Mining Substitution Rules: A Knowledge-based Approach using Dynamic Ontologies

Rupal Sethi and B. Shekar
Decision Sciences and Information Systems Area,
Indian Institute of Management Bangalore,560076, Bangalore, Karnataka, India

Keywords: Substitution Rules, Interestingness, Affordances, Dynamic Ontology, Market Basket.

Abstract: Association Rule Mining has so far focused on generating and pruning positive rules using various interestingness measures. However, there are very few studies that explore the mining process of substitution rules. These studies have incorporated a limited definition of substitution, either in statistical terms or based on manager's static knowledge. Here we attempt to provide a customer-centric model of substitution rule mining using the lens of affordance. We adopt a knowledge-based approach involving a *dynamic ontology* wherein objects are positioned based on the affordances they are preferred for. This contrasts with the traditional static ontology approach that highlights manager's static knowledge base. We develop an *Expected-Actual Substitution Framework* to compare relatedness between items in the static and dynamic ontologies. We present *Affordance-Based Substitution* (ABS) algorithm to mine substitution rules based on the proposed approach. We also come up with a novel interestingness measure that enhances the quality of our substitution rules thus leading to effective knowledge discovery. Empirical analyses are performed on a real-life supermarket dataset to show the efficacy of ABS algorithm. We compare the generated rules with those generated by another substitution rule mining algorithm from the literature. Our results show that substitution rules generated through ABS algorithm capture customer perceptions that are generally missed by alternate approaches.

1 INTRODUCTION AND RELATED WORK

The field of Artificial Intelligence (AI) is actively exploring the use of formal ontologies as a way of specifying knowledge for solving problems related to diagnosis, planning and design (Chandrasekaran, Josephson and Benjamins, 1999; Gruber, 1995). However the knowledge elicitation process in AI is restricted to a static representation in Knowledge Based Systems (Nau and Chang, 1986). Static knowledge-based systems assume a one-shot computation, usually triggered by a user query, often missing to consider dynamic scenarios where there is a need to react and evolve in the presence of incoming information (Brewka et al., 2016).

In most knowledge-based systems, problem solving is done by manipulating rules of the form "IF conditions THEN actions", often labelled as association rules (Galárraga et al., 2013). Association Rule (AR) Mining is one of the popular techniques of data mining which discovers relationships between groups of items. Algorithms like Apriori (Agrawal and Srikant, 1994) mine rules on the basis of frequency of occurrence of items in transaction data. This approach is restricted to positive association rules only. Positive AR is a relationship between items or groups of items which exists in the transaction set. One possible positive AR may be "IF customers buy *bread*, THEN *butter* is bought along with it".

Recently there has been a shift of focus from positive associations to substitution relations (Chen and Lee, 2015). Substitution relations depict items that are purchased as replacement to another item (Teng, Hseih and Chen, 2005). For example, Bread-Bun, Pepsi-Coke or Chair-Stool. These items may not be a part of the same transaction (Shekar and Natarajan, 2006). Thus, traditional AR mining algorithms (Agrawal and Srikant, 1994) cannot generate rules comprising substitute items. We classify research done on substitution rule mining under two categories: objective approach and subjective approach, given in Figure 1.

Sethi, R. and Shekar, B.
Mining Substitution Rules: A Knowledge-based Approach using Dynamic Ontologies.
DOI: 10.5220/0006577400730084
In *Proceedings of the 10th International Conference on Agents and Artificial Intelligence (ICAART 2018) - Volume 2*, pages 73-84
ISBN: 978-989-758-275-2

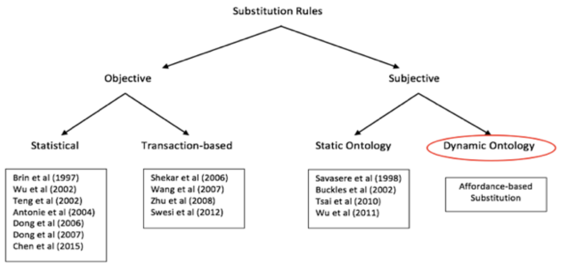


Figure 1: Classification of related work in substitution rule mining.

Objective substitution is divided into two more categories, namely *statistical* and *transaction-based*. Statistics-based approach encompasses algorithms that generate substitute items through measures like correlation. On the other hand, transaction-based research refers to conceptual work where substitution is defined not from the point-of-view of statistics but more from the angle of transactional nuances.

In order to statistically tap items absent in transactions, Brin, Motwani and Silverstein (1997) discuss substitutes in the context of mining *negative association rules*. Negative rules depict relationships between items that conflict each other. These negative relationships may help in identifying substitute items. Wu, Zhang and Zhang (2002) use an objective measure of interestingness to calculate covariance between items and thus identify negative relationships. Antonie and Zaiane (2004) extended the work of Brin et al (1997) by using Pearson's correlation coefficient as a measure of negative association and developed an algorithm to generate negative rules with a sliding correlation coefficient threshold. Chen and Lee (2015) furthered this work to generate non-redundant substitution rules by combining the concept of frequent closed itemsets with Pearson correlation coefficient. One of the pioneering research works to formally define substitution rules and provide the relevant mining process was by Teng, Hseih and Chen (2005). They use chi-square as a measure of interestingness to access the correlation between items.

Shekar and Natarajan (2006) define substitutability from a transactional orientation. They define direct substitutes as items that are purchased individually in different transactions and indirect substitutes as items that are purchased along with a common object. Similar approach has been adopted by Wang, Liu and Ma (2007) where they measure intensity of substitutability through rules containing composite items. Suppose rule $X \vee Y \rightarrow Z$ signifies occurrence of Z in the presence of either X or Y. Then, X and Y would act as substitutes, since one of them is required for the occurrence of Z.

Subjective substitution category takes into account the usage of subjective interestingness measures. Here, one strand of research pertains to tapping the manager's domain knowledge through *static ontology*. The research that we present here is towards understanding customer perceptions of substitution through *dynamic ontology*.

Savasere, Omiecinski and Navathe (1998) use a hierarchical structure to define substitutability based on the position of an item in the taxonomy. They restrict their definition to sibling substitutions wherein items that are siblings in a taxonomy are expected to exhibit similar behavior and hence are substitutable. They use *unexpectedness* as a measure of interestingness for capturing negative relationships. A similar approach has been adopted by Yuan, Buckles, Yuan and Zhang (2002). They also use the concept of locality of similarity while defining sibling rules. Sibling rules are a pair of positive association rules where both siblings are expected to be related to the same consequent. This approach is however restricted to a static ontology that does not accommodate changes in customer purchase patterns.

