CHEN, Y.Y., WIRATUNGA, N. and LOTHIAN, R. 2019. Integrating selection-based aspect sentiment and preference knowledge for social recommender systems. *Online information review* [online], EarlyCite. Available from: https://doi.org/10.1108/OIR-02-2017-0066.

Integrating selection-based aspect sentiment and preference knowledge for social recommender systems.

CHEN, Y.Y., WIRATUNGA, N., LOTHIAN, R.

2019

Online Information Review

Integrating Selection-based Aspect Sentiment and Preference Knowledge for Social Recommender Systems

Journal:	*Online Information Review*
Manuscript ID	OIR-02-2017-0066.R3
Manuscript Type:	Research Paper
Keywords:	social recommender systems, aspect-level sentiment analysis, preference graph, feature selection

Abstract

Purpose - Recommender system approaches such as collaborative and content-based filtering rely on user ratings and product descriptions to recommend products. More recently, recommender system research has focused on exploiting knowledge from user-generated content such as product reviews to enhance recommendation performance. In this work, we show that the performance of a recommender system can be enhanced by integrating explicit knowledge extracted from product reviews with implicit knowledge extracted from analysis of consumer's purchase behaviour.

Design/methodology/approach – We introduce a sentiment and preference-guided strategy for product recommendation by integrating not only explicit, user-generated and sentiment-rich content but also implicit knowledge gleaned from users' product purchase preferences. Integration of both of these knowledge sources helps to model sentiment over a set of product aspects. We show how established dimensionality reduction and feature weighting approaches from text classification can be adopted to weight and select an optimal subset of aspects for recommendation tasks. We compare our proposed approach against several baseline methods as well as the state-of-the-art *Better* method, which recommends products that are superior to a query product.

Findings - Evaluation results from seven different product categories show that aspect weighting and selection significantly improves state-of-the-art recommendation approaches.

Research limitations/implications – The proposed approach recommends products by analysing user sentiment on product aspects. Therefore, the proposed approach can be used to develop recommender systems that can explain to users why a product is recommended. This is achieved by presenting an analysis of sentiment distribution over individual aspects that describe a given product.

Originality/value – This paper describes a novel approach to integrate consumer purchase behaviour analysis and aspect-level sentiment analysis to enhance recommendation. In particular, we introduce the idea of aspect weighting and selection to help users identify better products. Furthermore, we demonstrate the practical benefits of this approach on a variety of product categories and compare our approach with the current state-of-the-art approaches.

1. Introduction

Traditional recommendation techniques employ user ratings to infer user preferences. The most common approach is collaborative filtering (CF) (Sarwar et al. 2001; Koren et al. 2009) where ratings of an existing user community with similar preferences to the target user drive recommendation judgements. However, CF models are plagued with cold-start and data sparsity problems (Esparza et al. 2011). To overcome these limitations, content-based approaches exploit product descriptions to build product profiles which identify products that are of particular interest to users (Pazzani and Bilsus, 2007; Lops et al. 2011). The key to accurate recommendation in content-based approaches is to have the right product representation. The standard approach in content-based approaches is to use a set of relevant keywords that appear in the product descriptions and leverage a keyword-based method to identify similar products for recommendation. However, these techniques fail to consider users' purchase experiences and preferences which are key to their purchase decisions. To overcome these weaknesses, additional sources of information are integrated into recommender systems.

Social information, like product reviews, contains user opinions about different aspects of a product. Consider the following review example:

"This is an excellent camera which produces great images and with great resolutions."

Here, the reviewer expresses a positive opinion on two aspects of a camera - "*images*" and "*resolutions*". Such fine-grained opinions are important in that they explain the consumer's preferences that drive their purchase decisions and should naturally influence the workings of recommender systems. However, reliance on user-generated reviews for product representation has two limitations:

- Social media text is characterised by a diverse vocabulary. A product may have hundreds of aspects which are not of equal importance to consumers when making a purchase decision (Zha et al. 2014). Therefore, methods to infer aspect importance are needed.
- Natural language processing (NLP) based product aspect extraction techniques that rely on Part-Of-Speech (POS) tagging and syntactic parsing are known to be less robust when applied to informal text (Owoputi et al. 2013). As a result, it is not unusual to have large numbers of spurious content incorrectly extracted as aspects. However, previous work ignores the selection of aspects and thus limits the potential of using reviews for recommendation.

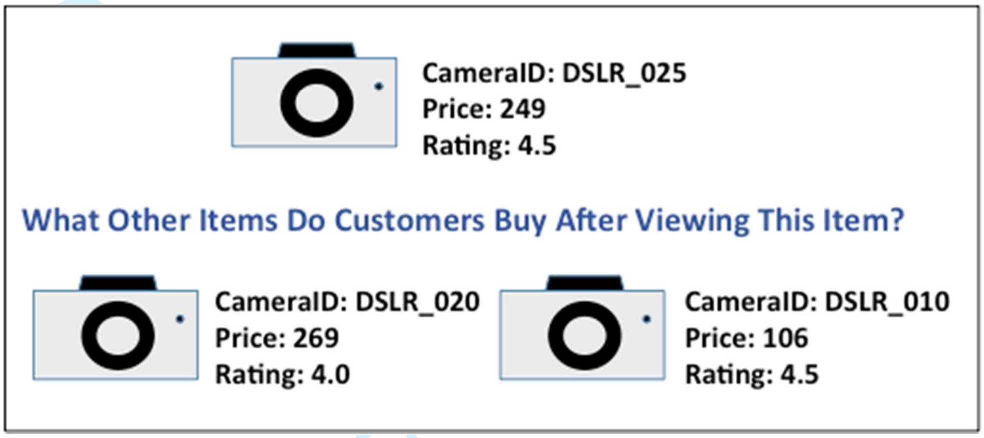


Figure 1: Product Information

To address the first limitation, we exploit consumer purchase behaviour analysis to improve recommendations. Figure 1 shows that in addition to typical information about camera DSLR_025 (e.g. price and rating), there is also information about user preferences (e.g. what users typically buy after viewing this camera). We observe that DSLR_020 and DSLR_010 are products that many users purchased after viewing DSLR_025. Based on this information, we generate two preference relations in which DSLR_020 is preferred over DSLR_025 and DSLR_010 is preferred over DSLR_025. The list of purchased products provides valuable insights about the preference of users. Therefore, the preference relations on products can be used to model aspect importance. This is because purchase choices are based on comparison of products; which involves comparison of aspects of these products. In particular, a user's purchase preferences hint at aspects that are likely to have influenced their purchase decisions and as such the aspects are deemed more important. We capture all preference relations between products using a preference graph and analyse this structure to infer the importance of aspects.

Feature selection is known to enhance accuracy in text classification by identifying redundant and irrelevant features (Yang and Pedersen, 1997). Therefore, we address the second limitation by proposing to adopt feature selection heuristics in aspect selection.

The main contributions of this paper are as follows:

1. Introduce an effective aspect weighting algorithm based on knowledge gathered from product reviews and consumer purchase behaviour analysis.
2. Extend the aspect weighting algorithm to adopt feature selection techniques from text classification.
3. Formulate a preference-based product ranking model that combines contributions 1 and 2 for recommendation.

