ICIC Express Letters
Volume **17**, Number **3**, March **2023**
 ISSN 1881-803X
pp. **315–326**

USING DEEP LEARNING MODEL WITH MULTIPLE INPUTS FOR THAI DEFAMATORY TEXT CLASSIFICATION ON PUBLIC FACEBOOK COMMENTS

Patipan Watjanapron and Orawan Chaowalit*

Center of Excellence in AI and NLP
Department of Computing
Faculty of Science
Silpakorn University
6, Rajamankha Nai Road, Muang District, Nakhon Pathom 73000, Thailand
watjanapron_p@silpakorn.edu; *Corresponding author: chaowalit_o@su.ac.th

Received June 2022; accepted August 2022

Abstract. *This research aims to classify Thai defamatory messages or sentences on Facebook, where the learned text is derived from the comments under the pictures or under the articles of the person being mentioned. It indicates whether the text is within the scope of defamation or non-defamation by using the text and special extracted features of the content as input data. We present the model creation to classify defamatory statements on Facebook by using three deep learning techniques: Long Short-Term Memory (LSTM), Bidirectional Long Short-Term Memory (Bi-LSTM), and Convolutional Neural Network (CNN). Results show that CNN using Thai2fit for word embedding that combines two key feature inputs – term frequency of dictionary judgment related to seven types of defamation and Part-of-Speech (POS) tag provides the best result. We observe that CNN integrated with the presented features is more effective than LSTM and Bi-LSTM, which are set up with the same input. This research focuses on defamatory statements that comprise unique legal characteristics. Our results relate to the study conducted by Wenpeng Yin et al. who comparatively experimented the performances of CNN and RNN models with each domain text type. Our results suggest that the efficiency of a model depends on the nature of the data. We set up a simple model for our research in which tuning model parameters can result in improved efficiency.*
Keywords: Defamatory, Deep learning, Text classification, Social media, Machine learning, Convolutional neural network, Judgment

1. **Introduction.** Regarding the current communication methods of humans, apart from face-to-face conversation, one of the most essential communication methods is social networking. Many Social Media Platforms (SMPs) are available, such as Facebook, Twitter, Instagram, and TikTok. Each has different selling points. Instagram and TikTok are SMPs that focus on disseminating user stories mainly via pictures and videos. Likewise, Facebook and Twitter are SMPs with large numbers of users and can distribute contents flexibly, such as texts, articles, images, and videos. For all these platforms, users can communicate through their comments; for example, comments on Facebook posts or replies on Twitter posts. However, many negative comments cause disturbance to others, whether in terms of false or distorted statements. These negative comments may impact people, organizations or any business and are thus considered cyber abuses.

Cyber abuses are user behaviors on SMPs that hurt others in society, whether to cause harassment or embarrassment to others. They can be in many forms, such as hate speech, racism, sexism, sarcasm, bullying, trolling or profanity, and lead to physical or mental damage. Cyber abuses can cause suicidal thoughts or even suicidal behaviors in victims.

DOI: 10.24507/icicel.17.03.315

Suicidal behaviors and thoughts have been studied in children and young people, with causes and effects from cyberbullying (CB) [1]. Cyber abuse is one of the most serious problems of the world. In many countries, laws protecting rights, freedoms, and human dignity have been adopted to control the behaviors of the members of societies, including their comments on social media. Therefore, when anyone injures others on social media, they will be considered guilty and punished to prevent cyber abuses that lead to self-harm.

Thailand has laws that protect cyber abuse victims from defamation on social media, according to the Criminal Code. Significant sections in this area are as follows: Section 326: Defamation, Section 328: Defamation by Publication, and Section 393: Public Insults [2].

Such laws can be applied to preventing cyber abuses. However, all statements may not comply with defamation laws and cannot be prosecuted, even some are vulgar words. As a result, cyber abuse victims rely on legal experts to advise whether they should sue or prosecute.

For this reason, we recognize the limitations and importance of cyber abuses and the potential causes of self-harm. The International Human Rights Federation (Article 19) has recently reported that more than 20,000 defamation cases involving cyber abuses are waiting for trials, not to mention that many victims have not proceeded with their cases in any trial process. At the moment, researchers [3-5] have been developing Natural Language Processing (NLP) algorithms for hate speech detection on social media by using deep learning technologies and Neural Networks (NNs). The only research in this area performed in terms of Thai language was by Arreerard and Senivongse [6] who used machine learning Support Vector Machine (SVM) and Naïve Bayes with multiple word extraction techniques. It resulted in 74% accuracy. Therefore, our research aims to classify defamation sentences on Thai Facebook, where the learned text is derived from the comments under the pictures or under the articles of the person being mentioned, using three deep learning techniques: Long Short-Term Memory (LSTM), Bidirectional Long Short-Term Memory (Bi-LSTM), and Convolutional Neural Network (CNN), combining special features.

2. **Literature Review.** Given that many countries have recognized the problems and effects of cyber abuses, researchers have been applying cutting-edge technologies to tackle all the issues relating to them. In 2021, Sangwan and Bhatia [7] categorized the two most remarkable types of cyber abuses: Cyber-Hate (CH) and CB.

