A Unified Framework for Utility Based Measures for Mining Itemsets [*]

Hong Yao
Department of Computer Science, University of Regina
Regina, SK, Canada S4S 0A2
yao2hong@cs.uregina.ca

Howard J. Hamilton
Department of Computer Science, University of Regina
Regina, SK, Canada S4S 0A2
hamilton@cs.uregina.ca

Liqiang Geng
Department of Computer Science, University of Regina
Regina, SK, Canada S4S 0A2
gengl@cs.uregina.ca

ABSTRACT

A pattern is of utility to a person if its use by that person contributes to reaching a goal. Utility based measures use the utilities of the patterns to reflect the user's goals. In this paper, we first review utility based measures for itemset mining. Then, we present a unified framework for incorporating several utility based measures into the data mining process by defining a unified utility function. Next, within this framework, we summary the mathematical properties of utility based measures that will allow the time and space costs of the itemset mining algorithm to be reduced.

Categories and Subject Descriptors

H.2.8 [**Database Management**]: Database Applications—*data mining*

General Terms

Measures

Keywords

Data Mining, Knowledge Discovery, Interestingness Measures, Utility Based Measures, Utility Based Data Mining

1. INTRODUCTION

Data mining can be regarded as an algorithmic process that takes data as input and yields patterns, such as classification rules, itemsets, association rules, or summaries, as output. For example, frequent itemsets can be discovered from market basket data and used to derive association rules for predicting the conditional probability of the purchase of certain items, given the purchase of other items [1, 2, 9]. An *itemset* is a set of items. The goal of frequent itemset mining is to identify all *frequent itemsets*, i.e., itemsets that have at least a specified minimum *support*, which is the percentage of transactions containing the itemset. In this paper, we focus on itemset mining.

Interestingness measures can play an important role in knowledge discovery. These measures are intended for selecting and ranking patterns according to their potential interest to the user. For example, itemset mining is based on the assumption that only itemsets with high support are of interest to users. That is, the support measure uses frequency as an estimate of the utility of a pattern to a user.

Measuring the interestingness of discovered patterns is an active and important area of data mining research. A comprehensive study of twenty-one measures that were originally developed in diverse fields such as statistics, social science, machine learning, and data mining is presented by Tan et al. [19]. Hilderman and Hamilton [8] theoretically and empirically evaluated twelve diversity measures used as heuristic measures of interestingness for ranking summaries generated from dataset. Yao et al. [21] presented a simple and unified framework for the study of quantitative measures associated with rules. Most research on interestingness measures has focused on using a statistical or mathematical method to evaluate the usefulness of rules [10], but such a method is not trivial for a human expert to understand. In general, it is not easy for user to choose one of the measures, because even data mining specialists or practitioners may not be familiar with all available measures.

In practice, the frequency of occurrence may not express the semantics of applications, because the user's interest may be related to other factors, such as cost, profit, or aesthetic value. For example, simply choosing the frequent itemsets does not reflect the impact of any factor except the frequency of the items. The usefulness of the support measure is reduced by problems with the quantity and quality of the mining results. First, a huge number of frequent itemsets that are not interesting to the user are often generated when the minimum support is low. For example, there may be thousands of combinations of products that occur in 1% of the

UBDM'06, August 20, 2006, Philadelphia, Pennsylvania, USA. Copyright ©2006 ACM 1-59593-440-5/06/0008...$5.00.

transactions. If too many uninteresting frequent itemsets are found, the user is forced to do additional work to select the rules that are indeed interesting. Second, the quality problem is that support, as defined based on the frequency of itemsets, is not necessarily an adequate measure of a typical user's interest. A sales manager may not be interested in frequent itemsets that do not generate significant profit. In other word, frequent itemsets may only contribute a small portion of the overall profit, whereas non-frequent itemsets may contribute a large portion of the profit [12]. The following example shows that support based itemset mining may lead to some high profit itemsets not being discovered due to their low support.

Example 1. Consider the small transaction dataset shown in Table 1 and the unit profit for the items shown in Table 2. Each value in the transaction dataset indicates the quantity sold of an item. Using Table 1 and 2, the support and profit for all itemsets can be calculated (see Table 3). For example, since for the 10 transactions in Table 1, only two transactions, t_8 and t_9, include both items B and D, the support of the itemset BD is $2/10 = 20\%$. Since t_8 includes one B and one D, and t_9 includes one B and ten Ds, a total of two Bs and eleven Ds appear in transactions containing the itemset BD. Using the Table 2, the profit for each item B is 100 and the profit for each item D is 1. Thus, the profit of the itemsets BD could be considered to be $2 \times 100 + 11 \times 1 = 211$. The profit of the other itemsets in Table 3 can be obtained in a similar fashion. Supposing that the minimum support is 40%, the frequent itemsets in Table 3 are D, A, AD, and C, but the four most profitable itemsets are BD, B, AC, and CD, all of which are infrequent itemsets.

Transaction ID	Item A	Item B	Item C	Item D
t_1	4	0	1	0
t_2	2	0	0	6
t_3	0	0	1	30
t_4	3	0	0	5
t_5	1	0	0	6
t_6	4	0	2	10
t_7	2	0	0	8
t_8	1	1	1	1
t_9	0	1	0	10
t_{10}	5	0	0	9

Table 1: A transaction dataset.

Item Name	Profit ($)
Item A	5
Item B	100
Item C	38
Item D	1

Table 2: The profit table for the items.

