Expert Systems With Applications 114 (2018) 267–288

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

Extracting sentiment knowledge from pros/cons product reviews: Discovering features along with the polarity strength of their associated opinions

Monireh Alsadat Mirtalaie[a,*], Omar Khadeer Hussain[a], Elizabeth Chang[a], Farookh Khadeer Hussain[b]

[a] School of Business, University of New South Wales, Building 27, Level 1, Room 106, Canberra 2610, Australia
[b] Centre for Artificial Intelligence, School of Software, Faculty of Engineering and IT, University of Technology, Sydney, Australia

ARTICLE INFO

Article history:
Received 28 April 2018
Revised 21 June 2018
Accepted 20 July 2018
Available online 22 July 2018

Keywords:
Feature-based sentiment analysis
Pros/cons product reviews
Syntactic rules
Polarity strength
Product tree

ABSTRACT

Sentiment knowledge extraction is a growing area of research in the literature. It helps in analyzing users' opinions about different entities or events, which can then be utilized by analysts for various purposes. Particularly, feature-based sentiment analysis is one of the challenging research areas that analyzes users' opinions on various features of a product or service. Of the three formats for the product reviews, our focus in this paper is limited to analyzing the pros/cons type. Due to the nature of pros/cons reviews, they are mostly concise and follow a different structure from other review types. Therefore, specialized techniques are needed to analyze these reviews and extract the customers' discussed product features along with their personal attitudes. In this paper, we propose the Pros/Cons Sentiment Analyzer (PCSA) framework that exploits dependency relations in extracting sentiment knowledge from pros/cons reviews. We also utilize two different lexicons to ascertain the polarity strength of the extracted features based on the customers' opinions. Several experiments are conducted to evaluate the performance of PCSA in its different phases.

1. Motivation

With the continued growth of user-contributed data, *social media analytics* have, of late, received immense attention. The objective of research in this area is to tap into social media information either at a *macro* or *micro* level, extract insights from this information and apply it to various further analysis. In current literature, the observed insights from social media have been applied in areas such as marketing (Norman & Ann, 2011), crime detection (Bright, 2017), epidemic management, policy making (Xie et al., 2013) and so forth. The reason for the continued growth of social media analytics is not only related to the reduced cost and time required to extract such insights compared to previously utilized methods such as focus groups (Pasierbinska-Wilson, 2016) but also because it provides access to the opinions of a wider and more diverse group of people who engage in social media. However, depending on the purpose of an organization, an appropriate method and tool needs to be developed before it can reap the benefits of social media analytics.

Our objective in this paper is to develop an approach for tapping into the knowledge embedded in the product reviews. In other words, the broad objective of this study is to develop an automatic tool that utilizes customers' reviews related to a product and ascertains the customers' feelings toward various features of that product. Product reviews are broadly presented in three formats. The first format is known as the *pros/cons* type in which customers are given a template on which to record their different likes and dislikes in relation to a product. The second format is the *detailed* type on which the customers, in addition to reporting on the pros and cons of a product, also write a detailed opinion in the provided fields. The third one is the *free* format that does not have any fields or template and in which the customer expresses their opinion either as a pro/con and/or as a detailed type. As the structure of each review type is different, a specialized technique is required to analyze them (Liu & Hu, 2005). Our focus in this paper is limited to extracting sentiment knowledge from the *pros/cons* review type. The objective of knowledge extraction is to discover the product features discussed by the customers along with their opinions reported in the pros/cons reviews,

* Corresponding author.
E-mail addresses: m.mirtalaie@unsw.edu.au (M.A. Mirtalaie), o.hussain@adfa.edu.au (O.K. Hussain), e.chang@adfa.edu.au (E. Chang), farookh.hussain@uts.edu.au (F.K. Hussain).

https://doi.org/10.1016/j.eswa.2018.07.046
0957-4174/

assessing their polarity strength and determining the overall sentiment value of different features being discussed. Although the existing literature reports several techniques for sentiment knowledge extraction, they mostly focus on analyzing the detailed type. As the pros/cons reviews contain incomplete sentences and short phrases, the techniques which have been developed for the detailed type are not suitable for extracting knowledge from this format (Liu & Hu, 2005). Apart from the specific grammatical structure of this review type, for the following reasons, the state-of-the-art methods in detailed type cannot be utilized in mining sentiment knowledge from the pros/cons type:

- *Challenges associated with defining seed words and a lack of opinion words in pros/cons type reviews:* Existing approaches utilize syntactic rules to mine features and personal attitudes from the detailed reviews (Kang & Zhou, 2016; Liu, Gao, Liu, & Zhang, 2015; Poria, Cambria, Ku, Gui, & Gelbukh, 2014). They start with utilizing some general adjectives (opinions) such as *good, bad,* and the like as seed words, which are then used to extract the product's features along with more opinion words. However, a challenge in pros/cons reviews is that they may not contain any opinion words at all. In other words, due to the specific format of these reviews (providing separate spaces for mentioning likes and dislikes), the customers may not use any opinion words in their comments. For example, they might only write 'screen' instead of 'good screen' in a pros review. Therefore, the mentioned approach cannot be applied in analyzing them. Another way is to initiate the approach using some of the product's features as the seed words. However, since different products vary in their features, unique seed words need to be defined in each case, thus making the process cumbersome.
- *Pruning redundant results:* A typical step in almost all of the studies in feature-based sentiment analysis is to prune the results once the extraction task is over. The main objective of this step is to remove the redundant mined features. Different techniques such as global pruning (Qiu, Liu, Bu, & Chen, 2011) have been developed to accomplish this task. Due to the brevity of customers' comments in pros/cons type reviews, approaches that are applicable and effective for this type of review are needed.

Pros/cons type reviews have been analyzed in some of the studies for sentiment knowledge extraction as they contain a rich resource of product features (Ganapathibhotla & Liu, 2008; Liu & Hu, 2005; Yu, Zha, Wang, & Chua, 2011). However, they have the following shortcomings:

- *Determining the intensity of emotions embedded in the pros/cons reviews:* Unlike the approaches developed for sentiment knowledge extraction on detailed reviews, existing studies on the pros/cons reviews do not extract the opinion words associated with the features mentioned in the reviews. They argue that pros and cons reviews inherit the positive or negative polarity, respectively. In this way, they assign an overall polarity value of either $+1$ or -1 to these reviews (Ganapathibhotla & Liu, 2008). However, this does not capture the different level/s of emotions expressed by customers provided that they use opinion words in their comments (Fang & Zhan, 2015). For example, in writing a pros review, if a customer uses a sentiment word such as *excellent* to state his personal attitude toward a feature, the level of satisfaction expressed by that customer would be higher compared to the case that he utilizes the word *good* in his comment. This situation is more complicated when the customer mentions no opinion word in his review. For more clarfication, in our earlier example, the positive association that the phrase 'good screen' carries is stronger than the case where the customer just mentions 'screen' as a liked feature in a pros review without expressing any sentiment word. New methods are needed that not only extract the opinion words from pros/cons reviews but also determine their polarity strength according to the intensity of the emotion embedded in them.
- *Dealing with negative (cons) reviews:* Existing approaches that extract features using grammatical relationships define a single set of rules for analyzing both the *pros* and *cons* reviews. As we explain in this paper, it does not result in a high level of accuracy and needs to be addressed.

In this paper, we propose the ***P***ros/***C***ons ***S***entiment ***A***nalyzer (PCSA) framework that addresses the aforementioned shortcomings. We focus on customer opinions that are expressed in the pros/cons review format and utilize NLP techniques to mine product features along with their associated personal attitudes. To improve the accuracy of the extraction process, we add a double propagation step that broadens the scope of searching for features and opinion words in the reviews. After the features and opinion words are extracted, we utilize the product tree in order to prune redundant results. Finally, we propose an approach to determine the polarity strength of each extracted feature on a continuous scale of -1 to 1 according to the intensity of the associated opinion. The paper is organized as follows. Section 2 presents a review of the literature for three different sentiment knowledge extraction tasks, namely feature/opinion extraction, polarity determination and sentiment aggregation. Section 3 presents our proposed PCSA framework, its different phases, and the working of each of them in detail. The experiment results in Section 4 demonstrate the accuracy of different phases of the PCSA framework in sentiment knowledge extraction against the results generated by the human mind and also an existing approach. Section 5 concludes the paper with a discussion on future work.

2. Literature review

Sentiment analysis or opinion mining has received significant attention in recent times and is generally done at three levels, namely *document, sentence and aspect (or feature).* Document-level analysis determines the overall sentiment orientation of a document. Such an approach is useful if the document contains information about a single feature (Araque, Corcuera-Platas, Sánchez-Rada, & Iglesias, 2017; Manek, Shenoy, Mohan, & Venygopal, 2016). On the other hand, in the sentence-level analysis, each sentence is first checked for being subjective or objective. Then, in case of having a subjective sentence, its positive or negative orientation is determined (Liu, 2012). Aspect or feature-level techniques perform finer-grained analysis to first find the feature-opinion pairs in a given sentence and then ascertain their sentiment classifications (Schouten & Frasincar, 2016). This approach is based on the fact that a user may express his personal attitudes about more than one feature of an entity (product/service) in a sentence, hence the main targets of the customers' opinions should be identified, separately (Ganeshbhai & Shah, 2015).

The process of feature-level sentiment analysis can be divided into three tasks namely, *feature-opinion extraction, polarity determination, and results aggregation* (Schouten & Frasincar, 2016). The first task mines feature-opinion pairs in the documents. Polarity determination classifies the expressed opinions based on its positive or negative orientation. Finally, to obtain an incisive overview, the sentiment value for each aspect is aggregated in the third step (Mirtalaie, Hussain, Chang, & Hussain, 2017b; Schouten & Frasincar, 2016). The existing work in the literature on each task is discussed in detail in the next subsections.

2.1. Feature-opinion extraction

Approaches to identify features and their related opinions in a document can be divided into four categories, namely frequency-based, supervised machine learning methods, semi-supervised or unsupervised machine learning methods, and rule-based methods (Liu, 2015; Schouten & Frasincar, 2016; Suleman & Vechtomova, 2015).

Frequency-based approaches consider frequently used nouns to most likely represent the features in a document. Hu and Liu (2004) utilize association rule mining to discover the frequent item sets in the online reviews.Manek et al. (2016) exploit term frequency to extract features from the users' comments. To enhance the accuracy of the results, the Gini Index is used to measure the impurity of extracted aspects. Chen, Ferrer, Wiratunga, and Plaza (2015) developed a hybrid approach that is a combination of the frequency-based method and supervised information gain for feature extraction. Frequency-based methods search for the most frequently discussed features of a product whereas supervised information gain identifies product features that have the greatest discriminatory power in determining the top-ranked products. They check the validity of their method using reviews on three different digital cameras (Chen et al., 2015). However, a shortcoming of this approach is its inability to identify features that are not frequently mentioned by users.

In the *supervised machine learning* category of methods, Samha, Li, and Zhang (2015) proposed a Conditional Random Field (CRF)-based approach to extract product features and their corresponding opinions. Using a tag list, the dataset is labeled before removing abnormal characters and HTML tags to enhance the results. Next, OpenNLP is employed to detect and split sentences before tagging with parts of speech (POS) terms and training the CRF model on the dataset. Two different datasets from shopping websites are used to validate the model. Liu and Hu (2005) proposed a supervised pattern mining model for analyzing pros/cons reviews. The reviews are split into segments before manually tagging each word with the POS as well as the product's features. Using the association rule mining technique, rules that are helpful in extracting features from product reviews are developed. Alghunaim, Mohtarami, Cyphers, and Glass (2015) developed a vector-based supervised algorithm to extract features from a document by using a support vector machine. In (Kim & Hovy, 2006), pros/cons reviews were utilized to find the main reason of customers' like/dislikes of a product. They first break each pros/cons review to its comma delimited phrases. Then, by querying the extracted phrases in the detailed reviews, they find the corresponding full sentence to each of the pros/cons phrases. Finally, a Maximum Entropy model is trained based on the collected data to find the pros/cons from the free-style reviews. The major drawback of supervised techniques stems from their pre-requisite to have an annotated database which is both costly and time-consuming (Suleman & Vechtomova, 2015).

To address the aforementioned issues, *semi-supervised or unsupervised machine learning* methods are used. In relation to these methods, Bagheri, Saraee, and De Jong (2013) proposed a language-independent model to detect explicit and implicit features of products and their sentiment polarities in online product reviews. The proposed model first detects comments in separate sentences and tags each of them into their POS to find initial candidates for product features. It then searches for multi-word features using the generalized FLR (Frequencies and Left and Right of the current word) method. Heuristic rules are used to filter less informative aspects. Some features are selected as a seed set for the bootstrapping algorithm that uses both frequency and inter-related information between words to score features. Two feature pruning thresholds are introduced to remove redundant words. Finally, a graph-based approach is utilized to extract implicit features. Huang, Etzioni, Zettlemoyer, Clark, and Lee (2012) proposed a smartphone application or RevMiner to extract feature-opinion pairs from a large corpus of user reviews that represent a structured representation of people's opinions on restaurants. The proposed algorithm involves the following major steps: in the first step, using an automated bootstrapping method, features, and their ratings from the corpus are identified. In the second step, to determine the polarity of each feature, the average of the Yelp ratings for all the reviews containing that feature is computed. In the third step, they defined five clusters and group all the related features together to form categories used in the Color Bar interface. Carter and Inkpen (2015) proposed a new confidence-based co-training algorithm to extract product features along with their related sentiments from customers' reviews. Compared to the other approaches, the developed approach requires a smaller amount of labeled data in the training phase. In the co-training approach, two classifiers work on the same data using different variables. In this study, collocations and syntactic relations are considered as two variables for classifying the data. The support vector machine is utilized as a method for sentiment classification. A case study using customer reviews on restaurants and laptops was chosen to evaluate the algorithm's performance. Although the semi- or unsupervised techniques require no or less labeled training data, one of their pre-requisites is to provide opinions as seed words. As discussed in Section 1, this might work for detailed type reviews; however, since pros/cons type of reviews may not contain any opinion words, it might not be practical to apply these approaches on them.

Rule-based methods utilise the grammatical relationships between different words in a sentence to extract the features and their corresponding opinion words. Poria et al. (2014) developed dependency rules using SenticNet to extract explicit product features from online reviews. They also utilize an existing labeled corpus to extract implicit features along with their categories. Kang and Zhou (2016) proposed a rule-based method to extract product features from both subjective and objective online reviews. After pre-processing online reviews, the proposed algorithm extracts features in subjective sentences using a double propagation approach. They developed rules to discover product features in comparative sentences. To handle objective sentences, syntactic rules are defined to detect the part-whole relation of product features in online reviews. Different pruning strategies are proposed to improve the precision of feature extraction. Qiu et al. (2011) developed a propagation approach to iteratively mine features and opinions in a given document. To bootstrap the algorithm, a seed sentiment lexicon is used. Using syntactic rules, the algorithm extracts more features and opinion words iteratively. Zhuang, Jing, and Zhu (2006) proposed a hybrid rule-based method to extract features and opinions from movie reviews. Utilizing the frequency-based approach, the candidate features are first mined as seed words. Then, synonyms for each extracted feature are identified using WordNet. Finally, the syntactic rules are mined by investigating the dependencies between different identified seed words in the reviews. Liu et al. (2015) proposed an automated rule selection algorithm that selects a subset of the most accurate rules to identify product features from social media content. To test the accuracy of the rules, they were evaluated based on their precision and recall. Finally, the F1-score is used as the performance evaluation measure to select the most powerful rules in feature extraction. In another study by Ganapathibhotla and Liu (2008), a ruled-based method was proposed to extract the preferred entity from the customers' reviews. They first manually extract the comparative sentences from both detailed and pros/cons reviews. Next, they defined rules that help in extracting the customer's preferred product (entity) from different types of the comparative sentences. A two-fold rule based method was proposed by Rana and Cheah (2017) for extracting features from the product reviews (Rana & Cheah, 2017). In the first fold, they developed grammatical rules that are capable of extracting domain independent feature phrases. However, in the second fold, they extracted the domain dependent features using SenticNet 4 lexicon. They utilize both the frequency and similarity-based approaches to prune the non-related results. Liu and Seneff (2009) proposed a rule-based framework for analyzing pros/cons restaurant reviews. They utilized linguistic patterns to extract related combinations of adverbs, adjectives and nouns from reviews. Using the LM-Based topic clustering model, they then automatically group the extracted features into meaningful clusters. To compute the polarity, they prepare a list of adjectives and adverbs. They then consider the average of the detailed reviews' ratings where they contain an adjective/adverb and question the polarity strength of it. Finally, depending on whether the adjective conveys a positive attitude or a negative one, they follow different approaches for computing the final polarity score of a phrase. SenticNet is one of the prominent rule-based tools which have been developed to fulfill different natural language processing tasks such as feature/concept extraction, name entity recognition and polarity detection (Cambria, Poria, Gelbukh, & Nacional, 2017). It utilizes linguistic patterns as well as the deep learning techniques to extract features and concepts from the sentences. To detect polarity of the words, it first utilizes linguistic patterns and where pattern is found, it uses the trained machine learning classifiers to accomplish the job.

