs, and (5) our stacking ensemble strategy using new hybrid text representation.

set to the sklearn default (i.e., no tuning, except for the random state of the non-deterministic models that was fixed). In the last stage, (5) we extracted the probability output from the first dense layer of WangchanBERTa using 50D as a DL feature. We then concatenated the extracted probability output with the probability outputs generated from ML models across distinct text representations as a new hybrid text representation/feature, which can be defined as (8).

$$hybridFeat = \left[f_{50D}\left(Pr\right), f_{word2vec}\left(Pr\right), f_{tfid1}\left(Pr\right), f_{bow1}\left(Pr\right)\right] \quad (8)$$

where Pr represents the predicted probability. Finally, the meta-learner model that was performed a final prediction, was chosen based on the best performer from the pool of the eleven ML-based models. At this phase, for the sake of simplicity, all the meta classifiers were tuned with the same hyperparameter range that is shown in Table 3.

The experiments were written in Python and executed on a system with an Intel i9-10900k CPU @3.7GHz, RTX 2080Ti, and 32GB of RAM. In this study, ML-based models were built with scikit-learn version 1.0.2 [58] The DL-based models were implemented using TensorFlow version 2.6 [59] and PyTorch version 1.12 [60].

D. RESULTS AND DISCUSSION

This section explains an analysis of the results in detail.

1) COMPARISON OF ALL MACHINE-LEARNING BASELINE MODELS

To select the base learners for our stacking ensemble strategy, the performance of SETAR was compared to that of the baseline ML-based models for the TT, TX, WS, and KT datasets. Fig. 2 provides a summary of the top 15 baseline models for each dataset's training and independent test sets, ranked by F1 averaged from 10-repeated hold-out on training set. Table 4 summarizes the best performer of each dataset.

TABLE 4. The performance comparison of best performing ML classifiers on the training and test dataset.

Evaluation strategy	Dataset	Method	ACC	Pre	Rec	F1
10-repeated holdout	TT	ET-BOW1	0.648	0.666	0.582	0.621
	TX	LR-TF12	0.699	0.693	0.716	0.704
	WS	LGBM-TF12	0.705	0.630	0.512	0.563
	KT	SVM-W2VTF	0.754	0.723	0.612	0.663
Testing set	TT	ET-BOW1	0.657	0.669	0.600	0.632
	TX	LR-TF12	0.684	0.681	0.693	0.686
	WS	LGBM-TF12	0.710	0.686	0.516	0.564
	KT	SVM-W2VTF	0.760	0.732	0.621	0.672

With regard to results, on the TT dataset, the results indicate that ET-BOW1 (i.e., ET using BOW with unigram sentiment features) produced the best performance in terms of (ACC=0.648, F1=0.621). For the TX dataset, the best performance was LR with TF-IDF12 (both unigram and bi-grams sentiment feature), revealing a performance in terms of (ACC=0.699, F1=0.704), while LGBM using TF-IDF12 shows the highest performance in terms of (ACC=0.705, F1=0.563) for the WS dataset. SVM with Wor2Vec performed the best on the KT dataset, in terms of (ACC=0.754, F1=0.663) for testing sets. The results revealed that ET, LR, LGBM, SVM, MLP, and PLS algorithms demonstrate strong performances across four datasets. Followed by RF, which

