Analyzing User Reviews in Thai Language toward Aspects in Mobile Applications

Abstract—**As more and more Thais own mobile devices, mobile applications are high in demand. Before installing mobile applications, many users read reviews written by other users to determine whether or not the application is worth using. In addition, mobile application developers also rely on user reviews to get insight information on which aspects of the mobile application users like or do not like and why. They can use the information to market the beloved aspects of their software product and improve on the problematic ones. However, when there are many reviews, it is difficult to comprehend information in the user reviews. Several researches in recent years aim to extract opinions and sentiments from various texts or documents such as Twitter, webboards, and software product reviews. Most of these researches are for English documents. For Thai language, researches usually focus on other contexts such as hotel reviews or general opinions on Twitter. In this paper, we present an approach to analyze user reviews written in Thai based on techniques in natural language processing, topic modeling, and sentiment analysis. The approach aims to help Thai users and developers discover dynamically, instead of pre-determined, various aspects and associated sentiments from a vast amount of user reviews. The result of the approach is a list of aspects with associated opinions and sentiments to help users assess mobile applications and provide summarized user feedbacks for developers.**

I. Introduction

Electronic devices especially smart phones and tablets are used so commonly nowadays that they could have been parts of our bodies. Mobile applications are therefore high in demand and have been developed increasingly. To make mobile applications desirable, not only must developers follow their own visions, but they should also listen to user feedbacks to understand what users want in mobile applications.

By listening to user feedbacks and improving mobile applications accordingly, resulting applications would suit users better and even attract more users, otherwise users might stop using applications and in the end, there might be no one using the applications at all. According to the survey [1] that asked software engineers to rate and identify important questions about software engineering practices, the second most ”essential” question software engineers wanted to know was ”What parts of a software product are most used and/or loved by customers?” The answer to this question would be very useful, so developers can add, remove, or improve their application features to ultimately satisfy user needs.

One way to answer this question is to analyze user comments or reviews in application stores such as Apple App Store or Google Play Store. Users usually write reviews to praise or complain about mobile applications allowing other users to assess quality of the applications and enabling developers to get user feedbacks and improve application features.

However, with so many user reviews, it is difficult or takes too much time to comprehend what users feel about mobile applications. Some comments might not be informative. For example, comments such as ”ดี” (meaning ”good”) cannot tell specifically which aspect of the application is good. In addition, product rating cannot tell all the stories; it can only provide overall application preference but cannot give details which aspects users like or do not like.

This paper therefore aims to analyze user reviews written in Thai to automatically extract aspects from the reviews and perform sentiment analysis to reveal which features users like or do not like. The approach is based on natural language processing, topic modeling, and sentiment analysis techniques. We automatically collect user reviews from Google Play Store using a JavaScript script and pre-process these raw user reviews with word segmentation and part-of-speech tagging tools. Next, the sentiment analysis and topic modeling are performed to discover aspects and associated opinions and sentiments. The approach is then evaluated with precision, recall, F-measure, and accuracy.

The remaining of this paper is organized as follows. Section II describes related work in more details. Section III lays out theories needed for review analysis. Section IV presents our approach. Secton V discusses our results, limitations, and future works. Section VI concludes our paper.

II. Related Work

There are a number of researches on opinion mining and sentiment analysis related to our work. This section describes two groups of researches. The first group is on opinion mining and sentiment analysis of user reviews of mobile applications. The second group is on mining of Thai texts.

1) Research on opinion mining and sentiment analysis of user reviews of mobile applications: Various researches are in this group. Most of them analyzed user reviews written in English on mobile applications. We will discuss three of these works [2]–[4]. Another work on evaluating software quality in use from user reviews [5] is also discussed here.

Chen et al. [2] presented an AR-Miner framework which filtered reviews and extracted only informative reviews using a classification technique called Expectation Maximization for Naive Bays (EMNB) [6], which is a semi-supervised machine learning algorithm. The resulting informative reviews were then grouped into topics. The work applied two topic modeling techniques, Latent Dirichlet Allocation (LDA) [7] and Aspect and Sentiment Unification Model (ASUM) [8], and compared results of these techniques.

Gunzmam and Laalej [3] analyzed user reviews to identify features and extract their associated sentiments. Identifying features were done by finding expressions of two or more words that commonly occurred together such as <*view picture*>, or <*user interface*>. User sentiments were analyzed using SentiStrength [9] for lexical sentiment extraction. Finally, LDA was used to group various features into coherent topics. The approach was evaluated with precision and recall where the truth set was created by the two authors and other seven trained coders who manually analyzed user reviews.

