Comparison of Keywords Extraction Techniques in Kickstarter and Indiegogo Projects

Woottikarn Hongwiengchan[1] and Jian Qu[2]

[1,2]Faculty of Engineering and Technology, Panyapiwat Institute of Management, Nonthaburi, Thailand
E-mail: Woottikarnhon@ pim.ac.th, jianqu@pim.ac.th

Received: October 28, 2021/ Revised: January 1, 2022/ Accepted: February 21, 2022

***Abstract*—There are many fake projects on Kickstarter and Indiegogo, and they are usually hard to distinguish from real projects. This research is a pioneer study to try to find a way for helping humans to identify possible fake projects. We propose to extract keywords from the projects, the extracted keywords would give the user a better understanding of the project. We compared keyword extraction for crowdfunding projects by using RAKE, NLTK, LIAAD/YAKE, BERT, and Gensim models. We measured the keyword extraction performance of each model using the precision, recall, and F1 scores. According to the results, the NLTK model is the most efficient, with a precision of 54.40% and an F1 of 70.47%.**

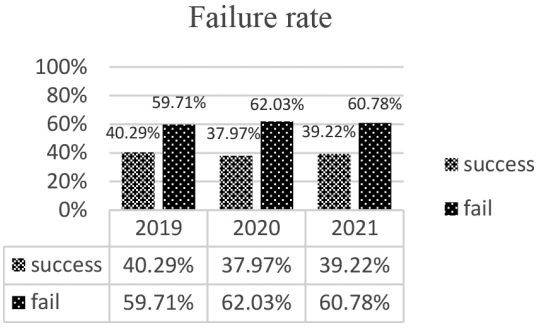
***Index Terms*—Kickstarter Projects, Indiegogo Projects, NLTK, Keyword Extraction**

I. Introduction

There are many projects on the official Kickstarter and Indiegogo websites, and more projects are being added each day. Some people use these projects to deceive others into investing. Even the project has no way of being able to succeed. According to Kickstarter's latest statistics, as of 2021, the average project failure and investor fraud rate is 60.78%, and the project success rate is only 39.22%. No published works are currently focused on research investigating fake projects on Kickstarter and Indiegogo because it is very difficult to detect and determine if they are fake or fraudulent projects that cannot be fulfilled. It is necessary to use relevant professional knowledge to examine and understand the project to determine the likelihood of success rates.

The data in Fig. 1 shows based on the statistics for 2019[1], 2020[2], and 2021[3]. We have found that the success rate of Kickstarter projects is low and the failure rate is higher than the project completion rate.

To reduce the number of scams from fake projects, we wanted to understand and learn more about the project by extracting keywords from project articles on Kickstarter and Indiegogo. Keyword extraction is the process of collecting words and phrases from the text. Generally, a keyword is a word that contains important information or the text that is the essence of the article, while some article keywords can be single words or consist of multiple words [1]. Keyword retrieval is a very complicated task. In Natural Language Processing (NLP) research, there are many methods of extracting core principles. Therefore, in this research, we try to find the most optimal way to extract keywords for Kickstarter and Indiegogo projects. The keywords will be used to find more information like theoretical principles that correspond to or support that the work in a project can be made and it really works. Thus, investors will make better decisions.



	2019	2020	2021
success	40.29%	37.97%	39.22%
fail	59.71%	62.03%	60.78%

Fig. 1. The failure rate of Kickstarter projects

II. Related Studies

When we read an article, sometimes we only want to know the main idea or the article's essence. Reading every text in the article will take quite a long time to understand the content, sometimes we are buried by information from many texts and loss the focus on the main ideas of the article. Therefore, by extracting

