Generalized Information Extraction for Thai Web Boards

Apichai Dangmadee[1], Parinya Sanguansat[1], and Choochart Haruechaiyasak[2]

[1]Faculty of Engineering and Technology,
Panyapiwat Institute of Management, Nonthaburi, Thailand
[2]Speech and Audio Technology Laboratory,
National Electronics and Computer Technology Center, Pathumthani, Thailand
E-mail: apichaidan@pim.ac.th, parinyasan@pim.ac.th, choochart.haruechaiyasak@nectec.or.th

***Abstract*—Web content extraction is a process to extract user specified information from web pages. Traditionally, the main approaches of web content extraction have been performed via rule based or pattern based. Typically, rule or pattern set is manually prepared by hand-engineering and can only be applied to each individual web site. To increase the efficiency, we have proposed a machine learning based approach by applying Long Short-Term Memory (LSTM) which is a sequence to sequence learning for dynamic extraction of title and content from web pages. Based on our error analysis, misclassified tokens are considered minority among the total correct sequence. To improve the performance, in this paper we propose a post processing technique by merging predicted tokens with minority tags into the majority one in the token sequence. To evaluate the performance, we use the same data set from our previous work which is a collection of web pages from 10 different Thai web boards such as *Dek-D*, *MThai*, *Sanook* and *Pantip*. The results of our post processing technique helps improve the accuracy up to 99.53%, an improvement of 0.11% from the previous proposed model. The overall improvement may seem little, however, for Title extraction, the accuracy is significantly improved from 88.04% to 100%.**

***Index Terms*— Web Content Extraction, LSTM, Sequence-to-Sequence Learning, Post processing**

I. INTRODUCTION

Web content extraction task is to extract information from web site which is important for business in several fields. At present, amount of social media has rapidly increased because people can simply access to the web including the social media through several devices such as desktop, mobile and tablet. It causes a large number of people to access the social media and web boards. In many web boards, there are many user comments about the products and services, that is very useful for business to tell them about how good or bad of their products and services. These data should be monitored and analyzed for improving the marketing. The information extraction is very important in this part. However, these data are huge and variety that is very hard to manage them manually.

In this paper, our main focus is to dynamically extract information from web pages. Our goal is to propose a web content extraction model which is more general for various web sites with high extraction accuracy.

This paper is organized as follows: Section II describes related work. Section III explains the proposed method for dynamically extract information. Section IV gives the experiments with the performance evaluation of our approach. Section V presents the conclusions.

II. Related work

There are several previous works in this field. Mohammed et al. [1] explored several techniques for information extraction from web pages which can be summarized to five techniques as follows:

1) Wrappers

Wrappers for content extraction is creating rules for extracting particular content from web pages. Baumgartner et al. [2] proposed the technique called *Lixto*. They implement *Lixto* by using wrapper technique in web information extraction. The pattern in a hierarchical order is used to create wrapper for translation from HTML into XML through extraction mechanism. The extraction mechanism is implemented by using both data extraction tree and string extraction. The data extraction tree use rule in program called *Elog* for specifying each element corresponding to tree path of HTML, while the string extraction is used to specify attribute conditions for required string out of tree path.

2) Template

Template detection for extracting is created by algorithms to detect HTML of web page. The content is plugged into the template for content extraction from web pages. Arasu et al. [3] proposed an algorithm, called *EXALG* for automated information extraction

from templates of web pages without preparing new learning examples. The *EXALG* processes in two stages. The first stage makes association each token with the same constructor in unknown templates to use as the input pages. The second stage uses setting of first stage to create template for extracting information from web pages.

3) Machine learning

Machine learning based content extraction is studying of system that can learn from the training data for clustering and classifying data in web content extraction. Soderland et al. [4] Information Extraction (IE presented Open Information Extraction system (OIE) by using *TextRunner*. A self-supervised learner is Naïve Bayes classifier to automatically label tuples for all possible relations which is used in the *Extractor* module. The *Extractor* generates candidate tuples and sends to the classifier. The tuple is assessed by assigning the probability by the *Assessor* for to extract tuples.

