# **Weighted Association Rule Mining**

# **Introduction**

Association rule mining is a procedure which is meant to find frequent patterns, correlations, associations, or causal structures from data sets found in various kinds of databases such as relational databases, transactional databases, and other forms of data repositories. Given a set of transactions, association rule mining aims to find the rules which enable us to predict the occurrence of a specific item based on the occurrences of the other items in the transaction. Association rules are employed today in many application areas including market basket analysis, web usage mining, intrusion detection, continuous production, and bioinformatics.

In order to select interesting rules from the set of all possible rules, constraints on various measures of significance and interest are used. The best-known constraints are minimum thresholds on support and confidence.

|  |  |
| --- | --- |
|  | I |
|  | II |

If A, B are assumed to be item set, according to the Equation I and Equation II, it can be described that support is the fraction of transactions containing the item set and confidence is probability of occurrence of {B} given {A} is present.

The classical model of association rule mining employs the support measure, which treats every transaction equally. In contrast, different transactions have different weights in real-life data sets. For example, in the market basket data, each transaction is recorded with some profit [1]. This led to the emergence of weighted association rule mining (WARM) concept. With this concept, support term has also evolved to weighted support. In order to be able to calculate the adjusted support, firstly transaction weights () are calculated from the item weights. denotes the inner-transaction space for the transaction in transaction space .

|  |  |
| --- | --- |
|  | III |

However, if we consider item weights, calculate the weights of item set according to Equation III and bias the support by multiplying it with the item set weight, a new set of adjusted support values are obtained[2].

There are two types of approach to weighting items or transactions. The first is to obtain rules based on the weighted support value from a weighted items or transactions. The other one is to obtain the rules according to the classical support value from not weighted items or transactions, the weighted support values of these rules are calculated and the elimination process is made according to this weighted support value.

In this study, the two approaches that mentioned above have been tested comparatively.

# **Literature Review**

The notion of association rule was first introduced by R. Agrawal and R. Srikant in 1993 [3]. It proposed the support-confidence measurement framework and reduced association rule mining to the discovery of frequent item sets. In 1994 Apriori algorithm is purposed for finding frequent item sets in a dataset for boolean association rule. Name of the algorithm is Apriori because it uses prior knowledge of frequent item set properties.

In the following years, many algorithms have been proposed to make the rules meaningful or extract the rules faster [4][5]. Featured algorithms: FP-Growth and Eclat are compared in terms of performance and the results are examined [6]. These algorithms strictly follow the classical measurement framework and produce the same results once the minimum support and minimum confidence are given.

However, the traditional association rule mining (ARM) model assumes that items have the same significance without taking account of their weight/attributes within a transaction or within the whole item space. But this is not always the case. For example, [𝑊𝑖𝑛𝑒→𝑆𝑎𝑙𝑚𝑜𝑛] may be more important than [𝐵𝑟𝑒𝑎𝑑 →𝑀𝑖𝑙𝑘] even though the former holds a lower support. This is because those items in the first rule usually come with more profit per unit sale, but the standard ARM simply ignores this difference [2]. As a result of this deficiency the concept of weighted association rule mining (WARM) has emerged.

Weighted support notion based on the weights assigned to both items as well as transactions was introduced by Ramkumar et al [7]. For filter of the rules that have a weighted support larger than a given threshold, an algorithm called WIS was proposed. Unlike Ramkumar et al [7], Cai et al. [8] defined weighted support based on only item weights.

In addition to the ARM, WARM weights items or transactions by differentiating them [7][8][9]. After the emergence of the concept of WARM, how weights are given is a matter of debate. For datasets such as market basket datasets, it was recommended to use the item or transaction cost as weight, whereas a more general weighting method, link based hits algorithm, was proposed [1].

# **Method**

In this study, two approaches have been tried in this section, these approaches are explained step by step. As shown in the Table 1 below, for both approaches a weight value should be assigned to the items.

Table 1. Method stages

|  |  |  |
| --- | --- | --- |
|  | Post-Processed Approach | Pre-Processed Approach |
|  | Assign weight values to items | Assign weight values to items |
|  | Select minimum support | Calculate transaction weights from each item’s weight |
|  | Run Eclat algorithm then gain rules | Select minimum weighted support |
|  | Separate rules to items | Run Weighted Eclat algorithm then gain rules |
|  | Calculate the transaction weight from each separated item’s weight |  |
|  | If weighted support less than selected minimum support, eliminate these rules |  |

## **Post-Processed Approach**

After assigning values to items, minimum support value is determined to restrict the rules. Then Eclat algorithm is used to obtain rules with support values greater than the minimum support values. These rules are rules consisting of items that do not have a weight value. The Eclat algorithm extract the rule, assuming all transactions or items are of the same importance. In this case, weighted support should be calculated according to the weight value specified. The rules with the calculated weighted support value below the minimum support value are eliminated. However, new rules cannot be added because it is filtered again through the rules. So there may be a sifting process at this stage, but the rules cannot be increased.