[1] https://www.shorturl.asia/wlv0n (last accessed on 15 October 2021)
[2] https://www.crowdcrux.com/kickstarter-statistics-in-2020/(last accessed on 15 December 2021)
[3] https://www.kickstarter.com/help/stats (last accessed on 15 December 2021)

keywords from the article with techniques in NLP, we can easily understand the main idea faster. There are many techniques used for keyword extraction in NLP. We have studied and applied five techniques for research: NLTK [2], RAKE [3], Gensim [4], YAKE [5], and BERT [6]. All five techniques are highly efficient keyword extraction algorithms and the library of each algorithm can be easily imported for use on Python platforms. Each technique uses a different algorithm. For example, RAKE has a list of stop words, phrase separators, and word separator sets. While RAKE is extracting keywords from the text when the system detects a match for the stop word list. Those words are excluded because they are considered meaningless. Words that are considered to carry meanings are associated with the text being described as bearing content and referred to as content speech. The input parameters section for the RAKE algorithm includes a list of stop words and a set of phrase separators and word separators. Use stops word and phrase separators to separate text from a document into important words or sentences. Most of these optional keywords help researchers to isolate the exact keywords needed to get information from the document. But no matter what technique, most N-gram, TF-IDF [7] word frequencies are used as statistical properties in keyword selection. Word frequency (TF) and inverse document frequency (IDF) are used to evaluate the effectiveness and importance of keywords in a document. After the assessment, TF-IDF removes any irrelevant or unsuitable words for keywords. NLP researcher Onan et al. proposed a machine-learning method and additional statistical properties for keyword separation [8]. Jian Qu et al. also proposed a method that combines statistical properties and adaptive rules to separate keywords [9]

Another technique widely used among researchers is KeyBERT, an easy-to-use keyword extraction technique. KeyBERT is a keyword extraction library from BERT embedding to get keywords that are most representative. KeyBERT has recently been proposed as a replacement for Word2Vec [10]. The concept of word2vec is to create a classifier to predict whether the context word which is likely to occur close to the target word selected. After that, Word2Vec takes the weights obtained from the classifier as word embedding, but BERT is different because it has its model and depends on the transformer library [11]. BERT operates in two steps including pre-training and tuning during pre-training. First, BERT models are trained without data labels. For customizing the BERT model start with pre-trained parameters and all parameters are fine-tuned using labeled data from the destination application. A distinctive feature of BERT is its unified architecture for tasks. Therefore, there is a difference between pre-training architecture and final downstream architecture.

However, all techniques used to extract words still need to be optimized and compared to human word extraction. Overall, the efficiency of automatic word extraction for English data of each technique is quite good, but it can also improve the accuracy and keyword selection criteria used to separate keywords from articles.

III. Our Approach

For this research, we used information from the official website of Indiegogo and Kickstarter. We select projects from a high investment amount of 150 projects, divided into 50 completed projects and 100 unsuccessful projects to learn about fake project information. We choose the number of unsuccessful projects over successful ones because there are many types of unsuccessful projects. For example, a project that can be accomplished has a theory to support its creation but suddenly stops, and projects do not continue. Some projects can create pieces and have a theory to support them, but they can't be used in real work. We will use the information of each project obtained from the search on the website for keyword extraction from multiple techniques to get good keywords and suitable ones. We applied five models, NLTK [2], RAKE [3], Gensim [4], LIAAD/YAKE [5], BERT [6], and we then compare them with the keyword extracted by ten students, nine of them are undergraduates and one graduate.

A. NLTK

NLTK (Natural Language Toolkit) [2] is an open-source software and API for NLP (Natural Language Processing) platform available for Python. It is a powerful tool for processing text data, parsing, classification, interception, tagging, semantic reasoning, and other computational linguistics. Installing the NLTK Module requires downloading and installing an additional bundle that will download dictionaries and other language and grammar frameworks required for full NLTK functionality. NLTK fully supports English.

Stop words do not have a clear meaning like 'and', 'a', 'it', and 'they' these have a meaningful impact when we use them to communicate, but these terms may mean nothing with computer analysis keyword extraction. Stop words are left in the entire data system and not included in NLTK text analyses because it is considered meaningless. Sometimes, using the NLTK module to extract words from informal articles, for example, from the internet, might be a problem. Therefore, we need to train these models on new datasets with informal languages first.