4) Visual Cues

Content extraction using visual cues is assumption on the structure of web page for easy extracting content from web pages. Cai et al. [5] proposed an approach by combining the Document Object Model tree (DOM) structure and the visual cues to be the vision-based content structure. Every DOM node is checked against with the visual cues. When all blocks are checked, these blocks are identified weight with *Visual separators* based on properties of its neighbour blocks. After all of blocks are processed, the final gets vision-based content structure for extraction content from the web page.

5) HTML features

Content extraction based on HTML features is extracting content from HTML's tag of web page. Gupta et al. [6] presented an approach by using several techniques from many previous works in content extraction. Their key concept is using the Document Object Model tree (DOM), rather than HTML markup in the web content extraction.

However, all of above techniques are not generalized for many web pages. To solve this problem, the machine learning approach is applied in our work. Finkel et al. [7] proposed a technique based on a sequence model by combining Gibbs sampling and CRF model for extracting information. They use Gibbs sampling to find the most possibly state sequence and then training by CRF model. They evaluate their technique by using the CoNLL NER task and CMU Seminar. Sun et al. [8] used Support Vector Machine (SVMs) in web content extraction task for classification web pages. They use data in WebKB data set. This data set was trained with SVMs and is extracted context features for classifying into four categories, i.e. student, faculty, course and project. Wu et al. [9] proposed an approach in automatic web content extraction by combination of learning and grouping. They apply DOM tree to extract element follow HTML tag and train each node DOM tree with learning model. Then the output from the learning is grouped candidate nodes, the noisy groups are removed and the selected group is refined.

Currently, Neural Networks (NN) is popular in the field of machine learning to solve difficult problems in natural language processing (NLP) task such as chunking, named entity recognition and part of speech tagging. Normally, NN is trained by backpropagation but PSO can be used for training [10]. NN have achieved excellent performance in NLP tasks. Chau et al. [11] proposed an approach to filter web page by applying machine learning-based which combines web content analysis and web structure analysis. They proposed NN-WEB and SVM-WEB which are compared with lexicon-based approach (LEXICON) and keyword-based support vector machine (SVM-WORD). Jagannatha et al. [12] presented technique for extracting text in Electronic Health Record (EHR) notes. They apply machine learning base on recurrent neural network (RNN) frameworks.

In our previous work, we apply recurrent neural network, named Long Short-Term Memory (LSTM), which learns token sequences to make prediction of label sequences in filtering title and content out of HTML for web content extraction. Our result is good for overall but not for the most important information, that is the title of each page.

In this paper, we extend our previous work to improve performance in post processing. Homma et al. [13] proposed an approach by applying hierarchical neural network for extracting information from documents. They use DOM tree to extract information out of HTML tag. Then these extracted data are trained with hierarchical network for classify a sentence.

Obviously HTML tag is a couple of beginning and ending tags which we can adopt this principle to capture HTML tag for improving our approach. Our previous work, the accuracy of title extraction is not good. Therefore, we want to extent our previous work for improve the performance of title extraction. We use the characteristic of web page design that it has only one title in one page. We apply this rule to post processing to correct the title extraction. The post processing will be applied after the LSTM results.

III. The Proposed Method

In previous work, we proposed an approach for web content extraction [14] which consists of three steps, i.e. web crawler, data preprocessing and processing. In this paper, we extend our previous work to improve performance with post processing which overall of our approach is shown in Figure I.

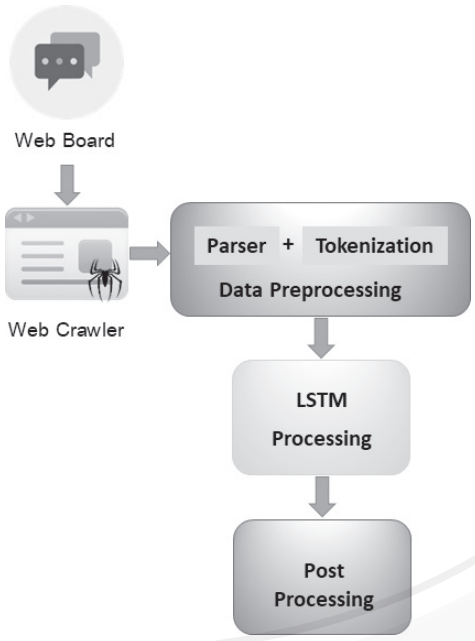

Fig. I. Overview of our approach for generalized information extraction.