In general, a pattern that is of interest to one user may not be of interest to another user, since users have different levels of interest in patterns. The support measure reflects the frequency of combinations of items, but it does not reflect their semantic significance. Thus, a user may incur a high computational cost that is disproportionate to what the user wants and gets [14]. A natural way for interesting measure may allow a user to express his or her concern about the usefulness of results since only the user know his or her information need. That is, to allow data mining to further its impact on real-world applications, it is appropriate to consider user-specified interestingness, which bring more semantics of applications into data mining process and evaluate how user's expectation affect the data mining process.

Itemsets	Support (%)	Profit ($)
A	**80**	110
B	20	**200**
C	**40**	190
D	**90**	85
AB	10	105
AC	30	**197**
AD	**70**	135
BC	10	138
BD	20	**211**
CD	30	**193**
ABC	10	143
ABD	10	106
ACD	20	150
BCD	10	139
$ABCD$	10	144

Table 3: The support, and the profits of all itemsets.

To make clear the opportunity for a unified framework, we survey measures of interestingness for utility based data mining of itemsets. *Utility based data mining* refers to allowing a user to conveniently express his or her perspectives concerning the usefulness of patterns as utility values and then finding patterns with utility values higher than a threshold [20]. A pattern is of utility to a person if its use by that person contributes to reaching a goal. People may have differing goals concerning the knowledge that can be extracted from a data set. For example, one person may be interested in finding the sales with the most profit in a transaction data set. Another person may be interested in finding the largest increase in gross sales. This kind of interestingness is based on user-defined utility functions in addition to the raw data [4, 5, 6, 11, 13, 16, 20]. In fact, to achieve a user's goal, two types of utilities for items may need to be identified. The *transaction utility* of an item is directly obtained from the information stored in the transaction dataset. For example, the quantity of an item in Table 1 is a kind of transaction utility. The *external utility* of an item is given by the user. It is based on information not available in the transaction dataset. For example, a user's beliefs about the profit associated with items is expressed in Table 2. External utility often reflects user preference and can be represented by a utility table or utility function. By combining a transaction dataset and a utility table (or utility function) together, the discovered patterns will better match a user's expectations than by only considering the transaction dataset itself. To find patterns that conform to a user's interests, in this paper, we present a unified framework to show how utility measures are incorporated into data mining process by defining a unified utility function. Furthermore, three mathematical properties of this unified utility function are identified to allow the time and space costs of the mining algorithms to be reduced.

The remainder of this paper is organized as follows. In Section 2, we survey utility based measures for mining itemsets.

A framework for incorporating these utility measures in the data mining process is presented in Section 3. In Section 4, the mathematical properties of utility based measures are identified. Finally, conclusions are drawn in Section 5.

2. UTILITY BASED MEASURES

Researchers have proposed interestingness measures for various kinds of patterns, analyzed their theoretical properties, evaluated them empirically, and proposed strategies for selecting appropriate measures for particular domains and requirements. In data mining research, most interestingness measures have been proposed for evaluating itemsets and association rules. In this paper, we concentrate on interestingness measures that depend on the utility (usefulness) of the itemsets.

We begin by reviewing pertinent notions used for itemset mining. Adapting from the notation used in the descriptions of other itemset mining approaches [5, 16], we let $I = \{i_1, \ldots, i_p, i_q, \ldots, i_m\}$ be a set of items, where each item is associated with an attribute of a transaction dataset T. Each transaction t_q in T is a subset of I. An itemset S is a subset of I, i.e., $S \subseteq I$. To simplify notation, we sometimes write an itemset $\{i_1, \ldots, i_k\}$ as $i_1 \ldots i_k$; e.g., $ABCD$ represents itemset $\{A, B, C, D\}$. We denote the support value of itemset S as $s(S)$ and the utility value of itemset S as $u(S)$.

Definition 1. The *transaction set of an itemset* S, denoted T_S, is the set of transactions that contain itemset S, i.e., $T_S = \{t_q \mid S \subseteq t_q, t_q \in T\}$.

For instance, consider the transaction dataset shown in Table 1, supposing itemset S is $S = AD$. By definition, $T_S = \{t_2, t_4, t_5, t_6, t_7, t_8, t_{10}\}$.

A *utility based measure* is a measure that takes into consideration not only the statistical aspects of the raw data, but also the utility of the mined patterns. Motivated by the decision theory, Shen et al. stated that the "interestingness of a pattern = probability + utility" [17]. Based on the user's specific objectives and the utility of the mined patterns, utility-based mining approaches may be more useful in real applications, especially in decision making problems.

In this section, we review utility based measures for itemsets. Since we use a unified notation for all methods, some representations differ from those used in the original papers.

The simplest method to incorporate utility is called *weighted itemset mining*, which assigns each item a weight representing its importance [5, 11]. For example, the weights may correspond the profitability of different items; e.g., a computer (item A) may be more important than a phone (item B) in terms of profit. Weights assigned to items are also called horizontal weights [13]. The weights can represent the price or profit of a commodity. In this scenario, two measures are proposed to replace *support*. The first one is called *weighted support*, which is defined as

$$support_w(S) = (\sum_{i_p \in S} w_p) s(S), \qquad (1)$$

where w_p denotes the weight of item i_p.

The first factor of the weighted support measure has a bias towards the rules with more items. When the number of the items is large, even if all the weights are small, the total weight may be large. The second measure, *normalized weighted support*, is proposed to reduce this bias and is defined as

$$support_{nw}(S) = \frac{1}{|S|}(\sum_{i_p \in S} w_p) s(S), \qquad (2)$$

where $|S|$ is the number of items in the itemset S.