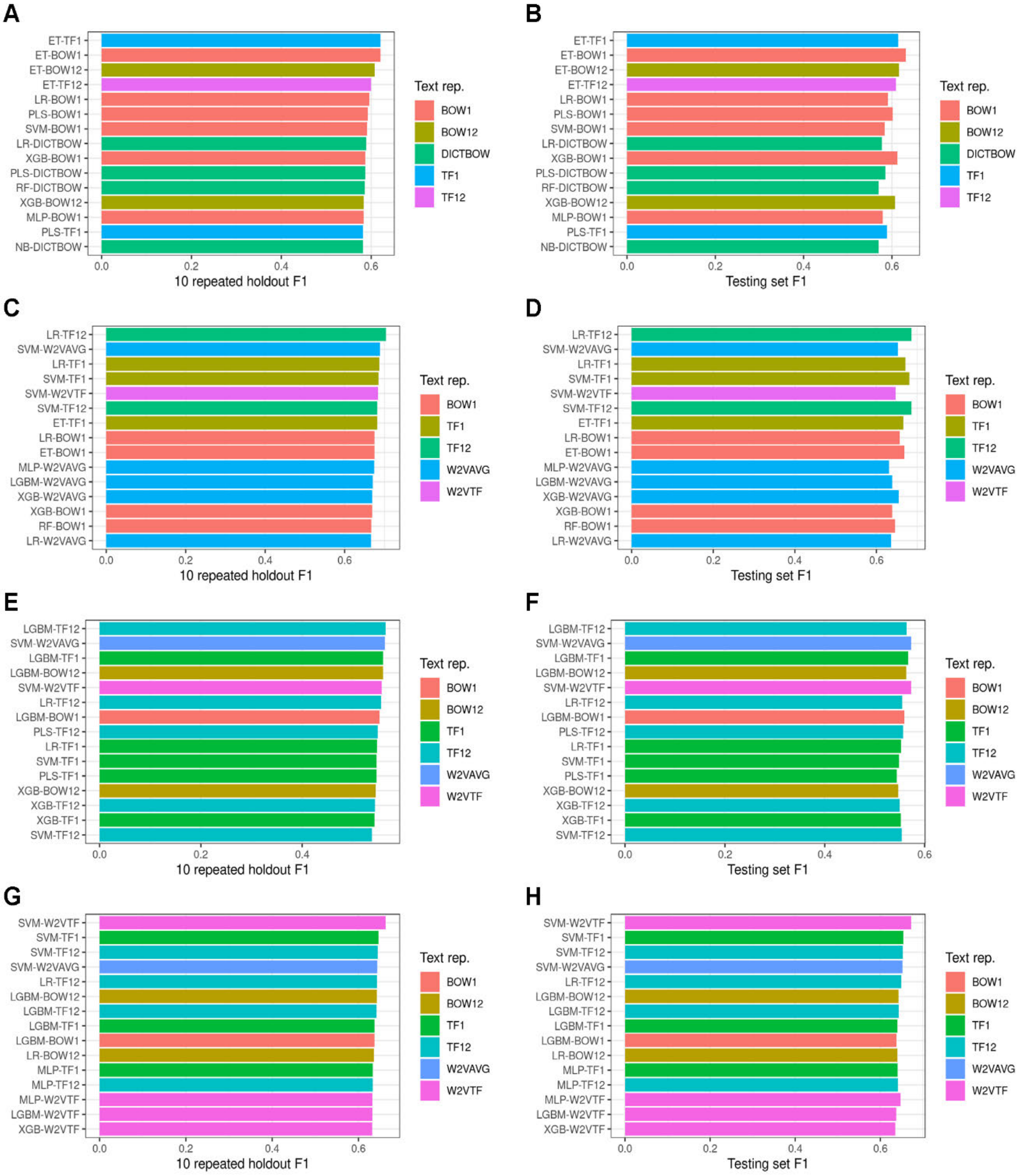

FIGURE 2. The performance evaluation of the top 15 base-line ML models of each dataset comprised of the TT (A, B), TX (C, D), WS (E, F), and KT (G, H) datasets. Where, A, C, E, and G represent the 10-repeated hold-out F1 of the top 15 base-line models, while B, D, F, and H represent the testing set F1 of the top 15 base-line models.

was also ranked in the top 15, appeared in the TT and TX datasets. Hence, these seven ML algorithms were chosen as base learners for the SETAR construction. Additionally, we observed that ML algorithms such as DT and 1NN, as well as text representations such as POS-Tagging and DICT-based, were not in the top 15 due to their poor performance.

2) COMPARISON BETWEEN SETAR AND ALL STATE-OF-THE ART MODELS

To exhibit the leverage of the proposed stacking ensemble approach, we compared the performance of SETAR to that of the proposed baseline ML and DL models. The top five baseline ML and DL models are plotted against SETAR performance in Fig. 3. As shown in Fig. 3 and Table 5, SETAR with different meta-learners outperformed the top five baseline models on 10-repeated holdout and testing datasets in terms of most of the classification evaluation metrics for all datasets. Regarding the 10-repeated holdout results, SETAR-LR outperformed the best-performing baseline model (WangchanBERTa) for the TT dataset in terms of (ACC=0.782 vs. ACC=0.758, F1=0.773 vs. F1=0.755), and it improved ACC by 2.4% and F1 by 1.8%. The best

TABLE 5. The comparison of SETAR and all state-of-the art models' training and testing set performance. Bold indicates the best performers.

Evaluation strategy	Dataset	Method	ACC	Pre	Rec	F1
10-repeated holdout	TT	SETAR-LR	**0.782**	**0.782**	**0.765**	**0.773**
		WangchanBERTa	0.758	0.755	0.756	0.755
		BERT	0.645	0.640	0.603	0.621
		ET-BOW1	0.648	0.666	0.582	0.621
		ET-TF1	0.644	0.690	0.566	0.621
		ET-BOW12	0.634	0.685	0.548	0.608
	TX	SETAR-LR	**0.775**	**0.776**	0.775	**0.776**
		WangchanBERTa	0.753	0.735	**0.796**	0.761
		LR-TF12	0.699	0.693	0.716	0.704
		SVM-W2VAVG	0.666	0.645	0.743	0.689
		LR-TF1	0.695	0.707	0.671	0.687
		SVM-TF1	0.683	0.682	0.689	0.685
	WS	SETAR-PLS	**0.745**	**0.673**	**0.601**	**0.634**
		WangchanBERTa	0.734	0.674	0.578	0.622
		LGBM-TF12	0.705	0.630	0.512	0.565
		SVM-W2VAVG	0.701	0.664	0.490	0.563
		LGBM-BOW12	0.704	0.631	0.504	0.560
		LGBM-TF1	0.703	0.629	0.505	0.560
	KT	SETAR-XGB	**0.831**	**0.802**	**0.780**	**0.791**
		WangchanBERTa	0.815	0.786	0.759	0.771
		BERT	0.767	0.726	0.679	0.701
		BiLSTM	0.764	0.716	0.682	0.699
		CNN	0.757	0.714	0.644	0.677
		SVM-W2VTF	0.754	0.723	0.612	0.663
Testing set	TT	SETAR-LR	**0.775**	**0.773**	0.761	**0.766**
		WangchanBERTa	0.763	0.760	**0.764**	0.762
		BERT	0.631	0.626	0.598	0.611
		ET-BOW1	0.657	0.669	0.600	0.632
		ET-TF1	0.640	0.668	0.568	0.614
		ET-BOW12	0.641	0.672	0.569	0.616
	TX	SETAR-LR	**0.765**	**0.767**	0.765	**0.766**
		WangchanBERTa	0.743	0.726	**0.792**	0.754
		LR-TF12	0.684	0.681	0.693	0.686
		SVM-W2VAVG	0.635	0.623	0.686	0.653
		LR-TF1	0.683	0.698	0.649	0.671
		SVM-TF1	0.683	0.685	0.679	0.681
	WS	SETAR-PLS	**0.748**	**0.678**	**0.597**	**0.635**
		WangchanBERTa	0.734	0.679	0.577	0.623
		LGBM-TF12	0.710	0.621	0.516	0.564
		SVM-W2VAVG	0.703	0.686	0.493	0.573
		LGBM-BOW12	0.709	0.630	0.508	0.563
		LGBM-TF1	0.712	0.624	0.520	0.567
	KT	SETAR-XGB	**0.831**	**0.805**	**0.775**	**0.790**
		WangchanBERTa	0.814	0.787	0.756	0.770
		BERT	0.768	0.727	0.678	0.701
		BiLSTM	0.765	0.717	0.683	0.700
		CNN	0.756	0.713	0.642	0.676
		SVM-W2VTF	0.760	0.732	0.621	0.672

