Effective Dependency Rule-based Aspect Extraction for Social Recommender Systems

Indicate Submission Type: Completed Research Paper

Y.Y. Chen
School of Computing Science and Digital Media, Robert Gordon University
Aberdeen, UK
y.y.chen@rgu.ac.uk

N. Wiratunga
School of Computing Science and Digital Media, Robert Gordon University
Aberdeen, UK
n.wiratunga@rgu.ac.uk

R. Lothian
School of Computing Science and Digital Media, Robert Gordon University
Aberdeen, UK
r.m.lothian@rgu.ac.uk

Abstract

Social recommender systems capitalise on product reviews to generate recommendations that are both guided by experiential knowledge and are explained by user opinions centred on important product aspects. Therefore, having an effective aspect extraction algorithm is crucial. Previous work has shown that dependency relation approaches perform well in this task. However, they can also lead to erroneous extractions. This paper proposes an effective aspect extraction approach that combines strengths of both dependency relations and frequent noun approaches. Further, we demonstrate how aspect-level sentiment analysis can be used to enrich product representations and thereby positively impact recommendation effectiveness. We empirically evaluate our proposed approach with the objective to recommend products that are 'better' than a given query product. A computational measure of 'better' is used in our experiments with five real-world datasets. Results show that our proposed approach achieves significantly better results than the existing state-of-the-art dependency-based methods in recommendation tasks.

Keywords: social recommender systems, aspect extraction, dependency relations, aspect-based sentiment analysis

Introduction

Recommender systems approaches such as collaborative filtering and content-based filtering rely on user ratings and product descriptions to recommend products (Koren et al., 2009; Sarwar et al., 2001; Lops et al., 2011). In recent years, recommender systems research has focused on exploiting knowledge from social media text and so the emergence of social recommender systems. Increasingly effort is being focused on utilising users' purchased experiences from product reviews to enhance recommender performance. Consider the following review example:

"The camera has a good lens but the battery is not very durable."

Here, the reviewer expresses conflicting opinions about two aspects of a camera - aspect lens connotes positive sentiment whilst aspect battery is negative. Clearly such fine-grained opinions are important, in that they explain a consumer's preferences that drive their purchase decisions and should naturally influence the workings of recommender systems. Thus, aspect extraction and aspect level sentiment analysis forms a crucial knowledge source for computational modelling of consumer preferences.

There are two main approaches for aspect extraction: supervised and unsupervised. A supervised approach is least favoured due to the challenges in obtaining ground truth data to evaluate the performance of new algorithms. The most popular unsupervised approach extracts aspects that are found to be frequent nouns and noun phrases. However, not all frequent nouns are aspects. For example, we may see "Canon" (identified as a noun) frequently occur in Canon camera product reviews but this is not a genuine product aspect. Technical aspects of a camera such as "aperture" (opening of a camera through which light travels) may only appear in reviews written by professional photographers and as such is likely to be missed out by being infrequent.

Existing work has shown that unsupervised dependency relation based approaches outperform both frequent noun and supervised approaches (Qiu et al. 2011, Poria et al. 2014). This is partly due to important infrequent aspects that get typically filtered-out being extracted by the dependency relation rules. Since dependency based methods extract aspects by means of syntactic relations between a pair of words in a sentence they are not restricted to just frequent aspects. Once a specific relation between words is identified, this triggers an aspect extraction involving these words. There are 47 dependency relations defined in the Universal Dependencies for English[1] and every sentence can trigger more than one dependency relation. Previous work selects a subset of the dependency relation rules without providing information on how the rules were chosen (Moghaddam & Ester 2012; Qiu et al. 2011; Poria et al. 2014). It is important to have the information in order to select relevant dependency rules as the irrelevant rules can result in erroneous aspects. Consider the examples in Figures 1 and 2 together with word dependencies.

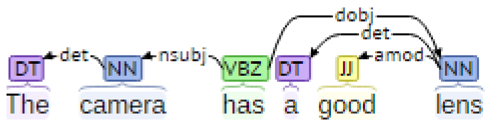


Figure 1. Dependency Relations – Example 1

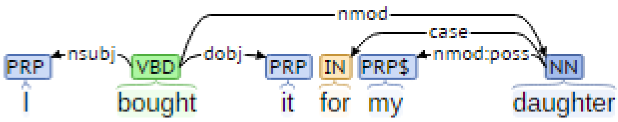


Figure 2. Dependency Relations – Example 2

In Figure 1, the dependency relation, ***amod (Adjectival Modifier),*** correctly connects noun (NN), *lens*, with adjective (JJ), *good*, to extract the target aspect, *lens*. However, not all dependency relations are relevant in extracting aspects. For instance, in Figure 2, noun (NN), *daughter*, is related by the dependency relation, ***nmod (Nominal Modifier),*** with the verb (VB) *bought*, but notice that although, *daughter* is a noun, it is not a valid aspect. This example demonstrates how the application of irrelevant dependency can lead to erroneous extractions of aspects, which invariably will have a detrimental effect on recommendation performance. Here the use of sentiment knowledge would have shown that *daughter* is not a valid aspect as it is not related to a sentiment-bearing verb. Similarly, frequency information may also have conveyed that *daughter* to be an infrequent noun in camera reviews. This calls for informed selection of dependency relations and heuristic rules to avoid spurious aspect extraction. In this paper, we propose an informed aspect extraction approach that combines the strengths of both dependency relations and rule-based frequent noun approaches. Further, we show how sentiment analysis can be used to create a computational model of aspect-level user preferences.