The traditional support measure is a special case of normalized weighted support, because when all weights for items are equal to 1, the normalized weighted support is identical to support. The Weighted Items (WI) approach [5] and the Value Added Mining (VAM) approach [11] use weighted items to capture the semantic significance of itemsets at the item level. Unlike frequent itemset mining, which treats all items uniformly, both of these approaches assume that items in a transaction dataset (columns in the table) have different weights to reflect their importance to the user.

Lu et al. proposed another data model by assigning a weight to each transaction [13]. The weight represents the significance of the transaction in the data set. Weights assigned to transactions are also called vertical weights [13]. For example, the weight can reflect the transaction time, i.e., more recent transactions can be given greater weights. Based on this model, *vertical weighted support* is defined as

$$support_v(S) = \frac{\sum_{t_q \in T_S} w_q}{\sum_{t \in T} w}, \qquad (3)$$

where w_q and w denote the vertical weight for transactions t_q and t, respectively.

The mixed weighted model [13] uses both horizontal and vertical weights. In this model, each item is assigned a horizontal weight and each transaction is assigned a vertical weight. *Mixed weighted support* is defined as

$$support_m(S) = support_{nw}(S) \cdot support_v(S). \qquad (4)$$

Both $support_v$ and $support_m$ are extensions of the traditional support measure. If all vertical and horizontal weights are set to 1, both $support_v$ and $support_m$ are identical to support.

Objective oriented utility based association (OOA) mining allows a user to set objectives for the mining process [17]. In this method, the attributes are partitioned into two groups, the target attributes and the non-target attributes. A *non-target attribute* (called an *nonobjective attribute* in [17] is only permitted to appear in the antecedents of association rules. A *target attribute* (called an *objective attribute* in [17]) is only permitted to appear in the consequents of rules. The target attribute-value pairs are assigned utility values. The mining problem is to find frequent itemsets of non-target attributes, such that the utility values of their corresponding target attribute-value pairs are above a threshold. For example, in Table 4 obtained from [17], *Treatment* is a non-target attribute, while *Effectiveness* and *Side-effect* are two target attributes. The goal of the mining problem is to find treatments with high effectiveness and mild side effects. The utility measure is defined as

$$u(S) = \frac{1}{s(S)} \sum_{t_q \in T_S} u(t_q), \qquad (5)$$

where S is the non-target itemsets to be mined (the *Treatment* attribute-value pairs in the example) and $u(t_q)$ denotes the utility of transaction t_q. The function $u(t_q)$ is defined as

$$u(t_q) = \sum_{i_p \in C_q} f(i_p), \qquad (6)$$

where C_q denotes the set of target items in transaction t_q and $f(i_p)$ is the utility function of item i_p, which denotes the utility associated with i_p. If there is only one target attribute and its weight equals to 1, $\sum_{t_q \in T_S} u(t_q)$ is identical to $s(S)$, and hence $u(S)$ equals to 1.

Continuing the example, we assign the utility values to the target attribute-value pairs shown in Table 5 and accordingly obtain the utility values for the treatments shown in Table 6. For example, *Treatment 5* has the greatest utility value 1.2, and therefore, it best meets the user specified target.

TID	Treatment	Effectiveness	Side-effect
t_1	1	2	4
t_2	2	4	2
t_3	2	4	2
t_4	2	2	3
t_5	2	1	3
t_6	3	4	2
t_7	3	4	2
t_8	3	1	4
t_9	4	5	2
t_{10}	4	4	2
t_{11}	4	4	2
t_{12}	4	3	1
t_{13}	5	4	1
t_{14}	5	4	1
t_{15}	5	1	1
t_{16}	5	3	1

Table 4: A medical dataset.

The approach of Lu et al. [13] and OOA mining approach [6, 17] both capture the semantic significance of itemsets at the transaction level. They assume that transactions in a dataset (rows in the table) have associated utility values that reflect their importance to the user.

Effectiveness			Side-effect		
Value	Meaning	Utility	Value	Meaning	Utility
5	Much better	1	4	Very serious	-0.8
4	Better	0.8	3	Serious	-0.4
3	No effect	0	2	A little	0
2	Worse	-0.8	1	Normal	0.6
1	Much worse	-1			

Table 5: Utility values for *Effectiveness* and *Side-effect.*

Itemset	Utility
Treatment =1	-1.6
Treatment =2	-0.25
Treatment =3	-0.066
Treatment =4	0.8
Treatment =5	1.2

Table 6: Utilities of the items.

Hilderman et al. proposed the Itemset Share framework that takes into account weights on both attributes and attribute-value pairs [7]. The precise impact of the purchase of an itemset can be measured by the itemset *share*, the fraction of some overall numerical value, such as the total value of all items sold. For example, in a transaction data set, the weight on an attribute could represent the price of a commodity, and the weight on an attribute-value pair could represent the quantity of the commodity in a transaction. Based on this model, in the Itemset Share framework, support is generalized. The *count support* for itemset S is defined as

$$count\_sup(S) = \frac{\sum_{t_q \in T_S} \sum_{i_p \in S} w(i_p, t_q)}{\sum_{t \in T} \sum_{i \in I} w(i, t)}, \qquad (7)$$

where $w(i_p, t_q)$ denotes the weight of attribute i_p for transaction t_q and $w(i_p, t_q) > 0$.

Similarly, the *amount support* is defined as

$$amount\_sup(S) = \frac{\sum_{t_q \in T_S} \sum_{i_p \in S} w(i_p, t_q) w(i_p)}{\sum_{t \in T} \sum_{i \in I} w(i, t) w(i)}, \qquad (8)$$

where $w(i_p)$ is the weight for attribute i_p and $w(i_p) > 0$.