1. Assign weight values to items:

Table 2. Samples of item’s weights

|  |  |
| --- | --- |
| **Movie** | **Weight** |
| Ğ¡Ğ¾Ğ»ÑÑ€Ğ¸Ñ | 4.138158 |
| Monsoon Wedding | 3.706204 |
| Terminator 3: Rise of the Machines | 4.256173 |
| Ariel | 3.401869 |
| Star Wars | 3.689024 |

1. Select minimum support:

In addition to selecting minimum support, the parameters in which the item sets specify the maximum and minimum number of items are determined.

|  |
| --- |
|  |

1. Run Eclat algorithm then gain rules:

As can be seen in the figure, the rules were extract with the classical support value without weight.

|  |  |  |
| --- | --- | --- |
| **Rule** | **Support** | **Confidence** |
| {Monsoon Wedding, Terminator 3: Rise of the Machines} => {Ğ¡Ğ¾Ğ»ÑÑ€Ğ¸Ñ} | 0.2309985 | 0.8072917 |

Table 3. A sample of rule (R) from Eclat algorithm

1. Separate rules to items:

While calculating weighted support for a rule, the weight equivalent of each item within the rule is sought. In order to achieve this, rules are divided into items.

Table 4. Separated item’s weights from rule (R)

|  |  |  |
| --- | --- | --- |
| **Separated Items** | **Weight** | **Symbol** |
| Ğ¡Ğ¾Ğ»ÑÑ€Ğ¸Ñ | 4.138158 | A |
| Monsoon Wedding | 3.706204 | B |
| Terminator 3: Rise of the Machines | 4.256173 | C |

1. Calculate the transaction weight from each separated item’s weight:

According to the Equation III,

|  |  |
| --- | --- |
|  | IV |

Table 5. Rule with support and weighted support values

|  |  |  |
| --- | --- | --- |
| **Rule** | **Support** | **Weighted Support** |
| {Monsoon Wedding, Terminator 3: Rise of the Machines} => {Ğ¡Ğ¾Ğ»ÑÑ€Ğ¸Ñ} | 0.2309985 | 0.931735 |

1. The weighted support is normalized and finally the weighted support rules that are below the minimum support value are eliminated.

## **Pre-Processed Approach**

Unlike the post-processed approach, the pre-process approach uses the weighted eclat (Weclat) algorithm. The Weclat algorithm generates rules based on process weights. In other words, with the Weclat algorithm more rules than the number of rules from the Eclat algorithm can be obtained. This is not possible in the first approach.

1. Assign weight values to items; Weights are assigned in the same way as in the first approach, as shown in Table 2.
2. The process weight is calculated as shown in Equations 3 and 4.
3. Minimum weighted support is determined since the transaction is carried out directly from the weighted support.
4. Run Weclat algorithm then gain rules:

Table 6. sample of rule from Weclat algorithm

|  |  |  |
| --- | --- | --- |
| **Rule** | **Weighted Support** | **Confidence** |
| {Monsoon Wedding, Terminator 3: Rise of the Machines} => {Ğ¡Ğ¾Ğ»ÑÑ€Ğ¸Ñ} | 0.2309985 | 0.8072917 |

# **Dataset Description**

2 different data sets were used in this project.

## **The Movies Dataset**

This dataset is provided from Kaggle [10] and consists of the following files:

|  |  |
| --- | --- |
| * **movies\_metadata.csv** * **keywords.csv** * **credits.csv** | * **links.csv** * **links\_small.csv** * **ratings\_small.csv** |

These files contain metadata for all 45,000 movies listed in the Full MovieLens Dataset. The dataset consists of movies released on or before July 2017. Data points include cast, crew, plot keywords, budget, revenue, posters, release dates, languages, production companies, countries, TMDB vote counts and vote averages.

This dataset also has files containing 26 million ratings from 270,000 users for all 45,000 movies. Ratings are on a scale of 1-5 and have been obtained from the official GroupLens website.