The rest of the paper is organized as follows: in section 2 we present the background research related to this work. Next in Section 3 we describe the process of social recommender systems. Our proposed

[1] *http://universaldependencies.org/en/dep/index.html*

aspect extraction algorithm is presented in Section 4. Further, we present our recommendation strategy in Section 5. Finally, evaluation results are presented in Section 6 followed by conclusions in Section 7.

Related Work

Social Recommender Systems

The increasing popularity of social media allows people to share their opinions, interact with other users and enrich people's social activities with their families and friends (e.g. Twitter, Facebook etc.). The growth of social recommender systems aims to leverage social knowledge with one primary objective, which is to improve the quality of predictions and recommendations.

Product reviews and social tags are a form of user feedback that can be gathered from e-commerce websites or any other social media platform such as Twitter and YouTube. Such resources are often more accurate in representing user preference (Zhuang et al. 2006). In e-commerce websites, users write product reviews and states their opinions on different aspects of purchased products. Therefore, it contains explicit knowledge that presents new opportunities for product recommendation. Chen & Wang (2013) apply Latent Class Regression models (LCRM) to consider both the overall ratings given by a user and aspect sentiment values to identify reviewers' preference. Yates et al. (2008) combine opinions on technical aspects from product manuals (e.g. physical dimensions) and aspects from reviews to construct product profiles that indicate the average user preferences. In contrast, Dong et al. (2016) evaluate a list of products based on the sentiment score for every aspect of a product. For a given query case, a set of products is retrieved based on *k* shared aspects between the query case (q) and candidate product case (c). However, their results show that when recommendation is solely based on sentiment scores of a product the recommended products are less similar to the query products. In our work, we recommend products that are similar and better than the query product.

Aspect Extraction

Social recommender systems that analyse product reviews for recommendation adapt works from the domain of sentiment analysis and aspect-oriented opinion mining. Recent approaches using deep neural network show performance improvement on standard datasets. Poria et al. (2016) combined convolutional neural network and linguistic patterns for aspect extraction. Similarly, Liu et al. (2015a) applied a recurrent neural network with pre-trained word embeddings to extract aspects. Whilst the aforementioned methods require linguistic features as input to the deep neural network, Wang et al. (2017) proposed a multi-layer attention network to capture the relations between words in a sentence instead of applying dependency relations to co-extract aspects and opinion terms. Note that the deep neural network approach is supervised learning and the approaches are evaluated on an annotated dataset. One of the major limitations of supervised approaches is the difficulty of obtaining annotated training data from social media text. Therefore, recommendation algorithms that leverage real-world datasets favours unsupervised approach over supervised approach. Specifically, unsupervised approaches do not require human labelled data to learn to extract genuine aspects. Therefore, in this work we advocate the use of unsupervised approaches in extracting product aspects.

Frequent Noun Approach

Prior research indicates that product aspects are generally nouns and compound nouns (Hu & Liu 2004). Therefore, the most common unsupervised approach in the current literature of review based recommendation systems involves the use of frequent nouns to identify potential aspects (Dong & Smyth 2016; Liu et al. 2013; Levi et al. 2012). In particular, the approach proposed by Hu & Liu (2004) use association mining to identify frequent nouns or noun phrases. Thereafter sentences are grouped by these aspects and sentiment scores are assigned to each aspect group. Popescu and Etzioni (2007) first extract nouns and noun phrases list from product reviews and prune the list with a frequency threshold. The remaining candidate aspects in the list are evaluated using Pairwise Mutual Information (PMI) between the candidate aspect and associated discriminator. Similarly, Bafna and Toshniwal (2013) determine genuine aspects by the frequency of their occurrence in reviews and their probability occurrence in a product specific corpus compared to a generic corpus. Moghaddam and Ester (2010) and Htay and Lynn (2013) use opinion pattern extraction approach to identify product aspects. In particular, Moghaddam and Ester (2010) extended the noun frequency approach by adding an opinion pattern to identify product aspects. Frequent noun approach, though effective but suffer from generating many incorrect aspects and removing less frequent aspects.

Dependency Relations Approach

Whilst there are many others statistical approaches to frequent noun extraction in aspect based opinion mining literature (Blair-Goldensohn & K. Hannan 2008); others argue that identifying aspects by means of dependency relation rules helps to identify low frequency aspects (Schouten & Frasincar 2016). This is because dependency relation rules are able to determine the semantic relationships between words. Zhuang et al. (2006) utilize dependency relations to find the aspect-sentiment pairs. However, they only apply dependency relation rules that frequently occur and the less frequent ones are removed. Moghaddam & Ester (2012) defined nine dependency relation rules to extract candidate phrases. Their work is different in a way that the syntactic dependencies that are applied in a pre-processing task to categorize the aspects using a topic model approach.

Liu et al. (2015b) proposed a supervised approach to select a subset of rules in extracting aspects so that the rules are not selected manually to reduce errors. However, there was no significant improvement observed. In some cases, only recall is improved. Qiu et al. (2011) proposed a propagation method to find all possible aspects and sentiments. During the searching process, sentiment words are considered to be adjectives and aspects are nouns or noun phrases. The key idea here is that with each known sentiment, more aspects can be found and vice versa. However, this approach could extract many nouns/noun phrases that are not aspects and therefore does not scale well with large datasets. This is because during propagation, adjectives that are not opinionated will be extracted as opinion words. Poria et al. (2014) proposed a rule-based approach that exploits common-sense knowledge and sentence dependency trees to extract aspects. Experimental results indicate that the rule-based approach outperform the bootstrapping method in Qiu et al. (2011).