Substitution is defined in statistical terms or defined on the basis of the static knowledge of a manager. On the contrary we define a substitute as a product that is similar to another product on the basis of customer perceptions that are essentially dynamic. Our definition is adopted from Nicholson and Snyder (2011) who define a pair of substitutes as two goods where one good may, as a result of changed conditions, replace the other in use. The changed conditions may occur as a result of change in customer goals while buying a product. This highlights the fact that research on substitution rule generation needs to focus on the function that defines substitution between two products.

In this paper, we define substitution using the lens of *affordance*. This lens points to various features or applications an item may be used. This may be in contrast to a manager's typical expectations. We use a knowledge-based *dynamic ontology* towards this purpose. A dynamic ontology is represented as a hierarchical structure where items are positioned based on the affordances behind their purchase. Unlike a static ontology where the position of items is based on manager's prior knowledge, a dynamic ontology is constructed "on the fly" on the basis of varying customer purchase patterns. The sets of substitutable items obtained from the proposed dynamic ontology are then used to generate substitution rules. Further we also define an interestingness measure for the proposed affordance-based substitution rules and use the same in evolving a classification framework.

2 PRELIMINARIES

2.1 Dynamic Ontology

An ontology is an explicit specification of a conceptualization (Gruber, 1995). In AI, ontologies are used to represent knowledge in the form of objects, concepts and relationships among them (Genesereth and Nilsson, 1987). Ontologies are always specific to the domain of discourse as they form the foundational vocabulary of the area under study (Chandrasekaran, Josephson and Benjamins, 1999). In this paper, we adopt the ontology-based approach in the context of a supermarket where knowledge representation is a scheme of classification of products. It is represented as a hierarchical structure with parents as classes and leaves as products in that class. This ontology structure captures the domain knowledge of a manager in terms of classification of all the products. However, the definition of categories of products and their distinctions are not restricted to a manager's knowledge (Ratneshwar and Shocker, 1991). A categorization has different connotations for customers as well as for a manager. Categorization is a means to simplifying information, better decision making and having efficient interpersonal communication among customers (Shocker, Bayus and Kim, 2004). This notion is not taken into account while creating static ontologies in a supermarket scenario. In such static representations, products never change their hierarchical positions once they get classified (Li and Tsai, 2009). Hence it is difficult for managers to update their knowledge from trends in purchase patterns.

We suggest *dynamic ontology* as an add-on to static knowledge representation. A dynamic ontology in the context of a market basket scenario, would overcome the problem of static product categorization by incorporating customer characteristics and temporality into the classification. Customer preferences depend on a number of factors such as context, age, time, location, trust and new experiences (Rana and Jain, 2015). Customers might categorize products based on physical resemblance (Butter and Margarine), perceived similarity of producers (Coke and Pepsi) or category label fit (Facewash and Bathing Bar) (Day, Shocker and Srivastava, 1979). Two other factors that shape dynamic categorization of products are word of mouth among customers (Lee and Lee, 2009) and seasonality (Rana and Jain, 2015). Thus, product categorization is contingent on both customer purchase patterns and manager's prior knowledge. We define dynamic ontology through a tree structure comprising affordances as classes and products as leaves. Products are linked to an affordance class through a containment function that defines the degree to which the particular affordance is connected to that product.

2.2 Affordance

The concept of affordance originates in ecological psychology. *Affordances* are viewed as relational action possibilities that emerge from interaction between an object and its user (Gibson, 1977). This interaction is contingent on the features of an object and the abilities of a goal-seeking user (Stoffregen, 2003). In the absence of either of these, affordance may not exist. Product categorization cannot be considered without taking into account the effects of purpose (Shocker, Bayus and Kim, 2004). Since one product can serve multiple purposes, different users may 'afford' it differently based on their goals (Leonardi, 2013). For instance, someone may use a remote control to switch on a television device while another may use it as a paper-weight. Customers' past experiences and knowledge are also important factors while defining product categories and hence substitutability between items. We use *affordance* as a lens to define the proposed dynamic ontology that helps to mine interesting substitution rules. In this paper, we define *affordance* as a ranked list of features of a product which are preferred by customers while buying the product. Choice and ranking reflect the intentions (or expectations) of a customer to accomplish a goal with the help of that product.

3 SUBSTITUTION USING DYNAMIC ONTOLOGY

Intuitively, customers tend to substitute products in the same category because of the similarity of functions served by the products. However, the role of substitution can be more abstract based on inter-category replacement of products. Depending on the situation or customer's goals, two products in very different categories may be substitutes to each other.

For example, in order to accomplish the goal of cleaning the toilet, a customer may either buy *Coke* or *Harpic* whichever is available or is less expensive. Hence substitution here, is based on the situation or the context rather than brand or physical resemblance of products. Traditionally in a static taxonomy, Coke and Harpic would form parts of different classes like Beverages and Toiletries respectively. Hence they might be distant from each other making them a less likely substitutable pair in the static ontological structure. However, with the help of a dynamic ontology Coke and Harpic would be substitutable based on the affordance shared by them. In this case 'acidic' is the common affordance.

3.1 Formal Characterization of Dynamic Ontology

We represent a dynamic ontology as a n-ary tree representing multiple classification of products (items) based on affordances preferred by the customers while purchasing them. This hierarchical structure comprises 'has-a' relationships instead of 'is-a' relationships. We represent dynamic ontology O_d as:

$$O_d = (V, E, \delta) \tag{1}$$

where V is a set of vertices that comprise affordances A as classes (non-leaf nodes) and products P as leaf nodes. E is the set of edges connecting the nodes and δ is the containment function that assigns weight to each edge in E. Weight represents the degree of match of an affordance ai for a product pj. This is given by the frequency of transactions that contain ai and pj together. δ also gives importance to the order of the ranking of ai for pj. For a particular transaction tk that contains product pj and affordance ai with a rank r_{p_j,a_i} , containment function δ is given by (2):

$$\delta^{t_k}_{p_j,a_i} = \frac{1}{r_{p_j,a_i}} \tag{2}$$

The rationale for formalizing δ as the inverse of rank r is as follows. Lower the value of r (i.e. feature being ranked higher in the ranklist), higher the degree of match of that affordance for the product. As r increases, the value of δ decreases. This is because of the decrease in importance for affordance ai given by the customer.

