A RP-growth method for discovering interesting Infrequent Patterns

Sneha Goud Gundu
Department of Computer Science, DE
Otto von Guericke Universität
Magdeburg, Germany
sneha.gundu@st.ovgu.de

Kushagra Kumar
Department of Computer Science, DKE
Otto von Guericke Universität
Magdeburg, Germany
kushagra.kumar@st.ovgu.de

Venkata Punnaiah Sastry Jammalamadaka
Department of Computer Science, DE
Otto von Guericke Universität
Magdeburg, Germany
Venkata.jammalamadaka@st.ovgu.de

Pranay Teja Arikatla
Department of Computer Science, DE
Otto von Guericke Universität
Magdeburg, Germany
pranay.arikatla@st.ovgu.de

Vinay Kumar Yadavelly
Department of Computer Science, DE
Otto von Guericke Universität
Magdeburg, Germany
vinaykumar.yadavelly@st.ovgu.de

Abhigna Domakonda
Department of Computer Science, DE
Otto von Guericke Universität
Magdeburg, Germany
abhigna.domakonda@st.ovgu.de

***Abstract*—The support and confidence framework does not guarantee that the association rules generated are interesting to the user. [1] In this paper, we are doing a statistical analysis of the association rules generated by the RP growth algorithm using statistical measures such as all_confidence, max_confidence, lift, cosine, coherence and Kulc. [1] RP growth algorithm generates infrequent itemsets hence these statistical measures are applied to the association rules for the infrequent patterns. We are trying to answer which statistical measure works better in the case of the sparse dataset.**

I. Introduction

Frequent item-set mining is a well studied and active area of research which represent main stream behaviours. Many efficient algorithms are proposed for mining the frequent item-sets [2]. However, infrequent item-sets are also interesting since they contain unknown knowledge and usually imply highly confident association rules which is useful to domain experts. For example in the area of medicine the expected responses are less interesting than the untypical responses which can be indicators of adverse reactions or rare diseases.These infrequent patterns also play essential role in other applications like recommendation systems and analysis of traffic accidents.

Existing frequent itemset mining algorithms can be used to find the infrequent patterns by giving the minimum support threshold as 1. But those algorithms need to access the frequent patterns which are time consuming [3] , [4] In this paper we focus on the infrequent item set mining problem and finding the association rules for the infrequent patterns and finding the interestingness measures for the different data sets. In order to avoid the accessing of the frequent patterns we used a top-down based approach which identifies the infrequent patterns first that means no time is wasted on mining the frequent patterns. Many infrequent pattern mining algorithms have been proposed in recent years which are adapted from either breadth-first or top-down approaches which suffered from time consuming and expensive candidate-generation step [5], [6]. Hence we have considered RP-tree which uses a tree structure to find the infrequent itemset we can say that RP tree can find the association rules more efficiently than the existing algorithms, and identifies 92-100% rare association rules that meet a confidence.

The structure of the paper is as follows, to begin with the II gives a brief introduction to the Negative item sets, statistical measures and some basic concepts. Section III gives the information on the related work. Eventually, we discussed about the Methods used for implementation in Section IV that helps to find the infrequent patterns and association rules, followed by in sectionV we discussed about the implementation of the statistical measures. Thereupon, we exhibited the results in Section VI in the form of images where our interesting measures are compared for the different datasets which are used for the evaluation and then concluded in Section VII.

II. Background

A. Negative itemset tree:

Let I=$\{i_1,i_2,i_3,...i_m\}$ be the set of all distinct items and the transaction database D = $\{t_1,t_2,...t_n\}$. Any non-empty subset X $\subseteq$ I is an itemset. Any itemset X with size $\mid X \mid$ = k is referred to as a k−itemset. A tuple T = (tid,X) is called a transaction , where tid is the transaction identifier. The support of an itemset X can be defined as the number of transactions T∈D where X: X.supp=$\mid \{T \in D \mid X \subseteq T\} \mid$. Based on the minimum support threshold the itemsets are categorized into three types:nonexistent, infrequent and frequent. An itemset X is frequent if and only if:(X.supp $\leq$ minSup). Otherwise, it is infrequent (0 $<$ X.supp $<$ minSup) or nonexistent (X.supp = 0).

In our early experiments we tried to extract all the infrequent itemsets by using a Negative itemset tree miner which uses the top down based approach.

1) Basic concepts:: Negative item: Given an itemset X = {a, b, c} it implies that all the three items belongs to X and all of them are positive items. Similarly the negaive items are defined as the items that does not exist in X. It is denoted as ¬ a.

Negative Itemset: Given an itemset I = $\{i_1,i_2,i_3,...i_m\}$ the negative representation of these itmset is the set of items that I does not have, which is denoted as ¬ I = $\{\neg\, i_1,\neg\, i_2,\neg\, i_3,...\neg\, i_m\}$.

Tid	Transactions
1	A B C
2	A B D
3	B C
4	A B
5	A B E
6	D E

(a)

Tid	Transactions
1	¬D ¬E
2	¬C ¬E
3	¬A ¬D ¬E
4	¬C ¬D ¬E
5	¬C ¬D
6	¬A ¬B ¬C

(b)

Fig. 1: Example transaction database (a) and its corresponding Negative-Rep database (b)

[7]

B. Statistical Measures

A support-confidence framework is used in most association rule mining algorithms.Although minimum support and confidence thresholds aid in the exclusion of a large number of uninterestingness rules, many of the rules generated are still uninteresting to the users. Unfortunately, this is especially true when mining for longer patterns or at low support thresholds. This has been a significant hurdle in the successful implementation of association rule mining.

