A CONDITIONAL FEATURE UTILIZATION APPROACH TO ITEMSET RETRIEVAL IN ONLINE SHOPPING SERVICES

Kwanho Kim
Department of Industrial and Management Engineering
Incheon National University
Incheon, 406-772, Republic of Korea
khokim@incheon.ac.kr

Yerim Choi
Department of Industrial Engineering
Seoul National University
Seoul, 151-744, Republic of Korea
iangoozh@gmail.com

Jonghun Park
Department of Industrial Engineering
Seoul National University
Seoul, 151-744, Republic of Korea
jonghun@snu.ac.kr

ABSTRACT

Due to the increasing number of items with a variety of descriptions for a product type, itemset retrieval is considered as an essential function for enhancing shopping experiences of customers in online malls. This paper considers an itemset retrieval problem to construct an itemset consisting of items belonging to the same product type against a query item in which a customer is interested. In contrast to the previous approaches that require additional prior information such as itemset memberships and the known number of itemsets, we propose a semi-supervised itemset retrieval model that can automatically find a target itemset for a query item based on two item features, namely textual description and price. Specifically, in order to precisely identify itemsets, the proposed model conditionally utilizes price feature of an item only when its textual description feature is relevant to that of a query item. Experiment results based on two real-world datasets show that the proposed model outperformed the other alternatives.

Keywords: Itemset retrieval; Semi-supervised approach; Conditional feature utilization; Finite mixture model; e-Commerce

1. Introduction

Owing to the recent proliferation of various types of online shopping services such as open markets, Internet auctions, and social commerce where sellers are allowed to vend their items with their own pricing strategies, there exist many items with diverse prices and various descriptions for a single product type across many online shopping malls [Ramachandran et al. 2011; Wu et al. 2011]. In Google Shopping, for instance, a product type named “Sony Micro Vault USB 16G” is sold in the forms of 65 distinct items with different prices and descriptions. While the availability of multiple items for a single product type offers a wide variety of purchase choices to a customer, it makes difficult for a customer to identify the items belonging to the same product type of interest.

To enhance the customer’s shopping experience, two types of search services, namely item search and itemset search, are often provided. The item search aims to retrieve relevant items to a user query that usually consists of words or phrases while the itemset search seeks to find the set of items belonging to the same product type as that of an item found to be interesting to a customer for the purpose of price comparison. This paper focuses on the itemset retrieval problem, and we refer a given item as a query item and the set of items to be suggested as a target itemset against the query item. In Figure 1, the relationship between a product type and items as well as the relationship between a query item and its target itemset are illustrated.

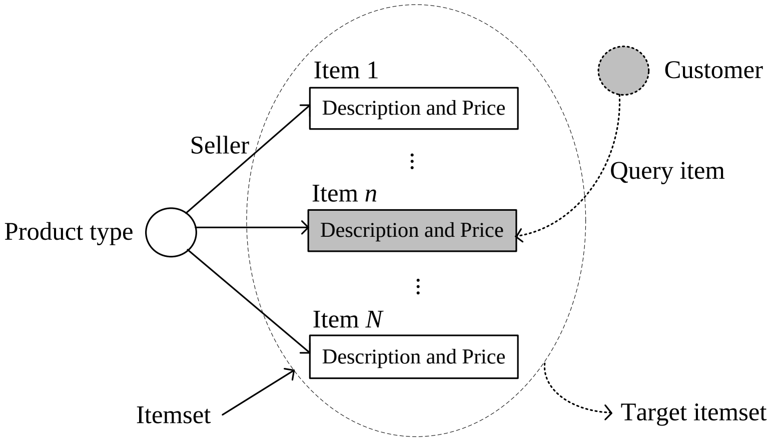


Figure 1: Concept of a query item of a customer and a target itemset against the query item.

Through facilitating automatic retrieval of the target itemset for a query item, customers are provided with comparison results for the items from the same product type, resulting in reduced item search costs [Tan et al. 2010]. In the meantime, the automatic itemset retrieval method is also beneficial to service providers as they are no longer required to prepare all the possible target itemsets in advance. Moreover, the target itemsets can be further utilized to improve the performances of item ranking [Kim et al. 2012b], item recommendation [Linden et al. 2003], and item bundling [Garfinkel et al. 2008] by allowing exploration of the relationships among items.

There have been many research results pertaining to the itemset retrieval. Yet, they have limited applicability due to the requirement of information such as the known number of itemsets [Kannan et al. 2011b; Kim et al. 2012a], the predefined hierarchical structure of itemsets [Abbott et al. 2011; Benjelloun et al. 2009], and the prior knowledge for adjusting model parameters [Kopcke et al. 2010; Wong et al. 2008].

Supervised approaches presented in [Abbott et al. 2011; Geng et al. 2012; Kim et al. 2013b] that require training data on item membership against an itemset may not be viable options for small and medium sized shopping services due to the significant amount of time and cost involved for obtaining the training data [Kannan et al. 2011a]. Furthermore, due to the dynamic nature of an online shopping service where a large number of items are newly created and frequently updated, maintaining such data up-to-date is a challenging task [Tang et al. 2001].

Accordingly, we consider a problem of target itemset retrieval that returns the items in the target itemset against a query item by only using textual descriptions and prices of items without relying on the aforementioned additional information. In this paper, each item is represented by its price and textual description, a short text snippet describing a specific product type. Note that the target itemset retrieval we consider is a special case of the itemset construction problem which groups items of same product type into an itemset [Lynch et al. 2000]. Utilizing only the textual description and price as features of an item facilitates wide usage of the proposed model since they are mandatory information in most online shopping services and other data such as product specification and seller detail are often unavailable.

Retrieving the target itemset against a query item, however, is not trivial due to the ambiguity problem caused by the existence of items with similar textual descriptions across different product types. In this paper, we assume that two items' product types are different if their purchase options are different, and such an ambiguity problem is frequently found in many online shopping services owing to the diversification of product types in terms of purchase options such as delivery condition, packaging unit, and specification [Kannan et al. 2011a] for meeting various customers' needs. In case of a diaper, for instance, product types are often varied along with the number of diapers in a package and diaper size, resulting in very similar textual descriptions of items belonging to different product types. As a result, it is hard to obtain satisfactory results when target itemsets are retrieved by depending only on textual descriptions of items.

On the other hand, the price feature alone does not provide a sufficient clue for identifying item memberships due to its indeterminacy nature. There are many product types including laptops, cell phones, and televisions at similar price ranges. Furthermore, even the items from a single product type may not have similar prices because of various pricing strategies and policies of sellers [Kim et al. 2013a].

Yet, when focusing on a group of product types that differ only in terms of some purchase option, the item price range for a particular product type becomes different from those for the other product types in the group. The difference in the price range is attributed to the decisive effect of purchase options on the item prices. In case of a diaper, for instance, when the same diaper is sold in the forms of various packaging units, product types containing more diapers in a package have higher price ranges than the other types containing fewer diapers.

Figure 2 demonstrates two product types that differ in the memory capacity, “Sony Micro Vault USB Flash Drive 16GB” and “Sony Micro Vault USB Flash Drive 32GB”. The unit of an item price is Korean Won (KRW). The textual descriptions of the items for both product types are similar to each other while the prices of the former product type are always lower than those for the latter in Figure 2.

Textual description	Price
Sony Micro Vault N-Series Click 16 GB USB 2.0 Flash Drive USM16GN	19,030
Sony USM32GN Micro Vault N-Series USB Drive (16GB)	22,264
Sony Micro Vault USM16GN Flash Drive	20,889
Sony Micro Vault USM16GN Flash Drive - 16 GB	25,102
Sony (USM16GN) Micro Vault USB flash drive - 16 GB	22,935
Average of prices	22,044
Variance of prices	5,152,967

(a) Product type named “Sony Micro Vault USB 16G”

Textual description	Price
Sony Media 32 GB Micro Vault N-Series USB Flash Drive, USM32GN	36,960
Sony USM32GN Micro Vault N-Series USB Drive (32GB)	38,489
Sony Micro Vault USM32GN Flash Drive - 32 GB	52,789
Sony Micro Vault USM32GN Flash Drive	45,089
NEW Sony Micro Vault USM32GN Flash Drive - 32 GB	36,289
Average of prices	41,923
Variance of prices	49,064,798

(b) Product type named “Sony Micro Vault USB 32G”

Figure 2: Examples of textual descriptions and prices of items for two product types (unit of price is KRW).

Motivated by the above remarks, we conditionally utilize the textual description and price features of each item in such a way that item’s relevance to a query item is investigated by considering the item’s price only when its textual description is similar to that of the query item in order to address the ambiguity and indeterminacy problems. Specifically, if an item is judged to be similar to a query item in terms of textual description, it becomes a candidate for a member of the target itemset against the query item. Otherwise, it is no longer examined. Subsequently, each candidate is re-evaluated with respect to its price feature. When the price of a candidate is likely to follow the price distribution of items in the target itemset, the candidate is finally judged as a member of the target itemset.

In this paper, we take a semi-supervised approach [Basu et al. 2002; Grira et al. 2004], and develop a target itemset retrieval model, named *cf*-SIM, based on the conditional feature utilization. The proposed model aims to automatically suggest the target itemset against a query item through considering the textual description and price features of items. *cf*-SIM estimates how likely an item belongs to the target itemset against a query item to determine the state of the item by using some available membership information. In detail, we define two possible states for each item against a query item, target state (called τ -state) and non-target state (called ν -state). The items in τ -state are regarded as the ones in the target itemset whereas the items in ν -state are not.

Furthermore, for effective estimation of the states of items against a query item, *cf*-SIM utilizes the item prices on a condition that their textual descriptions are relevant to that of the query item. We model the itemset membership of an item as a mixture of τ -state and ν -state by introducing two hidden variables that are respectively associated with its textual description and price features. The hidden variables represent the states of an item against a query item. Through using an expectation-maximization (EM) based algorithm, the values of the hidden variables and the parameters of *cf*-SIM are estimated, and the probability that an item is in a particular state is calculated. The effectiveness of *cf*-SIM is validated by using two real-world datasets. The experiment results show that *cf*-SIM is not only more effective in retrieving target itemsets than the alternatives considered but also robust against various settings.

The paper is organized as follows. In Section 2, the related studies on the itemset retrieval are presented. In Sections 3 and 4, the proposed model, *cf*-SIM, and its parameter estimation method that bases on an EM algorithm are developed, respectively. Next, the experiment results that validate the effectiveness of *cf*-SIM based on two real-world datasets are shown in Section 5. Finally, implications and limitations are discussed in Section 6, and the paper is concluded in Section 7.

2. Literature Review

Previous studies shown in Table 1 that have addressed the itemset construction problem can be broadly categorized into supervised and unsupervised methods depending on whether or not training data containing the known itemset memberships of items are utilized. The approaches based on a supervised method [Bilenko et al. 2005; Ding et al. 2002; Kannan et al. 2011a; Kannan et al. 2011b; Kim et al. 2013b; Kim et al. 2012a; Kim et al. 2006; Kim et al. 2008; Kirsten et al. 2010; Kopcke et al. 2010] attempted to predict the memberships of newly observed items by exploiting training data which are costly to obtain and sometimes not available. In contrast, the models based on an unsupervised method [Benjelloun et al. 2009; Geng et al. 2012; Wong et al. 2008] judge the item memberships by utilizing a similarity measure without relying on the training data. While the unsupervised models operate without requiring the known itemset memberships, they usually show comparably low performances, making them practically inapplicable.

In particular, Benjelloun’s model [Benjelloun et al. 2009] belongs to an unsupervised approach in which predefined similarity thresholds are used for the itemset construction whereas the proposed model takes a semi-supervised approach in which the query items, instead of the known item membership data, play a role of training data in estimating the parameters involved in the itemset construction. Furthermore, while our model bases on only the textual description and price features that are elementary information available in most online shopping malls, Benjelloun’s model assumes the category information of items which may require a significant amount of manual tasks for preparation. As a result, Benjelloun’s model is not applicable to our problem setting in which all the other information except the textual descriptions and prices is assumed to be unknown.