δ is updated dynamically with the recurrence of ai and pj in a transaction. At any point in time, resultant δ is the cumulative mean that includes the current occurrence along with prior occurrences.

$$\delta_{p_j,a_i} = \sum_{\forall t_k \in T} \delta^{t_k}_{p_j,a_i} \, / \, n \tag{3}$$

where $\mathrm{t_k}$ is the $\mathrm{k^{th}}$ transaction containing $\mathrm{p_j}$ and $\mathrm{a_i}$
n being the total number of transactions T

Our affordance-based approach to constructing a dynamic ontology adheres to the fact that each object may have multiple affordances, based on users' varying goals (Leonardi, 2013). Since the dynamic ontology is constructed from individual transactions and without resorting to a manager's knowledge base, it reveals differing sets of affordances for a product-purchase.

A static knowledge representation is based on the *objective reality* (Hirschheim and Klein, 1989) and this is given by a manager's domain knowledge and past experiences. Generally static hierarchical structures are not updated with changes in customer preferences. The proposed notion of dynamic ontology is based on the *subjective reality* that reflects changing customer perceptions exhibited through their purchase behavior. Subjectivity in the ontology is captured through affordances specified by customers in the appended transactions. This necessitates evaluation of ontological stances pertaining to the dynamic structure and differentiating them from those in a static structure.

3.1.1 Concepts

A concept C is defined as an affordance class which specifies the feature of a particular object because of which it is purchased by the user. The difference between a concept and an affordance is that concept is a single entry in the affordance ranklist specified in the transaction. For example, a customer might buy *Coke* with affordance (Taste=Sweet, Ingredient=Cola, Nature=Fizzy). Here, Taste, Ingredient and Nature are three different concepts.

3.1.2 Meta-concepts

A meta-concept $\mathrm{M^c}$ is defined as an additional attribute of abstract concept C which enhances the information about user purchase decision. For example, consider a concept *Taste*. Meta-concepts associated with *Taste* could be sweet, bitter, salty and the like.

3.1.3 Instances

An instance *I* is an actualization of concepts and meta-concepts in a real-world object. A product *p* is an instance of concept *C* if it contains *C* in the affordance ranklist. In the dynamic ontology, an instance *I* can be a direct descendant of a concept *C* or can be a descendant of the meta-concepts of *C*. For example, *biscuit* is an instance that belong to concept *taste* and meta-concept *salty*.

3.1.4 Relationship between Concepts

Two concepts C_1 and C_2 are related to each other *iff* there is at least one product *p* that is simultaneously classified under C_1 (or under one of the descendants of C_1) and under C_2 (or one of the descendants of C_2).

$$C_1 \rightleftharpoons C_2 \quad iff\ \exists p \in P:\ p \in objects\ (C_1) \wedge p \in objects\ (C_2)$$

We introduce a measure **Concept Relatedness (CR)** that captures the degree of similarity between two concepts. Two concepts would be highly similar if a lot of products share the two concepts in their affordance ranklist specifications. For example, concepts taste and ingredient would be related to each other since many edibles possess both concepts as affordances.

$$Concept\ Relatedness, CR = \frac{n(p)}{n(P)} \tag{4}$$

$$\text{where } \mathrm{n(p)} = |\exists \mathrm{p} \in \mathrm{P} : \mathrm{p} \in \text{objects } (\mathrm{C}_1) \wedge \ \mathrm{p} \text{ objects } (\mathrm{C}_2)|$$
$$\text{and } \mathrm{n(P)} \text{ is the total number of products in the transactions}$$

In a static representation, concepts are related only through their positions in the tree. However, in dynamic ontologies, concepts are related through a transaction-based factor *p*. This transaction-based factor highlights customer preferences for relating two affordances or features together. This approach contrasts with the manager-centric approach that is limited to his prior knowledge from the static tree structure.

3.1.5 Relationship between Meta-Concepts

Meta-concepts are related through their position in the tree. M_1^c and M_2^c are siblings since they belong to the same concept C. Sibling meta-concepts can either be mutually exclusive or mutually-inclusive based on the instances they share. For example, *Coke* is sweet but a biscuit may be both sweet and salty. Thus, in this case, the meta-concepts sweet and salty corresponding to concept *Taste* are mutually-inclusive.

In case of mutually-exclusive meta-concepts, there is only one path from a product to the related abstract concept. However, for mutually-inclusive meta-concepts, there may exist multiple paths from a product to the abstract concept. Thus, for mutually-inclusive meta-concepts, the containment function is defined in (5).

$$\delta_{p,c} = \sum_{\forall k} \delta_{p,M^c}\,(k)\ /\ n \tag{5}$$

$$\text{where } \mathrm{k} = 1{,}2..\mathrm{n} \text{ for all the possible paths from meta} - \text{concepts } \mathrm{M^c} \text{ to concept C}$$

We take weighted average of all possible containment functions from the mutually-inclusive meta-concepts to the concept. This is done to include all customer choices related to that concept mentioned in the affordance ranklist. The rankings for the same are also taken care of. For example, $\delta_{Biscuit,Taste}$ is calculated as weighted average of $\delta_{Biscuit,Taste=Sweet}$ and $\delta_{Biscuit,Taste=Salty}$. The weights are assigned based on the frequency of customers preferring sweet versus salt tastes for biscuits.

3.1.6 Relationship between Instances

Instances are related to each other through a containment function δ. A containment function defines the degree of match between instances I_1 and I_2 under concept C, represented through respective edge weights.

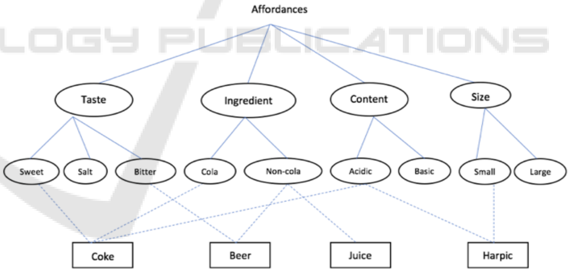


Figure 2: Sample dynamic ontology representation.