Aspect Level Sentiment Analysis

Sentiment classification assigns a positive or negative label to opinionated documents, paragraph or sentences. Unlike classical sentiment classification, aspect sentiment classification aims to consider the aspect in a sentence during classification. Therefore, the approaches proposed for aspect sentiment classification tend to differ from the classical ones. A common approach in aspect sentiment classification is the lexicon-based approach. This approach determines the polarity (positive or negative) and strength of sentiment expressed at word-level (e.g. SentiWordNet (Esuli and Sebastiani, 2006)). Increasingly aggregation is organised at the aspect level, since different users express different level of sentiment to the same aspect. Therefore, sophisticated methods are needed to aggregate these scores at the sentence, paragraph and document level and account for negation and other forms of sentiment modifiers (Muhammad et al. 2016; Chen & Wang 2013).

Social Recommendation Process

An overview of the social recommendation process appears in Figure 3. The final outcome of the recommendation process is a list of recommended products that are ranked on the basis of a *ProductScore*, with respect to a given query product. Central to this ranking is the computational model of aspect level user preferences derived from aspect-level sentiment scoring. For each product, we applied dependency relation rules to extract product aspects from product reviews. Given the extracted aspects we identify the sentiment words from the reviews that describe the aspects to compute the aspect sentiment score using a sentiment classification system for social media text.

Each product in the corpus is represented in a vector space model (VSM). In this model, a product is represented as a vector in n-dimensional space where each dimension corresponds to a separate aspect. If an aspect occurs in the reviews of a product, its value in the vector is a non-zero value. Therefore a product p can be represented as follows:

$$p = [a_{k,} a_{k+1}, a_{k+2} \dots . a_n]$$

Here, a_k is the value for an aspect and n is the size of the vector. In this work, we use the sentiment score of an aspect as its value in the vector space model.

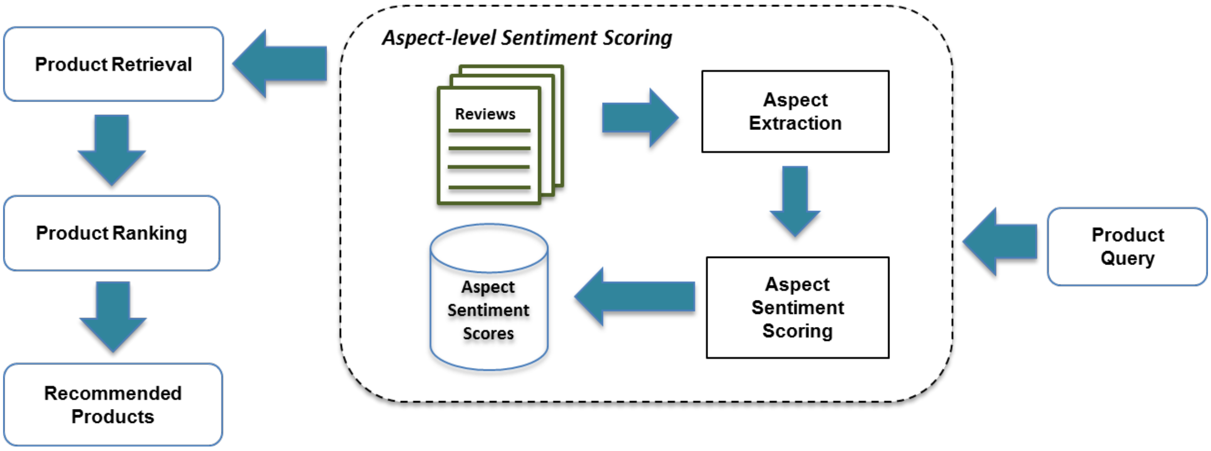


Figure 3. Social Recommendation Process

Dependency Rule-based Aspect Extraction

Selection of Dependency Relations

Dependency relations relate one word to another word in the same sentence. Each dependency relation has a head word and a modifier that modifies the head word. A head word is the word that determines the syntactic type of a phrase. A modifier is an optional element in a phrase which can be removed without affecting the grammar of the sentence. Both head and modifier can appear as a noun. Therefore, an aspect can appear as a head or a modifier in a dependency relation. Table 1 summarise a list of dependency relations from the previous works (Bancken et al., 2014, Zhuang et al., 2006) together with the extracted aspect and sentiment word. In each example, the word in bold is the extracted aspect word and the underlined word is the sentiment word. Here, we can see that each dependency relation relates words with different POS (Part-of-Speech) patterns. For instance, in the first example, ***nsubj***, connects the noun (NN), *sound*, with adjective (JJ), *clear*. Thus, ***nsubj*** connects words with POS pattern of (NN, JJ). Similarly, in the second example we can see that, ***nsubj***, connects words with POS patterns of (DT[2], NN. Based on these examples, we can summarise that the relation ***nsubj*** connects words in a sentence with POS patterns of {(NN, JJ), (DT, NN)}.