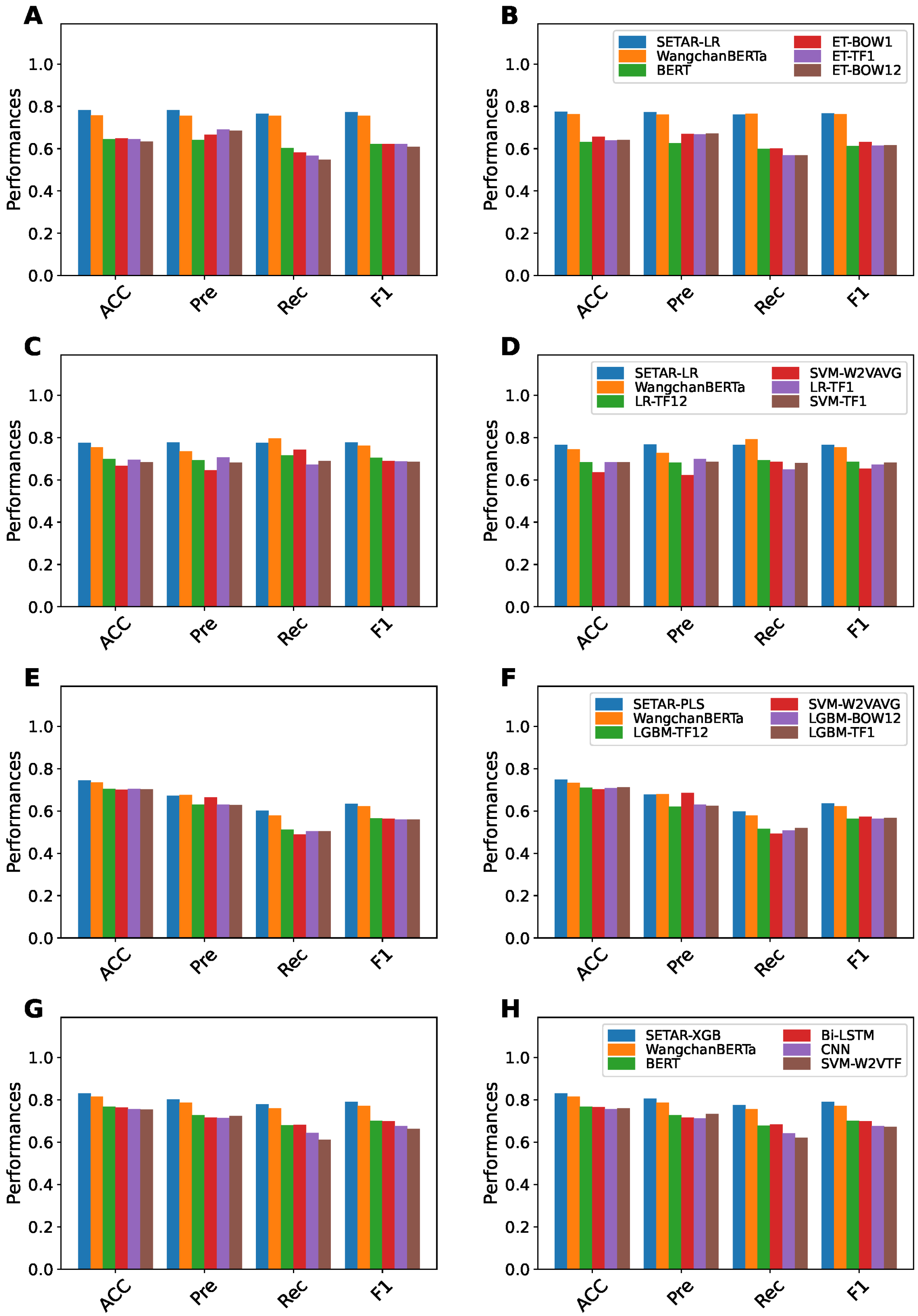

FIGURE 3. The predictive performance of all of the top 5 models was determined and compared to our proposed model (SETAR) in terms of accuracy, precision, recall, and F1score. A and B represent TT dataset, C and D represent TX dataset, E and G represent WS dataset, and G and H represent KT dataset.

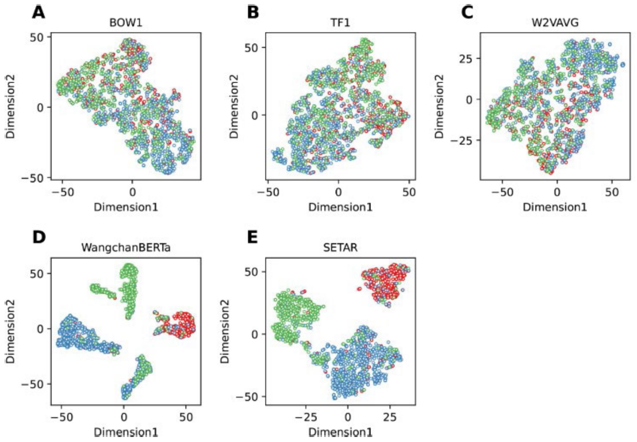

FIGURE 4. t-distributed stochastic neighbor embedding (t-SNE) distribution of positive, neutral, and negative sentiment of TT dataset (3 classes).

