Computer Science • 22(4) 2021 https://doi.org/10.7494/csci.2021.22.4.4028

Tuan Anh Tran
Jarunee Duangsuwan
Wiphada Wettayaprasit

# NOVEL FRAMEWORK FOR ASPECT KNOWLEDGE BASE GENERATED AUTOMATICALLY FROM SOCIAL MEDIA USING PATTERN RULES

**Abstract** *One of the factors that improve businesses in business intelligence is summarization systems that can generate summaries based on sentiment from social media. However, these systems cannot produce such summaries automatically; they use annotated datasets. To support these systems with annotated datasets, we propose a novel framework that uses pattern rules. The framework has two procedures: 1) pre-processing, and 2) aspect knowledge-base generation. The first procedure is to check and correct any misspelled words (bigram and unigram) by a proposed method and tag the parts-of-speech of all of the words. The second procedure is to automatically generate an aspect knowledge base that is to be used to produce sentiment summaries by sentiment-summarization systems. Pattern rules and semantic similarity-based pruning are used to automatically generate an aspect knowledge base from social media. In the experiments, eight domains from benchmark datasets of reviews are used. The performance evaluation of our proposed approach shows the highest performance when compared to other unsupervised approaches.*

**Keywords** opinion mining, aspect knowledge base, aspect extraction, pattern rules, social media

**Citation** Computer Science 22(4) 2021: 489–516

## 1. Introduction

With the rapid increase of internet technologies, social media such as Facebook, Twitter, e-commerce websites, etc. have become popular recently [2], which will affect internet users' feelings, comments, or feedback with others. User-generated content on social media is being increased day-by-day and is becoming a valuable source. These useful opinionated reviews can be found a huge volume of unstructured texts. To get relevant information by retrieving and mining opinions from the huge volume is one of the tasks in opinion mining. Because of the beneficial effects on understanding customer opinions, opinion mining has been studied by researchers in many languages [1,8,12,21,22,25,29,32] and applied to sentiment-summarization systems [11,13,14,33]. These summarization systems can help people observe different perspectives from their summaries to make decisions. For example, visual summaries, textual summaries, and structured summaries were generated from customer reviews in Figure 1 [5,11,14,19,37]. Improving businesses via these summarization systems is a target of business intelligence [7].

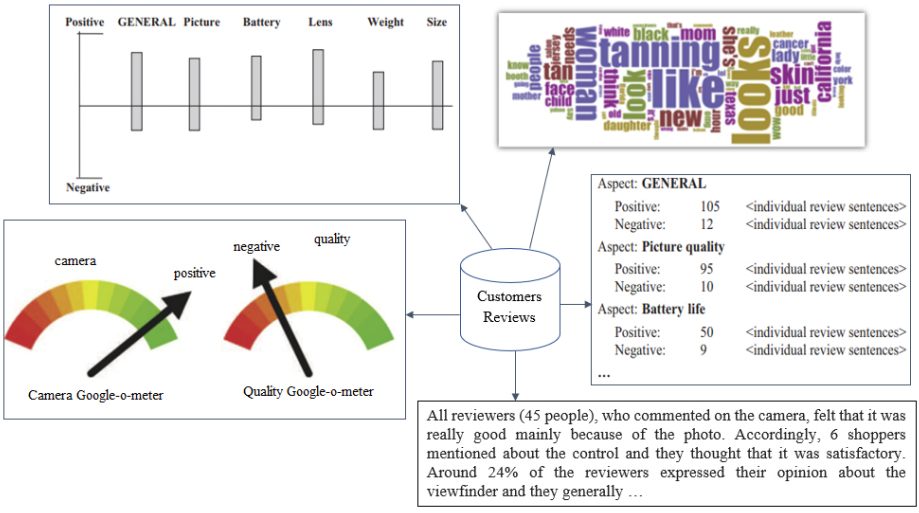



**Figure 1.** Sentiment summaries generated from customer reviews

To generate such summaries, these summarization systems feature three major steps [14]: 1) extracting the aspects (features) of an entity that have seen interest and been mentioned by Internet users; 2) determining the opinion sentences (positive/negative) in reviews; and 3) generating summaries based on the results of the first two steps. Although sentiment-summarization systems did not include Step 1 and/or Step 2, they could generate summaries such as [11,33,44], etc. These sentiment-summarization systems used aspects and polarities that were annotated. To

automatically generate sentiment summaries, these summarization systems need to do three steps automatically without providing annotations of aspects and polarities. Hence, the first step (aspect extraction) plays a vital role in these systems.

In textual reviews of social media, there are two kinds of opinions: coarse-grained opinions, and fine-grained opinions [25]. A coarse-grained opinion detects an overall opinion for an entity, whereas a fine-grained opinion is concerned with specific attributes of the entity. Extracting the specific attributes of the entity is useful because these extracted attributes are conveyed regarding what the customers' opinions on the details of the entity are. These specific attributes of the entity are called aspects. The result of an extracting process is called aspect extraction [5]. For example, from the opinion sentence, *"The camera is awesome",* an extracted aspect is *"camera".* Aspect extraction is a difficult task of sentiment analysis because customers have different ways of expressing their opinions.

In previous works, the syntactic-based studies that were used to extract aspect from customer reviews were rule-based. Most of these studies extracted aspects from a simple sentence. Some aspects missed extraction if some aspects were the same constituent of a simple sentence. For example, from the sentence, *"The speed and memory are very good",* the aspect of *"memory"* was extracted, but the aspect of *"speed"* was not.

To overcome these drawbacks, we propose an aspect knowledge-base generation using pattern rules (AKGPR) framework in this study to enhance a mechanism for sentiment summarization. Social media (e.g., product or service reviews from e-commerce websites) is an input for the AKGPR framework. The pre-processing processes correct misspelled words and tags the POS (part-of-speech) of each input word. The AKGPR framework uses pattern rules to automatically generate aspect candidates and useful information from social media. A large number of aspect candidates are extracted; however, many extracted aspect candidates are not associated with an entity in the text of social media. A semantic-based pruning technique is applied for the pruning process. An output that consists of extracted aspect candidates and useful information is an aspect knowledge base. The contributions in this work propose an aspect knowledge-base generation using pattern rules (AKGPR) framework that generates an aspect knowledge base by using pattern rules. Pattern rules that are based on syntax can extract aspects from both simple and compound sentences.

The rest of the paper is organized as follows. In Section 2, the background of and related work on aspect extraction are presented. In Section 3, the proposed framework architecture is explained in detail. Later, the test results and evaluation are explained in Section 4. Finally, the derived conclusions and future work are given in Section 5.

## 2. Background and related work

In this section, previous studies that are related to aspect extraction will be discussed. These studies are divided into supervised and unsupervised approaches based on

whether the use of annotated data for the training processes is required or not. Before introducing approaches to aspect extraction, we introduce some terminologies.

*Domain-independent and Domain-dependent Opinion* [38]: a domain-independent opinion is a word that always conveys only one sentiment, such as "good," "bad," "love," or "hate." The meaning of the opinion word is clearly positive or negative. A domain-dependent opinion is a word whose sentiment depends on an aspect in a specific domain. For example, "small" for a picture size means it is difficult to see (negative meaning), but "small" for a cellphone size means it is easy to carry (positive meaning).

*Explicit Aspect and Implicit Aspect* [25]: an explicit aspect occurs in an opinionated sentence, whereas an implicit aspect does not occur in an opinionated sentence. For example, the sentence, *"The picture quality is good."* has an explicit aspect of *"picture quality"* because *"picture quality"* occurs in the sentence. The sentence, *"The phone is heavy."* has an implicit aspect of "*weight*" because *"weight"* does not occur in the sentence and is being mentioned by the user.

*Subjective Aspect and Objective Aspect* [17]: a subjective aspect is an aspect that has a relationship with an opinion word, whereas an objective aspect does not have any relationship with an opinion word. For example, the sentence, *"This camera is awesome."* has a subjective aspect of *"camera"* because *"camera"* has a relationship with *"awesome."* The sentence, *"This phone comes with a rechargeable battery."* has an objective aspect of *"rechargeable battery"* because *"rechargeable battery"* does not have any relationship with any opinion word.

## 2.1. Aspect extraction using supervised approaches

Supervised approaches use annotated data in a training process; then, the output of the training process is used to extract aspect. Jin et al. [16] introduced a model that was based on lexicalized hidden Markov models (HMM) to extract aspects and opinions from customer reviews. The model was trained by multiple linguistic features. Jakob and Gurevych [15] used a conditional random field (CRF) model to extract aspects. A set of domain-independent features (e.g., POS tags, dependency relations, etc.) was used to train the CRF model. Before training, these models applied methods for selecting features. Liu et al. [25] proposed automated rule selection approaches that were used in a greedy algorithm and a local search algorithm. These proposed approaches were based on the dependency relationships between aspects and opinion words and are known as rule selection greedy (RSG and RSG$^+$), rule selection local search (RSLS and RSLS$^+$), and CRF$^+$ algorithms. Feng et al. [9] presented an approach to extracting aspects by adopting topic modeling and synonyms. Nawaz et al. [30] introduced an approach by using a normalized Google distance and ConcepNet (NGD + CNET) in order to apply to an aspect-extraction problem. Tubishat et al. [45] suggested the improved whale optimization algorithm + pruning algorithm (IWOA + PA) to extract aspects. Deep neural network models have been proposed to extract aspects. Poria et al. [34] suggested a model that

combined a convolutional neural network (CNN) with linguistic patterns for aspect extraction. Ying et al. [48], Li & Lam [24], and Li et al. [23] proposed models that were based on long short-term memory (LSTM) to extract aspects. In addition, Mai & Le [27] proposed models that were based on LSTM to extract aspects for Vietnamese reviews. The supervised approach had some cost for the training process, and its performance depended on the trained dataset.

## 2.2. Aspect extraction using unsupervised approaches

Unsupervised approaches do not require annotated data in the training process; these include studies that were statistic-based, ontology-based, or rule-based.