A correlation measure can be used to supplement the support-confidence framework for association rules to solve this issue. In this paper we examined six different correlation measures here to see which ones are best for mining sparse datasets. In which All Confidence, Coherence, Cosine, Kulczynski, and Max Confidence are the five known null-invariant correlation measurements.

1) Lift: The ratio of these variables is known as lift: target response divided by average response. When compared to a random choice targeting model, lift is a measure of how well a targeting model predicts cases with an elevated response.

$$lift(A, B) = \frac{P(A \cup B)}{P(A)P(B)}.$$

The correlation between the itemsets A and B using lift measure can be expressed as follows, if the value of the measure is 1 then A and B are independent, if the value is greater than 1 then the itemsets are positively correlated, if the value is less than 1 then the itemsets are negatively correlated.

2) All conf: All-confidence has a downward-closed closure property so, it can be used effectively in mining algorithms [8]

Consider A and B has two itemsets, the all_confidence measure of A and B is calculated as follows

$$all\_conf(A, B) = \frac{sup(A \cup B)}{max\{sup(A), sup(B)\}} = min\{P(A|B), P(B|A)\},$$

Where maxsup(A), sup(B) is the maximum support of the itemsets A and B. Likewise, the all_conf(A,B) is the minimum confidence of the two association rules of A and B i.e., "A->B" and "B->A".

3) Maxconf: Consider A and B has two itemsets, the max_confidence measure of A and B is calculated as follows

$$max\_conf(A, B) = max\{P(A \mid B), P(B \mid A)\}.$$

The max_conf(A,B) is the maximum confidence of the two association rules of A and B i.e., "A->B" and "B->A".

4) Kulczynski: It was proposed in 1927 by Polish mathematician S. Kulczynski. Consider A and B has two itemsets, the Kulczynski measure of A and B(abbreviated as Kulc) is calculated as follows

$$Kulc(A, B) = \frac{1}{2}(P(A|B) + P(B|A)).$$

It can be viewed as an average of two confidence measures. confidence probabilities: The probability of itemset B given itemset A, and the probability of itemset A given itemset B.

5) Cosine: Consider A and B has two itemsets, the cosine measure of A and B is calculated as follows

$$cosine(A, B) = \frac{P(A \cup B)}{\sqrt{P(A) \times P(B)}} = \frac{sup(A \cup B)}{\sqrt{sup(A) \times sup(B)}}$$
$$= \sqrt{P(A|B) \times P(B|A)}.$$

The cosine measure can be considered as a harmonized lift measure: However in cosine, because by taking the square root on the product of the probabilities of A and B, the cosine value is only influenced by the supports of A,B and A U B, and not by the total number of transactions.

6) Coherence: This statistical measure can explain the inherent relationships of the items as follows

$$Coherence(a,b) = (P(a|b)^{-1} + P(b|a)^{-1} - 1)^{-1}$$

To reflect the relationship between the frequent items, coherence is the strong measure but it is not suitable for the infrequent items.

The correlation degree with these null invariant measures is expressed as a real number between 0 and 1. The strongest correlations may have different values for different datasets.

Measure	Definition	Range	Null-invariant
Lift(a, b)	$\frac{P(ab)}{P(a)P(b)}$	[0, ∞]	No
AllConf(a, b)	$\frac{sup(ab)}{max\{sup(a),sup(b)\}}$	[0, 1]	Yes
Coherence(a, b)	$\frac{sup(ab)}{sup(a)+sup(b)-sup(ab)}$	[0, 1]	Yes
Cosine(a, b)	$\frac{sup(ab)}{\sqrt{sup(a)sup(b)}}$	[0, 1]	Yes
Kulc(a, b)	$\frac{sup(ab)}{2}\left(\frac{1}{sup(a)} + \frac{1}{sup(b)}\right)$	[0, 1]	Yes
MaxConf(a,b)	$max\left\{\frac{sup(ab)}{sup(a)}, \frac{sup(ab)}{sup(b)}\right\}$	[0, 1]	Yes

Fig. 2: Interesting Measures Definition

C. Datasets Description

Datasets list				
Dataset Name	DataSet type	Number of Instances	Item Ac-count	Average item count per transac-tion
Online Retail	Sparse	541909	2603	4.37
Teaching	Very Sparse	151	26	6.0
Kosarak	Sparse	990,002	41,270	8.1
BMS web-view1	Sparse	59,602	497	2.51
Skin Seg-menta-tion	Sparse	245,057	11	4.0

III. Related Work

Existing rare pattern mining approaches are based on the level wise exploration of the search space such as Breadth-first based approaches, which is replica of the Apriori [9] algorithm. In Apriori algorithm, the candidate itemsets size have been increased gradually, i.e., all itemsets with size 'k' have to be generated and sorted before setting down itemsets with size 'k + 1'. Which are then pruned using the downward closure property. By the end of pruning, when there are no new k+1 itemsets then Apriori aborts. we have some predefined algorithms, which were very effective to disclose rare itemsets and practice level-wise exploration related to Apriori are Rarity, AfRIM, ARIMA [6] which uses a pruned itemsets of apriori in a first mining step to generate rare itemset candidates bottom-up in a second step and Apriori-Inverse.

Troiano et al [10]. proposed the Rarity algorithm that follow a sequential manner by spotting the longest transaction within the database and utilize them to perform a top-down search for rare itemsets, thereby eliminating the frequent itemsets presented at the bottom layers. In order to implement the above method, Troiano et al. finds out that rare itemsets are at the top of the search space, so that bottom-up algorithms must first search through many layers of frequent itemsets. In Rarity, possibly there are two different ways to prune rare itemsets (candidates).