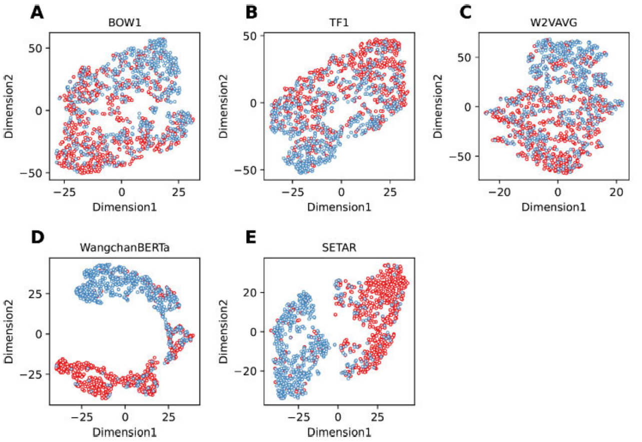

FIGURE 5. Accordingly, t-distributed stochastic neighbor embedding (t-SNE) for distribution of positive and negative sentiment of the TX dataset (2 classes).

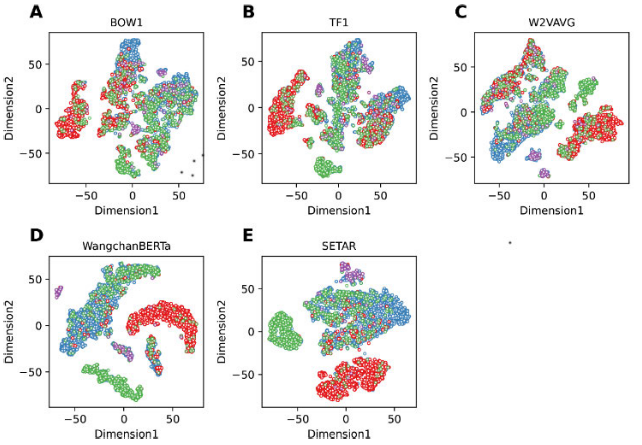

FIGURE 6. Accordingly, t-distributed stochastic neighbor embedding (t-SNE) for distribution of positive, neutral, and negative sentiment of the WS dataset (4 classes).

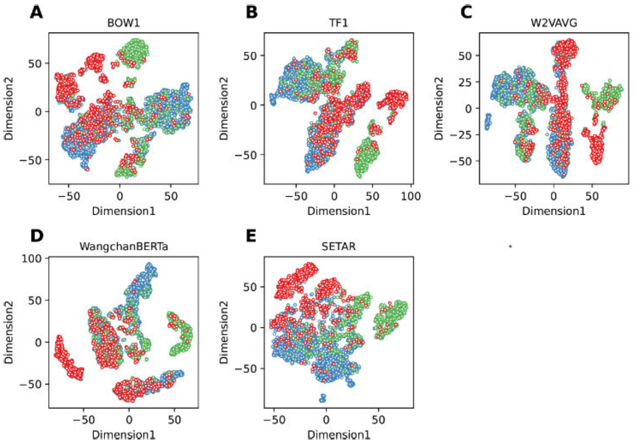

FIGURE 7. Accordingly, t-distributed stochastic neighbor embedding (t-SNE) for distribution of positive, neutral, and negative sentiment of the KT dataset (3 classes).

performance yielded by SETAR-LR was on the TX dataset, (ACC=0.775, F1=0.776) improved ACC by 2.2% and F1 by 1.5% in comparison to the best-performing baseline model. For larger-sized datasets such as WS and KT datasets, SETAR-PLS demonstrated the best performance for the WS dataset (ACC=0.745, F1=0.634), and it improved ACC by 1.1% and F1 by 1.2%. SETAR-XGB achieved the best performance on the KT dataset (ACC=0.831, F1=0.791) and improved ACC by 1.6% and F1 by 2% compared to the best-performing baseline model. In addition, we found that DL models such as BiLSTM and CNN performed well on large datasets such as KT dataset.

With regards to testing set results, a similar level of performance was produced. SETAR-LR achieved the best performance on the TT dataset (ACC=0.775, F1=0.763) with ACC improved by 1.2% and F1 by 0.4%. For TX dataset, SETAR-LR produced the highest performance in terms of (ACC=0.765, F1=0.766), improved ACC by 2.2% and F1 by 1.2%. For the WS dataset, SETAR-PLS produced the highest performance in terms of (ACC=0.748, F1=0.635), improved ACC by 1.4% and F1 by 1.2%. SETAR-XGB achieved the highest performance on the KT dataset (ACC=0.831, F1=0.79), increasing ACC by 1.7% and F1 by 2%. These results indicated that SETAR improved the performance and consistency of both evaluation strategies.

3) INTERPRETATION OF SETAR MODEL

To further exemplify the advantages of our proposed model, we also compared it to the top four text representations, including BOW1, TF-IDF1, Word2Vec, and Wangchan-BERTa. T-SNE [61] was plotted in Figs 4-7 to illustrate the classified results of high dimension pace on a 2D space scale. Due to the large number of samples from the testing dataset that must be displayed in the plot for the KT dataset, 10% of the samples are chosen at random for display. As shown in Figs. 4-7(A-D), the number of colored dots in the same cluster is greater than that of the SETAR model (Figs. 4-7(E)). This demonstrated that among the state-of-the-art models, SETAR produced the