The first group from the unsupervised approach is statistic-based, which uses frequency to extract aspects. All of the studies in this group [14, 22, 46] use an association rule-mining algorithm to extract aspect candidates. Hu & Liu [14] used association rule mining (ARM) to calculate the frequencies of noun phrases. Wei et al. [46] also used the association rule-mining algorithm to extract aspects via the proposed semantic-based product feature-extraction (SPE) method. Meanwhile, Li et al. [22] used the Apriori algorithm to extract aspect candidates for Chinese reviews.

The second group from the unsupervised approach is ontology-based. Marstawi et al. [28] and Konjengbam et al. [20] used an ontology to extract aspects from reviews. Marstawi et al. [28] built an ontology by using the GATE ontology editor. Meanwhile, Konjengbam et al. [20] built an ontology by using extracted aspect candidates and their relationships.

The third group from the unsupervised approach is rule-based. Some studies in this group [17,25,36] use dependency relationships between aspects and opinion words. Qiu et al. [36] suggested a double-propagation (DP) algorithm based on dependencies to extract aspects. However, there were incorrect aspects in this study because of propagation. To overcome this problem and increase the algorithm's accuracy, Liu et al. [25] and Kang & Zhou [17] proposed the DP extension method ($DP^+$) and rule-based methods (RubE), respectively. The $DP^+$ method [25] was used to extract aspects by extending the dependency relationships in DP (18 dependency relationships). RubE [17] was used to extract subjective aspects by using the extended DP method and was used to extract objective aspects by using a hybrid method (which combined a part-whole relationship and review-specific patterns). The other studies that used syntax are [3, 4, 13, 18, 26, 29, 35, 38, 39]. Htay et al. [13] proposed patterns for extracting aspects with their opinion words/phrases from customer reviews. The parts-of-speech (POS) of opinion words/phrases for extracting aspects are adjectives, adverbs, verbs, and nouns. Khan et al. [18] suggested hybrid dependency patterns to extract aspects from customer reviews. The hybrid dependency patterns were combined lexical relationships with an opinion context. Maharani et al. [26] introduced a set of syntactic patterns to extract aspects from customer reviews; these syntactic patterns were determined manually. Asghar et al. [3] introduced an aspect-based opinion-mining framework that was used to extract aspects by using

heuristic patterns. Mataoui et al. [29] proposed an approach that used syntactic rules for the Arabic language to extract aspects. This approach had five steps and was allowed to update the rules. Rana & Cheah [38] introduced a two-fold rules-based model (TF-RBM) to extract aspects; this model used sequential pattern rules that were combined with domain-independent opinions and domain-dependent opinions for the first and second folds, respectively. Rana & Cheah [39] introduced a sequential pattern rules-based (SPR) approach to automatically generate sequential pattern rules for extracting aspects. Bagheri et al. [4] and Poria et al. [35] proposed two models for extracting explicit and implicit aspects. Bagheri et al. [4] used POS patterns and heuristic rules to extract explicit aspects, and a graph was used to extract implicit aspects. Poria et al. [35] used the rule-based approach on common-sense knowledge and dependency trees to extract explicit and implicit aspects.

## 3. Proposed method

To automatically generate an aspect knowledge base from social media for the sentiment-summarization systems, the proposed framework of aspect knowledge-base generation using pattern rules (AKGPR) is illustrated in Figure 2 and is constituted by two main procedures: 1) pre-processing, which includes *misspelling correction* by using a dictionary, and a mismatch threshold $\lambda$ for updating new words in the dictionary (*POS Tagging*); and 2) aspect knowledge-base generation (AKG), which includes *aspect candidate extraction* by using an opinion lexicon and pattern rules, *aspect pruning* by using keywords (KW) and Word2Vec, and a similarity threshold $\varphi$ for selecting aspect candidates. The input of the proposed framework is social media; e.g., product or service reviews from e-commerce websites. The output of the proposed framework is an aspect knowledge base that will be used in the next steps of the sentiment-summarization systems. All of the functions are discussed in the next sections.

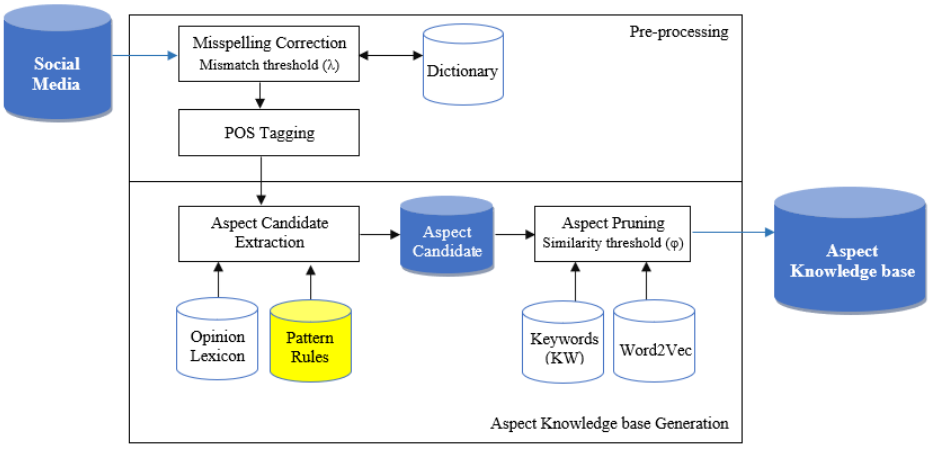



**Figure 2.** Architecture of aspect knowledge-base generation

### 3.1. Misspelling correction

This function aims to correct any misspelled words from a dictionary. Before checking and correcting misspelled words in the text of social media, this function will remove some special characters in the text of social media such as numbering or bulleting at the beginnings of sentences, pairs of quotations, html tags, etc. These special characters affect the extraction of the aspect; hence, eliminating these special characters from the text is necessary. Regular expressions are used to eliminate these special characters.

There are two types of misspelled words: non-word error, and real-word error [6]. A non-word error has no meaning and does not exist in the dictionary, while a real-word error exists in the dictionary and is not suitable in the context of that sentence. In this function, the non-word error process is conducted in this study. Our proposed idea is used to check and correct misspelled words in terms of bigrams and unigrams. Furthermore, new misspelled words and their correct replacements are also updated on the dictionary if their scores are lower than mismatch threshold $\lambda$. The smaller the score, the more two words match.

The Jaccard distance [31] and Levenshtein distance (edit distance) [10] can be used to check the dissimilarity of two words. The Levenshtein distance scores of the two words are the same; however, the Jaccard distance scores for those two words are different. Moreover, the Jaccard distance scores also show the total different characters in a set of characters of the misspelled word as well as the correct word. For example, the Levenshtein and Jaccard distance scores for "computer" with "computor" and "computar" are shown in Table 1. The Levenshtein distance scores for the two incorrect words for "computer" with "computor" and "computar" are equal to one; however, the Jaccard distance scores for "computer" with "computor" and "computar" are 1/8 (0.13) and 2/9 (0.22), respectively. Note that the 1/8 score means that there is one character that is different within eight characters (the set of characters of the misspelled word and the correct word). Hence, the Jaccard distance is used in our study to calculate the scores of a misspelled word and the corrected word. The difference between a misspelled word and a correct word is only one character within the set of characters of the misspelled word and the correct word. In our system, mismatch threshold $\lambda$ is set to 0.2.

**Table 1**

Example of Jaccard and Levenshtein distances between "computer" and other words

| Distance | Word | | | | | |
|---|---|---|---|---|---|---|
| | computer | computor | computar | computors | computars | computers |
| Jaccard | 0 | 1/8=0.13 | 2/9=0.22 | 2/9=0.22 | 3/10=0.30 | 1/9=0.11 |
| Levenshtein | 0 | 1 | 1 | 2 | 2 | 1 |

Samanta and Chaudhuri [41] suggested a method that is based on a ranking of each word in a confusion set to detect an unigram misspelled word and correct it. The

confusion set of a candidate word include words from a dictionary if the Levenshtein distance of the candidate word and the word in the dictionary was equal to one. Clark and Araki [6] and Singh and Sachan [42] proposed the detection of an unigram misspelled word and how to correct it if a corresponding correct word existed in the dictionary. Our system proposes misspelled words in terms of both bigrams and unigrams. The new misspelled words and their corrected words are also updated in the dictionary.

## 3.2. POS tagging

The proposed framework is designed to mine interesting aspects from social media by using pattern rules. Hence, all words in a text are needed to tag part-of-speech (POS). This is an aim of the POS-tagging function. The POS tagger from the SpaCy [43] library for Python is used.

## 3.3. Aspect candidate extraction

The purpose of the aspect candidate-extraction function is to extract aspect candidates from the social media text that was pre-processed in the previous step. The function uses pattern rules and an opinion lexicon dictionary. Before discussing the function in detail, this study introduces the pattern rules (PRs) and opinion lexicon (OL) dictionary as the following:

**Pattern rules (PRs)**

The pattern rules (PRs) are determined by using the relationship between the aspect and an opinion word. The relationships that are based on a syntactic structure are determined from the dependency tree of the categories [40]. The dependency tree of the categories is the fundamental constituent of the grammatical framework. The grammatical framework[1], which is currently maintained by Krasimir Angelov et al., is used to build the systems (e.g., translations, multi-language web tools, etc.) for more than 30 languages (e.g., English, French, etc.). From the dependency tree of the categories, the English language syntax of a sentence or a clause is considered and shown in Figure 3. For each node, the category (POS) and the description of the category are shown in Table 2.

Before discussing a method for determining the PRs, the study introduces a sequential pattern of a common noun (sCN) as the following:

In social media, Internet users can express their feelings about one or many attributes of the products/services that they purchased/used. These attributes can express together can describe in their comments/feedback and can be a single word for a noun, multiple words for a noun phrase, or a list of nouns/noun phrases. These attributes are usually potential aspects, and their categories (POS) are usually nouns. Hence, all of these attributes are needed to be detected. However, a common noun (CN) in Figure 3 has a main constituent that is a noun (N).

