Extracting Diverse Patterns with Unbalanced Concept Hierarchy

M. Kumara Swamy, P. Krishna Reddy, and Somya Srivastava

Centre of Data Engineering
International Institute of Information Technology-Hyderabad (IIIT-H)
Gachibowli, Hyderabad, India - 500032
kumaraswamy@research.iiit.ac.in, pkreddy@iiit.ac.in, somya@amazon.com

Abstract. The process of frequent pattern extraction finds interesting information about the association among the items in a transactional database. The notion of *support* is employed to extract the frequent patterns. Normally, in a given domain, a set of items can be grouped into a category and a pattern may contain the items which belong to multiple categories. In several applications, it may be useful to distinguish between the pattern having items belonging to multiple categories and the pattern having items belonging to one or a few categories. The notion of diversity captures the extent the items in the pattern belong to multiple categories. The items and the categories form a concept hierarchy. In the literature, an approach has been proposed to rank the patterns by considering the balanced concept hierarchy. In a real life scenario, the concept hierarchies are normally unbalanced. In this paper, we propose a general approach to calculate the rank based on the diversity, called *drank*, by considering the unbalanced concept hierarchy. The experiment results show that the patterns ordered based on *drank* are different from the patterns ordered based on *support*, and the proposed approach could assign the *drank* to different kinds of unbalanced patterns.

Key words: data mining, association rules, frequent patterns, diversity, diverse rank, interestingness, concept hierarchy, algorithms

1 Introduction

In the field of data mining, the process of frequent pattern mining has been widely studied [1]. The related concepts of frequent pattern mining are as follows [2]. Let $I = \{i_1, i_2, \cdots, i_n\}$ be the set of n items and D be the database of m transactions. Each transaction is identified with unique identifier and contains n items. Let $X \subseteq I$ be a set of items, referred to as an item set or a *pattern*. A pattern that contains k items is a k-item pattern. A transaction T is said to contain X if and only if $X \subseteq T$. The *frequency* or *support* of a pattern X in D, denoted as $f(X)$, is the number of transactions in D containing X. The support X, denoted as $S(X)$, is the ratio of its frequency to the $|D|$ i.e., $S(X) = \frac{f(X)}{|D|}$. The pattern X is frequent if its support is not less than the user-defined minimum support threshold, i.e., $S(X) \geq minSup$.

The techniques to enumerate frequent patterns generates large number of patterns which could be uninteresting to the user. Research efforts are on to discover interesting frequent patterns based on constraints and/or user-interest by using various interestingness measures such as closed [3], maximal [4], top-k [5], pattern-length [6] and cost (utility) [7].

Normally, in a given domain, a set of items can be grouped into a category and a pattern may contain the items which belong to multiple categories. In several applications, it may be useful to distinguish between the pattern having items belonging to multiple categories and the pattern having items belonging to the one or a few categories. The existing frequent pattern extraction approaches do not distinguish the patterns based on the diversity. The notion of diversity captures the extent of items in the pattern belong to multiple categories. The items and the categories form concept hierarchy. In [8], an effort has been made to rank the patterns based on diversity by considering balanced concept hierarchy. However, in real life scenarios, the concept hierarchies are unbalanced. In this paper, we have proposed an approach to assign the diverse rank, called *drank*, to patterns by considering unbalanced concept hierarchy. The proposed approach is a general approach which can be applied to calculate *drank* value by considering both balanced and unbalanced concept hierarchies. Experiments on the real-world data set show that patterns ordered based on *drank* are different from the patterns ordered based on *support*, and the proposed approach could assign the *drank* to different kinds of unbalanced patterns.

In the literature, the concept hierarchies have been used to discover the generalized association rules in [9] and multiple-level association rules in [10]. In [11], a keyword suggestion approach based on the concept hierarchy has been proposed to facilitate the user's web search. The notion of *diversity* has been widely exploited in the literature to assess the interestingness of summaries [12],[13],[14]. In [15], an effort has been made to extend the *diversity-based* measures to assess the interestingness of the data sets using the diverse association rules. The diversity is defined as the variation in the items' frequencies. Such a method cannot be directly applied to rank the patterns based on the diversity. Moreover, the work in [15] has focused on comparing the data sets using diverse association rules. In this paper, we developed a framework to compute the diversity of patterns by analyzing the categories of items.

