

Product Feature Taxonomy Learning based on User Reviews

Nan Tian¹, Yue Xu¹, Yuefeng Li¹, Ahmad Abdel-Hafez¹ and Audun Josang²

¹Faculty of Science and Engineering, Queensland University of Technology, Brisbane, Australia

²Department of Informatics, University of Oslo, Oslo, Norway

Keywords: Feature Extraction, Opinion Mining, Association Rules, Feature Taxonomy, User Reviews.

Abstract: In recent years, the Web 2.0 has provided considerable facilities for people to create, share and exchange information and ideas. Upon this, the user generated content, such as reviews, has exploded. Such data provide a rich source to exploit in order to identify the information associated with specific reviewed items. Opinion mining has been widely used to identify the significant features of items (e.g., cameras) based upon user reviews. Feature extraction is the most critical step to identify useful information from texts. Most existing approaches only find individual features about a product without revealing the structural relationships between the features which usually exist. In this paper, we propose an approach to extract features and feature relationships, represented as a tree structure called feature taxonomy, based on frequent patterns and associations between patterns derived from user reviews. The generated feature taxonomy profiles the product at multiple levels and provides more detailed information about the product. Our experiment results based on some popularly used review datasets show that our proposed approach is able to capture the product features and relations effectively.

1 INTRODUCTION

In recent years, the user generated online content exploded due to the advent of Web 2.0. For instance, online users write reviews to how they enjoy or dislike a product they purchased. This helps to identify features or characteristics of the product from users' point of view, which is an important addition to the product specification. However, to identify the relevant features from users' subjective review data is extremely challenging.

Feature-based opinion mining has attracted big attention recently. A significant amount of research has been proposed to improve the accuracy of feature generation for products (Hu and Liu, 2004a; Scaffidi et al., 2007; Hu et al., 2010; Zhang and Zhu, 2013; Popescu and Etzioni, 2005; Ding et al., 2008). However, most techniques only extract features; the structural relationship between product features has been omitted. For example, "picture resolution" is a common feature of digital camera in which "resolution" expresses the specific feature concept to describe the general feature "picture". Yet, existing approaches treat "resolution" and "picture" as two individual features instead of finding the relationship between them. Thus, the information derived by existing feature extraction approaches is not sufficient for gen-

erating a precise product model since all features are allocated in the same level and independent from each other.

Association rule mining is a well explored method in data mining (Pasquier et al., 1999). Based on association rules generated from a collection of item transactions, we can discover the relations between items. However, the amount of generated association rules is usually huge and selecting the most useful rules is challenging (Xu et al., 2011). In our research, we propose to identify a group of frequent patterns as potential features to assist selecting useful association rules. The selected rules are used to identify relationships between features. Furthermore, in order to ensure that the most useful rules are to be selected, we also propose to apply statistical topic modelling technique (Blei et al., 2003) to the selection of association rules.

Our approach takes advantages of existing feature extraction approaches and makes two contributions. Firstly, we present a method to make use of association rules to find related features. Secondly, we create a product model called feature taxonomy which represents the product more accurately by explicitly representing the concrete relationships between general features and specific features.

2 RELATED WORK

Our research aims to extract useful product information based on user generated information to create a product model. This work is closely related to feature-based opinion mining which has drawn many researchers' attention in recent years. In detail, identifying features that have been mentioned by users is considered the most significant step in opinion mining (Hai et al., 2013). Hu and Liu (2004) first proposed a feature-based opinion mining method to extract features and sentiments from customer reviews. They use pattern mining to find frequent itemsets (nouns). These itemsets are pruned and considered frequent product features. A list of sentiment words (adjectives) that are nearby frequent features in reviews can be extracted and used to identify those product features that cannot be identified by pattern mining. Scaffidi et al. (2007) improved the performance of feature extraction in their proposed system called Red Opal. Specifically, they made use of a language model to find features by comparing the frequency of nouns in the review and in common use of English. Those frequent nouns in both reviews and in common use are considered invalid features. Hu et al. (2010) make use of SentiWordNet to identify all sentences that may contain users' sentiment polarity. Then, the pattern mining is applied to generate explicit features based on these opinionated sentences. In addition, a mapping database has been constructed to find those implicit features represented by sentiment words (e.g., *expensive* indicates *price*). To enhance the accuracy of finding correct features from free text review, Hai et al (2013) proposed a novel method which evaluates the domain relevance of a feature by exploiting features' distribution disparities across different corpora (domain-dependent review corpus such as cellphone reviews and domain-irrelevant corpus such as culture article collection). In detail, the *intrinsic-domain relevance* (IDR) and *extrinsic-domain relevance* (EDR) have been proposed to benchmark if a examined feature is related to a certain domain. The candidate feature with low IDR and high EDR scores will be pruned.