CH is the expression of hate on social media occurring in the past few years. Researchers [8-10] have developed a speech recognition tool to detect hate speech on Indonesian Facebook and Twitter by machine learning and deep learning. In 2020, Modha et al. [11] applied SVM, CNN, attention-based model, BERT pretrained language model techniques to classifying the types of abuses by a dataset that comprised comments from Facebook and Twitter in English and code-mixed Hindi. In the same period, Mossie and Wang [12] used Word2Vec embedding and GRU techniques with the training dataset of 5,876 Facebook posts and 485,548 Amharic language comments. It resulted in an accuracy of 92.56% for hate speech detection on SMPs.

In 2018, many researchers developed deep learning and transfer learning to detect speech within the scope of CB on SMPs. Agrawal and Awekar [13] used CNN, LSTM, Bi-LSTM, Bi-LSTM with attention, logistic regression, SVM, RF, and Naïve Bayes and trained their model by the dataset compiled from Twitter and Wikipedia. Bu and Cho [14] presented the CNN and LSTM of deep learning for detecting comments related to CB on the Internet and analyzed their model by using the t-SNE algorithm for investigating the relationship between syntactic and semantic subsets.

The abovementioned related studies on CH and CB point out that deep learning models can effectively detect cyber abuses. Many researchers [15-19] have indicated that LSTM,

Bi-LSTM, and CNN are the techniques widely used for analyzing messages to classify cyber abuses. Regarding the accuracy values, LSTM is 97.19%, Bi-LSTM is 89.05%, and CNN is 97.06%, which are all remarkably high. However, it also depends on the language and text domain aspects. Therefore, we anticipate an effective technique to develop our domain text.

Regarding Thai research related to NLP, many works apply deep learning techniques in various fields [20-22]. However, studies on Thai defamatory statements on social media by deep learning are limited because Thai language has such special aspects that it is quite challenging to work on; for example, letters or words can convey more than one definition. PyThaiNLP [23] is specially developed for processing Thai language. It has many functions, such as tokenization, Part-of-Speech (POS), and spell check. It is also applied to various researches in recent years. Ayutthaya and Pasupa [24] analyzed customers' feelings from Thai messages by LSTM and CNN model for marketing research. Later, Pasupa and Ayutthaya [25] compared the efficiency of deep learning models: CNN, LSTM, and Bi-LSTM, which are used to extract different word features, including word embedding, POS tag, and sentic feature. CNN, which applies these three features, has the highest accuracy rate at 81.7% in the analysis of Thai children toward child tales from 1,152 sentences of more than 40 stories.

In the studies mentioned above, many researchers have developed machine learning and deep learning and applied several features to identify sentences or messages related to cyber abuses in many languages. Arreerard and Senivongse [6] developed a model to identify defamatory statements by machine learning SVM and Naïve Bayes using text as the input; extracting word n-grams, char n-grams, dependency structure, sentiment polarity features; and creating a verdict vocabulary dictionary from specific terms. Their research showed that the efficiency of the model is not high because of the complexity of Thai language and the identification of key elements of the text, which are difficult to distinguish. Our research applies deep learning techniques, including LSTM, Bi-LSTM, and CNN, and extracts important word features, namely, dictionary judgment, word embedding, and POS-tag.

3. **Methodology.**

This section describes the creation processes of a model for classifying defamatory messages on social media by deep learning. There are three main parts of the processes: data preparation, features extraction, and deep learning model. An overview of the modeling process is shown in Figure 1.

3.1. **Data preparation.**

3.1.1. *Data cleansing.* To improve the model's learning efficiency, texts or sentences used as the dataset must be cleansed. Sentences and messages from Facebook often contain special characters or symbols that are unessential in the model's learning process. They may also cause word division errors to occur. The data cleansing process begins by removing hashtags from messages because commenters often use them to highlight the keywords they want to emphasize. For texts, any other languages apart from Thai are deleted. Then, all the unnecessary special characters are removed. Finally, the emojis used to express feelings in the comments are eliminated.

3.1.2. *Tokenization.* For text classification analysis, separating words from sentences and converting them into numbers or vectors are necessary, so that we can use them as input for the model's learning. Some well-structured languages, such as those in English, where each word is separated from each other (spacing), may not have any problem. However, with regard to Thai language, one of the challenging problems related to language research is word tokenization because it is a language written next to each other without word spacing. Applying an algorithm, which tokenizes words accurately to divide Thai words,