Minh et al. [4] presented a semi-automated framework for mining user reviews when given keywords specified by analysts. The approach can also automatically extract keywords from nouns and verbs in user reviews. The keywords were clustered and expanded and then used to search for relevant reviews hopefully containing useful opinions. The evaluation was done by comparing analysis results with eight researchers.

Leopairote et al. [5] evaluated software quality in use by performing opinion mining of user reviews. Their approach constructed an ontology from the quality in use model, which is one of software quality models in ISO 9126, consisting of 4 characteristics: effectiveness, productivity, safety, satisfaction. Sentences in user reviews were then manually matched with terminologies defined in the ontology. To classify polarity of sentences into positive, negative, or neutral, the approach used sentences labeled to be pros and cons in reviews and also used two lists of sentiment words to construct rule-based classifiers.

This paper is similar to these researches, especially Gunzmam and Laalej's work, since we aim to analyze user reviews of mobile applications to extract aspects and user sentiments about them. These researches however analyzed user reviews written in English whereas our work focuses on user reviews written in Thai. Processing Thai texts is more difficult since Thai texts need to be segmented into sentences and words. In addition, there is no lexical sentiment resource like SentiStrength for Thai language.

2) Research on mining of Thai texts: Several researches have done works on processing and mining Thai texts for various purposes. Three works are discussed here.

Inrak and Sinthupinyo [10] applied latent semantic analysis (LSA) [11] to classify Thai texts from internet such as emails and blogs into six emotions: anger, disgust, fear, happiness, sadness, and surprise. They used SWATH [12] for word segmentation and ORCHID [13] corpus to tag parts of speech.

In 2010, Haruechaiyasak et al. [14] did opinion mining and sentiment analysis of hotel reviews written in Thai based on pre-determined features such as breakfast or service. From these features, related lexicons and a set of syntactic rules based on frequently occurred patterns were created to mine opinions and sentiments about these features.

In 2013, Haruechaiyasak et al. [15] presented S-Sense, a framework for analyzing sentiments on Thai social media contents. The framework crawled and collected texts from social media such as Twitter and Pantip, a popular Thai webboard, and performed basic text processing such as word segmentation. It then classified texts into a predefined set of topics, and provided intention analysis to classify each text into four classes: announcement, request, question and sentiment. The sentiments were further classified into positive and negative. Their subsequent work [16] generalizes by extracting and analyzing keywords with statistical sigificance from social media contents.

All works that process Thai texts have similar text processing steps such as word segmentation and part-of-speech tagging. Inrak and Sinthupinyo's work [10] has a quite different purpose since it focuses on emotions while the other works [14]–[16] extract opinions and sentiments. Our work is more simlar to the latter. There are some differences however. Haruechaiyasak et al. [14]'s work on hotel reviews has predetermined features where ours dynamically extracts aspects from review texts. Lastly, the S-Sense framework [15], [16] analyzes keywords and sentiments from social media. Ours is similar but we group keywords using LDA since some keywords can be grouped into similar aspects.

III. Background

Processing Thai texts to extract opinions and sentiments requires word segmentation, part-of-speech tagging, topic modeling, and sentiment analysis. This section describes these theories and researches that are basis for our work.

A. Word Segmentation

Processing English text is easier than processing Thai text since English words in a sentence have spaces between them and therefore can be easily distinguished using a space as a delimiter. However, Thai text is written continuously without spaces and difficult to find word boundaries. Therefore, there is a need for word segmentation when processing Thai text.

There are several word segmentation techniques such as Longest Matching [17], Maximal Matching [18], Probabilistic Model [19], and Feature-based Approach [20]. The word segmentation tool used in this paper is LexTo [21] developed by National Electronics and Computer Technology Center (NECTEC) employs the Longest Matching technique.

The Longest Matching technique attempts to match an entire string of Thai text with words contained in a given lexicon. If no word in the lexicon can be matched with the string, the last character of the string is removed. The technique then attempts to match the truncated string with the lexicon again. The matching and truncating are repeated until a word is found in the lexicon. The word is then removed from the text, and the entire process is itereated for the rest of the text.