[1] http://www.grammaticalframework.org/

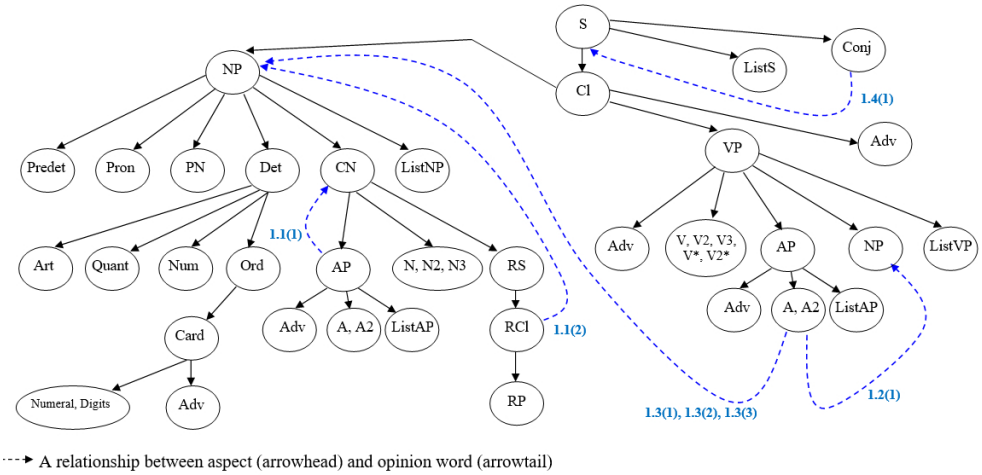



**Figure 3.** Dependency tree for Group 1 (adjective describes noun)

**Table 2**
List of categories [40]

| Category | Description of category | Category | Description of category |
|---|---|---|---|
| A/A2 | adjective | Ord | ordinal number |
| AP | adjectival phrase | PN | proper name |
| Adv | adverb | Predet | predeterminer |
| Art | Article | Prep | preposition |
| CN | common noun (without | Pron | personal pronoun |
| | determiner) | Quant | quantifier |
| Card | cardinal number | RCl | relative clause |
| Cl | declarative clause | RP | relative pronoun |
| Conj | conjunction | RS | relative |
| Det | determiner phrase | S | sentence |
| Digits | cardinal or ordinal in digits | V | verb |
| N | noun | V2 | two-place verb |
| N2 | relational noun | V3 | three-place verb |
| N3 | three-place relational | VP | verb phrase |
| | noun | V* | VA (adjective-complement verb) |
| NP | noun phrase | – | VV (VP-complement verb) |
| Num | number determining element | V2* | V2A (verb with NP/AP complement) |
| Numeral | cardinal/ordinal in words | – | V2V (verb with NP/V complement) |

To detect a sequence of these attributes in one sentence, the sequence of a common noun (sCN) pattern is identified (as illustrated in Figure 4). In Figure 4, an item in brackets () is optional, an arrow is a transition between two items, and the

symbol "/" means "or." Examples show that a single word, multiple words, or a list of words are determined by using the sCN pattern in Figure 4 as follows:

• *Determining single word (noun):*
Given a tagged sentence "*Audio*/N *is*/V2A *excellent*/A *./.*", the sCN can be determined as a single word (noun) – *"Audio"*.

• *Determining multiple words (noun phrase):*
Given a tagged sentence "*The*/Art *picture*/N *quality*/N *is*/V2A *great*/A *./.*", the sCN can be determined is multiple words – *"The picture quality"*.

• *Determining multiple words (noun phrase):*
Given a tagged sentence "*Quality*/N *of*/Prep *picture*/N *is*/V2A *good*/A *./.*", the sCN can be determined is multiple words – *"Quality of picture"*.

• *Determining list of nouns/noun phrases:*
Given a tagged sentence "*The*/Art *speed*/N *and*/Conj *memory*/N *are*/V2A *very*/Adv *good*/A *./.*", the sCN can be determined as a list of a noun/noun phrase – *"the speed"* and *"memory"*.

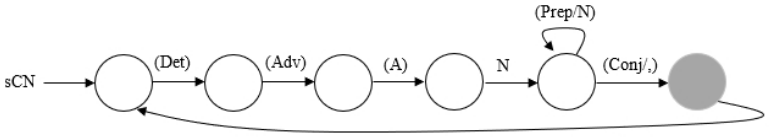



**Figure 4.** Diagram of representation of sequential pattern of common noun (sCN)

There are three steps for determining the pattern rules: 1) determining the groups; 2) determining the sub-groups; and 3) determining the pattern rules.

***Step 1***: *Determining groups*

The relationship between an aspect and an opinion word play a vital role in determining the pattern rules. These relationships convey the sentiment for the aspect; hence, they are grouped into groups according to the POS of the aspect and the opinion word. One group consists of those relationships whose constituents are the same (i.e., in the relationships, the POSs of the opinion words are the same, and the POSs of the aspects are the same).

With the POS of the aspect (adjective/noun/verb) and the POS of the opinion word (adjective/adverb/noun/verb) [14, 25], four groups of relationships are determined: 1) *adjective describes noun* (adjective is opinion word/noun is aspect); 2) *verb describes noun* (verb is opinion word/noun is aspect); 3) *noun describes noun* (noun is opinion word/other noun is aspect); and 4) *adjective describes verb* (adjective is opinion word/verb is aspect).

***Step 2***: *Determining sub-groups*

The sub-groups of each group are determined by using the positions of an aspect and an opinion word. The four positions that the aspect and the opinion word can be in are 1) noun phrase (NP), 2) verb phrase (VP), 3) sentence (S), and 4) compound

sentence (CS). For determining the sub-groups for each group, the steps that are carried out are as follows:

*Step 2.1*: Determine position in which opinion word and aspect could be.

*Step 2.2*: Check concurrence of position for opinion word and aspect in Step 2.1 – if position for opinion word and aspect is concurrent, save concurrence of position (note that concurrence of position must be followed by English grammar).

*Step 2.3*: Repeat Step 2.1 until all positions are checked.

More details of how to determine the sub-groups of Group 1 (adjective describes noun) are as follows: In this group, the POS of an opinion word is an adjective (A), and the POS of an aspect is a noun (N). For the NP position, the opinion word and the aspect are concurrent – such as "a/Art *good*/A and *camera*/N." The first sub-group (**Sub-group 1.1**) of Group 1 is $\mathbf{NP_{an}}$. For the VP position, the opinion word and the aspect are concurrent with a preposition between them – such as "I/Pron was/V2A *disappointed*/A *with*/Prep *quality*/N *./.*" The second sub-group (**Sub-group 1.2**) of Group 1 is $\mathbf{VP_{an}}$. For the S position, the opinion word and the aspect are concurrent when the opinion word is a verb phrase of this sentence and the aspect is a noun phrase of this sentence – such as "*audio*/N *is*/V2A *excellent*/A *./.*" The third sub-group (**Sub-group 1.3**) of Group 1 is $\mathbf{S_{an}}$. For the CS position, the opinion word and aspect are concurrent when the opinion word is in one constituent of one sentence and the aspect is in one constituent of the other sentence. A conjunction (e.g., "but", "and", etc.) combines these two sentences. For instance, "I/Pron used/V *the*/Art *player*/N and/Conj it/Pron is/V2A *good*/A *./.*" The fourth sub-group (**Sub-group 1.4**) of Group 1 is $\mathbf{CS_{an}}$.

By performing Steps 2.1–2.3, we also determine all of the sub-groups for Groups 2, 3, and 4. These sub-groups for Groups 1–4 are represented in Figure 3 (for Group 1) and Figure 5 (for Groups 2, 3, and 4). In Figures 3 and 5, an arrowtail is an opinion word, and an arrowhead is an aspect. All of the sub-groups for Groups 1–4 are described as follows:

- For Group 1 (adjective describes noun), four sub-groups are determined: Sub-group 1.1 is $NP_{an}$, Sub-group 1.2 is $VP_{an}$, Sub-group 1.3 is $S_{an}$, and Sub-group 1.4 is $CS_{an}$.
- For Group 2 (verb describes noun), two sub-groups are determined: Sub-group 2.1 is $VP_{vn}$, and Sub-group 2.2 is $S_{vn}$.
- For Group 3 (noun describes noun), three sub-groups are determined: Sub-group 3.1 is $VP_{nn}$, Sub-group 3.2 is $S_{nn}$, and Sub-group 3.3 is $CS_{nn}$.
- For Group 4 (adjective describes verb), one sub-group is determined: Sub-group 4.1 is $VP_{av}$.

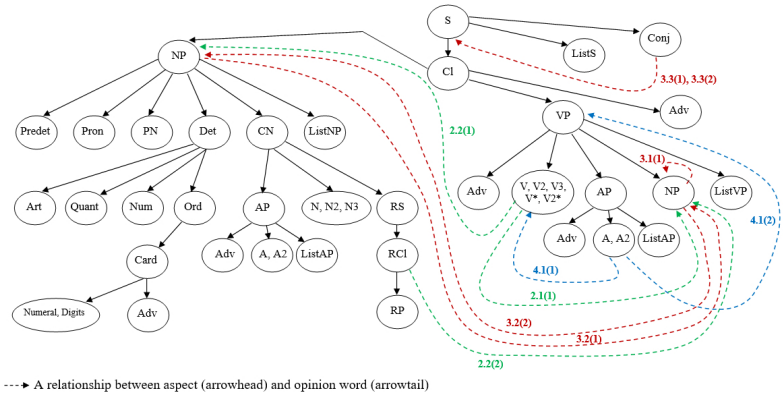



**Figure 5.** Dependency tree for Group 2 (verb describes noun), Group 3 (noun describes noun), and Group 4 (adjective describes verb)

Note that $NP_{xy}$ is a noun phrase that consists of an opinion word and an aspect. $VP_{xy}$ is a verb phrase that consists of an opinion word and an aspect. $S_{xy}$ is a sentence that consists of an opinion word in a verb phrase and an aspect in a noun phrase. $CS_{xy}$ is a compound sentence that consists of an opinion word in one sentence and an aspect in the other (where *xy* means that *x* is the POS of opinion word, and *y* is the POS of the aspect).

***Step 3**: Determining pattern rules*

A sub-group is constituted by constituents (POSs) that are compulsory and optional; hence, there are some different ways to represent a sub-group. To determine the pattern rules for each sub-group, the steps to perform are as follows:

*Step 3.1*: Determine constituents (POS) for sub-group including POS for opinion word and aspect.