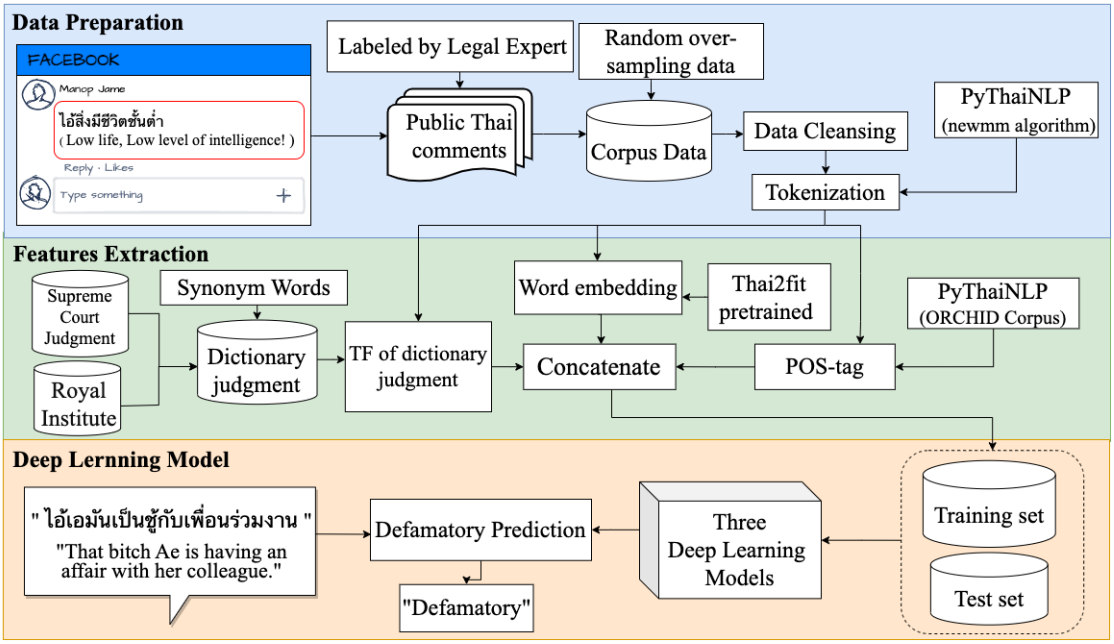

FIGURE 1. Systems architecture

FIGURE 2. Example of Thai language tokenization

จำเลยพูดถึงผู้เสียหายซึ่งเป็นหญิงรับราชการอยู่ที่สำนักงานที่ดินอำเภอว่า กระหรี่ที่ดิน และคำว่า"กระหรี่" หมายความว่า หญิงนครโสเภณีหรือหญิงค้าประเวณี แม้จำเลยจะไม่ได้กล่าวรายละเอียดว่าค้าประเวณีกับใคร ประพฤติสำส่อนในทางเพศกับใครบ้างก็เพียงพอที่จะถือว่าเป็นคำหมิ่นประมาทแล้ว

The defendant called the victim, a woman who is a government official at the district's Department of Lands as a land whore. The word "whore" means a prostitute or a person who has sex with someone for money. Although the defendant did not elaborate that the victim sells her service or has inappropriate sexual behaviors with whom, it is enough to be considered as defamation.

FIGURE 3. Judgment of the Supreme Court No. 2371/2522

is thus needed. We choose to tokenize words by using the PyThaiNLP library, algorithm newmm, (illustrated in Figure 2), which is an algorithm that divides each word in a sentence by using the maximal matching algorithm and Thai character cluster, which results in 88% precision. Although the precision is not 100%, our domain text is more suitable for this algorithm than others.

3.2. **Features extraction.**

3.2.1. *Dictionary judgment.* For defamation-related litigation, the judge looks at the composition of a sentence or a speech to see whether it constitutes an offense. Regarding the composition of the offense in the matter of impeachment, the judge picks up the terminology used to explain and give the definition of that word for determining the defendant's guilt (displayed in Figure 3). Thus, we compile the vocabulary from the petitions of the defamation cases related to Sections 326 and 328 of the Penal Code.

We also collect vocabularies from petitions related to Section 393 of the Penal Code, which mostly contains rude, insulting or disdainful terms. For vocabulary categorization, we refer to the research conducted by Arreerard [26] who proposed the categories of these

vocabulary groups in the dictionary. We decide to categorize them into the following seven groups:

1) Nouns: words used to refer to persons, animals, places, properties, states, symptoms or ideas.

2) Transitive verbs: verbs that depend on an object to support them to complete a sentence, for example, Mr. A "kills" Mr. B.

3) Intransitive verbs: verbs that do not depend on an object to complete a sentence, for example, Mr. A "is" dishonest.

4) Phrases or groups of words: words that are commonly used in sarcasm or figuratively used as indirect insults, such as "biting the hand that feeds you", which means to act ungratefully toward a person or a place that welcomes or treats you well.

5) Insults: words used to disdain or abuse others. This category also includes swear and rude words.

6) First-person pronouns: words used to describe oneself or identify defamatory persons, such as "ฉัน" and "ข้า" (I, me, mine, my, we, us, our, and ours).

7) Second- and third-person pronouns: words used to refer to one or more people who are accused, such as "มึง" (you), and "พวกมึง" (they).

We collect words from court judgments. Subsequently, we add synonyms and other words from the Royal Institute Dictionary. Finally, we have a judgment dictionary that contains 452 nouns, 155 transitive verbs, 269 intransitive verbs, 21 phrases or groups of words, 59 insults, 28 first-person pronouns, and 63 second- or third-person pronouns.

3.2.2. *POS-tag.* We decide to use POS-tag in hope that the model understands sentence structures well. Each word in a sentence is tagged to identify the location of each word type, such as nouns, verbs, pronouns, and adjectives. We use the Perceptron tagger, PyThaiNLP's library of POS-tag using the Perceptron algorithm. This library is divided into 47 word types, which are categorized by the ORCHID archive [27], which is a text corpus that outlines sentence boundaries, word boundaries, and word types in Thai language. We apply POS-tag as the input data for the experimental model to learn in two ways. The first one is POS-tag one-hot where we encode POS-tag to 47 dimensions, the same size obtained by tagging the library's POS-tag of the aforementioned PyThaiNLP. After separating words (tokenization), we look at each word in the sentence. If a word is found in 47 POS-tag types, then it is represented by "1"; otherwise, it is represented by "0". The second is POS-tag embedding, which is a POS conversion. Each type is a vector, and it represents each type of POS-tag.