B. RAKE

RAKE [3], also known as Rapid Automatic Keyword Extraction, is a highly effective keyword extraction technique.

RAKE is based on the observation that common keywords consist of multiple words with standard punctuation marks or pauses. Alternatively, we can say interchangeable words like 'and', 'of', and 'that'. First, the text in the document is split into an array of words separated by specific words. Secondly, the array is divided again into successive sequences of phrase-separated words and stop-word positions. Finally, the word location is determined. Then, we will combine words in one location and select them as keywords.

C. Gensim

Gensim [4] consists of several sub-technical extraction techniques such as Word2Vec, FastText, and Latent Semantic Indexing. Gensim automatically searches for keywords by examining statistically occurring patterns within its own dictionary. Furthermore, the techniques within Gensim are unsupervised. This means that as long as we have the dictionary, we can easily apply it to other texts.

D. YAKE/ LIAAD

YAKE [5] is a way to extract keywords based on text attributes automatically. It is a keyword extraction tool in Pyth. This method also provides an end-to-end keyphrase extraction pipeline to extract keywords from text documents. It uses a statistical text feature extracted from a single document to select the most important text keywords. YAKE's systems do not require training on a specific set of documents, language, and domain size.

E. BERT

BERT [6] uses a mechanism of relational learning between words of text in the form of a transformer. The details of the operation of the Transformer are described by Google [12].

The core principle of BERT is in Transformers, a new neural network architecture for linguistic understanding.

Transformers are word-processing models that involve other words in a sentence. Instead of verbatim processing, BERT can determine the full context of a word by looking at the words that come before and after them. It's especially useful for understanding the intent behind a search query. Fig. 2 shows the procedure of extracting keywords with BERT pre-training.

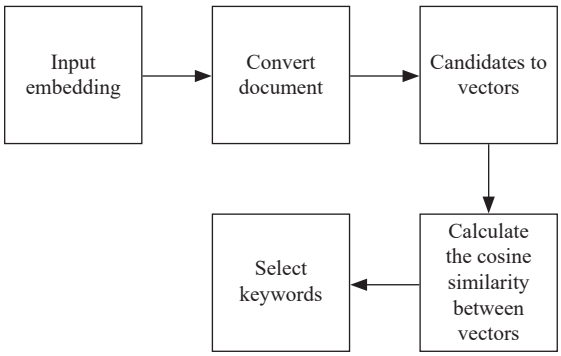


Fig. 2. Procedure of extracting keywords with BERT pre-training

F. Compare Keywords

In this research, we will manually generate five keywords to create a ground truth equal to the number of automatic keywords obtained from each model to compare with automatic keyword extraction. Human-generated keywords are done by ten students, including one undergraduate and one graduate. We analyzed and selected keywords based on their meaning, main content of the project, nature of work, type of work, and benefits.

Table I shows an example of a keyword extraction with ten students using the method of reading from each project article. If a student reads and does not understand the content, he or she will go find more information regarding the project, whether it's through the website, books, or others to understand the content or the algorithm that is required for decision-making and selection of keywords. We have a keyword extraction process as follows: First, we had nine undergraduate students read the article and select the right word or benefit for the project. In the second step, one graduate will compare all the keywords received from the nine undergraduate students, and in the last step, one graduate will select five suitable keywords. Keywords selection is primarily based on the keywords that are the essence of the content.

TABLE I
AN EXAMPLE OF MANUAL KEYWORDS EXTRACTION

Number	Manual Keyword
1	Wheel Reflector
2	Reflector With 360-degree
3	Reflective Power
4	Overlapping Design
5	Gapless Shining

In Fig. 3 shows a flow chart of the sequence of steps of keyword extraction with five techniques and Table II to Table VI shows an example of a keyword extraction by five techniques: NLTK, RAKE, YAKE/ LIAAD, Gensim, and BERT.