2.2. Polarity determination

Approaches to determine the polarity of an opinion word can be grouped into three types, namely *supervised machine learning, semi-supervised* or *unsupervised learning, and dictionary-based* techniques (Schouten & Frasincar, 2016).

Supervised machine learning techniques are used to classify the sentiments based on their polarity. Bhattacharjee, Das, Bhattacharya, Parui, and Roy (2015) proposed a supervised machine learning approach to analyze reviews in the telecom industry. Using a lexicon-based algorithm, the noise in the dataset is first reduced before extracting the features using noun frequencies. A cosine similarity-based algorithm is developed to classify the collected sentiments on a scale of highly negative to highly positive. Jeyapriya and Selvi (2015) proposed a naïve Bayesian classifier to calculate the probability of a positive and negative combination of words within each sentence. Reviews from different websites are collected and stop words are removed before applying stemming and POS tagging. After this step, noun and noun phrases are extracted along with their frequencies before checking them against the minimum support threshold required to be considered as a feature. The naïve Bayesian classifier is then used to predict the positive and negative polarity of the words in a sentence. After comparing all the words in the sentences, the sentiment orientation of each sentence is determined based on the subtraction of its positive and negative probabilities. Yu et al. (2011) utilized the description in the pros/cons reviews to train an SVM opinion classifier. Sentiment words within pros/cons reviews are collected, and their polarities are determined based on the explicit polarity orientation of the type of review. Next, an SVM is trained with the extracted sentiments. This classifier can be later used to determine the polarity of the free text reviews. However, similar to the drawback of supervised techniques in the feature-extraction task, polarity determination also needs a labelled corpora to train the employed machine learning technique, which is not practical.

Semi-supervised or *unsupervised* techniques address the issue of having a training corpus as required in supervised approaches. Lau, Li, and Liao (2014) proposed a semi-supervised fuzzy product ontology mining algorithm which can automatically predict the polarities of feature-level sentiments. The proposed approach first performs some pre-processing steps, such as removing stop words, before using the developed fuzzy product ontology to capture taxonomic and non-taxonomic relations. After this, consumer reviews, product ratings and product descriptions retrieved from websites are used to develop a probabilistic latent topic modeling for context-based sentiment analysis. Next, an analyzer determines the polarities of comments, both context-based and context-free lexicons. Finally, the mean of the sentiment polarity scores for each product's features is computed and presented based on the collected comments. García-Pablos, Cuadros, and Rigau (2015) proposed an unsupervised polarity detection technique. They utilized the Word2Vec tool to determine the polarity score of each word by considering its domain. After pre-processing the dataset, the semantic word vectors are computed. The similarity between the word and its relative known positive and negative seed words are used to determine its polarity value.

Dictionary-based methods are one of the most popular techniques for determining the polarity of reviews. They generally use a pre-built lexicon to discover the sentiment score of individual opinions (Agrawal & Siddiqui, 2009). Liu and Hu (2005) generated a lexicon that included more than 2000 positive and approximately 5000 negative words along with misspelled variations of the words that frequently appear in the reviews. In their lexicon, words are just annotated by their polarity orientation and no strength level has been mentioned for them. However, in *SentiStrength*, as another lexicon, 2310 sentiment words are labeled by the authors in the range -5 (strongly negative) to $+5$ (strongly positive), (Thelwall, 2013). Similarly, Finn Arup Nielsen (2011) collected 2477 unique words as well as sentiment phrases and labeled them on the scale of -5 to $+5$ according to their polarity strength. Although this approach is very popular because of its simplicity, it suffers from an inability to deal with context-dependent opinion words.

To enhance the results of the polarity determination methods, some studies deal with terms or words that modify the sentiment of an opinion. For instance, Benamara, Irit, Cesarano, Federico, and Reforgiato (2007) utilized linguistic analysis to determine the degree of adverbs. Using ten different annotators, approximately 100 adverbs were assigned a score between -1 to $+1$. The two common combinations of adjectives and adverbs are considered to propose different axioms and a score is allocated to each sentiment phrase with regard to the adjective and adverb polarity scores. Another approach that determines the polarity of a word deals with the presence of a negation word in a sentence. Hogenboom, Van Iterson, Heerschop, Frasincar, and Kaymak (2011) computed the sentiment score of each word using WordNet. In the presence of a negating word, the algorithm inverses the sentiment scores using a pre-defined strength factor. In this study, the authors assessed the impact of several approaches in determining the influence scope of negations. Based on their results, a fixed window size of two words following the negation word outperformed the other approaches.

2.3. Sentiment aggregation

As explained earlier, the aggregation process provides a concise summary of customer attitudes towards different features of a product. Mostly, the studies calculate the final result for each feature by considering the average or sum of the different sentiment scores (Lau et al., 2014). However, to enhance the performance of the aggregation algorithms, Suleman and Vechtomova (2015) proposed a two-step hierarchical clustering process. In the first step, they extract candidate nouns from the collected reviews. Using similarity measure, the candidates are grouped in clusters. Next, hypernyms are suggested for each noun candidate by searching the web. The clusters are then selected for merging, according to the similarity of the hypernym representation. Finally, the top K hypernyms are chosen as labels for each cluster. The proposed model is evaluated using datasets on two restaurants and one camera. In another study, Ye, Li, Guo, Qian, and Yuan (2015) proposed an unsupervised method for product feature summarization using word vector representation and k-means clustering. In the first step, unnecessary words from the collected reviews are removed before using Word2vec as a vector representation tool for unlabeled words. Finally, using a similarity measurement, the dataset is divided into different clusters, each of which is a representative of a product feature along with its various sub-features. The performance of the proposed approach is tested with three typical feature mining approaches using reviews collected from a shopping mall in China. Mukherjee and Joshi (2013) computed the sentiment score of a review with respect to the hierarchical relationship between different features of a product. They proposed an unsupervised approach that aggregates the sentiment about various features of a product and finally computes the overall polarity of a review considering the importance of each feature in the product's tree. ConceptNet is used to create the product-specific ontology tree, which shows the hierarchical relationships between different product attributes. The product tree is first constructed with the help of frequent nouns within customer reviews and relationships extracted from ConceptNet. Using a dependency parser, the opinion words describing each feature are then detected from the product reviews. Finally, by considering the height of a feature node in the ontology tree (the higher its importance, the closer its place to the root), a polarity aggregation formula is proposed to calculate the sentiment score at the document level.

ConceptNet is also used in other research to construct a domain-specific ontology for product reviews. To better converge the product features, Agarwal, Mittal, Bansal, and Garg (2015) expanded the ConceptNet results by adding the synonyms of concepts using WordNet. After this, frequent nouns are extracted from the reviews as the product features. Irrelevant features are pruned with the help of an ontology. Next, feature-specific opinions are obtained using the dependency parser. Considering the mean and standard deviation of the sentiment score for each term, ambiguous terms in the customer reviews are identified. The polarities of these ambiguous terms are finally computed based on the probability of their context terms belonging to the positive/negative class. Similar to previous studies, the importance of various features in aggregating the overall polarity score of each document is determined and tested on three different domains.

2.4. Summary of existing works and their limitations

Table 1 presents a summary of some of the discussed techniques in accomplishing the tasks of feature-opinion extraction and polarity determination in the sentiment analysis area. The papers are sorted based on the review format (shown in the second column) that each study has addressed. Columns 3 and 4 of Table 1 represent the techniques that were utilized by these studies in extracting features and determining the polarity of each feature/review, respectively. Finally, in the last column, the papers are categorized into two groups based on their proposed sentiment level. The value of this column is mentioned as '*Orientation*' if the study presents the polarity of a review/feature as either positive or negative; otherwise, the term *'Strength'* is used to show that the paper considers the intensity of the expressed opinions.

Examining Table 1, we can see that a few works have focused on analyzing pros/cons review types. However, analyzing the pros/cons reviews was not even the main objective of some of them and they leveraged the results of this format as the seed words for extracting knowledge from the detailed reviews (Kim & Hovy, 2006; Lau et al., 2014; Yu et al., 2011). While there is a growing trend providing the pros/cons review templates by websites, considering the specific structure of this format, it is necessary to develop a new method for extracting knowledge from this review type. According to the drawbacks mentioned in Section 1, the existing approaches related to the feature-opinion extraction, pruning and polarity determination of the detailed reviews are not suitable for analyzing the pros/cons reviews. To address these issues, we proposed a rule-based framework, namely Pros/Cons Sentiment Analyzer (PCSA), which does not need any training data or seed words to work. In the next section, different phases of PCSA are described in greater detail.

3. Pros/Cons Sentiment Analyzer (PCSA) framework

Fig. 1 shows our proposed PCSA framework, which is an adaptive unsupervised syntactic rule-based method for feature extraction and polarity determination. PCSA has three phases. First is the *pre-processing* phase in which using Stanford NLP tools (Chen & Manning, 2014; Toutanova, Klein, Manning, & Singer, 2003), the raw reviews are prepared for further analysis. Recent studies demonstrate that rule-based methods produce better results in mining product features compared to other approaches (Kang & Zhou, 2016; Liu et al., 2015; Poria et al., 2014). Hence, PCSA uses syntactic rules to extract the pairs of (candidate feature, candidate opinion) from the pros/cons reviews in the second phase; namely *Feature-opinion mining.* An extracted feature/opinion is termed a *candidate feature/opinion* at this stage since there is the possibility of it being a redundant result. To improve the accuracy of the extraction process, a double propagation approach is used to reanalyze those reviews that were not analyzed by the defined syntactic rules. As the final task of the second phase, candidate features and opinions that are actually non-features or non-opinion words are removed by a taxonomy-based pruning process. After accomplishing the pruning task and removing the redundant results, the remaining candidate features and opinions are considered *features* and *opinions,* respectively. *Polarity determination* is the third phase of PCSA in which two lexicons are utilized to ascertain the polarity strength of the extracted pairs. Finally, an overall positive and/or negative sentiment score is presented for each discussed feature. The details of the working of each phase are explained as follows.

3.1. Pre-processing phase

For product designers, analyzing *pros/cons* format reviews as opposed to detailed ones is easier as the positive and negative aspects of a review are pre-classified into their own group. At the same time, it has its own challenges due to the different structures and styles used by users to express their opinions in this type of review. While both of the following examples have been extracted from pros reviews, the writing structure of the comments varies. In the *pros-1* review, the customer expresses his satisfaction regarding three different features of a camera using sentiment (opinion) words such as *bright, good* and *well-designed*. However, although the reviewer of *pros-2* is also happy with the mentioned features, he uses no more descriptive words to show his personal attitude toward them.

pros-1 → *"Bright display, good white balance, well-designed ergonomics"*

pros-2 → *"20 fps Continuous shooting, Dual USC-II SD slots."*

Irrespective of the writing structure and style used, such raw reviews need to be first processed before further analysis can be undertaken at a fine-grained level. To do so, in the first stage of

Table 1
Summary of existing works in aspect-opinion extraction and polarity determination tasks.

Authors	Format of review that was addressed	Techniques used for feature extraction	Techniques used for sentiment classification	Sentiment level
Hu and Liu (2004)	Detailed type product reviews	Frequent nouns + association rule mining	Dictionary-based	Orientation
Manek et al. (2016)	Detailed type movie reviews	Frequent nouns + Gini Index	SVM classifier	Orientation
Chen et al. (2015)	Detailed type product reviews	Frequent nouns	Supervised information gain	Orientation
Samha et al. (2015)	Detailed type product reviews	Conditional random fields	-	-
Alghunaim et al. (2015)	Detailed type service reviews	Supervised vector representation model	SVM classifier	Orientation
Bagheri et al. (2013)	Detailed type product reviews	FLR-based multiword aspects + iterative bootstrapping algorithm	Dictionary-based	Orientation
Huang et al. (2012)	Detailed type service reviews	Automated bootstrapping algorithm	Using average of Yelp rating	Orientation
Carter and Inkpen (2015)	Detailed type product & service reviews	Co-training technique	SVM classifier	Orientation
Poria et al. (2014)	Detailed type product & service reviews	Syntactic rules	-	-
Kang and Zhou (2016)	Detailed type product & movie reviews	Syntactic rules + double propagation algorithm	-	-
Qiu et al. (2011)	Detailed type product reviews	Syntactic rules + double propagation algorithm	Context-based method	Orientation
Zhuang et al. (2006)	Detailed type movie reviews	Syntactic rules	Dictionary-based	Orientation
Liu et al. (2015)	Detailed type product reviews	Syntactic rules	-	-
Rana and Cheah (2017)	Detailed type product reviews	Syntactic rules + SenticNet	-	-
Cambria et al. (2017)	Detailed type product/service reviews	Syntactic rules + deep learning techniques	Linguistic patterns + machine learning classifiers	Strength
Bhattacharjee et al. (2015)	Detailed type telecom reviews	Frequency-based	Cosine similarity-based classifier	Strength
Jeyapriya and Selvi (2015)	Detailed type product reviews	Frequency-based	Naïve Bayesian classifier	Orientation
Agrawal and Siddiqui (2009)	Detailed type movie reviews	-	Dictionary-based + heuristic rules	Orientation
Li, Chen, Liou, and Lin (2014)	Detailed type product reviews	Frequency-based	Dictionary-based	Orientation
García-Pablos et al. (2015)	Detailed type product & service reviews	-	Unsupervised Word2Vec algorithm	Strength
Wang and Wang (2014)	Detailed type product reviews	Semi-supervised self-tagging algorithm + WordNet	Dictionary-based	Strength
Lau et al. (2014)	Detailed type + Pros/cons product reviews	Frequency-based + LDA	Word divergence (WD) measure	Orientation
Yu et al. (2011)	Detailed type + Pros/cons product reviews	Frequency-based + one class SVM classifier	Using pros/cons initial polarity + SVM classifier	Orientation
Ganapathibhotla and Liu (2008)	Detailed type + Pros/cons product reviews	-	Using pros/cons initial polarity + Dictionary	Orientation
Kim And Hovy (2006)	Detailed type + Pros/cons product reviews	Maximum Entropy classifier	-	-
Liu and Seneff (2009)	Detailed type + service reviews	Syntactic rules	Using review ratings	Strength
Liu and Hu (2005)	Pros/cons type product reviews	Pattern discovery	Using pros/cons initial polarity	Orientation
Pros/Cons Sentiment Analyzer (PCSA) proposed in this paper	Pros/cons type product reviews	Syntactic rules + double propagation	Pros/cons initial polarity + two different dictionaries + heuristic rules	Strength

this phase, the objective is to break down each review and identify all different candidate features along with their associated opinion words (if any) that are discussed by the user. Several tools have been developed (Manning et al., 2014) which can be employed to break the raw detailed reviews into sentences. We enhance the performance of existing tools by utilizing punctuation delimiters such as comma, semicolon, question marks and exclamation marks along with full stops or dots to break each pros/cons review into *individual statement(s)*. We consider an individual statement as a collection of words that usually contains a product feature along with a possible opinion word. The statements are further analyzed in the second phase for their discussed candidate feature(s) along with possible associated candidate opinion word(s).