The rest of the paper is organised as follows: in Section 2 we present the background research related to this work. In Section 3, we present the process of aspect-level sentiment analysis and aspect selection. Thereafter, we describe the process of aspect weight learning based on the knowledge gathered from product reviews and consumer purchase behaviour analysis for recommendation. Our evaluation results are presented in Section 4 followed by conclusions in Section 5.

2. Related Works

In the following sections, we explore how product reviews are used for recommendation and examine the state-of-the-art techniques in aspect-level sentiment analysis.

2.1 Social Recommender Systems

Social recommender systems aim to utilise social media in recommendation (Guy, 2015). The rich information embedded in product reviews allows social recommender systems to assess the quality of a product based on users' experiences, and elicit users' preferences from their written reviews and ratings. Chen and Wang (2013) applied Latent Class Regression models (LCRM) to consider both the overall ratings given by a user and aspect sentiment values to identify reviewers' preferences. Sun et al. (2015) mined affective text from users' comments using an ensemble learning-based method to recommend social media items. Dong et al. (2016) formulated a *Better* score to rank a list of products based on the sentiment score for every aspect of a product. Recommended products are retrieved and ranked based on the similarity and sentiments of the query product's aspects and the candidate product. However, their results show that when the recommendation is solely based on sentiment scores of a product the recommended products are less similar to the query product (Dong et al. 2016). This implies that the products recommended might be very different from what the user requires. Therefore, the recommendation strategy needs to be improved in such a way that priority is given to products that matches user's requirements.

A key issue with any recommendation technique is that neither user ratings nor product description are available in sufficient quantity. Implicit feedback aims to avoid this bottleneck by inferring user preferences from their interaction patterns. For instance, Kim et al. (2009) used dynamic expert group opinions to recommend domain specific web documents to users. The members in expert groups are adjusted according to the users' feedback. Therefore, the performance of the recommender system is highly dependent on expert groups' opinions. Similarly, in the tourism domain Christensen and Schiaffino (2015) used user profiles and social relations among the members registered in the system to generate recommendations to individual users and groups. The intuition in using this social relation is that two socially connected users are more likely to share similar interests. However, the relationships between users in the network change quickly over time which limits the recommender system in capturing recent user interests.

Implicit feedback is also based on observable user interactions with the system. In restaurant recommendation, Vasuden and Chakraborti (2014) estimated the utility of a restaurant by mining users' trails from a restaurant recommendation system. A user trail is a path that the user follows when searching for a product of interest. The path started from a restaurant as an entry point, users receive recommendations and they critique them (e.g., cheaper, creative…) to look for other restaurants that suit their preferences. This cycle continues until the user stops the search. An interesting observation from this work is that the users' trails were modelled as a preference graph to estimate the relative utilities of restaurants. In this paper, we infer user preferences from a preference graph by comparing the sentiment-rich user-generated content unique to the subgraph of interest.

Chen et al. (2014) estimated the utility of a product by combining product popularity and users' sentiment feedback of a product. Specifically, an aspect weighting algorithm using sentiment scores of aspects and view-purchased product pairs was proposed. However, their results showed that better performance is observed when the recommendation is ranked using PageRank algorithm (Page et al. 1999). One limitation observed in this approach is that it does not take into account that reviewers who voted strongly in favour of an aspect might overpower opinions of others. Therefore, sentiment scores are not a useful measure in estimation of importance of an aspect. In this work, we overcome this limitation by computing aspect weights using the polarity of a sentiment score (positive or negative) for each view-purchased product pair and evaluate our proposed approach using datasets from different product categories.

Social recommender systems that analyse product reviews for recommendation generally employ methods from aspect-level sentiment analysis (Dong et al. 2016; Liu et al. 2013; Levi et al. 2012) to extract product aspects and users' sentiment from product reviews. NLP-based aspect extraction techniques that rely on Part-of-Speech (POS) tagging and syntactic parsing are less robust when applied to informal text (Owoputi et al. 2013). As a result, a large number of spurious aspects are extracted. However, the effect of spurious content in product representation on recommender system performance was not discussed in the existing work. In this paper, we propose to integrate an aspect selection module in our recommendation model and evaluate its significance in improving recommendation performance.

Inspired by text classification research where feature selection is used successfully for dimensionality reduction, we explore the transferability of feature selection heuristics for aspect selection. Feature selection methods can be categorised into supervised and unsupervised algorithms. Supervised selection heuristics have been successfully employed to reduce dimensionality and achieve significant gains in text classification accuracy (Wiratunga et al. 2004). However, the main challenge in classifying terms in product reviews is lack of labelled data. This is because unlike typical classification tasks where class labels are explicitly defined for each document, product review labels need to be available for individual sentences, making this far more demanding. A comparative analysis of four traditional feature selection techniques: Information Gain, Mutual Information, Chi-squared Test and Document Frequency; showed that Document Frequency (DF), an unsupervised approach, is a reliable measure for selecting informative features (Yang and Pedersen, 1997). Similar findings are observed in Chen et al. (2015) where document frequency and Information Gain were applied. In this paper, we adopt DF as the feature selection technique to select relevant aspects.

2.2 Aspect-level Sentiment Analysis

There are two main tasks in aspect-level sentiment analysis: aspect extraction and sentiment classification. Aspect extraction focuses on extracting aspects that the reviewer refers to in a given review. Prior research indicates that product aspects are generally nouns and compound nouns (Nakagawa and Mori, 2002). Therefore, the most common approach in the current literature of review-based recommender systems involves the use of frequent nouns to identify potential aspects (Hu and Liu, 2004). The intuition is that frequent nouns are more likely to be relevant. However, this approach generates many spurious aspects. This is because in product reviews authors describe their experience or an event without giving any opinion. Furthermore, some nouns are extracted as aspects due to parsing errors. Instead of focusing on the frequency of an aspect, the dependency relations approach identifies aspects using semantic relationships between words (Qiu et al. 2011). Since dependency-based methods extract aspects by means of syntactic relations between pairs of words in a sentence they are not restricted to frequent aspects. Therefore, in this work we apply dependency relations to extract aspects for product representation.

Sentiment classification assigns a positive or negative label to opinionated documents, paragraphs or sentences. Unlike classical sentiment classification, aspect sentiment classification aims to consider the aspect in a sentence during classification. A common approach in aspect sentiment classification is the use of lexicons to determine 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 levels 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). In this work, our contribution is not focusing on sentiment classification. Therefore, we use the existing sentiment classification tool SmartSA (Muhammad et al. 2016) to accomplish this task.

3. Proposed Social Recommendation Process

An overview of the social recommendation process appears in Figure 2. 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 user reviews with dominant products inferred from the preference graph. We advocate the use of weighted aspect-level sentiment analysis and learn these weights by comparing the sentiment difference between node pairs in the preference graph.