1) All k-itemset candidates that are the subset of any of the frequent k+1 itemsets are removed as a candidate. why because, according to the downward closure property they must be frequent .
2) The supports for remaining candidates have been calculated, and only those that have a support below the threshold are used to generate the k−1 candidates. The candidates with supports above the threshold are used to prune k−1 candidates in the next iteration.

Adda et al. [5] recommended an approach similar to Rarity, which uses a top-down approach called "AfRIM". In AfRIM, rare itemset search begins with the itemset that contains all items found in the database. Candidate generation occurs by finding common k-itemset subsets between all combinations of rare k+1-itemset pairs in the previous level. Just like the Rarity algorithm, Candidates are pruned in the same method. Note that AfRIM examines itemsets that have zero support, which may be inefficient.

Koh et al [11] proposed Apriori-Inverse, which are used to mine perfectly rare itemsets,means that only consist of items below a maximum support threshold (maxSup). Except the initialisation, Apriori-Inverse is almost same as Apriori.Only 1-itemsets that fall below maxSup are used for generating 2-itemsets. All rare itemsets generated must have a support below maxSup, as Apriori-Inverse inverts the downward-closure property of Apriori. Further, itemsets must also meet an absolute minimum support. Since the set of perfectly rare-rules may only be a small subset of rare itemsets, Koh et al. also proposed several modifications that allow Apriori-Inverse to find near-perfect rare itemsets. The methods are based on

increasing maxSup during itemset generation, but using the original maxSup during rule generation.

IV. Methods for Implementation

A. Negative Itemset Tree Miner:

To build a negative itemset tree we consider a transaction databse D which is shown in fig1(a) is coverted into the negative-rep transaction database $\bar{D}$ as shown in 1(b) and the negative items in each transaction are rearranged in the descending order based on their frequency in $\bar{D}$. Each transaction is inserted into the NI-tree one by one based on the length of the each transaction.

Each node has a negative item ¬i , a count value 'C' and a list of child nodes 'l'. The root node consists of an itemset {is} which is considered as the list of items I, Count value C and list of child nodes which are the immediate successors of the root node are called the first layer nodes. The count value of the root node is initialized as zero and the count value of the child nodes is defined as the number of neg-rep transactions that end at each transaction. If the count of all the nodes in the path are zero, then the last node will be marked as termination node.

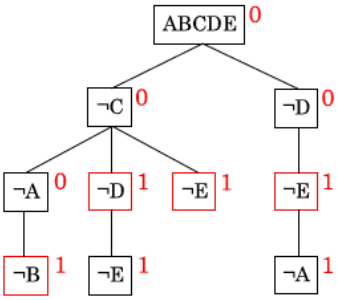

Fig. 3: Negative Itemset tree
[7]

Fig 3 is built based on the database shown in fig 1 After building a NI-tree we remove items from the root node as well as the child nodes which leads to a new NI-tree called the deducted tree. Initially we check all the first layer nodes one by one in the NI-tree that if a node is marked with an item in the itemset, then the node is attached to the new root node or we will check its child nodes recursively. The count value of the deleted nodes is added to the new root node. The process is terminated when all the first layer nodes are checked. The count value of the new root node is considered as the support of the new pattern which is generated from the subtraction of the NI-tree.

The example de-trees which are shown in Fig4 are built by excluding ¬C,¬C¬D and ¬C¬D¬B from the initial NI-tree in Fig3. In Fig4a ¬C is removed and its child nodes are attached to the new root node and the support of the new pattern generated is 0 since the count of the deleted node is 0. Similarly all the first layer nodes are checked in Fig4b,Fig4c.

First Layer Nodes Pruning: We used First layer pruning technique to get all the infrequent patterns without generating

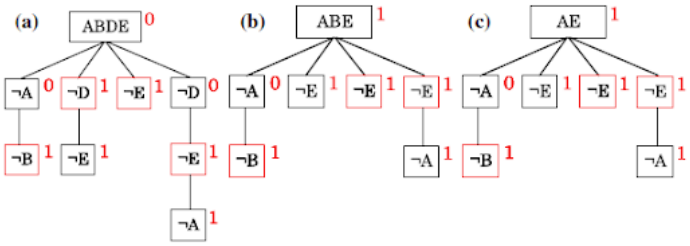

Fig. 4: Example de-trees
[7]

the non existing patterns, frequent patterns and to avoid duplicates. We followed the approach from [7] and developed the algorithm but we faced some challenges in the implementation as we are unclear about few steps in the algorithm and stuck at that point. Hence we started working on the RP tree.

B. RP-Tree Algorithm:

RP-Tree algorithm is an improvement over existing algorithms like ARIMA, AfRIM and Apriori-Inverse. To begin with, it avoids expensive itemset generation and pruning by using a tree data structure to find rare patterns. Additionally, it focuses on finding rare-item itemsets that generate interesting rules and does not look for uninteresting non rare-item itemsets. Also since the task is divided into a series of searches for short patterns, RP Tree is based on FP Growth which is efficient at finding long patterns.

RP-Tree algorithm performs one database scan to count item support, similar to FP Growth. During the second scan, since transactions that only have non-rare items cannot contribute to the support of any rare-item itemset, RP Tree uses only the transactions which include atleast one rare item to build the initial tree and prunes the others. Using this initial tree, RP Tree constructs conditional pattern bases and trees for each rare item only. Each conditional tree and its corresponding rare items are then used as arguments for FP Growth. minRareSup is the threshold which is then used to prune items from the conditional trees. The result is a union of all the results obtained from each of these calls to FP Growth which contain a rare-item or rare-item itemset. This results in the RP-Tree with a complete set of rare-item itemsets.