Lau et al. (2009) presented an ontology-based approach to profile the product. In detail, a number of ontology levels, such as feature level that contains identified features for a certain product and sentiment level in which sentiment words that describe a certain feature are stored, have been constructed (Lau et al., 2009). This method provides a simple product profile rather than extracting product features only.

The statistical topic modeling technique has been used in various fields such as text mining (Blei et al.,

2003; Hofmann, 2001) in recent years. Latent Semantic Analysis (LSA) is first proposed to capture the most significant features of a document collection based upon semantic structure of relevant documents (Lewis, 1992). Then, Probabilistic LSA (pLSA) (Hofmann, 2001) and Latent Dirichlet Allocation (LDA) (Blei et al., 2003) are proposed to improve the interpretation of results from LSA. These techniques have been proven more effective on document modeling and topic extraction, which are represented by topic-document and word-topic distribution, respectively. Particularly, multinomial distribution over words which is derived based upon word frequency can be generated to represent topics in a given text collection.

None of aforementioned feature identification approaches is able to identify the relationships between the extracted product features. The structural relationships that exist between features can be used to describe the reviewed product in more depth. However, how to evaluate and determine the relations between features is still challenging.

The remainder of the paper is organized as follows. The next section illustrates the construction process of our proposed feature taxonomy. Then, the evaluation of our approach is reported afterwards. Finally, we conclude and describe future direction of our research work.

3 THE PROPOSED APPROACH

Our proposed approach consists of two main steps: product taxonomy construction using association rules and taxonomy expansion based on reference features. The input of our system is a collection of user reviews for a certain product. The output is a product feature taxonomy which contains not only all generated features but also the relationships between them.

3.1 Pre-processing and Transaction File Generation

First of all, we construct a single document called an *aggregated review document* which combines all the reviews in a collection of reviews, keeping each sentence in the original reviews as one sentence in the constructed *aggregated review document*. Three steps are undertaken to process the review text in order to extract useful information. Firstly, we generate the part-of-speech (POS) tag for each word in the *aggregated review document* to indicate whether the word is a *noun*, *adjective* or *adverb* etc. For instance, after the POS tagging, "*The flash is very weak.*" would

be transformed to “*The/DT flash/NN is/VBZ very/RB weak/JJ .I.*”, where *DT*, *NN*, *VBZ*, *RB*, and *JJ* represent Determiner, Noun, Verb, Adverb and Adjective, respectively. Secondly, according to the thumb rule that most product features are nouns or noun phrases (Hu and Liu, 2004b), we process each sentence in the *aggregated review document* to only keep words that are nouns. All the remaining nouns are also pre-processed by stemming and spelling correction. Each sentence in the *aggregated review document* consists of all identified nouns of a sentence in the original reviews. Finally, a transactional dataset is generated from the *aggregated review document*. Each sentence which consists of a sequence of nouns in the *aggregated review document* is treated as a transaction in the transactional dataset.

3.2 Potential Features Generation

Our first task is to generate potential product features that are expressed by those identified nouns or noun phrases. According to (Hu and Liu, 2004a), significant product features are discussed extensively by users in reviews (e.g., “*battery*” for cameras). Upon this, most existing feature extraction approaches make use of pattern mining techniques to find potential features. Specifically, an itemset is a set of items (i.e., words in review text in this paper) that appear together in one or multiple transactions in a transactional dataset. Given a set of items, $I = \{i_1, i_2, \dots, i_n\}$, an itemset is defined as $X \subseteq I$. The support of an itemset X , denoted as $Supp(X)$, is the percentage of transactions in the dataset that contain X . All frequent itemsets from a set of transactions that satisfy a user-specified minimum support will be extracted as the potential features. However, not all frequent itemsets are genuine since some of them may be just frequent but meaningless. We use compactness pruning method proposed by (Hu and Liu, 2004a) to filter frequent itemsets. After the pruning, we can get a list of frequent itemsets that are considered potential features, denoted as *FP*.

3.3 Product Feature Taxonomy Construction

In this step, we propose to utilize association rules generated from the discovered potential product features to identify relations in order to construct a feature taxonomy.

Association rule mining can be described as follows: Let $I = \{i_1, i_2, \dots, i_n\}$, be a set of items, and the dataset consists of a set of transactions $D = \{t_1, t_2, \dots, t_m\}$. Each transaction t contains a subset of

items from I . Therefore, an association rule r represents an implication relationship between two itemsets which can be defined as the form $X \rightarrow Y$, where $X, Y \subseteq I$ and $X \cap Y = \emptyset$. The itemsets X and Y are called antecedent and consequent of the rule, respectively. To assist selecting useful rules, the support $Supp(X \cup Y)$ and the confidence $Conf(X \rightarrow Y)$ of the rule can be used (Xu et al., 2011).