Based on the data model in [7], Yao et al. proposed another utility measure [20], defined as

$$u(S) = \sum_{t_q \in T_S} \sum_{i_p \in S} w(i_p, t_q) w(i_p), \qquad (9)$$

where $w(i_p, t_q)$ denotes the utility value of attribute i_p for transaction t_q, $w(i_p)$ denotes the utility value of attribute i_p, $w(i_p, t_q) > 0$ and $w(i_p) > 0$.

This utility function is similar to amount support, except that it represents a utility value, such as the profit in dollars, rather than a fraction of the total weight of all transactions in the data set.

The Itemset Share (IS) approach [4] and the approach of Yao et al. [20] capture the semantic significance of numerical values that are typically associated with the individual items in a transaction dataset (cells in the table).

Table 7 summarizes the utility measures discussed in this section by listing the name of each measure and its data model. The data model describes how the information relevant to the utility is organized in the data set. All these measures are extensions of the support and confidence measures. No single utility measure is suitable for every application, because applications have different objectives and data models. Given a data set, one could choose a utility measure by examining the data models for the utility measures given in Table 7. For example, if one has a data set with weights for each row, then one might choose the vertical weighted support measure. By checking Table 7 carefully, we find that the difference among these models are: (1) different levels of granularity (item level, transaction level, and cell level) are used to specify the semantic significance of itemsets, and (2) different pruning strategies are developed according to the properties of these measure functions. For (1), we present a unified framework for utility base measures that incorporates existing utility based measures into data mining process in Section 3. For (2), we summarize the mathematical properties of the unified framework for utility base measures in Section 4.

3. A UNIFIED FRAMEWORK FOR UTILITY BASED MEASURES

During the knowledge discovery process, utility based measures can be used in three ways, which we call the roles of the utility based measures. Figure 1 shows these three roles. First, measures can be used to prune uninteresting patterns during the data mining process to narrow the search space and thus improve the mining efficiency. For example, a threshold for support can be used to filter out patterns with low support during the mining process and thus improve efficiency [2]. Similarly, a utility threshold can be defined and used for pruning patterns with low utility values [20]. Secondly, measures can be used to rank the patterns according to the order of their interestingness scores. Thirdly, measures can be used during post processing to select the interesting patterns. For example, after the data mining process, we can use the chi-square test to select the rules that have significant correlations [3]. The second and third approaches can also be combined by first filtering the patterns and then ranking them. For the second or third approach, utility based measures need not be incorporated into the data mining algorithm. In this paper, we concentrate on first method since it can improve the mining efficiency by reducing the time and space costs of the mining algorithm.

Now, formal definitions of key terms used in our unified utility framework for utility measures for mining itemsets are presented.

We denote the utility value of an itemset S as $u(S)$, which will be described in more detail shortly.

Definition 2. The *utility constraint* is a constraint of the form $u(S) \geq minutil$.

Definition 3. An itemset S is a *high utility pattern* if $u(S) \geq minutil$, where $minutil$ is the threshold defined by the user. Otherwise, S is a *low utility itemset.*

Based on the utility constraint, the unified utility framework for utility measures is defined as follows.

Definition 4. The *utility based itemset mining problem* is to discover the set H of all high utility itemsets, i.e.,

$$H = \{S \mid S \subseteq I, u(S) \geq minutil\}. \qquad (10)$$

For example, consider the itemsets in Table 3. If $u(S)$ is the profit of an itemset S and $minutil = 150$, then $H = \{B, C, AC, BD, CD, ACD\}$.

According to the survey presented in Section 2, $u(S)$ plays a key role in specifying utility based data mining problems. Different utility measures use different formulas for $u(S)$.

Now, we show how to define $u(S)$ in terms of a user defined utility function f. In Example 1, the profit of an itemset reflects a store manager's goal of discovering itemsets producing significant profit (e.g., $minutil = 150$). A user judges BD to be useful, since the profit of itemset BD is greater than $minutil$. We observe that the semantic meaning of profit can be captured by a function $f(x, y)$, where x is the quantity sold of an item and y is the unit profit of an item. The usefulness of an itemset is quantified as the product of x and y, namely, $f(x, y) = x \cdot y$. The value of x can be obtained from the transaction dataset and depends only on the underlying dataset [18]. On the other hand, the value of y is often not available in a transaction dataset and may depend on the user who examines the pattern [18]. Thus, in this case, the significance of an item is measured by two parts. One is the statistical significance of the item measured by parameter x, which is an objective term independent of its intended application. The other part is the semantic significance of the item measured by parameter y, which is a subjective term dependent on the application and the user. As a result, $f(x, y)$ combines objective and subjective measures of an item together. The combination captures the significance of the itemset for this application, which reflects not only the statistical significance but also the semantic significance of the itemset.

Definition 5. The *transaction utility value of an item*, denoted x_{pq}, is the value of an attribute associated with an item i_p in a transaction t_q.

For example, in Table 1, the quantity sold values in the transactions are the transaction utility values. If $i_4 = D$, then $x_{43} = 30$ is the transaction utility value of item D in transaction t_3.

In this paper, we restrict transaction utility variable values to numerical values, because, typically, transaction utility information can be represented in this form.

Measures	**Data models**	**Extension of**
Weighted support	Weights for items	Support
Normalized weighted support	Weights for items	Support
Vertical weighted support	Weights for transactions	Support
Mixed weighted support	Weights for both items and transactions	Support
OOA Target and non-target attributes	Weights on transaction for target attributes	Support
Count support	Weights for items and cells in data set	Support
Amount support	Weights for items and cells in data set	Support
Count confidence	Weights for items and cells in data set	Confidence
Amount confidence	Weights for items and cells in data set	Confidence
Yao et al.'s	Weights for items and cells in data set	Support

Table 7: Utility based interestingness measures.