*Step 3.2*: Check concurrence of POS for opinion word and aspect in sub-group in Step 3.1 – if POS for opinion word and aspect are concurrent, save and name concurrence of POS for sub-group (note that sub-group can be represented by many combinations of constituents' POSs).

*Step 3.3*: Decompose constituents into atomic constituents except for constituents that do not need to determine POS of opinion word (e.g., RCl constituent, CN constituent, etc.).

*Step 3.4*: Combine patterns in Step 3.3 (if any).

*Step 3.5*: Determine constituents of pattern in Step 3.4 for intensifier word, opinion word, and aspect.

*Step 3.6*: Repeat Step 3.1 until all possible POSs for constituents in sub-group are checked.

Note that *Step 3.2* is used to determine the *syntax-based pattern rules*. *Steps 3.3–3.5* are used to determine the *sequence-based pattern rules*. In the pattern rules, italicized words are opinion words, boldface words are aspect(s), intensifier words are adverbs (Adv), bracketed words are optional, and italicized boldface words are co-reference words. A constituent of a sentence may be repeated; hence, a subscript (e.g., "a," "b," etc.) shows the positions for that constituent.

More details of how to determine the pattern rules for *Sub-group* 1.1 ($NP_{an}$) are shown as follows: A noun phrase (NP) (in Figure 3) consists of many constituents, such as pronouns (Pron), adjective phrases (AP), common nouns (CN), relative clauses (RCl), etc. In Sub-group 1.1 ($NP_{an}$), we need to determine an adjective for an opinion word and a noun for an aspect. An AP consists of an adjective, a CN consists of a noun, and AP and CN are concurrent. The first syntax-based Pattern Rule 1.1(1) for Sub-group 1.1 is AP + CN. By decomposing the constituents of 1.1(1), the AP is decomposed into A or Adv + A. The CN does not need to be decomposed, as determining an opinion word is not necessary here. After decomposing, there are two possible sequence-based pattern rules for 1.1(1): A + CN, and Adv + A + CN. These pattern rules can be combined into (Adv) + A + CN. To determine the constituents for an opinion word and an aspect, the sequence-based pattern rule for 1.1(1) is (Adv) + *A* + **CN**. An RCl is a relative clause and can have an adjective. The second syntax-based Pattern Rule 1.1(2) is CN + RCl. To decompose the constituents of 1.1(2), the RCl is decomposed into RP + V2A + A or RP + V2A + Adv + A. The CN does not need to be decomposed, as determining an opinion word is not necessary here. After decomposing, there are two possible sequence-based pattern rules for 1.1(2): CN + RP + V2A + A, and CN + RP + V2A + Adv + A. These pattern rules can be combined into CN + RP + V2A + (Adv) + A. To determine the constituents for an opinion word and an aspect, the sequence-based pattern rule for 1.1(2) is **CN** + RP + V2A + (Adv) + *A*.

To perform Steps 3.1–3.6 as Sub-group 1.1, 20 pattern rules are determined (as shown in Table 3). The first column is the *pattern number*, the next two columns of Table 3 are *sub-group name* and *sub-group ID*, and the last two columns are syntax-based and sequence-based pattern rules, respectively, in which the *syntax-based pattern rule* column shows the pattern rules that are designed by using the dependency tree and syntax rules. The *sequence-based pattern rule* column presents the pattern rules that are sequentially generated in the details of the constituents from the syntax-based pattern rule. Note that one syntax-based pattern rule might be generated one or more times than one sequence-based pattern rule; for example, the syntax-based pattern rule of Sub-group 1.3(1) could be generated into two sequence-based pattern rules (P4 and P5).

In addition, the sequence-based rule can simultaneously determine more than one aspect if these exist in one sentence. All of the CN in the last column of Table 3 use the sequential pattern of a common noun (sCN) in Figure 4.

The examples show that potential aspects, opinion words, and intensifier words are detected by the pattern rules as follows:

• *Detect aspect, opinion word, and intensifier word in NP, Sub-group* 1.1(1) *or* $NP_{an}$:
Review sentence: *It is good pictures and white balance.*
Tagged sentence: *It*/Pron *is*/V2A *good*/A *pictures*/N *and*/Conj *white*/A *balance*/N *./.*

Sequence-based Pattern Rule (P1): (Adv) + *A* + **CN**.
Detected by Pattern Rule (P1): *It*/Pron *is*/V2A *good*/*A pictures*/**N** *and*/**Conj** *white*/**A** *balance*/**N** *./.*

• *Detect aspect, opinion word, and intensifier word in sentence S, Sub-group* 3.2(2b) *or* $S_{nn}$:
Review sentence: *Lens and memory are the best features.*
Tagged sentence: *Lens*/N *and*/Conj *memory*/N *are*/V2A *the*/Art *best*/A *features*/N *./.*
Sequence-based Pattern Rule (P15): **CN**+ V2A + (Adv) + *A* + *N*.
Detected by Pattern Rule (P15): *Lens*/**N** *and*/**Conj** *memory*/**N** *are*/V2A *the*/Art *best*/*A features*/*N ./.*
Note that there are no intensifier words in these examples; this is because adverbs do not exist in these tagged sentences.

**Opinion lexicon (OL) dictionary**

The opinion lexicon (OL) dictionary is one of the important factors for extracting aspect candidates. Opinion words in the OL dictionary are used to determine aspect candidates. The OL dictionary is built by combining two well-known opinion lexicon dictionaries (Hu & Liu's opinion lexicon [14] and MPQA's opinion lexicon [47]). The OL dictionary has 2759 positive words and 5552 negative words.

**Aspect Candidate-Extraction function**

The aspect candidate-extraction function is used to extract aspect candidates from social media (e.g., customer reviews) by using the pattern rules (PRs) in Table 3 and the OL dictionary (the social media text is pre-processed in the previous step). During the extraction of the aspect candidates, this function also extracts other useful information such as opinion words and intensifiers. Aspect candidates with useful information are used to generate the summaries. The definitions are introduced as follows:

let *ac* be aspect candidate;

let *ow* be opinion word in OL dictionary;

let *iw* be intensifier word.

**Definition 3.1** *An* **AOI** ***(aspect-opinion-intensifier)*** *is a set of quadruple elements* $< ac, ow, iw, tF >$ *in a review in Equation* (1)*:*

$$\text{AOI} = \{< ac_i, ow_i, iw_i, tF_i >\}, \tag{1}$$

where $i$ is the index of an aspect candidate, $1 \leq i \leq$ n, n is the number of extracted aspect candidates, and $tF_i$ is the total frequency of an extracted triple $(ac_i, ow_i, iw_i)$.

The *aspect candidate-extraction* algorithm in Algorithm 1 is used to extract aspect candidates and other useful information (e.g., opinion words and intensifiers) from the reviews. Line 1 is used to initialize the AOI, and Lines 2–14 are used to extract aspect candidates. The algorithm starts with a check for matching a sentence $S$ with each pattern $p$ (Line 4).

**Table 3**
Pattern rules for extracting aspect

| Pattern No. | Sub-group Name | Sub-group Id | Pattern Rule | |
|---|---|---|---|---|
| | | | Syntax-based | Sequence-based |
| P1 | $NP_{an}$ | 1.1(1) | *AP* + **CN** | (Adv) + *A* + **CN** |
| P2 | $NP_{an}$ | 1.1(2) | **CN** + *RCl* | **CN** + RP + V2A + (Adv) + *A* |
| P3 | $VP_{an}$ | 1.2(1) | V2A + (Adv) + *A2* + **NP** | V2A + (Adv) + *A* + Prep + **CN** |
| P4 | $S_{an}$ | 1.3(1a) | **CN** + V2A + *AP* | **CN** + V2A + (Adv) + *A* |
| P5 | $S_{an}$ | 1.3(1b) | **CN** + V2A + *AP* | $\mathbf{CN_a}$ + "(" + $\mathbf{CN_b}$ + ")" + V2A + (Adv) + *A* |
| P6 | $S_{an}$ | 1.3(2) | **CN** + RCl + V2A + *AP* | **CN** + RCl + V2A + (Adv) + *A*<br>Note: RCl is any pattern |
| P7 | $S_{an}$ | 1.3(3) | $\mathbf{CN_a}$ + V + (Prep) + $CN_b$ + V2A + *AP* | $\mathbf{CN_a}$ + V + (Prep) + $CN_b$ + V2A + (Adv) + *A*<br>Note: Prep is "by"; V is V+ed / V+ing |
| P8 | $CS_{an}$ | 1.4(1) | $Pron_1$ + V/V2A + **CN** + (Adv) + Conj + ***Pron₂*** + V2A + *AP* | $Pron_1$ + V/V2A + **CN** + (Adv) + Conj + ***Pron₂*** + V2A + (Adv) + *A*<br>Note: $Pron_2$ is a co-reference of CN |
| P9 | $VP_{vn}$ | 2.1(1) | (Adv) + *V2* + **NP** | (Adv) + *V2* + **CN** |
| P10 | $S_{vn}$ | 2.2(1) | **CN** + V2A + *V* | **CN** + V2A + *V*<br>Note: V is V+ed / V+ing |
| P11 | $S_{vn}$ | 2.2(2) | $CN_a$ + *RCl* + V2A + $\mathbf{CN_b}$ | $CN_a$ + RP + Pron + *V* + V2A + $\mathbf{CN_b}$ |
| P12 | $VP_{nn}$ | 3.1(1) | V3 + $NP_a$ + $\mathbf{NP_b}$ | V + Prep + $CN_a$ + Prep + $\mathbf{CN_b}$ |
| P13 | $S_{nn}$ | 3.2(1) | $CN_a$ + V2A + $\mathbf{CN_b}$ | $CN_a$ + V2A + $\mathbf{CN_b}$ |
| P14 | $S_{nn}$ | 3.2(2a) | $\mathbf{CN_a}$ + V2A + $CN_b$ | $\mathbf{CN_a}$ + V2A + $CN_b$ |
| P15 | $S_{nn}$ | 3.2(2b) | $\mathbf{CN_a}$ + V2A + $CN_b$ | **CN** + V2A + (Adv) + *A* + *N* |
| P16 | $S_{nn}$ | 3.2(2c) | $\mathbf{CN_a}$ + RCl + V2A + $CN_b$ | **CN** + RCl + V2A + (Adv) + *A* + *N*<br>Note: RCl is any pattern |
| P17 | $CS_{nn}$ | 3.3(1) | $CN_a$ + V2A + $\mathbf{CN_b}$ + (Adv) + Conj + ***Pron***+ V2A + $CN_c$ | $CN_a$ + V2A + $\mathbf{CN_b}$ + (Adv) + Conj + ***Pron*** + V2A + (Adv) + *A* + *N*<br>Note: Pron is a co-reference of $CN_b$ |
| P18 | $CS_{nn}$ | 3.3(2) | $CN_a$ + V2A + $\mathbf{CN_b}$ + Conj + V2A + $CN_c$ | $CN_a$ + V2A + $\mathbf{CN_b}$ + Conj + V2A + (Adv) + *A* + *N* |
| P19 | $VP_{av}$ | 4.1(1) | **VA** + *AP* | **VA** + (Adv) + *A* |
| P20 | $VP_{av}$ | 4.1(2) | V2A + *AP* + **VP** | V2A + (Adv) + *A* + **V** |

Note: italicized words are opinion words; boldface words are aspects; intensifier words are adverbs (Adv); bracketed words are optional; italicized boldface words are co-reference words.

