SMS Spam Detection Based on Long Short-Term Memory and Gated Recurrent Unit

Pumrapee Poomka, Wattana Pongsena, Nittaya Kerdprasop, and Kittisak Kerdprasop

***Abstract*—An SMS spam is the message that hackers develop and send to people via mobile devices targeting to get their important information. For people who are ignorant, if they follow the instruction in the message and fill their important information, such as internet banking account in a faked website or application, the hacker may get the information. This may lead to loss their wealth. The efficient spam detection is an important tool in order to help people to classify whether it is a spam SMS or not. In this research, we propose a novel SMS spam detection based on the case study of the SMS spams in English language using Natural Language Process and Deep Learning techniques. To prepare the data for our model development process, we use word tokenization, padding data, truncating data and word embedding to make more dimension in data. Then, this data is used to develop the model based on Long Short-Term Memory and Gated Recurrent Unit algorithms. The performance of the proposed models is compared to the models based on machine learning algorithms including Support Vector Machine and Naïve Bayes. The experimental results show that the model built from the Long Short-Term Memory technique provides the best overall accuracy as high as 98.18%. On accurately screening spam messages, this model shows the ability that it can detect spam messages with the 90.96% accuracy rate, while the error percentage that it misclassifies a normal message as a spam message is only 0.74%.**

***Index Terms*—SMS spam, natural language process, deep learning, long short-term memory, gated recurrent unit.**

I. Introduction

Presently, the communicational technology is rapidly glowing. This means everybody can access or receive the information easier than the past using web browsing, text messaging and emailing. With the ability of these technologies, the hacker can exploit the advantages of them to get the sensitive information, such as internet banking account, credit card information and phone number from a people by sending a faked bank transaction notification or a deceive advertisement message [1]. Moreover, the innocent people who getting message and panicked follow the instruction from the hacker and give their sensitive information to the hacker. Then, the hacker uses this information for getting asset from people. Finally, the innocent people lose asset to the hacker. This situation can make affectation by loss reputation or loss asset.

Short Message Service (SMS) spam is a type of the spam messages that hacker send it via mobile devices targeting to get their sensitive information [2], [3]. Nowadays, a SMS spam attack is greatly affecting various people who trust the information in the message and follow the instruction from the hacker. This problem can be alleviated if we have a tool that can effectively detect the spam messages.

The existing works related to the SMS spam detection, was developed based on machine learning techniques, such as Support Vector Machine and Naïve Bayes [2]. Although these methods have a high performance, they seem to be difficult to config the parameter in term of statistical data.

Currently, Deep Learning is a popular technique that is used to analyze data as it usually provides a high accuracy of the prediction [4], [5]. The algorithm that is commonly used for analyzing sequential data (e.g., text data) is Recurrent Neural Networks (RNNs) [5]. RNNs have the modified version of algorithms such as Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU). Many existing works claim that LSTM and GRU have a performance over other techniques in deep learning particularly for sequential data analysis. For example, Kraus and Feuerriegel [6] used economic news for developing a decision support system for assisting investors having more secure in their investment using LSTM. They also use transfer learning, word tokenization and word embedding to prepare data for their analysis. Dou *et al*. [7] developed knowledge graph of Chinese intangible cultural heritage using data from websites with Bidirectional GRU. Another example the model, which was developed by Hassan and Mahmood [8]. Their model based on LSTM targeting to classify movie reviews from IMDB dataset with word embedding technique.

For working with SMS spam, there is a lag of research that applying deep learning algorithm to develop models. Therefore, in this research, we aim to develop SMS spam classify model for preventing people from the effect of SMS spam. we propose to develop SMS spam classify model using Natural Language Process (NLP) techniques for data preparation and developing model based on GRU and LSTM. Moreover, we evaluate developed models by comparing the accuracy performance of deep learning model against the machine learning models. we aim to develop SMS spam classification model based on deep learning technique. Because the common type of SMS spam is text data, we use NLP which is the algorithm to make computer understand natural language same as human. Moreover, the popular of social network, text messaging and the article on websites presently. This information is easily collected and more effective for analyzing.

Manuscript received November 10, 2018; revised February 19, 2019.

Pumrapee Poomka, Wattana Pongsena, and Nittaya Kerdprasop are with the School of Computer Engineering, Suranaree University of Technology, Nakhon Ratchasima, 30000, Thailand (email: pumrapee.p@outlook.com, watthana.p@sskru.ac.th, nittaya@sut.ac.th).