The rest of the paper is organized as follows. In the next section, we explain about concept hierarchy and diversity of pattern. In section 3, we explain the approach to computing the *drank* of a pattern by considering balanced concept hierarchy. In section 4, we present the proposed approach. In section 5, we present experimental results. The last section contains summary and conclusions.

2 About Concept Hierarchy and Diversity of Patterns

The notion of concept hierarchy plays the main role in assigning the rank to a pattern based on the diversity. In this section, we explain about concept hierarchy and the basic idea employed in the proposed approach to calculate the diversity.

2.1 Concept Hierarchy

A pattern contains data items. A concept hierarchy is a tree in which the data items are organized in an hierarchical manner. In this tree, all the leaf nodes represent the *items*, the internal nodes represent the *categories* and the top node represents the *root*. The *root* could be a virtual node.

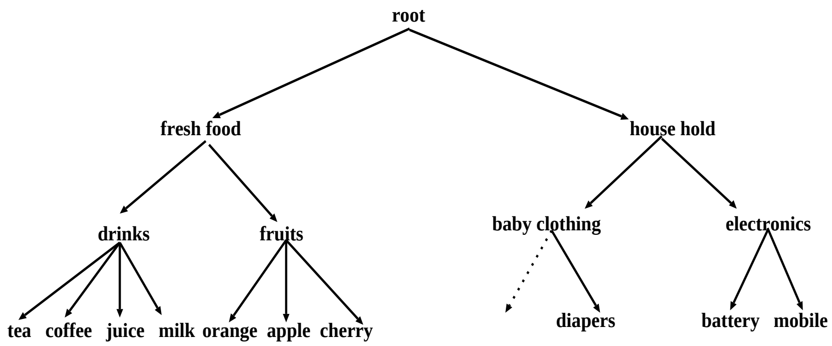

Fig. 1. An example of balanced concept hierarchy

Let C be a concept hierarchy. A node in C may be an item, category or root. The height of *roo*t node is 0. Let n be a node in C. The height of n, is denoted as $h(n)$, is equal to the number of edges on the path from *root* to n.

Figure 1 represents a concept hierarchy. In this, the items *orange*, *apple* and *cherry* are mapped to the category *fruits.* Similarly, the categories *drinks* and *fruits* are mapped to the category *fresh food.* Finally, the categories *fresh food* and *house hold* are mapped to *root*.

The concept hierarchy having height h has the same number of levels. The items at the given height are said to be at the same level. In C, all the lower-level nodes, except the *root*, are mapped to the immediate higher level nodes. In this paper, we consider the concept hierarchies in which a lower level node is mapped to only one higher level node.

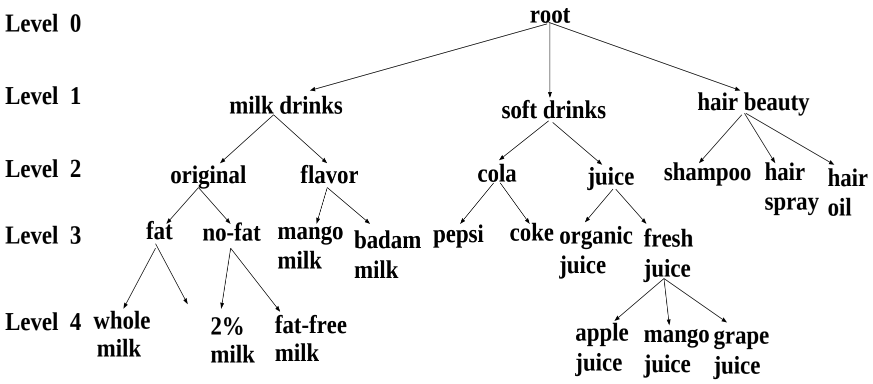

Fig. 2. An example of unbalanced concept hierarchy

The concept hierarchies can be balanced or unbalanced.