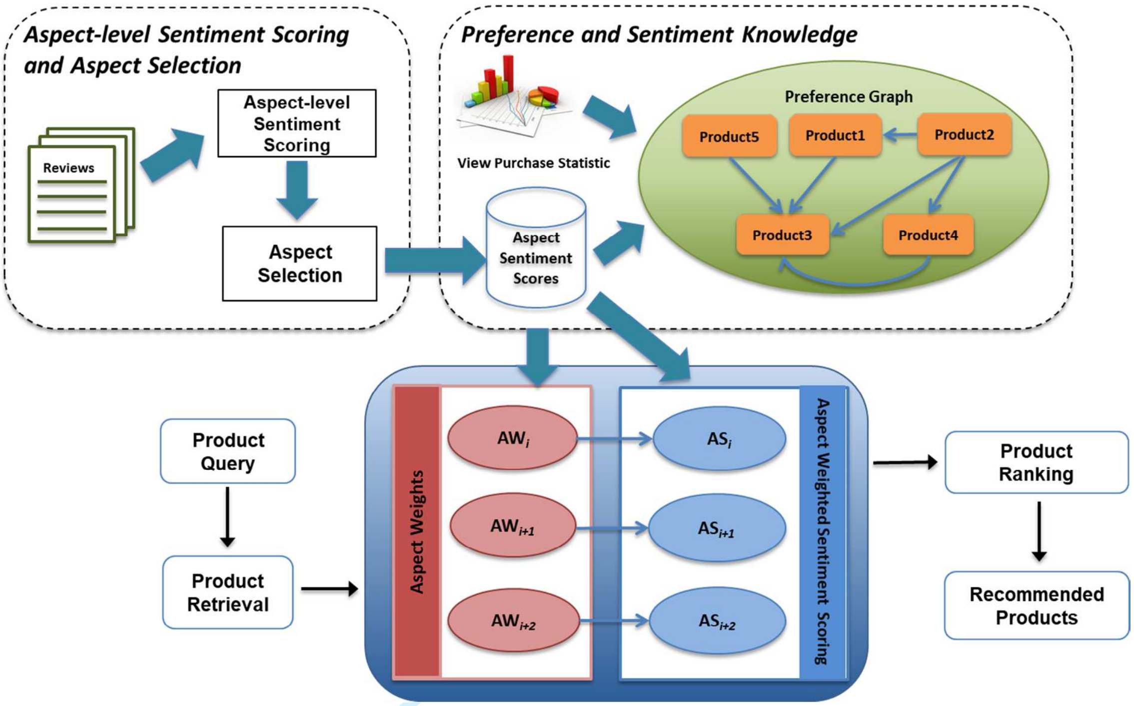


Figure 2: Social Recommendation Process

3.1 Aspect-level Sentiment Scoring

Existing work has shown that unsupervised dependency relations-based approaches outperform frequent noun approaches (Qiu et al. 2011). Therefore, in our work we extract aspects based on a set of selected dependency relations. Specifically, we use the Stanford CoreNLP[1] parser to return the dependency relations between words in a sentence. Then, noun terms that are related by the selected dependency relations are extracted as potential aspects. For example, after parsing the sentence "*The camera has a good lens*", the noun, "*lens*" is related to the adjective, "*good*" by the dependency relation ***amod (Adjectival Modifier)***. This means that "*good*" is an adjectival modifier of the noun "*lens*". Thus, we extract "*lens*" as an aspect. There are 47 dependency relations defined in the Universal Dependencies for English[2]. In this study, we extract aspects using a combination of selected dependency relations and rule-based frequent noun approaches, which achieved best recommendation performance among all the baselines approaches in previous study (Chen et al. 2017). Specifically, dependency relations that frequently relate nouns and sentiment words are selected to extract aspects. The list of selected dependency relations is summarised in Table 1[3].

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

Table 1: Selected Dependency Relations

Aspects extracted from dependency relations are not all genuine aspects. Therefore, we filter aspects using the following rules:

1. Aspects that are technical specifications (e.g. Sigma 18-250mm).

---

[1] *http://stanfordnlp.github.io/CoreNLP/*
[2] *http://universaldependencies.org/en/dep/index.html*
[3] *Dependency relations definitions can be found in the universal dependencies website*[2]

2. Aspects that have a frequency lower than 2 in the reviews.
3. Aspects that do not co-occur with sentiment words.

For each remaining aspect, we use the nearest adjective word to the aspect as the target sentiment word. In this work, we use SmartSA, a state-of-the-art lexicon-based sentiment classification system (Muhammad et al. 2016), to obtain the polarity score of the target sentiment word from SentiWordNet. The score is modified by SmartSA to take into consideration negation terms and lexical valence shifters that can change sentiment orientation. Negation terms and valence shifters are assumed to affect terms within a specific text window (Thelwall et al., 2012). Therefore, 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.

3.2 Aspect Selection
The aspect extraction algorithm described in Section 3.1 extracted more than a thousand unique aspects for each product domain in our datasets. Typically, all extracted aspects from reviews are used in recommendation. However, not all aspects are important for a purchase decision. Therefore, the extracted aspects are not all relevant for product representation.

Frequency is an unsupervised feature selection method that selects frequently occurring terms in a document. We compute the relative frequency of an aspect occurring over the set of reviews as follows:

$$FREQ(a_i) = \frac{f(a_i)}{\sum_{j=1}^{|A|} f(a_j)} \tag{1}$$

Here, $FREQ(a_i)$ returns the relative frequency of an aspect a_i appearing in reviews and A is a set of unique aspects. Frequent occurrence of aspects in online reviews is perceived as important and therefore aspects that are ranked at the top are selected for product representation.

3.3 Aspect Weighted Sentiment Scoring
Reviews are authored following the purchase of products and comprise user opinions in the form of positive and negative sentiment. Strength of sentiment expresses the intensity with which an opinion is stated with reference to a product (Turney, 2002). We exploit this information to rank our products, such that higher ranked products correspond to higher positive sentiment for important aspects. Therefore, we perform a finer-grained sentiment analysis of reviews by computing sentiment at the aspect level. $ProductScore$ of a product p_i, given a set of related reviews R^i is computed as a weighted summation of sentiment expressed at the aspect level as follows:

$$ProductScore(p_i, a_j) = \frac{\sum_{j=1}^{|A^i|} AspectWeight(a_j) * AspectSentiScore(p_i, a_j)}{\sum_{j=1}^{|A^i|} AspectWeight(a_j)} \tag{2}$$

where $AspectSentiScore$ 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 shared between the query and candidate product. Accordingly, the aspect-level sentiment score is:

$$AspectSentiScore(p_i, a_j) = \frac{\sum_{m=1}^{|R_j^i|} SentiScore(r_m)}{|R_j^i|} \tag{3}$$

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 $SentiScore$ is generated by the SmartSA system (described in Section 3.1) for each r_m.

A preference relation between a pair of products denotes the preference of one product over the other through the analysis of viewed and purchased product relationship. Figure 3 illustrates a preference graph, $G = (P, E)$, generated from a sample of data on Digital SLR Camera. The set of nodes, $p_i \in P$, represents products, and the set of directed edges, E, is preference relations, $p_j \succ p_i$, such that a directed edge from product p_i to p_j with $i \neq j$ represents that, for some users, p_j is preferred over product p_i. In some cases where $p_j \succ p_i$ and $p_i \succ p_j$, a bidirectional preference relation can be observed. For any p_i, we use E^i to denote incoming and E_i for outgoing product sets.