In majority of cases, RP Tree achieves far fewer itemsets for some datasets compared to FP growth, which in turn means that the time taken for rule generation is much less in comparison. RP tree also does not reduce the rule quality, but improves the overall rule quality in the set. However, the effect of minRareSup on quality of rules generated by RP tree need to be investigated. Also, there is a need to investigate ways to deal with noise and removing coincidental non-rare-item itemsets.

V. Implementation

To generate the infrequent items and their corresponding association rule from the dataset we are using spmf, an open-source software implemented in java. It is distributed under the

GPL v3 license. It has no dependency on other libraries and is fast and lightweight. We are using RP growth Algorithm [12] implementation for infrequent items generation and Agrawal and srikant [9] algorithm for association rule mining. The hyper parameters for generating the association rule generation are minimum support, minimum rare support, and confidence. we experimented with different configurations of minimum support, minimum rare support keeping confidence constant. The generated association rules are then passed to the python program which pre processes the output and stores it in a CSV file. The CSV file is then passed to another python program that computes the support values for antecedent and consequent of each association rule. It also computes all the statistical measures for each association rule in the CSV file and averages it out. The python programs are implemented by us.

VI. Evaluation

A. Performance measures of itemsets

In this section, We measure the statistical measures of the itemset generation for RP-Tree. We use different minsup and minraresup threshold across all experiments.

Support and confidence are two most highly used rule interestingness measures, where they deduce the usefulness and certainty of generated association rules. However, they have their own limitations which are discussed below. Support has its own drawback when it comes to rare items. Rarely occurring items in the dataset are dropped even though it could generate interesting and useful rules. In case of rare items, the distribution of support for the individual items are uneven due to the given fact of the power law distribution where most item are rarely used and few items are used most of the time, hence the rare item problem is essential for transaction data. [13]

The disadvantage of confidence is that it is sensitive to the frequency of the consequent in the transaction dataset .Due to the way confidence is calculated, even though there exists no correlation between items , consequents having higher support will inevitably produce higher confidence. [13]

Apart from support and confidence, there are other statistical measures such as lift and coherence which are not suitable for our study of interestingness on sparse datasets that are highlighted below. Although lift is not downward closed and immune to rare item problems, it is highly sensitive to noise in sparse datasets. Rare itemsets with low supports, which seldom occurs together, can generate huge lift values. Also lift is susceptible to null transactions i.e lift is null-invariant. Due to this reason, we are not considering lift as a statistical measure in our study. [14]

According to the reference [15] , although Coherence accurately reflects the relationship between the frequent items, it is not suitable to find the correlation between the itemsets in the infrequent/rare itemsets. In this Section, our discussion will be limited to null-invariant interestingness measures, such as All Confidence, Cosine, Kulc, and Max Confidence are based on the null-invariance property for rare itemsets. These interestingness measures ranges between 0 to 1, where 0 implies no co-occurrence between antecedents and consequents and 1 implies that they always appear together.

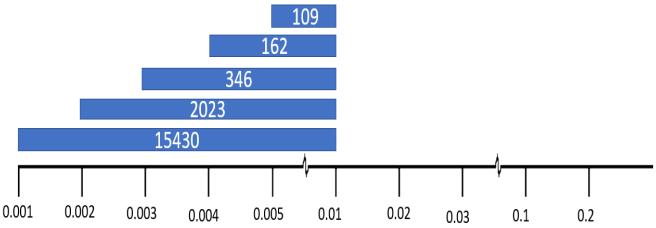

Fig. 5: Number of association rules to the respective ranges for Online Retail dataset

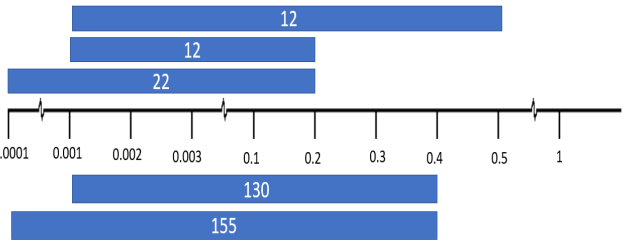

Fig. 6: Number of association rules to the respective ranges for Skin dataset

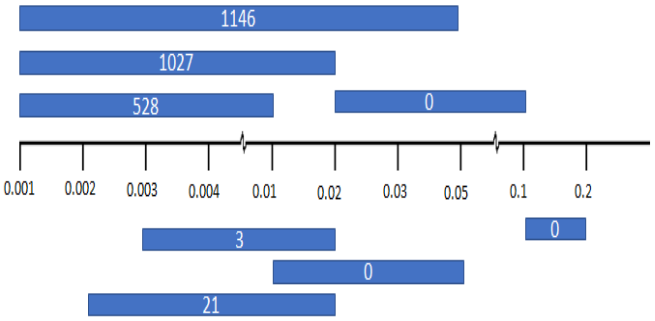

Fig. 7: Table: Number of association rules to the respective ranges for BMSWebView1 dataset

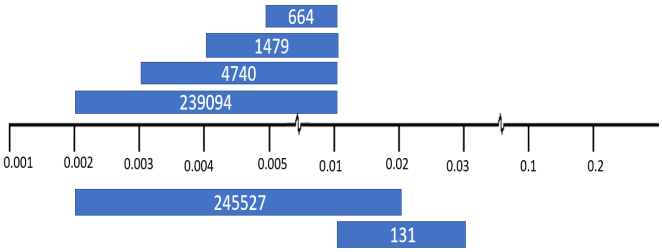

Fig. 8: Number of association rules to the respective ranges for Kosarak dataset

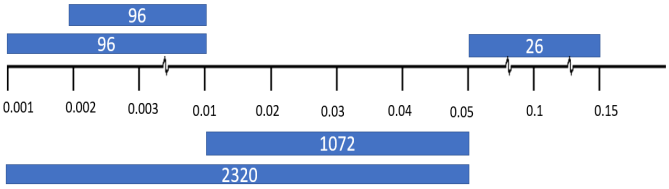

Fig. 9: Number of association rules to the respective ranges for Teaching dataset