- **Balanced concept hierarchy**: In balanced concept hierarchy, the height of all leaf level nodes is the same. The height of balanced concept hierarchy is equal to the height of a leaf level item. Figure 1 is an example of balanced concept hierarchy.
- **Unbalanced concept hierarchy**: In an unbalanced concept hierarchy, the height of at least one of the leaf level node is different from the height of other leaf level nodes. The height of unbalanced concept hierarchy is equal to the height of the leaf level node having maximum height. Figure 2 is an example of the unbalanced concept hierarchy.

2.2 Diversity of Patterns

The diversity of a pattern is based on the category of the items within it. If the items of a pattern are mapped to the same/few categories in a concept hierarchy, we consider that the pattern has low diversity. Relatively, if the items are mapped to multiple categories, we consider that the pattern has more diversity. We have developed an approach to assign the diversity for a given pattern based on the merging behavior in the corresponding concept hierarchy. If the pattern merges into few higher level categories quickly, it has low diversity. Otherwise, if the pattern merges into one or a few high level categories slowly, it has relatively high diversity value.

As an example, consider the concept hierarchy in Figure 1. For the pattern {tea, juice}, the items *tea* and *juice* are mapped to the next level category *drinks*. In this case, the merging occurs quickly. For the pattern {coffee, orange}, the items *coffee* is mapped to category *drinks* and item *orange* maps to the category *fruits*. Further, both the categories *drinks* and *fruits* are mapped to the category *fresh food*, and the category *fresh food* in turn maps to *root*. We say that the pattern {coffee, orange} is more diverse than the pattern {tea, juice} as the merging is relatively slow in case of {coffee, orange} as compared to {tea, juice}. Consider the pattern {milk, mobile} which is relatively more diverse than the pattern {coffee, orange} as both items merge at the *root*. The merging of {milk, mobile} occurs slowly as compared to {coffee, orange}.

3 Computing Diverse Rank With Balanced Concept Hierarchy

In this section, we explain the process of calculating diverse rank of the pattern, called *drank*, proposed in [8]. We also introduce the concepts *balanced pattern* and *projection of a pattern* which are useful in presenting the proposed approach in the next section.

Definition 1. ***Balanced Pattern (BP)****: Consider a pattern $Y = \{i_1, i_2, \cdots, i_n\}$ with 'n' items and a concept hierarchy of height 'h'. The pattern Y is called balanced pattern, if the height of all the items in Y is equal to 'h'.*

Definition 2. ***Projection of Balanced Concept Hierarchy for*** Y $(P(Y/C))$*: Let* Y *be* BP *and* C *be balanced concept hierarchy. The* $P(Y/C)$ *is the projection of* C *for* Y *which contains the portion of* C*. All the nodes and edges exists in the paths of the items of* Y *to the root, along with the items and the root, are included in* $P(Y/C)$*. The projection P(Y/C) is a tree which represents a concept hierarchy concerning to the pattern Y.*

Given two patterns of the same length, different merging behavior can be realized, if we observe how the items in the pattern are mapped to higher level nodes. That is, one pattern may quickly merge to few higher level items within few levels and the other pattern may merge to few higher level items by crossing more number of levels. By capturing the process of merging, we define the notion of diverse rank (*drank*). So, *drank*(Y) is calculated by capturing how the items are merged from leaf-level to *root* in $P(Y/C)$. It can be observed that a given pattern maps from the leaf level to the *root* level through a merging process by crossing intermediate levels. At a given level, several lower level items/categories are merged into the corresponding higher level categories.

Two notions are employed to compute the diversity of a BP: *Merging Factor (MF)* and *Level Factor (LF).*

We explain about MF after presenting the notion of generalized pattern.

Definition 3. ***Generalized Pattern (GP(Y, l, P(Y/C))):*** *Let* Y *be a pattern, 'h' be the height of P(Y/C) and 'l' be an integer. The* $GP(Y, l, P(Y/C))$ *indicates the* GP *of* Y *at level 'l' in P(Y/C). Assume that the* $GP(Y, l{+}1, P(Y/C))$ *is given. The* $GP(Y, l, P(Y/C))$ *is calculated based on the* GP *of* Y *at level* $(l{+}1)$*. The* $GP(Y, l, P(Y/C))$ *is obtained by replacing every item at level* $(l + 1)$ *in* $GP(Y, l + 1, P(Y/C))$ *with its corresponding parent at the level 'l' with duplicates removed, if any.*

The notion of merging factor at level l is defined as follows.