Figure 2 shows a sample dynamic ontology representation comprising concepts, meta-concepts and instances along with relationships. Solid lines represent complete containment and broken lines represent partial containment. Complete containment refers to the fact that a child only belongs to one parent. For instance, Sweet is completely contained by Taste i.e. it will not belong to any other concept such as Size or Ingredient. On the other hand, partial containment refers to the concept of multiple inheritance (Solé-Ribalta, Sánchez, Batet and Serratosa, 2014) in ontologies. Like, Beer inherits from two different concepts Taste and Ingredient.

Broken lines connect products in the form of instances to meta-concepts through containment function δ. Thus, two instances will be siblings only with respect to a concept-meta-concept pair. The degree of this sibling relationship between two instances with respect to a given concept is given by their respective containment functions. For example, in Figure 2, Coke and Harpic are siblings with respect to the concept (Content=Acidic) with varying degrees of match.

3.1.7 Affordance-based Substitute Sets

We define concept-level substitute sets, *S(c)* that are created from the dynamic ontology as follows:

$$S(c) = \{\ x \mid x\ \in LN\big(path_k(c)\big)\ \} \qquad (6)$$

$$\text{where c is concept corresponding to the affordance,}$$
$$\text{LN is a leaf node for path k starting from concept c}$$
$$k = 1,2..n$$

These substitute sets comprise all instances in the form of leaf nodes in the dynamic ontology which share common affordances. Consider the dynamic ontology representation given in Figure 2. For Taste as a concept, the members of its substitute set are:

$$S(Taste) = \{Coke, Beer\}$$

from two paths Taste-Sweet-Coke (k=1) and Taste-Bitter-Beer (k=2).

4 ITEM RELATEDNESS AND THE SUBSTITUTION FRAMEWORK

4.1 Expected Relatedness

Static ontology is represented as a *n*-ary tree which defines "is-a" relationships between products and their categories. It is based on manager's domain knowledge and is often not updated with changing trends in customer purchase patterns. A static ontology depicts manager's expectedness of products as substitutes based on their positions in the tree. We define a measure called *Expected Relatedness* to quantify manager's expectations of two products being related and hence substitutable. *Expected Relatedness* is defined in terms of hierarchical relationship with respect to a common ancestor item. Shaw, Xu and Geva (2009) have defined *Diversity*. We make use of this in defining *Expected Relatedness* as the complement of *Diversity*. Thus, we have the following for products p_1 and p_2

$$Diversity\ (p_1, p_2) = \frac{LD(p_1, ca) + LD(p_2, ca)}{2 * Tree\ Height} \qquad (7)$$

$$Expected\ Relatedness(p_1, p_2) = 1 - \frac{LD(p_1, ca) + LD(p_2, ca)}{2 * Tree\ Height} \qquad (8)$$

$$\text{LD}\ (\text{x}, \text{y}) = |\text{Hierarchy level of x} - \text{Hierarchy level of y}\ |$$
$$\text{and ca is the common ancestor of } \text{x}_1 \text{and } \text{x}_2$$

Diversity is the ratio of the average of number of levels for p_1 and that for p_2 (with respect to their common ancestor), to the height of the tree (Shaw et al, 2009). We define *Expected Relatedness* (ER) between two products as the complement of this ratio.

4.2 Affordance Relatedness

We need to operationalize the distance between two products in a dynamic ontology. This is done by introducing a measure called *Affordance Relatedness.* It captures customer perceptions of substituting products based on common features or applications.

$$Affordance\ Relatedness(p_1, p_2, c) = \left|\frac{\delta_{p_1,c}}{\sum_{\forall c_i \in C}\ \delta_{p_1,c_i}} - \frac{\delta_{p_2,c}}{\sum_{\forall c_i \in C}\ \delta_{p_2,c_i}}\right| \qquad (9)$$

Affordance Relatedness (AfR) is based on distances in a dynamic ontology. Distance between two products in a dynamic ontology cannot be calculated through their structural positions. This is because it is not constructed as levels of classification of products but as features shared among products. Thus we calculate relatedness between two products p_1 and p_2 sharing common concept (affordance) *c* as the difference in their containment functions for *c*. We consider normalized δ instead of absolute δ. This is to ensure that there is no over-estimation of AfR for the two products. Computed values of δ are obtained only from transactions containing the particular concept for a product. This value is not normalized for transactions that contain different concepts for the same product. Thus, it is necessary to normalize $\delta_{p,c}$ with the sum of containment functions related to all concepts shared by *p*. The difference after normalization thus presents the true picture of affordance relatedness for that product.

Cumulative Affordance Relatedness (CAfR) is the relationship between two products based on all concepts shared by them. We compute this by taking the complement of maximum of all AfRs relevant to p_1 and p_2.

$$CAfR(p_1, p_2) = 1 - Max(AfR(p_1, p_2, c_i)) \quad \forall c_i \in C_s \tag{10}$$

where C_s is the set of common concepts shared by p_1 and p_2

4.3 The E-A Substitution Framework

It is essential to assess the quality and interestingness of substitutable sets (or pairs) of products generated from the proposed dynamic ontology-based approach. This is done by comparing the item relatedness of substitute pairs through their positions in the static and dynamic ontologies. Substitution in the static ontology is based on manager's expectations resulting from prior domain knowledge. On the other hand, substitution in the dynamic ontology comes from actual customer purchase patterns. Thus, we propose an *Expected-Actual (E-A) Substitution Framework* to compare the quality of substitute sets obtained from our dynamic ontology with the static managerial knowledge. Horizontal direction of the E-A framework in Figure 3 represents the degree of relatedness between products in the dynamic ontology. This is based on the affordances they share. Vertical direction represents the degree of relatedness between products in the static ontology. This is based on the manager's expectations of their categorization. The comparison yields us four different possible substitutions.

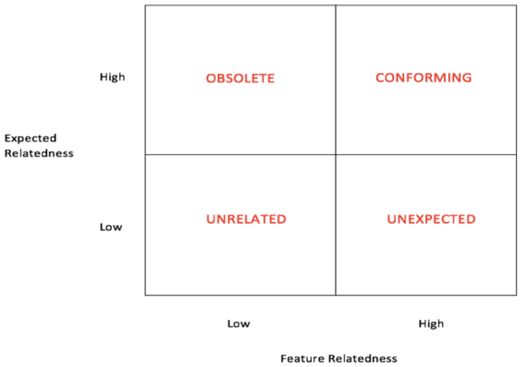


Figure 3: Expected-Actual Substitution Framework.