Antecedent	Consequent	Supp	Conf	All conf	Max conf	Coherence	Cosine	Kulc	Lift
{12387}	{12339}	0.0022	0.8874	0.1473	0.8874	0.1446	0.3615	0.5173	58.1229
{12827, 10307, 10295}	{12895}	0.002	0.8176	0.0334	0.8176	0.0332	0.1652	0.4255	13.4498
{12815, 10295}	{12895}	0.0023	0.8782	0.0378	0.8782	0.0376	0.1822	0.458	14.4474

Fig. 10: **BMSWebview1** dataset minsup = **0.01** and min-raresup=**0.001**

Antecedent	Consequent	Supp	Conf	All conf	Max conf	Coherence	Cosine	Kulc	Lift
{33457, 33465, 33437}	{33433, 33441}	0.001	0.9231	0.6818	0.9231	0.6452	0.7933	0.8024	625.1958
{33441, 33465, 33429}	{33433}	0.001	0.9531	0.0685	0.9531	0.0682	0.2554	0.5108	63.7578
{12815, 12679}	{12895}	0.0014	0.9032	0.0232	0.9032	0.0231	0.1447	0.4632	14.859

Fig. 11: **BMSWebview1** dataset minsup = **0.05** and min-raresup=**0.001**

Antecedent	Consequent	Supp	Conf	All conf	Max conf	Coherence	Cosine	Kulc	Lift
{32205, 12703}	{32213}	0.002	0.8188	0.0755	0.8188	0.0743	0.2486	0.4471	30.199
{12387}	{12339}	0.0022	0.8874	0.1473	0.8874	0.1446	0.3615	0.5173	58.1229
{12827, 10307, 10295}	{12895}	0.002	0.8176	0.0334	0.8176	0.0332	0.1652	0.4255	13.4498

Fig. 12: **BMSWebview1** dataset minsup = **0.02** and min-raresup=**0.001**

MinSup	Minraresup	supp	Conf.	allconf	maxconf	coherence	Cosine	Kulc	Lift
0.01	0.001	0.0011	0.8805	0.1987	0.8807	0.1882	0.378	0.5397	165.1795
0.05	0.001	0.0012	0.8719	0.1234	0.872	0.1177	0.2852	0.4977	98.6445
0.02	0.001	0.0012	0.8736	0.1318	0.8737	0.1256	0.2958	0.5028	106.2304

Fig. 13: Mean Values of **BMSWebview1** dataset

Antecedent	Consequent	Supp	Conf	All conf	Max conf	Coherence	Cosine	Kulc	Lift
{473}	{11, 6}	0.0057	0.8507	0.0174	0.8507	0.0173	0.1217	0.434	2.618
{265, 7}	{6}	0.0057	0.9398	0.0093	0.9398	0.0093	0.0937	0.4746	1.5487
{1956, 7}	{205}	0.0041	0.9624	0.186	0.9624	0.1847	0.4231	0.5742	44.0785

Fig. 14: **Kosarak** dataset minsup = **0.01** and minraresup=**0.003**

Antecedent	Consequent	Supp	Conf	All conf	Max conf	Coherence	Cosine	Kulc	Lift
{379,148, 6}	{218, 11}	0.004	0.9161	0.065	0.9161	0.0646	0.2439	0.4905	14.7991
{85, 27}	{86}	0.0047	0.9262	0.5069	0.9262	0.4872	0.6852	0.7166	100.1903
{448}	{11, 6}	0.0044	0.8516	0.0136	0.8516	0.0135	0.1074	0.4326	2.621

Fig. 15: **Kosarak** dataset minsup = **0.01** and minraresup=**0.004**

Antecedent	Consequent	Supp	Conf	All conf	Max conf	Coherence	Cosine	Kulc	Lift
{255}	{148, 6}	0.0078	0.9191	0.1198	0.9191	0.1185	0.3318	0.5194	14.1383
{537}	{11}	0.0054	0.8812	0.0148	0.8812	0.0148	0.1144	0.448	2.41
{255}	{218,11, 148, 6}	0.0072	0.8448	0.1424	0.8448	0.1387	0.3468	0.4936	16.8043

Fig. 16: **Kosarak** dataset minsup = **0.01** and minraresup=**0.005**

Antecedent	Consequent	Supp	Conf	All conf	Max conf	Coherence	Cosine	Kulc	Lift
{87}	{6}	0.017	0.9137	0.028	0.9137	0.028	0.1601	0.4709	1.5057
{27, 205, 6}	{11}	0.0117	0.8247	0.032	0.8247	0.0318	0.1624	0.4284	2.2555
{87}	{6, 7}	0.0155	0.8324	0.2097	0.8324	0.2012	0.4178	0.5211	11.2605

Fig. 17: **Kosarak** dataset minsup = **0.3** and minraresup=**0.01**

MinSup	Minraresup	supp	Conf.	allconf	maxconf	coherence	Cosine	Kulc	Lift
0.01	0.004	0.0052	0.9227	0.1647	0.9262	0.158	0.2832	0.5454	30.4189
0.01	0.003	0.0039	0.931	0.2113	0.9344	0.2034	0.3272	0.5729	57.2042
0.01	0.005	0.0063	0.9185	0.1486	0.9197	0.1417	0.2764	0.5341	21.9697