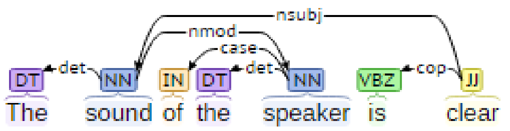
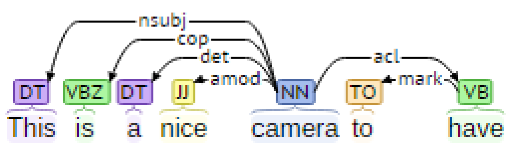
Dependency Relations	Definition	Example	Output (head, modifier)
Nominal subject (nsubj)	Nominal subject relation is a noun phrase which is the syntactic subject of a clause.		<u>clear</u>, **sound**
Adjectival modifier (amod)	Adjectival modifier of an NP is any adjectival phrase that serves to modify the meaning of the NP		**camera**, <u>nice</u>

Table 1. Examples of Dependency Relations

To validate the relevance of a dependency relation, we obtain the list of possible POS patterns that are generated by the dependency relation from an English corpus[1]. We use $patternPOS(dp)$ to retrieve the set of possible POS patterns for a given dependency relation dp where

[2] *DT - determiner*

$$patternPOS(dp) = \{(pos_{x_1}, pos_{y_1})_1, (pos_{x_2}, pos_{y_2})_2, \dots, (pos_{x_n}, pos_{y_n})_n\}$$

Here, pos_x and pos_y are the part-of-speech (POS) of the two words related by dp and $dp \in DP$. Based on the list of possible patterns, we formalise a function to retain relations that are relevant in extracting product aspects as follows:

$$RelevantDP = \{dp \in DP : isRelevant(dp)\}$$

$$isRelevant(dp) = \exists t \in Patterns : t \in patternPOS(dp)$$

$$Patterns = \{(N, N), (N, V), (V, N), (N, J), (J, N), (N, RB), (RB, N)\}$$

where $RelevantDP$ is a set of dp that satisfy the condition $isRelevant$. The condition $isRelevant$ is true if there exists a pattern t in a set of predefined patterns, $Patterns$, such as $patternPOS(dp)$ is true. Here, $Patterns$ is a set of POS patterns that is used to identify relevant dependency relations. In product reviews, there is a relationship between an aspect and the sentiment expressed on the aspect (Liu 2015). We consider aspects to be nouns and sentiment words to be adjectives, verbs and adverbs, which has been widely adopted in previous work (Popescu & Etzioni 2007; Hu & Liu 2004). Therefore, dependency relations that frequently relate nouns (N and N), noun and adjective (N and J), noun and verb (N and V) and nouns and adverb (N and RB) are relevant for aspect extraction. For example, "*picture quality*" is related by the relation, ***compound (Noun Compounds)***, and both terms are noun. Therefore, ***compound*** has a pattern of (N, N). Based on this condition, $isRelevant$ filters out 9 out of 47 dependency relations and leaves us with 38 dependencies for our experiments. We summarise the list of the selected dependency relations in Table 2.

Patterns	**Dependency Relations**
(N,J) (J,N)	acl, acl:relcl, advcl, amod, appos, advmod, ccomp, compound, conj, csubj, dep, det, discourse, dislocated, dobj, goeswith, list, mark, name, nmod:npmod, nmod:tmod, nsubj, nsubjpass, nummod, parataxis, remnant, vocative, xcomp
(N,V) (V,N)	cop, csubjpass, cc, case, iobj, reparandum
(N,RB) (RB,N)	cc:preconj, expl, neg
(N,N)	compound

Table 2. Selected Dependency Relations[3]

In our work, we use the Stanford CoreNLP[4] parser to generate the dependencies of the review text. The parser takes in a review sentence and produces a list of word pairs for each dependency relation. Let S be the set of sets of word pairs output from the parser where $S = \{s_1, s_2, \dots s_n\}$. The representation for each s is a set of pairs, $s = \{(w_{x_1}, w_{y_1})_1, (w_{x_2}, w_{y_2})_2 ,\dots.. (w_{x_n}, w_{y_n})_n\}$, each of which is composed of the words w_x and w_y from the sentence which are related by a selected dependency relation $sdp \in RelevantDP$. Here we consider noun terms that are related by selected dependency relation as potential aspects. This condition is implemented by the function $DirectRelations$ as follows:

$$DirectRelations_{sdp}(S) = \{a_1, a_2, \dots, a_n\}$$

Accordingly, the final output of $DirectRelations$ is a set of aspects, a, extracted from the selected dependency relations.

Rule-based Frequent Noun Approach

Potential aspects extracted from dependency relations are not all genuine aspects. Additional heuristic rules are required to extract meaningful aspects and filter incorrect aspects. Here, we describe the heuristics applied in our work to remove the incorrect aspects.

[3] *The full description on each dependency relations can be found at http://universaldependencies.org/en/dep/index.html*

[4] *http://stanfordnlp.github.io/CoreNLP/*

Pruning of technical specifications. In product reviews, users may express their opinions by describing technical details of an aspect. For example, "Sigma 18-250mm lens" is describing the size of the camera lens. During parsing, the size of the lens (18-250mm) is automatically parsed as noun. Therefore, technical details which extracted as potential aspects will be removed.

Global frequency pruning. Information presented in the product reviews is highly dynamic and written in an informal manner. Misspelled words and web address are automatically parsed as noun. We conduct frequency based pruning to retain aspects that occur greater than a specific threshold in the product reviews. We set the threshold to 2 which is commonly adopt in the literature (Hu and Liu, 2004, Qiu et al., 2011).

Co-occurrence of Sentiment Word. Not all nouns extracted from product reviews are aspects. One solution proposed by (Hu and Liu 2004) is by removing aspects that does not associate with sentiment words. Here, we adapt this approach to remove aspects that do not co-occur with sentiment word in the same sentence.