B. Part of Speech

In addition to word segmentation, we also need to tag each word a part of speech to identify whether this word is a (1) noun, (2) pronoun, (3) verb, (4) adverb, (5) adjective, (6) preposition, (7) conjunction, and (8) interjection. The categorization of parts of speech can vary depending on how precisely one wants to tag.

Part-of-speech tagging is a common step in natural language processing. There are several tools that can help tag parts of speech such as Nature Language ToolKit (NLTK) [22] and RDRPOStagger [23]. These tools need a corpus to learn parts of speech before being able to tag. NLTK has an English corpus and also allows any corpora from any languages to be added. RDRPOStagger has various corpora from seven languages including Thai. The Thai corpus in RDRPOStagger is the ORCHID [13] corpus developed by NECTEC. This paper uses the RDRPOStagger tool.

C. Topic Modeling

Opinion mining and sentiment analysis is a process that attempts to extract opinions and sentiments from given text [24]. For example, given user reviews of a product, one might want to know their opinions and feelings toward the product. In addition, one might want to know more in details which aspects or topics of the product users have opinions on. Topic modeling is a technique that helps discover underlying topics within given text. Latent Dirichlet Allocation (LDA) [7] is a common and effective topic modeling technique. LDA is based on a Bayesian model with the assumption that a given text contains a mixture of topics, and each topic contains various words with different probabilities.

There are various topic modeling tools. Our work uses gensim [25], which is a Python library requiring dependencies of the SciPy and NumPy libraries.

D. Sentiment Analysis

When extracting opinions about products, one also wants to know sentiments whether user opinions are positive, negative, or neutral. Two main approaches for sentiment analysis are machine learning based and lexicon-based. The machine-learning based approach learns and builds classifiers from texts labeled with sentiments. The classifiers are then used to classifiy sentiments. In the lexicon-based approach, each word has a sentiment score, and sentiments of given text is calculated from scores of words appearing in the text. SentiStrength [9] and SentiWordNet [26] provide lexical resources where English words are annotated with sentiment scores. Our work uses SentiWordNet as a sentiment lexical resource.

IV. Approach

The goal of our work is to extract opinions and sentiments about mobile applications from user reviews written in Thai. The work hopes to help Thai users assess mobile applications without needing to read all reviews and to also help Thai developers pinpoint where they can improve their software products. To achieve the goal, we employ 5 steps, shown in Figure 1, as follows: 1. Data Collection 2. Prepossessing 3. Sentiment Analysis 4. Topic Extraction 5. Summary.

We would like the entire process to be as automatic and accurate as possible, but there are several limitations. To process raw user reviews, most of the process can be done automatically but some steps need manual intervention. Following subsections describes these five steps and also states whether the step is done automatically or manually. We also make an assumption that one sentence has one sentiment to make the process possible.

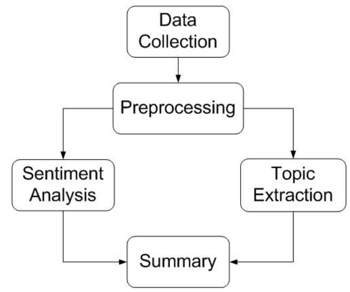

Figure 1. Overview of our approach

A. Data Collection

Table I
Number of reviews in each application

Application	No. of Reviews
Man Man	1,279
H-TV	691

We collected user reviews in Google Play Store from two mobile applications: ”แม่น แม่น” or ”Man Man” (a virtual keyboard) and ”H-TV” (online TV). The user reviews collected are dated between February 2015 and August 2016. The number of user reviews for each application is shown in Table I. The information for each review includes author, title, detail, rate, and review date. Table II shows examples of user reviews.

The collection process is done automatically using JavaScript to retrieve user reviews automatically from the Google Play Store website. The user reviews are then stored in a database for further analysis.

B. Prepossessing

After user reviews have been collected, the next step is to perform sentence segmentation, word segmentation and part-of-speech tagging.

1) sentence segmentation: One user review can contain several sentences expressing opinions about various aspects. Since we make an assumption that one sentence has one sentiment, we need to break reviews into a list of single sentences. Thai writing makes it difficult because there is no formal sentence boundary like a period or a question mark in Engish. Spaces in Thai can mark the end of a sentence or the end of a clause. Therefore, we cannot simply use spaces to indicate the end of sentences. Although there are several researches focusing on breaking Thai text into sentences, there are no tools that can easily be used. We therefore perform sentence segmentation manually. Once a tool is available, it can be applied to make this pre-processing step easier.