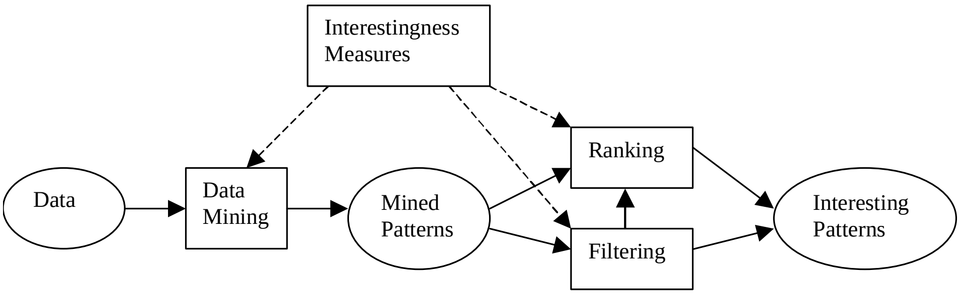

Figure 1: Roles of utility based measures.

Definition 6. The *external utility value of an item*, denoted y_p, is a real number assigned by the user such that for any two items i_p and i_q, y_p is greater than y_q iff the user prefers item i_p to item i_q.

The definition indicates that a external utility value is associated with a specific value in a domain to express user preference. In practice, the value of y_p is assigned by the user according to his interpretation of domain specific knowledge measured by some utility factors, such as cost, profit, or aesthetic value. For example, let $i_1 = A$ and $i_2 = B$. Using the Table 2, we have $y_1 = 5$ and $y_2 = 100$. The inequality $y_2 > y_1$ reveals that the store manager prefers item B to item A, since each item B earns more profit than each item A.

By obtaining the transaction utility value x_{pq} from a transaction dataset and the external utility value y_p from the user, a utility function to express the significance of an itemset can be defined as a two dimensional function $f(x, y)$.

Definition 7. A *utility function* f is a function $f(x, y) : (R, R) \rightarrow R$, where R is the set of real numbers.

Example 2. Consider the transaction dataset in Table 1 and the profit table in Table 2. Let items i_1, i_2, i_3, and i_4 be items A, B, C, and D, respectively. Suppose that the user defines utility function $f(x_{pq}, y_p)$ as $f(x_{pq}, y_p) = x_{pq} \cdot y_p$, where x_{pq} is the quantity sold of an item i_p in a transaction t_q, and y_p is the unit price of the item i_p. Then $f(x_{11}, y_1) = 4\times 5 = 20$, which indicates that the supermarket earns \$20 by selling four As in transaction t_1. Similarly, $f(x_{21}, y_2) = 0$, $f(x_{31}, y_3) = 1 \times 38 = 38$, and $f(x_{41}, y_4) = 0$.

The utility value of an item is the sum of the values of the utility function for each transaction.

Definition 8. The *utility value of an item i_p in an itemset S*, denoted $l(i_p, S)$, is the sum of the values of the utility function $f(x_{pq}, y_p)$ for each transaction t_q in T_S, i.e.,

$$l(i_p, S) = \sum_{t_q \in T_S} f(x_{pq}, y_p). \tag{11}$$

For example, consider the transaction dataset in Table 1 with the profit table in the Table 2. If $S = ACD$, then $T_S = \{t_6, t_8\}$, thus $l(A, S) = 4 \times 5 + 1 \times 5 = 25$.

The utility value of an itemset is represented by the sum of the utility values of every item in the itemset.

Definition 9. The *utility value of an itemset S*, denoted $u(S)$, is the sum of the utility value of each item in S, i.e.,

$$u(S) = \sum_{i_p \in S} l(i_p, S). \tag{12}$$

By substituting Equation 11 into Equation 12, we obtain

$$u(S) = \sum_{i_p \in S} \sum_{t_q \in T_S} f(x_{pq}, y_p). \tag{13}$$

For example, given $f(x_{pq}, y_p) = x_{pq} \cdot y_p$, for itemset $S = ACD$, we have $T_S = \{t_6, t_8\}$, then $u(S) = l(A, S) + l(C, S) + l(D, S) = 5 \times 5 + 3 \times 38 + 11 \times 1 = 150$.

Equation 13 indicates that user plays an important role in utility based itemset mining process since a user can measure the semantic significance of the itemset by using his own utility function $f(x, y)$. Therefore, an itemset that is of interest to one user, may be of no interest to another user, since users have different levels of interest in itemsets, as expressed by their utility functions. In other word, different itemsets may be discovered for two users according to their interests, as expressed by their utility functions.

Based on the utility formulation of an itemset (Equation (13)), an efficient algorithm, called UMining [20], has been developed to find the high profit itemsets from a dataset.

Now we show that the utility function $f(x, y)$ is a unified utility function. Let c be a constant. Table 8 summarizes the semantic significance of this unified utility function at the item level, the transaction level, and the cell level. Table 9 shows how to use our unified utility function to represent all existing utility based measures described in Section 2. In this framework, by defining deferent $f(x, y)$, several existing utility based measures can be obtained.

4. MATHEMATICAL PROPERTIES OF UTILITY BASED MEASURES

In this section, we analyze the mathematical properties of the utility function $f(x, y)$ to facilitate the design of efficient mining algorithms that will reduce the time and space costs of the mining process.

Three important mathematical properties of utility based measures, namely, the anti-monotone (or monotone) property, the convertible property, and the upper bound property, have been identified and used in existing utility based measures [1, 2, 5, 6, 13, 9, 16, 20].