Generating Recommendation

Similarity Retrieval

In recommender systems, the retrieval set for a target query consists of n products that are most similar to the query product. Similarity of a candidate product (C) in a given retrieval set in terms of the target query product (Q) is measured using the standard cosine similarity metrics below:

$$Sim(Q,C) = \frac{\sum_{i=1}^{n} Q_i C_i}{\sqrt{\sum_{i=1}^{n}(Q_i)^2}\sqrt{\sum_{i=1}^{n}(C_i)^2}}$$

Here Q_i and C_i is the weight of the ith aspect in product Q and C respectively which are computed using the common term weighting scheme, TF-IDF (Term Frequency-Inverse Document Frequency) as follows:

$$TFIDF(a,Q,P) = tf(a,Q) \times idf(a,P)$$

where P denotes the set of products in the corpus and a is an aspect in Q. The term frequency, $tf(a,Q)$, and inverse document frequency, $idf(a,C)$, are given as follows:

$$tf(a,Q) = 1 + \log(f_{a,Q})$$

$$idf(a,P) = \log(\frac{|P|}{|p \in P : a \in p|})$$

Here, $f_{a,Q}$ is the frequency of occurrence of aspect a in Q. Term frequency considered all aspects are equally important. However, aspect such as “machine” may frequently occur in the reviews of laptop products but have little importance. Therefore, the frequent aspect is offset by the frequency of the aspect in the entire corpus using idf. The idf of aspect a is obtained by dividing the total number of products by the number of products that contain a and then taking the logarithm of the division.

Aspect Sentiment Scoring

The process of classifying the polarity of the sentiment expressed for an aspect starts with a set of aspects extracted using our proposed approach. Aspects are known to be nouns and adjectives are important indicators of sentiment words (Hu and Liu, 2004, Popescu and Etzioni, 2007). Therefore, we find the nearest adjective word to an aspect in the same sentence as the target sentiment word.

The sentiment score of an aspect is generated by a sentiment analysis tool that capture contextual polarity called SmartSA (Muhammad et al., 2016) which obtains the sentiment score of sentiment bearing words from SentiWordNet (Esuli and Sebastiani, 2006). To determine the polarity of the sentiment, we begin by pre-processing the review text using standard text pre-processing steps: tokenization, POS tagging and lemmatisation. We did not convert the tokens into consistent case to preserve the sentiment expression that express through capitalisation. In addition, stop words filtering is not necessary since stop-words are typically not included in a lexicon or are associated with zero values in SentiWordNet and thus cannot influence polarity classification. SmartSA modifies sentiment score obtained from SentiWordNet to take into consideration of negation terms and lexical valence shifters (e.g. intensifier and diminishing terms) that can change sentiment orientation. Ideally, we can use the dependency relation **neg** (**Negation Modifier -** a dependency relation that relates between a negation term and the word it modifies) to identify negation terms. However, the non-standard writing style of users in social media text causes the Stanford CoreNLP parser unable to produce satisfactory results. For instance, the Stanford CoreNLP parser is unable to generate **neg** relations with the omission of apostrophe in the sentence "I dont like the screen of the camera". One solution to solve this problem is to adopt a window based approach. This is because modifiers such as negation terms and valence shifters are assumed to affect terms within a specific text window (Thelwall et al., 2012). Therefore, in order to capitalize on the contextual analysis offered by SmartSA, we adopt a window-based approach to extract a window of words pivoted on the target sentiment word as a document presented to the tool for sentiment scoring.

The resulting scores from SmartSA, $SentiScore$, determine the final orientation of the sentiment on each aspect for a product. At the product level, sentiment scores for each unique aspect are aggregated using an arithmetic mean. Formally, $ProductScore$ of a product is computed as follows:

$$ProductScore(p_i, a_j) = \frac{\sum_{j=1}^{|A^i|} AspectSentiScore(p_i, a_j)}{|A^i|}$$

$$AspectSentiScore(p_i, a_j) = \frac{\sum_{m=1}^{|R_j^i|} SentiScore(r_m)}{|R_j^i|}$$

Where R_j^i is a set of reviews for product p_i related to aspect a_j and $r_m \in R_j^i$. Here, $AspectSentiScore$ allows the sentiment of product, p_i, to be associated with individual aspects $a_j \in A^i$. Here, A^i is the subset of aspects that are shared between the query and candidate product.

Evaluation

In this section, we empirically evaluate our proposed aspect extraction algorithm applied to a ranking model for product recommendations. For this purpose, we collected data from Amazon during April 2014 and November 2014. In particular, we collected data from five different product categories: DSLR cameras, Laptops, Phones, Printers and TV. This data includes information about the product (product id, product name, price etc.), their reviews and overall user ratings. Since we are not focusing on the cold-start problem, products with less than 10 reviews are removed. Table 3 shows the descriptive statistics of the five datasets used in the experiments.

Descriptions	**DSLR**	**Laptops**	**Phones**	**Printers**	**Tv**
No. of products	56	116	51	82	52
No. of reviews	6206	3734	2595	11,442	5860

Table 3. Dataset Statistics

During recommendation, we choose the retrieval size (N) 15 for every query product. This is because in practice, the retrieval set size is limited by the number of products available in the dataset. For example, in DSLR dataset, there are 56 products available. After splitting the dataset into training and test set we have 35 products for training and 17 products for testing. Therefore, the number of products retrieved is never more than 16 (the last product is the target query product).

Compared Methods

We compare the recommendation performances that apply our proposed aspect extraction approach DirectRelations+ with five baseline methods: FREQ, MogDP, SenticNetDR and DirectRelations.