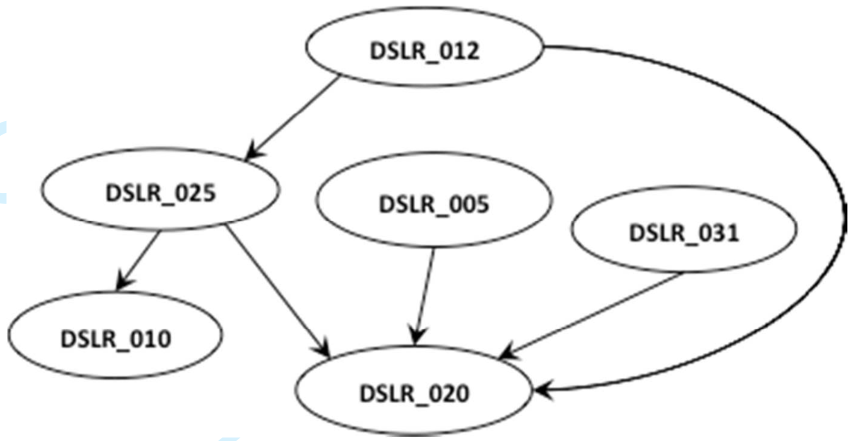


Figure 3: Preference sub-graph for DSLR cameras

A product purchase choice is a preference made on the basis of one or more aspects. The notion of aspect importance arises when the same set of aspects contributes to similar purchase decisions. Using this same principle, aspect weights are derived by comparing the aspect sentiment score differences between viewed and purchased product pairs in which $(p_x, p_y) \in \{(p_x, p_y)\}_{x \neq y}^{t}$

$$AspectWeight(a_j) = \frac{\sum_{x=1}^{|P|} \sum_{y=1}^{|P|} \delta(a_j, p_x, p_y)}{|t \in E|} \quad (4)$$

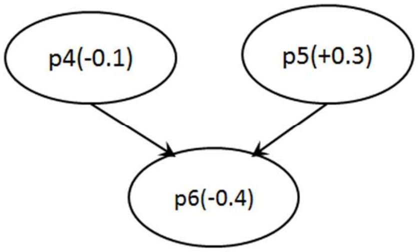
where either $p_x \succ p_y$ or $p_y \succ p_x$ or both, and t is the set of product preference pairs containing aspect a_j. We remove preference relations that relate from the product to itself such that $p_x \neq p_y$. This is because the preference difference for this relation is 0, which does not contribute to learning aspect weights. Accordingly, given a product pair where $p_x \succ p_y$, the preference difference score of this product pair for aspect a_j is computed as:

$$\delta(a_j, p_x, p_y) = |L_{min}(A, E)| + \delta'(a_j, p_x, p_y) \quad (5)$$

$$\delta'(a_j, p_x, p_y) = AspectSentiScore'(p_x, a_j) - AspectSentiScore'(p_y, a_j) \quad (6)$$

$$AspectSentiScore'(p, a) = \begin{cases} 1, & \text{if } AspectSentiScore(p, a) > h; \\ 0, & \text{otherwise.} \end{cases} \quad (7)$$

Here $|L_{min}(A, E)|$ is the lowest preference difference score obtained over all the aspects for all product preference pairs. This is required to avoid having negative aspect weights. In Equation 7, $AspectSentiScore'$ of aspect a in product p is 1 if $AspectSentiScore$ is greater than a threshold h and 0 otherwise. Our default value for h is set to 0 such that an aspect with an overall positive sentiment in the preferred product will be given greater importance. We illustrate our preference-based aspect weighting approach with the following example. Figure 4 illustrates the notion of preference difference calculation using a trivial three node preference graph. In Figure 4, the relation $p_6(-0.4) \succ p_5(+0.3)$ denotes that product p_6 is preferred over p_5 and they have an aspect sentiment score of -0.4 and $+0.3$ respectively for aspect *screen*. Here p_5 has a sentiment score greater than 0 and p_6 has a sentiment score of less than 0. Based on Equation 7, $AspectSentiScore'$ for $p_5 = 1$ and $p_6 = 0$. Next, the sentiment difference between the product pairs is calculated using Equation 5 and 6. As a result, *screen* has an aspect weight of 0.5.



Preference Relations	$\boldsymbol{\delta'(a_j, p_x, p_y)}$	$\boldsymbol{\delta(a_j, p_x, p_y)}$	**Aspect Weights (a_j)**
$p_6 \succ p_4$	$0 - 0 = 0$	$1 + 0 = 1$	$\frac{1}{2} = 0.5$
$p_6 \succ p_5$	$0 - 1 = -1$	$1 + (-1) = 0$	

Figure 4: Sub-graph for aspect *screen*

4. Evaluation

The aim of our evaluation is to demonstrate the benefit of aspect weights and selection in product recommendations. The following sections introduce the experiment datasets, recommendation strategies, evaluation metrics, baseline approaches, our proposed approaches and evaluation results.

4.1 Datasets

We collected data from Amazon.com during April 2014 and November 2014. In particular, we focused on seven product categories: DSLR cameras, Laptops, Tablets, Phones, Printers, Mp3Players and TV. The datasets include information about the product, their reviews and the list of products that other consumers buy after viewing a product. Since we are not focusing on the cold-start problem, products with less than 10 reviews are removed. Table 2 shows the descriptive statistics of the seven datasets used in the experiments.

Descriptions	**DSLR**	**Laptops**	**Tablets**	**Phones**	**Printers**	**Mp3**	**TV**
No. of products	56	121	122	51	82	55	52
No. of reviews	6206	3734	15,007	2595	11,442	4690	5860
No. of unique aspects	4298	1553	19,566	5379	2077	6771	1386
No. of product preference pairs	48	574	212	40	110	53	77

Table 2: Descriptive Statistics of Dataset

4.2 Recommendation Strategies

We evaluate recommender systems using a standard *k*-fold cross validation. We adopt stratified sampling to split the data into train and test sets and use each product in the test set as a query product. For every query product, we generate a ranked list of recommended products. The similarity of a candidate product in a given retrieval set in terms of the target query product is measured using the standard cosine similarity. Cosine similarity is a widely used conventional approach in content-based recommender systems (Sarwar et al. 2001; Pazzani and Bilsus, 2007). It is based on the assumption that users are likely to look for other candidate products (C) which are similar to the product that they are currently looking at (query product, Q). Accordingly, cosine similarity is defined below:

$$Sim(Q, C) = \frac{\sum_{i=1}^{n} Q_i C_i}{\sqrt{\sum_{i=1}^{n} (Q_i)^2} \sqrt{\sum_{i=1}^{n} (C_i)^2}} \quad (8)$$

Here Q_i and C_i are the weights of the i^{th} aspect in product Q and C respectively which are computed using the Term Frequency-Inverse Document Frequency (TF-IDF) term weighting scheme, as follows:

$$TF - IDF(a, Q, P) = tf(a, Q) \times idf(a, P) \quad (9)$$

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 $idf(a, P)$ are given as follows:

$$tf(a, Q) = 1 + \log(f_{a,Q}) \quad (10)$$

$$idf(a, P) = \log\left(\frac{|P|}{|p \in P : a \in p|}\right) \quad (11)$$

Here $f_{a,Q}$ is the frequency of occurrence of aspect a in Q. The idf of aspect a is obtained by taking the logarithm of the total number of products divided by the number of products that contain a.