Conforming Substitutes

Here *Cumulative Affordance Relatedness* is high and *Expected Relatedness* is also high. This results in the substitutable pair of products being conforming. This is because manager expects products to be substitutable (i.e. they belong to the same parent in the static ontology). In addition, they also share common concepts in their affordance ranklist (in the dynamic ontology) as highlighted by the purchase transactions. Hence, conforming substitutable pairs do not present any interesting knowledge to the manager.

Obsolete Substitutes

Cumulative Affordance Relatedness being low and *Expected Relatedness* being high, result in the substitutable pair of products becoming obsolete. This is because these products are expected to be substitutable by the manager but in actuality customers never substitute them on the basis of any common concept. Hence the manager needs to modify his existing knowledge about this pair of obsolete products through some actionable event.

Unrelated Substitutes

If *Cumulative Affordance Relatedness* and *Expected Relatedness* are both low, then the substitutable pair of products is unrelated. This is because these products are neither expected to be substitutable by the manager nor do they share any common affordance. Such unrelated pairs of substitutable products will not result in interesting negative rules getting generated.

Unexpected Substitutes

Here *Cumulative Affordance Relatedness* is high and *Expected Relatedness* is low. This is unexpected because the manager never expected these products to be substitutable. However in actuality customers often substitute them based on common concepts. Hence, these unexpected pairs of substitutable products yield most interesting insights for the manager.

4.4 Substitute Interestingness Measure

We define a composite measure of interestingness that encompasses all the four quadrants of the E-A framework. This measure is the arithmetic difference between positions of two products in the dynamic ontology (CAfR) and the same being represented in the static ontology (ER).

$$SI\ (p_1, p_2) = CAfR(p_1, p_2) - ER(p_1, p_2) \tag{11}$$

This measure of *Substitute Interestingness* (SI) represents the additional knowledge made available to the manager vis-à-vis his expected beliefs. The rationale behind subtraction is removal of the already known previous beliefs of the manager regarding substitution of products.

Consider the four scenarios of the E-A framework for the calculation of SI. Products falling in the bottom-left quadrant and those falling in the top-right quadrant will have low values for SI. This essentially says that conforming and unrelated substitutes do not provide rich insights for the manager. However, the bottom-right and the top-left quadrants will have high values for SI, thus telling the manager to investigate these unexpected and obsolete substitutes for possible useful interestingness insights.

5 AFFORDANCE-BASED SUBSTITUTION RULES

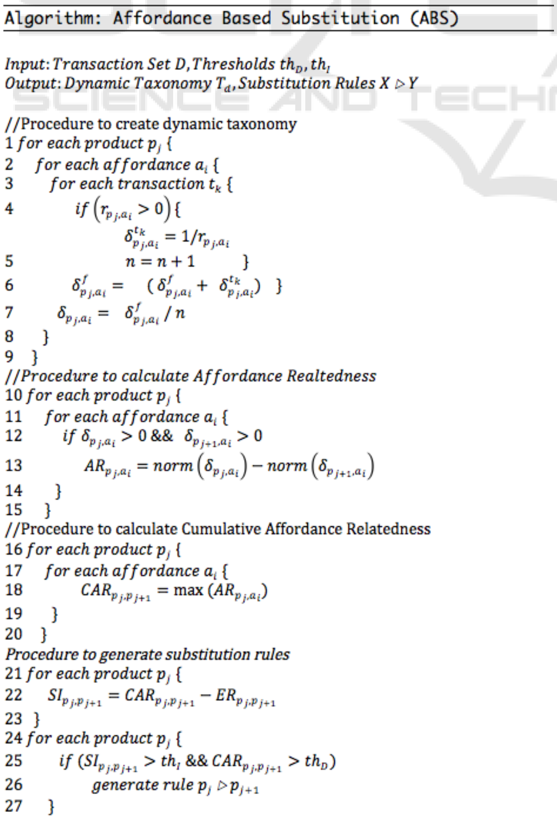
We mine substitution rules with the help of substitution sets obtained from the dynamic ontology. We generate rules comprising products with high affordance relatedness and high interestingness values.

Two products X and Y form a substitution rule $X \rhd Y$, if the following hold:

1) $CAfR(X, Y) \geq th_D$
2) $SI(X, Y) \geq th_I$

th_D and th_I are thresholds for dynamic ontology relatedness and interestingness, respectively.

The Affordance-based Substitution (ABS) algorithm for generating substitution rules is given below.



```
Algorithm: Affordance Based Substitution (ABS)

Input: Transaction Set D, Thresholds $th_D, th_I$
Output: Dynamic Taxonomy $T_d$, Substitution Rules $X \rhd Y$

//Procedure to create dynamic taxonomy
for each product $p_j$ {
  for each affordance $a_i$ {
    for each transaction $t_k$ {
      if $(r_{p_j,a_i} > 0)$ {
          $\delta^{t_k}_{p_j,a_i} = 1/r_{p_j,a_i}$
          $n = n + 1$        }
      $\delta^{f}_{p_j,a_i} = (\delta^{f}_{p_j,a_i} + \delta^{t_k}_{p_j,a_i})$ }
    $\delta_{p_j,a_i} = \delta^{f}_{p_j,a_i} / n$
  }
}
//Procedure to calculate Affordance Realtedness
for each product $p_j$ {
  for each affordance $a_i$ {
    if $\delta_{p_j,a_i} > 0$ && $\delta_{p_{j+1},a_i} > 0$
      $AR_{p_j,a_i} = norm(\delta_{p_j,a_i}) - norm(\delta_{p_{j+1},a_i})$
  }
}
//Procedure to calculate Cumulative Affordance Relatedness
for each product $p_j$ {
  for each affordance $a_i$ {
      $CAR_{p_j,p_{j+1}} = \max(AR_{p_j,a_i})$
  }
}
Procedure to generate substitution rules
for each product $p_j$ {
  $SI_{p_j,p_{j+1}} = CAR_{p_j,p_{j+1}} - ER_{p_j,p_{j+1}}$
}
for each product $p_j$ {
  if $(SI_{p_j,p_{j+1}} > th_I$ && $CAR_{p_j,p_{j+1}} > th_D)$
      generate rule $p_j \rhd p_{j+1}$
  }
```

6 DATA DESCRIPTION

To study the effectiveness of our model, we ran *ABS algorithm* on a real-world supermarket dataset. Transaction data *D* was obtained for a period of 13 months from November 2015 to December 2016. The transactions covered 113 items, that were then classified into 11 product categories. A random sample of 110 customers of the supermarket was drawn to collect the affordance data through a survey instrument. The descriptive statistics of the sample is given in Table 1. The survey pertained to ranking various features of the 11 selected product categories based on customer preferences while purchasing items from that category. We obtained a total of 4120 transactions. They were then appended with product and affordance ranking data. We obtained the static ontology pertaining to the 11 product categories from the manager of the supermarket. The static ontology consisted of 5 levels with categories such as personal care, edibles and the like.