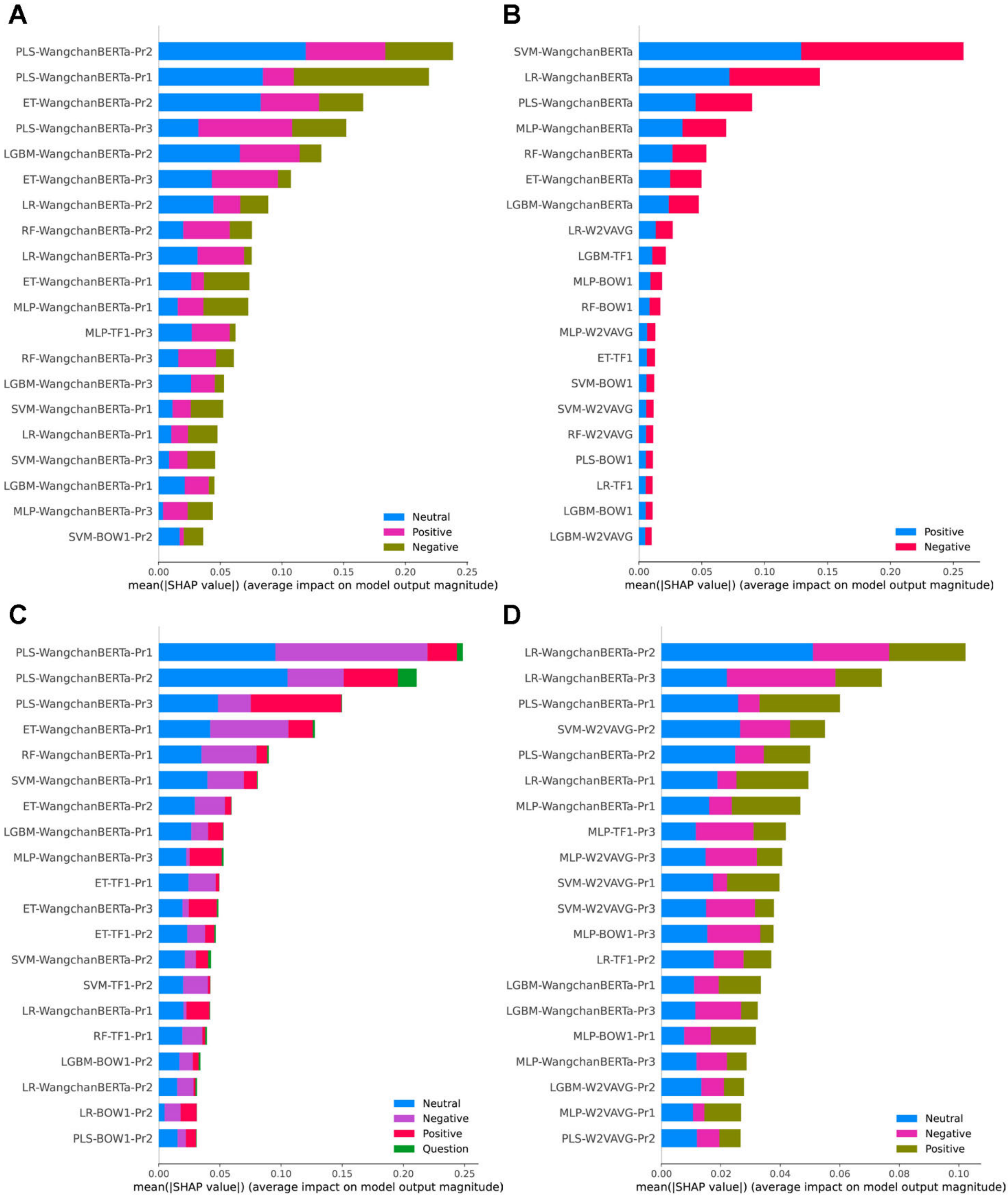

FIGURE 8. Twenty important features used in SETAR construction ranked by SHAP values with four distinct text representations. For multi-class problem, the suffix Pr represents the probability output of that class target. A, B, C, D represents the TT, TX, WS, and KT dataset, respectively.

highest level of discriminative power for identifying sentence sentiment polarity. In addition, by observing the number of clusters generated by each dataset, the results confirmed that SETAR accurately classified the sentence across all four datasets.

The feature important values of these base-learners need to be seen, in order to explain the proposed model's output, including the influence of each feature on sentiment polarity identification of each dataset. SHAP (Shapley Additive Explanations) [62], is a visualization technique that uses game-theoretic method to describe an output of ML models. Fig. 8 depicts SHAP values generated from the SHAP summary plot of the impact of the top twenty features of each dataset. As illustrated, the five most informative features with the highest SHAP values were identified by WangchanBERTa features for all datasets, except the KT dataset, where Word2Vec with average embedding was in the third rank. For the TX dataset, since it is a binary classification problem, therefore, we only require one feature vector. For the WS dataset, Pr4 was not among the top 20 features since the number of class questions was insufficient. Fig. 8 also revealed that PLS, SVM, and LR with WangchanBERTa text representation were the top three base learners across all datasets. Therefore, we can summarize that WangchanBERTa feature is currently the finest feature for usage in Thai sentiment classification.

E. CONCLUSION

SETAR, our proposed model, used stacking ensemble learning strategy by combining machine-learning and deep-learning features as a novel hybrid text representation. The base-learners were built using seven divergent complex machine-learning models: SVM, RF, MLP, LGBM, ET, PLS, and LR. They were constructed using four different text representations, included Word2Vec, TF-IDF1, and BOW1 with the extracted feature from the pre-trained Thai language model, WangchanBERTa. We explored the eleven ML-based models in terms of performance and established a final meta-learner. The experiment results revealed that LR, PLS, and XGB were the top performers dependent on the dataset. Our stacking ensemble strategy exceeded the baseline of the pre-trained language model in terms of the predictive performance of all the evaluation metrics for all benchmark datasets. Moreover, T-SNE and SHAP visualization techniques were employed to explain the importance of features that impact the SETAR model performance. In this study, we also provided an extensive and comprehensive comparison of the classification performance of eleven ML and four DL algorithms across five well-known text representations, and four domains on social media and written book. The experiment's results demonstrated the efficacy and robustness of our proposed model for a scarce-resource language such as Thai language in the sentiment classification task. Lastly, our developed Thai sentiment analysis corpus also proved the scalability and effectiveness of our proposed stacking ensemble strategy.

Future research should investigate the relevant ensemble learning methods, such as the application of the bagging or boosting mechanism to both homogeneous and heterogeneous language models. For example, researchers should attempt to stack with more diverse ensemble members, perhaps by incorporating deep learning models that include CNN, BiLSTM, and GRU as the base-learners, as well as meta-learners. Another beneficial direction of future research would be to handle imbalanced class that usually presents itself in social media data, as well as to investigate the interpretability of the model. We anticipate that our proposed SETAR will benefit NLP researchers and provide a free social listening tool to any small-medium business enterprises that engage in Thai sentiment analysis.