For easily describing our approach, we define some useful and important concepts as follows:

Definition 1 (Feature Taxonomy): A feature taxonomy consists of a set of features and their relationships, denoted as $FH = \{F, L\}$, F is a set of features where $F = \{f_1, f_2, \dots, f_n\}$ and L is a set of relations. The feature taxonomy has the following constraints:

- (1) The relationship between a pair of features is the sub-feature relationship. For $f_i, f_j \in F$, if f_j is a sub feature of f_i , then (f_i, f_j) is a link in the taxonomy and $(f_i, f_j) \in L$, which indicates that f_j is more specific than f_i . f_i is called the parent feature of f_j and denoted as $P(f_j)$.
- (2) Except for the root, each feature has only one parent feature. This means that the taxonomy is structured as a tree.
- (3) The root of the taxonomy represents the product itself.

Definition 2 (Feature Existence): For a given feature taxonomy $FH = \{F, L\}$, let $W(g)$ represent a set of words that appear in a potential feature g , let $ES(g) = \{a_i | a_i \in 2^{w(g)}, a_i \in F\}$ contain all subsets of g which exist in the feature taxonomy, $ES(g)$ is called the existing subsets of g , if $\bigcup_{a_i \in ES(g)} W(a_i) = W(g)$, then g is considered exist in FH , denoted as $exist(g)$, otherwise $\neg exist(g)$.

Opinion mining is also referred as sentiment analysis (Subrahmanian and Reforgiato, 2008; Abbasi et al., 2008; Wright, 2009). Adjectives or adverbs that appear together with product features are considered as the sentiment words in opinion mining. The following definition defines the sentiment words that are related to a product feature.

Definition 3 (Related Sentiments): For a feature $f \in F$, let $RS(f)$ denote a set of sentiment words which appear in the same sentences as f in user reviews, $RS(f)$ is defined as the related sentiments of f .

Definition 4 (Sentiment Sharing): For features $f_1, f_2 \in F$, the sentiment sharing between f_1 and f_2 is defined as $SS(f_1, f_2) = |RS(f_1) \cap RS(f_2)|$.

For deriving sub features using association rules, we need to select a set of useful rules rather than using all the rules. In the next two subsections, we will first

propose two methods to select rules, one method is to select rules based on the sentiment sharing among features and the other method is to select rules by using the word relatedness derived from the results generated by using the typical topic model technique method LDA (Blei et al., 2003); then introduce some strategies to update the feature taxonomy by adding sub features using the selected rules.

In order to explain the topic modelling based method, we first define some related concepts. Let $RE = \{r_1, r_2, \dots, r_M\}$ be a collection of reviews, each review consists of nouns only, $W = \{w_1, w_2, \dots, w_n\}$ be a set of words appearing in RE , and $Z = \{Z_1, \dots, Z_v\}$ be a set of pre-specified hidden topics. LDA can be used to generate topic models for representing the collection as a whole and also for each review in the collection. At the collection level, the topic model represents the collection RE using a set of topics each of which is represented by a probability distribution over words (i.e., nouns in the context of this paper) for topic. In this paper, we will use the collection level representation to find the relatedness between words.

At collection level, each topic Z_j is represented by a probability distribution over words, $\phi_j = \{p(w_1|Z_j), p(w_2|Z_j), \dots, p(w_n|Z_j)\}$, $\sum_{k=1}^n \phi_{j,k} = 1$, $p(w_k|Z_j)$ is the probability of word w_k being used to represent the topic Z_j . Based on the probability $p(w_k|Z_j)$, we can choose the top words to represent the topic Z_j .

Definition 5 (Topic Words): Let $\phi_j = \{p(w_1|Z_j), p(w_2|Z_j), \dots, p(w_n|Z_j)\}$ be the topic representation for topic Z_j produced by LDA and $0 \leq \delta \leq 1$ be a threshold, a set of the topic words for Z_j , denoted as $TW(Z_j)$, is defined as $TW(Z_j) = \{w|w \in W, p(w|Z_j) > \delta\}$.

Definition 6 (Word Relatedness): We use word relatedness to indicate how likely that two words have been used to represent a topic together. Let $w_i, w_j \in W$ be two words, the word relatedness between two words with respect to topic z is defined below:

$$WR_z(w_i, w_j) = \begin{cases} 1 - |p(w_i|z) - p(w_j|z)| & w_i \in TW(z) \\ & \text{and } w_j \in TW(z) \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

Definition 7 (Feature Topic Representation): For feature $f \in F$, let $WD(f)$ be a set of words appearing in f and $TW(z)$ be the topic words of topic z . If $WD(f) \subset TW(z)$, the feature topic representation of feature f for topic z is defined as $FTP(f, z) = \{(w, p(w|z)) | w \in WD(f)\}$.