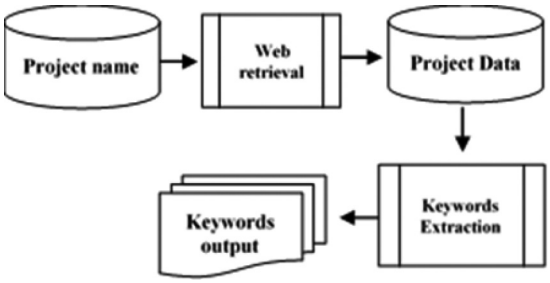


Fig. 3. Flow chart of extracting keywords by five techniques

TABLE II
AN EXAMPLE OF KEYWORDS EXTRACTION WITH RAKE

Original Text	FLECTR 360 OMNI – Edition 2020. The wheel reflector with 360-degree visibility! FLECTR 360 reflects car headlights wherever they come from. Sideways, from behind or in front – simply from ANY direction! It magically turns your rims into gapless shining retro-reflectors. Its overlapping design also doubles its reflective power. FLECTR 360 OMNI fits oval, round & sharp-edged rims from 26" and above with a maximum of 32 spokes. Exclusion: caliper brake wheels without a minimum curvature of 15 mm above the brake flank.
Output	**"Calliper brake wheels",** "Flectr 360 omni fits oval", "Round & sharp-edged rims", "Flectr 360 reflects car headlights", "Gapless shining retro-reflectors"

TABLE III
AN EXAMPLE OF KEYWORDS EXTRACTION WITH YAKE/ LIAAD

Original Text	FLECTR 360 OMNI – Edition 2020. The wheel reflector with 360-degree visibility! FLECTR 360 reflects car headlights wherever they come from. Sideways, from behind or in front – simply from ANY direction! It magically turns your rims into gapless shining retro-reflectors. Its overlapping design also doubles its reflective power. FLECTR 360 OMNI fits oval, round & sharp-edged rims from 26" and above with a maximum of 32 spokes. Exclusion: caliper brake wheels without a minimum curvature of 15 mm above the brake flank.
Output	**"Feflector 360 reflects car headlights",** "Feflector 360 reflects car headlights come", "Omni edition 2020 wheel reflector 360 degrees", "Degree visibility reflector 360 reflects car headlights", "Visibility reflector 360 reflects car headlights come"

TABLE IV
AN EXAMPLE OF KEYWORDS EXTRACTION WITH GENSIM

Original Text	FLECTR 360 OMNI – Edition 2020. The wheel reflector with 360-degree visibility! FLECTR 360 reflects car headlights wherever they come from. Sideways, from behind or in front – simply from ANY direction! It magically turns your rims into gapless shining retro-reflectors. Its overlapping design also doubles its reflective power. FLECTR 360 OMNI fits oval, round & sharp-edged rims from 26" and above with a maximum of 32 spokes. Exclusion: caliper brake wheels without a minimum curvature of 15 mm above the brake flank.
Output	**"Reflects",** "Brake", "Reflective", "Sharp", "Shining"

TABLE V
AN EXAMPLE OF KEYWORDS EXTRACTION WITH BERT

Original Text	FLECTR 360 OMNI – Edition 2020. The wheel Reflector with 360-degree visibility! FLECTR 360 reflects car headlights wherever they come from. Sideways, from behind or in front – simply from ANY direction! It magically turns your rims into gapless shining retro-reflectors. Its overlapping design also doubles its reflective power. FLECTR 360 OMNI fits oval, round & sharp-edged rims from 26" and above with a maximum of 32 spokes. Exclusion: caliper brake wheels without a minimum curvature of 15 mm above the brake flank.
Output	**"FLECTR",** "Edition", "OMNI", "Degree visibility", "Wheel reflector"

