Indonesian Journal of Electrical Engineering and Computer Science
Vol. 22, No. 2, May 2021, pp. 1199~1207
ISSN: 2502-4752, DOI: 10.11591/ijeecs.v22.i2.pp1199-1207

Algorithm for extracting product feature from e-commerce comment

Chanida Kaewphet, Nawaporn Wisitpongpun
Faculty of Information Technology and Digital Innovation, King Mongkut's University of Technology North Bangkok, Thailand

Article Info

Article history:

Received Mar 4, 2020
Revised Dec 5, 2020
Accepted Jan 11, 2021

Keywords:

Feature extraction
Frequency-based
Syntax analyzer
Syntax-based
TF-IDF

ABSTRACT

Reviews of e-commerce play an important role in online purchasing decisions. Consumers are likely to read reviews and comments on products from other consumers. In addition to those opinions that reflect consumers' trust in products, it also provides each product's distinctive properties. Today, there are many online reviews, resulting in enormous comments and suggestions. However, as fully reading reviews is quite difficult, this article presents 3 algorithms for automatic extraction of product features hidden in e-commerce reviews: a traditional frequency-based product feature extraction (F-PFE), syntax analyzer system (SAS), and the hybrid approach called the frequency and syntax-based product feature extraction (FaS-PFE). The proposed algorithms were tested against 4 different types of products: shampoo, skincare, mobile phone, and tablet, using reviews from amazon.com. Based on the product review used in this study, it was found that the SAS can help improve the performance in terms of precision by 15% when compared with the traditional F-PEE approach. When considering both the word frequency and syntax, FaS-PFE clearly outperforms the other two approaches with 94.00% precision and 95.13% recall.

This is an open access article under the CC BY-SA license.

Corresponding Author:

Chanida Kaewphet
Faculty of Information Technology and Digital Innovation
King Mongkut's University of Technology North Bangkok
518 Pracharat 1 Road, Wongsawang, Bangsue, Bangkok 10800, Thailand
Email: Chanida.k@email.kmutnb.ac.th

1. INTRODUCTION

Nowadays, the growth of technology has resulted in the development of websites to a great leap, especially in the e-commerce business, which is more than an online shopping website. For over a decade, a great variety of products and services have made their way onto e-commerce platforms. A good example of these includes entertainment, food, electronics, travel, and beauty [1]. The e-commerce platforms, nowadays, can learn customers' likes and dislikes and provide a recommendation to help customers buy products with confidence. In return, customers also contribute by providing comments or opinions on the products that they bought. These online reviews can be used to evaluate consumer attitudes towards products, services, or organizations. Purchasing decisions are usually influenced or affected by feedbacks or opinions expressed on e-commerce reviews [2]. Therefore, companies use sophisticated algorithms to understand the buying behaviors of consumers to increase the efficiency of their products and organization. However, as it is not possible for most consumers to read many reviews to find out whether to buy a product or not, they need a system that can automatically distinguish product features from e-commerce reviews and classify whether the consumer has positive or negative feelings toward the products [3], [4].

To extract the product features presented in e-commerce reviews, Minqing Hu, and Bing Liu [5] used the rule-based technique to extract product features based on frequently used words. While this is a simple and useful technique, it can only extract features that were represented by nouns. If features are represented by verbs or words that are rarely found, such features will be ignored by this technique. Also, the frequency-based technique ignores implicit features hidden in the reviews. Hao Wang, *et al.* [6] improved the traditional frequency-based product feature extraction (TF-IDF) by using the point mutual information (PMI) to reduce the dimensions of the appropriate features under the conditions specified. Nevertheless, PMI requires a considerable amount of time for calculations. Anitha and Karpagam [7] used frequent pattern mining algorithms and association mining algorithms for extracting product features from common word patterns. Anisha and Niranjan [8] used the apriori algorithm for feature extraction and classified product features using an unsupervised SentiWordNet method. The extracted feature along with its POS tag: adjective, adverb, verb, the noun was used to select opinion words. Finally, negation rules are used for the classification of reviews into positive and negative.