Among these files, only the files in the table were used;

Table 7. Used files to prepare Movie Dataset

|  |  |  |
| --- | --- | --- |
| **File** | **Columns** | **Description** |
| movies\_metadata.csv | adult, belongs to collection, budget, genres, homepage, id, imdb\_id, original\_language, original\_title, overview, popularity, poster\_path, production\_companies, production\_countries, release\_date, revenue, runtime, spoken\_languages, status, tagline, title, video, vote\_average, vote\_count | The main Movies Metadata file. Contains information on 45,000 movies featured in the Full MovieLens dataset. Features include posters, backdrops, budget, revenue, release dates, languages, production countries and companies. |
| ratings\_small.csv | userId, movieId, rating, timestamp | The subset of 100,000 ratings from 700 users on 9,000 movies. |

## **Online Retail Dataset**

This dataset is provided from UCI [11]. This is a transnational dataset which contains all the transactions occurring between 01/12/2010 and 09/12/2011 for a UK-based and registered non-store online retail. This dataset consists of only one file and has 541909 instance, 8 attribute.

Table 8. Columns and column’s description of Online Retail Dataset

|  |  |
| --- | --- |
| **Columns** | **Description** |
| InvoiceNo | Nominal, a 6-digit integral number uniquely assigned to each transaction. If this code starts with letter 'c', it indicates a cancellation. |
| StockCode | Product (item) code. Nominal, a 5-digit integral number uniquely assigned to each distinct product. |
| Description | Product (item) name. Nominal. |
| Quantity | The quantities of each product (item) per transaction. Numeric. |
| UnitPrice | Unit price. Numeric, Product price per unit in sterling. |
| CustomerID | Customer number. Nominal, a 5-digit integral number uniquely assigned to each customer. |

# **Data Preparation**

## **Data Preparation for Movies Dataset**

First the empty rows in the dataset were deleted. Items and transactions were weighted in two different ways.

### **Assigning weights of a movie**

Two methods are used;

1. Average rating for a movie (Avg): The average rate for a movie is calculated with the data obtained from the rating small.csv file and this is used as weight.
2. Number of views of a movie (Count): It is calculated how many people watch a movie with data from the rating small.csv file and this value is used as weight.

### **Assigning weights of a transaction**

Two methods are used:

1. HITS: Link based HITS algorithm used to weight transactions.
2. The average rating for all the movies that a user watched is calculated and used as the transaction weight.

## **Data Preparation for Online Retail Dataset**

* Dataset has some empty cells: These cells did not consider.
* Canceled invoices are removed from dataset.
* The invoice amount for each invoice was calculated and used as the weight of that transaction.

# **Experimental Results**

As shown in Table 9, when the support and confidence values were kept constant for the same dataset, there was a difference in item set and rule numbers obtained from different weighted datasets. However, it is possible to see increase of rule numbers in the pre-processed method, although there was no increase of rule numbers in post-processed method.

Table 9. Experiment results for Movies Dataset

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Weights** | **Algorithm** | **Min. Support** | **Confidence** | **#Item-sets** | **#Rules** |
| - | Eclat | 0,2 | 0,8 | 69 | 9 |
| Avg. | Post-Processed | 0,2 | 0,8 | 45 | 6 |
| Avg. | Weclat (Pre-Processed) | 0,2 | 0,8 | 75 | 10 |
| Count | Post-Processed | 0,2 | 0,8 | 58 | 8 |
| Count | Weclat (Pre-Processed) | 0,2 | 0,8 | 87 | 14 |
| HITS | Weclat | 0,2 | 0,8 | 115305 | 734147 |

Table 10 shows changes in the number of items and rules resulting from the change in support value.

Table 10. Experiment results for Online Retail Dataset

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Weights** | **Algorithm** | **Min. Support** | **Confidence** | **#Item-sets** | **#Rules** |
| - | Eclat | 0,01 | 0,8 | 333 | 18 |
| Invoice amount | Weclat | 0,01 | 0,8 | 19 | 3 |
| - | Eclat | 0,015 | 0,7 | 87 | 11 |
| Invoice amount | Weclat | 0,015 | 0,7 | 3 | 4 |
| - | Eclat | 0,02 | 0,8 | 36 | 3 |
| Invoice amount | Weclat | 0,02 | 0,8 | 0 | 0 |

As the figure shows, as the support value increased, the number of rules exponentially increased.