Table 1: Summary of the related work according to approaches, considered features, and required information.

Approaches	Considered features	Information required	References
Supervised approach	Textual description	Predefined hierarchical structure of itemsets and item images	Kim et al. 2006, Kim et al. 2008, Kanna et al. 2011a
		Predefined hierarchical structure of itemsets	Ding et al. 2002
		Predefined hierarchical structure of itemsets and prior knowledge for parameter adjustment	Abbott and Watson 2011
		The number of itemsets	Kanna et al. 2011b, Kim et al. 2012a
		The number of itemsets and click-through data	Kim et al. 2013b
	Textual description and price	Prior knowledge for parameter adjustment	Bilenko et al. 2005, Kirsten et al. 2010, Köpcke et al. 2010
Unsupervised approach	Textual description	Click-through data	Geng et al. 2012
		Prior knowledge for parameter adjustment	Wong et al. 2008
	Textual description and price	Predefined hierarchical structure of itemsets	Benjelloun et al. 2009

Besides textual descriptions and prices, there are three other types of information required in the previous research: the predefined hierarchical structure of itemsets, the number of itemsets, and the prior knowledge for parameter adjustment. Several studies including [Abbott et al. 2011; Ding et al. 2002; Kannan et al. 2011a; Kim et al. 2006; Kim et al. 2008] used the predefined hierarchical structure of itemsets such as UNSPSC (The United Nation

Standard Products and Services Code Systems) [Abels et al. 2006; Bergamaschi et al. 2002] which is a four level hierarchical classification schema. In some studies [Kannan et al. 2011a; Kim et al. 2013b; Kim et al. 2012a], the number of itemset should be known to construct itemsets while other methods [Bilenko et al. 2005; Kirsten et al. 2010; Kopcke et al. 2010; Wong et al. 2008] required the prior knowledge for parameter adjustment. In addition, some recent work considered miscellaneous information such as click-through information [Geng et al. 2012; Kim et al. 2013b] and item images [Kannan et al. 2011b] to construct itemset more accurately.

Specifically, Bilenko et al. [2005] addressed the itemset construction problem without using the predefined hierarchical structure of itemsets and the number of itemsets through applying a hierarchical agglomerative clustering method whose parameters were determined based on prior knowledge. However, they used a composite similarity function for clustering where the parameters for the similarity measure were trained by utilizing the known itemset memberships.

To reconcile the needs for the known itemset memberships, Wong et al. [2008] proposed an unsupervised method for constructing itemsets by enhancing Bilenko's work. They employed a linear combination of cosine similarity functions and applied a hierarchical agglomerative clustering method based on prior knowledge for determining clustering parameters. In the recent work by Geng et al. [2012], neither the itemset memberships nor prior knowledge for parameter adjustment were necessary for constructing itemsets, but it utilized miscellaneous information such as click-through data in addition to the elementary information such as textual descriptions and prices.

In contrast to the previous studies, however, the proposed method constructs itemsets without relying on the itemset membership data or prior knowledge for parameter adjustment. It only employs the elementary features, textual description and prices, and requires no additional information. Furthermore, while most state-of-the-art methodologies selectively utilizes either an item or a feature according to a certain condition, *cf*-SIM conditionally utilizes a feature of an item by observing the other features of the item, making it possible to address the ambiguity and indeterminacy problems arising in online shopping malls.

3. Proposed Target Itemset Retrieval Model

3.1. Target itemset retrieval based on conditional feature utilization

To address the ambiguity and indeterminacy problems mentioned in Section 1, *cf*-SIM conditionally utilizes the price feature of an item only if its textual description feature is relevant to that of a query item. While the textual description feature is always examined for all items during the itemset retrieval task, utilization of the price feature of an item is conditionally determined during its state estimation to enhance the retrieval performance. The proposed conditional feature utilization makes it possible for a target itemset to be composed of not only textually similar items but also the items with prices that are likely to appear in the target itemset.

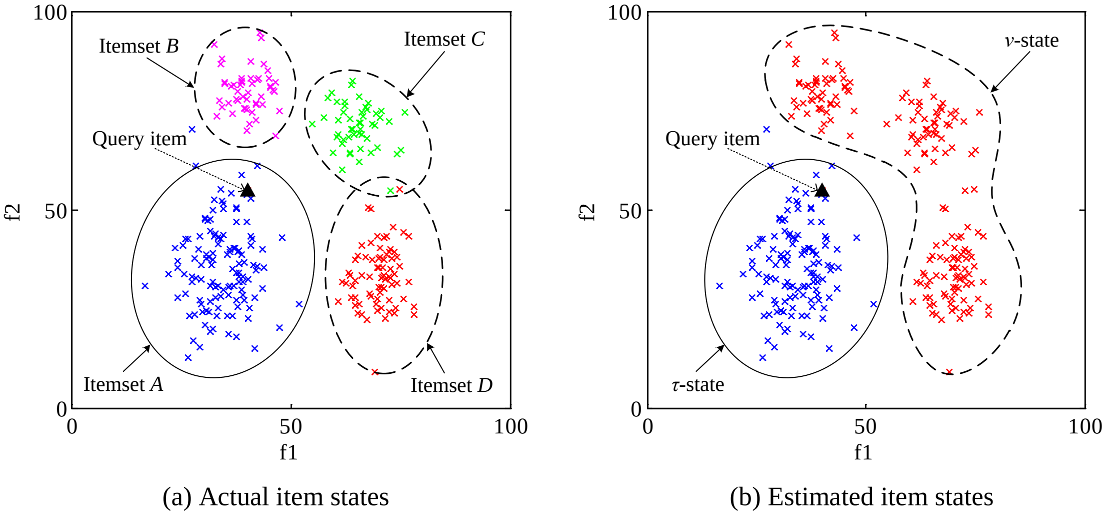
The underlying idea of the conditional feature utilization bases on the existing feature selection methods using finite mixture models that are designed to select a subset of features to improve clustering performances [Law et al. 2004; Zeng et al. 2009]. Yet, we have empirically found that direct application of those methods to shopping itemset construction did not yield satisfactory results when there exists conditional dependency among item features for some items.

From the authors' previous study [Kim et al. 2013a], it was observed that the textual description and price features of a shopping item in an itemset were not independent to each other. Accordingly, there is a room for more effectively constructing itemsets through exploiting the dependencies between the features when measuring similarities among items. Unlike the previous methods that attempt to select good features for representing the entire observations, *cf*-SIM conditionally determines the utilization of the price feature of an item based on the consideration of its textual description.

Specifically, the conditional utilization of an item's price feature is conducted by the following two steps. In the first step, the state of an item against a query item is determined by focusing only on the similarity between their textual descriptions. If the textual description of an item is similar to that of a query item, its state becomes τ -state, and the state is set to ν -state, otherwise. Therefore, the items containing more similar textual descriptions to that of a query item have higher probability to be in τ -state. Since the items in ν -state for the query item fail to be included in the target itemset for the query item in the first step, they will be ignored in the next step regardless of their prices.

In the second step, the items whose states are determined to be τ -state in the first step are then re-examined with respect to their prices. Two price distributions are employed to fit the prices of those items where one is for the prices of the items in τ -state, and the other is for the prices of the items in ν -state. The final state of an item is then decided by investigating which price distribution is more appropriate to explain the item's price. We remark that, among the items whose textual descriptions are similar to that of a query item, their states are set to τ -state only if their prices are likely to follow the same price distribution as that of the query item. That is, the state of an item is not necessarily set to τ -state for a query item even though it is similar to the query item in terms of either textual description or price in contrast to the conventional models [Basu et al. 2002; MacQueen 1967].

Figure 3 illustrates the process of state estimation for the items against a given query item, and each point corresponds to an item represented by two features, f1 and f2, that are assumed to respectively follow two Gaussian distributions. In this example, we consider one hundred items from four itemsets. The means and standard deviations of f1 associated with the four itemsets, *A*, *B*, *C*, and *D*, are (35, 6), (40, 4), (65, 4), and (70, 4), respectively, and those of f2 are (35, 10), (80, 5), (70, 6), and (35, 8), respectively. The boundaries of actual itemsets are shown in Figure 3 (a), and the desired states of items with respect to the query item are depicted in Figure 3 (b). Note that the goal of the proposed model is to distinguish the items belonging to itemset *A*, in which the query item exists, from the others.



(a) Actual item states

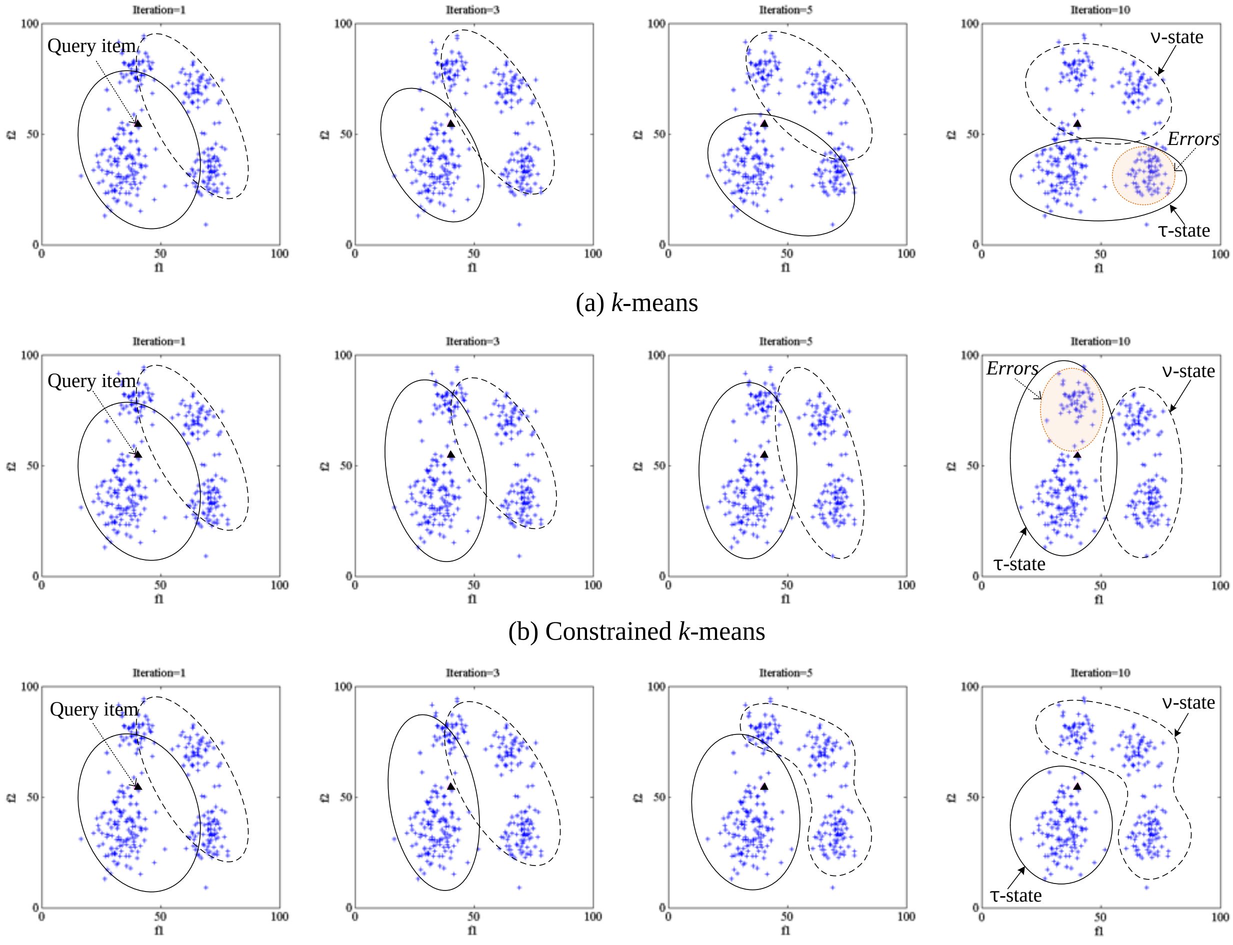
(b) Estimated item states

Figure 3: Graphical example for the concept of the conditional feature utilization by comparing the actual states of items with their estimated states.