Merging factor (MF(Y, l, P(Y/C))): Let Y be BP and l be the height. The merging factor indicates how the items of a pattern merge from the level $l + 1$ to the level l $(0 \leq l < h)$. If there is no change, the MF(Y,l) is 1. If all items merges to one node, the MF(Y,l) value equals to 0. So, the MF value at the level l is denoted by MF(Y,l, P(Y/C)) which is equal to the ratio of the number of nodes in (GP(Y, l, P(Y/C)-1) to the number of nodes in (GP(Y, l+1, P(Y/C)-1).

$$MF(Y, l, P(Y/C)) = \frac{|GP(Y,\ l,\ P(Y/C))| - 1}{|GP(Y,\ l+1,\ P(Y/C))| - 1} \tag{1}$$

We now define the notion of level factor to determine the contribution of nodes at the given level.

Level Factor (LF(l,P(Y/C)): For a given P(Y/C), h be the height of $P(Y/C)$ $\neq \{0, 1\}$. Let l be such that $1 \leq l \leq (h - 1)$. The LF value of P(Y/C) height l indicates the contribution of nodes at l to *drank*. We can assign equal, linear or exponential weights to each level. Here, we provide a formula which assigns the weight to the level such that the weight is in proportion the level number.

$$LF(l, P(Y/C)) = \frac{2 * (h - l)}{h * (h - 1)} \tag{2}$$

Diverse rank of a pattern Y: The *drank* of BF Y for a given C, is calculated by summing up the product of MF and LF from the leaf level to the *root* of $P(Y/C)$. The formula is as follows.

$$drank(Y, C) = \sum_{l=h-1}^{l=0} MF(Y, l, P(Y/C)) * LF(l, P(Y/C)) \tag{3}$$

where, Y is BP, h is height of P(Y/C).

4 Computing Diverse Rank With Unbalanced Concept Hierarchy

In this section, we explain the approach to assign the *drank* to unbalanced pattern. The term unbalanced pattern is defined as follows.

Definition 4. ***Unbalanced Pattern (UP)****: Consider a pattern Y and an unbalanced concept hierarchy U of height 'h'. A pattern is called unbalanced pattern, if the height of at least one of the item in Y is less than 'h'.*

The notion of unbalanced-ness depends on how the heights of the nodes in the concept hierarchy are distributed. It can be noted that we consider a pattern as unbalanced pattern, if the height of at least one item is less than the height of unbalanced concept hierarchy. Suppose, all the items of a pattern are at the height, say k. The pattern X is unbalanced, if k is less than the height of concept hierarchy.

The basic idea to compute *drank* of UP is as follows. We first convert the unbalanced concept hierarchy to balanced concept hierarchy called, "extended unbalanced concept hierarchy" by adding dummy nodes and edges. We calculate the *drank* of UP with Equation 3 by considering the "extended unbalanced concept hierarchy". Next, we reduce the *drank* in accordance with the number of dummy nodes and edges. So, the *drank* of UP is relative to the *drank* of the same pattern computed by considering all of its items are at the leaf level of the extended unbalanced concept hierarchy.

Given UP and the corresponding unbalanced concept hierarchy U, the following steps should be followed to calculate the *drank* of UP.

(i) Convert the U to the corresponding extended U.
(ii) Compute the effect of the dummy nodes and edges.
(iii) Compute the *drank*.

(i) Convert the Unbalanced Concept Hierarchy to Extended Unbalanced Concept Hierarchy

We define the extended unbalanced concept hierarchy as follows.