Kittisak Kerdprasop is with the School of Computer Engineering, Suranaree University of Technology, Nakhon Ratchasima, 30000, Thailand (email: kerdpras@sut.ac.th).

doi: 10.18178/ijfcc.2019.8.1.532

The rest of this research is organized as follows. Section II describes research framework, data, and methodology used for conducting this research. The experimental results are discussed in section III. Finally, section IV represents our conclusions and suggestion for future works.

II. Materials and Methods

A. Research Framework

In this research, we use Keras [9] that is a library in Python for deep learning model development based on Tensorflow backend [10]. It has a toolset for data preparation, such as word tokenization [11], padding and truncating data [12], and word embedding [13]. The word tokenization technique is used for taking text inputs into sequential data as index values of the words. The padding and truncating data techniques are used to make all sequence having the same length, while the word embedding technique is used to make more dimension of sequence into vector. After data preparation process, we train the model based on LSTM and GRU algorithms. Then, we evaluate the performance of the models and compare their performance with the model based of machine learning algorithms. The working flow of the framework shows in Fig. 1.

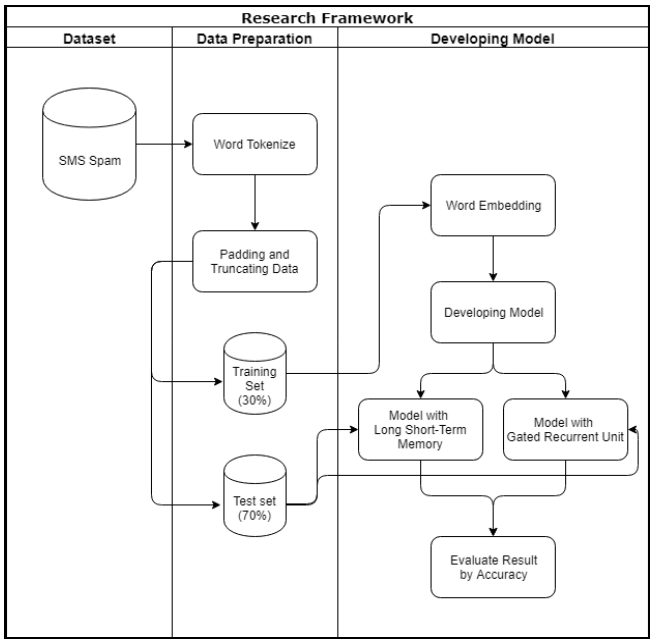

Fig. 1. Research framework.

TABLE I: Sample Record of SMS Spam Dataset

Message	Spam
Go until jurong point, crazy.. Available only in bugis n great world la e buffet... Cine there got amore wat...	0
Ok lar... Joking wif u oni...	0
Free entry in 2 a wkly comp to win FA Cup final tkts 21st May 2005. Text FA to 87121 to receive entry question(std txt rate)T&C's apply 08452810075over18's	1
Lol your always so convincing.	0
SIX chances to win CASH! From 100 to 20,000 pounds txt> CSH11 and send to 87575. Cost 150p/day, 6days, 16+ TsandCs apply Reply HL 4 info	1

B. Datasets

In this experiment, we use a SMS spam dataset proposed by Almeida and Hidalgo [14]. This dataset consists of approximately 5,574 records. It contains SMS text messaging conversations in English language, which include text and number in different length of sentences. All records in this dataset already labeled. The spam messages are labelled as 1 (747 records) and the normal messages are labelled as 0 (4,827 records). The example of the dataset illustrated in Table 1.

C. Data Preparation

In this process, Natural Language Process (NLP) is utilized for pre-processing natural language data. NLP is the process to make computer understanding the natural language as same as a human understands [15]. It has many techniques to preprocess data into a format that a computer can be understood. In this work, we transform the SMS text data into sequential data using NLP techniques in order to use it for developing SMS classification models using the LSTM and GRU algorithms. We also use word tokenization, padding data, truncating data and word embedding techniques for pre-processing data. The details of each technique that we use in the data pre-processing process are described as follows.

D. Word Tokenization

Word tokenization is the process that changes words in a sentence into index values represented by a number. In this process, we set number of interesting vocabulary words to create word tokenizer. After create tokenizer, we use the word tokenizer to convert words in a sentence into sequence data. The tokenizer changes word into index and set index to 0 for unknow words [11]. In addition, we create a tokenizer by set number of vocabulary words to 10,000 words. We also use the tokenizer to convert text data into sequence of index number of words from the tokenizer as demonstrated in Table 2.