If sentence $S$ matches pattern $p$, three items (an intensifier word, an opinion word, and aspect candidates) are extracted from sentence $S$ as pattern $p$ (Lines 5–7). In these items, a triple $(ac_i, ow_i, iw_i)$ for each aspect candidate is checked if an opinion word is in the OL dictionary. If the triple $(ac_i, ow_i, iw_i)$ is not in the AOI, then the triple is added to the AOI and $tF_i$ is assigned to one (Lines 10–12). If the triple $(ac_i, ow_i, iw_i)$ exists in the AOI, then its $tF_i$ value is increased by one (Line 14). On Line 15, the algorithm returns the AOI.

**Algorithm 1:** Aspect candidate extraction

**Input :** Review R, pattern rules $PRs$, opinion lexicon OL
**Output :** AOI (Aspect-Opinion-Intensifier)

```
1  AOI = < ac_i, ow_i, iw_i, tF_i > /* ac: aspect candidate; ow: opinion word; iw:
     intensifier word, tF: total frequency */
2  for each sentence S in review R do
3      for each rule p in PRs do
4          if S matches p then
5              iw_i ← extract intensifier word from S
6              ow_i ← extract opinion word from S
7              tmpaspect_i ← extract all aspect candidates from S
8              if ow_i exists in OL then
9                  for each aspect candidate ac_i in tmpaspect_i do
10                     if triple (ac_i, ow_i, iw_i) is not in AOI then
11                         add (ac_i, ow_i, iw_i) to AOI
12                         tF_i ← 1
13                     else
14                         tF_i ← tF_i + 1
15 return AOI
```

Some examples show that aspect candidates and useful information (AOI) are extracted by the aspect candidate-extraction function as follows:

• *Extract one aspect candidate from one sentence:*
Given a tagged sentence (S1): "*It*/Pron *is*/V2A *a*/Art *beautiful*/A *picture*/N *./*". In the tagged sentence, "*beautiful*/*A picture*/**N**" matches Pattern Rule P1, in which "*beautiful*" is an adjective and exists in the opinion lexicon (OL). "*picture*" is extracted as an aspect candidate. A triple (picture, beautiful, null) is not in the AOI; thus, (picture, beautiful, null, 1) is added to the AOI.

• *Extract one aspect candidate from one sentence:*
Given a tagged sentence (S2): "*A*/Art *picture*/N *is*/V2A *beautiful*/A *./*". In the tagged sentence, "*A*/**Art** *picture*/**N** *is*/V2A *beautiful*/*A*" matches Pattern Rule P4, in which "*beautiful*" is an adjective and exists in the opinion lexicon (OL). "*picture*" is extracted as an aspect candidate. A triple (picture, beautiful, null) exists in the AOI; thus, $tF_i$ is increased by one for the triple (picture, beautiful, null).

• *Extract two aspect candidates from one sentence:*
Given a tagged sentence (S3): "*I*/Pron *highly*/Adv *recommended*/V *this*/Quant *phone*/N *and*/Conj *lens*/N *./*". In the tagged sentence, "*highly*/Adv *recommended*/ *V this*/**Quant** *phone*/**N** *and*/**Conj** *lens*/**N**" matches Pattern Rule P9, in which "*recommended*" is a verb and exists in the opinion lexicon (OL). The words *"phone", "lens"* are extracted as aspect candidates. Triple (phone, recommended, highly) and (lens, recommended, highly) are not in the AOI; thus, (phone, recommended, highly, 1) and (lens, recommended, highly, 2) are added to the AOI.

• *Extract aspect candidate from a compound sentence:*
Given a tagged sentence (S4): "*My*/Pron *dad*/N *has*/V *a*/Art *phone*/N *and*/Conj *it*/Pron *has*/V *great*/A *functions*/N *./*". In the tagged sentence, "*My*/Pron *dad*/N *has*/V *a*/**Art** *phone*/**N** *and*/Conj *it*/***Pron*** *has*/V *great*/*A functions*/*N*" matches Pattern Rule P17, in which *"great"* is an adjective and exists in the opinion lexicon (OL), *"it"* is the co-reference of *"phone"*. The word "*phone*" is extracted as an aspect candidate. Moreover, in the tagged sentence, "*great*/*A functions*/**N**" matches Pattern Rule P1, in which *"great"* is an adjective and exists in the opinion lexicon (OL). The word *"functions*" is extracted as an aspect candidate. Triple (phone, great, null) and (functions, great, null) are not in the AOI; thus (phone, great, null, 1) and (functions, great, null, 2) are added to the AOI.

The AOIs for for Sentences S1 through S4 are briefed in Table 4. First column $i$ is an index of the aspect candidates, and the rest of the columns show the aspect candidates and their information $(ow_i, iw_i, tF_i)$.

**Table 4**
AOI (aspect-opinion-intensifier) for Sentences S1 through S4

| | AOI | | | |
|---|---|---|---|---|
| $i$ | $ac_i$ | $ow_i$ | $iw_i$ | $tF_i$ |
| 1 | picture | beautiful | null | 2 |
| 2 | phone | recommended | highly | 1 |
| 3 | lens | recommended | highly | 1 |
| 4 | phone | great | null | 1 |
| 5 | functions | great | null | 1 |

## 3.4. Aspect pruning

Choosing aspects from the aspect candidates in order to increase the overall accuracy of the system is an aim of the aspect-pruning function. To choose aspects from the candidates, all aspect candidates are calculated by a semantic similarity score between the candidates and the keywords. To attain a semantic similarity score between two words, we applied cosine similarity and Word2Vec (Word2Vec is a pre-trained model provided by SpaCy [43] with 300 dimensions).

For all aspect candidates, each candidate is calculated by its semantic similarity score with all keywords. After this calculation, a candidate will be eliminated if

its score is lower than a given threshold (similarity threshold $\varphi$). This threshold is a parameter and can be set up by the user.

Keyword KW is a set of words that is used to calculate the similarity scores between words and aspect candidates. The words in KW are built from two resources: 1) expert-based words, and 2) score-based words. The expert-based words are provided by experts, and the score-based words are those words that are chosen from the aspect candidates if their scores are the highest [4]. The formula for calculating the scores is shown in Equation (2):

$$score(a) = f(a) \cdot \sum_i \log_2\left(\left[\frac{f(a, b_i)}{f(a) \cdot f(b_i)}\right] \cdot N + 1\right), \tag{2}$$

where $a$ is the current aspect, *f(a)* is the number of sentences in the corpus where $a$ appeared, and $f(a, b_i)$ is the frequency of the co-occurrence of aspect $a$ and $b_i$ in each sentence. $b_i$ is the $i^{th}$ aspect in the list of aspect candidates, and $N$ is the number of sentences in the corpus.

**Algorithm 2:** Aspect pruning

**Input :** AOI (aspect-opinion-intensifier) = $\{< ac_i, ow_i, iw_i, tF_i >\}$, Keyword KW, similarity threshold $\varphi$, Word2Vec

**Output :** aspect knowledge base AK

```
1  AK ← ∅ /* AK is Aspect Knowledge base */
2  IA ← ∅ /* IA is a set of aspects in which the similarity score is lower than
     the threshold */
3  for each aspect candidate ac_i in AOI do
4      if ac_i is not in IA then
5          if ac_i is not in AK then
6              tSS ← calculate similarities between ac_i and each keyword of KW
                 using cosine similarity and Word2Vec
               /* tSS (temporary Similarity Score) is used to keep similarity
                 scores */
7              if similarity score of tSS ≥ φ then
8                  add (ac_i, ow_i, iw_i, tF_i) to AK
9              else
10                 add ac_i to IA
11         else
12             add (ac_i, ow_i, iw_i, tF_i) to AK
13 return aspect knowledge base AK
```

The *aspect-pruning* algorithm in Algorithm 2 is used to choose aspects from the candidates. Line 1 is used to initialize aspect knowledge base AK. Line 2 is used to initialize a set of aspects IA whose similarity scores are lower than the threshold. The set of the AOI has many members, and there are aspect candidates $ac$ of members that

could be the same. To do the step only one time for those members of the AOI that have the same aspect candidate $ac$, the set of aspects IA is used to keep the aspect candidate that is lower than the threshold. On Lines 3–12, the algorithm chooses aspects from the aspect candidates. At each aspect candidate $ac_i$, if $ac_i$ is not in IA and AK, then the semantic similarity scores with all keywords KW are calculated by using cosine similarity and Word2Vec and kept in tSS (Line 6). The temporary similarity score (tSS) is used to calculate the semantic similarity scores. If the semantic similarity score of tSS is greater than or equal to the given threshold $\varphi$, then the algorithm adds a quadruple $(ac_i, ow_i, iw_i, tF_i)$ to the AK (Lines 7–8); otherwise (i.e., all of the semantic similarity scores are lower than threshold $\varphi$), then $ac_i$ is added to the IA (Line 10). If $ac_i$ is in the AK and not in the IA, then a quadruple $(ac_i, ow_i, iw_i, tF_i)$ is added to the AK (Lines 11–12). In Line 13, aspect knowledge base AK is returned by the algorithm.