Definition 10. [15]. A constraint C is *anti-monotone* iff whenever an itemset S violates a constraint C, so does any superset of S. A constraint C is *monotone* iff whenever an itemset S satisfies a constraint C, so does any superset of S.

By definition, the Apriori property [2] that applied to the support measure is a special case of the anti-monotone property that focuses only on the support constraint.

Definition 11. [15]. An itemset $S_1 = i_1 \ldots, i_m$ is a *prefix itemset* of itemset $S_2 = i_1 \ldots, i_n$ if the items in S_1 and S_2 are listed in the same order and $m \ \leq n$.

For example, suppose we are given an itemset $ABCD$. By Definition 11, itemsets A, AB, and ABC are prefix itemsets of $ABCD$ with respect to the order $\langle A, B, C, D\rangle$

Based on the prefix itemsets of an itemset, the convertible property of the itemset is defined as follows.

Definition 12. [15]. A constraint C is *convertible anti-monotone* w.r.t. an order $\mathcal{O}$ on items if and only if whenever an itemset S satisfies property P, so do any prefix itemsets of S. A constraint C is *convertible monotone* w.r.t. an $\mathcal{O}$ on items if and only if whenever an itemset S violates property P, so do any prefix itemsets of S. A constraint C is *convertible* w.r.t. an order $\mathcal{O}$ if and only if it is convertible anti-monotone or convertible monotone w.r.t. the order $\mathcal{O}$.

The following example shows a constraint that is convertible.

Example 3. Consider the profit table for the items shown in Table 2. Let $avg(S) \geq 30$ be a constraint on the average profit of the itemset S. We have $avg(ABCD) = (5 + 100 + 38 + 1)/4 = 36$. If the items are sorted in unit profit descending order, we get $\langle B, C, A, D\rangle$. The itemset $BCAD$ has BCA, BC, and B as its prefix itemsets w.r.t. the order $\langle B, C, A, D\rangle$. Then we have $avg(BCA) = 47.67$, $avg(BC) = 69$, and $avg(B) = 100$. The average profit of the itemset $BCAD$ is greater than 30, as are the average profits for its all prefix itemsets according to the order $\langle B, C, A, D\rangle$. By definition, the constraint $avg(ABCD) \geq 30$ is convertible anti-monotone w.r.t. the order $\langle B, C, A, D\rangle$. Thus, it is convertible w.r.t. the order $\langle B, C, A, D\rangle$.

Definition 13. [15]. (Prefix monotone functions) Given an order O over a set of items I, a function f: $2^I \rightarrow R$ is a *prefix (monotonically) increasing function* w.r.t. O if and only if for every itemset S and its prefix S' w.r.t. O, $f(S') \leq f(S)$. A function g: $2^I \rightarrow R$ is a *prefix (monotonically) decreasing function* w.r.t. O if and only if for every itemset S and its prefix S' w.r.t. O, $g(S') \geq g(S)$.

Theorem 1. [15]. A constraint $u(S) \geq v$ (resp. $u(S) \leq v$) is convertible anti-monotone (resp., monotone) if and only if u is a prefix decreasing function. Similarly, $u(S) \geq v$ (resp. $u(S) \leq v$) is convertible monotone (resp., anti-monotone) if and only if u is a prefix increasing function.

Before defining an upper bound property for utility based measures, we first introduce some more terminology.

Definition 14. A $k-$*itemset*, denoted as S^k, is an itemset of k distinct items.

Definition 15. The set of all $(k-1)-$itemsets of a k-itemset S^k, denoted L^{k-1}, is the set $\{S^{k-1} \mid S^{k-1} \subset S^k\}$

For the 4-itemset $S^4 = ABCD$, by Definition 15, we have $L^3 = \{ACD, ABD, ABC, BCD\}$.

Definition 16. A *nonnegative utility function* f is a function $f(x, y) : (R, R) \rightarrow R^+$, where R is the set of real numbers, and R^+ is the set of nonnegative real numbers.

A function $f_1(x, y)$ with range $[-n, m]$, where $n, m \geq 0$, can be transformed to a nonnegative function by adding n to

Semantic Significance	**Utility Function** $f(x_{pq}, y_p)$	**Utility Value** $u(S)$
no semantic significance	$\sum_{i_p \in S} f(x_{pq}, y_p) = 1$	$u(S) = s(S)$
semantic significance on item	$\sum_{t_q \in T_S} f(x_{pq}, y_p) = s(S)$	$u(S) = \sum_{i_p \in S} f(i_p) \cdot s(S)$
semantic significance on transaction	$\sum_{i_p \in S} f(x_{pq}, y_p) = c$	$u(S) = c \cdot \sum_{t_q \in T_S} f(t_q)$
semantic significance on cell	$\sum_{i_p \in S} \sum_{t_q \in T_S} f(x_{pq}, y_p) \geq 0$	$u(S) = \sum_{i_p \in S} \sum_{t_q \in T_S} f(x_{pq}, y_p)$

Table 8: Semantic significance of utility function.