TABLE VI
AN EXAMPLE OF KEYWORDS EXTRACTION WITH NLTK

Original Text	FLECTR 360 OMNI – Edition 2020. The wheel reflector with 360-degree visibility! FLECTR 360 reflects car headlights wherever they come from. Sideways, from behind or in front – simply from ANY direction! It magically turns your rims into gapless shining retro-reflectors. Its overlapping design also doubles its reflective power. FLECTR 360 OMNI fits oval, round & sharp-edged rims from 26" and above with a maximum of 32 spokes. Exclusion: caliper brake wheels without a minimum curvature of 15 mm above the brake flank.
Output	**"Flectr 360 reflects car headlights wherever",** "Flectr 360 omni – edition 2020", "Flectr 360 omni fits oval", "Overlapping design also doubles", "Calliper brake wheels without"

G. Matching Keywords

Keyword matching is where the keywords obtained from five algorithms are compared with the keywords made by ten students. Using the following conditions: a keyword that matches 100% manual keyword will get one score point; if it matches 75%, it will get 0.75 points; if it matches 50%, it will get 0.50 points; if it matches 25%, it will get 0.25 points.

TABLE VII
AN EXAMPLE OF MATCHING KEYWORDS

Number	Manual Keyword	NLTK
1	Wheel Reflector	"Flectr 360 Reflects Car Headlights Wherever"
2	Reflector With 360-degree	"Flectr 360 Omni – Edition 2020",
3	Reflective Power	"Flectr 360 Omni Fits Oval",
4	Overlapping Design	"Overlapping Design Also Doubles",
5	Caliper Brake Wheels	"Caliper Brake Wheels Without"

From Table VII the result of the keyword matching is 2 points. There are two valid keywords: "overlapping design" and "caliper brake wheels".

H. Precision and Recall

To compare each model that was searched for keywords, we will use precision and recall as a measure. Precision is the number of keywords that match the keywords we generate divided by the total number of keywords we generate. The recall is the total number of keywords coming out of each model divided by the number of keywords we define.

$$recall = \frac{|\{Manual\ keyword\} \cup \{retrieved\ keyword\}|}{\{Manual\ \ keyword\}} \quad (1)$$

A recall is a manually generated fraction of a keyword successfully retrieved.

$$precision = \frac{|\{Manual\ \ keyword\} \cap \{retrieved\ keyword\}|}{\{retrieved\ keyword\}} \quad (2)$$

Precision is the ratio of Manual keywords to retrieved keywords.

$$F - score = \frac{2 * precision * recall}{(precision + recall)} \quad (3)$$

Traditional F measurements are harmonic that combine precision and recall.

IV. Results and Discussion

A. Results

In this research, keywords are automatically obtained by extracting keywords from various modeling algorithms and are compared with the keywords given by the 10 students, with nine undergraduate students reading the paper and choosing the right or useful word for the project, then one graduate compares the keywords all received from nine undergraduate students and five of the appropriate keywords were selected.

We compare it with the keywords obtained from each algorithm. The data in Table VIII shows the total number of keywords retrieved from each algorithm number of valid keywords and the number of invalid keywords.

TABLE VIII
ALL RETRIEVED KEYWORDS, VALID KEYWORDS, AND INVALID KEYWORDS OF DIFFERENT MODEL

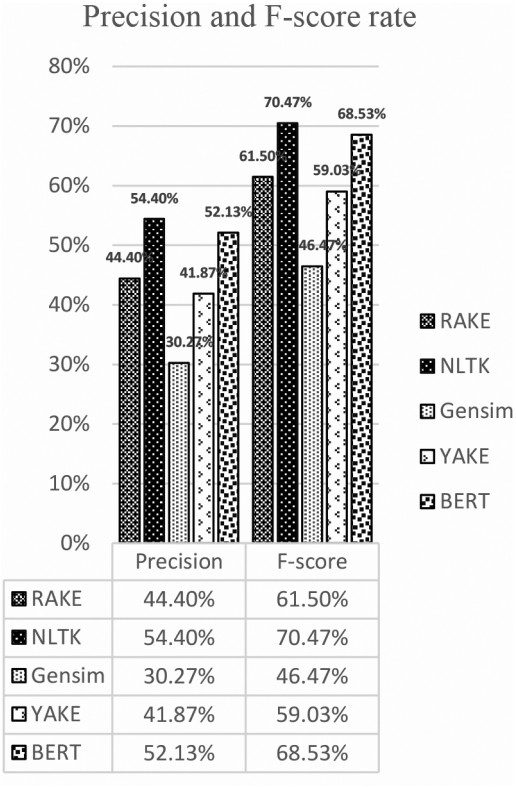
List	All Keywords	Valid Keyword	Invalid Keyword
RAKE	750	333	417
NLTK	750	408	342
YAKE/LIAAD	750	314	436
Gensim	750	227	523
BERT	750	391	359