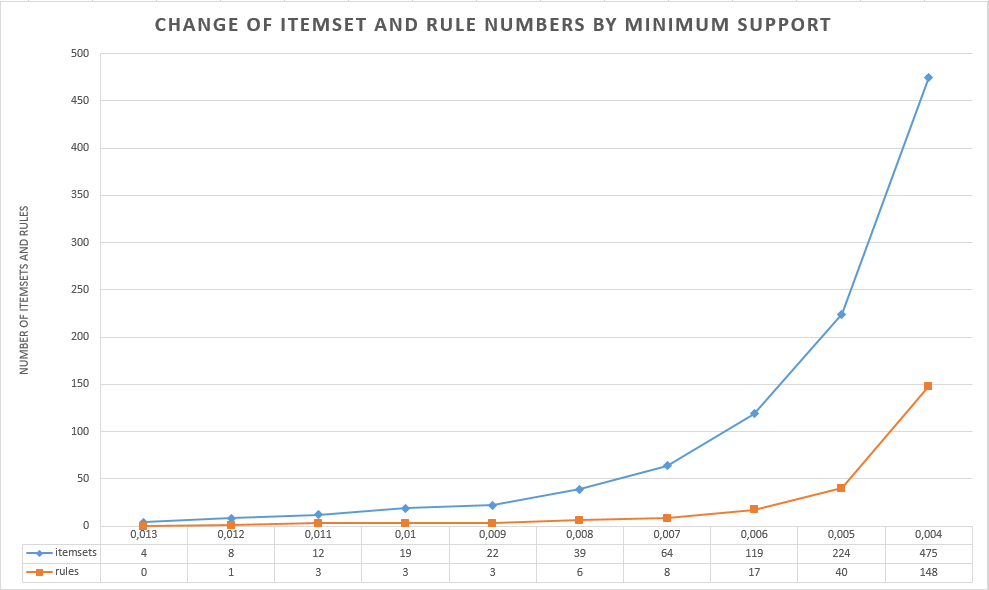


Figure 1. Change of item set and rule numbers by min. support

# **Conclusion**

In the [2] study, it is mentioned that the weighted support equation used in the post-processed approach is contrary to the understanding of downward closure property. It is also clear that the rules can never increase with the same approach. These implications disrupt the reliability of the results obtained from this approach. The weighted support equation used in the pre-process method is a more recent method and may increase the number of rules outside the filtering method.

REFERENCES

[1] K. Sun and F. Bai, “Mining weighted association rules without preassigned weights,” *IEEE Trans. Knowl. Data Eng.*, vol. 20, no. 4, pp. 489–495, 2008.

[2] F. Tao, F. Murtagh, and M. Farid, “Weighted Association Rule Mining using Weighted Support and Significance Framework,” *KDD 2003 Proc. ninth ACM SIGKDD Int. Conf. Knowl. Discov. Data Min.*, pp. 661–666, 2003.

[3] R. Agrawal, T. Imielinski, and A. Swami, “Mining Association Rules between Sets of Items in Large Databases,” *Proc. 1993 ACM SIGMOD Conf.*, p. 10, 1993.

[4] S. Brin, R. Motwani, J. D. Ullman, and S. Tsur, “Dynamic itemset counting and implication rules for market basket data,” *ACM SIGMOD Rec.*, vol. 26, no. 2, pp. 255–264, 2005.

[5] J. S. Park, M.-S. Chen, and P. S. Yu, “An Effective Hash-Based Algorithm for Mining Association Rules.”

[6] S. Pramod and O. P. Vyas, “Survey on Frequent Item set Mining Algorithms,” *Int. J. Comput. Appl. (0975 - 8887)*, vol. 1, no. 15, pp. 86–91.

[7] G. D. Ramkumar, S. Ranka, and S. Tsur, “Weighted Association Rules: Model and Algorithm,” *Proc.ACM SIGKDD*, pp. 1–13, 1997.

[8] C. H. Cai, A. W. C. Fu, C. H. Cheng, and W. W. Kwong, “Mining association rules with weighted items,” *Proc. - IDEAS 1998 Int. Database Eng. Appl. Symp.*, pp. 68–77, 1998.

[9] W. Wang, J. Yang, and P. S. Yu, “Efficient mining of weighted association rules (WAR),” no. February, pp. 270–274, 2004.

[10] “The Movie Dataset.” [Online]. Available: https://www.kaggle.com/rounakbanik/the-movies-dataset#movies\_metadata.csv.

[11] D. D. Chen, “Online Retail Dataset.” [Online]. Available: https://archive.ics.uci.edu/ml/datasets/online+retail.