3.2.3. *Word embedding.* Currently, extracting word properties by embedding words is becoming popular. It helps researchers understand the relationship of each word in Thai well. For Thai language, a significant application of Thai2Vec by PyThaiNLP is necessary. It is the pretraining with Thai Wikipedia data by using Universal Language Model Fine-tuning [28]. Regarding Thai2Vec, 60,000 words are found in the archive. Each word is replaced by a 300-dimensional vector. Recently, Thai2Vec has been developed and renamed as Thai2fit [29], and its scale has been adjusted to 400 dimensions.

3.3. **Deep learning model.** In this step, we divide the experiment into three types to compare algorithm performance.

3.3.1. *LSTM.* LSTM [30] is an algorithm developed from the Recurrent Neural Network (RNN), which is a deep learning algorithm based on the analysis of sequence data (Sequence) developed by RNN. It stores previous output values, compute them as the new input, and loop through the network from left to right.

3.3.2. *Bi-LSTM.* Bi-LSTM developed from LSTM has the same working process. However, a bidirectional simultaneous input function is added. Usually, the input data are added in one direction, from left to right, by the output. Although bidirectional input results in Bi-LSTM being slower than LSTM due to its large size, it is suitable for domain text that wants to understand word meanings from the context around it. Khongtum et al. [31] applied Bi-LSTM to performing entity recognition with Thai poem domain text. Complex contexts exist where Bi-LSTM works well in such a domain text.

3.3.3. *CNN.* NNs are networks or circuits of artificial nerve cells, which consist of nerve cells (Nodes) and nerve synapses (Dense). They are considered the model of human brain cell structure. CNN [32] is an architecture. It is developed from an NN with one input layer to receive data into the NN and is sent to the next hidden layer. A hidden layer can contain more than one layer; adding more layers results in more neurons, leading to high accuracy. The last part is the output layer, which receives values from the last hidden layer where the number of neurons in the output layer is equal to the number of classes. CNN is the addition of a layer of convolution processing into the NN. The convolution layer finds the relationship by extracting data features for CNN learning. The learning process comprises two parts: feature extraction and classification. Feature extraction is the process of bringing the local region of data to be learned gradually, with a filter or a kernel extracting special attributes for use during the classification process.

4. **Experiments.**

4.1. **Dataset.**

4.1.1. *Data collection.* We collect the learning dataset used for the defamatory statement classification model from one of the most popular SMPs – Facebook. Our dataset contains public user comments. The collected comments are placed under posts with topics about important or famous people who are interested in society, such as actors, politicians, and civil servants or companies and shops, as shown in Figure 4. Most of the messages collected are offensive, sarcastic or insulting. We collect 1,149 sentences of opinions, and they are regarded as defamatory. Defamatory sentences are labeled as 1, whereas non-defamatory sentences are labeled as 0. Experts, including legal professionals, prosecutors, and attorneys, are the ones to label the data used for the experiment. They consider whether the sentences are defamatory in accordance with Section 326 of the Defamation Code primarily based on the following elements.

1) Offenders or the people who raise accusations can be common and juristic persons.

2) Words or speeches that defame others. Whether speech words are true or false, if others are defamed, then the speech words are considered defamation. To illustrate, Mr. A is having an affair with Mr. B., and Mr. C knows about this relationship. When Mr. C tells others about it, he/she has raised an accusation that damages or defames others. Thus, it will be regarded as defamation.

3) The person being accused is the object of the action. Considering the concept of the Supreme Court, the judge must know who is accused or targeted by an accusation without having to specify that person's name. The one being accused can be an individual or a group of people.

4) For third parties, the offense of defamation under Section 326 states that defamation of others will be done to a third party if the offender has spread the insulting message of others to third parties.

5) Damages to others. If the offender's words cause damage(s) to the victim, whether true or false, decreasing the value or dignity of the person being accused, then the words are considered defamatory. Ultimately, we obtain a dataset containing 432 defamatory sentences and 717 non-defamatory sentences.

FIGURE 4. Examples of information collected from Facebook

4.1.2. *Random oversampling.* Information on sentences that are considered defamation is collected, so several necessary elements are found in sentences qualified as defamatory. The dataset obtained after the labeling by experts is tagged as defamatory. The amount of defamatory data is less than that of non-defamation data, thereby causing data imbalance. Therefore, we use oversampling to balance the dataset and reduce the overfitting problem. We randomly pick up data from the lesser class, which is Class 1, to make the data equal to Class 0. We randomly select 285 sentences. After random oversampling, we obtain a dataset of two classes with equal sentences. Each class is 717 sentences – a total of 1,434 sentences.

4.2. **Experiment setting.** In the experiment for evaluating the model effectiveness, as our dataset is small, splitting data for training and validation may yield different difficult or easy data to achieve reliable accuracy and low variance. Hence, we use the k-fold cross-validation method to divide the data [33]. Data for training and validation are divided into five equal parts ($k = 5$) by random sampling. Subsequently, we build and test the model until each information is trained and tested.

4.2.1. *Feature extraction setting.*

1) **Term Frequency (TF) of dictionary judgment:** We define a seven-dimensional matrix on the basis of the word types in the judgment dictionary described in Section 3. After tokenization, the words in sentences are searched for in the dictionary. If found, then the frequency will be counted. After the number of words found in the dictionary is completely counted, every value in the matrix is normalized by dividing all the frequency values. We obtain a seven-dimensional matrix, where each value does not exceed 1.