Several studies focus on analyzing the sentence structure to extract product features. Sheng Huang, *et al.* [9] proposed a product feature categorization technique that relies on the use of semantic knowledge from WordNet to calculate the similarity between product features. Hanqian WU, *et al.* [10] created different connection rules using the dependency parsing analysis tool in the Chinese language structure to perform more explicit feature extraction. However, as the methods for the structure of these languages depend on domains and require automatic annotations of implicit properties, the creation of rules for language structure is unable to cover the language principles thoroughly. Hani, *et al.* created language structure rules as criteria for extracting product features [11]. This technique is quite popular as it can be applied and create syntax language rules following the critiques of many languages, including complex languages, such as Chinese and Arabic [12]. An ontology technique is also used to extract product features from online reviews. Teja, *et al.* [13] proposed feature extraction from comments using the feature ontology tree (FOT) technique along with latent dirichlet allocation (LDA), an unsupervised learning model. The model in this research will create topics of pre-defined words from the document set and extract the product features using the OWL class or web ontology language. Pitchayasee, *et al*, [14] present a knowledge extraction system from online tourist reviews by using ontology technology as a knowledge base for word analysis. Jitamon Angsakul proposed to use the ontology technique as a knowledge base for extracting and storing knowledge, as well as translation tools, to help with semantic analysis for Thai tourism businesses [15]. However, while the use of knowledge about tourism makes the results more accurate, these statistical techniques may not be reliable if the size of knowledge is too small. Mohammad Fikri and Riyanarto Sarno [16] present a comparative study of sentiment analysis using SVM and SentiWordNet. In this research, the sentiment analysis uses the rule-based method with the help of the SentiWordNet and support vector machine (SVM) algorithm with term frequency-inverse document frequency (TF-IDF) as a feature extraction method.

This research focuses on the automatic extraction of product features using three different algorithms. The first proposed algorithm is the frequency-based product feature extraction (F-PFE) which is an enhanced version of the traditional frequency method which only extracts common words based on their occurrence frequency. The efficiency of F-PFE is increased by simply categorizing the list of common words using their associated synonyms and antonyms. The second algorithm is the syntax analyzer system (SAS) which relies on the language structure rules for extracting candidate feature terms and uses the synonyms and antonyms corpus for categorizing feature words. However, frequency-based feature selection may neglect rare features which do not occur frequently, while the syntax-based approach may omit certain features which occur frequently but do not conform to the pre-specified rules. Hence, we proposed a frequency and syntax-based product feature extraction (FaS-PFE) which is a hybrid approach that aims to overcome the barriers of both F-PFE and SAS. The remainder of this article is organized as follows: Section 2 presents the proposed product feature extractions. Section 3 presents the results and discussion, and finally in section 4 concludes the finding of this study.

2. RESEARCH METHOD

2.1. Frequency-based product feature extraction (F-PFE)

The process of extracting product features from e-commerce reviews using frequency-based product feature extraction (F-PFE) is divided into three main steps as follows Figure 1:

Step 1: Text pre-processing: Reviews or opinions from amazon.com will be pre-processed by using word segmentation and stop-word removal. This step also includes the removal of symbols in sentences, such as parentheses, bullets, or other symbols such as #, *, -, etc. [17-19].

Step 2: Stemming and Lemmatization: The output from the previous step, will be transformed into their root form. For example:

oily ➔ oil
moisturizer ➔ moisture

Step 3. Frequency-based product feature extraction (F-PFE) in this step consists of 3 sub-steps as follows:

1) Remove irrelevant words: After the derivation of root words from the review data, the next step is to remove irrelevant words that often appear in the comments. It is necessary to remove them from sentences before counting their corresponding occurrence frequency. Example of such words is specific words, dates, times, places, people, product categories, words that express feelings, words that are often found in comments and not a feature such as "recommend", "review", "comment", "free", "product", etc.