The similarity threshold $\varphi$ in Algorithm 2 of the aspect-pruning procedure is used to eliminate irrelevant candidates; the higher this value, the closer the semantic similarities of the two candidates are. Similarity threshold $\varphi$ is set to 0.8 in our study.

## 4. Results and discussion

### 4.1. Dataset

In our experiments, two benchmark datasets for the aspect-extraction task that have been used by many researchers are conducted for evaluation. The first dataset [14] has five review domains (Canon, Nikon, Nokia, MP3 player, and DVD player). The second dataset [25] has three review domains (computer, wireless router, and speaker). The detailed description for each domain is in the format domain_name(domain number, number of sentences, number of aspects). The detailed descriptions for all of the domains are as follows: Canon (D1, 597, 237); Nikon (D2, 346, 174); Nokia (D3, 546, 302); MP3 (D4, 1,716, 674); DVD (D5, 740, 296); Computer (D6, 531, 354); Router (D7, 879, 307); and Speaker (D8, 689, 440).

### 4.2. Experimental results

In this part, we compare the proposed AKGPR framework with other approaches for aspect extraction by using the precision ($P$), recall ($R$), and F1-score ($F1$) measures. $P$, $R$, and $F1$ based on true positives ($TP$), false positives ($FP$), and false negatives ($FN$) have been used by researchers [14, 34, 39]. To calculate these values, we use two sets: $E$ is the set of extracted aspects, and $A$ is the set of annotated aspects in the datasets. These formulas are $P = TP/(TP + FP)$, $R = TP/(TP + FN)$, and $F1 = (2 \cdot \mathrm{P} \cdot \mathrm{R})/(\mathrm{P} + \mathrm{R})$, where $TP$ is $|E \cap A|$, $FP$ is $|E \setminus A|$, and $FN$ is $|A \setminus E|$.

The unsupervised approaches that were used to compare with the proposed AKGPR framework are association rule mining (ARM) [14], semantic-based product feature extraction (SPE) [46], double propagation (DP) [36], DP$^+$ [25], the two-fold rule-based model (TF-RBM) [38], sequential pattern rule (SPR) [39], pattern

knowledge [13], hybrid dependency patterns [18], syntactic patterns [26], heuristic patterns [3], rule-based extraction (RubE) [17], and ontology [20].

Table 5 shows the comparisons of the performance of the experimented approaches for D1 through D8 in terms of (a) precision, (b) recall, and (c) F1-score. Each domain from two benchmark datasets is used to compare the performance of the proposed AKGPR framework with the unsupervised approaches (SPE, DP, DP$^+$, TF-RBM, SPR) shown in Tables 5a, 5b, and 5c. The first columns of the tables are the name of the data (such as D1, D2, etc.). The next columns are the approaches, and the last columns show the proposed AKGPR.

**Table 5**

Comparison of unsupervised approaches for D1–D8 domains:
a) precision; b) recall; c) F1-score

a)

| Number | SPE [46] | DP [25] | DP$^+$ [25] | TF-RBM [34] | SPR [39] | **Proposed AKGPR** |
|---|---|---|---|---|---|---|
| D1 | 0.49 | 0.60 | 0.47 | 0.71 | 0.77 | **0.92** |
| D2 | 0.47 | 0.60 | 0.46 | 0.83 | 0.83 | **0.89** |
| D3 | 0.57 | 0.58 | 0.46 | 0.90 | 0.84 | **0.91** |
| D4 | 0.44 | 0.54 | 0.46 | 0.70 | 0.82 | **0.83** |
| D5 | 0.52 | 0.53 | 0.46 | 0.83 | 0.79 | **0.91** |
| D6 | N/A | 0.63 | 0.52 | N/A | N/A | **0.88** |
| D7 | N/A | 0.55 | 0.43 | N/A | N/A | **0.77** |
| D8 | N/A | 0.56 | 0.44 | N/A | N/A | **0.90** |

b)

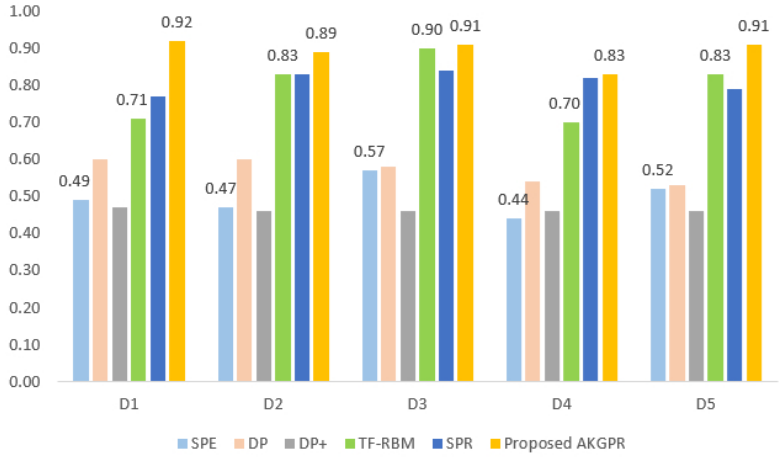
| Number | SPE [46] | DP [25] | DP$^+$ [25] | TF-RBM [34] | SPR [39] | **Proposed AKGPR** |
|---|---|---|---|---|---|---|
| D1 | 0.75 | 0.84 | 0.91 | 0.78 | 0.75 | **0.94** |
| D2 | 0.76 | 0.79 | 0.89 | 0.87 | 0.89 | **0.70** |
| D3 | 0.73 | 0.81 | 0.88 | 0.80 | 0.79 | **0.75** |
| D4 | 0.65 | 0.75 | 0.88 | 0.80 | 0.76 | **0.83** |
| D5 | 0.70 | 0.76 | 0.87 | 0.73 | 0.59 | **0.57** |
| D6 | N/A | 0.78 | 0.88 | N/A | N/A | **0.73** |
| D7 | N/A | 0.85 | 0.94 | N/A | N/A | **0.55** |
| D8 | N/A | 0.81 | 0.91 | N/A | N/A | **0.63** |

c)

| Number | SPE [46] | DP [25] | DP$^+$ [25] | TF-RBM [34] | SPR [39] | **Proposed AKGPR** |
|---|---|---|---|---|---|---|
| D1 | 0.59 | 0.70 | 0.62 | 0.74 | 0.76 | **0.93** |
| D2 | 0.58 | 0.68 | 0.61 | 0.85 | 0.86 | **0.78** |
| D3 | 0.64 | 0.68 | 0.60 | 0.85 | 0.81 | **0.82** |
| D4 | 0.52 | 0.63 | 0.61 | 0.75 | 0.79 | **0.83** |
| D5 | 0.60 | 0.62 | 0.60 | 0.78 | 0.68 | **0.71** |
| D6 | N/A | 0.70 | 0.66 | N/A | N/A | **0.80** |
| D7 | N/A | 0.67 | 0.59 | N/A | N/A | **0.64** |
| D8 | N/A | 0.67 | 0.60 | N/A | N/A | **0.74** |

From Table 5a and Figure 6, our proposed AKGPR framework had the highest precision in all of the domains (D1–D8) when compared with the other unsupervised approaches. From Table 5c, our proposed AKGPR framework had the highest F1-score in D1, D4, D6, and D8 (with values of 0.93, 0.83, 0.80, and 0.74, respectively) when compared with the other unsupervised approaches.

a)




b)

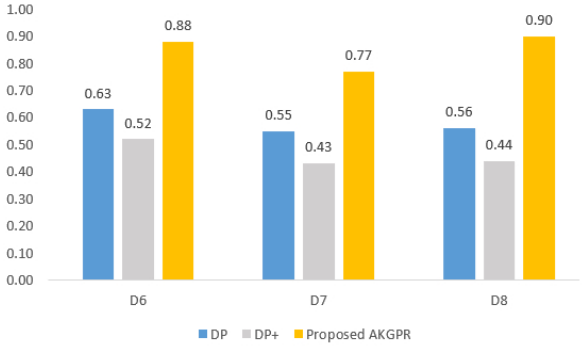



**Figure 6.** Comparison of precision of unsupervised approaches for D1–D8 domains: a) D1–D5 domains; b) D6–D8 domains

The supervised approaches compared among them are the convolutional neural network + linguistic patterns (CNN + LP) [34], CRF$^+$, RSG, RSG$^+$, RSLS and RSLS$^+$ [25], the topic model [9, 45], normalized Google distance + ConceptNet (NGD + CNET) [30, 45], and the improved whale optimization algorithm + pruning algorithm (IWOA + PA) [45].

The comparisons among the supervised approaches are shown in Table 6a, 6b, and 6c. Figure 7 shows the graph of the precision of the supervised approaches (from D1 through D8).