When f1 is considered, the items in itemsets *C* and *D* in Figure 3 (a) are determined to be dissimilar to the query item while the items in itemsets *A* and *B* are identified as relevant to the query item. Therefore, the states of items in *C* and *D* are set to ν -state while the items in itemsets *A* and *B* need to be further investigated based on f2. Subsequently, when f2 is considered for the items in *A* and *B*, the states of the items in itemset *A* are identified as τ -state since they are clearly distinguished from the others with respect to f2, as shown in Figure 3 (b).

We describe how our approach is different from the existing ones by comparing our model with *k*-means [MacQueen 1967] and constrained *k*-means [Basu et al. 2002] which are popular unsupervised and semi-supervised learning approaches. Figure 4 shows the comparison results through visualizing itemset construction based on the three models. First, the *k*-means only focuses on partitioning the items in terms of the similarity between items rather than identifying similar items against a query item. That is, it cannot make sure that a query item is in τ -state, as shown in Figure 4 (a). On the other hand, as the constrained *k*-means utilizes the membership of a query item, it ensures that the query item belongs to τ -state.

However, it may result in many errors, caused by the irrelevant items in terms of one of the features, as appeared in Figure 4 (b). Since this model heavily relies on the similarity between an item and a query item with respect to all the item features at the same time, it is possible that the states of items are falsely judged when the items are very similar to a query item in terms of any of the features. Compared to the previous approaches, *cf*-SIM is able to precisely retrieve the target itemset against a query item as shown in Figure 4 (c) owing to the proposed conditional feature utilization scheme. As a semi-supervised model, *cf*-SIM predicts the unknown labels of instances by utilizing the query items as labels. However, it differs from the conventional semi-supervised models in that it uses some of the feature conditionally instead of considering all the features together as in the conventional models.



(a) *k*-means

(b) Constrained *k*-means

(c) The proposed approach (*cf*-SIM)

Figure 4: Comparison results of itemset construction models: *k*-means, constrained *k*-means, and *cf*-SIM.

3.2. Model description

In this section, notations and details of *cf*-SIM are described. We consider a dataset that consists of independently and identically distributed N items, denoted by $I = \{(\mathbf{x}_n, y_n) \mid n = 1, \ldots, N\}$, where $\mathbf{x}_n$ and y_n represent textual description and price of the *n*-th item, respectively. The textual description of the *n*-th item is represented by an *L*-dimensional vector, denoted by $\mathbf{x}_n = < x_{n1} \cdots x_{nL} >$, where x_{nl} is 1, if the *l*-th term appears in the *n*-th item's textual description, and it is 0, otherwise. *L* represents the number of distinct terms found in the items' textual descriptions in a given dataset, and the textual description of an item is expressed as a term vector of length *L* in our model. Therefore, the proposed method has no limitation on the maximum length allowed for a textual description, and it works well as long as there is at least one word in a textual description. We denote a query item by $q = (\mathbf{x}_0, y_0)$ where $\mathbf{x}_0$ and y_0 are its textual description and price, respectively.

We assume that each term occurs independently of the others in the textual description of an item. This assumption, called a bag-of-words approach, has been widely employed in many information retrieval (IR) models [Ferrández 2011], and it has shown successful results in various types of information retrieval research [Amati et al. 2004; Robertson et al. 2000]. The reason for such wide adoption of the independence assumption is because it allows faster computation without significant performance degradation [Sebastiani 2002], compared to the models with the term dependency assumption [Lawrie et al. 2001; Peng et al. 2007]. Although it is an essential assumption for the similarity calculation between textual descriptions in the proposed method, we remark that this assumption can be relaxed by employing another similarity measure with the term dependency assumption. Furthermore, we assume that the price of an item follows a Gaussian distribution, and the textual description and price of an item is conditionally independent as suggested in Agrawal et al. [2011] and Ketter et al. [2007].

Two sets of hidden variables, $D = \{d_n \mid n = 1, \cdots, N\}$ and $P = \{p_n \mid n = 1, \cdots, N\}$, are defined to represent the state of each item based on its textual description and price against a query item, $q = (\mathbf{x}_0, y_0)$, where d_n and p_n are the binary hidden variables that indicate whether or not the *n*-th item is relevant to the query item in terms of its textual description and price, respectively. Use of two hidden variables not only makes the proposed model intuitive through respectively indicating relevance of an item to a query item in terms of two features but also makes derivation of formula easier as described in Equation A.1.

Specifically, d_n is 1, if $\mathbf{x}_n$ is textually relevant to $\mathbf{x}_0$, and 0, otherwise, while p_n is 1, if y_n and y_0 are considered to follow the same price distribution, and 0, otherwise. We remark that a pair of hidden variables is associated to an individual item in *cf*-SIM in contrast to the previous models that consider a single hidden variable associated to a feature across the entire items [Law et al. 2004; Zeng et al. 2009].

As a result, there are four possible combinations of the values of two binary hidden variables for each item. The state of the *n*-th item for a query item is indicated according to the values of its hidden variables, d_n and p_n, as follows.

Case 1) if $d_n = 1$ and $p_n = 1$, the *n*-th item is regarded to be in τ-state, indicating both its textual description and price are relevant to those of the query item.

Case 2) if $d_n = 1$ and $p_n = 0$, the *n*-th item is regarded to be in ν-state since $p_n = 0$ represents that its price is not likely to follow the same distribution as that of the query item.

Case 3) if $d_n = 0$ and $p_n = 1$, the *n*-th item is regarded to be in ν-state since $d_n = 0$ represents that its textual description is irrelevant to that of the query item.

Case 4) if $d_n = 0$ and $p_n = 0$, the *n*-th item is regarded to be in ν-state since $d_n = 0$ and $p_n = 0$ represent that both of its textual description and price are not relevant to those of the query item.

As the price feature of the *n*-th item becomes a determinant only for the items containing similar textual descriptions as mentioned in Section 3.1, the value of p_n contributes to the judgment of its final state only in the cases 1 and 2. On the contrary, when $d_n = 0$, the state of the *n*-th item is simply determined as ν-state regardless of the value of p_n as in the cases 3 and 4. Therefore, when $d_n = 0$, it is no longer necessary to investigate the *n*-th item irrespective of its price. The states of the items are illustrated in Figure 5 by using the same itemsets and items used in Figure 3 where the values of their associated hidden variables according to the four possible cases are annotated.

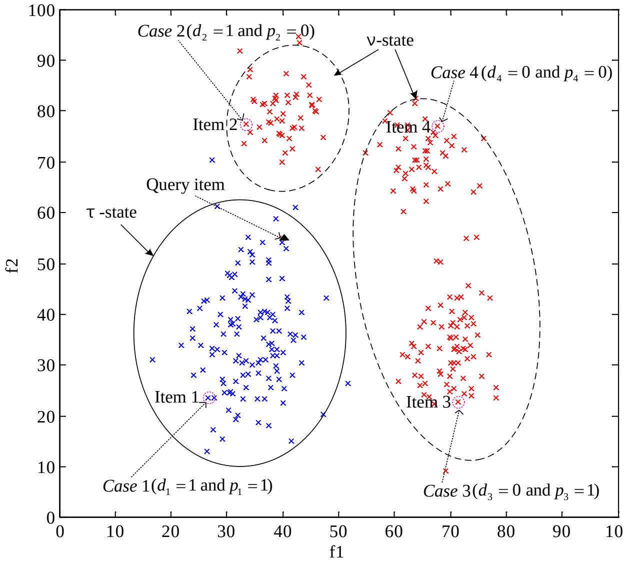


Figure 5: State examples of items according to the possible combinations of their hidden variable values.

To model the probability of an item to be in a particular state based on the possible combinations of values of the hidden variables, we adopt a finite mixture model that takes linear combinations of components to represent the presence of sub-populations by using a set of hidden variables [McLachlan et al. 1988]. In contrast to the original form of a finite mixture model that explains the state of an observation based on the mixture of all the possible states, we modify it to allow each feature of an item to be associated with a hidden variable that indicates whether or not the item feature is utilized for estimating its state. The density function of the *n*-th item is defined as:

$$\begin{aligned} f(\mathbf{x}_n, y_n) &= \sum_{i=0}^{1}\sum_{j=0}^{1} f(\mathbf{x}_n, y_n, d_n = i, p_n = j) \\ &= \sum_{i=0}^{1}\sum_{j=0}^{1} \Pr(d_n = i)\Pr(p_n = j \mid d_n = i) f(\mathbf{x}_n, y_n \mid d_n = i, p_n = j) \\ &= \sum_{i=0}^{1}\sum_{j=0}^{1} \delta_i \pi_{ij} f(\mathbf{x}_n, y_n \mid d_n = i, p_n = j) \end{aligned} \tag{1}$$

where $\Pr(\cdot)$ represents probability, $f(\cdot)$ is a density function, $\delta_i = \Pr(d_n = i)$ such that $0 \le \delta_i \le 1, i = 0,1$ and $\sum_{i=0}^{1} \delta_i = 1$, and $\pi_{ij} = \Pr(p_n = j \mid d_n = i)$ such that $0 \le \pi_{ij} \le 1, i, j = 0,1$ and $\sum_{i=0}^{1} \pi_{ij} = 1$.

In Equation (1), the density function of the *n*-th item, $f(\mathbf{x}_n, y_n)$, is calculated by the mixture of four joint density functions of the item, and its associated hidden variables, $d_n = i$, $i = 0,1$ and $p_n = j$, $j = 0,1$. δ_0 and δ_1 are the coefficients for the textual description, where the former represents the probability that the textual description of an item is irrelevant to that of a query item while the latter represents the probability of the relevant case. Similarly, π_{10} and π_{11} are the coefficients for the price where the former represents the probability that the price of an item follows the same price distribution as that of a query item while the latter represents that of the opposite case when the textual description of the item is determined to be relevant. Since p_n is not considered for the *n*-th item corresponding to the cases 3 or 4, three item density functions are defined for the following pairs of hidden variables, which are (i) $d_n = 1$ and $p_n = 1$, (ii) $d_n = 1$ and $p_n = 0$, and (iii) $d_n = 0$ regardless of p_n.

For the *n*-th item associated with $d_n = 1$ and $p_n = 1$ (*Case* 1), its density is calculated by using the set of parameters, $\theta_{11} = \{\omega_l, \mu, \sigma \mid l = 1,...,L\}$ where ω_l is the probability of the *l*-th term's appearance in the textual description of an item in τ-state, and μ and σ represent the mean and the standard deviation of the prices of the items in τ-state, respectively. Accordingly, the density function for the *n*-th item in τ-state, given the set of parameters, θ_{11}, is obtained by:

$$f(\mathbf{x}_n, y_n \mid \theta_{11}) = \Pr(\mathbf{x}_n \mid \theta_{11}) f(y_n \mid \theta_{11}) \tag{2}$$

where $\Pr(\mathbf{x}_n \mid \theta_{11}) = \prod_{l=1}^{L} (\omega_l)^{x_{nl}} (1-\omega_l)^{(1-x_{nl})}$ and $f(y_n \mid \theta_{11}) = N(y_n \mid \mu, \sigma^2) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left\{-\frac{(y_n - \mu)^2}{2\sigma^2}\right\}$.