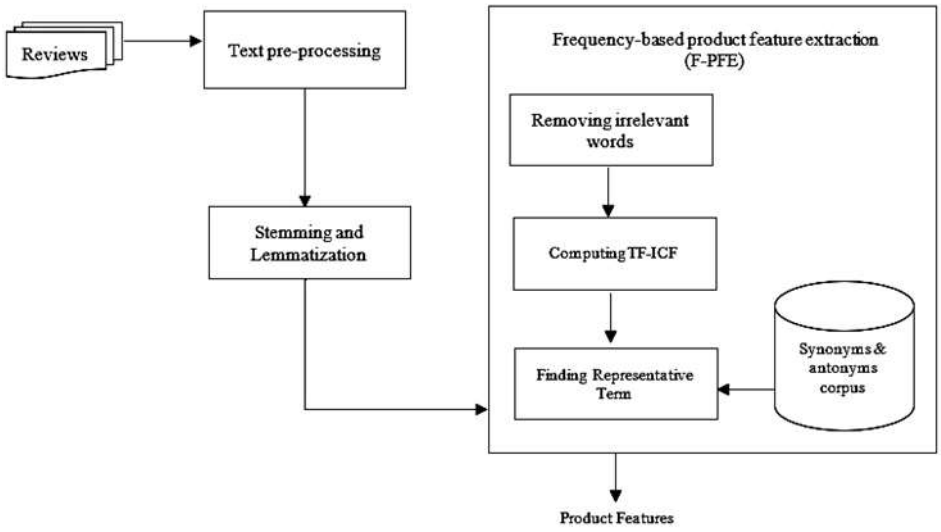

Figure 1. Frequency-based product feature extraction (F-PFE) algorithm

2) Compute TF-ICF (Term frequency-inverse comment frequency): This is a calculation that is based on TF-IDF which is an algorithm that provides scores of each word or term in a document by considering the proportion of words contained in the document and comparing to the number appearance of the most common words. Each comment, in this case, is equivalent to one document [20-21]. The term-frequency ($tf_{t,c}$) of each word can be derived from the number of times the word appears in a comment divided by the number of occurrences of the most frequent word in the comment formula [22-23], as is shown in (1);

$$tf_{t,c} = \frac{f_{t,c}}{\max\{f_{t,c}: t \in c\}} \quad (1)$$

where $f_{t,c}$ is the number of times the word *t* appears in the comments *c*. The inverse comment frequency, denoted by *icf* can be expressed by (2)

$$icf = log \frac{N}{|\{cf(t): t \epsilon\ c\}|} \quad (2)$$

where N is the total number of comments and $cf(t)$ is the number of comments which contain the word *t*. Finally, the TF-ICF or the term frequency-inverse comment frequency can be calculated by (3).

$$tf\_icf_{t,c} = tf_{t,c} \times icf_{t,c} \quad (3)$$

3) Find representative term ($F_{F\text{-}PFE}$) of each word by using *synonyms & antonyms corpus* from WordNet. In this step, we will construct the synonyms and antonyms corpus (C_{sa}) by finding the set of synonyms and antonyms (W^{sa}) corresponding to each word (w_i) obtained from the TF-ICF process. Besides, the

representative term (*f*) of each W^{sa} will be elected based on the TF-ICF value of each word in the set. More specifically, the word with the highest TF-ICF value will be elected as the representative term. The new TF-ICF value assigned to this representative term will be an average of the TF-ICF of all the $w_i \in W^{sa}$, as can be described by (4).

$$tf\_icf(f) = \frac{\sum_{i=1}^{n} tf\_icf(w_i)}{n} \quad (4)$$

where $tf\_icf(w_i)$ is the TF-ICF of each word w_i in a set of synonyms and antonyms W^{sa}
n is the number of words in a set of synonyms and antonyms W^{sa}
$tf\_icf(f)$ is an average TF-ICF value assigned to the representative term (*f*)

The algorithm for constructing C_{sa} is shown in Figure 2.

Algorithm :Constructing Synonyms & Antonyms Lexicon
Input: Set of word (*W*), Synonyms and Antonyms Corpus (C_{sa})**,**
Output: Synonyms and Antonyms Corpus (C_{sa})
Steps:

```
for each wi in W
    for each W_k^sa in C_sa
        if wi ∈ W_k^sa
            f = representative term of W_k^sa
            if TF_ICF (wi) > TF_ICF(f)
                f = wi
                change the representative term of W_k^sa to the new word f
            end
        else
            W_i^sa = find_set_of_synonym_and_antonym(wi)
            include W_i^sa in C_sa
end
```

Figure 2. Algorithm for constructing Synonyms and Antonyms Corpus

Finally, the TF-ICF of the representative terms will be normalized and the final set of the product features comprises representative terms ($F_{F\text{-}PFE}$) with associated TF-ICF of greater than 25%.

2.2. Syntax analyzer system (SAS)

The syntax analyzer system extracts product features by using rules derived from syntax structure that may likely results in product features. There are three steps involved in building the syntax analyzer system as is shown in Figure 3.