Definition 8 (Feature Relatedness): For features $f_i, f_j \in F$, if both features appear in a certain topic

z , then the feature relatedness between f_i and f_j with respect to z is defined as:

$$FR_z(f_i, f_j) = \min_{\substack{w_i \in WD(f_i) \\ w_j \in WD(f_j)}} \{WR_z(w_i, w_j)\} \quad (2)$$

3.3.1 Rule Selection

Let $R = \{r_1, r_2, \dots, r_n\}$ be a set of association rules generated from the frequent itemsets FP , each rule r in R has the form $X_r \rightarrow Y_r$, X_r and Y_r are the antecedent and consequent of r , respectively.

Assuming that f_e is a feature which has already been in the current feature taxonomy FH , to generate the sub features for f_e , we first select a set of candidate rules, denoted as $R_{f_e}^c$, which could be used to generate the sub features:

$$R_{f_e}^c = \{X \rightarrow Y | X \rightarrow Y \in R, X = f_e, \text{Supp}(X) > (Y)\} \quad (3)$$

As defined in Equation (3), the rules in $R_{f_e}^c$ should satisfy two constraints. The first constraint, $X = f_e$, specifies that the antecedent of a selected rule must be the same as the feature f_e . Sub features represent specific cases of a feature, they are more specific compared to the feature. The second constraint is based on the assumption that more frequent itemsets usually represent more general concepts, and less frequent itemsets usually represent more specific concepts. For instance, according to our observation toward features, a general feature (e.g., "picture", its frequency is 62) appears more frequently than a specific feature (e.g., "resolution", its frequency is 9) in reviews for the camera 2 in the dataset published by Liu (Ding et al., 2008). Therefore, only the rules which can derive more specific features will be selected.

However, not all selected rules represent correct sub-feature relationship. For instance, $mode \rightarrow auto$ is more appropriate for describing a sub-feature relationship rather than $camera \rightarrow auto$. Therefore, the rule $camera \rightarrow auto$ should not be considered when we generate the sub features for "camera". Upon this, we aim to prune the unnecessary rules before generating sub features for each taxonomy feature. Firstly, a feature and its sub features should share similar sentiment words since they describe the same aspect of a product at different abstract levels (e.g., *vivid* can be use to describe both *picture* and *color*). Therefore, we should select rules whose antecedent (representing the feature) and consequent (representing a possible sub feature) share as many sentiment words as possible because the more sentiment words they share, the more possible they are about the same aspect of the product. Secondly, based on topic models

generated from LDA, the more a feature and its potential sub feature appear in the same topics, the more likely they are related to each other.

Let f_X, f_Y be two features and $Z_{(f_X, f_Y)}$ be a set of topics that contains both features, the feature relatedness between f_X, f_Y with respect to all topics, denoted as $FR_{avg}(f_X, f_Y)$, is defined as the average feature relatedness between the two features over $Z_{(f_X, f_Y)}$:

$$FR_{avg}(f_X, f_Y) = \frac{\sum_{z \in Z_{(f_X, f_Y)}} FR_z(f_X, f_Y)}{|Z_{(f_X, f_Y)}|} \quad (4)$$

Based on this view, we propose the following equation to calculate a score for each candidate rule $X \rightarrow Y$ in $R_{f_e}^c$:

$$Weigh(X \rightarrow Y) = \alpha(Supp(Y) \times Conf(X \rightarrow Y)) + \beta \frac{SS(X, Y)}{|RS(X) \cup RS(Y)|} + \gamma FR_{avg}(X, Y) \quad (5)$$

$0 < \alpha, \beta, \gamma < 1$. The value of α, β , and γ is set to 0.8, 0.1, and 0.1, respectively in our experiment described in Section 4. There are three parts in Equation (5). The first part is used to measure the belief to the consequent Y by using this rule since $Conf(X \rightarrow Y)$ measures the confidence to the association between X and Y and $Supp(Y)$ measures the popularity of Y . The second part is the percentage of the shared sentiment words given by $SS(X, Y)$ over all the sentiment words used for either X or Y . Yet, the third part in the equation is the average feature relatedness between X and Y . Given a threshold σ , we propose to use the following equation to select the rules from the candidate rules in $R_{f_e}^c$. The rules in R_{f_e} will be used to derive sub features for the features in FP . R_{f_e} is called the rule set of f_e .

$$R_{f_e} = \{X \rightarrow Y | X \rightarrow Y \in R_{f_e}^c, Weigh(X \rightarrow Y) > \sigma\} \quad (6)$$

3.3.2 Feature Taxonomy Construction

Let $FH = \{F, L\}$ be a feature taxonomy which could be an empty tree, FP be a set of frequent itemsets generated from user reviews which are potential features, and R be a set of rules generated from user reviews. This task is to construct a feature taxonomy if F is empty or update the feature taxonomy if F is not empty by using the rules in R . Let UF be a set of features on the tree which need to be processed in order to construct or update the tree. If F is empty, the itemset in FP which has the highest support will be chosen as the root of FH , it will be the only item

in UF at the beginning. If F is not empty, UF will be F , i.e., $UF = F$.