TABLE II: Sample Data before and after Tokenized

Before tokenize	Go until jurong point, crazy.. Available only in bugis n great world la e buffet... Cine there got amore wat...
After tokenize	[49 471 4435 842 755 658 64 8 1327 88 123 351 1328 148 2996 1329 67 58 4436 144]

E. Padding and Truncating Data

In this process, we make all sequence in dataset having the same length for training using LSTM and GRU algorithms. We calculate the message length optimized based on (1). [12] After optimized the length of the message, we pad data that have the length lower than the optimal length by add zero at the beginning of the sequence until it has the same length with the optimal one. If the length of the data is larger than the optimal length, we truncate the data from the beginning until its length is equal to the optimal length.

$$Optimize = mean(len(x_i)) + 2 \times std(len(x_i)) \quad (1)$$

where x_i is the records of dataset, *len(x)* is the function to calculate the length of message, *mean(x)* is the function to calculate mean of data and *std(x)* is the function to calculate standard deviation of data. We calculate the message length optimize based on (1). As the result of calculating, the optimal length is 200 the cover data about 97.95% of datasets. Thus, we use this optimal length to control padding and truncating in sequence.

F. Word Embedding

Word embedding technique, which is used in this research was created by Pennington *et al*. [13]. This technique is used to change sequence of words that we already pre-processed into vector representation called embedding space that contains more dimensions than the normal word data using to train with LSTM and GRU algorithms. After padding and truncating data, we use the word embedding technique to make more dimension for data in sequence by setting the embedding size to 32.

After pre-processing the data with NLP techniques described above, we split the data into training set 30% and test set 70% [2]. As a result, the training data set contains totally 238 spam messages and 509 normal messages. For the testing dataset, it contains totally 1,436 spam messages and 3,391 normal messages as showed in Table 3.

TABLE III: DETAIL OF THE DATASET AFTER SPILTED

	Spam (record)	Not Spam (record)
Training set	238	1436
Test set	509	3391

G. Modeling

In this experiment, we develop the SMS spam classification models based on the two deep learning algorithms including LSTM and GRU algorithms. The details of each algorithm described as follows.

H. Long Short-Term Memory (LSTM)

LSTM is developed by Hochreiter and Schmidhuber [16], [17] in 1997. It improves the basic RNN algorithm that solves the vanishing problem by adding cell states for remembering or forgetting data. The cell states contain structure called cell gates. The cell gates consist of four parts including input gate, forget gate, memory-cell state gate, and output gate. The input is gate used to control the input data that is worthwhile to keep or not. The forget gate is used to control the previous hidden state that is to be kept in the memory cell of the current hidden state. The memory-cell state gate is used to update the data based on the information of the input gate and the forget gate. The output gate is used to compute the output data from the network based on the memory-cell state. The details of LSTM are illustrated in Fig. 2.

I. Gated Recurrent Unit (GRU)

The GRU is a type of deep learning algorithm that is improved from the LSTM algorithm to reduce the complexity of the structure of the algorithm by using update gate and reset gate [17], [18]. The update gate is used to control amount of the hidden state to be forwarded to the next state. The reset gate is used to define the significance of the previous hidden state information as showed in Fig. 3.

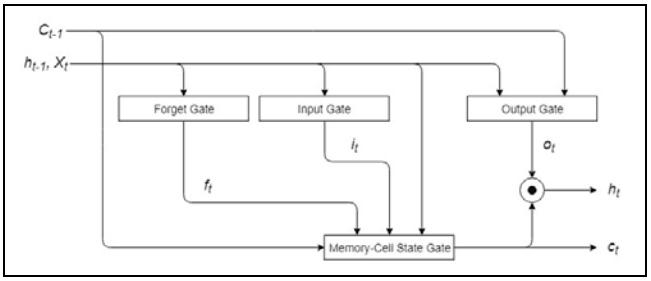

Fig. 2. Structure of LSTM cell.

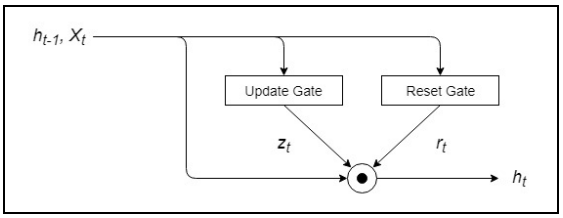

Fig. 3. Structure of GRU cell.