Fig. 18: Mean Values of **Kosarak** dataset

Antecedent	Consequent	Supp	Conf	All conf	Max conf	Coherence	Cosine	Kulc	Lift
{1394, 1658}	{1712}	0.0016	1.0	0.8373	1.0	0.8373	0.9151	0.9187	518.5732
{331, 892}	{2398, 2375}	0.0013	1.0	0.2805	1.0	0.2805	0.5297	0.6403	217.8091
{1008,1989}	{1394}	0.0015	1.0	0.0396	1.0	0.0396	0.199	0.5198	26.0008

Fig. 19: **Online Retail** dataset minsup = **0.01** and minraresup=**0.001**

Antecedent	Consequent	Supp	Conf	All conf	Max conf	Coherence	Cosine	Kulc	Lift
{1534, 1239}	{2232,1241,991943}	0.0023	1.0	1.0	1.0	1.0	1.0	1.0	435.9686
{1394, 759}	{324, 1989}	0.0031	1.0	0.5875	1.0	0.5875	0.7665	0.7938	190.0768
{188, 2398}	{2375}	0.0021	1.0	0.0539	1.0	0.0539	0.2322	0.527	26.037

Fig. 20: **Online Retail** dataset minsup = **0.01** and minraresup=**0.002**

Antecedent	Consequent	Supp	Conf	All conf	Max conf	Coherence	Cosine	Kulc	Lift
{984, 1240}	{1586}	0.0032	1.0	0.8081	1.0	0.8081	0.8989	0.904	254.896
{1703}	{1165}	0.0069	1.0	0.3328	1.0	0.3328	0.5769	0.6664	48.3071
{1534, 231}	{1943, 479}	0.0033	1.0	0.5189	1.0	0.5189	0.7203	0.7594	157.5317

Fig. 21: **Online Retail** dataset minsup = **0.01** and minraresup=**0.003**

Antecedent	Consequent	Supp	Conf	All conf	Max conf	Coherence	Cosine	Kulc	Lift
{2204, 1909}	{1825}	0.0056	1.0	0.3346	1.0	0.3346	0.5785	0.6673	59.4198
{1909}	{1825, 2204, 181}	0.0056	0.9006	0.9006	1.0	0.9006	0.949	0.9503	159.9023
{1989, 951}	{1394, 324}	0.0053	1.0	1.0	1.0	1.0	1.0	1.0	190.0768

Fig. 22: **Online Retail** dataset minsup = **0.01** and minraresup=**0.004**

Antecedent	Consequent	Supp	Conf	All conf	Max conf	Coherence	Cosine	Kulc	Lift
{1156,1317}	{1324}	0.0056	1.0	0.382	1.0	0.382	0.6181	0.691	67.7386
{2236}	{1534}	0.007	1.0	0.0713	1.0	0.0713	0.2671	0.5357	10.1574
{1839}	{1821}	0.0057	0.9997	0.9994	0.9997	0.999	0.9995	0.9995	174.2472

Fig. 23: **Online Retail** dataset minsup = **0.01** and minraresup =**0.005**

MinSup	Minraresup	supp	Conf.	allconf	maxconf	coherence	Cosine	Kulc	Lift
0.01	0.001	0.0015	0.9934	0.5108	0.9953	0.5102	0.6334	0.7531	383.5561
0.01	0.002	0.0026	0.9859	0.4457	0.9898	0.4439	0.5888	0.7178	185.3643
0.01	0.003	0.0044	0.9703	0.4401	0.9788	0.4354	0.5967	0.7094	102.467

Fig. 24: Mean Values of **Online Retail** dataset

Antecedent	Consequent	Supp	Conf	All conf	Max conf	Coherence	Cosine	Kulc	Lift
(0, 6)	(11)	0.0061	1.0	0.0077	1.0	0.0077	0.0879	0.5039	1.2619
(0, 8)	(11)	0.002	1.0	0.0025	1.0	0.0025	0.05	0.5012	1.2619
(0, 7)	(11)	0.0022	1.0	0.0028	1.0	0.0028	0.0529	0.5014	1.2619

Fig. 25: **Skin** dataset minsup = **0.2** and minraresup =**0.001**

Antecedent	Consequent	Supp	Conf	All conf	Max conf	Coherence	Cosine	Kulc	Lift
{0, 5, 7}	{11}	0.0001	1.0	0.0002	1.0	0.0002	0.0124	0.5001	1.2619
{0, 8, 5}	{11}	0.0008	1.0	0.001	1.0	0.001	0.0318	0.5005	1.2619
{0, 4}	{11}	0.0039	1.0	0.005	1.0	0.005	0.0705	0.5025	1.2619

Fig. 26: **Skin** dataset minsup = **0.2** and minraresup=**0.0001**

Antecedent	Consequent	Supp	Conf	All conf	Max conf	Coherence	Cosine	Kulc	Lift
{2, 4, 7}	{11}	0.0072	1.0	0.0091	1.0	0.0091	0.0955	0.5046	1.2619
{0, 9, 3}	{11}	0.0019	1.0	0.0024	1.0	0.0024	0.0495	0.5012	1.2619
{0, 9, 6}	{11}	0.0013	1.0	0.0016	1.0	0.0016	0.04	0.5008	1.2619

Fig. 27: **Skin** dataset minsup = **0.4** and minraresup=**0.001**

Antecedent	Consequent	Supp	Conf	All conf	Max conf	Coherence	Cosine	Kulc	Lift
{8, 1, 5}	{10}	0.0325	0.8925	0.1565	0.8925	0.1536	0.3738	0.5245	4.3003
{8, 2, 11}	{5}	0.1716	0.9561	0.4166	0.9561	0.4088	0.6311	0.6863	2.3218
{4}	{1, 11}	0.207	0.8819	0.8819	0.8944	0.7988	0.8881	0.8882	3.8103