However, before such processing in Phase 2 can be done, the grammatical role of each word along with its relationship and dependencies with the other words in a statement needs to be identified. According to (Qiu et al., 2011), this is a critical step in developing syntactic rules which can be later used for extracting candidate features and opinion words from each statement. This is achieved in stage 2 of this phase where we first tag each word in a statement with its Part of Speech (POS) such as noun, verb, adjective and so forth using the Stanford POS tagger (Toutanova et al., 2003). After that, we model the syntactic dependencies that exist between different words in a statement using the Stanford Dependency Parser (Chen & Manning, 2014).

Fig. 2 shows the workings of Phase 1. It shows that the raw pros/cons reviews are turned into individual statements, which are further processed and tagged with their POS and grammatical roles. The output is used further in Phase 2 to extract the candidate feature/s and opinion word/s from collected statements.

3.2. Feature-opinion mining phase

The broad objective of this phase is to analyze the processed reviews from Phase 1 and extract the features discussed along with their opinion words. This objective is achieved in three stages. In stage 1, syntactic rules are defined and applied to extract the candidate features and opinion words from the collected statements. While the objective of this step is to analyze all the gathered statements using these rules, in reality, this may not happen. Therefore,

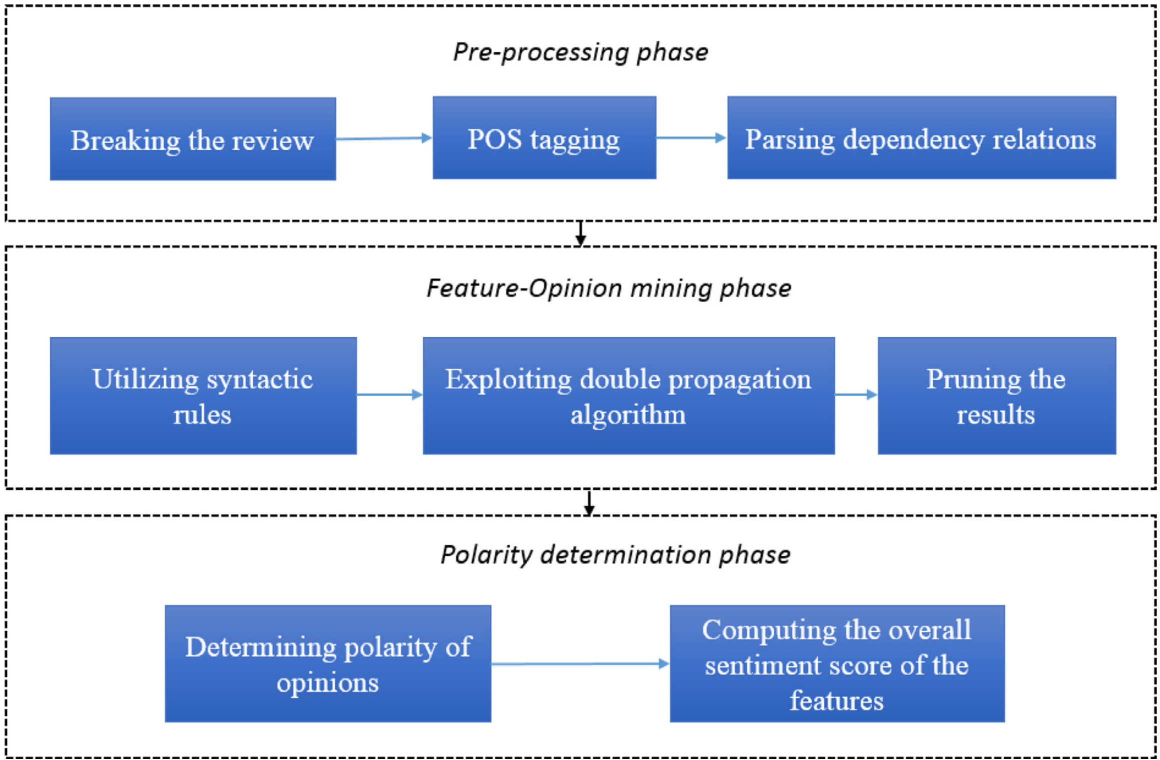


Fig. 1. An overview of the proposed Pros/Cons Sentiment Analyzer (PCSA) framework.

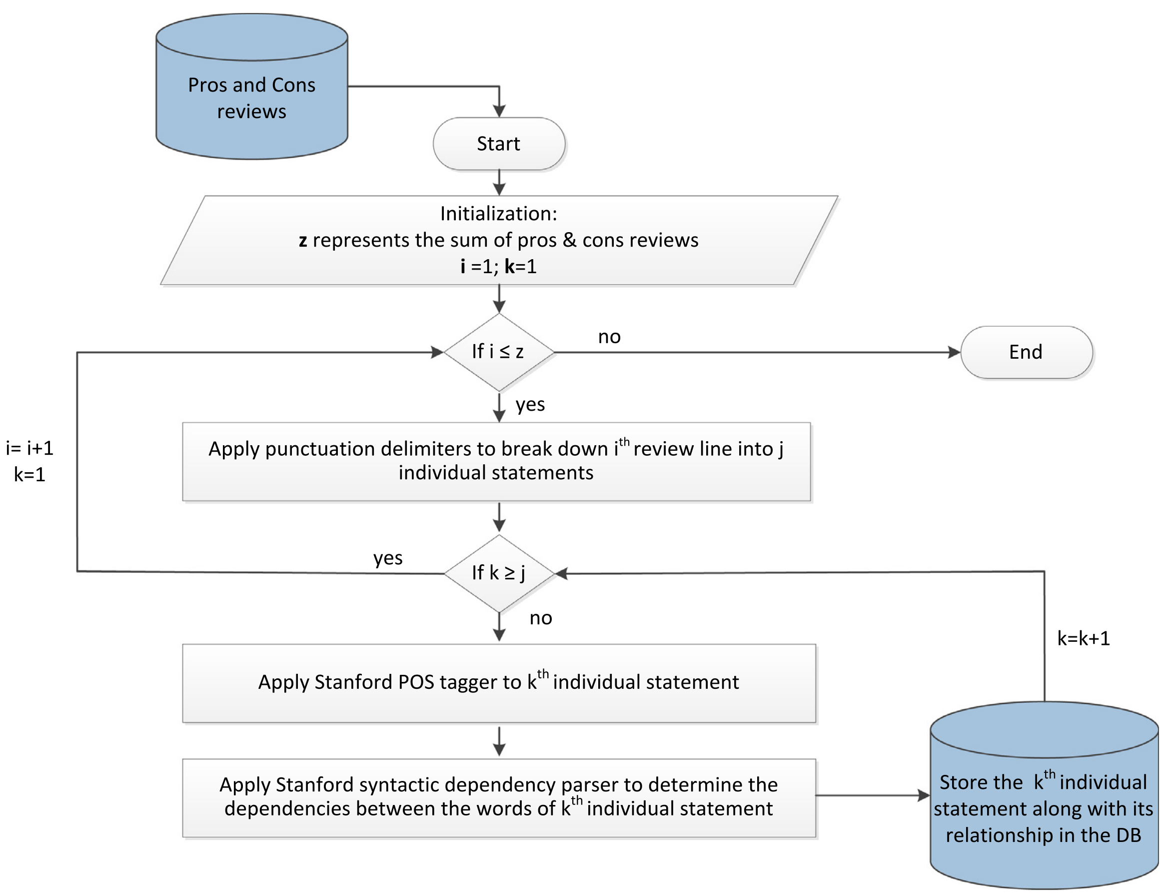


Fig. 2. Workings of PCSA's pre-processing phase.

to enhance the coverage of developed syntactic rules, a double propagation algorithm is generated in stage 2. The double propagation algorithm is an adaptive process that builds on the results of stage 1 with the aim of extracting the candidate features and opinion words from the leftover statements. Once all the statements have been analyzed, stage 3 prunes the candidate features and/or opinion words to remove any incorrect outputs. These are then used in Phase 3 to determine their polarity. The working of each stage of this phase is explained in the next sections.

Stage 1 → Candidate feature and/or opinion word extraction using syntactic rules

In this stage, the identified grammatical roles of words in each statement and the dependencies between them (from Phase 1) are utilized to extract the candidate features and/or opinion words. PCSA aims to extract candidate features from individual statements that are written from three different writing style perspectives; namely *subjective, objective* and *requesting*.

- A statement written from a *subjective* perspective expresses the writer's personal feelings, emotions and aesthetics when discussing the pros and cons.
- A statement written from an *objective* perspective lacks a personal opinion, and the writer uses known facts when discussing the pros and cons.
- A *requesting* perspective statement, as a subset of the *objective* statements, is that in which the writer expresses features that they wished exist in the product. For example, a statement such as*'lack of GPS'*, can be interpreted as the writer's request (suggestion) for adding GPS as one of the features to the next iteration of the product.

The grammatical relationships between the words of these three types of statements differ and, hence, different types of syntactic rules need to be defined when extracting the candidate features and opinion words from them. We discuss this further in the next sub-sections.

Step 1: Defining syntactic rules to extract candidate features and opinion words from pros statements

The aim of this set of rules is to analyze the pros statements written from a ***S***ubjective and ***O***bjective perspective to extract candidate ***F***eatures and ***O***pinions. From here on, in this paper, syntactic rules that extract candidate features and opinion words from statements written with a subjective perspective are abbreviated to *S-F&O* whereas *O-F* refers to rules that mine candidate features from the statements written from an objective perspective.

Depending on their style of writing, statements written from a subjective perspective are further sub-categorized as *direct* and *indirect*. In a direct subjective statement, the user describes the feature using an opinion word without any intermediate words such as '*The display is not bad*' or '*excellent battery life.*' In other words, a direct subjective review refers to cases where a feature is directly modified by an opinion word (Kang & Zhou, 2016). On the other hand, in an indirect subjective statement, an intermediate word is used to link the opinion word to the feature. For example, in '*Easy to handle body*' the feature *body* is linked to the opinion word *easy* through an intermediate word *handle*. In other words, an indirect subject review refers to cases where a product feature is indirectly modified by an opinion word through an intermediate word. In order to extract candidate features and opinion words from the collected statements, apart from the dependency templates proposed in Kang and Zhou (2016), Qiu et al. (2011), we define more syntactic rules based on our observation of pros/cons reviews. The rules are abbreviated as *S-F&O_D* and *S-F&O_I* as direct and indirect subjective statements, respectively. Similarly, an *O-F* set of syntactic rules deals with reviews that contain objective information or facts. An example of the *O-F* rule can be the part-whole relationship that is mentioned in the reviews by the customers such 'It has Wi-Fi.' In fact, this relationship reveals the belongingness of an object to another object without stating an opinion (Kang & Zhou, 2016). We use the cue verbs like 'have', 'consist' and 'contain' to mine the features in these objective statements. Most of the existing work does not consider reviews written from an objective perspective during sentiment analysis as they do not convey any personal attitudes (Liu, 2012). We believe that when a feature is mentioned by a user, in either a pros/cons review, it reveals their attitude towards that specific feature even if no opinion word/s modifies. Hence, in PCSA the objective statements are also analyzed for their discussed features.

Table 2 presents the defined syntactic rules for extracting features and opinion words from pros individual statements. The first column of Table 2 shows the rule ID. The category of the rule (whether it is dealing with a direct or indirect subjective or an objective individual statement) is shown in the second column. The defined syntactic rules are presented in column 3. Following the template adopted by Qiu et al. (2011), *A* in each rule stands for the candidate feature, *B* denotes the candidate opinion word, and *VB* represents the verb. Specific types of dependancies between any two words such as *amod, nsubj, dobj* are represented by *Dep* relationships. Possible sets of POS tags for potential features and opinion words are shown in the fourth and fifth columns, respectively. In the last column of Table 2, an example is provided for each discussed template.

Step 2: Defining syntactic rules to extract candidate features and opinion words from cons individual statements

As mentioned in Section 1, there are existing approaches that utilize one set of syntactic rules to analyze both pros and cons reviews. As a result of a detailed examination, we observed that the grammatical structure of negative statements is different from that of the positive ones. Hence, using a common set of grammatical rules for both pros and cons reviews can result in low accuracy of the extraction process. To address this issue, apart from the pros syntactic rules, we define some specialized grammatical rules for analyzing cons statements. Table 3 presents the developed rules for the cons reviews.

Step 3: Defining syntactic rules to extract requesting features from cons individual statements

As previously discussed, a requesting statement is one in which customers express features that they wished were present in the product. Such opinion types are considered a subset of objective statements, which can usually be found in the cons reviews. As mentioned in the literature (Jhamtani et al., 2015; Ramanand, Bhavsar, & Pedanekar, 2010), apart from explicit phrases such as hope, wish, could, want and so forth that are analyzed in Goldberg et al. (2009), customers' wishes are also expressed by their complaints or them discussing the lack of a specific capability in a product. In PCSA, we consider extracting such candidate features important as the product designers can incorporate them into the new product development cycle. However, in this study we focus on extracting customers' requests for improvement from their complaints written in an objective perspective and not on wish phrases such as 'I wish …, It could have been …' In Table 4, we present the common syntactical rules that the customers use while describing their requests.

Fig. 3 shows the workings of stage 1 of Phase 2. As shown in the figure, at the end of this stage, the individual statements from Phase 1 are processed, and the extracted candidate features and opinion words are stored in the *Extracted PROS database, Extracted CONS database* or *Requesting Feature Database*. Those leftover individual statements, which were not linked and matched with the defined rules, are stored in the *Unprocessed PROS/CONS database* for further analysis using the double propagation approach. This is explained in the next sub-section.

Table 2
The syntactic rules for analyzing pros individual statements and extracting candidate features and opinion words.

Rule ID	Category	Template	Feature output	Opinion output	Example
PROS 1	*S-F&O*	**VB -> Dep$_1$ <- B <- Dep$_2$<- A** VB ∈ {has, have, contain(s), include(s), consist(s)} Dep$_1$ ∈ *{dobj}* Dep$_2$ ∈ *{amod, advmod}*	*{NN, NNS, compound nouns}*	*{JJ, RB-JJ, JJR, JJS, JJ-JJ}*	It includes fast card slots
PROS 2	*S-F&O_D*	**A -> Dep-> VB – B** *VB* ∈ {is, are, work(s), perform(s)} Dep ∈ *{nsubj}*	*{NN, NNS, compound nouns}*	*{JJ, RB-JJ}*	The display is so bright
PROS 3	*S-F&O_D*	**A -> Dep$_1$-> VB –not- B** VB ∈ {isn't, aren't, is not, are not, work(s), perform(s)} Dep$_1$ ∈ {nsubj}	*{NN, NNS, compound nouns, JJ-NN}*	*Not* + *{JJ, RB-JJ}*	The display is not bad
PROS 4	*S-F&O_D*	**B -> Dep-> A** Dep ∈ *{amod, advmod, nsubj}*	*{NN, NNS, compound nouns, VB}*	*{JJ, RB-JJ, JJR, JJS, JJ-JJ}*	Good display Rugged body
PROS 5	*S-F&O_I*	**B -> Dep$_1$-> VB <- Dep$_2$ <- A** Dep$_1$ ∈ *{xcomp}* Dep$_2$ ∈ *{Ø, dobj}*	*{Ø, NN, NNS, compound nouns}*	*{JJ, RB-JJ, JJR, JJS, JJ-JJ}*	easy to handle body
PROS 6	*O-F*	**VB -> Dep$_1$ <-A** VB ∈ {has, have, contain(s), include(s), consist(s)} Dep$_1$ ∈ *{dobj}*	*{NN, NNS, compound nouns}*	-	It has Wi-Fi

Note: The underlined words in the last column are extracted either as a feature or opinion word by the corresponding rule.

Table 3
The syntactic rules to analyze cons individual statements and extract candidate features and opinion words.