Step 1 Text pre-processing: Similar to the frequency-based product feature extraction approach, customers' comments were processed by using word segmentation and stop-word removal.

Step 2 Stemming and Lemmatization: In addition to converting output word set from step 1 into their root forms (*R*), each word will be labeled with its corresponding part of speech [24-25]: nouns (NN, NNS), adjectives (JJS, JJR), adverbs (RB, RBR), verbs (VB, VBD), etc. [26-28].

Step 3 Syntax-based feature extraction: This process consists of 3 sub-steps as follows: 1) Similar to the frequency-based approach, irrelevant and specific words will be removed before further processing. 2) Syntax analyzer considers a *bigram*, which is a word-pair or two consecutive words in a sentence. If a bigram conforms to any of the specified syntax rules, there is a hidden product feature within the bigram. The rule used in this study is based on *Part-of-Speech* or *POS* tags. For the utmost convenience and ease of use, we considered only four different tag groups in the syntax analysis rule, including nouns: NN, NNS, verbs: VB, VBD, VBG, VBN, VBP, VBZ. Adverbs: RB, RBR, RBS, and adjectives: JJ, JJR, JJS. For example, the sentence.

"*makes my hair clean, soft, and smooth without weighing it down! Touchable volume and silky finish.*"

will become:

"makes hair clean soft smooth without weigh touch volume silk finish".

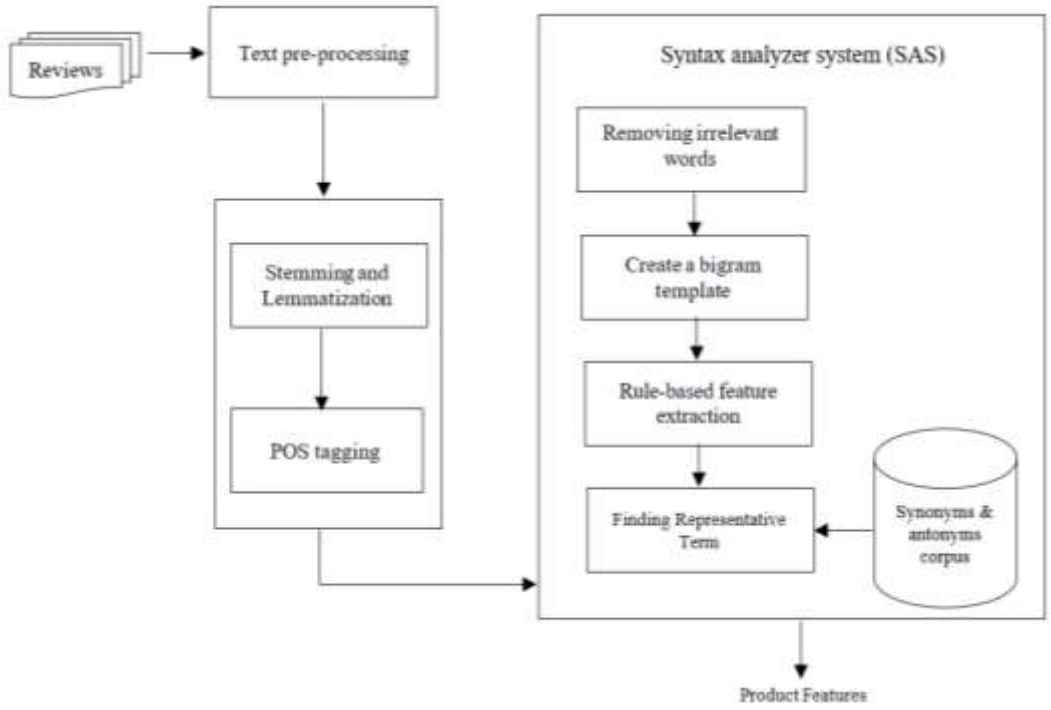

Figure 3. Syntax Analyzer System

After text pre-processing. The bigram representing this sentence is shown in Figure 4. Each word in a bigram pair will be tagged with its corresponding POS and will be analyzed one by one [29].