Fig. 28: **Skin** dataset minsup = **0.4** and minraresup=**0.0001**

Antecedent	Consequent	Supp	Conf	All conf	Max conf	Coherence	Cosine	Kulc	Lift
{0, 4}	{11}	0.0039	1.0	0.005	1.0	0.005	0.0705	0.5025	1.2619
{0, 9, 3}	{11}	0.0019	1.0	0.0024	1.0	0.0024	0.0495	0.5012	1.2619
{0, 7}	{11}	0.0022	1.0	0.0028	1.0	0.0028	0.0529	0.5014	1.2619

Fig. 29: **Skin** dataset minsup = **0.5** and minraresup=**0.001**

MinSup	Minraresup	supp	Conf.	allconf	maxconf	coherence	Cosine	Kulc	Lift
0.2	0.001	0.0037	1.0	0.0046	1.0	0.0046	0.0634	0.5023	1.2619
0.2	0.0001	0.0023	1.0	0.0029	1.0	0.0029	0.0472	0.5015	1.2619
0.4	0.001	0.096	0.9417	0.2522	0.9421	0.239	0.3925	0.5971	2.4447

Fig. 30: Mean Values of **Skin** dataset

Antecedent	Consequent	Supp	Conf	All conf	Max conf	Coherence	Cosine	Kulc	Lift
{3, 54}	{2, 13}	0.0066	1.0	0.0625	1.0	0.0625	0.25	0.5312	9.4375
{44}	{1, 17}	0.0066	1.0	0.125	1.0	0.125	0.3536	0.5625	18.875
{32}	{14, 15}	0.0066	1.0	0.1667	1.0	0.1667	0.4082	0.5833	25.1667

Fig. 31: **Teaching** dataset Min sup = **0.01** and minraresup=**0.001**

Antecedent	Consequent	Supp	Conf	All conf	Max conf	Coherence	Cosine	Kulc	Lift
{54}	{3, 13}	0.0066	1.0	0.1	1.0	0.1	0.3162	0.55	15.1
{3, 54}	{2, 13}	0.0066	1.0	0.0625	1.0	0.0625	0.25	0.5312	9.4375
{3, 13, 54}	{2}	0.0066	1.0	0.007	1.0	0.007	0.0836	0.5035	1.0559

Fig. 32: **Teaching** dataset minsup = **0.01** and minraresup=**0.002**

Antecedent	Consequent	Supp	Conf	All conf	Max conf	Coherence	Cosine	Kulc	Lift
{19}	{1, 3}	0.0728	0.8462	0.22	0.8462	0.2115	0.4315	0.5331	2.5554
{19, 3}	{2}	0.0728	1.0	0.0769	1.0	0.0769	0.2774	0.5385	1.0559
{2, 19}	{3}	0.0728	0.8462	0.1429	0.8462	0.1392	0.3477	0.4945	1.6593

Fig. 33: **Teaching** dataset minsup = **0.15** and minraresup=**0.05**

MinSup	Minraresup	supp	Conf.	allconf	maxconf	coherence	Cosine	Kulc	Lift
0.01	0.001	0.0066	1.0	0.147	1.0	0.147	0.3091	0.5735	22.1939
0.01	0.002	0.0066	1.0	0.147	1.0	0.147	0.3091	0.5735	22.1939
0.05	0.01	0.0136	1.0146	0.1225	1.0146	0.1239	0.3014	0.5686	9.4301

Fig. 34: Mean Values of **Teaching** dataset

B. Results Interpretation

1) ***BMSWebview1****:* It has been observed in the BMSWebview1 dataset, as the range between the minimumrare support and minimum support increases, number of association rules also increases as depicted in the Fig 7. In the case of minraresup = 0.001, and minsup = 0.01, the total number of association rules generated are 528 compared to 1146 association rules fetched between the range of minraresup = 0.001 and minsup = 0.05. We observed that no association rules were generated when the minraresup was set starting from 0.01 to varying minsup , such that minsup $>$ minraresup .

Fig 10 , 11 and 12 shows three randomly chosen association rules generated for different minsup and minraresup.

We observed that the higher the range between minsup and minraresup, the closer the relationship between the Antecedent and consequent. For instance, in the case of minsup = 0.05 and minraresup = 0.001 , which is the maximum range that we tried for this dataset, all the highest statistical measures occurred under this range which in turn implies that the antecedent and consequent are strongly correlated. But in contrast, from the mean value Table 13 of BMSWebview1, the all-conf and cosine values are near to 0 which implies antecedent and consequent are negatively correlated, with kulc nearing to 0.5 results in neutral relationship whereas Maxconf heading towards 1 implies that antecedent and consequent are strongly associated. This results in ambiguity in the relationship between the antecedent and consequent itemsets. In order to solve this, Imbalance ratio can be examined. Imbalance ratio (IR)is the ratio which access the imbalances between the itemsets. It ranges from 0 to 1. More the imbalance ratio, larger the difference between the itemsets. If IR = 0 , then we can say that the itemsets are balanced. This ratio is independent to the number of null-transactions and total number of transactions in the dataset. In our study, we are not computing IR and can be taken up as a future work.