Rule ID	Category	Template	Feature output	Opinion output	Example
CONS 1	*S-F&O*	**VB -> Dep$_1$ <- B <- Dep$_2$<- A** VB ∈ {have, contain, include, consist} Dep$_1$ ∈ *{dobj}* Dep$_2$ ∈ *{amod, advmod}*	*{NN, NNS, compound nouns}*	*{JJ, RB-JJ, JJR, JJS, JJ-JJ}*	It doesn't have a good display
CONS 2	*S-F&O_D*	**A -> Dep$_1$-> VB –not- B** VB ∈ {isn't, aren't, is not, are not, does not work (perform), doesn't work (perform)} Dep$_1$ ∈ {nsubj}	*{NN, NNS, compound nouns, JJ-NN}*	*Not* + *{JJ, RB-JJ}*	The display is not bright
CONS 3	*S-F&O_D*	**A -> Dep$_1$-> VB - B** *VB* ∈ *{is, are}* Dep$_1$ ∈ *{nsubj}*	*{NN, NNS, compound nouns, JJ-NN}*	*{JJ, RB-JJ}*	The button is inconvenient
CONS 4	*S-F&O_I*	**B -> Dep$_1$-> VB <- Dep$_2$** <- A Dep$_1$ ∈ *{xcomp}* Dep$_2$ ∈ *{Ø, dobj}*	*{Ø, NN, NNS, compound nouns}*	*{JJ, RB-JJ, JJR, JJS, JJ-JJ}*	Difficult to zoom lens
CONS 5	*S-F&O_I*	**A -> Dep-> Prone to** *Dep* ∈ *{amod, nsubj}*	*{NN, NNS, compound nouns, JJ-compound nouns}*	-	LCD prone to glare
CONS 6	*S-F&O_I*	**B-Dep$_1$-> A -> Dep$_2$-> IN <-Dep$_3$ <-A'** Dep$_1$ ∈ {*amod, advmod*} Dep$_2$ ∈ {subject} Dep$_3$ ∈ {object} IN ∈ {in}	*{NN, NNS, compound nouns}*	*{Ø, JJ, RB-JJ, JJR, JJS, JJ-JJ}*	Bad focus hunting in video AF
CONS 7	*S-F&O_I*	**B-Dep$_1$-> A -> IN <-Dep$_2$ <-A'** Dep$_1$ ∈ {*xcomp*} Dep$_2$ ∈ {object} IN ∈ {in}	*{VB, NN, NNS, compound nouns}*	*{JJ, RB-JJ, JJR, JJS, JJ-JJ}*	Slower to focus in AF mode
CONS 8	*S-F&O_D*	**B -> Dep-> A** Dep ∈ *{amod, advmod, nsubj}*	*{NN, NNS, compound nouns, VB}*	*{JJ, RB-JJ, JJR, JJS, JJ-JJ}*	Heavy body

Note: The underlined words in the last column are extracted either as a feature or opinion word by the corresponding rule.

Table 4
The syntactic rules to analyse cons individual statements and extract the requesting features.

Rule ID	Category	Template	Feature output	Opinion output	Example
CONS 9	*Requesting*	**{lack(s), no, absence, miss(s)} + A**	*{NN, NNS, NP, JJ-NN}*	-	No GPS
CONS 10	*Requesting*	**Do (does) + not + VB -> Dep <- A** VB ∈ {have, contain, include, consist, come with} Dep ∈ {dobj}	*{NN, NNS, compound nouns}*	-	It doesn't include CF card

Note: The underlined words in the last column are extracted as a requesting feature by the corresponding rule.

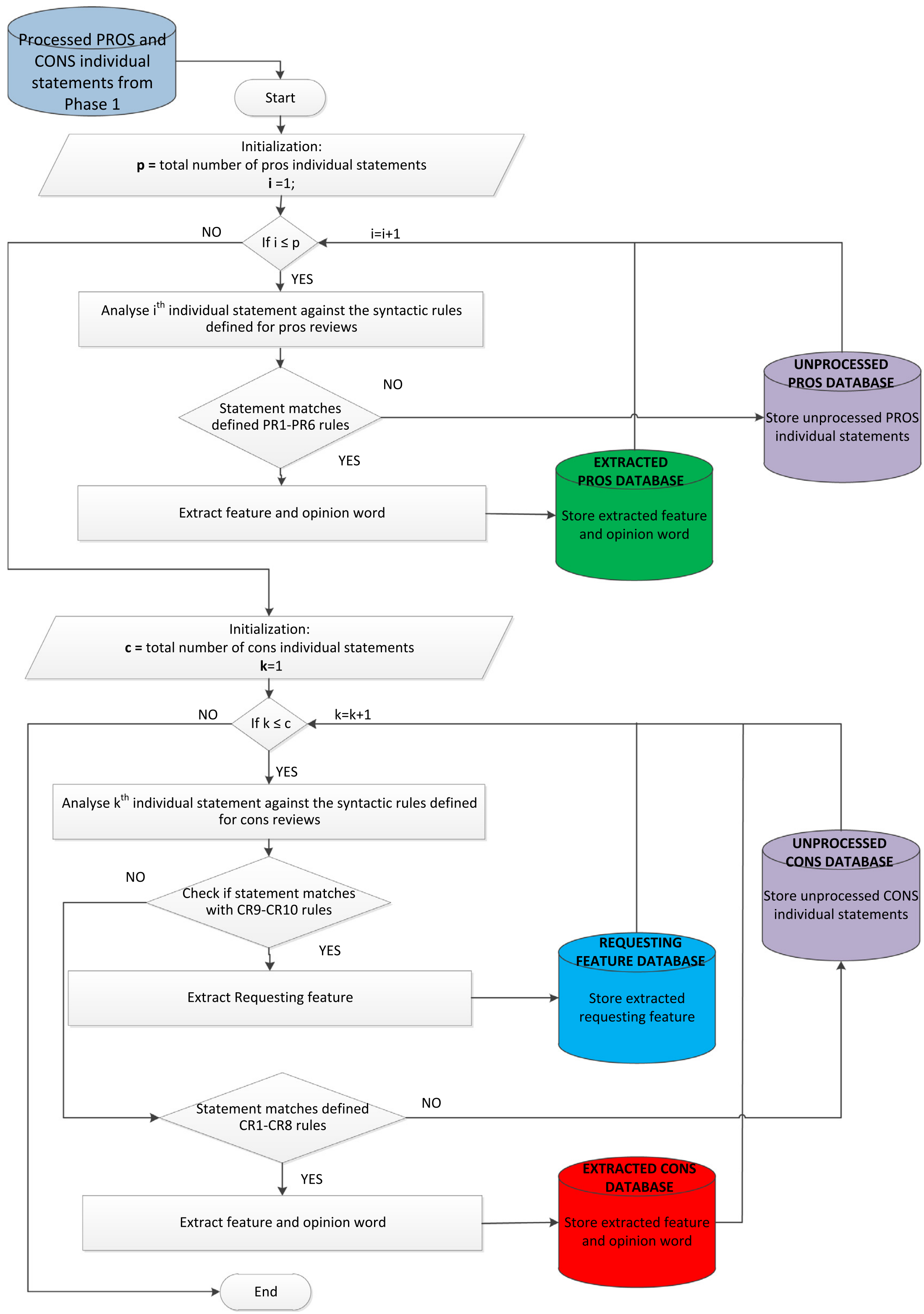


Fig. 3. The workings of PCSA's candidate feature-opinion extraction using syntactic rules.

Stage 2 → Exploiting the double propagation (DP) algorithm

In this stage, the unprocessed individual statements from the *Unprocessed PROS and CONS databases* of Fig. 3 are re-analyzed using a double propagation algorithm. Double propagation is a bootstrapping technique which utilizes a set of seed words to mine opinion words and features using the dependencies between them (Liu, 2015). In other words, the unprocessed statements are first searched for the previously identified (extracted) features and/or opinion words. Based on a set of pre-defined syntactic relationships that might exist between those identified features and/or opinion words and the rest of the words in that statement, new candidate features or opinions will be extracted. Such a propaga-

Table 5
The double propagation dependency rules.

Rule ID	Template	Feature output	Opinion output	Example
DPR1	**A -> Dep-> A'** Dep ϵ *{Conj}*	*{NN, NNS, compound nouns, VB}*	-	Display and view finder
DPR2	**B -> Dep-> B'** Dep ϵ *{Conj}*	-	*{JJ, RB-JJ, JJR, JJS, JJ-JJ}*	amazing and light
DPR3	**B -> Dep-> A** Dep ϵ *{amod, advmod, nsubj}*	-	*{JJ, RB-JJ, JJR, JJS, JJ-JJ}*	Very slow autofocus
DPR4	**B -> Dep-> A** Dep ϵ *{amod, advmod, nsubj}*	*{NN, NNS, compound nouns, VB}*	-	Very slow autofocus
DPR5	**A -> Dep-> B** Dep ϵ *{subj, obj}*	*{NN, NNS, compound nouns, VB}*	-	AF system slow
DPR6	**A -> Dep-> B** Dep ϵ *{subj, obj}*	-	*{JJ, RB-JJ, JJR, JJS, JJ-JJ}*	AF system slow

Note: In this table, the words which are marked by a solid line are the identified ones, and the words with a dashed underline are extracted using DP.

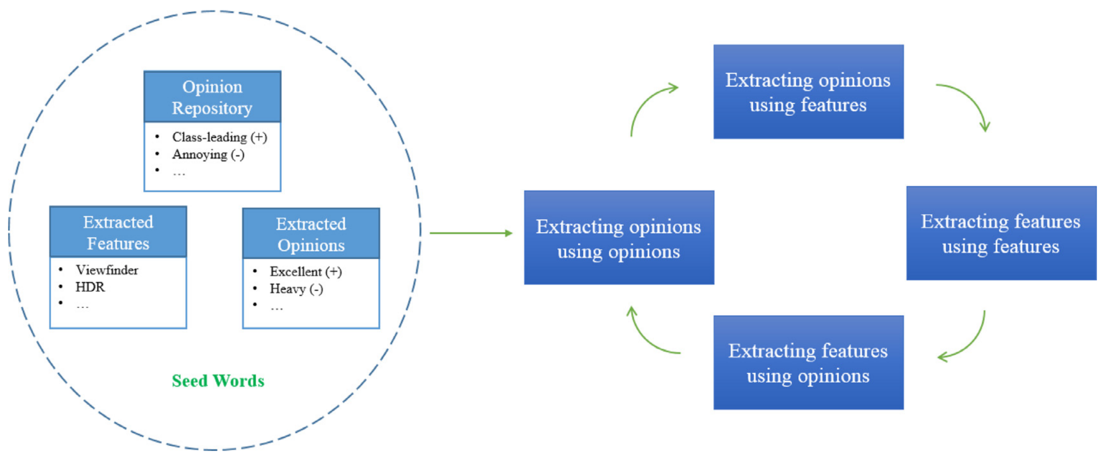


Fig. 4. The double propagation approach applying to unprocessed statements.

tion process is carried out until no more features/opinion words can be detected (Qiu et al., 2011). As shown in Fig. 4, PCSA adopts a systematic approach for carrying out double propagation using seed words. The seed words consist of the identified candidate features and opinion words that are used to extract possible candidate opinions and features from unprocessed statements. While extracting opinions, a lexicon of positive/negative opinion words (Liu & Hu, 2005) is also used to expand the coverage. The different steps in the double propagation process are explained further in the next sub-sections.

Step 1: Utilizing identified candidate features to reanalyze unprocessed individual statements and extract candidate feature and opinion words from them

The focus in this step is to utilize each identified candidate feature and search for it in each unprocessed individual statement. In the case of a match, syntactic rules DPR 1, 3 and 6 defined in Table 5 are applied. DPR 1 syntactic rule analyzes if the matched candidate feature has a conjunction POS after it, as it represents another candidate feature that should be extracted. For the matched and extracted candidate features, DPR 3 and DPR 6 syntactic rules check if they have the mentioned dependencies or grammatical patterns with an opinion word. This process is repeated for each identified candidate feature using the unprocessed individual statement. The results are stored in the *Extracted PROS and CONS database (V1)* as shown in Fig. 5.

Step 2: Utilizing identified opinions to reanalyse unprocessed individual statements and extract candidate opinion words and features

The focus in this step shifts to opinion words. Each unprocessed individual statement is searched for the candidate opinion words using previously extracted opinion words along with those that are defined in the repository of positive/negative opinion words (Liu & Hu, 2005). If the search results in a match, then syntactic rule DPR2, as shown in Table 5, is applied to see if there is a conjunction POS after the matched opinion word, as it represents another candidate opinion word that should be extracted. For the matched and extracted opinion words, DPR 4 and DPR 5 syntactic rules check if it has the mentioned dependencies or grammatical patterns with a candidate feature. The results are stored in the *Extracted PROS and CONS database version (V2)* as shown in Fig. 5.

The working of the double propagation task is shown in Fig. 5. After the processing of DP steps 1 and 2, we may have pairs such as (candidate feature, candidate opinion word), (candidate feature, θ) or (θ, candidate opinion word). An instance of (candidate feature, θ) is when a user does not utilize any opinion words while writing a pros/cons review (e.g. 'Display, battery'). Similarly, there may be cases where a user describes the overall performance of a product using an opinion word (e.g. 'fast'). Hence, the output of PCSA will look like (θ, candidate opinion word).

Stage 3 → Pruning the candidate feature results

In this stage, the results of the first two stages are pruned.

(a) Remove candidate opinion words which represent a part of a feature: There may exist cases where the extracted candidate opinion words do not represent any personal attitude of the customer. Instead, it describes an attribute of a feature that has been mistakenly extracted by the dependency parser tools as an opinion word. However, occurrence of such errors is inevitable. Moreover, the information that we obtain from these candidate opinion words can help us extract customers' opinions at a more fine-grained level. In PCSA we detect such candidate opinion words in our collected pairs and add those to their corresponding candidate feature. For instance, in the term *raw photo,* the word *raw* and *photo* are mined as a candidate opinion word and a candidate feature, respectively. However, 'raw' is a type of photo and does

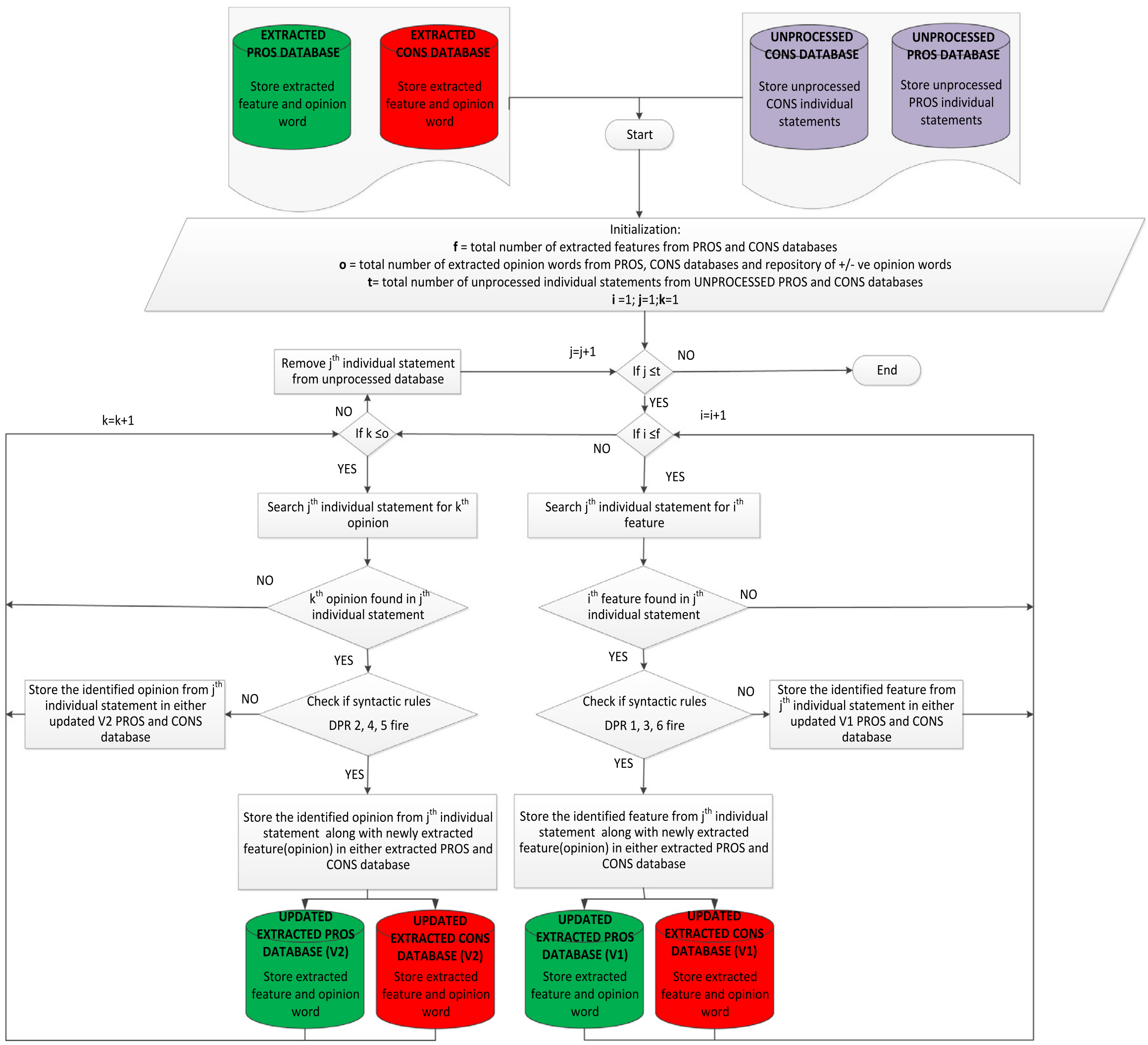


Fig. 5. The workings of PCSA's double propagation approach.

not represent any emotion about this feature. While such opinion words mostly carry no polarity, they can be utilized in the aggregation step to obtain more fine-grained results. In other words, rather than assigning the polarity of this statement to the feature 'photo', after applying the pruning step, we assign to the 'raw photo'.