For the *n*-th item associated with $d_n = 1$ and $p_n = 0$ (*Case* 2), its item density is evaluated by using the set of parameters, $\theta_{10} = \{\omega_l, \mu', \sigma' \mid l = 1,...,L\}$ where μ' and σ' represent the mean and the standard deviation of the prices of the items in ν-state, respectively. The density function for the *n*-th item in ν-state under $d_n = 1$ and $p_n = 0$, given the set of parameters, θ_{10}, is calculated as follows.

$$f(\mathbf{x}_n, y_n \mid \theta_{10}) = \Pr(\mathbf{x}_n \mid \theta_{10}) f(y_n \mid \theta_{10}) \tag{3}$$

Finally, for the *n*-th item associated with $d_n = 0$ (*Cases* 3 and 4), its density is determined by using the set of parameters, $\theta_0 = \{\omega_l', \mu', \sigma' \mid l = 1,...,L\}$, where ω_l' is the probability of the *l*-th term's appearance in the textual description of an item in ν-state. The density function for the *n*-th item in ν-state under $d_n = 0$ regardless of p_n, given the set of parameters, θ_0, is computed as:

$$f(\mathbf{x}_n, y_n \mid \theta_0) = \Pr(\mathbf{x}_n \mid \theta_0) f(y_n \mid \theta_0) \tag{4}$$

3.3. Parameter estimation

The parameter set of *cf*-SIM, $\Theta = \{\theta_{11}, \theta_{10}, \theta_0\}$, is estimated so that it can maximize the complete data log-likelihood function over the entire items and their hidden variables, defined in Equation (5) (See Appendix A for details).

$$\ln f(I, D, P | \Theta) = \ln \prod_{n=0}^{N} f(\mathbf{x}_n, y_n, d_n = i, p_n = j \mid \Theta) \tag{5}$$

The parameters that maximize Equation (5) cannot be directly obtained since the sets of hidden variables are unknown. Accordingly, we employ an EM algorithm which is one of the widely used methods to find the maximum likelihood solutions for models having hidden variables through maximizing the conditional expectation of their complete data log-likelihood functions [Dempster et al. 1977]. In EM algorithm, two steps, the expectation step and the maximization step, are iterated until a certain criterion is satisfied. We denote a variable pertaining to the *m*-th iteration of EM algorithm by using (*m*) as its superscript.

In the expectation step, the proposed EM algorithm computes the probability that $d_n = i$, $i = 0,1$, and $p_n = j$, $j = 0,1$, at the *m*-th iteration, given the *n*-th item and the parameters by fixing $\Theta^{(m)}$, as defined in Equation (6) (See Appendix B for details).

$$\rho_n^{(m)}(i, j) = \Pr(d_n = i, p_n = j \mid \mathbf{x}_n, y_n, \Theta^{(m)}) \tag{6}$$

Finally, in the maximization step, the parameters are updated by fixing $\rho_n^{(m)}(i, j)$, $n = 0,\ldots,N$, $i, j = 0,1$, and calculated by using Equation (6).

$$\Theta^{(m+1)} = \arg\max_{\Theta} Q(\Theta, \Theta^{(m)}) \tag{7}$$

where $Q(\Theta, \Theta^{(m)})$ is the expected value of the complete data log-likelihood function defined in Equation (5) (See Appendix C for details).

4. Target itemset retrieval

Figure 6 summarizes the target itemset retrieval algorithm under the proposed framework. A query item and a set of items of which the itemset memberships are sought to be identified are given as an input of the algorithm. The parameters of *cf*-SIM such as the term appearance probabilities, and the mean and variance of prices are initialized according to the textual description and price of the query item (line 1 in Figure 6). Then, the parameters of *cf*-SIM and the hidden variables for each item are iteratively estimated (lines 2 to 4). In line 3, the values of each item's hidden variables are estimated by using Equation (6).

Specifically, the estimation is carried out by using the marginal density function of the item shown in Equation (1) as well as the conditional density functions presented in Equations (2), (3), and (4). In line 4, through using the estimates in line 3, the parameters necessary for the next iteration are updated based on Equation (7). The algorithm repeats lines 3 and 4 until the likelihood obtained by using Equation (5) converges.

Based on the estimated values for d_n and p_n of each item, the membership of an item against the query item are determined in lines 5 and 6. According to $\rho_n(i, j)$ that represents the probability that the *n*-th item corresponds to the case $d_n = i$ and $p_n = j$ where $i = 0,1$ and $j = 0,1$, the algorithm determines the state of the *n*-th item to be τ-state, if $\rho_n(1,1) > \rho_n(0,0) + \rho_n(1,0) + \rho_n(0,1)$, and ν-state, otherwise.

$\rho_n(1,1)$ is the probability that the *n*-th item belongs to the target itemset, while $\rho_n(0,0)$, $\rho_n(1,0)$, and $\rho_n(0,1)$ are the probabilities that the *n*-th item belongs to one of the other non-relevant itemsets. We compare the probabilities in a binary manner so that only when the probability of being in the target itemset is higher than that of the other cases, the *n*-th item is judged to be a member of the target itemset. Finally, the algorithm returns the item membership results for the set of items (line 7).

```
Input: query item q, and set of items, D.
Output: memberships of items in D against q.

1: Initialize the parameters of cf-SIM, $\Theta$, according to q.
2: Repeat until the likelihood value of cf-SIM, shown in Equation (5), is converged.
3:     Estimate the values of each item's hidden variables, $d_n$ and $p_n$, by using Equation (6).
4:     Update the parameters of cf-SIM based on the estimated values by using Equation (7).
5: For each item in D
6:     Judge its membership to be the target itemset of q if the probability of $d_n = p_n = 1$ is
           greater than the other cases.
7: Return the memberships of items in D.
```

Figure 6: Target itemset retrieval algorithm under *cf*-SIM.

5. Experiments

5.1. Datasets

For experimentation, two real-world datasets were used in this research to show the robustness of *cf*-SIM. Dataset I is obtained from an online shopping service, Best Buyer (http://www.bb.co.kr), which is one of the most popular price comparison shopping services in Korea and provides itemsets by gathering items from over 140 distinct online shopping malls. Dataset II is constructed from a popular price comparison service in UK, called idealo (http://www.idealo.co.uk), and it consists of items from over 180 distinct online shopping malls. Table 2 summarizes the datasets involved in the experiments.

Table 2: Statistics of the two datasets considered.

Attributes	Dataset I	Dataset II
The number of items	14,815	22,497
The number of itemsets	200	189
The average number of items in an itemset	74	191
The (min, max) number of items in an itemset	(50, 141)	(50, 1632)
The average length of descriptions	612	84
The average price	32,010 (KRW)	137.521 (Pound)

Since the collected items from these services were from a number of shopping malls, the datasets can be considered to represent a significant portion of items available in the market [Koças 2005]. For each dataset, we randomly selected itemsets each of which consists of more than 50 items, and as a result, the total 14,815 items for dataset I and 22,497 items for dataset II were used.

To evaluate the effectiveness of target itemset retrieval tasks, we utilized available itemset membership judgment results as follows: For dataset I, we used the results previously made by about 150 human experts for the purpose of providing high quality price comparison service. Unfortunately, the quality of itemset membership judgments for dataset II is not known, which implies potential errors in the membership labels for the collected items in dataset II. We remark that only the information on the query items is utilized during the parameter estimation of *cf*-SIM since the memberships are assumed to be unknown in this paper. The itemset memberships of the collected items were used only in the process of evaluating the performances of the proposed model.

5.2. Experiment Settings

The parameters of *cf*-SIM, $\Theta = \{\theta_{11}, \theta_{10}, \theta_0\}$, as well as the textual description coefficients, $\delta_i,\ i = 0,1$, and the price coefficients, $\pi_{ij},\ i, j = 0,1$, for each state were estimated as follows. For query item, $q = (\mathbf{x}_0, y_0)$, the parameters for the items in τ -state, θ_{11}, were set as $\omega_l = x_{0l}$, $l = 1,...,L$, $\mu = y_0$, and σ to be the standard deviation of the prices of the entire items in the dataset. On the other hand, for the parameters for items in ν -state, θ_{10} and θ_0, we randomly set ω_l and ω_l' to 0 or 1, and $\mu^{'}$ was set to be a random value between the maximum and minimum of item prices while $\sigma^{'}$ was set in the same way as σ. Moreover, the other parameters, $\delta_i,\ i = 0,1$, and $\pi_{ij},\ i, j = 0,1$, were also set to random values between 0.0 and 1.0.

We enforced the standard deviations, σ and $\sigma^{'}$, to have a limiting constant between 0.15 and 0.3 to prevent a singularity problem that causes too small variances in a finite mixture model by following the suggestion in Reynolds et al. [1995]. To avoid an overflow problem during calculation of probability, we set the minimum probability for the textual description of each item that appears in Equations (1) to (4), to be 2.225^{-308}. At each iteration of the EM algorithm, the parameters of *cf*-SIM were estimated by using Equation (A.1) and updated by using Equations (C.3) to (C.13) until the difference between the log-likelihood values of two successive EM iterations was less than 10^{-8}, as proposed by Dempster et al. [1977].

For the purpose of comparison, an unsupervised model, named UIM, as well as two semi-supervised models, respectively named SIM1 and SIM2, were considered in the experiments. UIM was implemented by adopting *k*-means, one of the popular clustering methods successfully applied in various problems such as document clustering analysis [Mahdavi et al. 2009], image segmentation [Ng et al. 2006], and outlier detection [Kyung et al. 2007].

On the other hand, SIM1 was implemented by use of the constrained *k*-means, an extended version of *k*-means that utilizes some known membership information on the observations [Basu et al. 2002]. Since there has been no semi-supervised method proposed for the shopping itemset construction problem to the best of the authors' knowledge, we employed a semi-supervised support vector machine (SSVM) as SIM2, which is designed to predict unknown labels based on a small fraction of labeled observations by using linear kernels [Sindhwani et al. 2006].

SSVM is known for its impressive results in many application scenarios involving both textual and non-textual features [Hoi et al. 2008; Wang et al. 2010]. Similar to *cf*-SIM, judgment made by UIM, SIM1, and SIM2 is binary since the number of itemsets is assumed to be unknown. For UIM and SIM1, the Euclidean distance was employed as a similarity measure between items. For SIM2, we used a semi-supervised SVM library, called SVMlin, which bases on a linear kernel and a deterministic annealing method to estimate parameters [Sindhwani 2006].

Table 3: Confusion matrix presenting the number of items according to their actual and estimated states against a query item.

		Estimated state	
		τ -state	ν -state
Actual state	τ -state	*TP*	*FN*
	ν -state	*FP*	*TN*

For each query item, performances of the itemset retrieval models were measured based on precision and recall which are widely adopted metrics to evaluate how precisely a model estimates the actual states of observations in many applications including information retrieval, classification, and clustering analysis [Jardine et al. 1971; Modha et al. 2003]. Precision represents the fraction of correctly estimated items among the items predicted as τ -state whereas recall indicates the fraction of items correctly estimated as τ -state among the entire items in τ -state actually. The precision and the recall of a model for a given query item are respectively calculated by using Equations (8) and (9), where TP, FN, FP, and TN respectively stand for true positive, false negative, false positive, and true negative, as defined in Table 3.

$$P = \frac{TP}{TP + FP} \tag{8}$$

$$R = \frac{TP}{TP + FN} \tag{9}$$

In addition, we used F1 metric that summarizes the precision and recall into a single measure as their harmonic mean [Dai et al. 2013]. The F1 of a model item is obtained as follows.

$$F = \frac{2 \times P \times R}{P + R} \tag{10}$$

5.3. Experiment Results

Based on the datasets, the effectiveness of *cf*-SIM was examined in terms of precision, recall, and F1. To show the robustness of the proposed model, various configurations regarding the number of items per itemset, denoted as *K*, and the number of itemsets, denoted as *M*, were investigated, and we have repeatedly carried out the experiments by using each of the collected items as a query item. As *M* becomes bigger, identifying the item states becomes more difficult since the larger number of itemsets leads to the larger number of possible distributions for textual descriptions and prices of items.

Before comparing *cf*-SIM with the others, the effects of individual features were investigated. We denote the model that uses only the textual description feature as *cf*-SIM (T) and the model that utilizes only the price feature as *cf*-SIM (P). Table 4 presents that *cf*-SIM performed best in terms of precision and F1 for both datasets, followed by *cf*-SIM (T) and *cf*-SIM (P). While *cf*-SIM (T) achieved better recall than *cf*-SIM, it yielded quite low precision, resulting in much lower F1 performance than *cf*-SIM. Moreover, *cf*-SIM (P) showed the worst performances for all the measures considered, suggesting that the price feature alone is not sufficient to judge an itemset. These imply that incorporating both of the textual description and price as features is beneficial to enhance the itemset retrieval performance.