**Table 6**
Comparison of supervised approaches for D1–D8 domains:
a) precision; b) recall; c) F1-score;

a)

| Number | CRF$^+$ [25] | RSG [25] | RSG$^+$ [25] | RSLS [25] | RSLS$^+$ [25] | CNN + LP [34] | Topic model [45] | NGD + CNET [45] | IWOA + PA [45] |
|---|---|---|---|---|---|---|---|---|---|
| D1 | 0.73 | 0.83 | 0.81 | 0.83 | 0.80 | 0.93 | 0.85 | 0.88 | 0.91 |
| D2 | 0.70 | 0.83 | 0.83 | 0.86 | 0.86 | 0.82 | 0.89 | 0.81 | 0.91 |
| D3 | 0.76 | 0.76 | 0.76 | 0.76 | 0.77 | 0.90 | 0.87 | 0.91 | 0.93 |
| D4 | 0.70 | 0.72 | 0.69 | 0.73 | 0.70 | 0.92 | 0.88 | 0.90 | 0.93 |
| D5 | 0.64 | 0.75 | 0.78 | 0.77 | 0.78 | 0.93 | 0.88 | 0.89 | 0.92 |
| D6 | 0.64 | 0.78 | 0.76 | 0.74 | 0.78 | N/A | N/A | N/A | N/A |
| D7 | 0.71 | 0.69 | 0.72 | 0.70 | 0.72 | N/A | N/A | N/A | N/A |
| D8 | 0.69 | 0.71 | 0.73 | 0.71 | 0.74 | N/A | N/A | N/A | N/A |

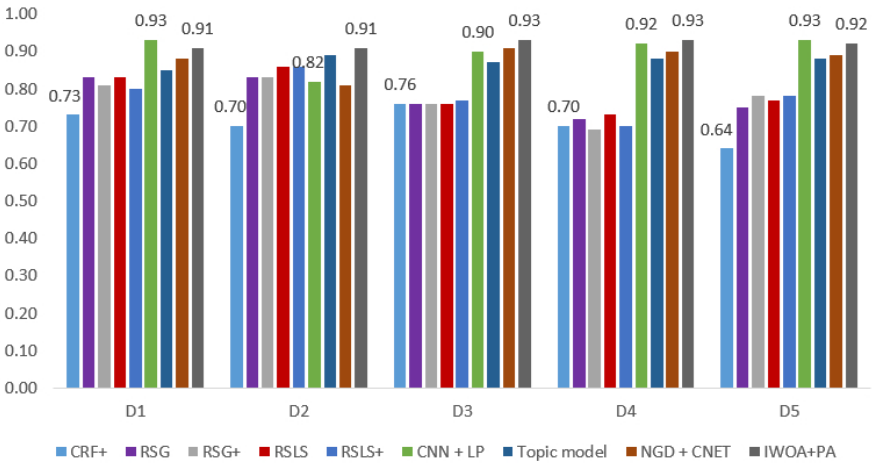
b)

| Number | CRF$^+$ [25] | RSG [25] | RSG$^+$ [25] | RSLS [25] | RSLS$^+$ [25] | CNN + LP [34] | Topic model [45] | NGD + CNET [45] | IWOA + PA [45] |
|---|---|---|---|---|---|---|---|---|---|
| D1 | 0.58 | 0.74 | 0.75 | 0.75 | 0.76 | 0.85 | 0.84 | 0.80 | 0.93 |
| D2 | 0.76 | 0.76 | 0.83 | 0.75 | 0.85 | 0.87 | 0.87 | 0.85 | 0.92 |
| D3 | 0.56 | 0.67 | 0.73 | 0.69 | 0.79 | 0.84 | 0.90 | 0.87 | 0.93 |
| D4 | 0.57 | 0.66 | 0.78 | 0.67 | 0.77 | 0.86 | 0.87 | 0.87 | 0.95 |
| D5 | 0.54 | 0.58 | 0.68 | 0.58 | 0.67 | 0.88 | 0.90 | 0.82 | 0.91 |
| D6 | 0.56 | 0.69 | 0.74 | 0.74 | 0.71 | N/A | N/A | N/A | N/A |
| D7 | 0.61 | 0.74 | 0.79 | 0.77 | 0.79 | N/A | N/A | N/A | N/A |
| D8 | 0.57 | 0.67 | 0.69 | 0.70 | 0.70 | N/A | N/A | N/A | N/A |

c)

| Number | CRF$^+$ [25] | RSG [25] | RSG$^+$ [25] | RSLS [25] | RSLS$^+$ [25] | CNN + LP [34] | Topic model [45] | NGD + CNET [45] | IWOA + PA [45] |
|---|---|---|---|---|---|---|---|---|---|
| D1 | 0.65 | 0.78 | 0.78 | 0.79 | 0.78 | 0.88 | 0.84 | 0.84 | 0.92 |
| D2 | 0.73 | 0.79 | 0.83 | 0.80 | 0.85 | 0.84 | 0.88 | 0.83 | 0.91 |
| D3 | 0.64 | 0.71 | 0.75 | 0.73 | 0.78 | 0.87 | 0.88 | 0.89 | 0.93 |
| D4 | 0.63 | 0.69 | 0.73 | 0.70 | 0.73 | 0.89 | 0.87 | 0.88 | 0.94 |
| D5 | 0.58 | 0.65 | 0.73 | 0.66 | 0.72 | 0.90 | 0.89 | 0.85 | 0.91 |
| D6 | 0.60 | 0.73 | 0.75 | 0.74 | 0.74 | N/A | N/A | N/A | N/A |
| D7 | 0.66 | 0.72 | 0.75 | 0.73 | 0.75 | N/A | N/A | N/A | N/A |
| D8 | 0.62 | 0.69 | 0.71 | 0.71 | 0.72 | N/A | N/A | N/A | N/A |

a)




b)

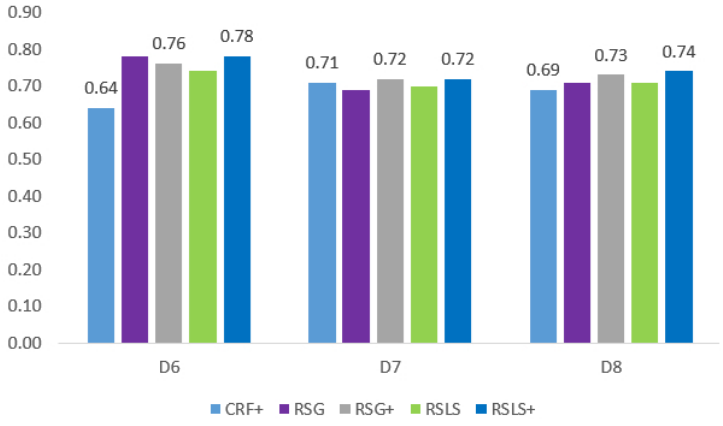



**Figure 7.** Comparison of precision of supervised approaches for D1–D8 domains: a) D1–D5 domains; b) D6–D8 domains

Table 7 shows the performance of the precision, recall, and F1-scores among the proposed AKGPR and the other unsupervised approaches for the group of domains. The*Data* of each group is shown in the second column. The proposed AKGPR has the highest precision in all groups. For the recall of Group 1, the proposed AKGPR has the highest value (0.82). For the F1-score, the proposed AKGPR has the highest value for Groups 1, 2, 3, and 5 (0.85, 0.82, 0.84, and 0.73, respectively).

**Table 7**
Comparison of proposed AKGPR with other unsupervised approaches

| Group No. | Data | Approach | Precision | Recall | F1-Score |
|---|---|---|---|---|---|
| 1 | D1, D2, D4 | Syntactic Patterns [26] | 0.63 | 0.73 | 0.67 |
| | | **Proposed AKGPR** | **0.89** | **0.82** | **0.85** |
| 2 | D1, D3, D4, D5 | Ontology [20] | 0.79 | 0.79 | 0.79 |
| | | **Proposed AKGPR** | **0.89** | **0.77** | **0.82** |
| 3 | D1, D2, D3, D4 | Pattern knowledge [13] | 0.73 | 0.86 | 0.79 |
| | | Hybrid dependency patterns [18] | 0.79 | 0.72 | 0.75 |
| | | Heuristic Patterns [3] | 0.83 | 0.71 | 0.77 |
| | | **Proposed AKGPR** | **0.89** | **0.81** | **0.84** |
| 4 | D1, D2, D3, D4, D5 | ARM [14] | 0.80 | 0.72 | 0.76 |
| | | SPE [46] | 0.49 | 0.72 | 0.59 |
| | | DP [25] | 0.57 | 0.79 | 0.66 |
| | | $DP^+$ [25] | 0.46 | 0.89 | 0.61 |
| | | RubE [17] | 0.88 | 0.87 | 0.88 |
| | | TF-RBM [38] | 0.79 | 0.80 | 0.79 |
| | | SPR [39] | 0.81 | 0.76 | 0.78 |
| | | **Proposed AKGPR** | **0.89** | **0.76** | **0.81** |
| 5 | D6, D7, D8 | DP [25] | 0.58 | 0.81 | 0.68 |
| | | $DP^+$ [25] | 0.46 | 0.91 | 0.68 |
| | | **Proposed AKGPR** | **0.85** | **0.64** | **0.73** |

## 5. Conclusion

In this work, the aspect knowledge-base generation using pattern rules (AKGPR) framework to automatically generate an aspect knowledge base from social media is proposed in order to support sentiment-summarization systems. The proposed AKGPR framework could extract aspects in the forms of a single noun, a noun phrase, and a verb (along with their useful information). With the proposed AKGPR framework, two kinds of aspects (single and multi-word) are extracted by using the pattern rules. Tagging the POS for each word and running the misspelling-correction procedure are needed before extracting the aspects. The semantic similarity-pruning method is used to choose aspects from aspect candidates to be in the aspect knowledge base. The proposed AKGPR framework has the highest performance in terms of its precision when compared to the other unsupervised approaches. The pattern rules of the proposed AKGPR framework (which can be applied to different domains; e.g., business, tourism, etc.) do not need to incur any costs for the annotated datasets in the training phase. In future work, we plan to extend the patterns with slang words and emoticons.

## Acknowledgements

*We would like to express our thanks to the Division of Computational Science, Faculty of Science, Prince of Songkla University, for supporting our research.*

## References

[1] Adela L., Ulfeta M.: Improving sentiment analysis for twitter data by handling negation rules in the Serbian language, *Computer Science and Information Systems*, vol. 16(1), pp. 289–311, 2019.

[2] Aichner T., Jacob F.: Measuring the Degree of Corporate Social Media Use, *International Journal of Market Research*, vol. 57(2), pp. 257–276, 2015.

[3] Asghar M.Z., Khan A., Zahra S.R., Ahmad S., Kundi F.M.: Aspect-based opinion mining framework using heuristic patterns, *Cluster Computing*, vol. 22, pp. 7181–7199, 2017.

[4] Bagheri A., Saraee M., de Jong F.: Care more about customers: Unsupervised domain-independent aspect detection for sentiment analysis of customer reviews, *Knowledge-Based Systems*, vol. 52, pp. 201–213, 2013.

[5] Bing L.: *Sentiment Analysis: Mining Opinions, Sentiments, and Emotions*, Cambridge University Press, New York, USA, 2015.

[6] Clark E., Araki K.: Text Normalization in Social Media: Progress, Problems and Applications for a Pre-Processing System of Casual English, *Procedia – Social and Behavioral Sciences*, vol. 27, pp. 2–11, 2011.