(b) Remove candidate features that are not features of the product: This issue arises if the extracted candidate features are not real features. They need to be screened out of the final results. For example, in a pros statement, '*good suggestion*' matches with rule PROS 3 of Table 2 and 'suggestion' is extracted as the candidate feature. However, this word is not a feature of a product and needs to be removed before the next step. The existing literature attempts to address such issues by using global pruning (Qiu et al., 2011). The logic behind this approach is to decide about the validity of a candidate feature based on the other words in its vicinity. However, this may not work in pros/cons reviews due to the brevity of the writing style.

(c) Deal with acronyms mentioned by users in their reviews: Since the acronyms in the product reviews are mostly context-based, existing generic dictionaries such as WordNet may not contain their expanded version. Hence, metrics such as semantic similarity proposed by Carenini, Ng, and Zwart (2005) are not helpful in identifying the validity of these terms.

These issues are common during sentiment knowledge extraction and need to be addressed to remove any ambiguity before the next phase of determining the polarity of each feature. To address the aforementioned issues, in PCSA, we propose a pruning method that uses the product-tree. The pruning approach includes the following steps:

Step 1: Generating a product-tree from the product specification document

A product tree (taxonomy) is a schematic representation of various features of a product in the form of parent-child and has-quality relations (Mirtalaie et al., 2017b). Fig. 6 shows a part of a product tree for a camera (Nikon D800) generated using its product specification document. A product specification document is mostly made by the manufacturers or experts and describes various features of a product. This document is freely available on the Internet for a wide range of different products. Since it contains almost all components of a product, it is beneficial to use for a comprehensive pruning process. Unlike the traditional approach of using dictionaries in generating a product tree (Lau et al., 2014; Mukherjee & Joshi, 2013), we utilize a product specification document in this step, which will result in a complete product tree. The product specification document is mostly in the form of a table. The first column presents the high-level features of a product (e.g. flash or image for camera). However, the attributes or sub-features of each of these high-level features are described in the rest of the columns. Each cell in the product specification table is indexed in a way that it not only shows its current level (column), but also maintains the link with its associated high-level feature, if any exist. The procedure of generating the tree starts by placing

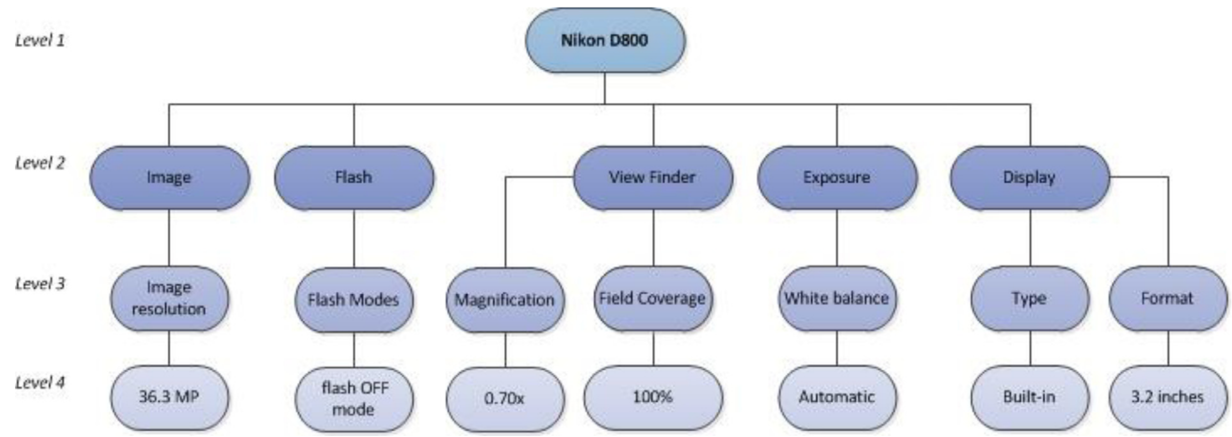


Fig. 6. A part of a product tree for a camera.

the whole product as the tree-root. Next, the high-level features are formed as the first level of the product-tree. Finally, the rest of the nodes are arranged and linked to the first level based on the pre-defined indices.

Step 2: Dealing with acronyms and pre-processing the product-tree

The first objective of this step is to deal with the acronyms mentioned by the users' and link them to the product tree. The next goal is to pre-process the product tree before utilizing it in the pruning of the candidate features. The approach adopted to achieve each objective is explained below.

Acronyms mentioned in a review are only considered if they have been repeated more than two times in the review. In such cases, we ask the expert to manually assign/tag the acronyms to the right nodes in the product tree (e.g. assigning 'mp' to 'megapixel'). It should be noted that the repetition threshold of two is not fixed and it can be modified according to the analyst's requirements. Moreover, the users may not mention the exact numeric value of a feature in their comments. For instance, while referring to the resolution of a camera, a user might mention '36 MP' instead of '36.3 MP'. To avoid a mismatch, we pre-process the product tree and update the tree nodes with its absolute numerical value where it contains decimal numbers. Another issue arises once the user refers to one specific feature with different terminologies (Mirtalaie, Hussain, & Chang, 2016; Mirtalaie, Hussain, Chang, & Hussain, 2017a). For example, words like 'image', 'picture' and 'photo' have similar meanings and refer to a common word in the product tree. To address this issue, the WordNet's synsets are retrieved, and different groups are defined to link the words with common meaning (Miller, 1995). We also gather those words that are mostly related to the functionality or design of the product such as *performance, ergonomics, construction,* and the like. This group of words is called *functional terms,* and we assign them to the root node of the tree if they are detected within customers' reviews. As the last pre-processing step, the tree nodes are then labeled by the groups' elements if they have any common words with the defined groups. Therefore, the algorithm would be able to identify the correct features even if they are mentioned in different terminologies.

Once the pre-processing of the product tree is complete, PCSA prunes the extracted candidate opinion words/features. This is explained in the next steps.

Step 3: Pruning the opinion words

Consider the terms in the tree as t_i and the extracted opinion as o_j, Eq. (1) is used to assign either a value of 1 or 0 to a pair of (t_i, pj).

$$o\_match(t_i, o_j) = \begin{cases} 1 & if\ t_i\ matches\ o_j \\ 0 & otherwise \end{cases} \quad (1)$$

Since the terms in the tree are mostly related to the product's various features, if a candidate opinion obtains a value of 1 for *o_match*, it is possible that this opinion word presents part of a feature. Hence, we add these opinion words to their corresponding features after an expert checks their validity.

Step 4: Pruning the extracted candidate features

Similarly, considering the terms in the tree as t_i and the extracted candidate features as p_j, Eq. (2) is used to assign either a value of 1 or 0 to a pair of (t_i, pj).

$$f\_match(t_i, p_j) = \begin{cases} 1 & if\ t_i\ matches\ p_j \\ 0 & otherwise \end{cases} \quad (2)$$

The candidate features with a *f_match* equal to 1 are considered the product's features. For candidate features with *f_match* of 0, we first apply stemming to convert it to its stem. As a result of the stemming process, the plural form of a noun is transformed to its singular one (Jeyapriya & Selvi, 2015). Hence, if the presence of a suffix stopped the algorithm from finding a match between the candidate feature and the nodes in the tree, the case can be addressed after applying the stemming process. Step 4 is repeated after stemming to ascertain the *f_match* value. For those terms that still have the *f_match* of 0, PCSA uses a frequency-based technique to confirm if the words are redundant. The frequency of a term in the obtained results is known as one of the metrics in the existing studies for pruning incorrect results (Kang & Zhou, 2016). In other words, if a term has been repeated more than a pre-defined threshold, it is likely to be a product feature. Utilizing this approach, for the features with *f_match* of 0 that has been repeated more than β times in our dataset, we ask the expert to confirm their validation. In the case where the expert approves that the term represents a product feature, we add it to the feature database.

The working of the pruning step is shown in Fig. 7. Once the pruning step is done, the polarity of each feature is determined in the next phase.

Phase 3 - Polarity determination of each feature

In this phase, the polarity of each feature is ascertained with respect to its associated opinion word, if any exists. In PCSA, we utilize two different lexicons to complete this task. First, the polarity of the extracted opinion words is checked against SentiStrength, a general lexicon-based software to detect sentiment strength in short texts (Thelwall, 2013). In scenarios where SentiStrength returns a neutral polarity for the opinion word, we utilize the Bing Lexicon (Liu & Hu, 2005). Unlike SentiStrength, Bing Lexicon does not assign any polarity strength to the opinion words and only classifies them as either positive or negative. We utilize

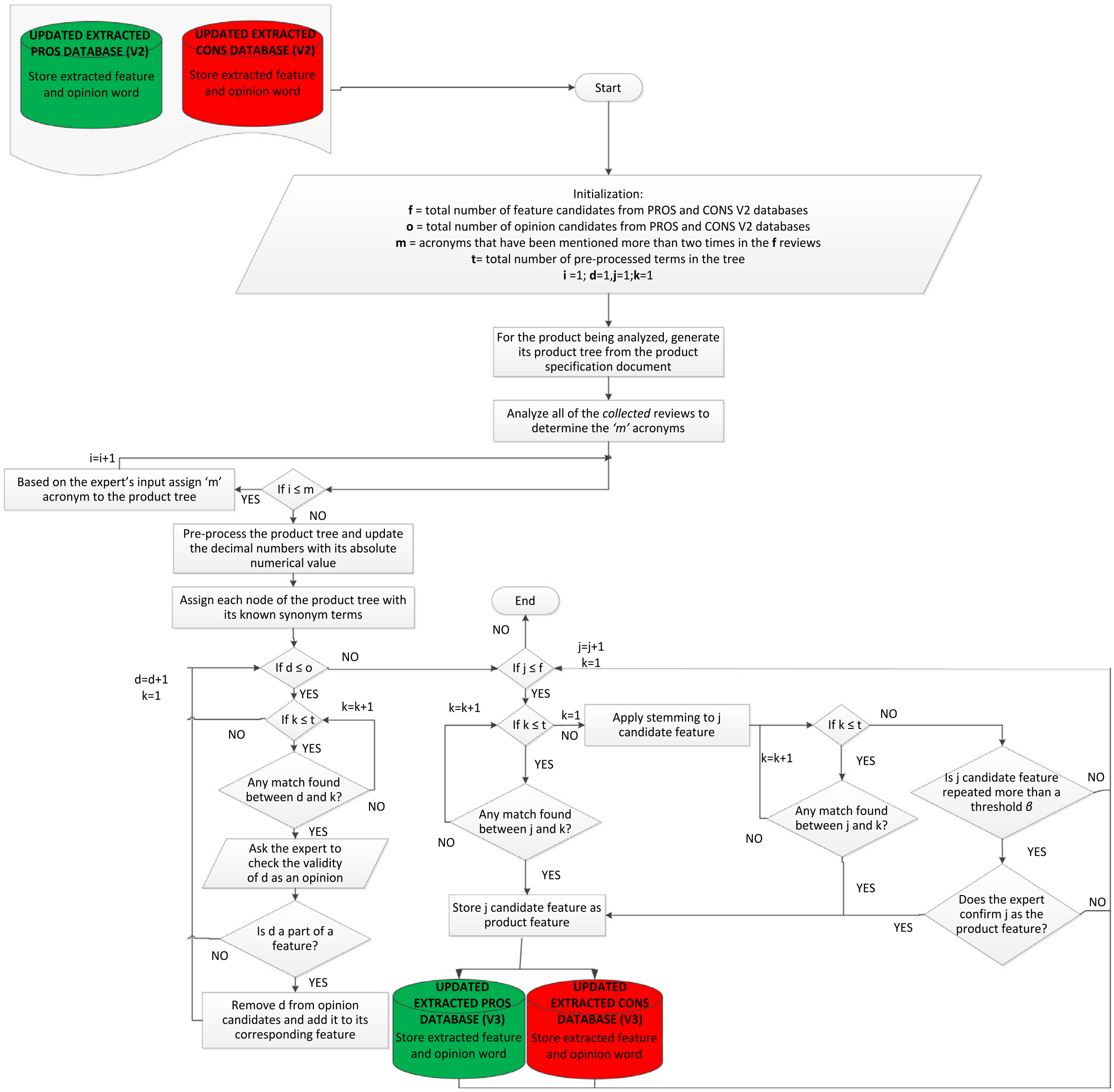


Fig. 7. The working of PCSA's pruning step.

this lexicon as a complementary one since it deals with the misspelled words as well thereby enhancing SentiStrength's results. This phase's objective is achieved in two stages. In the first stage, the polarity of each extracted (feature, opinion) pair is determined as a value in the range of −1 to +1 by using a heuristic method. In the second stage, the final sentiment score of each extracted feature is computed.

As mentioned in stage 2.2, the outputs of our feature-opinion word extraction process can be in the form of a full pair such as (good, viewfinder), or a feature-only pair like (display, Ø), or an opinion-only one such as (Ø, fast). The process of determining the polarity of each output type is explained in the following.