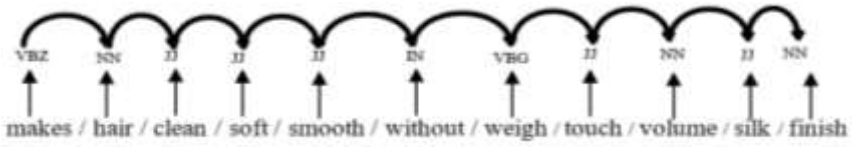

Figure 4. Shows the step of word pairing one by one

According to Figure 4, SAS outputs 10 bigrams with the following POS structures: VBZ+NN, NN+JJ, JJ+JJ, JJ+JJ, JJ+IN, IN+VBG, VBG+JJ, JJ+NN, NN+JJ, and JJ+NN. By labeling the feature in the training data set, we can obtain the rules by using a decision tree algorithm. Examples of rules are:

1) Feature of "hair (NN) + clean (JJ)" is clean
2) Feature of "volume (NN) + silk (JJ)" is silk

SAS classifies all words from comments into a set of feature terms. From 4000 comments from 4 different types of products (1000 comments per product), we were able to derive 10 syntax rules. However, due to the diversity of the languages used in the comments, we can only cover common cases in consumers' reviews and were not able to create rules that cover all possible cases. 3) Synonyms & antonyms corpus from WordNet is used in this step to find the set of representative features for each word. This process is similar to finding the representative feature term in the last step of the frequency-based product feature extraction.

2.3. Frequency and syntax-based product feature extraction (FaS-PFE)

The frequency and syntax-based feature extraction (FaS-PFE) is a hybrid approach that aims to overcome the drawback of both F-PFE and SAS. More specifically, F-PFE selects words based on occurrence frequency, hence, rare feature terms may not be selected by F-PFE. On the other hand, SAS selects feature terms based on semantic rules so frequently occurred feature terms may not get chosen by SAS if they appear in a sentence that does not conform to the rules as in Figure 5.

Hence, FaS-FE considers the feature term selected by both algorithms using a certain criterion as can be described by (5).

$$F_{FaS-PFE} = F_{SAS} \cup F^{th}_{F-PFE} \quad (5)$$

where F^{th}_{F-PFE} is the set of features obtained from F-PFE with *occurrence frequency* (*OF*) higher than the threshold *th*. The term *t* is a member of F^{th}_{F-PFE} if its occurrence frequency *OF*(*t*), the number of comments in which term *t* appears divided by the total number of comments, is above the pre-defined threshold *th*.

According to (4), FaS-PFE chooses feature terms according to the pre-defined condition. In FaS-PFE, the representative feature terms consist of the feature terms obtained from SAS or F_{SAS} and features obtained from F-PFE with occurrence frequency above a certain threshold. In this study, we set the threshold to be 25%. For example, if a certain feature term appears in 250 comments from the total number of 3,000 comments, the occurrence frequency of this particular feature term is 25%.

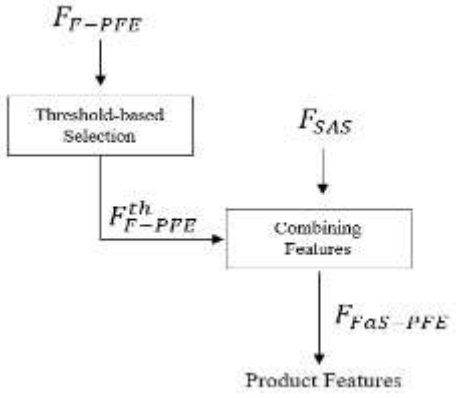

Figure 5. Frequency and Syntax based Product Feature Extraction

3. RESULTS AND DISCUSSION

To evaluate the performance of the proposed product feature extraction algorithms, we used 12,000 product review comments from the amazon.com website which consists of 3,000 comments on shampoo, 3,000 comments on skincare products, 3,000 comments on tablets, and 3,000 comments on mobile phones. Experts were asked to read each product's comments and listed all the features associated with each product. These actual features will later be used for evaluating the precision, recall, and overall performance in terms of the F1 score of the proposed algorithm [30]. Let F be a set of features obtained from the proposed algorithm and FA be the actual features listed by experts, the (6)-(8) describe the definition of each performance metric.