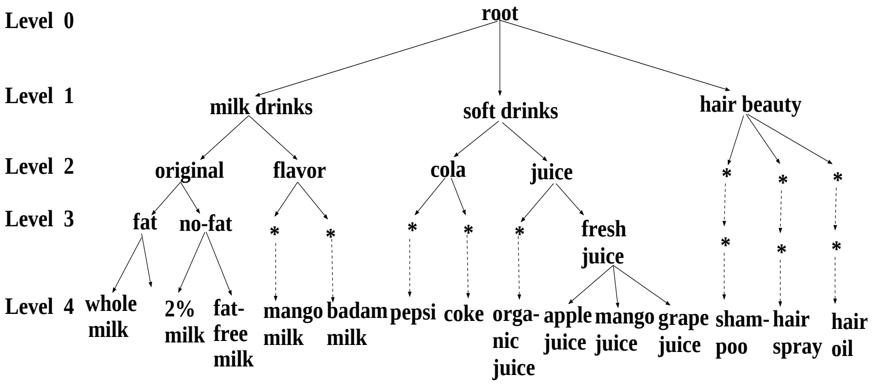

Fig. 3. Extended Unbalanced Concept Hierarchy for the Figure 2

Definition 5. ***Extended Unbalanced Concept Hierarchy (E)****: For a given unbalanced concept hierarchy U with height 'h', we convert U into extended U, say E, by adding dummy nodes and edges such that the height of each leaf level item is equal to 'h'.*

Figure 3 shows the extended unbalanced concept hierarchy of Figure 2. In Figure 3, '$*$' indicates the dummy node and dotted line indicates the dummy edge.

We define the projection of extended unbalanced concept hierarchy for Y as follows.

Definition 6. ***Projection of Extended Unbalanced Concept Hierarchy of*** Y ***(P(Y/E))****: Let Y be UP, U be unbalanced concept hierarchy, and E be the corresponding extended unbalanced concept hierarchy of U. The projection of E for the unbalanced pattern Y is $P(Y/E)$. The $P(Y/E)$ contains the portion of U which includes all the paths of the items of Y from the root.*

It can be noted that, in addition to real nodes/edges, $P(Y/E)$ may contain dummy nodes/edges.

As an example, consider the unbalanced concept hierarchy shown in Figure 4(i). In this figure, the items *a, b, c,* and *d* are located at different levels. We find the longest path $\langle root, l, k, a\rangle$ in the unbalanced concept hierarchy. The additional dummy nodes and edges are added such that all the items are at the height *h*. This extended unbalanced concept hierarchy is shown in Figure 4(ii). The projections of the patterns {a, b}, {b, c}, {b, d}, and {c, d} are shown in Figures 4(iii), 4(iv), 4(v), and 4(vi) respectively.

(ii) Compute the Effect of the Dummy Nodes and Edges

We define the notion of adjustment factor to compute the effect of the dummy nodes and edges.

Adjustment factor (AF): We define the *AF* at the given level. The *Adjustment Factor (AF)* at level l helps in reducing the *drank* by measuring the contribution of dummy edges/nodes relative to the original edges/nodes at the

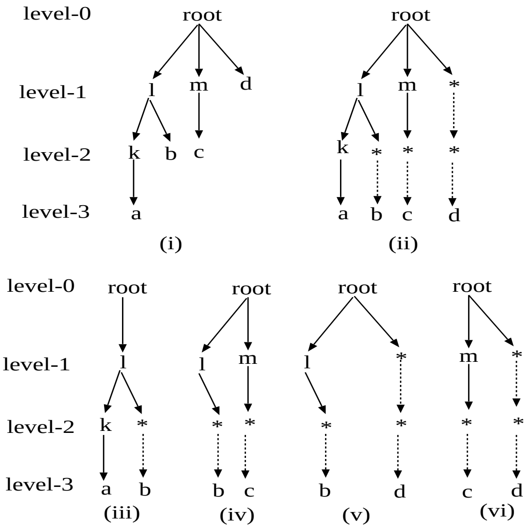

Fig. 4. (i) Unbalanced Concept Hierarchy, (ii) Extended Unbalanced Concept Hierarchy of (i). The projection of Extended Unbalanced Concept Hierarchies for the patterns {a, b}, {b, c}, {b, d}, and {c, d} are shown in (iii), (iv), (v), and (vi) respectively.

level l. The AF for a pattern Y at a level l should depend on the ratio of number of real edges formed with the children of the real nodes in $P(Y/E)$ versus total number of edges formed with the children of real and dummy nodes at l in $P(Y/E)$. The value of AF at a given height should lie between 0 and 1. If the number of real edges is equals to zero, AF is zero. If the pattern at the given level does not contain dummy nodes/edges, the value of AF becomes 1. Note that the AF value is not defined at the leaf level nodes as children do not exist. The AF for Y at height l is denoted as $AF(Y, l, P(Y/E))$ and is calculated by the following formula.

$$AF(Y, l, P(Y/U)) = \frac{\#\ of\ Real\ Edges\ of\ UP(Y, l, P(Y/E))}{\#\ of\ Total\ Edges\ of\ UP(Y, l, P(Y/E))} \tag{4}$$

where *numerator* is the number of edges formed with the children of the real nodes and *denominator* is the number of edges formed with the children of both real and dummy nodes at the level l in $P(Y/E)$.