Table 1: Descriptive Statistics of 110 customers surveyed.

	Mean	**Minimum**	**Maximum**
Age	26	24	30
Work Experience (months)	22.46	0	65
Gender	Male=78%	Female=22%	

7 RESULTS AND ANALYSIS

7.1 Dynamic Ontology

We ran the ABS algorithm on the supermarket dataset to generate affordance-based substitution rules. Feature rankings recorded from the survey

Table 2: Matrix representation of dynamic ontology from supermarket dataset.

Del	**A1**	**A2**	**A3**	**A4**	**A5**	**A6**	**A7**	**A8**
P001	0.97	0.49	0.40	-	-	-	-	-
P002	0.66	0.43	-	0.65	0.44	-	-	-
P003	-	0.42	-	-	0.65	0.81	-	-
P004	0.90	-	0.29	0.38	-	-	0.43	0.36
P005	0.60	0.36	-	-	0.73	0.44	-	0.26
P006	-	0.48	-	0.46	-	0.93	-	-
P007	0.52	-	-	0.70	0.48	-	-	-
P008	-	-	-	-	0.65	0.82	-	0.39
P009	0.70	0.41	-	0.75	-	-	-	-
P010	0.41	0.32	-	-	0.70	-	-	-
P011	0.73	0.38	-	-	0.64	-	-	0.38

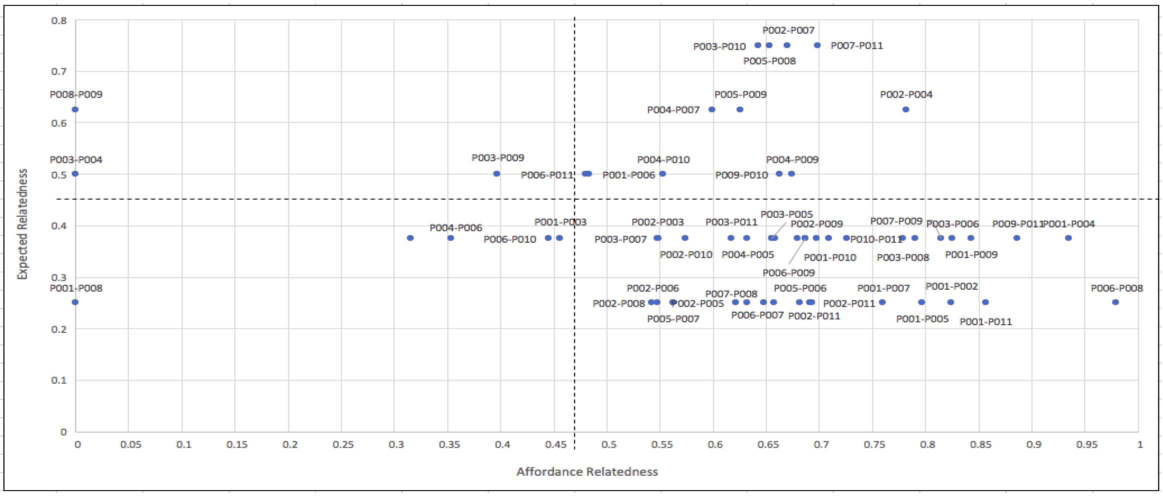


Figure 4: E-A framework for the substitute pairs.

were categorized into 8 affordance classes (A_i). The resultant dynamic ontology included edge weights (containment functions) $\delta_{p,a}$ for each product-affordance pair. The matrix representation of the dynamic ontology is given in Table 2.

7.2 Relatedness Measures and E-a Framework

We computed the two relatedness measures, CAfR and ER, for the substitute sets obtained from the dynamic ontology constructed from the supermarket dataset. The E-A framework based on the distribution of the resulting 55 substitute pairs is presented in Figure 4.

Substitute pairs are distributed across all four quadrants of the E-A framework. Analysis of the four quadrants in Figure 4 is as follows:

Conforming Substitutes

This quadrant provides the least interesting knowledge for a manager. Product pairs here are expected to be substitutable with respect to the manager's static ontology as well as through customer oriented dynamic ontology. For instance, P007-P011 (Facewash-Soap) are conforming substitutes. Both are siblings in the static ontology (ER=0.75) and have a high affordance relatedness in dynamic ontology (CAfR=0.70).

Obsolete Substitutes

This quadrant provides interesting information with which a manager may modify prior knowledge about obsolete substitutes pairs. For example, consider P003-P009 (Hair Oil-Shampoo). Although both are siblings in the static ontology (ER=0.5), they do not substitute each other in the dynamic ontology (CAfR=0.4). Thus, it is evident that static ontology will not suffice for the analysis of such substitution rules.

Unrelated Substitutes

Unrelated substitutable products will not result in the generation of interesting substitution rules because of low relatedness values in both static and dynamic ontologies. Consider P001-P008 (Room Freshener-Toothbrush) neither lie close to each other in the static ontology (ER=0.25) nor do they have any common affordances in the dynamic ontology (CAfR=0).

Unexpected Substitutes

Major concentration of pairs occurs in this quadrant leading to interesting insights for the manager. The large concentration also highlights the necessity for dynamic ontology approach. This may help in updating the static ontology-based manager's knowledge. Consider P001-P004 (Room Freshener-Deodorant). Both products lie in altogether different classes in the static ontology, personal care and household respectively (ER=0.375). However, they share many concepts, like fragrance, alcohol content, packaging and size (CAfR=0.56). Thus the proposed approach will result in their being classified as substitutes.