- **FREQ**: a combination of shallow NLP and statistical methods to extract aspects (Dong et al., 2015, Hu & Liu, 2004, Justeson & Katz, 1995). This baseline returns the most frequent nouns and noun phrases of the reviews in each dataset. The frequency is calculated based on how frequently the aspect co-occurs with a sentiment word in the same sentence. Nouns that have a frequency greater than some fixed threshold are retained. Here, we set threshold to 30% as implemented in previous work (Dong et al., 2015).
- **MogDP**: a set of nine dependency relation rules to extract aspects (Moghaddam & Ester 2012). The list of dependency relations that are applied in the rules are *amod, acomp, nsubj, cop, dobj, compound, conj* and *neg*.
- **SenticNetDR:** a rule-based approach that exploits common-sense knowledge and dependency relations to extract aspects (Poria et al. 2014). The list of dependency relations applied in this approach includes *advmod, amod, advcl, xcomp, cop, cc, conj, dobj, ccomp* and *compound*.
- **DirectRelations:** implements the proposed approach without frequency filtering.
- **DirectRelations+:** implements the proposed approach with the rule-based frequent noun approach.

Evaluation

To validate the ranking model, we use overall user ratings as the measure of product quality. Using the standard leave-one-out methodology, we quantify the effectiveness of our ranking model using two evaluation metrics and report statistical significance using paired t-Test at 95% confidence level. We chose Mean Average Precision and Rank Improvement because they have been applied to measure rankings of products in previous work (Dong et al. 2013; Zhang et al. 2011).

- **MAP (Mean Average Precision):** Measures average precision across multiple queries. The aim of MAP is to evaluate the recommendation performance by considering the rank of the top products in the recommended list such that the higher the top products are ranked, the higher the MAP value. To evaluate our proposed approach, we produce test cases in the form of $(p_q, Superior)$ as ground-truth where p_q is a query product and $Superior$ is the corresponding top 3 candidate products that are similar to p_q and have a higher overall user ratings than p_q. The MAP is defined as follows:

$$MAP@N = \frac{1}{N}\sum_{j=1}^{N}\frac{1}{Q_j}\sum_{k=1}^{Q_j} Precision(k)$$

  where Q_j is a set of similar products for query j, N is the number of queries and $Precision(k)$ is precision at kth similar product in the retrieval set.

- **RI (Rank Improvement):** The average gain in rank position of recommended products over the left-out query product is computed relative to a benchmark product ranking. Specifically, for each query product we generate the top-n $(n = 3)$ recommendations using the proposed approach. Then, we rank all recommended products by users' rating and then calculate the quality of a recommended product in terms of its percentage rank difference to the query product. This approach estimates the degree in which the recommended product is superior to the query product. Here, we generate the benchmark ranking according to the overall user ratings of products.

$$RI\% = \frac{\sum_{i=1}^{n} benchmark(p_q) - benchmark(p_i)}{n * |P-1|}$$

Here, *benchmark* returns the position of a recommended product p_i on the benchmark product ranking. The greater the gain of the recommended product over the query product (p_q) the better. Suppose the query product is ranked 40th on the reference ranking of 81 unique products, and the recommended product is ranked 20th on the reference ranking list, then the recommended product will have a relative rank improvement of 25%.

Results and Discussions

Table 4 and 5 show the results in MAP@15 and RI% respectively of the approaches FREQ, MogDP, SenticNetDR, DirectRelations and DirectRelations+ evaluated on five datasets. The asterisks (*) in the tables indicate the significance of the result compared to the state-of-the-art methods (FREQ, MogDP, SenticNetDR). Numbers in bold indicate that the proposed approach achieved best results over other approaches. In Table 4, results on all datasets show that DirectRelations+ achieved significant improvement over all the baseline approaches. Specifically, the results demonstrate that combining rule-based frequent noun approach with for informed selection of dependency relations achieves 15% improvement on average. Similar observations can be made for RI% in Table 5 where DirectRelations+ performs best in all datasets. Specifically, our proposed approach recommends products with a relative rank improvement of between 7.0% and 27.7%. Given the retrieval set of 15 products, this means that DirectRelations+ is recommending products that are, on average, up to 4 rank positions better than the query product in terms of overall user rating.

In Table 5, the RI% for baseline approaches, in most cases, is less than 7%. Since recommending a product with one rank position better than the query product will result in 7% rank improvement, RI less than 7% suggests that the baseline approaches recommend products that rank below the query product in most test cases. This observation further emphasis the effectiveness of combining rule-based frequent noun approach with informed selection of dependency relations in aspect extraction to improve recommendation performance. FREQ is observed to perform better than other baseline approaches. However, the comparison between DirectRelations+ and FREQ indicates that the former is superior to the latter in both MAP and RI%. This suggests that filtering technical specifications and the use of sentiment knowledge to filter aspects is crucial in extracting meaningful aspects for product recommendation.

Methods	DSLR	Laptops	Phones	Printers	Tv
FREQ	0.630	0.401	0.596	0.351	0.471
MogDP	0.657	0.401	0.514	0.344	0.506
SenticNetDP	0.638	0.404	0.584	0.312	0.485
DirectRelations	0.635	0.422	0.594	0.308	0.513
DirectRelations+	**0.740***	**0.497***	**0.644***	**0.397***	**0.566***

Table 4. Results for MAP@15

Methods	DSLR	Laptops	Phones	Printers	Tv
FREQ	21.7	2.3	8.4	2.3	3.4
MogDP	18.9	5.5	3.0	3.4	2.3
SenticNetDP	15.1	4.0	8.3	3.9	2.7
DirectRelations	16.8	12.4	7.5	1.1	4.9
DirectRelations+	**27.7***	**17.9***	**10.2***	**7.3**	**7.0***

Table 5. Results for RI%

One of the important observations from the results is that, recommendation performance does not benefit from approaches that produce high coverage of aspects. In Figure 5, FREQ, SenticNetDP and DirectRelations extract a large number of aspects (between 5395 to 23,167 aspects). However, these approaches did not perform well compared to DirectRelations+ which have, on average, 2837 aspects. Table 6 shows the top 10 most frequent aspects in DSLR. Note that the list of most frequent aspects extracted by FREQ, SenticNetDP and DirectRelations contains more incorrect aspects than MogDP and DirectRelations+. This suggests that extracting a large number of aspects increases the opportunity of having erroneous aspects. Further, we observed that MogDP extracts the lowest number of aspects with an average of 970 aspects across all products. The poor performance of MogDP suggests that additional dependency relations are required to extract aspects that improve recommendation performance.