Stage 1: Determining the polarity value of extracted pairs

Step 1: Determining the polarity of features mentioned without an opinion word

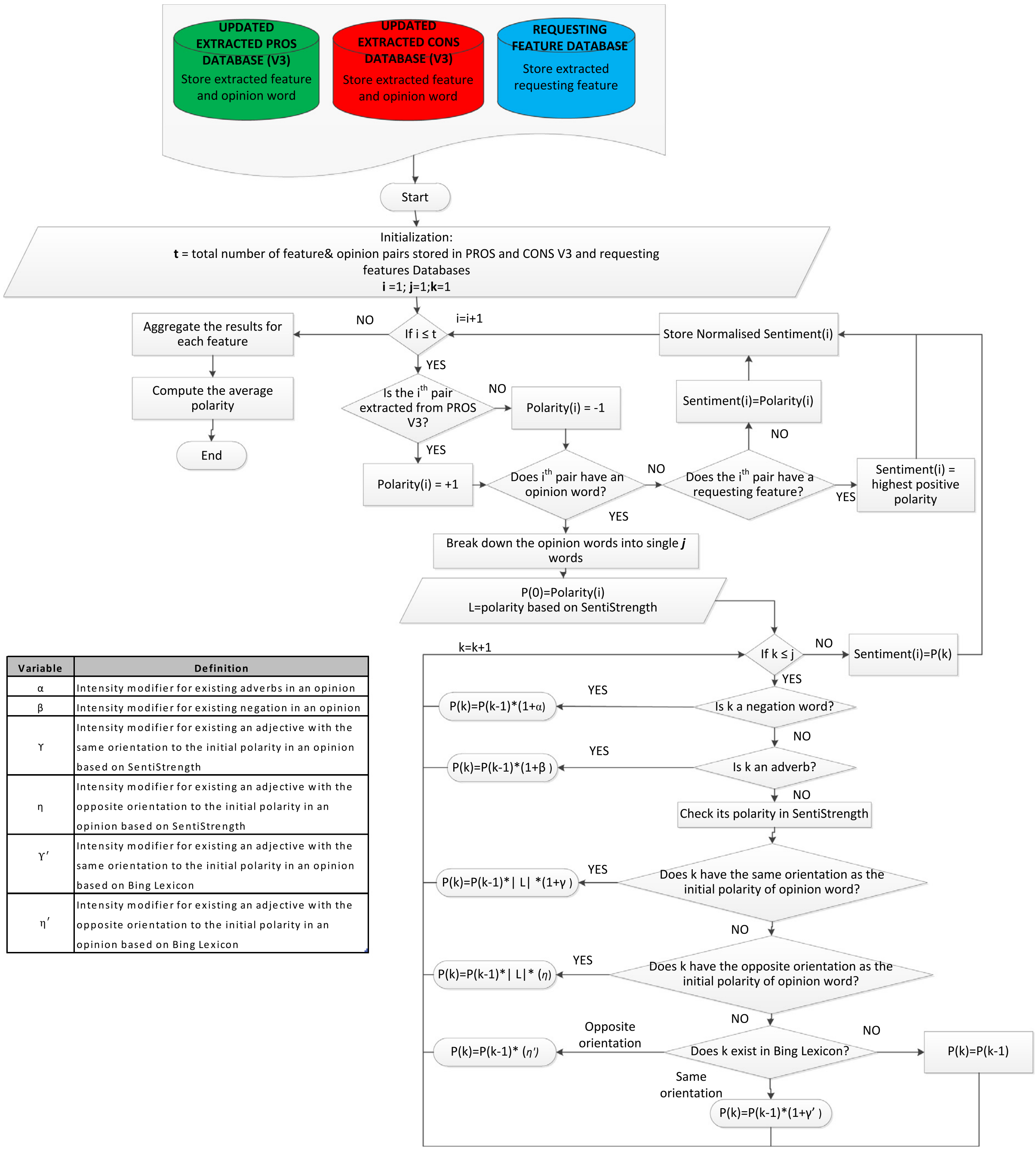
This refers to reviews where the user only points out a product feature as a pros/cons without expressing any complementary personal feelings. In the aforementioned example, let us consider that the user mentions *'Display'* as a pros but they have not added any further details about their opinion. In such a case, we assign the least positive score to the feature in accordance with the orientation of the customer's comment. In other words, for the feature-only pairs (except the requesting features) mentioned in the pros or cons review, the algorithm assigns the least positive or negative polarity score, respectively.

Step 2: Determining the polarity of requesting features

Another type of feature-only pair is the *wishing list* or *requesting features*. Unlike the features mentioned in step 1, requesting features are given the highest positive polarity score to reflect the customer's desire for them.

Step 3: Determining the polarity of features associated with an opinion word

Since we are dealing with pros/cons type reviews, the orientation of the extracted pairs are known beforehand. To incorporate this information in PCSA, we develop a heuristic method to adjust the polarity of a feature in the presence of adverbs or sentiment words. We consider that the following combinations of (adjective, feature) and (adverb+ adjective, feature) occur in the pairing of an opinion word/phrase with a feature. However, other combinations such as (adverb + adverb + adjective, feature) can be adapted by making subtle changes to the working of PCSA. An overview of



Variable	Definition
α	Intensity modifier for existing adverbs in an opinion
β	Intensity modifier for existing negation in an opinion
ϒ	Intensity modifier for existing an adjective with the same orientation to the initial polarity in an opinion based on SentiStrength
η	Intensity modifier for existing an adjective with the opposite orientation to the initial polarity in an opinion based on SentiStrength
ϒ′	Intensity modifier for existing an adjective with the same orientation to the initial polarity in an opinion based on Bing Lexicon
η′	Intensity modifier for existing an adjective with the opposite orientation to the initial polarity in an opinion based on Bing Lexicon

Fig. 8. The workings of PCSA's polarity determination step.

the proposed heuristic method is presented in Fig. 8. The proposed algorithm first breaks down each opinion phrase into single words ($O_{(i)}$), wherever it includes more than one word. Based on the initial orientation of the extracted pair (discussed as a pros or cons), the algorithm assigns it an initial polarity score of either +1 or −1, respectively. Next, each word in the opinion phrase is analyzed and its initial polarity score is modified according to the predefined *intensity factors*. The main objective of exploiting these intensifiers is to reflect the polarity strength of a pair while an adjective, adverb or a negation has been used in its opinion phrase. These intensity factors are defined by the user in the range of 0 to 1. For those opinion words that have been categorized as *neutral* by Sentistrength, PCSA checks them against the Bing Lexicon to ensure that they carry no polarity. The following scenarios are considered when analyzing the opinion word/phrase of the extracted pairs:

Occurrence of a negation word. In linguistics, negation indicates the process of inverting the initial sentiment of a word. Some examples of negation words in the English language are 'not,' 'nowhere,' '*n't' (couldn't), 'by no means' (Hogenboom et al., 2011). In PCSA, we consider only 'not' and '*n't' as the negation words due to their popularity for being used in pros/cons reviews. In the presence of a negation word, depending on the value of α, the algorithm adjusts the initial polarity given to a feature. According to (Hogenboom et al., 2011), a negated sentiment does not carry equal polarity strength as its non-negated counterpart. In other words, 'not bad' does not convey the same positive feeling as that of 'good'. Therefore, we choose a lower value for α compared to that of the *adjective intensifier* (ϒ).

Occurrence of adverbs. In PCSA, we focus on the usage of *adverbs of degree* in an opinion. This type of adverb represents the intensity of emotion (Lobeck, 2000). According to (Benamara et al., 2007), adverbs of degree are of the five types, namely affirmation: *such as exactly, totally, certainly…;* doubt: *seemingly, apparently…;* intensifiers: *extremely, immensely…;* weakeners: *slightly, weakly…;* and negation: *hardly, barely… etc.* Due to the popularity of using affirmation adverbs and intensifiers in product reviews, we only collected adverbs representing these two groups. In the presence of an adverb, PCSA modifies the polarity of a feature with a weight of β. Like the approach in Benamara et al. (2007), we consider a smaller value for β as compared to that of the *adjective intensifier* (Υ).

Occurrence of words that have the same sentiment orientation as that of the review's initial polarity. To determine the polarity score of $O_{(i)}$ when it is neither a negation word nor an adverb, its sentiment orientation needs to be ascertained. As previously mentioned, a pair has an initial positive or negative polarity when it is mentioned as a pros or cons, respectively. Therefore, in analyzing a pros review statement, if an $O_{(i)}$ carries a positive feeling such as *'good'*, it is considered to be the same sentiment orientation type as that of the review. We utilize Sentistrength to find the polarity strength of $O_{(i)}$. Taking into account the absolute amount of the obtained *polarity strength*, the initial polarity is adjusted by Υ in cases where $O_{(i)}$ is categorized as the *same sentiment orientation.* Due to the significant role of adjectives in expressing the feelings (Benamara et al., 2007), we consider the highest value, or 1, in our range for Υ. If SentiStrength returns the polarity value as zero, we use the Bing Lexicon to determine the sentiment orientation of the word $O_{(i)}$. In case that the Bing Lexicon returns the sentiment orientation the same as that of review's initial polarity, we adjust its polarity with Υ^{*}. However, as there is a conflict between the results of employed lexicons, we pick a lower value for Υ^{*} as compared to that of Υ to reflect the existing uncertainty.

Occurrence of words that have the opposite sentiment orientation as that of the review's initial polarity. There may be cases where $O_{(i)}$ does not convey the same sentiment orientation as that of the review from which it has been extracted. Such cases happen for the following reasons:

- The customer mentions his negative experience by mistake in a pros review or vice versa. For example in a pros review, the customer may mention 'it is heavy'.
- The sentiment word is a context-based one. For instance, the word 'large', which can be perceived as either positive or negative depending on its context.
- A negation keyword negates the sentiment. For example, although 'good' carries a positive orientation, we expect 'not good' to happen in a cons review as the negation will invert the polarity of the sentiment.

In order to reduce the adverse effect of such errors on the final results, in PCSA we adjust the polarity of these words with a value close to 0. Hence, if SentiStrength returns a polarity value opposite to that of the review for an $O_{(i)}$, the algorithm diminishes its sentiment value with η. This will result in giving more strength to the negation words and then to adverbs as compared to opinions with an opposite sentiment orientation. In the case where the word is deemed neutral based on SentiStrength but shows an opposite orientation in the Bing Lexicon, the algorithm modifies its polarity with η'. Similar to the reasoning discussed in the previous section, we allocate a smaller value to η' as compared to that of η.

Occurrence of words without any sentiment (neutral). There may be cases where $O_{(i)}$ has no positive/negative polarity due to the incompleteness of the sentiment lexicon, errors in extracting the feature-opinion pairs, or using the objective opinions (e.g. foldable). In such cases, PCSA will not update the initial polarity of a word and will leave it with the initial assigned value of (± 1).

Step 4: Determining the polarity of words that represent implicit features of the product

This group contains the implicit aspects or high-level features where the user only mentions an opinion word without indicating any specific feature. Unlike the explicit features, implicit features are not obviously stated and mostly refer to a high-level feature or functionality of the product. For example, the word 'heavy' implicitly expresses the negative opinion of the customer towards the *weight* of the product while no feature has been explicitly used. We use the method discussed in Fig. 8 to determine the polarity of these opinions. Since they are mainly related to the high-level features of a product, we assign their polarity value to the whole product (tree-root).

Stage 2: Computing the polarity value of each feature

In order to ensure consistent results, the obtained polarity strength is normalized to remain in the range of [−1, +1]. Once the polarity of the extracted pairs is determined, the next step is to compute the overall sentiment score of each discussed feature. Similar to the approach used in Lau et al. (2014), PCSA calculates the mean of obtained positive/negative polarity scores by each feature considering all the reviews which discuss that feature. The polarity results for the features that have common meanings (refer to step 2 of stage 2.3) are aggregated. For instance, the final sentiment score of the feature 'image' includes the ones for 'picture' and 'photograph' as well. The mean of the implicit opinions' polarity scores as well as those of the *functional features* is considered as the overall sentiment score of the whole product. Finally, the features are presented along with their obtained positive and negative scores according to the customers' opinions. These results can be utilized for several purposes such as assessing the performance of different product components by the product manufacturers, estimating the product's sales and so forth.

Considering the following pros review of a camera from www.testfreaks.com, Fig. 9 summarizes how each phase of PCSA works to extract features from this comment and assign polarity strength to them.

"The Nikon D800 boasts stellar photos, excellent videos, quality/price ratio, and a relatively streamlined shooting design."

4. Experimental results

To demonstrate and evaluate the effectiveness of PCSA in relation to feature-opinion extraction and polarity determination, we carry out a range of experiments and compare the results with existing techniques. For feature-opinion extraction, PCSA's results are compared with SenticNet (Cambria et al., 2017), frequency-based technique (Chen et al., 2015; Hu & Liu, 2004; Lau et al., 2014; Manek et al., 2016) and the rule-based approach proposed by Liu and Seneff (2009). To check the perfomance of the proposed polarity determination approach, SentiStrength (Thelwall, 2013) and AFINN (Nielsen, 2011) are used as the baseline methods. In relation to feature-opinion extraction, evaluation metrics such as *True Positive (TP), False Positive (FP), False Negative (FN), precision, recall*, and *F1-score* (Lau et al., 2014; Qiu et al., 2011; Schouten & Frasincar, 2016; Zhao, Rubinstein, Gemmell, & Han, 2012) are used to compare the results of PCSA with state-of-the-art techniques. To check the accuracy of the polarity determination of reviews, *correlation* and *Mean Squared Error (MSE)* metrics are used. We report our experimental results in four categories as follows.

Category 1 reports the results of PCSA's different steps in accomplishing the feature-opinion extraction. *Category 2* compares

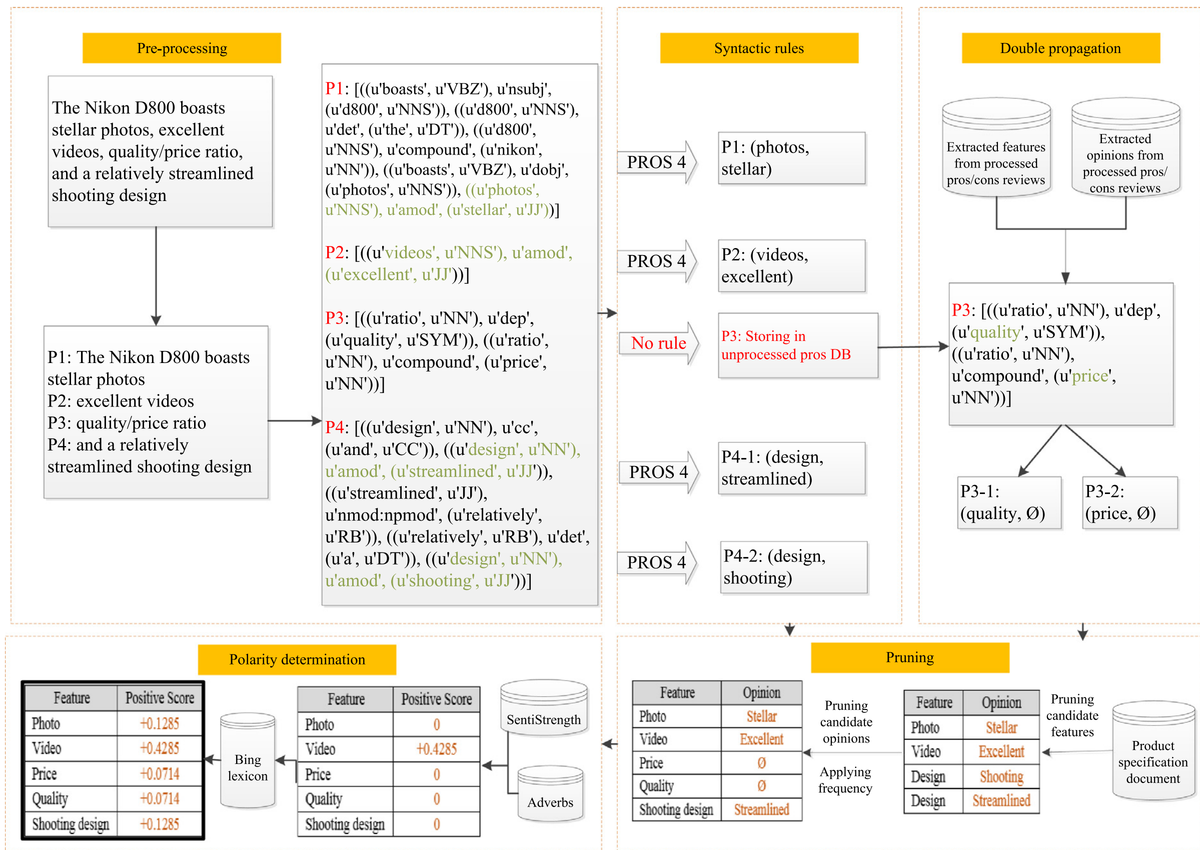


Fig. 9. Showing the application of PCSA on a real-world example feature-opinion extraction and polarity determination.

the performance of PCSA in extracting features against state-of-the-art techniques. *Category 3* demonstrates PCSA's results in determining the polarity strength of the opinions and compares it with those of two polarity lexicons. *Category 4* presents the aggregated sentiment scores for the discussed features.