Consider the frequent pattern $Y = \{whole\ milk,\ pepsi, coke, shampoo\}$ in Figure 3. The level of the item *whole milk* is 4. As l is between $(0, h)$, we calculate the number of edges at level 3. At the level 3, the number of real edges in $P(Y/E)$ is 1 and the total number edges including real and dummy edges in $P(Y/E)$ is 4, i.e., $AF(Y, 4, P(Y/U)) = \frac{1}{4} = 0.25$. Similarly, the AF value at level 2, level 1, level and 0.75, 1, and 1 respectively.

(iii) Computing the *drank* of UP

The *drank* of UP is a function of MF, AF and LF.

Definition 7. ***Diverse rank of a frequent pattern Y (drank(Y)):*** *Let Y be the pattern and U be the unbalanced concept hierarchy of height 'h'. The drank of Y, denoted by $drank(Y)$, is given by the following equation.*

$$drank(Y,U) = \sum_{l=h-1}^{l=0} [MF(Y,l,P(Y/E)) * AF(Y,l,P(Y/E))] * LF(l,P(Y/E)) \tag{5}$$

where, h is the height of the $P(P/E)$, E is the extended unbalanced concept hierarchy, $MF(Y,l,P(Y/E))$ is the MF of Y at level l, $LF(l,P(Y/E))$ is the LF at level l and $AF(Y,l,P(Y/E))$ is the AF of Y at level l.

It can be noted that Equation 5 can be used for computing *drank* of the patterns with both balanced and unbalanced concept hierarchy as the values of AF becomes 1 at all levels in case of balanced concept hierarchy.

5 Experiment Results

For conducting experiments, we have considered the groceries data set which contains 30 days of point-of-sale transaction data from a typical local grocery outlet. The data set contains 9,835 transactions and 168 items. The average transaction size in the data set is 4.4. The maximum and minimum transaction size is 32 and 1 respectively. To generate a concept hierarchy for the items, a web based Grocery API provided by Tesco [20] (a United Kingdom Grocery Chain Store) is used. Some of the items of the transactional data that were not listed in the concept hierarchy of Tesco are added manually by consulting the domain experts. There are 220 items (excluding internal nodes) available in the concept hierarchy of Tesco. We have removed the items which do not exist in the transactional data. The distribution of remaining 168 items at different levels in the concept hierarchy are shown in Table 1.

Table 1. Distribution of items in unbalanced concept hierarchy.

Level No.	No. of items
0	0
1	1
2	29
3	104
4	34

Table 2. Distribution of items in the simulated concept hierarchy.

Level No.	No. of items
4	5
5	14
6	12
7	36
8	17
9	25
10	12
11	40
12	7

Top Diverse Patterns: In Table 3, we present the list of the top 3-item patterns ordered by *drank*. In this table, the first column shows the pattern, the second column shows the *support* of the pattern, the third column shows the *drank* of UP, the fourth column shows the *drank* of UP with extended concept hierarchy

(E), and the final column shows the difference from *drank* of UP and the *drank* of UP with E.

Top Frequent Patterns: Table 4 contains the list of top 3-item patterns ordered by *support* value.

From these two tables, we can observe that there are no common patterns between them. The results show that the pattern having the highest *drank* value may not be the patterns with the highest *support*. Similarly, the patterns with the highest *support* may not have the highest value of *drank*. So, the patterns ordered by *drank* indicates a different kind of knowledge which could not be extracted by *support*.