4.3 Evaluation Metrics

To validate the ranking model, we use overall user ratings as the measure of product quality. We quantify the effectiveness of our ranking model using two evaluation metrics and report statistical significance using the paired t-Test at 95% confidence level. Following the previous work (Chen et al. 2017; Dong et al. 2013; Zhang et al. 2010a), we apply Mean Average Precision (MAP) and Rank Improvement (RI) as evaluation metrics to measure rankings of products.

- MAP@15: 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. Here, we choose retrieval size N =15 because our retrieval set size is limited by the number of products in the dataset. For example, in the Phones dataset, there are 51 products available. After splitting the dataset into training and test set we have 35 products for training and 16 products for testing. Therefore, the number of products retrieved for each query product is never more than 15 (the last one being the target query product). To evaluate our proposed approach, we generate ground truth for each query product in the test set in the form of $(p_q, better)$ where p_q is a query product and $better$ is a ranked list that consists of the corresponding top 3 candidate products that are similar to and have a higher overall user rating (*'better'*) than p_q. MAP is defined as follows:

$$MAP@N = \frac{1}{N}\sum_{j=1}^{N}\frac{1}{|Q_j|}\sum_{k=1}^{|Q_j|} Precision(k) \quad (12)$$

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

- RI: The average gain in rank position of recommended products over the query product is computed relative to a benchmark ranking. This approach estimates the degree to which the recommended product is '*better*' than the query product. We generate the benchmark ranking according to the overall user ratings of products.

$$RI(\%) = \frac{\sum_{i=1}^{n} benchmark(p_q) - benchmark(p_i)}{n * |P - 1|} \quad (13)$$

  Here, $benchmark$ returns the position of a recommended product p_i on the benchmark product ranking and n is the number of products in the recommended list. We set $n = 3$ because products that are ranked at the top are likely to get users' attention or clicks. Therefore, it is important that the products that are ranked at the top are better than the query product. In this metric, the greater the rank gain of the recommended product over the query product (p_q) the better the recommendation. Suppose the query product is ranked 40th on the benchmark ranking of 81 unique products, and the recommended product is ranked 20th on the benchmark ranking list, then the recommended product will have a relative rank improvement of 25%.

4.4 Baseline Approaches for Social Recommender Systems

Based on the survey of previous work, we identified three benchmark recommendation algorithms to be included in our comparative study: PageRank algorithm, Cosine similarity and *Better* score.

*4.4.1 PageRank algorithm (**PageRank**)*

PageRank algorithm recommends items based on their popularity in a graph-based structure. In this approach, an item is a node in the graph and a link between two nodes represents a relationship between the two items. Previous research exploit this relationship to gauge popularity of an item, whereby a PageRank score is computed by evaluating quality and quantity of links to a node (Chen et. al. 2014; Ding, 2011; Wang and Wang, 2014). The most popular item in the graph will have the highest PageRank score.

Several studies indicate that product popularity is a powerful form of feedback that influence users' purchase decision (Celma and Cano, 2008; Zhu and Huberman, 2014; Salganik, 2006). However, Chen et al. (2014) argued that popularity of a product is not the only reason that influences consumer purchase decisions. They suggested that there is a need to leverage further dimensions of knowledge sources such as users' sentiment from product reviews for product recommendation. This paper aims to fill in the gap by comparing a popularity-based approach with our proposed approach, which capitalise on users' sentiment knowledge to recommend products.

More formally, the PageRank score is defined as (Page et al. 1999):

$$PageRank(p_i) = \sum_{p_j \in E^i} \frac{PageRank(p_j)}{|E_j|} \qquad (14)$$

where E^i is the set of all viewed products over which p_i is preferred and E_j is the set of products that are preferred after viewing p_j. By applying this equation to our preference graph described in Section 3.3, we obtain an overall preference score for each product included in our preference graph.

*4.4.2 Cosine Similarity (**Cosine**)*

Social recommender systems that utilise product reviews to identify features for product representation is a form of content-based recommendation. Product aspects help describe the content and when given a query's content (in the form of aspects) we are able to compare each candidate product's aspect value with that of the query product. Each product is represented by a vector in an *n*-dimensional space, where each dimension corresponds to a separate aspect. Here, the value of an aspect represents its frequency in reviews. We make use of the cosine similarity presented in Equation 8 in Section 4.2, to compute the similarity between candidate and query products and then select the *k* most similar candidate products for recommendation. This similarity-based ranking and recommendation method (Pazzani and Billsus, 2007; Sarwar et al., 2001; Dong et. al., 2016) is our second baseline approach.

*4.4.3 Better score (**Better**)*

The cosine similarity approach is a simple method for product recommendation. However, the availability of users' sentiments in product reviews suggests an alternative recommendation approach that includes users' sentiments. The state-of-the-art approach that utilises users' sentiments in a content-based recommender system is the *Better* score (Dong et al. 2016). Formally, the *Better* score is defined as (Dong et al. 2016):

$$better(A, Q, C) = \frac{Sentiment(A, C) - Sentiment(A, Q)}{2} \quad (15)$$

$$Better(Q, C) = \frac{\sum_{\forall A \in Aspects(Q) \cap Aspects(C)} better(A, Q, C)}{|Aspects(Q) \cap Aspects(C)|} \quad (16)$$

One major limitation observed in this approach is that it assumes that users place equal importance to all aspects relevant to a product. Our aim in this paper is to address this weakness by inferring aspect importance from preference relations generated over product view-purchased relations discussed in Section 1.

4.5 Our Proposed Approaches

The following are the variations of our proposed $ProductScore$ used to rank products.

1. Preference (**Pref**) uses aspect weights and aspect sentiment scores when generating product scores in Equation 2 without considering the sentiment threshold in Equation 7.
2. Pref+SentimentThreshold (**PrefST**) uses aspect weights and aspect sentiment scores when generating product scores in Equation 2 with sentiment threshold in Equation 7.
3. **AllAspects** uses the average of all aspect sentiment scores in Equation 3.
4. AspectSelection (**AS**) is similar to **AllAspects** but only considers a subset of aspects selected by frequency ($FREQ$).
5. **PrefST+AS** is our proposed approach that combines **AS** and aspect weights (**PrefST**). We repeat the experiment of approach 2 by using only a subset of aspects selected by frequency ($FREQ$).

4.6 Results and Discussion

In this section, we first discuss the results from comparing baseline methods to preference-based aspect weights (**PrefST**). We then highlight the results from comparing the effects of using all or a subset of aspects for recommendation. Finally, we identify a performance difference between **PrefST** and **PrefST+AS** and discuss important findings from the experiments.