A. LSTM Networks

Recurrent Neural Network (RNN) is a neural network that is feed forward general sequence learning. The data sequence is fed forward for learning sequence to sequence between input and output [15] In backward part, RNN maintains historical information for adjusting parameter of network to predict the current output [16] RNN is shown in Figure II In practice, if there are a large number of learning sequences, RNN will not be able to capture long term dependencies [17]. Long Short-term Memory (LSTM) which are invented by Sepp Hochreiter and Urgen Schmidhuber [18] can solve this problem to capture long term dependencies. LSTM is improved from standard RNN which hidden layer of LSTM is replaced with memory block [19] In memory block, it can add or remove data that is controlled by a gate [17]. For this advantage, LSTM can store information in long periods of time and it can avoid the vanishing gradient problem [19].

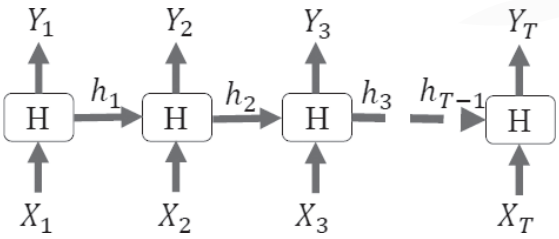

Fig. II. A sample feedforward RNN model.

In previous paper [14], we proposed our model as shown in Figure III.

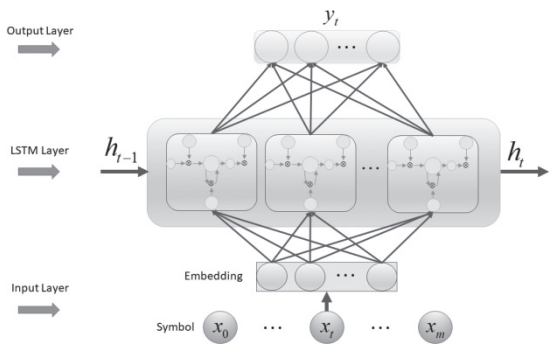

Fig. III. The proposed model for web content extraction.

This LSTM model is applied directional LSTM network by considering only forward direction of input sequence, i.e., input sequence is fed from left to right direction [20] This model has one input layer which the input is fixed dimensionality as maximum length (15,576) of tokens per one sequence. LSTM layer is consist of 128 nodes and using only one output layer by softmax activation function that the number of nodes is equal to the number of labels.

B. Our Proposed Approach

The performance for generalized information extraction relies on design in Figure I. This diagram consists of four steps. The first three steps will be described in this subsection and the last step will be presented in the next subsection.

The first step, we crawl data from the target web boards. We clean HTML tags that are not necessary such as script, style, link, meta and button. The example result is shown in Figure IV.

```
<html>
<head>
</head>
<body >
<h1> This is title  </h1>
<div> Content is here </div>
</body>
</html>
```

Fig. IV. Example of cleaned result.

The second step is data preprocessing. we parse the crawled data into token sequence as shown in Figure V by each text string is parsed and tokenized automatically to achieve word sequence. In Figure VI., we show inserting markers to cover title and content sequences. For title sequence, the opening marker is "<<<T>>>" and the closing marker is "<<</T>>>". For content sequence, the opening marker is "<<<C>>>" and the closing marker is "<<</C>>>".

After that, we automatically insert labels by reading maker to cover word sequence as shown Figure VII. The title token is labeled with T, while the content token is labeled with C and other tokens are labeled with O.

Fig. V. Example of tokenized result.

Fig. VI. Example of marked result.

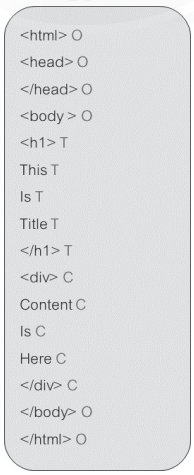

Fig. VII. Example of labeled result.