TABLE IV: HYPERPARAMETER SETUP FOR LSTM AND GRU

Parameter	LSTM	GRU
Vocabulary	10,000	10,000
Max sequence length	200	200
Embedding size	32	32
Unit layers	2 layers (32, 32)	2 layers (32, 32)
Dropout	0.2	0.2
Feature layer	Fully-Connected (16, ReLU)	Fully-Connected (16, ReLU)
Classifying layer	Fully-Connected (1, Sigmoid)	Fully-Connected (1, Sigmoid)
Optimizer	Adam	Adam
Learning rate	0.01	0.01
Loss function	Binary cross-entropy	Binary cross-entropy
Training epoch	10	10

For our model development, we set hypermeters and structure of the networks as showed in Table 4. In addition, we add a dropout layer after the unit layer in order to prevent the models from the over-fitting issue. Then, we use fully-connected with Rectifier function (ReLU) as an activation function to down-sampling from the unit layer. In the Classifying layer, we use fully-connected with Sigmoid function as an activation function to predict whether it is a spam or a normal message. In the training process, we choose Adam optimizer as an optimizer, binary cross-entropy as loss function because it is a binary classification. The learning rate is 0.01, and the model learning cycle is set to 10 epochs.

III. Results and Discussions

After training the models, we evaluate the performance of the model using the testing dataset. The results demonstrate that the accuracy of the model-based LSTM and GRU is similar. However, the LSTM model (90.96%) catch the spams more accurate than the GRU model (86.25%). In term of blocking normal message, both LSTM and GRU models have blocking rate lower than 1%. Based on these results, it could be concluded that overall, the LSTM model can classify SMS spam better than the GRU model.

We also compare the results of the proposed model based-deep learning algorithms with the comparative work based on machine learning algorithms including Support Vector Machine (SVM) and Naïve Bayes (NB) models proposed by Almeida and Hidalgo [2]. These models are developed using the same dataset with our proposed models. The results of comparison show in Table 6. As can be seen in Table 6, the LSTM and GRU algorithms have a higher performance than the SVM and NB algorithms. From these results, it might be stated that the deep learning algorithms provide a better performance than the models developed based on model based-machine learning algorithms for SMS spam classification in English language.

TABLE V: Result of LSTM and GRU with Test Set

Models	Accuracy (%)	Spam catch (%)	Block non-spam (%)
LSTM	98.18	90.96	0.74
GRU	98.03	86.25	0.21

TABLE VI: Comparing Results of Deep Learning with Machine Learning Algorithms

Models	Accuracy (%)	Spam catch (%)	Block non-spam (%)
SVM [2]	97.64	83.10	0.18
NB [2]	92.05	48.53	1.42
LSTM	**98.18**	**90.96**	**0.74**
GRU	98.03	86.25	0.21

IV. Conclusion

In this research, we propose SMS spam classification models based on deep learning algorithms including LSTM and GRU. We used NLP techniques for pre-processing SMS text data into sequence using word tokenization, padding data, truncating data and word embedding technique. In addition, we developed model based on deep learning algorithms including LSTM and GRU. Finally, we evaluated models using test set spilt from SMS spam dataset. The results show that the performance of the LSTM model outperforms other models with 98.18% accuracy. In addition, it catches overall spam message with 90.96% and catches normal message as a spam message with 0.74% error. Moreover, LSTM and GRU model, which are deep learning algorithms provide a better performance than the model based on machine learning algorithms including Support Vector Machine and Naïve Bayes.

This research was study-case about developing SMS spam classification model based on deep learning algorithms. For future works, we aim to enhance the performance of the model by collecting more data from various source to develop model. We expect to develop the model that can be used to help people in real-world.

Acknowledgment

This research has been financially supported by Data and Knowledge Engineering Research Unit, Suranaree University of Technology (SUT), and the National Research Council of Thailand. The first and second authors have been supported by scholarships from SUT and the Ministry of Science and Technology, Thailand, respectively.