Without losing generality, assuming that F is not empty and the set of features currently on the tree, UF is the set of features which need to be processed to update or construct the tree. For each feature in UF , let f_e be a feature in UF , i.e., $f_e \in UF$ and $X \rightarrow Y \in R_{f_e}$ be a rule with $X = f_e$, the next step is to decide whether or not Y should be added to the feature taxonomy as a sub feature of f_e . There are two possible situations: Y does not exist in the feature taxonomy, i.e., $\neg exist(Y)$ and Y does exist in the taxonomy, i.e., $exist(Y)$. In the first situation, the feature taxonomy will be updated by adding Y as a sub feature of f_e , i.e., $F = F \cup \{Y\}$, $L = L \cup (f_e, Y)$, and Y should be added to UF for further checking.

In the second situation, i.e., Y already exists in the taxonomy, i.e., according to Definition 2, there are two cases, $Y \notin ES(Y)$ (i.e., Y is not in the tree) or $Y \in ES(Y)$ (i.e., Y is in the tree). In the first case, Y is not considered a sub feature of f_e and consequently, no change is required to the tree. In the second case, $\exists f_y \in F$, f_y is the parent feature of Y , i.e., $P(Y) = f_y$ and $(f_y, Y) \in L$. Now, we need to determine whether to keep f_y as the parent feature of Y or change the parent feature of Y to f_e . That is, we need to examine f_y and f_e to see which of them is more suitable to be the parent feature of Y . The basic strategy is to compare f_y and f_e to see which of them has more sentiment sharing and feature relatedness with Y . Let f_p, f_c be a potential parent feature and sub feature, respectively. We propose a ranking equation to indicate how likely f_c is related to f_p : $Q(f_p, f_c) = \frac{SS(f_p, f_c)}{RS(f_c)} + FR_{avg}(f_p, f_c)$. Thus, if $Q(f_y, Y) < Q(f_e, Y)$, the link (f_y, Y) will be removed from the taxonomy tree, (f_e, Y) will be added to the tree, otherwise, no change to the tree and f_y is still the parent feature of Y .

3.3.3 Algorithms

The construction of the feature taxonomy is to generate a feature tree by finding all sub features for each feature. In this section, we will describe the algorithms to construct the feature taxonomy. As mentioned above, if the tree is empty, the feature with the highest support will be chosen as the root. So, at the very beginning, F and UF contain at least one item which is the root. Algorithm 1 describes the method to construct or update a feature taxonomy.

After the taxonomy construction, some potential features may be left over in RF and have not been added to the taxonomy. The main reason is because these itemsets may not frequently occur in the reviews

Algorithm 1: Feature Taxonomy Construction.**Input:**
 $R, FH = \{F, L\}, FP.$
Output:
 FH, RF // RF is the remaining features which are not added to FH after the construction

- 1: if $F = \emptyset$, then $root := \operatorname{argmax}_{f \in FP} \{supp(f)\}$,
 $F := UF := \{root\}$;
- 2: else $UF := F$;
- 3: for each feature $f_e \in UF$
- 4: if $R_{f_e} \neq \emptyset$ //the rule set of f_e is not empty
- 5: for each rule $X \rightarrow Y \in R_{f_e}$
- 6: if $\neg exist(Y)$ // Y does not exist on the tree
- 7: $F := F \cup \{Y\}, L := L \cup (f_e, Y)$,
 $UF := UF \cup \{Y\}, FP := FP - \{Y\}$;
- 8: else // Y exists on the tree
- 9: if $Y \in ES(Y)$ and $Q(f_y, Y) < Q(f_e, Y)$
 // f_y is Y 's parent feature
- 10: $L := L \cup (f_e, Y), L := L - (f_y, Y)$;
 //add (f_e, Y) and remove (f_y, Y)
- 11: else // $Y \notin ES(Y)$, Y is not on the tree
- 12: $FP := FP - \{Y\}$;
- 13: endfor
- 14: endif
- 15: $UF := UF - \{f_e\}$; //remove f_e from UF
- 16: endfor
- 17: $RF := FP$

together with the features that have been added in the taxonomy. In order to prevent valid features from being missed out, we check those remaining itemsets in RF by examining the shared sentiment words and feature relatedness between the remaining itemsets and the features in the taxonomy. Let $FH = \{F, L\}$ be the constructed feature taxonomy, RF be the set of remaining potential features, for a potential feature g in RF , the basic strategy to determine whether g is a feature or not is to examine the Q ranking between g and the features in the taxonomy. Let $F_g = \{f | f \in F, Q(f, g) > 0\}$ be a set of features which are related to g , if $F_g \neq \emptyset$, g is considered a feature. The most related feature is defined as $f_m = \operatorname{argmax}_{f \in F_g} \{Q(f, g)\}$. g will be added to the taxonomy with f_m as its parent feature. If there are multiple such features f_m which have the highest ranking score with g , the one with the highest support will be chosen as the parent feature of g .