2) **POS-tag one-hot:** We create a 47-dimensional vector which has equal size to the word type in the ORCHID. After tokenization, each sentence has a 47-dimensional vector with each dimension represented as "0" or "1". If a word is found in 47 POS-tag types, then it is represented by "1"; otherwise, the value is "0".

3) **POS-tag embedding:** To embed a POS-tag, we match the words derived from the tokenization. The order of each word type in the ORCHID is 47 types. The POS-tag embedding dimensions are 48, with another dimension being added to the case of an unknown word.

4) **Word embedding:** To replace words with vectors, we set the word embedding vector size to 400 dimensions, which is the same size as Thai2fit from PyThaiNLP, a pretrained word embedding Thai language trained to learn with the Thai Wikipedia dataset. We also set the input sentence length as equal to the longest sentence length after the tokenization.

We create an embedding matrix and map a word vector from Thai2fit. If any word is not found in the dictionary, then we will randomly substitute a new vector and take the matrix as the word embedding weight. In addition, we turn off weight updates from our datasets during training.

4.2.2. *Deep learning model setting.*
1) **LSTM and Bi-LSTM:** We design a simple LSTM and Bi-LSTM network model and adjust LSTM parameters the same way as Bi-LSTM, which is illustrated in Figure 5. For the LSTM architecture, we create an embedding layer. Then, we attach it with the LSTM layer comprising 128 hidden nodes. In addition, Bi-LSTM architecture is configured that same way as LSTM. Then, each model is concatenated to the other input features presented in the experiment.

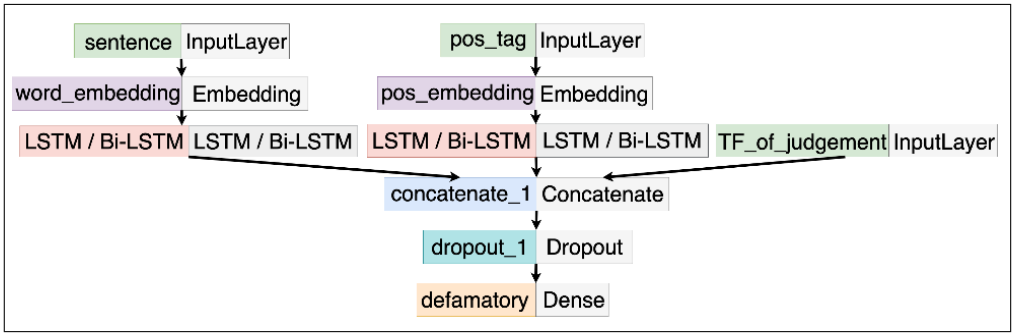

FIGURE 5. LSTM and Bi-LSTM structures and parameters

2) **CNN:** Using CNN in this research, we set parameters by creating Convolutional Layer 1, which defines a filter size of 128 and a kernel size of 4, using the activation function rectified linear unit. We then extract the most important part of the data and increase the processing efficiency to be fast by defining the max pooling layer. Subsequently, we flatten to convert matrix n-dimensions to one dimension to prepare it for concatenation with other input features, as displayed in Figure 6.

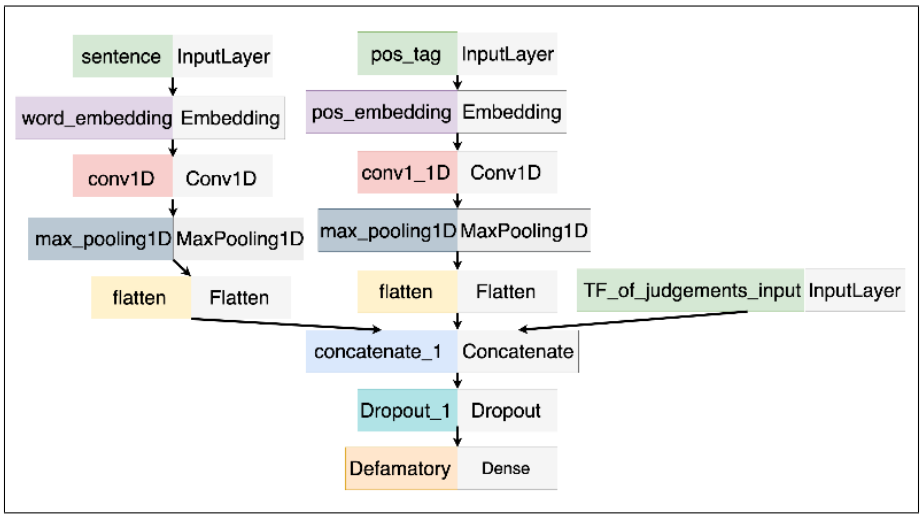

FIGURE 6. CNN structures and parameters

For all models, we use a dropout layer of 0.2 size to avoid overfitting and apply it with a 1-dimensional layer for classifying, which applies the activation function sigmoid. The loss function is binary cross entropy, which uses learning rate and batch size of 0.0001 and 64, respectively.

5. **Result and Discussion.** The performances of the three models are compared by combining three different features: word embedding, TF of dictionary judgment, and POS-tag as the input to train the model. For each experiment, we use 300 learning epochs. In this study, the model performances are evaluated by considering various aspects, such as precision, recall, F1-score, and accuracy based on the average calculation of the experiment

using five-fold cross-validation. When dividing the data to train and validate each fold, the numbers of the two classes are unequal (imbalanced classes), so we use a weighted average as the basis for estimating the overall average efficiency.