Table 4: Performance comparison results with respect to the features employed by *cf*-SIM for two datasets.

Dataset	Dataset I			Dataset II		
Methods	*cf*-SIM	*cf*-SIM (T)	*cf*-SIM (P)	*cf*-SIM	*cf*-SIM (T)	*cf*-SIM (P)
Features	Both	Textual description	Price	Both	Textual description	Price
Precision	0.57	0.20	0.19	0.48	0.12	0.51
Recall	0.77	0.93	0.48	0.89	0.97	0.43
F1	0.61	0.32	0.10	0.53	0.18	0.13

The graphs in Figures 7 and 8 show the performance comparison results between *cf*-SIM and the alternatives, UIM, SIM1 and SIM2, when M was varied from 10 to 50 with an increment of 10 under $K = 50$ and $K = 100$. Overall, *cf*-SIM outperformed the alternatives for most combinations of K and M in terms of F1 across datasets. While SIM2 showed better performance than *cf*-SIM when $M = 10$ and $K = 100$ in dataset I, it underperformed compared to *cf*-SIM under the rest of the settings, and its performances tend to rapidly degrade as M increases under both $K = 50$ and $K = 100$. This indicates that *cf*-SIM is more robust than SIM2 for the itemset retrieval tasks especially in case of dealing with many itemsets.

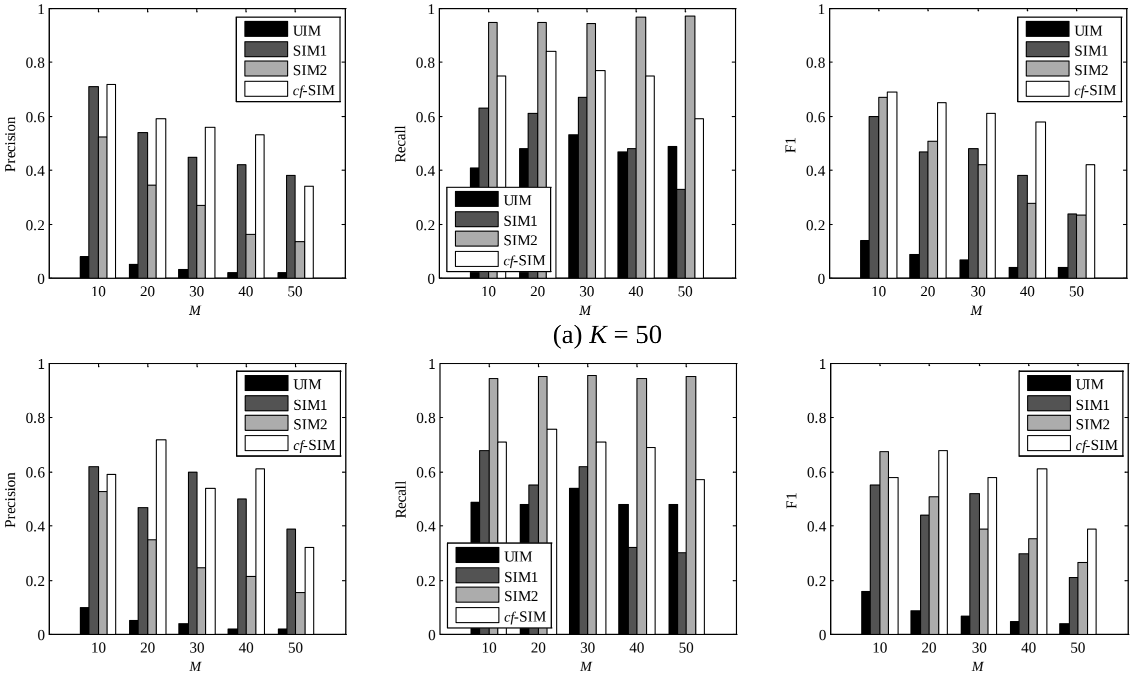
Specifically, for dataset I, the average F1 results achieved by *cf*-SIM, UIM, SIM1, and SIM2 were 0.58, 0.08, 0.42, and 0.43, respectively. Their respective average precisions were 0.57, 0.05, 0.48, and 0.30 while average recalls were 0.78, 0.48, 0.61, and 0.95. For dataset II, the average F1 values by *cf*-SIM, UIM, SIM1, and SIM2 were 0.53, 0.08, 0.43, and 0.37, respectively. Their average precisions were 0.66, 0.06, 0.34, and 0.24, respectively while average recalls were 0.89, 0.32, 0.84, and 0.99. These suggest that the conditional utilization of the price feature depending on each item's textual description was effective for identifying the itemset memberships.

In terms of precision, *cf*-SIM yielded better results than the alternatives for most combinations of K and M. For dataset I, *cf*-SIM outperformed the others except the following cases: (i) $M = 50$ and $K = 50$, (ii) $M = 10$ and $K = 100$, (iii) $M = 30$ and $K = 100$, and (iv) $M = 50$ and $K = 100$ in which it was outperformed by SIM1. Such unsatisfactory results can be attributed to the trade-off relationship between precision and recall [Buckland et al. 1994] since *cf*-SIM focuses more on reducing false-negative errors. On the other hand, *cf*-SIM showed higher precision results than the others for all settings in dataset II. The poor precision performances of UIM can be explained by the fact that it only tries to cluster items into two itemsets without investigating which itemset represents τ-state as it does not use the membership information of a query item for retrieving a target itemset.

Interestingly, SIM2 outperformed the others in recall while its precision values were quite low across all the experiment settings considered. This indicates that SIM2 frequently tends to judge items to be in a target itemset for a query item without sufficient evidences. For dataset II, when $K = 100$, the recall performances of *cf*-SIM, SIM1, and SIM2 were over 0.9. We conjecture that such high recall results achieved for dataset II are attributed to the existence of a substantial amount of erroneous items in dataset II whose itemset memberships are misjudged.

Such incorrect judgments are possibly due to the aforementioned ambiguity problem which induces mistakes in determining the itemset memberships of items with similar descriptions during the itemset membership judgment process by a human expert or automated program. For an itemset with many incorrectly judged items, the price feature of an item loses its discriminant power since it is hard to properly fit the item prices in the itemset to a right price distribution, and the itemset construction models will have to determine the itemset memberships only by textual description features, resulting in higher recalls than the case of dataset I in which the itemset memberships were carefully judged by a number of human experts.

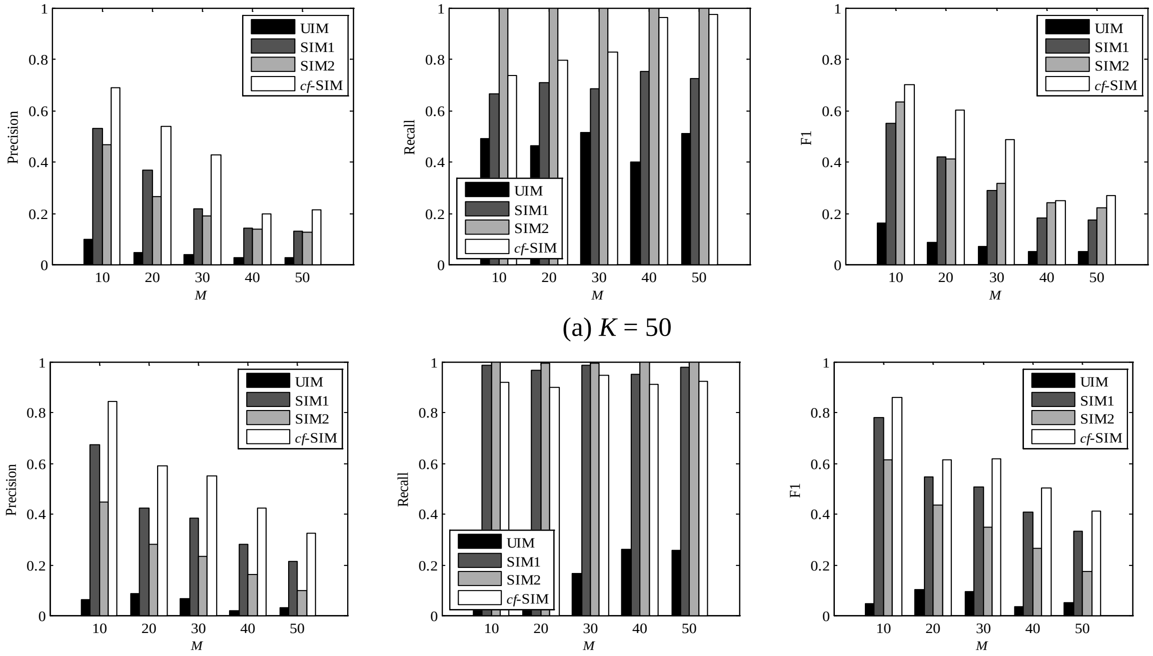
In addition, we conducted the experiments to show the effectiveness of *cf*-SIM along with the various prices of a query item. As noted in Section 3.1, it becomes more difficult to identify the items in τ-state as price differences between the query item and the items in ν-state become smaller. To indicate the price difference between a query item and an item to be retrieved, we define the relative location of a query item's price among those of the items in τ-state, denoted as $r = (max - y_0)/(max - min)$ where *max* and *min* respectively represent the maximum and the minimum prices among the prices of the items in τ-state for the query item. Figures 9 and 10 depict the improvement ratios of *cf*-SIM over the alternatives along with r for two datasets. The improvement ratio was calculated as the performance difference between *cf*-SIM and each alternative over the performance of *cf*-SIM where the performance corresponds to each metric considered. The improvements obtained by *cf*-SIM became bigger as r increased in most cases, implying that *cf*-SIM is more effective than the others as the price of a query item is located further from the centroid of the item prices of the target itemset. That is, while all the models considered are prone to fail to retrieve extremely priced items, *cf*-SIM worked relatively better in identifying them.



(a) *K* = 50

(b) *K* = 100

Figure 7: Performance comparison results for UIM, SIM, and *cf*-SIM in terms of precision, recall, and F1 when varying *M* (Dataset I).



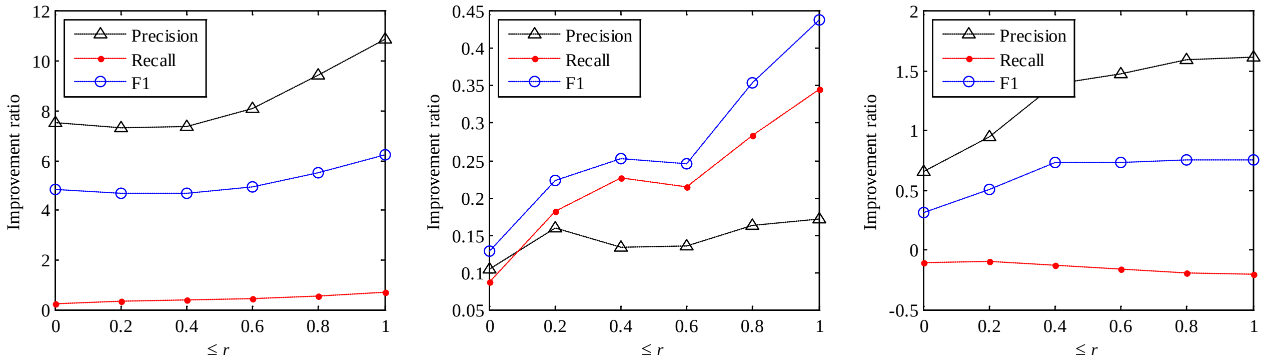
(a) *K* = 50

(b) *K* = 100

Figure 8: Performance comparison results for UIM, SIM, and *cf*-SIM in terms of precision, recall, and F1 when varying *M* (Dataset II).