4.1. Category 1: Effectiveness of PCSA in feature/opinion extraction from user reviews

To test the effectiveness of PCSA, we crawled pros/cons reviews on a digital camera (Nikon D800) from different websites including www.cnet.com and www.testfreaks.com etc., and collected 400 pros and cons type reviews. In the pre-processing step, the reviews were split into statements using different delimiters. Next, Stanford tools were used to identify the POS tags as well as the dependency relations within each statement. Depending on whether it is a pros or cons review, PCSA applies the syntactic rules defined in Tables 2, 3 and 4 to extract the product features from each, along with the opinion word (if any). Next, the double propagation algorithm defined in Table 5 is used to expand the coverage of PCSA and find the feature/opinion word. Finally, to enhance the results, PCSA leverages a taxonomy-based pruning technique to remove the redundant results. To gain insight into the efficacy of each individual step of PCSA, its performance is ascertained using the following metrics (Zhuang et al., 2006):

$$Precision = \frac{No.\ of\ correctly\ mined\ feature/opinion\ pairs}{No.\ of\ all\ mined\ feature/opinion\ pairs} \quad (2')$$

$$Recall = \frac{No.\ of\ correctly\ mined\ feature/opinion\ pairs}{No.\ of\ all\ correct\ feature/opinion\ pairs} \quad (3)$$

$$F1\ score = \frac{2 \times precision \times recall}{precision + recall} \quad (4)$$

We have asked one of our colleagues to determine the discussed feature(s) along with its associated opinion word, if any exist, from each individual statement. This set of results is termed as 'human mind set' and the performance of PCSA in extracting features and opinion words is evaluated against that. Table 6 presents the abovementioned metrics after applying the different steps of PCSA in mining only the features from the pros/cons reviews. A separate computation of these metrics, as shown in Table 7, has been undertaken to show the effectiveness of PCSA in extracting opinion words from the reviews. The second column of Tables 6 and 7 presents the performance of PCSA in extracting features and opinions while applying syntactic rules. The third column indicates PCSA's results after using DP along with the syntactic rules. Finally, the fourth column reports the effectiveness of PCSA's pruning technique in enhancing the results from DP and syntactic rules.

According to the values of *precision* and *recall* in Tables 6 and 7 when only the syntactic rules are used, PCSA's performance is acceptable in both extracting candidate features and opinions. Applying the next layer of PCSA, Double Propagation, results in a significant improvement in the performance of the evaluation metrics. As the pruning task is responsible for removing the redundant results, its main impact is seen in the *precision* value. However, since our taxonomy is not complete, it is likely that PCSA removes a correct candidate feature (e.g. *warranty*) and its corresponding opinion word from the final results as it cannot find any match for it in the product-tree. Hence, a slight drop in the recall value is expected after applying the pruning method. Despite losing a few correct results, using the pruning method results in increased precision. In other words, the proposed pruning technique performs well in removing redundant results. The average of the F1 score in Tables 5 and 6 shows a notable improvement in PCSA's accuracy as

Table 6
Results of PCSA in feature extraction in different stages.

	Syntactic rules (SR)		SR+ Double Propagation (DP)		SR+ DP+ Pruning	
	Pros	Cons	Pros	Cons	Pros	Cons
Precision	0.470	0.745	0.810	0.749	0.882	0.872
Recall	0.461	0.562	0.794	0.839	0.788	0.817
F1 score	0.466	0.640	0.802	0.792	0.832	0.844
Average F1 score	0.553		0.797		**0.838**	

Table 7
Results of PCSA in opinion mining in different stages.

	Syntactic rules (SR)		SR+ Double Propagation (DP)		SR+ DP+ Pruning	
	Pros	Cons	Pros	Cons	Pros	Cons
Precision	0.646	0.595	0.792	0.575	0.884	0.712
Recall	0.763	0.668	0.935	0.861	0.888	0.791
F1 score	0.699	0.630	0.857	0.690	0.886	0.749
Average F1 score	0.664		0.773		**0.817**	

Table 8
Results of the syntactic rules, which are specifically defined for the cons reviews.

	Precision	Recall	F1-score
Cons specific syntactic rules	0.561	0.776	0.647

the different layers of DP and pruning are added to the syntactic rules.

Moreover, as mentioned in Section 3.2, in addition to the rules which have been developed to analyze both pros and cons reviews, a different rule set is developed to deal with cons reviews. Using this cons specific rule set, we were able to further analyze 11% of the cons dataset. This proves that the cons reviews are different in grammatical structure from the pros reviews. Specific syntactic rules should be defined to analyze them more accurately. Table 8 presents the performance of the cons rules in feature and opinion extraction. The high value of *recall* indicates that these rules perform well in extracting the actual features/opinions from the cons reviews.

4.2. Category 2: Comparing PCSA's results in feature extraction against existing methods

To show the effectiveness of PCSA in extracting features from pros/cons type reviews, we compare its results with three existing approaches using our collected dataset. As the baseline method for analyzing pros/cons type of reviews, we select the approach proposed by Liu and Seneff (2009). We modeled their approach based on the syntactic rules mentioned in their paper and applied it to our dataset. Furthermore, as extracting frequent nouns is an established technique for identifying features from almost all review types (as shown in Table 1), we utilized it as our second baseline method. We extracted 580 nouns from our dataset and considered those that were repeating at least ten times as product features. As the last benchmarking, we ran SenticNet (Cambria et al., 2017) over our dataset and compared its results in feature extraction with PCSA.

The performance of selected techniques is evaluated in the different statistical metrics such as True Positive (TP), False Positive (FP), and False Negative (FN). TP represents the terms that are correctly labeled as the product's features. FP demonstrates those cases that were incorrectly extracted as the product's features. Real features which were not extracted are referred as False Negatives. To gain more insight into the performance of different techniques in feature extraction, we also compute the precision, recall, and F1-score for all techniques.

The performance of PCSA and the baseline methods in feature extraction over the collected dataset is reported in Table 9. It is shown that PCSA outperforms the state-of-the-art approaches in extracting more correct features (TP), fewer redundant words (FP) and missing fewer actual features (FN). Considering *precision* and *recall*, although SenticNet shows an acceptable *recall* value, its *precision* is poor. In other words, SenticNet is good in mining the correct features, however, apart from product features, it extracts many redundant words as well. Unlike SenticNet, the method proposed by Liu and Seneff (2009) performs better in extracting fewer non-features compared to its ability in capturing actual product features from the reviews (higher *precision* as compared to the *recall* value). Although utilizing frequent nouns as the product's features presents promising *precision* and *recall*, PCSA outperforms this technique in all of the statistical metrics. Overall, PCSA has a higher average F1 score value compared to all baseline methods thus demonstrating its superiority in feature extraction from pros/cons type reviews.

Table 9
Comparing results of PCSA against existing techniques in extracting features from pros/cons type reviews.

	Liu et al. (2009)		Frequent nouns		SenticNet		PCSA	
	Pros	Cons	Pros	Cons	Pros	Cons	Pros	Cons
True Positive	354	156	507	206	450	242	**567**	**259**
False Positive	114	88	196	101	469	427	**76**	**38**
False Negative	366	161	212	111	269	75	**153**	**58**
Precision	0.756	0.639	0.721	0.671	0.490	0.346	**0.882**	**0.872**
Recall	0.492	0.492	0.705	0.650	0.626	0.763	**0.788**	**0.817**
F1-score	0.596	0.556	0.713	0.660	0.549	0.476	**0.832**	**0.844**
Average F1 score	0.576		0.686		0.513		**0.838**	

4.3. Category 3: Demonstrating PCSA's results in polarity determination and comparing it with the results of existing techniques

In this category, we explain the working of PCSA's polarity determination algorithm using an example and then compare its performance with two state-of-the-art techniques against the results mentioned in the 'human mind set'.

As previously discussed, in order to determine the polarity of the pros/cons type reviews, unlike the existing approaches, PCSA considers not only the orientation of this format of reviews but also ascertains the strength of the extracted opinion words. In the following example, we explain the workings of this phase as shown in Fig. 8. Consider the following *pros* type of review for a camera:

Pros: *battery, foldable display, good lens, very excellent images, low noise, the shutter is not bad, fast.*

As a result of applying the first two phases of our proposed methodology, seven individual pairs of (feature, opinion) are extracted from this review which are shown in Table 10. Since these pairs are discussed in a pros review, they are assigned an initial polarity of (+1). Next, the initial polarity is modified using various intensifiers according to the presence of different types of opinion words. For instance, the feature 'images' in the row #4 has a combination of adverb and adjective in its associated opinion, namely *very + excellent*. PCSA breaks it down and recognizes *very* as an adverb and *excellent* as an adjective with the same orientation. Using the pre-defined intensifiers for the presence of an adverb as well as an adjective, the algorithm then ascertains the sentiment score for this pair. Similarly, for the feature 'noise' in the row #5, PCSA determines that the opinion word *low* has an opposite orientation with that of a pros statement and uses the respective intensifier to compute its sentiment score. In cases where SentiStrength returns a polarity value of zero for an opinion word, like rows #2 and #7, the orientation of the word is double-checked by the Bing Lexicon. If the Bing Lexicon reports an orientation other than neutral for an opinion word (e.g. #7), the initial polarity of that word is modified based on the pre-defined intensifiers. The last column presents the normalized sentiment score for each pair. Table 10 shows the procedure of computing the sentiment scores for each extracted feature. The values of the intensity factors are 0.33, 0.75, 1, 0.25, 0.8 and 0.2 for α, β, Υ, η, $\Upsilon^{'}$, and $\eta^{'}$, respectively. As the *requesting features* possess high popularity based on the customers' opinions, we consider the highest sentiment score for them. In the case of an existing implicit feature or *opinion word only pair* like *fast*, we consider the whole product (camera) as the corresponding aspect.

4.3.1. Evaluating the performance of the polarity determination algorithm

Since the existing work on pros/cons type of reviews mostly consider the orientation of these reviews rather than determining their polarity strength, we lack a baseline method with which to compare the result of this phase[1]. Hence, we compare the polarity determination result of PCSA against two approaches. The first one, AFINN, which is a list of 2477 English words and phrases that are annotated in the scale of −5 to +5 based on their polarity strength (Nielsen, 2011). We compare PCSA's results with SentiStrength, as the second benchmarking, to show the efficacy of our heuristic approach in determining the polarity of words and phrases in the collected dataset. To do so, we asked one of our colleagues, who is a native English speaker, to assign a polarity value in the range of

[1] In (J. Liu & Seneff, 2009), the polarity of pros/cons reviews are computed in the scale of 1 to 5 based on the ratings of the reviews. However, we could not model their polarity assignment approach as we lacked the review's ratings in our collected dataset.

Table 10
Example of workings of the polarity determination phase.

Serial number	(Feature, Opinion)	Initial polarity	Negation (α=0.33)	Adverb ($\beta = 0.75$)	SentiStrength score (L)	Bing Lexicon	Same orientation ($\Upsilon = 1$, $\Upsilon^{'} = 0.8$)	Opposite orientation ($\eta = 0.25$, $\eta^{'} = 0.20$)	Normalized sentiment score
1	(battery, Ø)	+1	-	-	-	-	-	-	(battery, +0.0714)
2	(display,foldable)	+1	-	-	0	No	-	-	(display, +0.0714)
3	(lens,good)	+1	-	-	+1	-	P(1) = 1*1*(1+1) = 2	-	(lens, +0.1428)
4	(images,very excellent)	+1	-	P(1)=1*(1+0.75)= 1.75	- +3	- -	- P(2) = 1.75*3*(1+1) = 10.5	- -	(images, +0.75)
5	(noise, low)	+1	-	-	-1	-	-	P(1) = 1*1*(0.25) = 0.25	(noise, +0.0178)
6	(shutter, not-bad)	+1	P(1)=1*(1+0.33)= 1.33 -	- -	- -1	- -	- -	- P(2) = 1.33*1*(0.25) = 0.3325	(shutter, +0.0237)
7	(Ø, fast)	+1	-	-	0	Yes	P(1) = 1*(1+0.8) = 1.8	-	(camera, +0.1285)

Table 11
Results of PCSA against AFINN and SentiStrength in assigning polarity strength to the extracted opinions.

	PCSA		AFINN		SentiStrength	
	Pros	Cons	Pros	Cons	Pros	Cons
Correlation with the expert	0.8152	0.7846	0.5090	0.5899	0.7533	0.7398
Average Correlation	**0.7999**		0.5494		0.7465	
MSE	0.0077	0.0095	0.0138	0.0128	0.0088	0.0080
Average MSE	**0.0086**		0.0133		0.0084	

Table 12
Aggregated polarity results for each feature.

Feature	No. positive statements	Average positive score	No. negative statements	Average negative score
Autofocus/AF	29	0.394	18	-0.191
Flash	7	0.214	1	-0.085
Viewfinder	14	0.110	0	0
Video	20	0.827	9	-0.079
Battery	6	0.229	15	-0.187
Menu	9	0.265	2	-0.107
Memory card	17	0.285	15	-0.383
Image	39	0.671	14	-0.090

$+1$ to -1 to the extracted opinion words. The performance of PCSA and the selected benchmarking approaches in determining the polarity strength of the extracted opinion words are compared with that of human mind using correlation and Mean Squared Error (MSE) metrics (Benamara et al., 2007; Schouten & Frasincar, 2016). MSE measures the average of the squared deviations between the observed values and the actual (true) ones. It is computed using Eq. (5) (Wang & Bovik, 2009):

$$MSE = \frac{1}{n}\sum_{i=1}^{n}\left(\hat{y}_i - y_i\right)^2 \tag{5}$$

In Eq. (5), $\hat{y}$ represents the vector of predicted polarity by each approach, y is the actual sentiments' vector (expert's scores), and n stands for the total number of extracted opinions.

Table 11 presents the results of PCSA, AFINN and SentiStrength against that given by the expert and computes the correlation and MSE as the evaluation metrics. PCSA's result shows higher correlation with that of the expert (human mind) in analyzing opinions from both the pros and cons type of reviews as compared to SentiStrength and AFINN. The strong positive correlation between PCSA and the expert's scoring indicates that the results of PCSA are close to that of a human mind in assigning polarity strength to the opinion words.

Considering MSE, PCSA outperforms AFINN in both pros and cons reviews. However, in comparing the PCSA's polarity scores with those of SentiStrength, although PCSA ends in better MSE in analyzing pros reviews, it performs slightly weaker in dealing with the opinion words extracted from the cons reviews. This can be a result of our basic assumption in this study, by which we consider no sentiment category as *neutral* in PCSA. Based on this assumption, since we are dealing with the pros/cons reviews, even posting an objective comment (a fact) about a product by the customer conveys a postive/negative feeling towards that phenomenon. Considering the more frequent usage of objective opinions by customers in writing the cons reviews, more neutral polarity scores might be assigned by a human mind to the opinion words extracted from cons reviews. Hence, this assumption ends in a greater deviation in cons reviews as compared to that of the pros ones.

4.4. Category 4: PCSA's aggregated sentiment results for each feature

Once the polarity of all pairs is determined, PCSA aggregates the results for each feature considering its probable synonyms. It then presents the final sentiment score as the average of the obtained polarity scores. In our considered dataset, PCSA was able to extract 100 unique features. Of these, six features were labeled *requesting features* (e.g. GPS, small raw option, image stabilizer, screen swivel etc.). Table 12 presents a snapshot of the results for some of the camera's features in the collected reviews.

As a practical application of the results shown in Table 12, product designers can use this information to determine those features that are more liked/disliked by customers. They can also gain insight from the *requesting features* in developing the next generation of their product and capturing a broader market share.