In the third step, the prepared data is fed in input layer of LSTM model. Each token is mapped into an embedding in embedding layer with 512 of embedding size. Then feed in LSTM layer. After that, the result is computed at output layer by finding highest probability which correspond with label which is shown in Figure VIII.

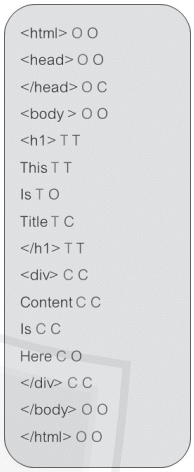

Fig. VIII. Example of classified result.

C. Post Processing

In this paper, the post processing is proposed for improving the performance of our approach. This process is applied after the LSTM classification. The procedure is shown in Figure IX which is consist of four steps as follows:

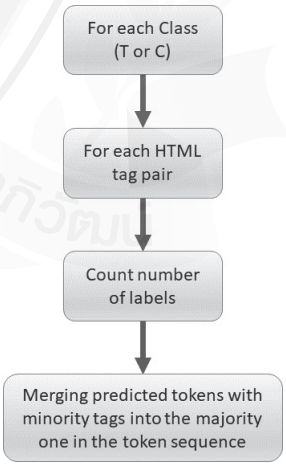

Fig. IX. Post processing procedure.

Firstly, all of tokens, that are classified into title and content, are investigated. By look around the HTML tags that cover them, counting the number of major labels and then reassign this major label to all tokens in this HTML tag.

Example result of the first three steps is shown in Figure X which the second column is target and third

column is answer of system. From this example, the major label in the third column is "T" and they are bounded by HTML tag of <h1>…</h1>.

Fig. X. Example of classified result by the first three steps.

Finally, after the major label is defined that is "T" in this example. All of tokens in detected tag <h1> will be changed to T. That makes the result correct.

Fig. XI. Final result example.

IV. Experimental Results

For our experiment, we crawl data from web page and then parse and tokenize before classification. The labels are title, content and others. We compare our current method with our previous one to show the improvement.

A. Data sets

Data are collected from 10 Thai web board i.e. *Dek-D*[1], *MThai*[2], *Sanook*[3], *Jeban*[4], *Pantip*[5], *Khaosod*[6], *Kaohoon*[7], *Kapook*[8], *Beartai*[9] and *Postjung*[10]. Obviously, these web boards are popular web board in Thailand that can be indicated by the number of posts and number of viewers. Each collected web boards are divided into two sets; a training set of 16 web pages and a test set of 4 web pages. Therefore, we have entire data which consists of 200 web pages. In classification, we classify in three labels i.e. Title, Content and Others. The title token is labeled with T. The content token is labeled with C. The other tokens are labeled with O. Table I shows number of each token in labeled T, C and O on the train and test data sets. The number of samples is different from the numbers in our previous work [14] because we change the positions of parsing and tokenizing of the HTML tag.

TABLE I
Label Statistics of Each Token

Label	Number of Labels	
	Train	Test
T	3,433	853
C	85,865	21,490
O	733,389	181,160

B. Results

We train our training data set with our LSTM model as described in Section III. We create the LSTM model using *Keras* [21] with *Tensorflow* [22] as the back engine. We test our approach with test data set. We compare our previous work that is trained by LSTM only with this approach which is combine with the post processing. Table II shows averaging accuracy on our data sets. We achieved 99.96% in results of training. The results of testing, it improved from 99.42% to 99.53%. The classification results in each label is shown in Table III. In label T, LSTM with post processing achieved the accuracy rate of 100% which is higher than LSTM only (88.04%). And LSTM with post processing achieved the accuracy rate of 97.72%, which is higher than 97.11% of previous one for label C on the test data set. Obviously, the post processing improves all performances, especially for title. That is very useful for business because we can aware what people is interesting now. The classification results of each web boards are shown in table IV. *Dek-D* gets the highest accuracy with scored of 100% because the structure of *Dek-D* is simpler in capturing HTML tag pair than others. *Pantip* gets the lowest accuracy with scored of 98.51% because it has the same pattern for contents and comments.