$$Precision = \frac{|F \cap F_A|}{|F|} \quad (6)$$

$$Recall = \frac{|F \cap F_A|}{|F_A|} \quad (7)$$

Both precision and recall indicate the ability of the algorithms to extract correct product features. In particular, precision measures the fraction of the retrieved product features that are actual features. On the other hand, the recall is the fraction of the relevant product features that are successfully retrieved. However, an algorithm with high precision does not necessarily imply high performance in terms of recall and vice versa. Hence, both precision and recall should always be considered when evaluating the performance of an

algorithm. Alternatively, the F1 score, which is the mean of the precision and recall, can be used to measure the performance of an algorithm and can be expressed by (8),

$$F_1 = 2 \times \frac{Precision \times Recall|}{Precision + Recall} \quad (8)$$

3.1. Performance of Frequency-based Product Feature Extraction (F-PFE)

Table 1 shows the efficiency of F-PFF. It was found that the average precision and recall are 77.68% and 90.91%, respectively. The results indicate that F-PFE may not be able to capture rare feature terms that do not appear frequently in the comments. Hence, the precision is only 77.68%. However, 90.91% of the feature terms extracted by F-PFE were actual features. When considering how well F-PFE performs across different types of products, we found that F-PFE performs better on IT products (tablets and mobile phones) than on beauty products (shampoo and skincare). This may be because IT products have explicit specifications that consumers often consider when choosing the products. In particular, consumers are usually considered weight, camera, battery lifetime, etc. when buying a tablet and mobile phone. On the other hand, the feature of beauty products was slightly harder to extract because consumers may use complex and complicated sentences to express what they like and do not like about the products.

Table 1. The efficiency of the frequency-based product feature extraction.

Product	Precision	Recall	F1
Shampoo	73.08%	95.00%	82.61%
Skincare	75.00%	88.24%	81.08%
Tablet	77.78%	91.30%	84.00%
Mobile phone	84.85%	89.10%	86.92%
Overall (Average)	77.68%	90.91%	83.65%

3.2. Performance of Syntax analyzer system (SAS)

Shown in Table 2, SAS improves performance in terms of precision significantly. More specifically, the overall precision improves by almost 15%. The recall, on the other hand, drops roughly by 12% when SAS is used to extract shampoo features. This may be due to the "free-format" nature of the language used for reviewing the shampoo products considered in this study. Interestingly, the recalls of other products' features were slightly better than those obtained from F-PFE. To compare the overall performance, SAS performs much better than F-PFE as is indicated by the F1 score.

Table 2. The efficiency of the syntax analyzer system.

Product	Precision	Recall	F1
Shampoo	92.31%	82.76%	87.27%
Skincare	92.31%	88.89%	90.57%
Tablet	92.59%	92.59%	92.59%
Mobile phone	93.30%	96.60%	94.90%
Overall (Average)	92.63%	90.21%	91.33%

3.3. Performance of the Frequency and Syntax-base Product Feature Extraction (FaS-PFE)

When considering both factors, frequency-base, and syntax-base, FaS-PFE clearly outperforms the other approaches. As is shown in Table 3, the average precision and recall of this approach are 94.00% and 95.13%, respectively. This is higher than that of the other two approaches. Besides, FaS-PFE performs equally well across all different types of products.

Table 3. The efficiency of the frequency and syntax-based product feature extraction.

Product	Precision	Recall	F1
Shampoo	96.15%	92.59%	94.34%
Skincare	90.48%	95.00%	92.68%
Tablet	92.59%	96.15%	94.34%
Mobile phone	96.77%	96.77%	96.77%
Overall (Average)	94.00%	95.13%	94.53%

4. CONCLUSION

In this study, we proposed three different techniques used for extracting product features from online review comments: frequency-based product feature extraction (F-PFE), syntax analyzer system (SAS), and frequency and syntax-based product feature extraction (FaS-PFE). F-PFE used the frequency of word occurrences in the comments described by TF-ICF to extract candidate product features. This approach achieves 77.68% precision and 90.91% recall. SAS, on the other hand, used syntax rules to elect features term from the bigram. By considering the syntax, SAS was able to extract feature terms better than F-PFE. It can achieve 92.63% precision and 90.21% recall. Last but not least, when combining the result of both F-PFE and SAS using a threshold-based criterion, the hybrid approach or FaS-PFE is a clear-cut winner among the three approaches. The precision and recall of FaS-PFE are 94.00% and 95.13%, respectively. When considering different product types, we found that all three algorithms perform better when extracting features from the products with many explicit feature specifications such as IT products (tablets and mobile phones). However, FaS-PFE is robust enough to perform equally well on all product types.