4.6.1 Evaluation on Aspect Weighted Sentiment Scoring

Tables 3 and 4 list the results in terms of MAP@15 and RI respectively on the seven datasets. An asterisk (*) in the tables indicates statistical significance compared to the baseline methods (**PageRank**, **Cosine**, **Better**) and results with two asterisks (**) indicate statistical significance compared to the baseline methods and the variations of other proposed $ProductScore$ (e.g. **Pref**, **PrefST**, **AllAspects**, **AS**, **PrefST+AS**). First of all, we observe that **Cosine** which does not consider sentiment of aspects, underperforms compared to an aspect weighted sentiment driven approach (**PrefST**) in DSLR, Laptops, Tablets, Phones, Mp3Players and TV in MAP. This finding supports those reported in Dong et al. (2016) where similarity-based approaches that do not consider sentiment of aspects fail to recommend products higher ranked than the query product. In Table 4, the RI for **PageRank** and **Cosine** is less than 7% in most cases. Since recommending a product with one rank position better than the query product results in 7% rank improvement, a RI of less than 7% suggests that **PageRank** and **Cosine** recommend products that rank below the test query product in most cases. In contrast, sentiment-driven approaches such as **Better**, **Pref** and **PrefST** have RI greater than 7% in most cases. This emphasises the effectiveness of sentiment-driven approaches. Although **Better** is a sentiment-driven approach, comparing sentiment values of aspects between products without considering aspect importance does not produce a good product representation. As a result, we can see that **PrefST** consistently outperforms **Better** in both evaluation metrics.

In the **PrefST** approach, the sentiment threshold applied to the preference difference score in Equation 7 helps improve the overall recommendation performance. This is demonstrated in Tables 3 and 4 where consistent improvement is observed across all product categories with **PrefST**. Specifically, the improvement gained by adding the sentiment threshold to **Pref** is on average 4.75% and 17.63% for MAP and RI respectively. This shows that setting a sentiment threshold for each product preference pair to determine the sentiment polarity of an aspect is superior to the approach that does not consider a sentiment threshold in estimating aspect weights. Furthermore, there is another benefit of applying Equation 7 to aspect weighting. For instance, given an aspect, a, appearing only in the reviews of p_1 and p_2, where p_1 is preferred over p_2. The sentiment score of a for products p_1 and p_2 are -0.198 and -0.344 respectively. Therefore, it is reasonable to say that a is not a good aspect in both products and that there is no evidence to suggest a is an important aspect. Consequently, weight 0 should be assigned to a. In contrast, the preference difference score between p_1 and p_2 for aspect a is a positive difference (0.146) in the **Pref** approach. Therefore, setting a sentiment threshold helps to overcome the issue caused by negative sentiment.

Methods	DSLR	Laptops	Tablets	Phones	Printers	Mp3Players	TV
PageRank[1]	0.295	0.375	0.403	0.366	0.365	0.618	0.394
Cosine[2]	0.398	0.357	0.389	0.393	0.379	0.443	0.445
Better[3]	0.674	0.484	0.492	0.600	0.353	0.548	0.550
Pref	0.720*	0.498[1,2]	0.555*	0.634[1,2]	0.360	0.613[2,3]	0.584[1,2]
PrefST	0.746*	0.527*	0.560*	0.649*	0.414[1,3]	0.623[2,3]	0.607*
AllAspects	0.740*	0.497[1,2]	0.567*	0.644*	0.397	0.635[2,3]	0.566[1,2]
AS	0.740*	0.581**	0.504[1,2]	0.587[1,2]	0.437[1,3]	0.663[2,3]	***0.688***
PrefST+AS	***0.805***	***0.644***	***0.581****	***0.690****	***0.480***	***0.717***	0.655**

Table 3: MAP@15[4]

Methods	DSLR	Laptops	Tablets	Phones	Printers	Mp3Players	TV
PageRank[1]	6.23	3.45	14.89	-9.04	4.21	16.57	8.25
Cosine[2]	12.72	-0.21	6.83	0.71	1.93	-1.57	-2.65
Better[3]	21.61	13.83	17.57	7.68	9.33	8.70	6.71
Pref	25.80[1,2]	15.42[1,2]	20.97*	10.20*	8.54[1,2]	11.90[2,3]	9.19[2]
PrefST	26.80*	18.70*	22.51*	10.81*	12.05[1,2]	12.69[2,3]	12.61[2,3]
AllAspects	27.67*	17.90[1,2]	22.50*	10.15[1]	7.31[2]	11.16[1,2]	7.01[2]
AS	***28.89****	22.36*	22.68*	10.10[1,2]	13.86*	16.71[2,3]	16.41*
PrefST+AS	***28.89****	***26.46****	***23.67***[*]	***11.16****	***22.43***	***18.15***[2,3]	***18.21****

Table 4: RI(%)[4]

DSLR	Laptops	Tablets	Phones	Printers	Mp3	TV
camera	laptop	use	use	printer	product	picture
lens	buy	tablet	phone	ink	battery	this tv
this camera	screen	buy	buy	product	thing	quality
buy	work	work	time	set	make	buy
picture	like	time	work	print	quality	sound
take	price	price	like	quality	mp3	screen
like	time	screen	battery	buy	work	set
photo	this laptop	well	look	work	music	work
quality	keyboard	product	purchase	this printer	play	remote
feature	window	one	quality	set up	charge	price

Table 5: Top 10 Most Frequent Shared Aspect Words

One important observation from the results is that the MAP score for **PrefST** in Mp3 is close to the **PageRank** score. In RI**, PageRank** performs better than **PrefST**. The reason for this poor performance might be explained by the number of shared aspects between query and candidate products. In the Mp3 dataset, the number of shared aspects is consistently higher than with other datasets. Specifically, the average minimum number of shared aspects for the DSLR, Laptops,

[4] (1,2) indicates the proposed approach achieved a significant improvement over baseline approach 1 and 2 (e.g. PageRank and Cosine).

Tablets, Phones, Printers and TV datasets is between 1 and 21 (mean= 8.8, standard deviation = 7.1). However, the minimum number of shared aspects in Mp3 is 80, which is the highest among all datasets. To ensure the aspects extracted are relevant in product representation, we examined the list of aspects that are frequently shared between query and candidate products. Table 5 shows a list of the aspects which are the most frequently shared between query and candidate products in all product categories. Here, we observe that there are a number of spurious aspects. For instance, the terms "take", "like" and "thing" are not aspects of a product. This shows that some extracted aspects contain spurious aspects. Thus, a higher number of shared aspects increases the likelihood of spurious aspects contributing to the $ProductScore$ computation, leading to a poor performance.

In order to distinguish the effect of aspect weights, we compare the results of **PrefST** with **AllAspects**. The comparison between **PrefST** and **AllAspects** indicates that the former is performing better than the latter in most cases. For instance, in the TV datasets **PrefST** achieves 7.24% and 79.89% higher MAP and RI than **AllAspects**. Similarly, in Printers there is an improvement of 4.28% for MAP and 64.84% for RI. More importantly, we observe that **PrefST** achieved significant improvements in majority of the datasets. This shows that **AllAspects,** which does not consider aspect importance, had a disadvantage in ranking products. This implies that a combination of user product purchase preferences and sentiment knowledge can more accurately infer the importance of an aspect.