2) ***Kosarak****:* Similar to the BMSWebview1 dataset, for kosarak also, as the range between the min rare support and minimum support increases, the number of association rules also increases as mentioned in the Fig 8. Having observed from tables, diversified statistical measures from the kosarak dataset have resulted. It is difficult to argue that for which setting of minsupport and minraresup gives us the values for the statistical measures that can unequivocally prove that there exists a strong relationship between the antecedent and consequent. A similar observation has been found for the mean values of the statistical measure of the kosarak dataset which can be seen in the Fig 18.

3) ***Online Retail****:* As from the Fig 5 it can be observed that, in the case of Online Retail dataset, as the range between the min rare support and minimum support decreases, the number of association rules also decreases. Interestingly, for the Online Retail dataset, all the statistical measures were closer to 1 i.e antecedent and consequent are strongly correlated and for the study of interestingness, we would recommend using minsup = 0.01 and minraresup = 0.005 to get higher interesting patterns among rare itemsets.

4) ***Skin and Teaching****:* Skin and Teaching datasets behave similar to other datasets i.e as the range between the min rare support and minimum support increases, the number of association rules also increases as mentioned in the Fig 6 and 9. For very sparse datasets, we hypothesize that these statistical

measures create obscurity among the rare itemsets. Since Skin and Teaching datasets are very sparse, we cannot conclude which statistical measure provides more interestingness among the rare item sets as all_conf and cosine shows negatively correlated while kulc measures neutral and max_conf claims strongly positively associated. This results in skewness in the datasets. Therefore, Information ratio (IR) can serve as a complementary measure to further evaluate the degree of skewness of the data. [16]

VII. Conclusion and future work

In this paper, we have discussed the comparison of six statistical measures for five different sparse datasets. We observe that there is no measure that is consistently better than others in all cases. Nonetheless, there are instances in which few measures are highly correlated with each other, e.g., lift values are more than 1 i.e., highly positively associated due to its sensitivity to null-transactions which resulted in poor distinguishing pattern association relationships and coherence [6] being not suitable for rare item sets . Hence, we considered four statistical measures i.e. allconf, maxconf, cosine, kulc for comparison.

Having observed such ambiguity in the results, due to the balanced skewness of the datasets, unfortunately, there is no perfect answer for " which measure shows more or less interestingness among itemsets?" For future work, it would be interesting to see how these null-invariant interestingness measures work for dense datasets and high dimensional datasets using the Negative Tree.

References

[1] J. Han, M. Kamber, and J. Pei, *Mining Frequent Patterns, Associations, and Correlations*, pp. 243–278. 12 2012.

[2] J. Han, J. Pei, and Y. Yin, "Mining frequent patterns without candidate generation," *Mining Frequent Patterns Without Candidate Generation*, pp. 1–12, 01 2000.

[3] W. Song, B. Yang, and Z. Xu, "Index-bittablefi: An improved algorithm for mining frequent itemsets," *Knowledge-Based Systems*, vol. 21, pp. 507–513, 08 2008.

[4] G. Liu, H. Lu, J. Yu, W. Wang, and X. Xiao, "Afopt: An efficient implementation of pattern growth approach," 12 2003.

[5] M. Adda, L. Wu, and YiFeng, "Rare itemset mining," pp. 73 – 80, 01 2008.

[6] L. Szathmary, A. Napoli, and P. Valtchev, "Towards rare itemset mining," vol. 1, pp. 305–312, 11 2007.

[7] Y. Lu, F. Richter, and T. Seidl, "Efficient infrequent pattern mining using negative itemset tree," in *Complex Pattern Mining - New Challenges, Methods and Applications* (A. Appice, M. Ceci, C. Loglisci, G. Manco, E. Masciari, and Z. W. Ras, eds.), vol. 880 of *Studies in Computational Intelligence*, pp. 1–16, Springer, 2020.

[8] E. Omiecinski, "Alternative interest measures for mining associations in databases," *Knowledge and Data Engineering, IEEE Transactions on*, vol. 15, pp. 57– 69, 02 2003.

[9] R. Agrawal and R. Srikant, "Fast algorithms for mining association rules," pp. 487–499, 01 1994.

[10] L. Troiano, G. Scibelli, and C. Birtolo, "A fast algorithm for mining rare itemsets," pp. 1149 – 1155, 01 2010.

[11] Y. S. Koh and N. Rountree, "Finding sporadic rules using apriori-inverse," pp. 153–168, 05 2005.

[12] S. Tsang, Y. S. Koh, and G. Dobbie, "Rp-tree: Rare pattern tree mining," pp. 277–288, 08 2011.

[13] T. Imielinski, A. Swami, and R. Agrawal, "Mining association rules between sets of items in large databases," *ACM SIGMOD*, pp. 207–216, 01 1993.

[14] S. Brin, R. Motwani, J. Ullman, and S. Tsur, "Dynamic itemset counting and implication rules for market basket data," *ACM SIGMOD Record*, vol. 26, 12 2001.

[15] T.-Y. Li and X.-M. Li, "New criterion for mining strong association rules in unbalanced events," pp. 362–365, 09 2008.

[16] T. Wu, Y. Chen, and J. Han, "Re-examination of interestingness measures in pattern mining: A unified framework," *Data Min. Knowl. Discov.*, vol. 21, pp. 371–397, 11 2010.

[17] R. Hilderman and H. Hamilton, *Knowledge Discovery and Measures of Interest*, vol. 638. 01 2001.

[18] L. Vu and G. Alaghband, "Efficient algorithms for mining frequent patterns from sparse and dense databases," *Journal of Intelligent Systems*, 08 2014.

[19] S. Darrab, D. Broneske, and G. Saake, *RPP Algorithm: A Method for Discovering Interesting Rare Itemsets*, pp. 14–25. 07 2020.

[20] M. Riondato and F. Vandin, "Finding the true frequent itemsets," 01 2013.