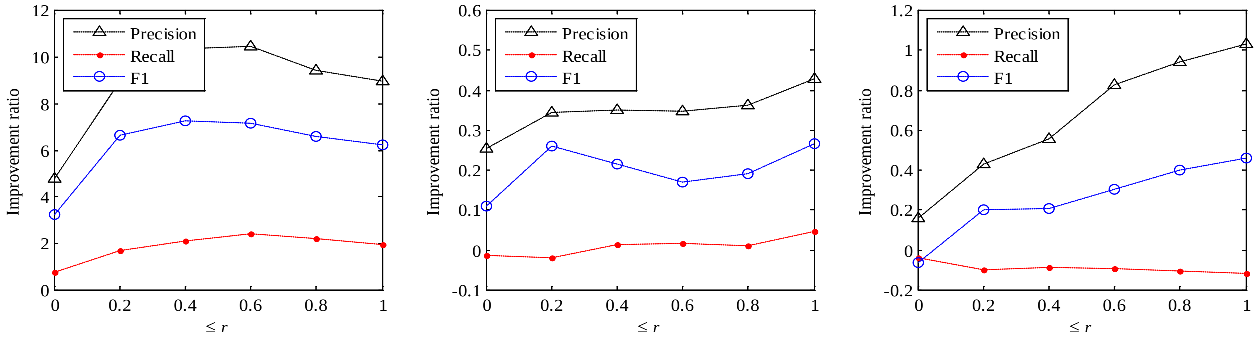
Figures 9 (a) and 10 (a) show that *cf*-SIM performed better than UIM, and the improvements in F1 over UIM were greater than 500% for two datasets when averaging on *r*. Although *cf*-SIM failed to significantly exceed UIM in recall, the improvements over UIM in precision were noticeable. As shown in Figures 9 (b) and 10 (b), *cf*-SIM respectively outperformed SIM1 by 13% and 9% for datasets I and II in terms of F1 when *r* = 0 while the respective improvements were more than 44% and 27% when *r* = 1.

Furthermore, Figures 9 (c) and 10 (c) indicate that *cf*-SIM produced better performances in F1 and precision than SIM2 as *r* increases. The improvements of *cf*-SIM over SIM2 in F1 respectively were 63% and 25% for datasets I and II when averaging on *r*. Although *cf*-SIM achieved lower recall than SIM2 as *r* increases, it resulted in much better F1 than SIM2. Compared to other research results that reported 10 to 20% of performance improvements based on semi-supervised methods [Beitzel et al. 2005; Wu et al. 2012; Zheng et al. 2013], the proposed model appears to have made successful improvements. All the statistical tests were carried out under *p*-value < 0.01 of a paired *t*-test.



(a) UIM (b) SIM1 (c) SIM2

Figure 9: Improvements of *cf*-SIM over the alternatives (Dataset I).



(a) UIM (b) SIM1 (c) SIM2

Figure 10: Improvements of *cf*-SIM over the alternatives (Dataset II).

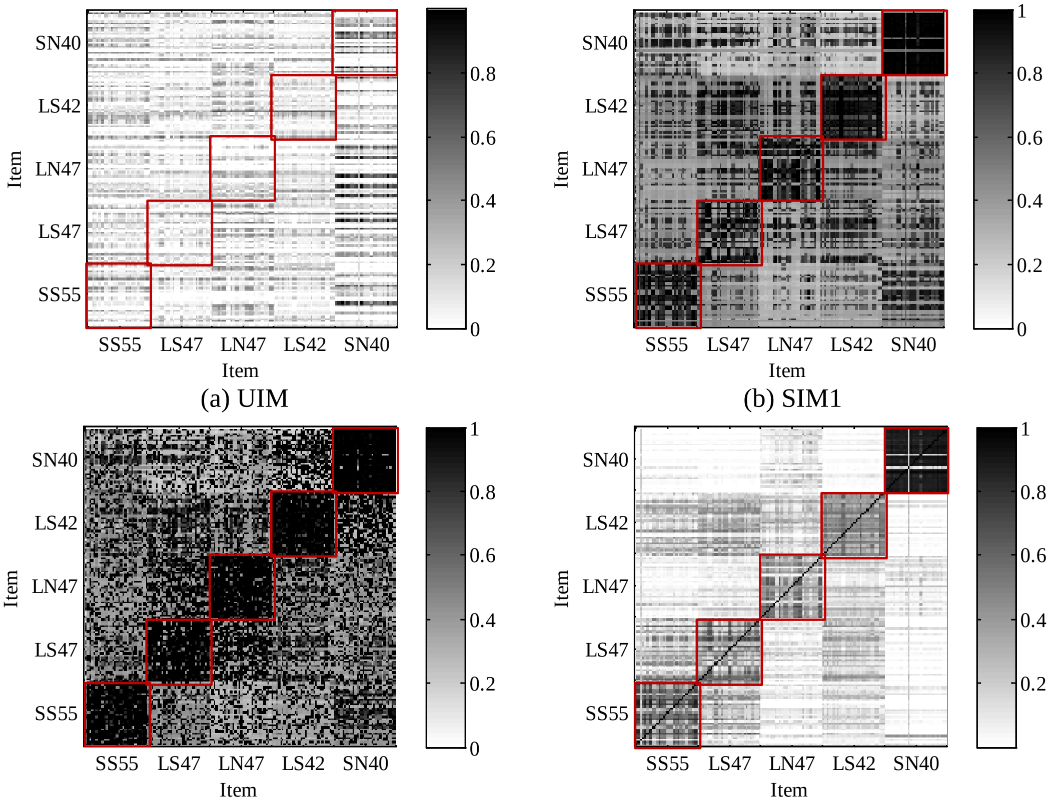
In summary, *cf*-SIM showed satisfactory itemset construction results consistently for both datasets I and II, suggesting its robustness against different datasets. In particular, *cf*-SIM performed relatively well even with dataset II that appears to contain some noisy labels. Moreover, considering the fact that the textual description languages and price units were different for datasets I and II, we believe that *cf*-SIM is a viable method for international application.

Finally, for the five itemsets selected in dataset I (presented in Table 5) we visualized the comparison results for the models considered as heat maps that present how a model successfully identified the actual states of items. For each itemset, 30 items were randomly selected for this experiment. In Figure 11, the brightness of each cell for a pair of items represents the probability of how likely they are in the same state. As shown in Figure 11 (a), UIM failed to correctly retrieve the target itemset for a query item in most cases, and the precision, recall, and F1 of UIM were respectively 0.12, 0.43, and 0.18 on the average.

While SIM1 shows better results than UIM, its performance is still unsatisfactory as shown in Figure 11 (b). The precision, recall, and F1 of SIM1 were 0.72, 0.63, and 0.64, respectively. Figure 11 (c) indicates that SIM2 was successful in detecting the items for a target itemset but it showed quite poor performances for identifying the items that are not in the target itemset, resulting in a large number of false positive errors. The precision, recall, and F1 of SIM2 respectively were 0.43, 0.98, and 0.60. Finally, *cf*-SIM achieved satisfactory results through effectively distinguishing the target itemsets for query items as shown in Figure 11 (d). The precision, recall, and F1 of *cf*-SIM were 0.71, 0.78, and 0.76, respectively.

Table 5: Statistics of five example itemsets.

Name	Itemset	L	Price (KRW)			
			Min	Max	Average	Standard deviation
SS55	Samsung 3D Smart TV 55'	577	2,290,350	3,011,450	2,633,840	1,92,164
LS47	LG 3D Smart TV 47'	601	1,288,310	2,167,110	1,505,653	244,793
LN47	LG 3D TV 47'	553	1,676,950	4,000,000	1,950,369	302,366
LS42	LG 3D Smart TV 42'	690	930,340	1,520,790	1,099,284	162,346
SN40	Samsung TV 40'	447	790,000	1,428,330	1,004,933	156,791



(a) UIM

(b) SIM1

(c) SIM2

(d) *cf*-SIM

Figure 11: Heat maps for the five example itemsets (the boxes indicate the items in correct τ -state for their respective itemsets).

6. Implications and Limitations

Through utilizing *cf*-SIM in practice, service providers are expected to significantly reduce time and cost for itemset construction, which leads to better service quality and enhanced customer satisfaction. Since our model does not require additional prior information such as the itemset memberships and the predefined number of itemsets, more practical deployments of *cf*-SIM are expected compared to the conventional models. Moreover, based on the fact that *cf*-SIM works well even when the number of available items is limited, it appears to be useful for small and medium sized online shopping services. At the same time, customers are likely to be provided with more correct itemsets when using *cf*-SIM, which will contribute to enhancing customer's trust, eventually promoting more sales.

Owing to the ability of *cf*-SIM that retrieves itemsets without relying on the aforementioned additional information, the target itemset for a query item can be easily retrieved upon a customer's request. Furthermore, *cf*-SIM can also be used to filter out potential fraud items whose prices are much lower than those of the normal items [Kim et al. 2013a] since the target itemset identified by using *cf*-SIM are not likely to include the items with very low prices compared to those of others in the market. As a result, extremely priced items due to the sellers' pricing strategies and promotions are not likely to be included in the retrieved itemset under the proposed framework. Nevertheless, when such extreme cases are frequently observed, our model will be able to retrieve such items through additional learning of the probability distributions for the items based on those new observations.

There still exist rooms for further enhancement of the performance of *cf*-SIM. First, some supplementary information can be incorporated to achieve better itemset retrieval results. For instance, information related to manufacturers, sellers, and transaction logs that provide valuable evidences for specifying the product types of items can be further utilized to improve *cf*-SIM. Second, through estimating the number of itemsets, *cf*-SIM will be able to infer the item states more sophisticatedly through enriching the parameters. Finally, there is a possibility to better retrieve extremely priced non-fraud items through incorporating information obtained from social network services in addition to the elementary item features.

7. Conclusions

In this research, we presented a novel model, *cf*-SIM, which aims to retrieve target itemsets against a query item through conditionally utilizing the price feature of an item to address the ambiguity and indeterminacy problems caused by the items with similar textual descriptions and prices. Specifically, we calculated the probability that an item is in the target itemset for a query item by using a finite mixture model and proposed a modified version of EM algorithm for parameter estimation of *cf*-SIM. The experiment results based on two real-world datasets show that *cf*-SIM performs better than the other alternatives considered. As future work, we plan to explore additional item features that can help *cf*-SIM more precisely estimate the item states, and further enhance the parameter estimation process for the proposed model through incorporating prior distributions of textual descriptions and prices.

Acknowledgment

This work was supported by the National Research Foundation of Korea (NRF) grants funded by the Korea government (MSIP) (No. 2013R1A2A2A03013947 and 2011-0030814) as well as the Korea Government (MEST) (No. 2012R1A6A3A01038452), and partly by the human behavior and decision process simulation technology research funded by the agency for defense development.

REFERENCES

Abbott, A.A. and I. Watson, "Ontology-Aided Product Classification: A Nearest Neighbour Approach," *Lecture Notes in Computer Science*, Vol. 6880:348-362, 2011.

Abels, S. and A. Hahn, "Empirical Study on Usage of Electronic Product Classification Systems in E-Commerce Organizations in Germany," *Journal of Electronic Commerce in Organizations*, Vol. 4, No. 1:33-47, 2006.

Agrawal, R., S. Ieong, and R. Velu, "Ameliorating Buyer's Remorse," *Proceedings of the International Conference on Knowledge Discovery and Data Mining*, pp. 351-359, San Diego, CA, USA, 2011.

Amati, G., C. Carpineto, and G. Romano, "Comparing Weighting Models for Monolingual Information Retrieval," *Lecture Notes in Computer Science*, Vol. 3237:310-318, 2004.

Basu, S., A. Banerjee, and R.J. Mooney, "Semi-Supervised Clustering by Seeding," *Proceedings of the International Conference on Machine Learning*, pp. 27-34, Sydney, Australia, 2002.

Beitzel, S.M., E.C. Jensen, O. Frieder, D.D. Lewis, A. Chowdhury, and A. Kolcz, "Improving Automatic Query Classification Via Semi-Supervised Learning," *Proceedings of the IEEE International Conference on Data Mining*, pp. 8-15, Houston, TX, USA, 2005.

Benjelloun, O., H. Garcia-Molina, D. Menestrina, Q. Su, S.E. Whang, and J. Widom, "Swoosh: A Generic Approach to Entity Resolution," *The International Journal on Very Large Data Bases*, Vol. 18, No. 1:255-276, 2009.