Table II
Example of Reviews

Author	**Title**	**Review**	**Rate**	**Date** (mm/dd/yyyy)
โชคชัย มหาวงนันท์	โชคชัย มหาวงศ์นันท์	ใช้ได้ดีครับ	5	10/04/2015
bie slow life		พักหลังนี่อัพบ่อยนะครับ	4	09/19/2015
ornanohg Hongrrimon		ชอบค่ะใช้ง่าย มีตัวการ์ตูนให้ด้วย	5	09/20/2015
Terdsak chompusri		เรียบง่ายแต่ใช้ได้ดีจริงๆครับชอบมาก	5	09/22/2015
Worapote Panomauppatum	วรพจน์ พนมอุปถัมภ์	ใช้ได้เยี่ยมมาก	5	09/25/2015
Nate Makboon	เนตร มากบุญ	ดีมากครับ สะดวกดีแม่นสุดยอด	5	09/24/2015

2) word segmentation: As mentioned in Section III, we use LexTo [21] from NECTEC to perform word segmentation. LexTo can segment words very well if words are spelled correctly. However, processing raw user reviews is difficult because of the informal language and slangs used in the reviews. There are also many spelling errors, either accidentally or intentionally to emphasize the meaning such as ”มากกกกกก”, causing the tool to segment words incorrectly.

Today, more and more texts are written informally. It is not practical to manually correct these words so that the word segmentation tool can perform correctly. In our work, we want this step to be automatic and therefore, we sacrifice accuracy.

3) POS tagger: We use the RDRPOStagger tool [23] with the ORCHID corpus [13] for POS tagging. Since some slang and misspelled words are segmented incorrectly, the POS tool tags those words as ”unknown”. In the sentiment analysis and topic modeling steps, only words tagged with nouns, verbs, and adjective/adverb are used. Moreover, since one word can be tagged with more than one POS, this POS annotation is used to identify various meanings of one word. For example, the word ”ฉัน” has two meanings with different parts of speech. As a noun, it means ”I”. As a verb, it means ”eat”.

C. Sentiment Analysis

Once words in user reviews are tagged with POS, sentiments in these reviews can be analyzed. Our work applies the lexicon-based approach. However, there is no resource that annotates each Thai word with sentiment scores. Therefore, the English SentiWordNet [26] is used in combination with an electronic Thai-English dictionary called LEXiTRON [27] developed by NECTEC. Note that some words in the reviews are already written in English. For these English words, their sentiment scores can be retrieved from SentiWordNet without using LEXiTRON.

We assign sentiment scores only to words tagged with nouns, verbs, and adjective/adverb. To find score for each of these Thai words, our script automatically looks up in the LEXiTRON dictionary to retrieve its English translation with the same POS. The next step is to look for sentiment scores in SentiWordNet for the English word with the same POS. In SentiWordNet, each word is assigned with three numerical scores: *Pos*, *Neg*, *Obj*. These numbers indicate how positive, negative, and ”objective” (or neutral) the word is where each number is between [0.0,1.0] and their sum is 1.0. In our approach, we assign only one numerical score to a word by subtracting the *Neg* score from the *Pos* score. Since *Obj* means neutral, we do not take this number into account.

Moreover, further analysis is also needed. For example, a word preceded with ไม่, which means ”no” or ”not”, will have its sentiment score reversed. In addition, some Thai words have more than one associated English translations with the same POS which have different sentiment scores. Some Thai words may have more than one associated English translations but none has the same POS. In both cases, all sentiment scores from all translations are averaged and then assigned to the Thai word. Table III shows top ten words for each sentiment found in user reviews from Man Man application.

Table III
Top 10 words for each sentiment in Man Man application

Negative		**Positive**	
Word	***Sentiment***	***Word***	***Sentiment***
ลบ	-0.33621	น่ารัก	0.21843
เสียดาย	-0.17095	รัก	0.20107
เกลียด	-0.16666	เพลิน	0.17563
ดุ	-0.15297	ดี	0.16622
สายตายาว	-0.12500	สวย	0.16310
ขยายตัว	-0.09566	สุดยอด	0.15476
ห่วย	-0.09071	มันส์	0.15085
ปวด	-0.07943	ไว	0.12500
เสียใจ	-0.07943	ชอบ	0.12246
ไม่ดี	-0.06995	สนุก	0.10604

When sentiment scores are assigned to all nouns, verbs, adjectives, and adverbs in a sentence, the sentiment score for the sentence is calculated by averaging these sentiment scores.