Table 3. Patterns ordered by *drank*

Top 10 3-item diverse patterns	**Support** (%)	**drank of UP**	**drank of UP with E**	**Diff.**
{rolls-buns, soda, sausages}	1.0	1.00	1.0	0.00
{soda, rolls-buns, other vegetables}	1.0	1.00	1.0	0.00
{rolls-buns, soda, shopping bags}	0.6	1.00	1.0	0.00
{soda, whole milk, shopping bags}	0.7	0.89	1.0	0.11
{rolls-buns, whole milk, newspapers}	0.8	0.89	1.0	0.11
{rolls-buns, bottled water, other vegetables}	0.7	0.89	1.0	0.11
{rolls-buns, bottled water, yogurt}	0.7	0.89	1.0	0.11
{rolls-buns, soda, whole milk}	0.6	0.89	1.0	0.11
{rolls-buns, soda, yogurt}	0.9	0.89	1.0	0.11
{rolls-buns, bottled water, whole milk}	0.9	0.89	1.0	0.11

Table 4. Patterns ordered by *support*

Top 10 frequent patterns	**Support** (%)	**drank of UP**	**drank of UP with E**	**Diff.**
{whole milk, other vegetables, root vegetables}	2.3	0.29	0.33	0.04
{yogurt, whole milk, other vegetables}	2.2	0.31	0.33	0.02
{rolls-buns, whole milk, other vegetables}	1.8	0.67	0.75	0.08
{whole milk, tropical fruit, other vegetables}	1.7	0.44	0.50	0.06
{rolls-buns, yogurt, whole milk}	1.6	0.78	0.83	0.05
{yogurt, whole milk, root vegetables}	1.5	0.31	0.33	0.02
{yogurt, whole milk, tropical fruit}	1.5	0.31	0.33	0.02
{whipped sour cream, whole milk, other vegetables}	1.5	0.31	0.33	0.02
{whole milk, pip fruit, other vegetables}	1.4	0.44	0.50	0.06
{soda, whole milk, other vegetables}	1.4	0.67	0.75	0.08

The *drank* value of UP is obtained after reducing the effect of dummy nodes/edges from E. It can be noted that the *drank* of UP with E indicates the value of diversity without considering AF. So, the value in the final column of Table 3 and Table 4 indicates the degree of unbalanced-ness. If the value is low, the UP is less imbalanced and if the value is high, the UP is highly imbalanced. However, in Table 3 and Table 4, the values in the last column are very low. This is due to the fact that the concept hierarchy is relatively more balanced.

Influence of Adjustment Factor: In this experiment, we change the concept hierarchy such that it becomes very unbalanced. For this, we increase the level of some items. A random number from the list {1, 2, 4, 4, 6, 8, 8} is chosen to add the number of edges to increase the height of the items. The height of the simulated concept hierarchy is 12 and the distribution of items are shown in the Table 2.

Table 5 provides the details of *drank* of UP by considering the simulated concept hierarchy. In this table, the first column shows the pattern, the second column shows the *drank* of UP, the third column shows the *drank* of UP with E, and the last column shows difference between the *drank* of UP and the *drank* of UP with E. The values in the last column show that the notion of AF, along with MF and LF, helps in computing the *drank* for all kinds of patterns including less unbalanced patterns to high unbalanced patterns.

Table 5. Patterns by considering the simulated concept hierarchy

Patterns	**drank of UP**	**drank of UP with E**	**Diff.**
{whole milk, beef, root vegetables}	0.705	0.866	0.161
{whole milk, root vegetables, frozen vegetables}	0.750	0.933	0.183
{rolls-buns, whole milk, pork}	0.634	0.933	0.299
{pastry yogurt, other vegetables}	0.416	0.804	0.388
{yogurt, pip fruit, other vegetables}	0.454	0.866	0.412
{rolls-buns, whole milk, pip fruit}	0.434	0.933	0.499
{whole milk, yogurt, other vegetables}	0.036	0.805	0.769
{rolls-buns, whole milk, yogurt}	0.045	0.938	0.893
{rolls-buns, whole milk, other vegetables}	0.032	0.933	0.901

6 Summary and Conclusions

Finding interesting patterns is one of the issues in frequent pattern mining. Several interestingness measures have been proposed to extract the subset of frequent patterns according to the application's requirements. In this paper, we have proposed a new interestingness measure to rank the patterns based on diversity. We have proposed a general approach to assign the *drank* to the patterns by considering unbalanced concept hierarchy. For computing the *drank* of a pattern, the unbalanced concept hierarchy is being converted into balanced concept hierarchy by adding dummy nodes and edges. The notion of *adjustment factor* is proposed to remove the effect of the dummy nodes and edges. The *drank* is calculated using the notions of *merging factor*, *level factor* and *adjustment factor*. The experiments on the real world data set show that the patterns with high *drank* are different from the patterns with high *support*. Also, the proposed approach could assign the *drank* to patterns having different degrees of unbalanced-ness.