The data in Table IX shows the results the precision, recall and F-score of different model: RAKE, NLTK, YAKE/LIAAD, Gensim and BERT.

TABLE IX
THE PRECISION, RECALL, AND F-SCORE OF DIFFERENT MODEL.

List	Precision	Recall	F-score
RAKE	44.40%	100%	61.50%
NLTK	54.40%	100%	70.47%
YAKE/LIAAD	41.87%	100%	59.03%
Gensim	30.27%	100%	46.47%
BERT	52.13%	100%	68.53%

The data in Fig. 4 shows the results precision, and F-score rate of different models: RAKE, NLTK, YAKE/LIAAD, Gensim, and BERT.



	Precision	F-score
RAKE	44.40%	61.50%
NLTK	54.40%	70.47%
Gensim	30.27%	46.47%
YAKE	41.87%	59.03%
BERT	52.13%	68.53%

Fig. 4. The Precision and F-Score of Different Methods

B. Discussions

When we applied five algorithms: RAKE, NLTK, LIAAD/YAKE, GENSIM, and BERT, to extract keywords from a total of 150 projects. We studied and compared five techniques to find the model that extracts the best keywords, we also studied the behavior of keyword extraction in each model and compared all the keywords extracted in the same model. We have found the keyword extraction characteristics of each model: RAKE model the extracted keywords will focus on the topic, traits, and abilities mentioned by the article. NLTK model will focus on the key traits, components, abilities, models, or versions. LIAAD/YAKE model focuses on the capabilities and specific words of the content. The gensim model will focus on a single word that describes abilities or traits, and finally, the BERT model will focus on thematic content. It will extract the keywords into long sentences. It discusses topics, potentials, and sentences that describe the nature of the equipment or objects the project will create.

V. Conclusion and Future Work

There are many projects on Indiegogo and Kickstarter. Unfortunately, it is not easy to identify unsuccessful projects. To identify possible fake information in each project, we need to isolate the keywords for each project and further analyze the data for each project. Currently, there are several ways to extract keywords from the English text. Therefore, we have compared the effectiveness of keyword extraction methods for various Indiegogo and Kickstarter projects through experiments in this research.

To get information about the various factors of the fake project. So we take the data of 150 projects for experimentation. There are both successful and unsuccessful projects. In our experiment, we compared the keywords extracted from ten students to the keywords extracted from each algorithm to find an algorithm that extracts the keywords closest to the students. We found the NLTK model to be more efficient than other keyword extraction methods. It has a precision rate of 54.40% and a 70.47% F score, with BERT coming in second with a precision rate of 52.13% and a 68.53% F score.

The extracted keywords will make it easier for investors to search for more information, learn about the details and theories that support the project, and help make investment decisions easier. When analyzing the information obtained, we can determine that the project is likely to be successful. Such as Project Titan. As we read about the project, we will focus on artificial gill technology and size. When we take this information into further research, we can see that in theory it can actually be done. However, due to the small size of the artificial gum, it cannot be used for artificial respiration for 45 minutes, so this project is a fake project. The method of separating keywords from project data is the solution to help investors not be scammed into investing in fake projects.

This research is a pioneer study to try to find a way for helping humans to identify possible fake projects, In the future, we will improve the model and train algorithms to improve the efficiency of keyword extraction and we may extend this research by creating modules to examine projects on Kickstarter and Indiegogo for identifying possible fake projects.