Furthermore, we find that several negative sentences are assigned with positive score. Upon closer look, we find a pattern in these sentences where there is a word ชอบ preceding with a negative verb such as ”ชอบค้างบ่อยๆ”, which means the application often freezes. The word ชอบ in Thai means ”like”, and if used in front of a verb, it also means ”often”. Since the POS tool tags ชอบ as a verb which associates with the ”like” meaning, its sentiment score is returned with a high positive number. This type of sentences is therefore incorrectly assigned with positive sentiment. Hence, we eliminate the word ชอบ that precedes a verb to increase accuracy.

D. Topic/Aspect Extraction

In addition to sentiment analysis, we also want to pinpoint what aspects users are talking about in the reviews. We use

LDA, a topic modeling technique, to extract topics/aspects. We supply only nouns, verbs, adjectives, and adverbs in all user reviews to the LDA Python tool.

When using the LDA technique, a number of topics must be specified. Since we do not know exactly how many aspects users are talking about in the reviews, we have experimented on different numbers of topics ranging from 10, 11, ..., 20. We find that main topics stay put when increasing the number of topics and as the number of topics gets higher, the more refined topics seem to be. We then choose the number of topics to be 20 so that it is not too few nor too many to comprehend the information. To make this process more flexible, we can make this number of topics configurable.

After running LDA, for each topic, several words and associated probabilities that the words belong that topic are returned. LDA does not return the "name" of the topic. Thus, to interpret the LDA result, we manually choose a few words with high probabilities as the name of the topic/aspect so that users and developers can easily understand results.

E. Summary

After sentences are annotated with sentiment scores and aspects are extracted, we summarize the information by assigning sentiment scores to the extracted aspects. The assigning process is done as follows. For each word belonging in a topic, we find all the sentences containing the word and collect all their sentiment scores. The absolute maximum sentiment score is then assigned to the topic.

In addition, we also count how many sentences with positive and negative sentiments for each topic so that developers can see in more details how much users like or dislike the aspects.

V. Result

Table IV displays aspects and sentiments after analyzing user reviews using our approach. Due to space limit, the table only shows results from "Man Man" application. In the table, the number of positive and negative sentences are also shown to give developers a sense of how often aspects were mentioned in user reviews. Figure 2 visualizes the results from Table IV with a graph and sorts aspects by sentiment scores.

The sentiment scores shown in Table IV has the highest positive and negative scores of 0.5153 and -0.5156, respectively. Upon examining SentiWordNet, we find that highest *Pos* and *Neg* scores are both 0.75 since the *Obj* score is always nonzero. Since we use averages and absolute maximums when assigning scores to words and sentences, resulting sentiment scores for aspects should be in the range of [-0.75,0.75]. Thus, we can categorize the resulting scores to *low*, *medium*, *high* when absolute scores are [0,0.25), [0.25, 0.5), [0.5,0.75], respectively. Hence, the sentiment score of 0.5153 and -0.5156 in Table IV can be considered high positive and high negative.

We evaluated our approach by calculating precision, recall, F-measure, and accuracy using truth values from data labeled manually by one person not related to our research. To manually label data, the person was given aspects/topics generated from LDA in the topic modeling step and then

Table IV
20 topics/aspects discovered from Man Man user reviews

Topic/Aspect	Sentiment	# Pos. Sentences	# Neg. Sentences
เปลี่ยน ภาษา	0.2770	35	9
พัฒนา ยอด	0.2666	47	6
sticker หน่อย ขยาย	0.2046	36	15
สะดวก สวย	0.3926	84	13
สายตา ขนาด ใหญ่	0.5153	143	26
ยกเว้น ทำนาย	0.2315	19	2
ปรับปรุง ยาก	0.2035	17	11
ปุ่ม หาย	-0.5156	49	23
ชอบ แม่น	0.3926	142	14
สุดยอด ลอง	0.3926	145	12
แป้น รวน	-0.3352	24	9
แก้ไข สี	0.2167	40	15
พิมพ์ ง่าย	0.5153	201	12
เสียง เสียดาย	0.4066	17	4
ปรับ ขนาด ค้าง	0.5153	63	13
ปุ่ม แจ่ม ซับซ้อน	-0.5156	30	17
เพิ่ม อิโมจิ	0.4066	35	1
เดา คำ	-0.5156	44	9
พิมพ์ พลาด	0.5153	68	11
แป้น ใหญ่	0.3396	82	13