The experiment shown in Table 1 illustrates that the three models give the best performances when applying three features as input. The applied features are word embedding, TF of dictionary judgment, and POS-tag embedding. In the comparison among models, CNN shows the highest performance, with precision (P), recall (R), F1-score (F), and accuracy (A) of 86.17%, 85.91%, 85.59%, and 85.91%, respectively. The LSTM and Bi-LSTM models provide similar performance values at accuracies of 83.19% and 83.75%, respectively. The CNN model is integrated with the features presented outperforming LSTM and Bi-LSTM. Some specific legal elements in this research identify defamatory sentences. This factor relates to the research conducted by Yin et al. [34] who compared the performances of CNN and RNN models with each type of domain text. The results demonstrate that the efficiency of a model depends on the nature of the data.

Table 1. Percentage experiment results

Feature input	LSTM				Bi-LSTM				CNN			
	P	R	F	A	P	R	F	A	P	R	F	A
No. 1	80.81	80.54	80.53	80.40	83.02	82.77	82.75	82.77	83.51	83.26	83.25	83.26
No. 2	83.32	81.03	80.16	81.03	83.29	82.57	82.5	82.57	84.67	84.37	84.34	84.17
No. 3	78.37	77.69	77.29	77.69	83.18	80.68	80.09	80.68	83.94	83.75	83.74	83.75
No. 4	82.84	82.49	82.45	82.49	83.2	82.91	82.87	82.91	84.65	84.31	84.27	84.31
No. 5	82.27	81.79	81.77	81.79	82.92	82.56	82.53	82.56	84.67	84.37	84.33	84.38
No. 6	82.04	81.73	81.71	81.73	83.41	82.77	82.71	82.77	85.3	85.01	84.97	85
No. 7	**83.49**	83.05	82.97	83.05	83.87	83.54	83.49	83.54	85.18	84.68	84.82	84.86
No. 8	83.46	**83.19**	**83.17**	**83.19**	**84.05**	**83.75**	**83.54**	**83.75**	**86.17**	**85.91**	**85.59**	**85.91**

No. 1: Word embedding
No. 2: TF of dictionary judgment + POS-tag one-hot
No. 3: TF of dictionary judgment + POS-tag embedding
No. 4: Word embedding + TF of dictionary judgment
No. 5: Word embedding + POS-tag one-hot
No. 6: Word embedding + POS-tag embedding
No. 7: Word embedding + TF of dictionary judgment + POS-tag one-hot
No. 8: Word embedding + TF of dictionary judgment + POS-tag embedding

Furthermore, Thai2fit pretrained word embedding does not significantly improve model performance. The model learns from Wikipedia archive, which differs from our unique dataset that is full of profane sentences. Consequently, we cannot weight the words to understand the relationships among them as easily as it should be.

Table 1 presents that No. 8 with CNN is the most efficient. To evaluate whether the predictive effect of the generated model is accurate, we experiment with the newly created unseen dataset, which consists of 100 sentences. The prediction results show the confusion matrix in Figure 7. The model is reasonably predictable from the True Positive and True Negative observations.

Moreover, our dataset size is small because its data are specialized, which are difficult to collect and require expert opinions. We compare our experiment with old previous ones (state of the art) by using two machine learning methods: Naïve Bayes and SVM. Both apply word extraction and word characteristics, which present the TF of dictionary judgment and POS-tag one-hot. SVM provides precision, recall, F1-score, and accuracy at 76.28%, 79.76%, 77.90%, and 77.33%, respectively. Our experiment results are close to those obtained by Arreerard and Senivongse [6] who used a different test dataset. Their experiment revealed that SVM has the best accuracy and F1-score at 74% and 64%, respectively. Our experiment notably has better efficiency.

The experiment on feature inputs Nos. 7 and 8 with two-word vector forms indicates a slight, nonsignificant increase efficiency. This remark is probably because the TF of dictionary judgment only has seven dimensions compared with the POS-tag that has

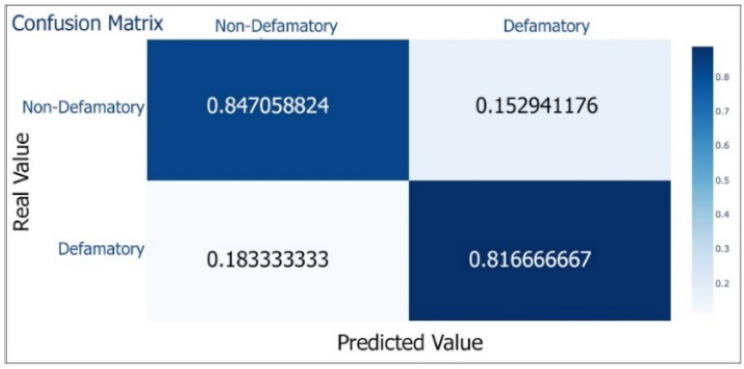

FIGURE 7. Confusion matrix

47 and 48 dimensions. Thus, dictionary judgment has a lower weight score than POS-tag. Regarding the performance comparison during the model training using all methods, the data are divided into train and validation datasets with five-fold cross-validation. Training accuracy and error (loss) values are recorded in the best fold training. Each model receives the same dataset because we assign the same random state. The accuracy and loss values during the model training, using the input feature of No. 8 illustrated in Figure 8, indicates that the accuracy values of all three models have similar values. By contrast, the loss values of LSTM and Bi-LSTM are more prone to overfitting than that of the CNN model.