Acknowledgements

The first author studied a keyword extraction algorithm, experimented, and co-drafted the manuscript. The last author gave advice, suggested an experimental method, and co-drafted the manuscript. The first and second authors contribute 50% equally to this work.

The first author would like to thank the scholarship sponsor from CP ALL.

References

[1] F. Bayatmakou, A. Ahmadi, and A. Mohebi, "Automatic Query-Based Keyword and Keyphrase Extraction," in *Proc. 2017 Artificial Intelligence and Signal Processing Conference*, 2017, pp. 325-330.

[2] E. Loper and S. Bird, "NLTK: The Natural Language Toolkit," in *Proc. The ACL Interactive Poster and Demonstration Sessions*, 2020, pp. 62-69.

[3] S. Rose, D. Engel, and N. Cramer, *Automatic Keyword Extraction from Individual Documents*. John Wiley & Sons, Inc., New Jersey, 2010, pp. 1-20.

[4] M. M. Haider, M. A. Hossin, H. R. Mahi et al., "Automatic Text Summarization Using Gensim Word2Vec and K-Means Clustering Algorithm," in *Proc. IEEE Region 10 Symposium*, 2020, pp. 283-286.
[5] R. Campos, V. Mangaravite, A. Pasquali et al., "YAKE! Keyword Extraction from Single Documents using Multiple Local Features," *Information Sciences*, vol. 509, pp. 257-289. Jan. 2020.
[6] N. Karacapilidis, N. Kanakaris, and N. Giarelis, "A Comparative Assessment of State-of-the-Art Methods for Multilingual Unsupervised Keyphrase Extraction," in *Proc. IFIP International Conference on Artificial Intelligence Applications and Innovations*, 2021, pp. 635-645.
[7] C. Zhang, "Automatic Keyword Extraction from Documents using Conditional Random Fields," *Journal of Computational Information Systems*, vol. 4, no. 3, pp. 1169-1180, Jun. 2008.
[8] A. Onan, S. Korukoğlu, and H. Bulut, "Ensemble of Keyword Extraction Methods and Classifiers in Text Classification," *Expert Systems with Applications*, vol. 57, pp. 232-247, Sep. 2006.
[9] J. Qu, NL. Minh, A. Shimazu, "Web Based English-Chinese OOV Term Translation Using Adaptive Rules and Recursive Feature Selection," in *Proc. The 25th Pacific Asia Conference on Language, Information and Computation*, 2011, pp. 1-9.
[10] T. Mikolov, K. Chen, G. Corrado et al., "Efficient Estimation of Word Representations in Vector Space," *arXiv*, no. 1301.3781. pp. 1-10. Jan. 2013.
[11] J. Devlin, M. W. Chang, K. Lee et al., "BERT Pre-training of Deep Bidirectional Transformers for Language Understanding," *arXiv,* no. 1810.04805v2, pp. 1-16. May. 2019.
[12] A. Vaswani, N. Shazeer, N. Parmar et al., "Attention is All You Need," in *Proc. Advances in Neural Information Processing Systems*," 2017, pp. 5998-6008.

Woottikarn Hongwiengchan is currently studying for the Master degree of Engineering (Engineering and Technology), Panyapiwat Institute of Management, Thailand. Him research interests are natural language processing and AI.

Jian Qu is a full-time lecturer at the Faculty of Engineering and Technology, Panyapiwat Institute of Management. He received Ph.D. with Outstanding Performance award from Japan Advanced Institute of Science and Technology, Japan, in 2013. He received B.B.A with Summa Cum Laude honors from the Institute of International Studies of Ramkhamhaeng University, Thailand, in 2006, and M.S.I.T from Sirindhorn International Institute of Technology, Thammasat University, Thailand, in 2010. His research interests are natural language processing, intelligent algorithms, machine learning, machine translation, information retrieval, and image processing.