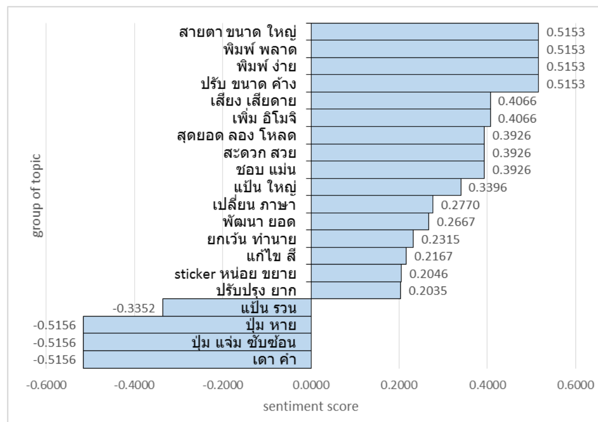

Figure 2. Sentiment scores for topics/aspects from Man Man user reviews

Table V
Evaluation results for sentiment analysis of extracted aspects

Application	Precision	Recall	F-Measure	Accuracy
Man Man	0.7188	0.6538	0.6848	0.55
H-TV	0.5252	0.5333	0.5293	0.5

labeled each aspect/topic with either positive or negative sentiment. Sentiment scores are not evaluated since it is highly subjective and can be difficult to analyze accuracy. Table V shows the evaluation results. The average F-measure from both applications is 0.6071.

We have not evaluated whether the LDA produced accurate topics/aspects. For topic modeling, accuracy evaluation is very subjective and time-consuming. Different persons may have very different points of view on how to identify topics/aspects in large amount of data. Since the LDA technique is widely known and used for topic modeling, we assume for now that results from LDA is acceptably accurate. Future work can be done to evaluate this step using similar human-intensive process as in Gunzmam and Laalej's work [3].

Limitation and Possible Future Works

Our work still has some limitations such as limitations in word segmentations of informal texts, slangs, and misspelled words. This causes LEXiTRON and SentiWordNet not being able to find translations nor sentiment scores. Moreover, some words have several meanings, and LEXiTRON returns several translations. Thus, we do not get precise meanings nor precise sentiment scores. Possible future work is to create a Thai sentiment lexical resource to help Thai researchers perform sentiment analysis easier and more accurate.

For topic modeling, our work fixes a number of topics and therefore are not flexible or dynamic enough. For future work, we can allow the number to be configurable or determined dynamically from additional information extracted from applications such as a list of features or application size.

The evaluation process is still not ideal since only one person creates truth values. We will ask more persons to label data as our future work. More mobile applications can also be analyzed to expand our case studies. We can also make the approach into a tool where users specify a name of a mobile application and let the tool analyze and display results.

VI. Conclusion

This paper presented an approach to analyze user reviews of mobile applications to discover aspects and their associated sentiments. The approach applied natural language processing steps such as word segmentation, part-of-speech tagging as well as topic modeling and sentiment analysis techniques. The average F-measure from our approach is 0.6071 which can be considered good, but more work can be done to improve the results such as improving underlying tools as discussed in the paper. We believe that our work should benefit users and developers since it reduces time in analyzing user reviews to assess mobile applications and help exposed aspects and sentiments buried in a vast amount of user reviews.

References

[1] A. Begel and T. Zimmermann, "Analyze this! 145 questions for data scientists in software engineering," in *Proceedings of the 36th International Conference on Software Engineering*. ACM, 2014, pp. 12–23.

[2] N. Chen, J. Lin, S. C. Hoi, X. Xiao, and B. Zhang, "AR-Miner: mining informative reviews for developers from mobile app marketplace," in *Proceedings of the 36th International Conference on Software Engineering*. ACM, 2014, pp. 767–778.

[3] E. Guzman and W. Maalej, "How do users like this feature? a fine grained sentiment analysis of app reviews," in *22nd IEEE International Requirements Engineering Conference (RE)*. IEEE, 2014, pp. 153–162.

[4] P. M. Vu, T. T. Nguyen, H. V. Pham, and T. T. Nguyen, "Mining user opinions in mobile app reviews: A keyword-based approach," in *Automated Software Engineering (ASE), 2015 30th IEEE/ACM International Conference on*. IEEE, 2015, pp. 749–759.