4.6.2 Evaluation on Aspect Selection

The objective of using feature selection metrics is to exploit important aspects to rank products. We assess the effect of increasing aspect subset size on recommendation performance by using $FREQ$ in Section 3.2. We empirically test on aspect subset sizes between 1 to 200 top-ranked aspects and report the best results (bold italic) in Tables 3 and 4. The number of aspects selected for **AS** and **PrefST+AS** is shown in Figure 5 and 6. We first compare the results of product ranking recommendation with aspect selection (**AS**) and without aspect selection (**AllAspects**) to assess the importance of aspect selection. Results show that **AS** performs best in RI where **AS** improves upon **AllAspects** in all datasets except for Phones. More importantly, we observe that the results for Mp3 improve with aspect selection. The observations on the results for **AS** suggest that integrating an aspect selection module in our recommendation model can effectively remove erroneous aspects for product representation. Furthermore, recall from the analysis in Section 4.6.1, that among all datasets Mp3 has the highest number of shared aspects. The improvement observed in Mp3 suggests that given a dataset that contains a high number of shared aspects, selecting relevant aspects gives a better product representation compared to using aspect weighted sentiment scoring method that do not consider sentiment threshold (**Pref**).

Next, we compare the results of aspect selection **(AS)** with and without aspect weights (**PrefST+AS**). The comparison between **AS** and **PrefST+AS** shows that **PrefST+AS** gives better results in most cases. Specifically, Figure 5 shows that the number of aspects required to achieve a significant improvement in MAP over the baselines is no more than 126 aspects. In some cases, only 5 aspects are required to achieve results with a significant improvement over the baselines. Similar observations can be made with the results for RI in Figure 6 where, on average, 60 aspects are required to achieve significant improvements. Furthermore, we observe that assigning weights to selected aspects reduce the number of aspects required for recommendation. For instance, in Tablets the MAP score is improved by 3.7% by reducing the number of aspects from 59 to 22. The performance improvement gained by assigning aspect weights to selected aspects suggests that frequent aspects are the most relevant for product representation. Our findings from aspect selection evaluation supports that of Zhang et al. (2010b), where one of the important factors that determine the importance of aspects is feature frequency.

Combining the evaluation results of aspect weighted sentiment scoring and aspect selection, we can conclude that considering aspect importance using user product purchase preferences and sentiment knowledge not only increases the chances of ranking the top products in higher position, but also recommends products that are '*better*' than the query product. Results comparing weighted (**PrefST** and **PrefST+AS**) and non-weighted approaches (**AllAspects** and **AS**) confirm that weighted approaches outperform non-weighted approaches and that the aspect weights can be effectively inferred from sentiment and purchase preferences. Our initial observation on the performance of **PrefST** demonstrates that products that are represented by large numbers of aspects reduce the opportunity of ranking good rating products at the top. This observation is further confirmed by the findings in the results of **PrefST+AS** where selecting a small number of relevant aspects improved recommendation performance. A statistical analysis on the different variations of the proposed approaches shows that **PrefST+AS** achieve significant improvement in majority of the datasets in MAP (5 out of 7) and 1 dataset (Printers) in RI. Although there is no significant difference on the variations of the proposed approach in RI, **PrefST+AS** remains the best-performing approach in RI with significant improvements over baseline approaches in majority of the datasets (6 out of 7).

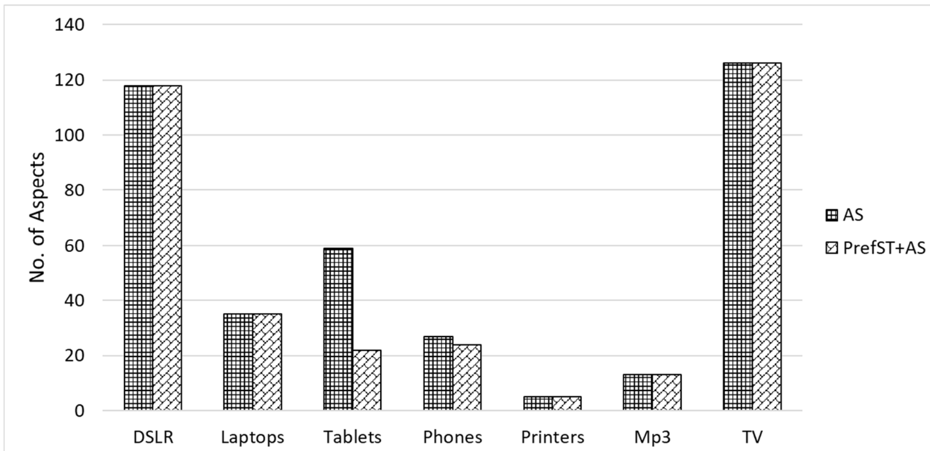


Figure 5: Number of Aspects Selected (MAP)

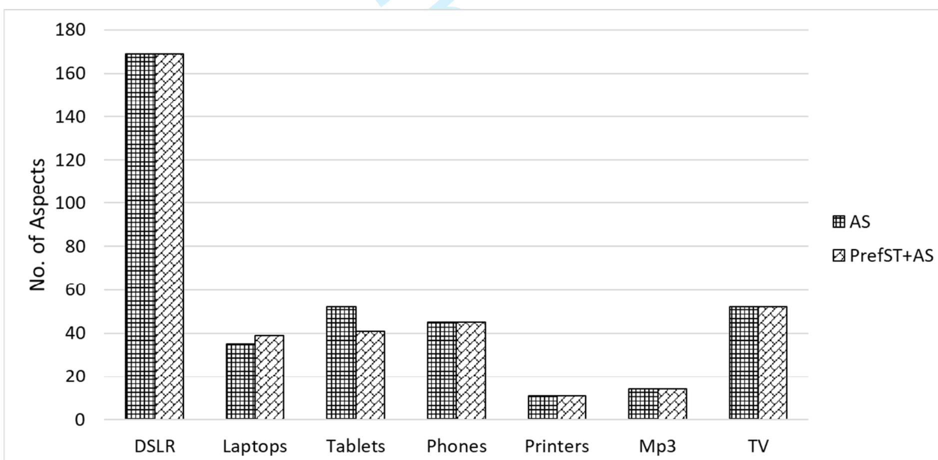


Figure 6: Number of Aspects Selected (RI)

5. Conclusion

Social media has created new opportunities for recommender systems research. The high volume of user-generated content available provides an opportunity to improve recommendation algorithms. However, user-generated content has two limitations: (1) Product reviews contain opinions on hundreds of aspects but not all aspects are equally important in representing a product. (2) Aspects extracted using NLP-based techniques alone, typically contains irrelevant or spurious content that can negatively affect recommendation performance. To address these challenges, effective aspect weighting and selection are needed to capitalise on knowledge gathered from user opinions. In this paper, we integrate consumer purchase behaviour analysis and aspect-level sentiment analysis to estimate aspect weights. Specifically, we improve the aspect weighting algorithm by comparing the polarity of aspects in every product preference pair. We apply frequency to evaluate aspect usefulness and selecting the top aspects to avoid using spurious aspects for product representation. The selected aspects are weighted using our proposed aspect weighting algorithm to generate a ranked list of products for recommendation. We demonstrate the effectiveness of our proposed approach in a realistic recommendation setting using benchmarks generated from user ratings. Our results demonstrate that preference knowledge gathered from consumer purchase behaviour analysis can be exploited using a graph-based model together with sentiment polarity to infer aspect weights. Further, selecting aspects that frequently occur in product reviews improves recommendation performance compared to no aspect selection. Finally, our results demonstrated that consistent significant improvements on recommendation performance can be observed when the selected aspects are weighted.