5. Conclusions and future work

In this paper, we proposed the PCSA framework to analyze users' sentiments at the feature-level in the pros/cons types of product reviews. The proposed method first extracts the product's features and their corresponding opinions in the reviews using syntactic rules. The coverage of the grammatical rules is extended using a double propagation algorithm. To enhance the accuracy of PCSA in feature-opinion extraction, a novel taxonomy-based approach is proposed to prune the redundant results. Finally, the polarity of the opinion words is determined by taking into consideration the intensity of the expressed emotions. Through various experiments, we demonstrated the superiority of PCSA in feature-opinion extraction and polarity determination.

To the best of our knowledge, few studies have focused on analyzing the pros/cons format of reviews. PCSA is the first framework to determine the polarity strength of such reviews apart from considering only their sentiment orientations. However, it has its own limitations. First, it can only be utilized for analyzing product pros/cons reviews and not the service/movie ones. Second, since the syntactic rules are developed based on English grammatical structures, the proposed framework is language dependent. Next, the result of the pruning step relies on the completeness of the product specification document. Lastly, we consider the same sentiment strength for all adverbs while computing the polarity score of the extracted opinion words. We will address this limitation in our future work to consider the intensity degree of different adverbs. We also aim to extend our model so that it can be applied to the next task of sentiment analysis, which is aggregating the results from online reviews. We will also investigate how to consider the intensity degree of different adverbs during polarity determination. One more extension to this study is automating the process of extracting syntactic rules when being applied over large datasets.

Supplementary materials

Supplementary material associated with this article can be found, in the online version, at doi:10.1016/j.eswa.2018.07.046.

References

Agarwal, B., Mittal, N., Bansal, P., & Garg, S. (2015). Sentiment analysis using common-sense and context information. *Computational Intelligence and Neuroscience, 2015*, 9 Article ID 715730. doi:10.1155/2015/715730.

Agarwal, S., & Siddiqui, T. J. (2009). Using syntactic and contextual information for sentiment polarity analysis. In *Proceedings of the 2nd international conference on interaction sciences information technology, culture and human - ICIS '09* (pp. 620–623). doi:10.1145/1655925.1656037.

Alghunaim, A., Mohtarami, M., Cyphers, S., & Glass, J. (2015). A vector space approach for aspect based sentiment analysis. In *In Proceedings of NAACL-HLT* (pp. 116–122).

Araque, O., Corcuera-Platas, I., Sánchez-Rada, J. F., & Iglesias, C. A. (2017). Enhancing deep learning sentiment analysis with ensemble techniques in social applications. *Expert Systems with Applications, 77*, 236–246. doi:10.1016/j.eswa.2017.02.002.

Bagheri, A., Saraee, M., & De Jong, F. (2013). Care more about customers: Unsupervised domain-independent aspect detection for sentiment analysis of customer reviews. *Knowledge-Based Systems, 52*, 201–213. doi:10.1016/j.knosys.2013.08.011.

Benamara, F., Irit, S., Cesarano, C., Federico, N., & Reforgiato, D. (2007). Sentiment analysis : Adjectives and adverbs are better than adjectives alone. In *In Proc of int conf on weblogs and social media* (pp. 1–4). http://doi.org/citeulike-article-id:9387439.

Bhattacharjee, S., Das, A., Bhattacharya, U., Parui, S. K., & Roy, S. (2015). Sentiment analysis using cosine similarity measure. In *IEEE 2nd international conference on Recent Trends in Information Systems (ReTIS)* (pp. 27–32). doi:10.1109/ReTIS.2015.7232847.

Bright, D. A. (2017). Using social network analysis to design crime prevention strategies: A case study of methamphetamine manufacture and trafficking. In B. LeClerc, & E. U. Savona (Eds.), *Crime prevention in the 21st century: Insightful approaches for crime prevention initiatives* (pp. 143–164). Cham: Springer International Publishing. doi:10.1007/978-3-319-27793-6_11.

Cambria, E., Poria, S., Gelbukh, A., & Nacional, I. P. (2017). Sentiment analysis is a big suitcase. *IEEE Intelligent Systems*, 74–80. doi:10.1109/MIS.2017.4531228.

Carenini, G., Ng, R. T., & Zwart, E. (2005). Extracting knowledge from evaluative text. In *K-CAP '05: Proceedings of the 3rd international conference on Knowledge capture* (pp. 11–18). doi:10.1145/1088622.1088626.

Carter, D., & Inkpen, D. (2015). Inferring aspect-specific opinion structure in product reviews using co-training. In *16th international conference in computational linguistics and intelligent text processing: 9042* (pp. 225–240). doi:10.1007/978-3-642-28601-8.

Chen, D., & Manning, C. (2014). A Fast and accurate dependency parser using neural networks. In *Proceedings of the 2014 conference on Empirical Methods in Natural Language Processing (EMNLP)* (pp. 740–750). doi:10.3115/v1/D14-1082.

Chen, Y. Y., Ferrer, X., Wiratunga, N., & Plaza, E. (2015). Aspect selection for social recommender systems. In E. Hüllermeier, & M. Minor (Eds.). In *Case-based reasoning research and development. ICCBR 2015. Lecture Notes in Computer Science: 9343*. Cham: Springer.

Fang, X., & Zhan, J. (2015). Sentiment analysis using product review data. *Journal of Big Data, 2*, 5. doi:10.1186/s40537-015-0015-2.

Ganapathibhotla, M., & Liu, B. (2008). Mining opinions in comparative sentences. In *Proceedings of the 22nd international conference on computational linguistics - COLING '08* (pp. 241–248). *1*(August). doi:10.3115/1599081.1599112.

Ganeshbhai, S. Y., & Shah, B. K. (2015). Feature based opinion mining: A survey. In *Souvenir of the 2015 IEEE international advance computing conference, IACC 2015* (pp. 919–923). doi:10.1109/IADCC.2015.7154839.

García-Pablos, A., Cuadros, M., & Rigau, G. (2015). Unsupervised word polarity tagging by exploiting continuous word representations. *Procesamiento de Lenguaje Natural, 55*, 127–134.

Goldberg, A. B., Fillmore, N., Andrzejewski, D., Xu, Z., Gibson, B., & Zhu, X. (2009). May all your wishes come true : A study of wishes and how to recognize them. In *Proceedings of human language technologies: The 2009 annual conference of the north american chapter of the association for computational linguistics on - NAACL '09* (pp. 263–271). doi:10.3115/1620754.1620793.

Hogenboom, A., Van Iterson, P., Heerschop, B., Frasincar, F., & Kaymak, U. (2011). Determining negation scope and strength in sentiment analysis. In *Conference proceedings - IEEE international conference on systems, man and cybernetics* (pp. 2589–2594). doi:10.1109/ICSMC.2011.6084066.

Hu, M., & Liu, B. (2004). Mining opinion features in customer reviews. In *19th national conference on artifical intelligence* (pp. 755–760). doi:10.1145/1014052.1014073.

Huang, J., Etzioni, O., Zettlemoyer, L., Clark, K., & Lee, C. (2012). Revminer: An extractive interface for navigating reviews on a smartphone. In *Proceedings of the 25th annual ACM symposium on user interface software and technology* (pp. 3–12). doi:10.1145/2380116.2380120.

Jeyapriya, A., & Selvi, C. S. K. (2015). Extracting aspects and mining opinions in product reviews using supervised learning algorithm. In *2nd international conference on electronics and communication systems, ICECS 2015* (pp. 548–552). IEEE. doi:10.1109/ecs.2015.7124967.

Jhamtani, H., Chhaya, N., Karwa, S., Varshney, D., Kedia, D., & Gupta, V. (2015). Identifying suggestions for improvement of product features from online product reviews. In T.-Y. Liu, C. N. Scollon, & W. Zhu (Eds.), *Social informatics* (pp. 112–119). Switzerland: Cham: Springer International Publishing.

Kang, Y., & Zhou, L. (2016). RubE: Rule-based methods for extracting product features from online consumer reviews. *Information & Management, 54*(2). doi:10.1016/j.im.2016.05.007.

Kim, S.-M., & Hovy, E. (2006). Automatic identification of pro and con reasons in online reviews. In *Proceedings of the COLING/ACL on main conference poster sessions* (pp. 483–490). doi:10.3115/1273073.1273136.

Lau, R. Y. K., Li, C., & Liao, S. S. Y. (2014). Social analytics: Learning fuzzy product ontologies for aspect-oriented sentiment analysis. *Decision Support Systems, 65*(2014), 80–94. doi:10.1016/j.dss.2014.05.005.

Li, Y. M., Chen, H. M., Liou, J. H., & Lin, L. F. (2014). Creating social intelligence for product portfolio design. *Decision Support Systems, 66*, 123–134. doi:10.1016/j.dss.2014.06.013.

Liu, B. (2012). *Sentiment analysis and opinion mining*. Morgan & Claypool Publishers. doi:10.2200/S00416ED1V01Y201204HLT016.

Liu, B. (2015). *Sentiment analysis: mining opinions, sentiments, and emotions*. Cambridge: Cambridge University Press. doi:10.1017/CBO9781139084789.

Liu, B., & Hu, M. (2005). Opinion observer: analyzing and comparing opinions on the web. In *Proceedings of the 14th international World Wide Web conference (WWW-2005)*.

Liu, J., & Seneff, S. (2009). Review sentiment scoring via a parse-and-paraphrase paradigm. In *Proceedings of the 2009 conference on empirical methods in natural language processing volume 1 - EMNLP '09: 1* (p. 161). doi:10.3115/1699510.1699532.

Liu, Q., Gao, Z., Liu, B., & Zhang, Y. (2015). Automated rule selection for aspect extraction in opinion mining. In *Twenty-fourth International Joint Conference on Artificial Intelligence (IJCAI 2015)* (pp. 1291–1297).

Lobeck, A. C. (2000). *Discovering grammar: An introduction to english sentence structure*. Oxford University Press.

Manek, A. S., Shenoy, P. D., Mohan, M. C., & Venugopal, K. R. (2016). Aspect term extraction for sentiment analysis in large movie reviews using Gini Index feature selection method and SVM classifier. *World Wide Web*. doi:10.1007/s11280-015-0381-x.

Manning, C., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S., & McClosky, D. (2014). The Stanford CoreNLP natural language processing toolkit. In *Proceedings of 52nd annual meeting of the association for computational linguistics: system demonstrations* (pp. 55–60). doi:10.3115/v1/P14-5010.

Miller, G. A. (1995). WordNet: A lexical database for english. *Communications of the ACM, 38*(11), 39–41.

Mirtalaie, M. A., Hussain, O. K., & Chang, E. (2016). FEATURE : New product development using feature- drift based framework for unique aspect recommendation. *IEEE international conference on e-business engineering*. doi:10.1109/ICEBE.2016.43.

Mirtalaie, M. A., Hussain, O. K., Chang, E., & Hussain, F. K. (2017a). A decision support framework for identifying novel ideas in new product development from cross-domain analysis. *Information Systems, 69*, 59–80. doi:10.1016/j.is.2017.04.003.

Mirtalaie, M. A., Hussain, O. K., Chang, E., & Hussain, F. K. (2017b). Sentiment analysis of specific product's features using product tree for application in new product development. In *Advances in Intelligent Networking and Collaborative Systems (INCoS 2017)* (pp. 82–95). Springer.

Mukherjee, S., & Joshi, S. (2013). Sentiment aggregation using conceptnet ontology. In *Proceedings of the sixth international joint conference on natural language processing* (pp. 570–578). (October)Retrieved from. http://www.aclweb.org/anthology/I13-1065 .

Nielsen, F. Å. (2011). A new ANEW: Evaluation of a word list for sentiment analysis in microblogs. In *CEUR workshop proceedings: 718* (pp. 93–98). doi:10.1016/j.knosys.2015.06.015.

Norman, B., & Ann, M. J. (2011). Mapping and leveraging influencers in social media to shape corporate brand perceptions. *Corporate Communications: An International Journal, 16*(3), 184–191. doi:10.1108/13563281111156853.

Pasierbinska-Wilson, Z.. *Will social data insights kill the marketing focus group?* Retrieved 2 February 2018, from https://www.martechadvisor.com/articles/bi-ci-amp-data-visualization/will-social-data-insights-kill-the-marketing-focus-group/.

Poria, S., Cambria, E., Ku, L.-W., Gui, C., & Gelbukh, A. (2014). A rule-based approach to aspect extraction from product reviews. *Workshop on Natural Language Processing for Social Media (SocialNLP)*, 28–37.

Qiu, G., Liu, B., Bu, J., & Chen, C. (2011). Opinion word expansion and target extraction through double propagation. *Computational Linguistics, 37*, 9–27. doi:10.1162/coli_a_00034.

Ramanand, J., Bhavsar, K., & Pedanekar, N. (2010). Wishful thinking - Finding suggestions and "buy" wishes from product reviews. In *Proceedings of the NAACL HLT 2010 workshop on computational approaches to analysis and generation of emotion in text* (pp. 54–61).

Rana, T. A., & Cheah, Y. N. (2017). A two-fold rule-based model for aspect extraction. *Expert Systems with Applications, 89*, 273–285. doi:10.1016/j.eswa.2017.07.047.

Samha, A. K., Li, Y., & Zhang, J. (2015). Aspect - Based opinion mining from product reviews using conditional random fields. *AusDM 2015: The 13th Australasian Data Mining Conference*. University of Technology.

Schouten, K., & Frasincar, F. (2016). Survey on aspect-level sentiment analysis. *IEEE Transactions on Knowledge and Data Engineering, 28*(3), 813–830. doi:10.1109/TKDE.2015.2485209.

Suleman, K., & Vechtomova, O. (2015). Discovering aspects of online consumer reviews. *Journal of Information Science*, 1–15. doi:10.1177/0165551515595742.

Thelwall, M. (2013). Heart and soul: Sentiment strength detection in the social web with sentistrength. *Proceedings of the CyberEmotions, 5*, 1–14. doi:10.1007/978-3-319-43639-5_7.

Toutanova, K., Klein, D., Manning, C. D., & Singer, Y. (2003). Feature-rich part-of-speech tagging with a cyclic dependency network. In *Proceedings of the 2003 conference of the North American chapter of the association for computational linguistics on human language technology - NAACL '03: 1* (pp. 173–180). doi:10.3115/1073445.1073478.

Wang, H., & Wang, W. (2014). Product weakness finder: An opinion-aware system through sentiment analysis. *Industrial Management & Data Systems, 114*(8), 1301–1320. Retrieved from http://www.emeraldinsight.com/doi/pdfplus/10.1108/IMDS-02-2014-0069 .

Wang, Z., & Bovik, A. C. (2009). Mean squared error: Lot it or leave it? A new look at signal fidelity measures. *IEEE Signal Processing Magazine, 26*(1), 98–117. doi:10.1109/MSP.2008.930649.

Xie, Y., Chen, Z., Cheng, Y., Zhang, K., Agrawal, A., Liao, W.-K., & Choudhary, A. (2013). Detecting and tracking disease outbreaks by mining social media data. In *Proceedings of the twenty-third international joint conference on artificial intelligence*. AAAI Press.

Ye, K., Li, L., Guo, M., Qian, Y., & Yuan, H. (2015). Summarizing product aspects from massive online review with word representation. *Knowledge Science Engineering Management, 9403*(71572029), 318–323. doi:10.1007/978-3-319-25159-2.

Yu, J., Zha, Z.-J., Wang, M., & Chua, T.-S. (2011). Aspect ranking: Identifying important product aspects from online consumer reviews. In *Proceedings of the 49th annual meeting of the association for computational linguistics* (pp. 1496–1505). doi:10.1109/CC.2013.6488828.

Zhao, B., Rubinstein, B. I. P., Gemmell, J., & Han, J. (2012). A Bayesian approach to discovering truth from conflicting sources for data integration. *Vldb, 5*(6), 550–561. doi:10.14778/2168651.2168656.

Zhuang, L., Jing, F., & Zhu, X.-Y. (2006). Movie review mining and summarization. In *Proceedings of the 15th ACM International conference on information and knowledge management - CIKM '06* 43. doi:10.1145/1183614.1183625.