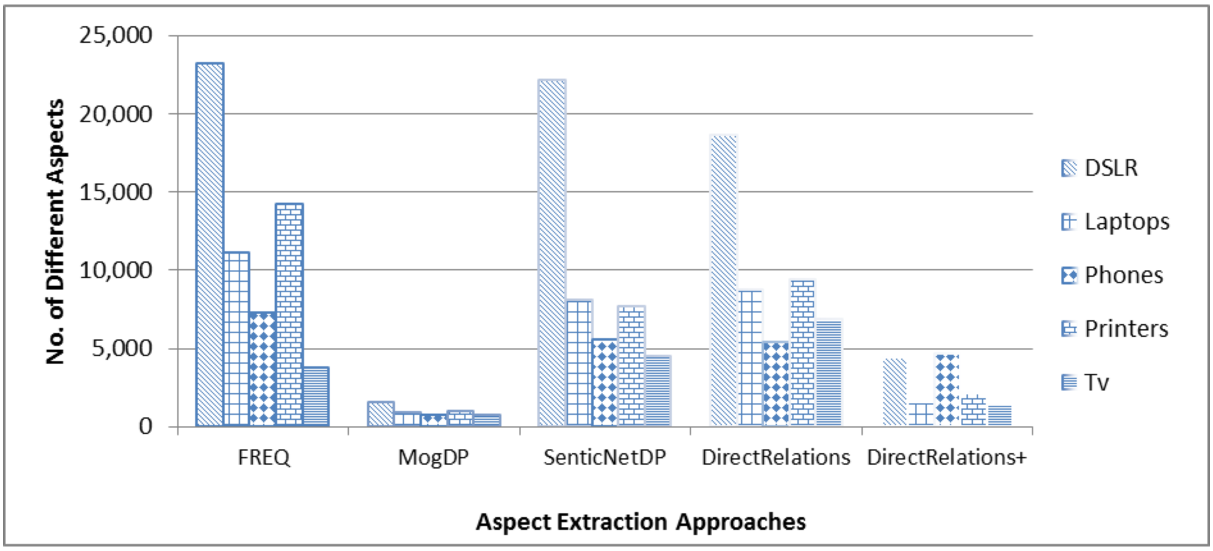


Figure 5. Number of Different Aspects Extracted by Different Approaches

Methods	**Aspects**
FREQ	camera[5], lens, canon, nikon, video, time, quality, ones, lot, point
MogDP	camera, lens, picture, quality, feature, shot, photo, price, time, dslr
SenticNetDP	camera, purchase, nikon, price, lens, picture, work, use, love, lot
DirectRelations	camera, purchase, lens, price, canon, time, picture, hand, work, use
DirectRelations⁺	camera, lens, picture, feature, quality, shot, dslr, price, lightning, focus

Table 6. Top 10 Most Frequent Aspects for DSLR (genuine aspects are underlined)

Conclusion and Future Work

In this paper, we proposed an informed aspect extraction approach for recommender systems. We have demonstrated the benefit of this approach in a realistic recommendation setting using benchmarks generated from real user ratings. The results demonstrated that combining for informed selection of dependency relations and a rule-based frequent noun approach significantly improves state-of-the-art dependency-based methods in a recommendation setting. In particular, we confirmed that filtering technical specifications and using sentiment knowledge to filter aspects is crucial in extracting aspects. Our results also show that recommendation performance benefits from applying a larger set of dependency relations rather than a small set of dependency relations.

Our proposed aspect extraction approach extracted, on average more than 2000 aspects. Typically all extracted aspects from reviews are used in recommendation. However, it is not realistic to assume that all aspects are equally important to users when making a purchase decision. Furthermore, there are cases where ordinary nouns that are not aspects but are being extracted due to parsing errors. Therefore, the list of aspects extracted is not all relevant in recommendation. In our future work, we plan to develop methods to identify a subset of aspects that are important to users when making a purchase decision.

References

Bafna, K. and Toshniwal, D. 2013. “Feature based Summarization of Customers' Reviews of Online Products,” *Procedia Computer Science*, 22, pp. 142-151.

[5] *Camera is an aspect commonly used in product reviews to express opinions for the camera as a whole.*

Bancken, W., Alfarone, D., and Davis, J.Bancken, W., Alfarone, D., Davis, J. 2014. "Automatically Detecting and Rating Product Aspects from Textual Customer Reviews," in *1st Inter. Workshop on Interactions between Data Mining and Natural Language Processing at ECML/PKDD*, pp. 1–16.

Blair-Goldensohn, S., Hannan, K., McDonald, R., Neylon, T., Reis, G., Reynar, J. 2008. "Building a Sentiment Summarizer for Local Service Reviews," in *WWW Workshop on NLP in the Information Explosion Era*, p. 14.