References

[1] N. Jindal and B. Liu, "Review spam detection," in *Proc. of the 16th International Conference on World Wide Web*, 2007, pp. 1189-1190.
[2] T. A. Almeida, J. M. G. Hidalgo, and T. P. Silva, "Towards SMS spam filtering: Results under a new dataset," *International Journal of Information Security Science*, vol. 2, No. 1, pp. 1-18, 2013.
[3] T. A. Almeida, T. P. Silva, I. Santos, and J. M. G. Hidalgo, "Text normalization and semantic indexing to enhance instant messaging and SMS spam filtering," *Knowledge-Based Systems*, vol. 108, pp. 25-32, 2016.
[4] J. Schmidhuber, "Deep learning in neural networks: An overview," *Neural Networks*, vol. 61, pp. 85-117, 2015.
[5] Y. LeCun, Y. Bengio, and G. Hinton. "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436-444, 2015.
[6] M. Kraus and S. Feuerriegel, "Decision support from financial disclosures with deep neural networks and transfer learning," *Decision Support Systems*, vol. 104, pp. 38-48, 2017.
[7] J. Dou, J. Qin, Z. Jin, and Z. Li, "Knowledge graph based on domain ontology and natural language processing technology for Chinese intangible cultural heritage," *Journal of Visual Languages & Computing*, vol. 48, pp. 19-28. 2018.
[8] A. Hassan and A. Mahmood, "Deep learning for sentence classification," in *Proc. of 2017 IEEE Long Island Systems, Applications and Technology Conference (LISAT)*, 2017, pp. 1-5.
[9] F. Chollet, *et al.* (2015). Keras. [Online]. Available: https://keras.io
[10] M. Abadi, *et al.*, "Tensorflow: A system for large-scale machine learning," in *Proc. of the 12th USENIX conference on Operating Systems Design and Implementation*, 2016, pp. 265-283.
[11] B. Habert *et al.*, "Towards tokenization evaluation," in *Proc. of the International Conference on Language Resources and Evaluation*, 1998, pp. 427-431.
[12] M. E. H. Pedersen. (2018). TensorFlow Tutorial—#20 Natural Language Processing. [Online]. Available: https://github.com/Hvass-Labs/TensorFlow-Tutorials/blob/master/20_Natural_Language_Processing.ipynb
[13] J. Pennington, R. Socher, and C. Manning, "Glove: Global vectors for word representation," in *Proc. of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2014, pp. 1532-1543.
[14] T. A. Almeida and J., M., G. Hidalgo. (2011). SMS Spam Collection v.1. [Online]. Available: http://www.dt.fee.unicamp.br/~tiago/smsspamcollection/
[15] F. Olsson, "A literature survey of active machine learning in the context of natural language processing," *Swedish Institute of Computer Science*, 2009.
[16] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735-1780, 1997.
[17] B. Premjith, K. P. Soman, and M. A. Kumar, "A deep learning approach for Malayalam morphological analysis at character level," *Procedia Computer Science*, vol. 132, pp. 47-54, 2018.
[18] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," *E-print arXiv: 1412.3555*, 2014.

Pumrapee Poomka is a master student, School of Computer Engineering, Suranaree University of Technology (SUT), Thailand. He received his B.E. in computer engineering from Suranaree University of Technology, Thailand, in 2018. His research of interest includes artificial intelligence, machine learning and data mining.

Watthana Pongsena is a Ph.D. student at School of Computer Engineering, Suranaree University of Technology (SUT), Thailand. He received his B.E. and M.E. in computer engineering from Suranaree University of Technology, Thailand, in 2008 and 2012. His research of interest includes software engineering, data mining, artificial intelligence, and human-computer interaction.

Nittaya Kerdprasop is an associate professor and the head of Data Engineering Research Unit, School of Computer Engineering, Suranaree University of Technology (SUT), Thailand. She received her B.S. in radiation techniques from Mahidol University, Thailand, in 1985, M.S. in computer science from the Prince of Songkla University, Thailand, in 1991 and Ph.D. in computer science from Nova Southeastern University, U.S.A., in 1999. Her research of interest includes data mining, artificial Intelligence, logic, and constraint programming.

Kittisak Kerdprasop is an associate professor at the School of Computer Engineering, Chair of the School, and the head of Knowledge Engineering Research Unit, SUT. He received his bachelor degree in Mathematics from Srinakarinwirot University, Thailand, in 1986, MS in computer science from the Prince of Songkla University, Thailand, in 1991 and Ph.D. in computer science from Nova Southeastern University, U.S.A., in 1999. His current research includes machine learning and artificial intelligence.