7.3 Substitution Rule Generation

Computation of interestingness measure SI was done for the generated substitution rules. We looked at its effectiveness in pruning the substitution rules. Variation in the fraction of rules generated with respect to changing SI values is given in Figure 5. The plot is for three threshold values of CAfR, namely 0.5, 0.7 and 1. This is because CAfR being greater than the threshold is one of the necessary conditions for generating substitution rules. We note that generated rules increases by 30% when CAfR threshold is increased from 0.5 to 0.7, and by 17% when increased from 0.7 to 1. This shows that our substitution rules comprise items with high affordance relatedness and low expected relatedness.

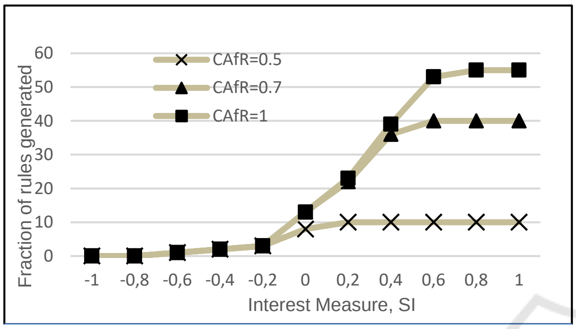


Figure 5: Fraction of rules generated through ABS algorithm for varying SI and CAFR.

7.4 Comparison between ABS and SRM Algorithm

We compare our algorithm with Substitute Rule Mining (SRM) algorithm developed by Teng et al (2005). We show that substitution rules generated by ABS are better in terms of efficiency and quality than those generated by SRM.

The supermarket dataset was appended with complement items as highlighted by Teng et al (2005, p.162). The SRM algorithm was then used to generate substitution rules. We present the changes in the cardinality of substitution rules with varying support thresholds in Figure 6. The variation follows a non-linear curve as the support threshold decreases.

The 189 substitution rules generated by SRM algorithm with a support threshold of 0.2 are then compared with the 55 substitution rules generated by ABS algorithm. The comparison of SRM and ABS algorithms is presented in terms of support and *SI* interest measures in Figure 7.

Out of the 189 rules generated, SRM misses out 27 substitution rules generated by ABS algorithm. Since substitution rules generated by ABS are indicative of customer perceptions while substituting two products, missing these rules is indicative of exclusion of changing purchase patterns. Thus we find that ABS algorithm has more potential to capture customer-perceptions. For instance, the substitution rule *P001* ▷ *P002* generated by ABS having interest SI 0.57 is not generated by SRM. This loss in information is critical because P001 and P002 are substitutable through an affordance relatedness (CAfR) of 0.82. A high CAfR highlights P001 and P002 sharing a lot of concepts in common making them highly substitutable by customers. Comparison of the two algorithms also reveals cases where substitution rules having a low negative value of SI (from ABS) have a very high support from SRM. Rule *P004* ▷ *P008* (SI = -0.06, Support = 0.67) is one such case. This is essentially misspecification pointing to over estimation of substitution rules.

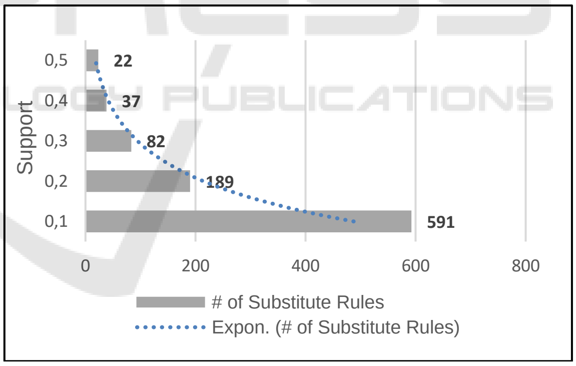


Figure 6: Number of substitution rules generated by SRM.

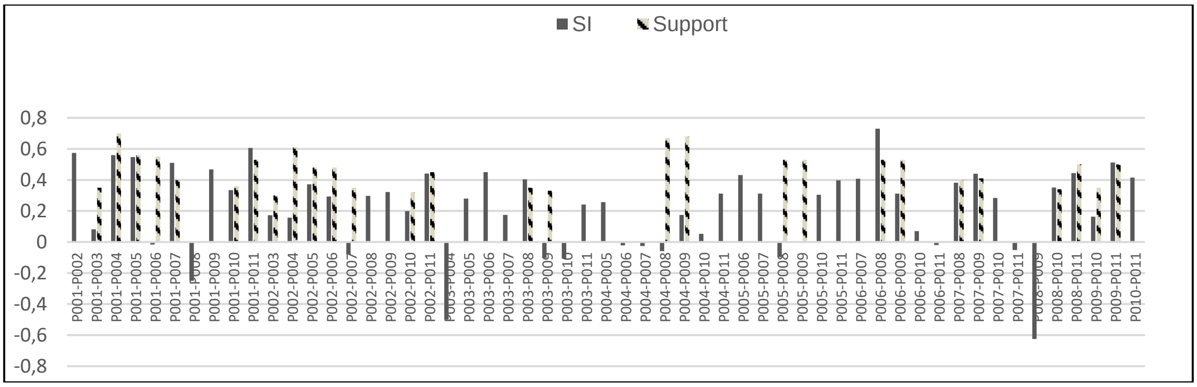


Figure 7: Comparison between ABS and SRM algorithm using support and SI values.

8 CONCLUSIONS

AR Mining literature has focused a lot on generating positive rules. Researchers have now started proposing efficient algorithms for mining substitution rules (Chen and Lee, 2015). However these algorithms have restricted their definition of substitution to either statistics or a manager's static knowledge. We extend the work on substitution rule mining by introducing a customer-centric view on substitution using the lens of affordance.

In a static ontology positioning of items is based on manager's previous knowledge. We propose the concept of a *dynamic ontology* that is constructed "on the fly" based on varying customer purchase patterns. This variation in purchase patterns is tapped using affordance specified as a ranked list together with products specified in each purchase transaction. We propose a containment function that assigns a value for each product-affordance pair. These values are then used to form substitute sets leading to generation of substitution rules. We provide an *Expected-Actual (EA) Substitution framework* that helps classify pairs of substitute products into four categories: *conforming, obsolete, unrelated* and *unexpected* substitutes. We also use this framework to come up with a novel interestingness measure that compares item relatedness between static and dynamic ontologies and prunes redundant substitution rules.