Bergamaschi, S., F. Guerra, and M. Vincini, "A Data Integration Framework for E-Commerce Product Classification," *Lecture Notes in Computer Science*, Vol. 2342:379-393, 2002.

Bilenko, M., S. Basil, and M. Sahami, "Adaptive Product Normalization: Using Online Learning for Record Linkage in Comparison Shopping," *Proceedings of the International Conference on Data Mining*, pp. 58-65, Houston, TX, USA, 2005.

Buckland, M.K. and F.C. Gey, "The Relationship between Recall and Precision," *Journal of the American Society for Information Science*, Vol. 45, No. 1:12-19, 1994.

Dai, Z., A. Sun, and X.Y. Liu, "Crest: Cluster-Based Representation Enrichment for Short Text Classification," *Lecture Notes in Computer Science*, Vol. 7819:256-267, 2013.

Dempster, A.P., N.M. Laird, and D.B. Rubin, "Maximum Likelihood from Incomplete Data Via the Em Algorithm," *Journal of the Royal Statistical Society. Series B (Methodological)*, Vol. 39, No. 1:1-38, 1977.

Ding, Y., M. Korotkiy, B. Omelayenko, V. Kartseva, V. Zykov, M. Klein, E. Schulten, and D. Fensel, "Goldenbullet: Automated Classification of Product Data in E-Commerce," *Proceedings of the International Conference on Business Information Systems*, Poznan, Poland, 2002.

Ferrández, A., "Lexical and Syntactic Knowledge for Information Retrieval," *Information Processing and Management*, Vol. 47, No. 5:692-705, 2011.

Garfinkel, R., R. Gopal, B. Pathak, and F. Yin, "Shopbot 2.0: Integrating Recommendations and Promotions with Comparison Shopping," *Decision Support Systems*, Vol. 46, No. 1:61-69, 2008.

Geng, X., X. Fan, J. Bian, X. Li, and Z. Zheng, "Optimizing User Exploring Experience in Emerging E-Commerce Products," *Proceedings of the International Conference Companion on World Wide Web*, pp. 23-32, Lyon, France, 2012.

Grira, N., M. Crucianu, and N. Boujemaa, "Unsupervised and Semi-Supervised Clustering: A Brief Survey," *A Review of Machine Learning Techniques for Processing Multimedia Content*, Vol. 1:9-16, 2004.

Hoi, S.C., R. Jin, J. Zhu, and M.R. Lyu, "Semi-Supervised Svm Batch Mode Active Learning for Image Retrieval," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1-7, Anchorage, AK, USA, 2008.

Jardine, N. and C.J. van Rijsbergen, "The Use of Hierarchic Clustering in Information Retrieval," *Information Storage and Retrieval*, Vol. 7, No. 5:217-240, 1971.

Kannan, A., I.E. Givoni, R. Agrawal, and A. Fuxman, "Matching Unstructured Product Offers to Structured Product Specifications," *Proceedings of the International Conference on Knowledge Discovery and Data Mining*, pp. 404-412, San Diego, CA, USA, 2011a.

Kannan, A., P.P. Talukdar, N. Rasiwasia, and Qifa Ke, "Improving Product Classification Using Images," *Proceedings of the IEEE International Conference on Data Mining*, pp. 310-319, San Diego, CA, USA, 2011b.

Ketter, W., J. Collins, M. Gini, P. Schrater, and A. Gupta, "A Predictive Empirical Model for Pricing and Resource Allocation Decisions," *Proceedings of the International Conference on Electronic Commerce*, pp. 449-458, Minneapolis, MN, USA, 2007.

Kim, K, B.S. Chung, J.Y. Jung, and J. Park, "Revenue Maximizing Itemset Construction for Online Shopping Services," *Industrial Management and Data Systems*, Vol. 113, No. 1:96-116, 2013a.

Kim, K., Y. Choi, and J. Park, "Pricing Fraud Detection in Online Shopping Malls Using a Finite Mixture Model," *Electronic Commerce Research and Applications*, Vol. 12, No. 3:195-207, 2013b.

Kim, K., B.S. Chung, Y. Yang, J.Y. Jung, and J. Park, "A Cost-Conscious Item Grouping Method for Online Shopping Services," *Telecommunications Review*, Vol. 22, No. 2:323-335, 2012a.

Kim, K., B.S. Chung, Y. Yang, and J. Park, "A Feature-Based Item Ranking Approach to Itemset Construction in Price Comparison Shopping Services," *Proceedings of the International Conference on Computer, Networks, Systems, and Industrial Applications*, pp. 3-8, Jeju Island, Korea, 2012b.

Kim, Y., T. Lee, J. Chun, and S. Lee, "Modified Naïve Bayes Classifier for E-Catalog Classification," *Lecture Notes in Computer Science*, Vol. 4055:246-257, 2006.

Kim, Y., T. Lee, S. Lee, and J.H. Park, "Exploiting Attribute-Wise Distribution of Keywords and Category Dependent Attributes for E-Catalog Classification," *Lecture Notes in Computer Science*, Vol. 5226:985-992, 2008.

Kirsten, T., L. Kolb, M. Hartung, A. Gross, H. Köpcke, and E. Rahm, "Data Partitioning for Parallel Entity Matching," *Proceedings of the Conference on Very Large Data Bases*, Vol. 3, No. 2, Singapore, 2010.

Koças, C., "A Model of Internet Pricing under Price-Comparison Shopping," *International Journal of Electronic Commerce*, Vol. 10, No. 1:111-134, 2005.

Kopcke, H., A. Thor, and E. Rahm, "Learning-Based Approaches for Matching Web Data Entities," *IEEE Internet Computing*, Vol. 14, No. 4:23-31, 2010.

Law, M.H.C., M.A.T. Figueiredo, and A.K. Jain, "Simultaneous Feature Selection and Clustering Using Mixture Models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 26, No. 9:1154-1166, 2004.

Lawrie, D., W.B. Croft, and A. Rosenberg, "Finding Topic Words for Hierarchical Summarization," *Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 349-357, New Orleans, LA, USA, 2001.

Linden, G., B. Smith, and J. York, "Amazon.Com Recommendations: Item-to-Item Collaborative Filtering," *IEEE Internet Computing*, Vol. 7, No. 1:76-80, 2003.

Lynch, J.G. and D. Ariely, "Wine Online: Search Costs Affect Competition on Price, Quality, and Distribution," *Marketing Science*, Vol. 19, No. 1:83-103, 2000.

MacQueen, J., "Some Methods for Classification and Analysis of Multivariate Observations," *Proceedings of the Berkeley Symposium on Mathematical Statistics and Probability*, Vol. 1:281-297, Berkeley, CA, USA, 1967.

Mahdavi, M. and H. Abolhassani, "Harmony K-Means Algorithm for Document Clustering," *Data Mining and Knowledge Discovery*, Vol. 18, No. 3:370-391, 2009.

McLachlan, G.J. and K.E. Basford, Mixture Models: Inference and Applications to Clustering, New York: Marcel Dekker, 1988.

Modha, D.S. and W.S. Spangler, "Feature Weighting in K-Means Clustering," *Machine Learning*, Vol. 52, No. 3:217-237, 2003.

Ng, H.P., S.H. Ong, K.W.C. Foong, P.S. Goh, and W.L. Nowinski, "Medical Image Segmentation Using K-Means Clustering and Improved Watershed Algorithm," *Proceedings of the IEEE Southwest Symposium on Image Analysis and Interpretation*, pp. 61-65, Denver, CO, USA, 2006.

Peng, J., C. Macdonald, B. He, V. Plachouras, and I. Ounis, "Incorporating Term Dependency in the Dfr Framework," *Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 843-844, Amsterdam, Netherland, 2007.

Ramachandran, K.K., K.K. Karthick, and M.S. Kumar, "Online Shopping in the Uk," *International Business and Economics Research Journal*, Vol. 10, No. 12:23-36, 2011.

Reynolds, D.A. and R.C. Rose, "Robust Text-Independent Speaker Identification Using Gaussian Mixture Speaker Models," *IEEE Transactions on Speech and Audio Processing*, Vol. 3, No. 1:72-83, 1995.

Robertson, S.E., S. Walker, and M. Beaulieu, "Experimentation as a Way of Life: Okapi at Trec," *Information Processing and Management*, Vol. 36, No. 1:95-108, 2000.

Sebastiani, F., "Machine Learning in Automated Text Categorization," *ACM computing surveys*, Vol. 34, No. 1:1-47, 2002.

Sindhwani, V., "Fast Linear Svm Solvers for Supervised and Semi-Supervised Learning," Available at http://vikas.sindhwani.org/svmlin.html, Accessed in 2014.

Sindhwani, V. and S.S. Keerthi, "Large Scale Semi-Supervised Linear Svms," *Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 477-484, Seattle, WA, USA, 2006.

Tan, C.H., H.H. Teo, and I. Benbasat, "Assessing Screening and Evaluation Decision Support Systems: A Resource-Matching Approach," *Information Systems Research*, Vol. 21, No. 2:305-326, 2010.

Tang, F.F. and X. Xing, "Will the Growth of Multi-Channel Retailing Diminish the Pricing Efficiency of the Web?," *Journal of Retailing*, Vol. 77, No. 3:319-333, 2001.

Wang, Z., J. Shawe-Taylor, and A. Shah, "Semi-Supervised Feature Learning from Clinical Text," *Proceedings of the IEEE International Conference on Bioinformatics and Biomedicine*, pp. 462-466, 2010.

Wong, T.L., T.S. Wong, and W. Lam, "An Unsupervised Approach for Product Record Normalization across Different Web Sites," *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 2:1249-1254, Chicago, IL, USA, 2008.

Wu, L., Y. Cai, and D. Liu, "Online Shopping among Chinese Consumers: An Exploratory Investigation of Demographics and Value Orientation," *International Journal of Consumer Studies*, Vol. 35, No. 4:458-469, 2011.

Wu, Z., J. Wu, J. Cao, and D. Tao, "Hysad: A Semi-Supervised Hybrid Shilling Attack Detector for Trustworthy Product Recommendation," *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 985-993, Beijing, China, 2012.

Yoon, K.A., O.S. Kwon, and D.H. Bae, "An Approach to Outlier Detection of Software Measurement Data Using the K-Means Clustering Method," *Proceedings of the International Symposium on Empirical Software Engineering and Measurement*, pp. 443-445, Madrid, 2007.

Zeng, H. and Y.M. Cheung, "A New Feature Selection Method for Gaussian Mixture Clustering," *Pattern Recognition*, Vol. 42, No. 2:243-250, 2009.

Zheng, X., S. Zhu, and Z. Lin, "Capturing the Essence of Word-of-Mouth for Social Commerce: Assessing the Quality of Online E-Commerce Reviews by a Semi-Supervised Approach," *Decision Support Systems*, Vol. 56:211-222, 2013.