Algorithm 2 formally describes the method mentioned above to expand the taxonomy by adding the remaining features.

After the expansion, the features left over in RF are not considered as features for this product.

Algorithm 2: Feature Taxonomy Expansion.**Input:**
 $FH = \{F, L\}, RF.$
Output:
 FH

- 1: for each feature $g \in RF$
- 2: if $(F_g := \{f | f \in F, Q(f, g) > 0\}) \neq \emptyset$
- 3: $M := \{a | a \in F_g \text{ and}$
 $Q(a, g) = \max_{f \in F_g} \{Q(f, g)\}\}$
- 4: $f_m := \operatorname{argmax}_{f \in M} \{supp(f)\}$
- 5: $F := F \cup \{g\}, L := L \cup (f_m, g)$
- 6: $RF := RF - \{g\}$

4 EXPERIMENT AND EVALUATION

We use three datasets in the experiments. Each dataset contains user reviews for a certain type of digital cameras. One dataset is used in (Hu and Liu, 2004a), while the other two are used in (Ding et al., 2008). Each review in the datasets has been manually annotated. In detail, a human examiner read a review sentence by sentence. If a sentence is considered indicating the user's opinions, such as positive and negative, all possible features in the sentence that are modified by sentiment words are tagged. We take these annotated features as the correct features to evaluate the performance of our proposed method in feature extraction. The number of reviews and number of annotated features are 51 and 98 for camera 1, 34 and 75 for camera 2, and 45 and 105 for camera 3.

Our proposed feature taxonomy captures both product features and relations between features. Therefore, the evaluations are twofold: feature extraction evaluation and structural relations evaluation.

4.1 Feature Extraction Evaluation

First of all, we evaluate the performance of our approach by examining the number of accurate features in user reviews that have been extracted. We use the feature extraction method (FBS) proposed in (Hu and Liu, 2004a) as the baseline for comparison. In addition, in order to examine the effectiveness of using the sentiment sharing measure, the feature relatedness measure, and the combination of the two, we conduct our experiment in four runs:

- (1) *Rule*: construct the feature taxonomy by only utilizing the information of association rules (i.e., support and confidence value only) without using the sentiment sharing and the feature relatedness measures;

- (2) *SS*: construct the feature taxonomy by taking the information of association rules and the sentiment sharing measure without using the feature relatedness measure;
- (3) *FR*: construct the feature taxonomy by taking the information of association rules and the feature relatedness measure without using the sentiment sharing measure;
- (4) *Hybrid*: the sentiment sharing and the feature relatedness are combined together with the information of association rules to construct the feature taxonomy.

Table 1: Recall Comparison.

	Camera 1	Camera 2	Camera 3	Average
FBS	0.57	0.63	0.57	0.59
Rule	0.38	0.52	0.45	0.45
SS	0.56	0.65	0.58	0.60
FR	0.56	0.67	0.58	0.60
Hybrid	0.56	0.68	0.58	0.61

Table 2: Precision Comparison.

	Camera 1	Camera 2	Camera 3	Average
FBS	0.45	0.42	0.51	0.46
Rule	0.55	0.57	0.74	0.62
SS	0.62	0.57	0.63	0.61
FR	0.60	0.56	0.63	0.60
Hybrid	0.62	0.59	0.68	0.63

Table 3: F1 Score Comparison.

	Camera 1	Camera 2	Camera 3	Average
FBS	0.50	0.50	0.54	0.51
Rule	0.45	0.54	0.56	0.52
SS	0.59	0.61	0.60	0.60
FR	0.58	0.61	0.60	0.60
Hybrid	0.59	0.63	0.63	0.62

Table 1, 2, 3 illustrate the recall, precision, and F1 score results produced in the four runs, respectively. From the results, we can see that using both the sentiment sharing and feature relatedness can obtain better feature extraction performance than the use of association rule's information only. In particular, the hybrid method, which uses both sentiment sharing and feature relatedness, achieves the best results in most cases. However, the size of the review dataset and the number of annotated features can affect the precision and recall, which makes the values of the precision and recall vary in different range for different datasets. For instance, camera 3 has higher precision values than camera 2 due to more reviews in camera 3 dataset than that in camera 2 dataset, but camera 3 has lower recall values than camera 2 due to more manually annotated features in camera 3 dataset.