[5] W. Leopairote, A. Surarerks, and N. Prompoon, "Evaluating software quality in use using user reviews mining," in *Proceedings of the 10th International Joint Conference on Computer Science and Software Engineering (JCSSE)*, May 2013, pp. 257–262.

[6] K. Nigam, A. K. McCallum, S. Thrun, and T. Mitchell, "Text classification from labeled and unlabeled documents using EM," *Machine learning*, vol. 39, no. 2-3, pp. 103–134, 2000.

[7] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent Dirichlet allocation," *Journal of machine Learning research*, vol. 3, no. Jan, pp. 993–1022, 2003.

[8] Y. Jo and A. H. Oh, "Aspect and sentiment unification model for online review analysis," in *Proceedings of the fourth ACM international conference on Web search and data mining*. ACM, 2011, pp. 815–824.

[9] M. Thelwall, K. Buckley, G. Paltoglou, D. Cai, and A. Kappas, "Sentiment strength detection in short informal text," *Journal of the American Society for Information Science and Technology*, vol. 61, no. 12, pp. 2544–2558, 2010.

[10] P. Inrak and S. Sinthupinyo, "Applying latent semantic analysis to classify emotions in Thai text," in *2nd International Conference on Computer Engineering and Technology (ICCET)*, vol. 6, 2010, pp. 450–454.

[11] T. K. Landauer, P. W. Foltz, and D. Laham, "An introduction to latent semantic analysis," *Discourse processes*, vol. 25, no. 2-3, pp. 259–284, 1998.

[12] P. Charoenpornsawat. (1999) SWATH (Smart Word Analysis for THai). [Online]. Available: http://www.cs.cmu.edu/~paisarn/software.html

[13] 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.

[14] C. Haruechaiyasak, A. Kongthon, P. Palingoon, and C. Sangkeettrakarn, "Constructing Thai opinion mining resource: A case study on hotel reviews," in *8th Workshop on Asian Language Resources*, 2010, pp. 64–71.

[15] C. Haruechaiyasak, A. Kongthon, P. Palingoon, and K. Trakultaweekoon, "S-Sense: a sentiment analysis framework for social media sensing," in *Sixth International Joint Conference on Natural Language Processing*, 2013, pp. 6–13.

[16] Speech and Audio Technology Laboratory, National Electronics and Computer Technology Center. S-Sense. [Online]. Available: http://www.ssense.in.th/

[17] Y. Poovarawan and W. Imarrom, "Thai syllable separator by dictionary," in *Proceedings of the 9th Annual Meeting on Electrical Engineering of the Thai Universities*, Khonkaen, Thailand, December 1986.

[18] V. Sornlertlamvanich, "Word segmentation for Thai in machine translation system," *Machine Translation, National Electronics and Computer Technology Center, Bangkok*, pp. 50–56, 1993.

[19] K. Asanee, T. Chalathip, and S. Sapon, "A statistical approach to Thai word filtering," *Kasetsart Engineering Journal*, vol. 6, pp. 26–35, 1995.

[20] P. Charoenpornsawat, "Feature-based Thai word segmentation," Master's thesis, Computer Engineering, Master. Chulalongkorn University, Bangkok, 1999.

[21] National Electronics and Computer Technology Center. LexTo : Text lexeme tokenizer. [Online]. Available: http://www.sansarn.com/lexto

[22] NLTK Project. (2015) Natural language toolkit. [Online]. Available: http://www.nltk.org

[23] D. Q. Nguyen, D. Q. Nguyen, D. D. Pham, and S. B. Pham, "RDRPOSTagger: A ripple down rules-based part-of-speech tagger," in *Proceedings of the Demonstrations at the 14th Conference of the European Chapter of the Association for Computational Linguistics*, 2014, pp. 17–20.

[24] B. Liu and L. Zhang, "A survey of opinion mining and sentiment analysis," in *Mining text data*. Springer, 2012, pp. 415–463.

[25] R. Řehůřek and P. Sojka, "Software framework for topic modelling with large corpora," in *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*. Valletta, Malta: ELRA, May 2010, pp. 45–50, http://is.muni.cz/publication/884893/en.

[26] S. Baccianella, A. Esuli, and F. Sebastiani, "SentiWordNet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining." in *LREC*, vol. 10, 2010, pp. 2200–2204.

[27] National Electronics and Computer Technology Center. LEXiTRON. [Online]. Available: http://lexitron.nectec.or.th/2009_1