TABLE II
Average Accuracy (%)

Data set	LSTM	LSTM + Post processing
Test	99.42	99.53

[1] Dek-D, https://www.dek-d.com/
[2] MThai, https://talk.mthai.com/
[3] Sanook, https://news.sanook.com/
[4] Jeban, http://www.jeban.com/
[5] Pantip, https://pantip.com/
[6] Khaosod, https://www.khaosod.co.th/
[7] Kaohoon, https://www.kaohoon.com/
[8] Kapook, https://www.kapook.com/
[9] Beartai, https://www.beartai.com/
[10] Postjung, https://board.postjung.com/

TABLE III
AVERAGE ACCURACY (%) OF EACH LABEL

Label	LSTM	LSTM + Post processing
T	88.04	100
C	97.11	97.72
O	99.75	99.75

TABLE IV
AVERAGE ACCURACY (%) OF EACH WEB BOARD

Web board	LSTM	LSTM + Post processing
Dek-d	99.68	100
MThai	98.98	99.10
Sanook	99.91	99.94
Jeban	99.92	99.92
Pantip	98.42	98.51
Khaosod	99.89	99.89
Kaohoon	99.85	99.85
Kapook	99.06	99.13
Beartai	99.01	99.05
Postjung	99.58	99.85

V. CONCLUSION

In this paper, we presented a generalized information extraction for Thai web boards. We improve the performance from our previous work by including the post processing. Our previous technique achieved the accuracy of 99.42% when processed with post processing, it can achieve the accuracy of 99.53%. In classification of title, we achieved the score of 100% which is the best of result. Moreover, the post processing depends on LSTM classification results because the post processing count number of classified labels before correct them to the major one. For content classification, we achieved the scored of 97.72%, which is a better than the old result. Obviously, the post processing has improved overall performance of our proposed approach. The technique has to use high quality hardware because training set has a large number of data. This is important problem in this research. For future study, data set preparing for training is an important portion for in this field because this is one part of achieved high accuracy.

ACKNOWLEDGMENT

This research is supported by NSTDA University Industry Research Collaboration (NUI-RC) Scholarships of National Science and Technology Development Agency and Panyapiwat Institute of Management. The authors would like to thank CP ALL public company limited for research fund and Mr. Sunchai Booncharoen. And also we gratefully acknowledge the support of NVIDIA Corporation with the donation of the Titan Xp GPU used for this research.