Measures	**Unified Utility Function** $f(x_{pq}, y_p)$
Support	$\sum_{i_p \in S} f(x_{pq}, y_p) = 1$
Weighted support	$\sum_{i_p \in S} f(x_{pq}, y_p) = w_p$
Normalized weighted support	$\sum_{i_p \in S} f(x_{pq}, y_p) = w_p / \lvert S \rvert$
Vertical weighted support	$\sum_{t_q \in T_S} f(x_{pq}, y_p) = w_q / c$
Mixed weighted support	$f(x_{pq}, y_p) = w_p \cdot w_q / c$
OOA Target and non-target attributes	$\sum_{i_p \in S} f(x_{pq}, y_p) = u_q(S)$
Count support	$f(x_{pq}, y_p) = w(i_p, t_q) / c$
Amount support	$f(x_{pq}, y_p) = w(i_p, t_q) \cdot w(i_p) / c$
Yao et al.'s	$f(x_{pq}, y_p) = w(i_p, t_q) \cdot w(i_p)$

Table 9: Utility based interestingness measures.

all values. Also a nonpositive function $f_2(x, y) \leq 0$ can be transformed to its absolute value, namely $|f_2(x, y)|$ such that $|f_2(x, y)| \geq 0$. Thus, all results obtained for nonnegative utility function can also be applied to function f_1 or f_2. Note that f could be a monotone, a non monotone, a convertible, or non convertible function.

Using Definitions 14-16, an upper bound on the utility value of the itemset S^k can be obtained as follows.

Theorem 2. (Utility Upper Bound Property) [20]. Let $u(S^k)$ be the utility value of a $k-$itemset S^k defined according to Equation(13) based on a nonnegative utility function f. Then the following property holds

$$u(S^k) \leq \frac{\sum_{S^{k-1} \in L^{k-1}} u(S^{k-1})}{k-1} \qquad (14)$$

Example 4. For a 4-itemset $S^4 = ABCD$, by Definition 15, we obtain $L^3 = \{ACD, ABD, ABC, BCD\}$. Thus, by Theorem 2, we have

$$u(ABCD) \leq \frac{u(ABC) + u(ACD) + u(ABD) + u(BCD)}{3}.$$

It is important to realize that Theorem 2 indicates that the utility value of itemset S^k is limited by the utilities of all its subset of itemsets of size $(k-1)$.

By exploiting the anti-monotone (or monotone) property, the convertible property, and the upper bound property, efficient algorithms have been developed. More precisely, the Apriori algorithm [2] is based on the anti-monotone property. The FIC^A algorithm suggested by Pei et al. [16] is based on the convertible anti-monotone property, and the FIC^M algorithm suggested by Pei et al. [16] is based on the convertible monotone property. The UMining algorithm [20] is based on the upper bound property. All these algorithms reduced the number of the mined results by exploiting one of the properties of utility based measures.

Now, we consider the mathematical properties of the utility based measures discussed in Section 2. The Weighted Items approach [5] and the Value Added Mining approach [11] reflect the semantic significance of itemsets at the item level by defining different weights on items. Since there is always a decreasing order based on the weights of all items, a prefix monotone function can be defined as $\sum_{i_p \in S} f(i_p)$ for itemset S w.r.t. the descending order of the weights of the items, where $f(i_p)$ is the weight of the item i_p. The Vertical Weighted Support approach [13] and the OOA approach [6] capture the semantic significance of itemsets at the transaction level. Since there is always a decreasing order based on the weights of all transactions, a prefix monotone function is defined as $\sum_{t_q \in T_S} f(t_q)$ for itemset S w.r.t. the descending order on the weights of the transactions, where $f(t_q)$ is the weight of transaction t_q. Thus, the utility functions of the Weighted Items measure, the Value Added Mining, the Vertical Weighted Support measure, and the OOA measure satisfy the convertible property. The Mixed Weighted Support approach [13], the Itemset Share approach [4, 20] and Yao et al. [20] capture the semantic significance of itemsets at the cell level. Since these three approaches use a nonnegative utility function, by Theorem 2, the utility function of the Mixed Weighted Support measures, the Itemset Share measures (*count_sup amount_sup*), and Yao et al. satisfy the upper bound property. Table 10 summaries the mathematical properties of the utility based measures discussed in Section 2.

Utility Measures	**Mathematical Property**
Support	anti-monotone property
Weighted support	convertible property
Normalized weighted support	convertible property
Vertical weighted support	convertible property
Mixed weighted support	upper bound property
OOA non-target attributes	convertible property
Count support	upper bound property
Amount support	upper bound property
Yao et al.'s	upper bound property

Table 10: Mathematical properties of utility based measures.

Theorem 3. The mathematical properties of utility measures shown in Table 10 are correct.

Proof: For the support measure, Agrawal et al. [2] showed that it satisfies the anti-monotone property. Now, we prove that Equations (1), (2), (3), and (5) satisfy the convertible property. For Equations(1) and (2), a prefix monotone function can be defined w.r.t. the descending order of the weights of the items. By Theorem 1, they satisfy the convertible property. Similarly, for Equations (3) and (5), a prefix monotone function can be defined w.r.t. the descending order of the weights of the transactions. By Theorem 1, they also satisfy the convertible property. Now we prove that Equations (4), (7), (8) and (9) satisfy upper bound property. For Equations (4), $w_p > 0$. For Equations(7), (8) and (9), $w(i_p, t_q) > 0$ and $w(i_p) > 0$. Thus, Equations (4), (7), (8) and (9) are nonnegative functions. By Theorem 2, they satisfy upper bound property. □

Theorem 3 indicates that we can design an efficient pruning strategy for these utility measures by using the identified mathematical properties. In other words, it is possible to incorporate these properties into the algorithms used for these utility measures.

5. CONCLUSIONS

This paper formalizes all existing utility measures for itemset mining that are known to the authors. We provide three research contributions towards utility based itemset mining in this paper.

First, we formalize the semantic significance of utility measures. Existing utility based measures employ various representations for the semantics significance of applications for the same dataset, which lead to different measures and procedures for determining interestingness. Based on the semantics of applications, we classified the utility based measures into three categories, namely, item level, transaction level, and cell level.