ACKNOWLEDGEMENTS

Firstly, we are grateful to express sincere thanks to our faculties who gave support. Secondly, we would like to express our gratitude to all the authors of the papers included in this research. Finally, a special thanks to the conference team for accepting our research.

REFERENCES

[1] Electronic Transactions Development Agency, "ETDA reveals that the value of Thai e-Commerce has grown consistently Shoots up to 3.2 trillion baht in 2018" 2018. [Online]. https://www.etda.or.th/

[2] C. Chotildakitika, "The attitude and social media marketing that affecting purchase decision of personalized products via online channel," Bangkok University, Pathumthani, Thailand, 2018.

[3] H. Kaur ,V. Mangat and Nidhi, "A Survey of Sentiment Analysis techniques," *2017 International conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud)*, pp. 921-925, 2017, doi: 10.1109/I-SMAC.2017.8058315

[4] V. M. Pradhan, J. Vala and P. Balani, "A Survey on Sentiment Analysis Algorithms for Opinion Mining," *International Journal of Computer Applications (0975- 8887)*, vol. 133, no. 9, pp 7-11, 2016.

[5] H. Minqing and L. Bing, "Mining opinion features in customer reviews," in *Proceedings of the 19th national conference on Artificial intelligence*, San Jose, California, 2004, pp. 755-760.

[6] W. Hao, Y. Yang, K. Jie, Z. Xinhui, W. Chao and D. Jianyong, "Research on Feature Mining Algorithm Based on Product Reviews," *2019 IEEE International Conference on Artificial Intelligence and Computer Applications (ICAICA)*, pp. 205-21, 2019, doi: 10.1109/ICAICA.2019.8873491

[7] A. S. and K. K., "Feature Extraction of Customer Reviews Using Frequent Pattern Mining Algorithm," *International Journal for Modern Trends in Science and Technology.*, vol. 03, no. 09, pp. 91-95, 2017.

[8] A. P. Rodrigues and N. N. Chiplunkar, "Mining online product reviews and extracting product features using unsupervised method," in *2016 IEEE Annual India Conference (INDICON)*, Bangalore, India, 2016.

[9] H. Sheng, L. Xinlan, P. Xueping, N. Zhendong, "Fine-grained Product Features Extraction and Categorization in Reviews Opinion Mining," *2012 IEEE 12th International Conference on Data Mining Workshops*, pp. 680-686, 2012, doi: 10.1109/ICDMW.2012.53

[10] W. Hanqian, L. Tao and X. Jue, "Fine-Grained Product Feature Extraction in Chinese Reviews," *2017 International Conference on Computing Intelligence and Information System (CIIS)*, pp. 327-331, 2017.

[11] N. Hani , M. Warih and S. Siti , "Feature extraction and opinion classification using class sequential rule on customer product review," *2016 4th International Conference on Information and Communication Technology (ICoICT)*, pp. 1-5, 2016, doi: 10.1109/ICoICT.2016.7571891

[12] R. P. Venkata and R. V. Smrithi, "Recommending products to customers using opinion mining of online product reviews and features," *2015 International Conference on Circuits, Power and Computing Technologies [ICCPCT-2015]*, pp. 1-5, 2015, doi: 10.1109/ICCPCT.2015.7159433

[13] S. D. Teja, V. B. Vishnu and D. Ramesh, "Extracting Product Features from Reviews Using Feature Ontology Tree Applied on LDA Topic Clusters," *2016 IEEE 6th International Conference on Advanced Computing (IACC)*, pp. 163-168, 2016, doi: 10.1109/ICoICT.2016.7571891

[14] P. Kitwatthanathawon, T. Angskun and J. Angskun, "A Knowledge Extraction System from Online Reviews using Fuzzy Logic," *2012 Ninth International Conference on Computer Science and Software Engineering (JCSSE)*, pp. 189-196, 2012, doi: 10.1109/JCSSE.2012.6261950

[15] J. Angskun, "The Design and Development of a Knowledge Extraction and Retrieval System via Online GIS for Thailand Tourism Business," Suranaree University of Technology, Nakhon Ratchasima, Thailand, 2012.