ACKNOWLEDGMENT

This work was supported by the Information Technology Service Center (ITSC) of Chiang Mai University for computing resources.

REFERENCES

[1] M. Hajiali, "Big data and sentiment analysis: A comprehensive and systematic literature review," *Concurrency Comput., Pract. Exper.*, vol. 32, no. 14, p. e5671, Jul. 2020, doi: 10.1002/cpe.5671.

[2] X. Fang and J. Zhan, "Sentiment analysis using product review data," *J. Big Data*, vol. 2, no. 1, p. 5, Jun. 2015, doi: 10.1186/s40537-015-0015-2.

[3] L. Yue, W. Chen, X. Li, W. Zuo, and M. Yin, "A survey of sentiment analysis in social media," *Knowl. Inf. Syst.*, vol. 60, no. 2, pp. 617–663, Aug. 2019, doi: 10.1007/s10115-018-1236-4.

[4] M. Z. Ansari, M. B. Aziz, M. O. Siddiqui, H. Mehra, and K. P. Singh, "Analysis of political sentiment orientations on Twitter," *Proc. Comput. Sci.*, vol. 167, pp. 1821–1828, Jan. 2020, doi: 10.1016/j.procs.2020.03.201.

[5] M. Hämäläinen, P. Patpong, K. Alnajjar, N. Partanen, and J. Rueter, "Detecting depression in Thai blog posts: A dataset and a baseline," in *Proc. 7th Workshop Noisy User-Generated Text (W-NUT)*, 2021, pp. 20–25, doi: 10.18653/v1/2021.wnut-1.3.

[6] M. R. Islam, M. A. Kabir, A. Ahmed, A. R. M. Kamal, H. Wang, and A. Ulhaq, "Depression detection from social network data using machine learning techniques," *Health Inf. Sci. Syst.*, vol. 6, no. 1, p. 8, Aug. 2018, doi: 10.1007/s13755-018-0046-0.

[7] J. Kazmaier and J. H. van Vuuren, "A generic framework for sentiment analysis: Leveraging opinion-bearing data to inform decision making," *Decis. Support Syst.*, vol. 135, Aug. 2020, Art. no. 113304, doi: 10.1016/j.dss.2020.113304.

[8] N. C. Dang, M. N. Moreno-García, and F. De la Prieta, "Sentiment analysis based on deep learning: A comparative study," *Electronics*, vol. 9, no. 3, p. 483, Mar. 2020, doi: 10.3390/electronics9030483.

[9] M. Hoang, O. A. Bihorac, and J. Rouces, "Aspect-based sentiment analysis using BERT," in *Proc. 22nd Nordic Conf. Comput. Linguistics*. Turku, Finland: Linköping Univ. Electronic Press, Sep. 2019, pp. 187–196. Accessed: Aug. 27, 2022. [Online]. Available: https://aclanthology.org/W19-6120

[10] T. G. Dietterich, "Ensemble methods in machine learning," in *Multiple Classifier Systems* (Lecture Notes in Computer Science). Berlin, Germany: Springer, 2000, pp. 1–15, doi: 10.1007/3-540-45014-9_1.

[11] J. Briskilal and C. N. Subalalitha, "An ensemble model for classifying idioms and literal texts using BERT and RoBERTa," *Inf. Process. Manage.*, vol. 59, no. 1, Jan. 2022, Art. no. 102756, doi: 10.1016/j.ipm.2021.102756.

[12] K. Pasupa and T. S. Na Ayutthaya, ''Hybrid deep learning models for Thai sentiment analysis,'' *Cognit. Comput.*, vol. 14, no. 1, pp. 167–193, Jan. 2022, doi: 10.1007/s12559-020-09770-0.

[13] G. Wang, J. Sun, J. Ma, K. Xu, and J. Gu, ''Sentiment classification: The contribution of ensemble learning,'' *Decis. Support Syst.*, vol. 57, pp. 77–93, Jan. 2014, doi: 10.1016/j.dss.2013.08.002.

[14] J. Kazmaier and J. H. van Vuuren, ''The power of ensemble learning in sentiment analysis,'' *Expert Syst. Appl.*, vol. 187, Jan. 2022, Art. no. 115819, doi: 10.1016/j.eswa.2021.115819.

[15] P. Chormai and ekapolc, ''PyThaiNLP/wisesight-sentiment: First release,'' Zenodo, Sep. 22, 2019, doi: 10.5281/zenodo.3457447.

[16] S. Sirihattasak, M. Komachi, H. Ishikawa, and H. Ishikawa, ''Annotation and classification of toxicity for Thai Twitter,'' in *Proc. 2nd Workshop Text Anal. Cybersecurity and Online Saf. (TA-COS)*, 2018, p. 1.

[17] S. Chotirat and P. Meesad, ''Part-of-speech tagging enhancement to natural language processing for Thai wh-question classification with deep learning,'' *Heliyon*, vol. 7, no. 10, Oct. 2021, Art. no. e08216, doi: 10.1016/j.heliyon.2021.e08216.

[18] N. Khamphakdee and P. Seresangtakul, ''Sentiment analysis for Thai language in hotel domain using machine learning algorithms,'' *Acta Inf. Pragensia*, vol. 10, no. 2, pp. 155–171, Sep. 2021.

[19] D. Dangi, D. K. Dixit, and A. Bhagat, ''Sentiment analysis of COVID-19 social media data through machine learning,'' *Multimedia Tools Appl.*, vol. 81, no. 29, pp. 42261–42283, Jul. 2022, doi: 10.1007/s11042-022-13492-w.

[20] P. Cen, K. Zhang, and D. Zheng, ''Sentiment analysis using deep learning approach,'' *J. Artif. Intell.*, vol. 2, no. 1, pp. 17–27, 2020, doi: 10.32604/jai.2020.010132.