The second contribution is that we defined a unified utility function to represent all existing utility based measures, as shown in Table 9. According to our classification, the transaction utility and the external utility of an itemset is defined, and then a general unified framework was developed to define a unifying view of the utility based measures for itemset mining. That is, existing utility based measures can be represented by this unified utility function.

The third contribution is that the mathematical properties of the utility based measures were identified and analyzed. These properties can facilitate the design of efficient pruning strategies for utility based itemset mining and help current itemest algorithms to reflect the different utilities by using different pruning strategies.

Future research could consider a method for automating the elicitation of different itemset utilities, and then incorporating these different utilities into current itemset mining algorithms [2, 20]. In addition, to make our utility function more practicable, the unified utility function could be extend to a fuzzy utility function for fuzzy utility values.

6. REFERENCES

[1] Agrawal R., Imielinski T., and Swami, A.N. Mining association rules between sets of items in large databases. In *Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data*, Washington, D.C., 1993, 207–216.

[2] Agrawal R. and Srikant, R. Fast algorithms for mining association rules. In *Proceedings of the 20th International Conference on Very Large Databases*, Santiago, Chile, 1994, 487–499.

[3] Bay, S.D. and Pazzani, M.J. Detecting change in categorical data: Mining contrast sets. In *Proceedings of the Fifth International Conference on Knowledge Discovery and Data Mining*, San Diego, USA, 1999, 302-306.

[4] Barber, B. and Hamilton, H.J. Extracting share frequent itemsets with infrequent subsets. *Data Mining and Knowledge Discovery*, 7(2), 2003, 153–185.

[5] Cai, C.H., Fu, A.W.C., Cheng, C.H., and Kwong, W.W. Mining association rules with weighted items. In *Proceedings of the IEEE International Database Engineering and Applications Symposium*, Cardiff, UK, 1998, 68–77.

[6] Chan, R., Yang, Q., and Shen, Y.D. Mining high utility itemsets. In *Proceedings of the 3rd IEEE International Conference on Data Mining*, Melbourne, Florida, 2003, 19-26.

[7] Hilderman, R.J., Carter, C.L., Hamilton, H.J., and Cercone, N. Mining market basket data using share measures and characterized itemsets. In *Proceedings of the Second Pacific Asia Conference on Knowledge Discovery in Databases*, Melbourne, 1998, 72-86.

[8] Hilderman, R.J. and Hamilton, H.J. Measuring the interestingness of discovered knowledge: A principled approach. *Intelligent Data Analysis*, 7(4), 2003, 347-382.

[9] Mannila, H., Toivonen, H., and Verkamo, A.I. Efficient algorithms for discovering association rules. In *Proceeding of the AAAI Workshop on Knowledge Discovery in Databases*, Seattle, Washington, 1994, 181-192.

[10] Geng, L. and Hamilton, H.J. Interestingness measures for data mining: A survey. *ACM Computing Surveys.* To appear.

[11] Lin, T.Y., Yao, Y.Y., and Louie, E. Value added association rules. In *Proceedings of the 6th Pacific-Asia Conference on Knowledge Discovery and Data Mining*, Taipei, Taiwan, 2002, 328-333.

[12] Liu, Y., Liao, W.K., and Choudhary, A. A Fast High Utility Itemsets Mining Algorithm. In *Proceedings of the Workshop on Utility-Based Data Mining in conjunction with the 11th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD 2005)*, Chicago, Illinois, 2005, 90-99.

[13] Lu, S., Hu, H., and Li, F. Mining weighted association rules. *Intelligent Data Analysis*, 5(3), 2001, 211–225.

[14] Ng, R., Lakshmanan, L. V. S., Han, J., and Pang, A. Exploratory mining and pruning optimizations of constrained associations rules. In *Proceedings of the 1998 ACM SIGMOD International Conference on Management of Data*, Seattle, Washington, 1998, 13-24.

[15] Pei, J. and Han, J. Can we push more constraints into frequent pattern mining? In *Proceedings of the 6th International Conference on Knowledge Discovery and Data Mining*, Boston, Massachusetts, 2000, 350–354.

[16] Pei, J., Han, J., and Lakshmanan, L. V. S. Pushing convertible constraints in frequent itemset mining. *Data Mining and Knowledge Discovery*, 8(3), 2004, 227-252.

[17] Shen, Y. D., Zhang, Z. and Yang, Q. Objective-oriented utility-based association mining. In *Proceedings of the 2002 IEEE International Conference on Data Mining*, Maebashi City, Japan, December 2002, 426-433.

[18] Silberschatz A. and Tuzhilin, A. On subjective measures of interestingness in knowledge discovery. In *Proceedings of the 1st International Conference on Knowledge Discovery and Data Mining*, Montreal, Quebec, Canada, 1996, 275–281.

[19] Tan, P.N., Kumar, V., and Srivastava, J. Selecting the right objective measure for association analysis. *Information Systems*, 29(4), 2004, 293-313.

[20] Yao H. and Hamilton, H.J. Mining itemset utilities from transaction databases, *Data & Knowledge Engineering.* To appear.

[21] Yao Y.Y. and Zhong N. An analysis of quantitative measures associated with rules, Methodologies for Knowledge Discovery and Data Mining. In *Proceedings of the 3rd Pacific-Asia Conference on Knowledge Discovery and Data Mining*, Beijing, China, 1999, 26-28.

[22] Zhang, H., Padmanabhan, B., and Tuzhilin, A. On the discovery of significant statistical quantitative rules. In *Proceedings of the Tenth International Conference on Knowledge Discovery and Data Mining*, Seattle, USA, August 2004, 374-383.