[16] M. Fikri and R. Sarno, "Mohammad Fikri, Riyanarto A comparative study of sentiment analysis using SVM and SentiWordNet," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 13, no. 3, pp. 902-909, 2019, doi: 10.11591/ijeecs.v13.i3.pp902-909

[17] W. Kaolim, "Keyword extraction using sequential pattern mining," Silpakorn University Repository : SURE, 2014. [Online]. Available: https://bit.ly/31tJIXa.

[18] L. Jongwon, L. Jaeseung and J. Hoekyung, "Main keyword comparison based on document analysis system," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 19, no. 3, pp. 1533-1539, 2020, doi: 10.11591/ijeecs.v19.i3.pp1533-1539

[19] V. Korde, "Text Classification and Classifiers:A Survey," *International Journal of Artificial Intelligence & Applications (IJAIA)*, vol. 3, no. 2, pp. 85-99, 2012, doi: 10.5121/IJAIA.2012.3208

[20] P. Somjin, "Opinion mining for online teaching evaluation," School of Information Technology Institute of Social Technology Suranaree University of Technology, Nakhon Ratchasima, Thailand, 2015.

[21] M. Ali Fauzi, "Random Forest Approach for Sentiment Analysis in Indonesian Language," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 12, no. 1, pp. 46-50, 2018, doi: 10.11591/ijeecs.v12.i1.pp46-50

[22] H. Wu, R. Luk, K. Wong and K. Kwok, "Interpreting TF-IDF term weights as making relevance decisions," *ACM Transactions on Information Systems*, vol. 26, no. 3, 2008, doi: 10.1145/1361684.1361686

[23] K. T. Uçar, "How to Calculate TF-IDF (Term Frequency–Inverse Document Frequency) in Python," iyzico. engineering, 2018 [Online]. Available: https://bit.ly/3np4pMw

[24] A. Ramachandran, "NLP Guide: Identifying Part of Speech Tags using Conditional Random Fields," analytics-vidhya, 2018. [Online]. Available: https://bit.ly/2ECFoM8.

[25] D. Godayal, "An introduction to part-of-speech tagging and the Hidden Markov Model," freecodecamp.org, 2018. [Online]. Available: https://bit.ly/2Ly9Bzu.

[26] tutorialspoint.com, "Basics of Part-of-Speech (POS) Tagging," tutorialspoint.com, 2018. [Online]. Available: https://bit.ly/2QpxUij.

[27] M. Rouse, "part-of-speech tagger (PoS tagger)," whatis.techtarget.com, 2018. [Online]. Available: https://bit.ly/2Ks3BYA.

[28] J. Awwalu, S. E. Abdullahi, A. E. Evwiekpaefe, "Part Of Speech Tagging: A Review Of Techniques," *FUDMA Journal of Sciences (FJS)*, vol. 4, no. 2, pp. 712-721, 2020, doi: 10.33003/fjs-2020-0402-325

[29] J. Daniel and M. H. James , "N-gram Language Models," 2019. [Online]. Available: https://stanford.io/3r0Cy7n.

[30] E. Pachawongsakda, "Classifier evaluation metrics," Data Mining Trend, 2015. [Online]. Available: http://dataminingtrend.com/2014/classifier_evaluation_metrics/.

BIOGRAPHIES OF AUTHORS

Chanida Kaewphet received her B.B.A.in business information technology from Rajamangala University of Technology Suvarnabhumi and M.S.Tech.Ed. in Computer Education from King Mongkut's University of Technology North Bangkok in 2009, and 2011, respectively. Currently, she is a lecturer in the Department of Information Systems and Business Computer, Faculty of Business Administration and Information Technology, Rajamangala University of Technology Suvarnabhumi, Thailand.

Nawaporn Wisitpongpun received her B.S., M.S., and Ph.D. degrees in electrical and computer engineering from Carnegie Mellon University in 2000, 2002, and 2008, respectively. From 2003 to 2009 she was also a research associate in the Electrical and Control Integration Laboratory, General Motors Corporation. Presently, she is an assistant to the president for research and information technology and a lecturer in the Faculty of Information Technology at King Mongkut's University of Technology North Bangkok, Thailand. Her research interests include traffic modeling, chaos in the Internet, and cross-layer network protocol design for wireless networks, social network analysis, and digital government.