4.2 Structural Relation Evaluation

The evaluation of the relations requires the standard taxonomy or knowledge from experts (Tang et al.,

2009). Since there is no existing standard taxonomy available for comparison, we manually created taxonomy for the three cameras according to the product technical specifications provided online by manufacture organizations^{1, 2, 3}. From the product specifications on these websites, each camera has a number of attributes such as *lens system* and *shooting modes*. In addition, each attribute may also have several sub attributes. For instance, the *shooting modes* of the camera contains more specific attributes (e.g., *intelligent auto* and *custom*). Based upon such information, we create the product feature taxonomy for three digital cameras and use the taxonomy as the testing taxonomy, called Manual Feature Taxonomy (*MFT*), to evaluate the relations within our proposed feature taxonomy.

Due to the difference between the technical specifications from domain experts and the subjective reviews from online users, the words used to represent a feature in user reviews are very often different from the words for the same feature specified by domain experts in the product specification. For example, the feature *lens system* in the testing taxonomy and the feature *lens* in our generated taxonomy should be the same according to common knowledge even though they are not exactly matched with each other. Because of this fact, we will determine the match between two features based on overlapping of the two features rather than exact matching.

Let $MFT = \{F_{MFT}, L_{MFT}\}$ be the testing taxonomy with F_{MFT} being a set of standard features given by domain experts and L_{MFT} being a set of links in the testing taxonomy. For a given link $(f_{Fp}, f_{Fc}) \in L$ in the constructed product feature taxonomy and two features $f_{Mp}, f_{Mc} \in F_{MFT}$ in the testing taxonomy, the link (f_{Fp}, f_{Fc}) is considered matched with (f_{Mp}, f_{Mc}) and therefore represent a correct feature relation if the following conditions are satisfied:

1. $W(f_{Mp}) \cap W(f_{Fp}) \neq \emptyset$ and $W(f_{Mc}) \cap W(f_{Fc}) \neq \emptyset$
2. There exists a path in MFT , $\langle f_{Mp}, f_1, f_2, \dots, f_n, f_{Mc} \rangle$, (f_{Mp}, f_1) , (f_i, f_{i+1}) , $(f_n, f_{Mc}) \in L_{MFT}$, $i = 1, \dots, n - 1$

We examine the testing taxonomy and the constructed taxonomy to identify all matched links in the constructed taxonomy. The traditional measures precision and recall are used to evaluate the correctness of the feature relations in the constructed feature taxonomy. Let $ML(FH)$ denote the matched links in

¹<http://www.canon.com.au/Personal/Products/Camerasand-Accessories/Digital-Cameras/PowerShot-S100>

²<http://www.nikonusa.com/en/Nikon-Products/Product/Compact-Digital-Cameras/26332/COOLPIX-S4300.html>

³http://www.usa.canon.com/cusa/support/consumer/digital_cameras/powershot_g_series/powershot_g3#Specifications

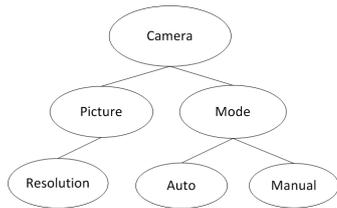


Figure 1: Constructed Feature Taxonomy.

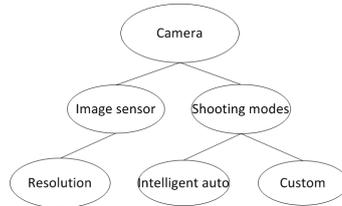


Figure 2: Testing Feature Taxonomy.

the constructed taxonomy, the precision and recall are defined as : Precision = $ML(FH)/|L|$ and Recall = $ML(FH)/|L_{MFT}|$.

Table 4: Recall and Precision of Relation Evaluation

	Relations in MFT	Relations in FH	Recall	Precision
Camera 1	75	97	0.40	0.46
Camera 2	63	97	0.57	0.65
Camera 3	71	102	0.51	0.57

Table 4 illustrates the evaluation results including the number of relations within the testing taxonomy, the number of relations within our generated taxonomy, recall and precision for the three different cameras, respectively. From the results, we can see that our generated feature taxonomy correctly capture around 50% of the relationships. Figure 1 and Figure 2 show a part of the feature taxonomy generated from our proposed approach and the testing taxonomy generated based on the product specification available online given by domain experts, respectively. From the comparison, our generated feature taxonomy identifies the relation between *picture* and *resolution*. Although the testing taxonomy uses more technical terms, which are *image sensor* instead of *picture*; in fact, they refer to the same attribute of the camera according to common knowledge. Similarly, the *(mode, auto)* and *(shooting modes, intelligent auto)* indicate the same relationship between two features.