Chen, L. and Wang, F. 2013. "Preference-based Clustering Reviews for Augmenting E-commerce Recommendation," *Knowledge-Based Systems*, 50, pp. 44–59.

Dong, R., Schaal, M., OMahony, M., McCarthy, K., Smyth, B. 2013. "Opinionated Product Recommendation," in *Inter. Conf. on Case-Based Reasoning*, pp. 44-58.

Dong, R., & Smyth, B. 2016. "Personalized Opinion-Based Recommendation," in *International Conference on Case-Based Reasoning*, pp. 93-107, Springer International Publishing.

Esuli, A., Sebastiani, F. 2006. "Sentiwordnet: A Publicly Available Lexical Resource for Opinion Mining," in *Proc. Language Resources and Evaluation Conference*, pp. 417–422.

Hu, M. and Liu, B. 2004. "Mining and Summarising Customer Reviews," in *Proc. Of ACM SIGKDD Inter. Conf. on Knowledge Discovery and Data Mining*, KDD '04, pp. 168-177.

Koren, Y., Bell, R., and Volinsky, C. 2009. "Matrix Factorization Techniques for Recommender Systems," *Computer*, 8, pp. 30-37.

Levi, A., Mokryn, O., Diot, C., & Taft, N. 2012. "Finding a Needle in a Haystack of Reviews: Cold Start Context-based Hotel Recommender System," in *Proc. of the 6th ACM Conf. on Recommender Systems*, pp. 115-122, ACM.

Liu, P., Joty, S.R. and Meng, H.M., 2015a. Fine-grained Opinion Mining with Recurrent Neural Networks and Word Embeddings. In EMNLP (pp. 1433-1443).

Liu, Q., Gao, Z., Liu, B., & Zhang, Y. 2015b. "Automated Rule Selection for Aspect Extraction in Opinion Mining," in *IJCAI*, pp. 1291-1297.

Liu, B. 2015. *Sentiment analysis: Mining Opinions, Sentiments, and Emotions*, Cambridge University Press.

Lops, P., De Gemmis, M., and Semeraro, G. 2011. "Content-based Recommender Systems: State of the Art and Trends," in *Recommender systems handbook*, pp.73-105. Springer.

Moghaddam, S., Ester, M. 2010. "Opinion Digger: An Unsupervised Opinion Miner from Unstructured Product Reviews," in *Inter. Conf. on Information and Knowledge Management*, pp. 1825-1828.

Moghaddam, S., Ester, M. 2012. "On the Design of Lda Models for Aspect-based Opinion Mining," in "Proc. Inter. Conf. on Information and Knowledge Management", CIKM '12, ACM, pp. 803-812.

Muhammad, A., Wiratunga, N., and Lothian, R. 2016. "Contextual Sentiment Analysis for Social Media Genres," *Knowledge-Based Systems*, 108, pp. 92-101.

Popescu, A., Etzioni, O. 2007. "Extracting Product Features and Opinions from Reviews," in *Natural Language Processing and Text Mining*, pp. 9–28.

Poria, S., Cambria, E. and Gelbukh, A., 2016. "Aspect extraction for opinion mining with a deep convolutional neural network," *Knowledge-Based Systems*, 108, pp.42-49.

Poria, S., Cambria, E., Ku, L.-W., Gui, C., and Gelbukh, A. 2014. "A Rule-based Approach to Aspect Extraction from Product Reviews," in *2nd Workshop on NLP for Social Media*, pp. 28-37.

Qiu, G., Liu, B., Bu, J., and Chen, C. 2011. "Opinion Word Expansion and Target Extraction through Double Propagation," *Computational linguistics*, 37(1):9-27.

Sarwar, B., Karypis, G., Konstan, J., and Riedl, J. 2001. "Item-based Collaborative Filtering Recommendation Algorithms," in *10th Inter. Conf. on World Wide Web*, pages 285-295. ACM.

Schouten, K., & Frasincar, F. 2016. "Survey on Aspect-level Sentiment Analysis," *IEEE Transactions on Knowledge and Data Engineering*, 28(3), 813-830.

Thelwall, M., Buckley, K., and Paltoglou, G. 2012. "Sentiment Strength Detection for the Social Web," *Journal of the American Society for Information Science and Technology*, 63(1):163-173.

Wang, W., Pan, S.J., Dahlmeier, D. and Xiao, X., 2017. "Coupled Multi-layer Attentions for Co-extraction of Aspect and Opinion Terms," in *AAAI*, 2017.

Yates, A., Joseph, J., Popescu, A.-M., Cohn, A. D., and Sillick, N. 2008. "Shopsmart: Product Recommendations through Technical Specifications and User Reviews," in *17th ACM Conference on Information and Knowledge Management*, pp. 1501-1502. ACM.

Yin, Y., Wei, F., Dong, L., Xu, K. Xu, Zahng, M., and Zhou, M. 2016. "Unsupervised word and dependency path embeddings for aspect term extraction," in Proc. of 25th Inter. Joint Conf. on Artificial Intelligence *(IJCAI'16), AAAI Press 2979-2985.*

Zhang, K., Cheng, Y., Liao, W. K., & Choudhary, A. 2011. "Mining Millions of Reviews: A Technique to Rank Products based on Importance of Reviews," in *13th Inter. Conf. on E-Commerce* (p. 12). ACM.

Zhuang, L., Jing, F., Zhu, X.-Y. 2006. "Movie Review Mining and Summarization," in *Proc. of the 15th ACM International Conference on Information and Knowledge Management*, ACM, pp. 43-50.