As a part of future work, we are planning to refine the approach by considering all types of unbalanced hierarchies. We are also planning to investigate how the notion of diversity influences the performance of frequent pattern based clustering, classification and recommendation algorithms.

References

1. J.Han, H.Cheng, D.Xin, and X. Yan.: Frequent pattern mining: current status and future directions. Data Min. Knowl. Discov. 15, 1, pp. 55–86, 2007.
2. R.Agrawal, and R.Srikant.: Fast algorithms for mining association rules. 20th Intl. Conf. on VLDB, pp. 487–499, 1994.
3. M.J.Zaki and C.-J. Hsiao.: Efficient algorithms for mining closed itemsets and their lattice structure. IEEE TKDE, 17 (4), pp. 462–478, 2005.
4. T.Hu, S.Y. Sung, H.Xiong, and Q.Fu.: Discovery of maximum length frequent itemsets. Information Sciences, 178 (1), pp. 69–87, 2008.
5. Quang Tran Minh, Shigeru Oyanagi, and K.Yamazaki.: Mining the k-most interesting frequent patterns sequentially. Intelligent Data Engineering and Automated Learning ? IDEAL 2006, LNCS, pp. 620–628, 2006.
6. J.Wang, J.Han, Y.Lu, and P.Tzvetkov.: TFP: an efficient algorithm for mining top-k frequent closed itemsets. IEEE TKDE, 17 (5), pp. 652–663, 2005.
7. J.Hu and A.Mojsilovic.: High-utility pattern mining: A method for discovery of high-utility item sets. Pattern Recogn, 40 (11) pp. 3317–3324, 2007.
8. S. Somya and R. Uday Kiran and P. Krishna Reddy.: Discovering Diverse-Frequent Patterns in Transactional Databases. International Conference on Management of Data (COMAD 2011), Bangalore, India, pp. 69-78, 2011.
9. R.Srikant and R.Agrawal.: Mining generalized association rules. VLDB, Zurich, Switzerland, pp. 407–419, 1995.
10. J.Han and Y.Fu.: Mining multiple-level association rules in large databases. IEEE TKDE, 11 (5), pp. 798–805 1999.
11. Y.Chen, G.-R. Xue, and Y.Yu.: Advertising keyword suggestion based on concept hierarchy. WSDM '08, USA, ACM, pp. 251–260, 2008.
12. L.Geng and H.J. Hamilton.: Interestingness measures for data mining: a survey. ACM Comput. Surv., 38 (3), pp.1–32, 2006.
13. R.J. Hilderman and H.J. Hamilton.: Knowledge Discovery and Measures of Interest. Kluwer Academic Publishers, Norwell, USA, 2001.
14. N.Zbidi, S.Faiz, and M.Limam.: On mining summaries by objective measures of interestingness. Machine Learning, 62, pp. 175–198, 2006.
15. R.A. Huebner.: Diversity-based interestingness measures for association rule mining. ASBBS-2009, Las Vegas, 2009.
16. S.Brin, R.Motwani, and C.Silverstein.: Beyond market baskets: Generalizing association rules to correlations. SIGMOD Rec. 26 (2), NY, USA, pp. 265–276, 1997.
17. B.Liu, W.Hsu, L.-F. Mun, and H.-Y. Lee.: Finding interesting patterns using user expectations. IEEE TKDE, 11 (6), pp. 817–832, 1999.
18. K.McGarry.: A survey of interestingness measures for knowledge discovery. Knowl. Eng. Rev., 20, pp. 39–61, 2005.
19. E.Omiecinski.: Alternative interest measures for mining associations in databases. IEEE TKDE, 15 (1), pp. 57– 69, 2003.
20. Tesco.: Grocery api. https://secure.techfortesco.com/tescoapiweb/, 2013.