As aforementioned, the online users and manufacture experts may describe the same feature by using totally different terms or words. This does affect the performance (both recall and precision) of our proposed approach in feature relationship identification negatively. For instance, the user may prefer using “*manual*” to depict a specific camera mode option. By contrast, the manufacture experts usually pick the term “*custom*” to describe this sub feature which be-

longs to “*shooting modes*”. In such a case, the two relations: *(mode, manual)* and *(shooting modes, custom)* cannot match.

5 CONCLUSION AND FUTURE WORK

In this paper, we introduced a product feature taxonomy learning approach based on frequent patterns and association rules. The objective is to not only extract product features mentioned in user reviews but also identify the relationship between the generated features. The results of our experiment indicate that our proposed approach is effective in both identifying correct features and structural relationship between them. Particularly, the feature relationships captured in the feature taxonomy provide more detailed information about products. This leads us to represent products profiles as multi-levels of feature, rather than a single level as most other methods do.

In the future, we plan to improve and evaluate our proposed product model by utilizing semantic similarity tools. For instance, the vocabulary mismatch can be handled by examining the semantic similarity when we undertake the structural relation evaluation. In addition, we plan to develop a review recommender system that makes use of the proposed product model in order to identify high quality reviews. The structural relations of the product model are able to assist identifying some characteristics of reviews, such as how a certain feature and its sub features have been discussed and how many different features have been covered. Our system will therefore aim at recommending reviews based upon such criteria to help users make purchasing decisions.

REFERENCES

- Abbasi, A., Chen, H., and Salem, A. (2008). Sentiment analysis in multiple languages: Feature selection for opinion classification in web forum. *ACM Transactions on Information Systems*, 26(3).
- Blei, D. M., Ng, A. Y., and Jordan, M. I. (2003). Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:993 – 1022.
- Ding, X., Liu, B., and Yu, P. S. (2008). A holistic lexicon-based approach to opinion mining. In *Proceedings of the 2008 International Conference on Web Search and Data Mining*, pages 231 – 240.
- Hai, Z., Chang, K., Kim, J., and Yang, C. (2013). Identifying features in opinion mining via intrinsic and extrinsic domain relevance. *IEEE Transactions on Knowledge and Data Engineering*, pages 1 – 1.

- Hofmann, T. (2001). Unsupervised learning by probabilistic latent semantic analysis. *Machine Learning*, 42(1 - 2):177 – 196.
- Hu, M. and Liu, B. (2004a). Mining and summarizing customer reviews. In *10th ACM SIGKDD international conference on Knowledge discovery and data mining*.
- Hu, M. and Liu, B. (2004b). Mining opinion features in customer reviews. In *Proceedings of the 19th national conference on Artificial intelligence*.
- Hu, W., Gong, Z., and Guo, J. (2010). Mining product features from online reviews. In *IEEE International Conference on E-Business Engineering*, pages 24 – 29.
- Lau, R. Y., Lai, C. C., Ma, J., and Li, Y. (2009). Automatic domain ontology extraction for context-sensitive opinion mining. In *Proceedings of the Thirtieth International Conference on Information Systems*.
- Lewis, D. D. (1992). An evaluation of phrasal and clustered representations on a text categorization task. In *Proceedings of the 15th ACM International Conference on Research and Development in Information Retrieval*, pages 177 – 196.
- Pasquier, N., Bastide, Y., Taouil, R., and Lakhal, L. (1999). Efficient mining of association rules using closed itemset lattices. *Information Systems*, 24(1):25 – 46.
- Popescu, A.-M. and Etzioni, O. (2005). Extracting product features and opinions from reviews. In *Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*, pages 339–346.
- Scaffidi, C., Bierhoff, K., Chang, E., Felker, M., Ng, H., and Jin, C. (2007). Red opal: Product-feature scoring from reviews. In *Proceedings of the 8th ACM conference on Electronic commerce*, number 182 - 191.
- Subrahmanian, V. S. and Reforgiato, D. (2008). Ava: Adjective-verb-adverb combinations for sentiment analysis. *IEEE Intelligent Systems*, pages 43 – 50.
- Tang, J., Leung, H.-f., Luo, Q., Chen, D., and Gong, J. (2009). Towards ontology learning from folksonomies. In *Proceedings of the 21st international joint conference on Artificial intelligence*, pages 2089 – 2094.
- Wright, A. (2009). Our sentiments, exactly. *Communications of the ACM*, 52(4):14 – 15.
- Xu, Y., Li, Y., and Shaw, G. (2011). Representations for association rules. *Data and Knowledge Engineering*, 70(6):237 – 256.
- Zhang, Y. and Zhu, W. (2013). Extracting implicit features in online customer reviews for opinion mining. In *Proceedings of the 22nd international conference on World Wide Web companion*, pages 103 – 104.