Our proposed recommendation approach does not require individual user preferences in providing recommendation to users. Therefore, the proposed sentiment and preference-guided strategy for product recommendation is a feasible solution to recommend products to new users (e.g. cold-start users) when their preference is not known by the recommender system. Further, a key research implication from our proposed recommender system is its ability to provide explanations on the recommended products to users, due to its reliance on aspect sentiment to recommend products. Being able to justify a recommendation using aspects, weights, and user opinions provides a first step towards providing users with explanations for their recommended products. Our results also confirm that erroneous aspects extracted from product reviews have a detrimental effect on recommendation performance. For future work, we will explore effective aspect selection algorithms that can accurately select genuine and useful aspects for product representation.

References

Celma, Ò. and Cano, P., 2008, August. From hits to niches?: or how popular artists can bias music recommendation and discovery. In *Proceedings of the 2nd KDD Workshop on Large-Scale Recommender Systems and the Netflix Prize Competition* (p. 5). ACM.

Chen, L. and Wang, F. (2013). Preference-based clustering reviews for augmenting e-commerce recommendation. Knowledge-Based Systems, 50, pp.44-59.

Chen, Y.Y., Ferrer, X., Wiratunga, N. and Plaza, E. (2014). September. Sentiment and preference guided social recommendation. In *International Conference on Case-Based Reasoning* (pp. 79-94). Springer International Publishing.

Chen, Y.Y., Ferrer, X., Wiratunga, N. and Plaza, E., (2015). Aspect selection for social recommender systems. In *International Conference on Case-Based Reasoning* (pp. 60-72). Springer International Publishing.

Chen, Y.Y., Wiratunga, N. and Lothian, R., (2017). Effective dependency rule-based aspect extraction for social recommender systems. ASSOCIATION FOR INFORMATION SYSTEMS.

Christensen, I., Schiaffino, S. and Armentano, M., 2016. Social group recommendation in the tourism domain. *Journal of Intelligent Information Systems*, *47*(2), pp.209-231.

Ding, Y., Yan, E., Frazho, A. and Caverlee, J., (2009). PageRank for ranking authors in co-citation networks. Journal of the American Society for Information Science and Technology, 60(11), pp.2229-2243.

Dong, R., Schaal, M., O'Mahony, M.P., McCarthy, K. and Smyth, B. (2013). Opinionated product recommendation. In International Conference on Case-Based Reasoning (pp. 44-58). Springer Berlin Heidelberg.

Dong, R., O'Mahony, M.P., Schaal, M., McCarthy, K. and Smyth, B., (2016). Combining similarity and sentiment in opinion mining for product recommendation. *Journal of Intelligent Information Systems*, *46*(2), pp.285-312.

Esparza, S.G., O'Mahony, M.P. and Smyth, B. (2011). Effective product recommendation using the real-time web. In Research and Development in Intelligent Systems XXVII (pp. 5-18). Springer London.

Esuli, A., Sebastiani, F. (2006). Sentiwordnet: A publicly available lexical resource for opinion mining, in: Proc. Language Resources and Evaluation Conference, pp. 417–422.

Guy, I. (2015). Social recommender systems. In Recommender Systems Handbook, (pp. 511-543). Springer.

Hu, M. and Liu, B. (2004). Mining and summarizing customer reviews. In Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining (pp. 168-177). ACM.

Koren, Y., Bell, R. and Volinsky, C. (2009). Matrix factorization techniques for recommender systems. Computer, 42(8).

Levi, A., Mokryn, O., Diot, C., and Taft, N. (2012). Finding a needle in a haystack of reviews: cold start context-based hotel recommender system. In Proceedings of the sixth ACM conference on Recommender systems (pp. 115-122). ACM.

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 US.

Liu, H., He, J., Wang, T., Song, W., & Du, X. (2013). Combining user preferences and user opinions for accurate recommendation. Electronic Commerce Research and Applications, 12(1), 14-23.

Muhammad, A., Wiratunga, N. and Lothian, R. (2016). Contextual sentiment analysis for social media genres. Knowledge-Based Systems, 108, pp.92-101.

Nakagawa, H. and Mori, T. (2002). A simple but powerful automatic term extraction method. In COLING-02 on COMPUTERM 2002: second international workshop on computational terminology-Volume 14 (pp. 1-7). ACL.

Owoputi, O., O'Connor, B., Dyer, C., Gimpel, K., Schneider, N. and Smith, N.A. (2013). Improved part-of-speech tagging for online conversational text with word clusters. ACL.

Page, L., Brin, S., Motwani, R. and Winograd, T. (1999). The PageRank citation ranking: Bringing order to the web. Stanford InfoLab.

Pazzani, M.J. and Billsus, D. (2007). Content-based recommendation systems. In The adaptive web (pp. 325-341). Springer Berlin Heidelberg.

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.

Salganik, M.J., Dodds, P.S. and Watts, D.J., 2006. Experimental study of inequality and unpredictability in an artificial cultural market. *science*, *311*(5762), pp.854-856.

Sarwar, B., Karypis, G., Konstan, J. and Riedl, J. (2001). Item-based collaborative filtering recommendation algorithms. In Proceedings of the 10th international conference on WWW (pp. 285-295). ACM.

Sun, J., Wang, G., Cheng, X. and Fu, Y. (2015). Mining affective text to improve social media item recommendation. *Information Processing & Management*, *51*(4), pp.444-457.

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.

Turney, P.D. (2002). Thumbs up or thumbs down?: semantic orientation applied to unsupervised classification of reviews, in: Proc. Annual Meeting on ACL, 2002, pp. 417–424.

Vasudevan, S.R. and Chakraborti, S. (2014). Mining user trails in critiquing based recommenders. In Proceedings of the 23rd International Conference on WWW (pp. 777-780). ACM.

Wang, H., & Wang, W. (2014). Product weakness finder: an opinion-aware system through sentiment analysis. *Industrial Management & Data Systems*, *114*(8), 1301-1320.

Wiratunga, N., Koychev, I. and Massie, S. (2004). Feature selection and generalisation for retrieval of textual cases. In European Conference on Case-Based Reasoning (pp. 806-820). Springer Berlin Heidelberg.

Woo Kim, S., Chung, C.W. and Kim, D. (2009). An opinion-based decision model for recommender systems. *Online Information Review*, *33*(3), pp.584-602.

Yang, Y. and Pedersen, J.O. (1997). A comparative study on feature selection in text categorization. In Icml (Vol. 97, pp. 412-420).

Zha, Z.-J., Yu, J., Tang, J., Wang, M., and Chua, T.-S. (2014). Product aspect ranking and its applications. Knowledge and Data Engineering, IEEE Transactions on, 26(5):1211-1224.

Zhang, K., Narayanan, R. and Choudhary, A.N. (2010a). Voice of the Customers: Mining Online Customer Reviews for Product Feature-based Ranking. WOSN, 10, pp.11-11.

Zhang, L., Liu, B., Lim, S. H., and O'Brien-Strain, E. (2010b). Extracting and ranking product features in opinion documents. In Proceedings of the 23rd international conference on computational linguistics (pp. 1462–1470). ACL.

Zhu, H. and Huberman, B.A., 2014. To switch or not to switch: understanding social influence in online choices. *American Behavioral Scientist*, *58*(10), pp.1329-1344.