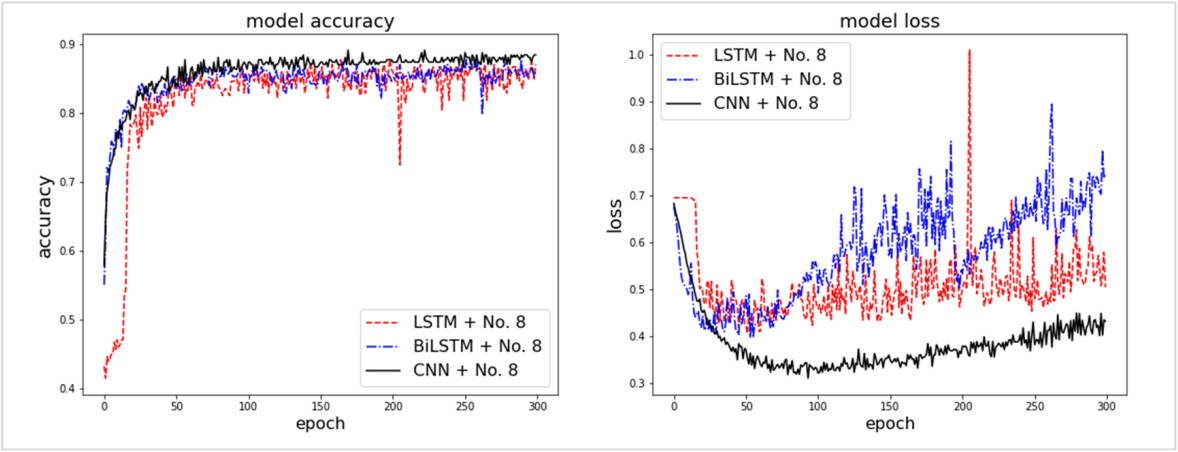

FIGURE 8. Plot for accuracy and loss values on training and validation

6. **Conclusion.** This article presents a model to classify defamatory messages on Facebook by using three deep learning techniques: LSTM, Bi-LSTM, and CNN. We use text feature extraction, TF of dictionary judgment, and POS-tag. We also weight Thai2fit pretrained word vector in Thai language for embedding words to help the model further understand the elements of defamatory sentences. The experiment results indicate that CNN using Thai2fit is employed to embed words combined with two key attributes as input: TF of dictionary judgment and POS-tag embedding. CNN also exhibits the best performance, with precision, recall, F1-score, and accuracy of 86.17%, 85.91%, 85.59%, and 85.91%, respectively. In our research, a simple model is set up, in which tuning model parameters can result in improved efficiency. In the future, we will develop a platform that can be used to help determine whether posted messages are considered defamatory according to Thai law. We will also collect additional data and develop our model. We may use blended learning, as in Dyoniputri and Afiahayati [35], to help isolate keyword attributes for improved performance.

REFERENCES

[1] A. John, A. C. Glendenning, A. Marchant, P. Montgomery, A. Stewart, S. Wood and K. Hawton, Self-harm, suicidal behaviours, and cyberbullying in children and young people: Systematic review, *Journal of Medical Internet Research*, vol.20, no.4, e9044, 2018.

[2] Thailand Law Library, *Defamation*, 2018, https://library.siam-legal.com/thai-law/criminal-code-defamation-sections-326-333, Accessed on 20 April 2021.

[3] S. Morzhov, Avoiding unintended bias in toxicity classification with neural networks, *2020 26th Conference of Open Innovations Association (FRUCT)*, pp.314-320, 2020.

[4] A. S. Uban and L. P. Dinu, On transfer learning for detecting abusive language online, *International Work-Conference on Artificial Neural Networks*, pp.688-700, 2019.

[5] K. Kumari, J. P. Singh, Y. K. Dwivedi and N. P. Rana, Aggressive social media post detection system containing symbolic images, *Conference on e-Business, e-Services and e-Society*, pp.415-424, 2019.

[6] R. Arreerard and T. Senivongse, Thai defamatory text classification on social media, *2018 IEEE International Conference on Big Data, Cloud Computing, Data Science & Engineering (BCD)*, pp.73-78, 2018.

[7] S. R. Sangwan and M. P. S. Bhatia, Soft computing for abuse detection using cyber-physical and social big data in cognitive smart cities, *Expert Systems*, e12766, 2021.

[8] T. L. Sutejo and D. P. Lestari, Indonesia hate speech detection using deep learning, *International Conference on Asian Language Processing (IALP)*, pp.39-43, 2018.

[9] E. Sazany and I. Budi, Deep learning-based implementation of hate speech identification on texts in Indonesian: Preliminary study, *2018 International Conference on Applied Information Technology and Innovation (ICAITI)*, pp.114-117, 2018.

[10] N. I. Pratiwi, I. Budi and M. A. Jiwanggi, Hate speech identification using the hate codes for Indonesian tweets, *Proc. of the 2019 2nd International Conference on Data Science and Information Technology,* pp.128-133, 2019.