[21] Y. Zhang and B. Wallace, ''A sensitivity analysis of (and practitioners' guide to) convolutional neural networks for sentence classification,'' Apr. 2015, *arXiv:1510.03820*, doi: 10.48550/arXiv.1510.03820.

[22] A. Krizhevsky, I. Sutskever, and G. E. Hinton, ''ImageNet classification with deep convolutional neural networks,'' in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 25, 2012, pp. 1–9. Accessed: Jun. 15, 2021. [Online]. Available: https://papers.nips.cc/paper/2012/hash/c399862d3b9d6b76c8436e924a68c45b-Abstract.html

[23] S. Hochreiter and J. Schmidhuber, ''Long short-term memory,'' *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997, doi: 10.1162/neco.1997.9.8.1735.

[24] C. Chen, R. Zhuo, and J. Ren, ''Gated recurrent neural network with sentimental relations for sentiment classification,'' *Inf. Sci.*, vol. 502, pp. 268–278, Oct. 2019, doi: 10.1016/j.ins.2019.06.050.

[25] B. Jang, M. Kim, G. Harerimana, S.-U. Kang, and J. W. Kim, ''Bi-LSTM model to increase accuracy in text classification: Combining word2vec CNN and attention mechanism,'' *Appl. Sci.*, vol. 10, no. 17, p. 5841, Aug. 2020, doi: 10.3390/app10175841.

[26] S. Sivakumar and R. Rajalakshmi, ''Self-attention based sentiment analysis with effective embedding techniques,'' *Int. J. Comput. Appl. Technol.*, vol. 65, no. 1, pp. 65–77, Jan. 2021, doi: 10.1504/IJCAT.2021.113651.

[27] L.-C. Yu, J. Wang, K. R. Lai, and X. Zhang, ''Refining word embeddings for sentiment analysis,'' in *Proc. Conf. Empirical Methods Natural Lang. Process.* Copenhagen, Denmark: Association for Computational Linguistics, Sep. 2017, pp. 534–539, doi: 10.18653/v1/D17-1056.

[28] R. Rajalakshmi, S. Selvaraj, R. F. Mattins, P. Vasudevan, and M. A. Kumar, ''HOTTEST: Hate and offensive content identification in Tamil using transformers and enhanced STemming,'' *Comput. Speech Lang.*, vol. 78, Mar. 2023, Art. no. 101464, doi: 10.1016/j.csl.2022.101464.

[29] D. Wang, J. Su, and H. Yu, ''Feature extraction and analysis of natural language processing for deep learning English language,'' *IEEE Access*, vol. 8, pp. 46335–46345, 2020, doi: 10.1109/ACCESS.2020.2974101.

[30] Z.-X. Liu, D.-G. Zhang, G.-Z. Luo, M. Lian, and B. Liu, ''A new method of emotional analysis based on CNN–BiLSTM hybrid neural network,'' *Cluster Comput.*, vol. 23, no. 4, pp. 2901–2913, Dec. 2020, doi: 10.1007/s10586-020-03055-9.

[31] A. Pimpalkar and R. J. R. Raj, ''MBiLSTMGloVe: Embedding GloVe knowledge into the corpus using multi-layer BiLSTM deep learning model for social media sentiment analysis,'' *Expert Syst. Appl.*, vol. 203, Oct. 2022, Art. no. 117581, doi: 10.1016/j.eswa.2022.117581.

[32] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, ''BERT: Pre-training of deep bidirectional transformers for language understanding,'' May 2019, *arXiv:1810.04805*, doi: 10.48550/arXiv.1810.04805.

[33] L. Khan, A. Amjad, N. Ashraf, and H.-T. Chang, ''Multi-class sentiment analysis of Urdu text using multilingual BERT,'' *Sci. Rep.*, vol. 12, no. 1, Mar. 2022, Art. no. 1, doi: 10.1038/s41598-022-09381-9.

[34] 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,'' Jul. 2019, *arXiv:1907.11692*, doi: 10.48550/arXiv.1907.11692.

[35] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever, ''Language models are unsupervised multitask learners,'' *OpenAI Blog*, vol. 1, no. 8, p. 9, 2019.

[36] L. Breiman, ''Bagging predictors,'' *Mach. Learn.*, vol. 24, no. 2, pp. 123–140, Aug. 1996, doi: 10.1007/BF00058655.

[37] D. Opitz and R. Maclin, ''Popular ensemble methods: An empirical study,'' *J. Artif. Intell. Res.*, vol. 11, pp. 169–198, Aug. 1999, doi: 10.1613/jair.614.

[38] D. H. Wolpert, ''Stacked generalization,'' *Neural Netw.*, vol. 5, no. 2, pp. 241–259, Jan. 1992.

[39] H. Saleh, S. Mostafa, A. Alharbi, S. El-Sappagh, and T. Alkhalifah, ''Heterogeneous ensemble deep learning model for enhanced Arabic sentiment analysis,'' *Sensors*, vol. 22, no. 10, p. 3707, May 2022, doi: 10.3390/s22103707.

[40] R. Tesmuang and N. Chirawichitchai, ''Sentiment analysis of Thai online product reviews using genetic algorithms with support vector machine,'' *Prog. Appl. Sci. Technol.*, vol. 10, no. 2, Nov. 2020, Art. no. 2, doi: 10.14456/past.2020.8.

[41] N. Goyal, J. Du, M. Ott, G. Anantharaman, and A. Conneau, ''Larger-scale transformers for multilingual masked language modeling,'' May 2021, *arXiv:2105.00572*, doi: 10.48550/arXiv.2105.00572.

[42] L. Lowphansirikul, C. Polpanumas, N. Jantrakulchai, and S. Nutanong, ''WangchanBERTa: Pretraining transformer-based Thai language models,'' Mar. 2021, *arXiv:2101.09635*, doi: 10.48550/arXiv.2101.09635.

[43] T. Kudo and J. Richardson, ''SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing,'' Aug. 2018, *arXiv:1808.06226*, doi: 10.48550/arXiv.1808.06226.