Our substitution rule mining process is operationalized through an *Affordance Based Substitution (ABS)* algorithm. A real-life super-market dataset is used to test the efficacy and effectiveness of the ABS algorithm. Our results show that substitution rules generated through ABS algorithm have better quality in terms of interestingness for a manager. We compare our approach with the approach given by Teng et al (2005). The comparison shows that several high-quality rules generated by ABS get missed by SBM algorithm (Teng et al, 2005). This highlights the fact that changing customer-perceptions are not reflected in current substitution rule mining algorithms. We also attempt to place the generated pairs of substitute products in our *E-A Substitution framework.* This placement and distribution provides useful managerial insights hitherto not present in the data mining literature.

REFERENCES

Agrawal, R., & Srikant, R. (1994). Fast algorithms for mining association rules. In *Proc. 20th int. conf. very large data bases, VLDB* (Vol. 1215, pp. 487-499).

Antonie, M. L., & Zaïane, O. R. (2004). Mining positive and negative association rules: an approach for confined rules. In *European Conference on Principles of Data Mining and Knowledge Discovery* (pp. 27-38). Springer Berlin Heidelberg.

Brewka, G., Ellmauthaler, S., Gonçalves, R., Knorr, M., Leite, J., & Pührer, J. (2016). Towards Inconsistency Management in Reactive Multi-Context Systems. In *DARe@ ECAI*.

Brin, S., Motwani, R., & Silverstein, C. (1997). Beyond market baskets: Generalizing association rules to correlations. In *Acm Sigmod Record* (Vol. 26, No. 2, pp. 265-276). ACM.

Chandrasekaran, B., Josephson, J. R., & Benjamins, V. R. (1999). What are ontologies, and why do we need them?. *IEEE Intelligent Systems and their applications*, *14*(1), 20-26.

Chen, Y. C., & Lee, G. (2015). Mining Non-Redundant Substitution Rules between Sets of Items in Large Databases. *J. Inf. Sci. Eng.*, *31*(2), 659-674.

Day, G. S., Shocker, A. D., & Srivastava, R. K. (1979). Customer-oriented approaches to identifying product-markets. *The Journal of Marketing*, 8-19.

Galárraga, L. A., Teflioudi, C., Hose, K., & Suchanek, F. (2013, May). AMIE: association rule mining under incomplete evidence in ontological knowledge bases. In *Proceedings of the 22nd international conference on World Wide Web* (pp. 413-422). ACM.

Genesereth, M. R., & Nilsson, N. J. (1987). Logical foundations of artificial. *Intelligence. Morgan Kaufmann*, *2*.

Gibson, J. J. (1977). The theory of affordances. Hilldale, USA.

Gruber, T. R. (1995). Toward principles for the design of ontologies used for knowledge sharing? *International journal of human-computer studies*, *43*(5-6), 907-928.

Hirschheim, R., & Klein, H. K. (1989). Four paradigms of information systems development. *Communications of the ACM*, *32*(10), 1199-1216.

Lee, J., & Lee, J. N. (2009). Understanding the product information inference process in electronic word-of-mouth: An objectivity–subjectivity dichotomy perspective. *Information & Management*, *46*(5), 302-311.

Leonardi, P. M. (2013). When does technology use enable network change in organizations? A comparative study of feature use and shared affordances. *MIS Quarterly*, *37*(3), 749-775.

Li, S. T., & Tsai, M. H. (2009). A dynamic taxonomy for managing knowledge assets. *Technovation*, *29*(4), 284-298.

Nau, D. S., & Chang, T. C. (1986). Hierarchical representation of problem-solving knowledge in a frame-based process planning system. *International Journal of Intelligent Systems*, *1*(1), 29-44.

Nicholson, W., & Snyder, C. (2011). *Microeconomic theory: Basic principles and extensions*. Nelson Education.

Rana, C., & Jain, S. K. (2015). A study of the dynamic features of recommender systems. *Artificial*

Intelligence Review, *43*(1), 141-153.

Ratneshwar, S., & Shocker, A. D. (1991). Substitution in use and the role of usage context in product category structures. *Journal of Marketing Research*, 281-295.

Savasere, A., Omiecinski, E., & Navathe, S. (1998). Mining for strong negative associations in a large database of customer transactions. In *Data Engineering, 1998. Proceedings., 14th International Conference on* (pp. 494-502). IEEE.

Shaw, G., Xu, Y., & Geva, S. (2009). Interestingness Measures for Multi-Level Association Rules. *Proceedings of ADCS 2009*, 27-34.

Shekar, B., & Natarajan, R. (2006). Investigations into Relatedness-based Interestingness of Association Rules: A Transaction-driven Analysis. In *Information Reuse and Integration, 2006 IEEE International Conference on* (pp. 522-527). IEEE.

Shocker, A. D., Bayus, B. L., & Kim, N. (2004). Product complements and substitutes in the real world: The relevance of “other products”. *Journal of Marketing*, *68*(1), 28-40.

Solé-Ribalta, A., Sánchez, D., Batet, M., & Serratosa, F. (2014). Towards the estimation of feature-based semantic similarity using multiple ontologies. *Knowledge-Based Systems*, *55*, 101-113.

Stoffregen, T. A. (2003). Affordances as properties of the animal-environment system. Ecological Psychology, 15(2), 115-134.

Teng, W. G., Hsieh, M. J., & Chen, M. S. (2005). A statistical framework for mining substitution rules. *Knowledge and Information Systems*, *7*(2), 158-178.

Wang, K., Liu, J. N., & Ma, W. M. (2006). Mining the Most Reliable Association Rules with Composite Items. In *Data Mining Workshops, 2006. ICDM Workshops 2006. Sixth IEEE International Conference on* (pp. 749-754). IEEE.

Wu, X., Zhang, C., & Zhang, S. (2002). Mining both positive and negative association rules. In *19th International Conference on Machine Learning*. Morgan Kaufmann.

Yuan, X., Buckles, B. P., Yuan, Z., & Zhang, J. (2002). Mining negative association rules. In *Computers and Communications, 2002. Proceedings. ISCC 2002. Seventh International Symposium on* (pp. 623-628). IEEE.