[11] S. Modha, P. Majumder, T. Mandl and C. Mandalia, Detecting and visualizing hate speech in social media: A cyber watchdog for surveillance, *Expert Systems with Applications*, vol.161, 113725, 2020.

[12] Z. Mossie and J. H. Wang, Vulnerable community identification using hate speech detection on social media, *Information Processing & Management*, vol.57, no.3, 102087, 2020.

[13] S. Agrawal and A. Awekar, Deep learning for detecting cyberbullying across multiple social media platforms, *European Conference on Information Retrieval*, pp.141-153, 2018.

[14] S. J. Bu and S. B. Cho, A hybrid deep learning system of CNN and LRCN to detect cyberbullying from SNS comments, *International Conference on Hybrid Artificial Intelligence Systems*, pp.561-572, 2018.

[15] H. H. Saeed, K. Shahzad and F. Kamiran, Overlapping toxic sentiment classification using deep neural architectures, *2018 IEEE International Conference on Data Mining Workshops*, pp.1361-1366, 2018.

[16] S. K. Maity, A. Chakraborty, P. Goyal and A. Mukherjee, Opinion conflicts: An effective route to detect incivility in Twitter, *Proc. of the ACM on Human-Computer Interaction*, pp.1-27, 2018.

[17] H. Mohaouchane, A. Mourhir and N. S. Nikolov, Detecting offensive language on Arabic social media using deep learning, *2019 IEEE 6th International Conference on Social Networks Analysis, Management and Security (SNAMS)*, pp.466-471, 2019.

[18] M. Anand and R. Eswari, Classification of abusive comments in social media using deep learning, *2019 3rd International Conference on Computing Methodologies and Communication (ICCMC)*, pp.974-977, 2019.

[19] J. Chen, S. Yan and K. C. Wong, Verbal aggression detection on Twitter comments: Convolutional neural network for short-text sentiment analysis, *Neural Computing and Applications*, vol.32, no.15, pp.10809-10818, 2020.

[20] S. Thattinaphanich and S. Prom-on, Thai named entity recognition using Bi-LSTM-CRF with word and character representation, *The 4th International Conference on Information Technology (InCIT)*, pp.149-154, 2019.

[21] T. Chiewhawan and P. Vateekul, Explainable deep learning for Thai stock market prediction using textual representation and technical indicators, *Proc. of the 8th International Conference on Computer and Communications Management*, pp.19-23, 2020.

[22] T. Horsuwan, K. Kanwatchara, P. Vateekul and B. Kijsirikul, A comparative study of pretrained language models on Thai social text categorization, *Asian Conference on Intelligent Information and Database Systems*, pp.63-75, 2020.

[23] W. Phatthiyaphibun, *PyThaiNLP*, GitHub, 2018, https://github.com/PyThaiNLP/pythainlp, Accessed on 14 July 2021.

[24] T. S. N. Ayutthaya and K. Pasupa, Thai sentiment analysis via bidirectional LSTM-CNN model with embedding vectors and sentic features, *International Joint Symposium on Artificial Intelligence and Natural Language Processing (iSAI-NLP)*, pp.1-6, 2018.

[25] K. Pasupa and T. S. N. Ayutthaya, Thai sentiment analysis with deep learning techniques: A comparative study based on word embedding, POS-tag, and sentic features, *Sustainable Cities and Society*, vol.50, 101615, 2019.

[26] R. Arreerard, *Defamatory Text Classification on Online Social Media*, Master Thesis, Computer Science Department of Computer Engineering, Faculty of Engineering, Chulalongkorn University, 2018.

[27] V. Sornlertlamvanich, T. Charoenporn and H. Isahara, *ORCHID: Thai Part-of-Speech Tagged Corpus*, National Electronics and Computer Technology Center Technical Report, pp.5-19, 1997.

[28] J. Howard and S. Ruder, Universal language model fine-tuning for text classification, *arXiv Preprint*, arXiv: 1801.06146, 2018.

[29] C. Polpanumas, *ULMFit Language Modeling, Text Feature Extraction, and Text Classification in Thai Language*, GitHub, 2019, https://github.com/cstorm125/thai2fit, Accessed on 14 July 2021.

[30] S. Hochreiter and J. Schmidhuber, Long short-term memory, *Neural Computation*, vol.9, no.8, pp.1735-1780, 1997.

[31] O. Khongtum, N. Promrit and S. Waijanya, The entity recognition of Thai poem compose by Sunthorn Phu by using the bidirectional long short term memory technique, *International Conference on Multi-Disciplinary Trends in Artificial Intelligence*, pp.97-108, 2019.

[32] R. Hecht-Nielsen, Theory of the backpropagation neural network, *International 1989 Joint Conference on Neural Networks*, pp.65-93, 1992.

[33] T. T. Wong and P. Y. Yeh, Reliable accuracy estimates from k-fold cross validation, *IEEE Transactions on Knowledge and Data Engineering*, vol.32, no.8, pp.1586-1594, 2019.

[34] W. Yin, K. Kann, M. Yu and H. Schütze, Comparative study of CNN and RNN for natural language processing, *arXiv Preprint*, arXiv: 1702.01923, 2017.

[35] H. Dyoniputri and Afiahayati, A hybrid convolutional neural network and support vector machine for dysarthria speech classification, *International Journal of Innovative Computing, Information and Contro*, vol.17, no.1, pp.111-123, 2021.