REFERENCES

[1] S. M. Al-Ghuribi and S. Alshomrani, "A comprehensive survey on web content extraction algorithms and techniques," *Int. Conf. Inf. Sci. Appl. ICISA 2013*, Jan. 2013.
[2] R. Baumgartner, R. Baumgartner, S. Flesca, G. Gottlob, S. Flesca, and G. Gottlob, "Visual web information extraction with lixto," *Proc. Int. Conf. Very Large Data Bases*, 2001, pp. 119-128.
[3] A. Arasu, H. Garcia-Molina, A. Arasu, and H. Garcia-Molina, "Extracting structured data from Web pages," *ACM SIGMOD Int. Conf. Manag. Data*, 2003, pp. 337-348.
[4] S. Soderland, M. Broadhead, M. Banko, M. J. Cafarella, and O. Etzioni, "Open information extraction from the web," *Int. Jt. Conf. Artif. Intell*., 2007, pp. 2670-2676.
[5] D. Cai, S. Yu, J.-R. Wen, and W.-Y. Ma, "Extracting content structure for web pages based on visual representation," *Proc. 5th Asia-Pacific web Conf. Web Technol. Appl.*, 2003, pp. 406-417.
[6] S. Gupta, G. Kaiser, D. Neistadt, and P. Grimm, "DOM-based content extraction of HTML documents," *Proc. twelfth Int. Conf. World Wide Web WWW 03*, 2003, pp. 207.
[7] J. R. Finkel, T. Grenager, and C. Manning, "Incorporating non-local information into information extraction systems by gibbs sampling," *Acl*, no. 1995, pp. 363-370, 2005.
[8] A. Sun, E.-P. Lim, and W.-K. Ng, "Web classification using support vector machine," *Proc. fourth Int. Work. Web Inf. data Manag. WIDM 02*, vol. 78, pp. 96-99, Apr. 2002.
[9] S. Wu, J. Liu, and J. Fan, "Automatic Web Content Extraction by Combination of Learning and Grouping," *Proc. 24th Int. Conf. World Wide Web–WWW'15*, pp. 1264-1274, 2015.
[10] C. Jeenanunta and K. D. Abeyrathn, "Combine Particle Swarm Optimization with Artificial Neural Networks for Short-Term Load Forecasting," *Int. Sci. J. Eng. Technol.*, vol. 1, no. 1, pp. 25-30, 2017.
[11] M. Chau and H. Chen, "A machine learning approach to web page filtering using content and structure analysis," *Decis. Support Syst.*, vol. 44, no. 2, pp. 482-494, 2008.
[12] A. N. Jagannatha and H. Yu, "Bidirectional RNN for Medical Event Detection in Electronic Health Records," *Naacl2016*, pp. 473-482, 2016.
[13] Y. Homma, K. Sadamitsu, and K. Nishida, *A Hierarchical Neural Network for Information Extraction of Product Attribute and Condition Sentences,* pp. 21–29.
[14] A. Dangmadee, P. Sanguansat, and C. Haruechaiyasak, "Web Content Extraction Using LSTM Networks," *2017 2nd Int. Conf. Sci. Technol.*, 2017.
[15] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to Sequence Learning with Neural Networks," pp. 1-9, 2014.
[16] Z. Huang, W. Xu, and K. Yu, *Bidirectional LSTM-CRF Models for Sequence Tagging,* 2015.
[17] G. Lample, M. Ballesteros, S. Subramanian, K. Kawakami, and C. Dyer, *Neural Architectures for Named Entity Recognition,* 2016.
[18] S. Hochreiter and J. Urgen Schmidhuber, "Long Short-Term Memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735-1780, 1997.
[19] P. Wang, Y. Qian, F. K. Soong, L. He, and H. Zhao, *A Unified Tagging Solution: Bidirectional LSTM Recurrent Neural Network with Word Embedding,* 2015.
[20] H. Zen and H. Sak, "Unidirectional long short-term memory recurrent neural network with recurrent output layer for low-latency speech synthesis," *ICASSP, IEEE Int. Conf. Acoust. Speech Signal Process. Proc.*, pp. 4470-4474, Aug. 2015.
[21] F. Chollet, "keras," *GitHub repository. GitHub*, 2015.
[22] Martin Abadi *et al.*, *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems.* 2015.

Apichai Dangmadee received his B. Eng. degree from the Department of Electronic and Computer System Engineering, University of Silpakorn, in 2015. He got an experience National Electronics and Computer Technology Center (NECTEC), National Science and Technology Development Agency (NSTDA) in Module development system backup DICOM and Raw format and also have demonstrated significant results in the field of application development by JAVA, Create Database by MySQL. His research interest is Machine Learning and Natural Language Processing.

Parinya Sanguansat is associate professor in electrical engineering and head of computer engineering at the Panyapiwat Institute of Management (PIM), Thailand. He graduated B.Eng., M.Eng. and Ph.D. in electrical engineering from Chulalongkorn University. He has more than ten years of experience including Machine Learning, Image processing and Computer Vision. He is the consultant at Tellvoice Technology Limited for Natural Language Processing and Chatbot framework. He got many research grants from both private and public organization including CP ALL public company limited and The Thai Research Fund. He published many research in IEEE and others. He has written several books about Machine Learning and MATLAB programming.

Choochart Haruechaiyasak received his Ph.D. degree from the Department of Electrical and Computer Engineering, University of Miami, in 2003. After receiving his degree, he has worked as a researcher at the National Electronics and Computer Technology Center (NECTEC), National Science and Technology Development Agency (NSTDA). His current research interests include data and text mining, natural language processing, information retrieval, and data management. One of his objectives is to promote R&D works through IT applications for government offices and business sectors in Thailand. He is also a visiting lecturer at several universities in Thailand.