[44] K. Pasupa, P. Netisopakul, and R. Lertsuksakda, ''Sentiment analysis of Thai children stories,'' *Artif. Life Robot.*, vol. 21, no. 3, pp. 357–364, Sep. 2016, doi: 10.1007/s10015-016-0283-8.

[45] S. Sirihattasak, M. Komachi, and H. Ishikawa. (2018). *Annotation and Classification of Toxicity for Thai Twitter*. Accessed: Jul. 17, 2022. [Online]. Available: https://www.semanticscholar.org/paper/Annotation-and-Classification-of-Toxicity-for-Thai-Sirihattasak-Komachi/5a6d84a00037b76cc8c30bdd5ad6dd0ffd73d201

[46] E. Partalidou, E. Spyromitros-Xioufis, S. Doropoulos, S. Vologiannidis, and K. I. Diamantaras, ''Design and implementation of an open source Greek POS tagger and entity recognizer using spaCy,'' Dec. 2019, *arXiv:1912.10162*, doi: 10.48550/arXiv.1912.10162.

[47] P. Thiengburanathum, ''A comparison of Thai sentence boundary detection approaches using online product review data,'' in *Advances in Networked-Based Information Systems* (Advances in Intelligent Systems and Computing), L. Barolli, K. F. Li, T. Enokido, and M. Takizawa, Eds. Cham, Switzerland: Springer, 2021, pp. 405–412, doi: 10.1007/978-3-030-57811-4_40.

[48] W. Phatthiyaphaibun, K. Chaovavanich, C. Polpanumas, A. Suriyawongkul, L. Lowphansirikul, and P. Chormai, ''PyThaiNLP: Thai natural language processing in Python,'' Zenodo, Jun. 2016, doi: 10.5281/zenodo.3519354.

[49] Y. Zhang, R. Jin, and Z.-H. Zhou, ''Understanding bag-of-words model: A statistical framework,'' *Int. J. Mach. Learn. Cybern.*, vol. 1, nos. 1–4, pp. 43–52, Dec. 2010, doi: 10.1007/s13042-010-0001-0.

[50] T. Mikolov, K. Chen, G. Corrado, and J. Dean, ''Efficient estimation of word representations in vector space,'' Sep. 2013, *arXiv:1301.3781*. Accessed: Nov. 22, 2020.

[51] V. Sornlertlamvanich, N. Takahashi, and H. Isahara. (1998). *Thai Part-of-Speech Tagged Corpus: ORCHID*. Accessed: Jul. 19, 2022. [Online]. Available: https://www.semanticscholar.org/paper/Thai-Part-of-speech-Tagged-Corpus%3A-ORCHID-Sornlertlamvanich-Takahashi/b2cd2c2b07285f114244a886d02b5b7cb69a203e

[52] D. R. Rice and C. Zorn, "Corpus-based dictionaries for sentiment analysis of specialized vocabularies," *Political Sci. Res. Methods*, vol. 9, no. 1, pp. 20–35, Jan. 2021, doi: 10.1017/psrm.2019.10.

[53] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," Oct. 2013, *arXiv:1310.4546*, doi: 10.48550/arXiv.1310.4546.

[54] N. Kalchbrenner, E. Grefenstette, and P. Blunsom, "A convolutional neural network for modelling sentences," Apr. 2014, *arXiv:1404.2188*, doi: 10.48550/arXiv.1404.2188.

[55] M. D. Zeiler, "ADADELTA: An adaptive learning rate method," Dec. 2012, *arXiv:1212.5701*, doi: 10.48550/arXiv.1212.5701.

[56] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," Jan. 2017, *arXiv:1412.6980*, doi: 10.48550/arXiv.1412.6980.

[57] J. Snoek, H. Larochelle, and R. P. Adams, "Practical Bayesian optimization of machine learning algorithms," Aug. 2012, *arXiv:1206.2944*, doi: 10.48550/arXiv.1206.2944.

[58] F. Pedregosa et al., "Scikit-learn: Machine learning in Python," Jun. 2018, *arXiv:1201.0490*, doi: 10.48550/arXiv.1201.0490.

[59] M. Abadi et al., "TensorFlow: Large-scale machine learning on heterogeneous distributed systems," Mar. 2016, *arXiv:1603.04467*, doi: 10.48550/arXiv.1603.04467.

[60] A. Paszke et al., "PyTorch: An imperative style, high-performance deep learning library," Dec. 2019, *arXiv:1912.01703*, doi: 10.48550/arXiv.1912.01703.

[61] L. van der Maaten and G. Hinton, "Visualizing data using t-SNE," *J. Mach. Learn. Res.*, vol. 9, no. 86, pp. 2579–2605, 2008.

[62] S. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," Nov. 2017, *arXiv:1705.07874*. Accessed: Jan. 30, 2021.

PREE THIENGBURANATHUM received the B.Sc. degree in computer science from Colorado State University, USA, the M.Sc. degree in computer science from the University of Colorado at Denver, USA, and the Ph.D. degree in computing informatics from the Faculty of Science and Technology, Bournemouth University, U.K. He is currently a full-time Assistant Professor with the Software Engineering Department, Chiang Mai University, Thailand. His research interests include machine learning, recommendation systems, and artificial intelligence.

PHASIT CHAROENKWAN received the bachelor's and master's degrees in computer science from Chiang Mai University, Chiang Mai, Thailand, and the Ph.D. degree in bioinformatics from the National Yang Ming Chiao Tung University, Taiwan. He was formally a full-time Lecturer in computer science and digital communications with Mae Fah Luang University and Maejo University, respectively. He is currently an Assistant Professor and a full-time Professor of modern management and information technology with the College of Arts, Media and Technology, Chiang Mai University. He was also a Bioinformatics Researcher in Taiwan. His works are mainly focused on bioinformatics and biomedical informatics, especially QSAR. However, he also assists in the use of machine learning approaches for business improvement.