Appendix A: Derivation of the complete data log-likelihood

The complete data log-likelihood function for the proposed model in Equation (5) of the *n*-th item with two hidden variables, d_n and p_n, is derived by using the conditional density functions shown in Equations (2), (3), and (4). The complete data log-likelihood over the entire items and their hidden variables, given the set of parameters of *cf*-SIM, is computed as follows:

$$
\begin{aligned}
\ln f(I,D,P\mid\Theta) &= \ln\prod_{n=0}^{N} f(\mathbf{x}_n, y_n, d_n = i, p_n = j\mid\Theta) \\
&= \ln\prod_{n=0}^{N}\Pr(d_n = i)\Pr(p_n = j\mid d_n = i) f(\mathbf{x}_n, y_n\mid d_n = i, p_n = j,\Theta) \\
&= \sum_{n=0}^{N}\ln\delta_i\pi_{ij} f(\mathbf{x}_n, y_n\mid\theta_{ij}) \\
&= \sum_{n=0}^{N}\ln\delta_i\pi_{ij}[f(\mathbf{x}_n, y_n\mid\theta_0)]^{(1-i)}[f(\mathbf{x}_n, y_n\mid\theta_{10})]^{i(1-j)}[f(\mathbf{x}_n, y_n\mid\theta_{11})]^{ij} \\
&= \sum_{n=0}^{N}\Big(\ln\delta_i + \ln\pi_{ij} + (1-i)\ln f(\mathbf{x}_n, y_n\mid\theta_0) + i(1-j)\ln f(\mathbf{x}_n, y_n\mid\theta_{10}) + ij\ln f(\mathbf{x}_n, y_n\mid\theta_{11})\Big) \\
&= \sum_{n=0}^{N}\left(\begin{array}{l}
\ln\delta_i + \ln\pi_{ij} + (1-i)\ln\left[\prod_{l=1}^{L}(\omega_l')^{x_{nl}}(1-\omega_l')^{(1-x_{nl})}\right]\left[\frac{1}{\sigma'\sqrt{2\pi}}\exp\left\{-\frac{(y_n-\mu')^2}{2\sigma'^2}\right\}\right] \\
+i(1-j)\ln\left[\prod_{l=1}^{L}(\omega_l)^{x_{nl}}(1-\omega_l)^{(1-x_{nl})}\right]\left[\frac{1}{\sigma'\sqrt{2\pi}}\exp\left\{-\frac{(y_n-\mu')^2}{2\sigma'^2}\right\}\right] \\
+ij\ln\left[\prod_{l=1}^{L}(\omega_l)^{x_{nl}}(1-\omega_l)^{(1-x_{nl})}\right]\left[\frac{1}{\sigma\sqrt{2\pi}}\exp\left\{-\frac{(y_n-\mu)^2}{2\sigma^2}\right\}\right]
\end{array}\right)
\end{aligned}
\tag{A.1}
$$

where $\mathbf{x}_0$ and y_0 respectively are an *L*-dimensional vector of term appearances and a price of a given query item, and $\Theta = \{\theta_{11}, \theta_{10}, \theta_0\}$ is a set of parameters.

We remark that we do not need to consider relative weighting between the probabilities pertaining to textual relevance and price relevance, since they are separately marginalized and then multiplied, as shown in Equations (C.3) through (C.12).

Appendix B: Parameter estimation in the expectation step

The state probability for the *n*-th item at the *m*-th EM iteration is computed with fixed $\Theta^{(m)}$ by using the responsibility function presented in Equation (B.1).

$$
\begin{aligned}
\rho_n^{(m)}(i,j) &= \Pr(d_n = i, p_n = j\mid\mathbf{x}_n, y_n, \Theta^{(m)}) \\
&= \frac{f(\mathbf{x}_n, y_n, d_n = i, p_n = j\mid\Theta^{(m)})}{\sum_{i=0}^{1}\sum_{j=0}^{1} f(\mathbf{x}_n, y_n, d_n = i, p_n = j\mid\Theta^{(m)})} \\
&= \frac{\delta_i\pi_{ij} f(\mathbf{x}_n, y_n\mid d_n = i, p_n = j, \Theta^{(m)})}{\sum_{i=0}^{1}\sum_{j=0}^{1}\delta_i\pi_{ij} f(\mathbf{x}_n, y_n\mid d_n = i, p_n = j, \Theta^{(m)})}
\end{aligned}
\tag{B.1}
$$

where the probability that the *n*-th item is in ν-state is summation of $\rho_n^{(m)}(0,0)$, $\rho_n^{(m)}(0,1)$, and $\rho_n^{(m)}(1,0)$, while τ-state probability for the item is $\rho_n^{(m)}(1,1)$.

Appendix C: Parameter estimation in the maximization step

While fixing $\rho_n^{(m)}(i,j)$, $i,j=0,1$, by using Equation (B.1), the updated parameters of *cf*-SIM for the (*m*+1)-th EM iteration are obtained as follows:

$$\Theta^{(m+1)} = \arg\max_{\Theta} Q(\Theta,\Theta^{(m)}) \tag{C.1}$$

where $Q(\Theta,\Theta^{(m)})$ is the expected value of the complete data log-likelihood shown in Equation (A.1), given the estimated parameters at the *m*-th EM iteration, and it is computed as follows:

$$Q(\Theta,\Theta^{(m)}) = \sum_{n=0}^{N}\sum_{i=0}^{1}\sum_{j=0}^{1}\left\{\rho_n^{(m)}(i,j)\ln f(\mathbf{x}_n, y_n, d_n=i, p_n=j \mid \Theta)\right\} \tag{C.2}$$

The parameters of *cf*-SIM, θ_{11}, θ_{10} ,and θ_0 , are calculated as follows. First, the set of parameters at the *m*-th EM iteration, $\theta_{11}^{(m+1)}$, for the items in τ -state for a query item, $(\mathbf{x}_0, y_0)$, is estimated as:

$$\hat{\omega}_l^{(m+1)} = \frac{\sum_{n=0}^{N}\sum_{i=0}^{1}\sum_{j=0}^{1}\rho_n^{(m)}(i,j)ix_{nl}}{\sum_{n=0}^{N}\sum_{i=0}^{1}\sum_{j=0}^{1}\rho_n^{(m)}(i,j)i} = \frac{\sum_{n=0}^{N}\sum_{j=0}^{1}\rho_n^{(m)}(1,j)x_{nl}}{\sum_{n=0}^{N}\sum_{j=0}^{1}\rho_n^{(m)}(1,j)} \tag{C.3}$$

$$\hat{\mu}^{(m+1)} = \frac{\sum_{n=0}^{N}\sum_{i=0}^{1}\sum_{j=0}^{1}\rho_n^{(m)}(i,j)ijy_n}{\sum_{n=0}^{N}\sum_{i=0}^{1}\sum_{j=0}^{1}\rho_n^{(m)}(i,j)ij} = \frac{\sum_{n=0}^{N}\rho_n^{(m)}(1,1)y_n}{\sum_{n=0}^{N}\rho_n^{(m)}(1,1)} \tag{C.4}$$

$$\hat{\sigma}^{(m+1)} = \sqrt{\frac{\sum_{n=0}^{N}\sum_{i=0}^{1}\sum_{j=0}^{1}\rho_n^{(m)}(i,j)ij(y_n-\mu^{(m)})^2}{\sum_{n=0}^{N}\sum_{i=0}^{1}\sum_{j=0}^{1}\rho_n^{(m)}(i,j)ij}} = \sqrt{\frac{\sum_{n=0}^{N}\rho_n^{(m)}(1,1)(y_n-\mu^{(m)})^2}{\sum_{n=0}^{N}\rho_n^{(m)}(1,1)}} \tag{C.5}$$

Second, the set of parameters at the *m*-th EM iteration, $\theta_{10}^{(m+1)}$, for the items in ν -state such that their descriptions are similar to that of the query item but their price distributions differ from that of the query item, is estimated as:

$$\hat{\omega}_l^{(m+1)} = \frac{\sum_{n=0}^{N}\sum_{i=0}^{1}\sum_{j=0}^{1}\rho_n^{(m)}(i,j)ix_{nl}}{\sum_{n=0}^{N}\sum_{i=0}^{1}\sum_{j=0}^{1}\rho_n^{(m)}(i,j)i} = \frac{\sum_{n=0}^{N}\sum_{j=0}^{1}\rho_n^{(m)}(1,j)x_{nl}}{\sum_{n=0}^{N}\sum_{j=0}^{1}\rho_n^{(m)}(1,j)} \tag{C.6}$$

$$\hat{\mu}'^{(m+1)} = \frac{\sum_{n=0}^{N}\sum_{i=0}^{1}\sum_{j=0}^{1}\rho_n^{(m)}(i,j)i(1-j)y_n}{\sum_{n=0}^{N}\sum_{i=0}^{1}\sum_{j=0}^{1}\rho_n^{(m)}(i,j)i(1-j)} = \frac{\sum_{n=0}^{N}\rho_n^{(m)}(1,0)y_n}{\sum_{n=0}^{N}\rho_n^{(m)}(1,0)} \tag{C.7}$$

$$\hat{\sigma}'^{(m+1)} = \sqrt{\frac{\sum_{n=0}^{N}\sum_{i=0}^{1}\sum_{j=0}^{1}\rho_n^{(m)}(i,j)i(1-j)(y_n-\mu'^{(m)})^2}{\sum_{n=0}^{N}\sum_{i=0}^{1}\sum_{j=0}^{1}\rho_n^{(m)}(i,j)i(1-j)}} = \sqrt{\frac{\sum_{n=0}^{N}\rho_n^{(m)}(1,0)(y_n-\mu'^{(m)})^2}{\sum_{n=0}^{N}\rho_n^{(m)}(1,0)}} \tag{C.8}$$

Next, at the *m*-th iteration of EM algorithm, the set of parameters, $\theta_0^{(m+1)}$, for the items in ν -state such that their descriptions are not similar to that of the query item, is estimated as:

$$\hat{\omega}_l'^{(m+1)} = \frac{\sum_{n=0}^{N}\sum_{i=0}^{1}\sum_{j=0}^{1}\rho_n^{(m)}(i,j)(1-i)x_{nl}}{\sum_{n=0}^{N}\sum_{i=0}^{1}\sum_{j=0}^{1}\rho_n^{(m)}(i,j)(1-i)} = \frac{\sum_{n=0}^{N}\sum_{j=0}^{1}\rho_n^{(m)}(0,j)x_{nl}}{\sum_{n=0}^{N}\sum_{j=0}^{1}\rho_n^{(m)}(0,j)} \tag{C.9}$$

$$\hat{\mu}'^{(m+1)} = \frac{\sum_{n=0}^{N}\sum_{i=0}^{1}\sum_{j=0}^{1}\rho_n{}^{(m)}(i,j)(1-i)y_n}{\sum_{n=0}^{N}\sum_{i=0}^{1}\sum_{j=0}^{1}\rho_n{}^{(m)}(i,j)(1-i)} = \frac{\sum_{n=0}^{N}\sum_{j=0}^{1}\rho_n{}^{(m)}(0,j)y_n}{\sum_{n=0}^{N}\sum_{j=0}^{1}\rho_n{}^{(m)}(0,j)} \tag{C.10}$$

$$\hat{\sigma}'^{(m+1)} = \sqrt{\frac{\sum_{n=0}^{N}\sum_{i=0}^{1}\sum_{j=0}^{1}\rho_n{}^{(m)}(i,j)(1-i)(y_n-\mu'^{(m)})^2}{\sum_{n=0}^{N}\sum_{i=0}^{1}\sum_{j=0}^{1}\rho_n{}^{(m)}(i,j)(1-i)}} = \sqrt{\frac{\sum_{n=0}^{N}\sum_{j=0}^{1}\rho_n{}^{(m)}(0,j)(y_n-\mu'^{(m)})^2}{\sum_{n=0}^{N}\sum_{j=0}^{1}\rho_n{}^{(m)}(0,j)}} \tag{C.11}$$

Lastly, the textual description coefficient and the price coefficient are estimated as:

$$\hat{\delta}_i{}^{(m+1)} = \frac{\sum_{n=0}^{N}\sum_{j=0}^{1}\rho_n{}^{(m)}(i,j)}{\sum_{n=0}^{N}\sum_{i=0}^{1}\sum_{j=0}^{1}\rho_n{}^{(m)}(i,j)} = \frac{\sum_{n=0}^{N}\sum_{j=0}^{1}\rho_n{}^{(m)}(i,j)}{N} \tag{C.12}$$

$$\hat{\pi}_{ij}{}^{(m+1)} = \frac{\sum_{n=0}^{N}\rho_n{}^{(m)}(i,j)}{\sum_{n=0}^{N}\sum_{i=0}^{1}\sum_{j=0}^{1}\rho_n{}^{(m)}(i,j)} \times \frac{\sum_{n=0}^{N}\sum_{i=0}^{1}\sum_{j=0}^{1}\rho_n{}^{(m)}(i,j)}{\sum_{n=0}^{N}\sum_{j=0}^{1}\rho_n{}^{(m)}(i,j)} = \frac{\sum_{n=0}^{N}\rho_n{}^{(m)}(i,j)}{\sum_{n=0}^{N}\sum_{j=0}^{1}\rho_n{}^{(m)}(i,j)} \tag{C.13}$$