[7] Dedić N., Stanier C.: Measuring the Success of Changes to Existing Business Intelligence Solutions to Improve Business Intelligence Reporting. In: *Research and Practical Issues of Enterprise Information Systems*, pp. 225–236, Cham 2016.

[8] Erşahin B., Aktaş Ö., Kilinç D., Erşahin M.: A hybrid sentiment analysis method for Turkish, *Turkish Journal of Electrical Engineering & Computer Sciences*, vol. 27(3), pp. 1780–1793, 2019.

[9] Feng J., Yang W., Gong C., Li X., Bo R.: Product Feature Extraction via Topic Model and Synonym Recognition Approach. In: *Big Data*, pp. 73–88, Springer Singapore, 2019.

[10] Freeman A.T., Condon S.L., Ackerman C.M.: Cross Linguistic Name Matching in English and Arabic: A "One to Many Mapping" Extension of the Levenshtein Edit Distance Algorithm. In: *Proceedings of the Main Conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics*, pp. 471–478, HLT-NAACL '06, USA, 2006.

[11] Gerani S., Carenini G., Ng R.T.: Modeling content and structure for abstractive review summarization, *Computer Speech & Language*, vol. 53, pp. 302–331, 2019.

[12] Gliwa B., Zygmunt A., Dąbrowski M.: Building sentiment lexicons based on recommending services for the Polish language, *Computer Science*, vol. 17(2), pp. 163–185, 2016.

[13] Htay S.S., Lynn K.T.: Extracting product features and opinion words using pattern knowledge in customer reviews, *The Scientific World Journal*, vol. 2013, pp. 1–5, 2013.

[14] Hu M., Liu B.: Mining and Summarizing Customer Reviews. In: *Proceedings of the 10th International Conference on Knowledge Discovery and Data Mining (SIGKDD)*, pp. 168–177, USA, 2004.

[15] Jakob N., Gurevych I.: Extracting Opinion Targets in a Single- and Cross-Domain Setting with Conditional Random Fields, 2010.

[16] Jin W., Ho H.H., Srihari R.K.: OpinionMiner: a novel machine learning system for web opinion mining and extraction. In: *Proceedings of the 15th International Conference on Knowledge Discovery and Data Mining (SIGKDD)*, pp. 1195–1204, ACM, USA, 2009.

[17] Kang Y., Zhou L.: RubE: Rule-based methods for extracting product features from online consumer reviews, *Information & Management*, vol. 54(2), pp. 166–176, 2017.

[18] Khan K., Baharudin B., Khan A.: Identifying Product Features from Customer Reviews Using Hybrid Patterns, *International Arab Journal of Information Technology*, vol. 11(3), pp. 281–286, 2014.

[19] Kherwa P., Sachdeva A., Mahajan D., Pande N., Singh P.K.: An approach towards comprehensive sentimental data analysis and opinion mining. In: *2014 IEEE International Advance Computing Conference (IACC)*, pp. 606–612, 2014.

[20] Konjengbam A., Dewangan N., Kumar N., Singh M.: Aspect ontology based review exploration, *Electronic Commerce Research and Applications*, vol. 30, pp. 62–71, 2018.

[21] Lazhar F.: Implicit feature identification for opinion mining, *Internation Journal of Business Information Systems*, vol. 30(1), pp. 13–30, 2019.

[22] Li S., Zhou L., Li Y.: Improving aspect extraction by augmenting a frequency-based method with web-based similarity measures, *Information Processing & Management*, vol. 51(1), pp. 58–67, 2015.

[23] Li X., Bing L., Li P., Lam W., Yang Z.: Aspect Term Extraction with History Attention and Selective Transformation. In: *Proceedings of the 27th International Joint Conference on Artificial Intelligence (IJCAI'18)*, pp. 4194–4200, 2018.

[24] Li X., Lam W.: Deep Multi-Task Learning for Aspect Term Extraction with Memory Interaction. In: *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing (EMNLP 2017)*, pp. 2886–2892, Copenhagen, Denmark, 2017.

[25] Liu Q., Gao Z., Liu B., Zhang Y.: Automated rule selection for opinion target extraction, *Knowledge-Based Systems*, vol. 104(15), pp. 74–88, 2016.

[26] Maharani W., Widyantoro D.H., Khodra M.L.: Aspect Extraction in Customer Reviews Using Syntactic Pattern, *Procedia Computer Science*, vol. 59, pp. 244–253, 2015.

[27] Mai L., Le B.: Aspect-Based Sentiment Analysis of Vietnamese Texts with Deep Learning. In: *Intelligent Information and Database Systems (ACIIDS 2018). Lecture Notes in Computer Science*, vol. 10751, pp. 149–158, Springer, 2018.

[28] Marstawi A., Sharef N.M., Aris T.N.M., Mustapha A.: Ontology-Based Aspect Extraction for an Improved Sentiment Analysis in Summarization of Product Reviews (ICCMS '17). pp. 100–104, 2017.

[29] Mataoui M., Hacine T.E.B., Tellache I., Bakhtouchi A., Zelmati O.: A new syntax-based aspect detection approach for sentiment analysis in Arabic reviews. In: *Proceedings of the 2nd International Conference on Natural Language and Speech Processing (ICNLSP)*, pp. 1–6, 2018.

[30] Nawaz A., Asghar S., Naqvi S.H.: A segregational approach for determining aspect sentiments in social media analysis, *The Journal of Supercomputing*, vol. 75(5), pp. 2584–2602, 2019.

[31] Niwattanakul S., Singthongchai J., Naenudorn E., Wanapu S.: Using of Jaccard Coefficient for Keywords Similarity. In: *International MultiConference of Engineers and Computer Scientists (IMECS 2013)*, vol. I, 2013.

[32] Parlar T., Ozel A.S., Song F.: Analysis of data pre-processing methods for the sentiment analysis of reviews, *Computer Science*, vol. 20(1), pp. 123–141, 2019.

[33] Piao Z., Park S.M., On B.W., Choi G.S., Park M.S.: Product reputation mining: bring informative review summaries to producers and consumers, *Computer Science and Information Systems*, vol. 16(2), pp. 359–380, 2019.

[34] Poria S., Cambria E., Gelbukh A.: Aspect extraction for opinion mining with a deep convolutional neural network, *Knowledge-Based Systems*, vol. 108(15), pp. 42–49, 2016.

[35] Poria S., Cambria E., Ku L.W., Gui C., Gelbukh A.: A Rule-Based Approach to Aspect Extraction from Product Reviews. In: *Proceedings of the 2nd Workshop on Natural Language Processing for Social Media (SocialNLP)*, pp. 28–37, 2014.

[36] Qiu G., Liu B., Bu J., Chen C.: Opinion Word Expansion and Target Extraction through Double Propagation, *Computational Linguistics*, vol. 37(1), pp. 9–27 2011.

[37] Ramamonjisoa D., Murakami R., Chakraborty B.: Comments Analysis and Visualization Based on Topic Modeling and Topic Phrase Mining. In: *Proceedings of the 3rd International Conference on E-technologies and Business on the Web (EBW2015)*, p. 1, 2015.

[38] Rana T.A., Cheah Y.N.: A two-fold rule-based model for aspect extraction, *Expert Systems with Applications*, vol. 89(15), pp. 273–285, 2017.

[39] Rana T.A., Cheah Y.N.: Sequential patterns rule-based approach for opinion target extraction from customer reviews, *Journal of Information Science*, vol. 45(5), pp. 643–655, 2019.

[40] Ranta A.: *Grammatical Framework: Programming with Multilingual Grammars*, CSLI Publications, Center for the Study of Language and Information, 2011.

[41] Samanta P., Chaudhuri B.B.: A simple real-word error detection and correction using local word bigram and trigram. In: *Proceedings of the 25th Conference on Computational Linguistics and Speech Processing (ROCLING 2013)*, pp. 211–220, Taiwan, 2013.

[42] Singh S.K., Sachan M.K.: SentiVerb system: classification of social media text using sentiment analysis, *Multimedia Tools and Applications*, vol. 78(22), pp. 31109–32136, 2019.

[43] Spacy: Spacy Guides, 2020. https://spacy.io/.

[44] Tran T.A., Duangsuwan J., Wettayaprasit W.: A Novel Automatic Sentiment Summarization from Aspect-based Customer Reviews. In: *Proceedings of the 15th International Joint Conference on Computer Science and Software Engineering (JCSSE)*, pp. 1–6, Nakhonpathom, Thailand, 2018.

[45] Tubishat M., Idris N., Abushariah M.: Explicit aspects extraction in sentiment analysis using optimal rules combination, *Future Generation Computer Systems*, vol. 114, pp. 448–480, 2021.

[46] Wei C.P., Chen Y.M., Yang C.S., Yang C.C.: Understanding what concerns consumers: a semantic approach to product feature extraction from consumer reviews, *Information Systems and e-Business Management*, vol. 8(2), pp. 149–167, 2010.

[47] Wilson T., Wiebe J., Hoffmann P.: Recognizing contextual polarity in phrase-level sentiment analysis. In: *HLT '05: Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing October 2005*, pp. 347–354, 2005.

[48] Yin W., Kann K., Yu M., Schütze H.: Comparative Study of CNN and RNN for Natural Language Processing, *arXiv Preprint*, vol. arXiv: 1702.01923, 2017.

## Affiliations

**Tuan Anh Tran**
Prince of Songkla University, Faculty of Science, Artificial Intelligence and Informatics Innovations ($AI^3$) Research Lab, Division of Computational Science, Hat Yai, 90110, Thailand, tattinsphue@yahoo.com

**Jarunee Duangsuwan**
Prince of Songkla University, Faculty of Science, Artificial Intelligence and Informatics Innovations ($AI^3$) Research Lab, Division of Computational Science, Hat Yai, 90110, Thailand, jarunee.d@psu.ac.th

**Wiphada Wettayaprasit**
Prince of Songkla University, Faculty of Science, Artificial Intelligence and Informatics Innovations ($AI^3$) Research Lab, Division of Computational Science, Hat Yai, 90110, Thailand, wiphada.w@psu.ac.th

**Received**: 28.10.2020
**Revised**: 13.07.2021
**Accepted**: 18.09.2021