Fuzzy Weighted Association Rule Mining with Weighted Support and Confidence Framework

Muhammad Sulaiman Khan [1]
Dr Maybin Muyeba [1]
Dr Frans Coenen [2]

[1]Liverpool Hope University
[2]Liverpool University

Outline of the Presentation

Organised as follows:

- **Introduction**
  - Classical Association Rule Mining (ARM)
  - Fuzzy Association Rule Mining
  - Downward Closure Property (DCP)
  - Weighted Association Rule Mining
    - Weighted ARM Support & Confidence
    - Classical WARM Types
    - Classical WARM Issues
- **Our Contribution**
  - Binary Weighted Association Rule Mining (BWARM)
  - Fuzzy Weighted Association Rule Mining (FWARM)
- **Problem Definition, Methodology & Application**
- **FWARM Algorithm**
- **Experimental Results**
- **Conclusion & Further work**

Association Rule Mining

- **Association Rule Mining (ARM)**
  - Data Mining Technique for rule extraction
  - Typically used to determine customer buying Patterns from ***large*** market basket data/Transactions.
  - Association rules are expressions of the form

$$X \rightarrow Y$$

where X and Y are item sets and $X \cap Y = f$

- Classical ARM utilises binary/boolean data

Classical ARM (Boolean)

- **Support**

$$Supp(X \rightarrow Y) = Supp(X \cup Y)$$

- **Confidence**

$$Conf(X \rightarrow Y) = Supp(X \cup Y) / Supp(X),$$

a conditional probability i.e. Given X, we can determine Y.

Fuzzy ARM (Quantitative)

- **Fuzzy ARM**
  - Applies to non-boolean and relational databases with quantitative attributes.
  - Determine rules of the form:

    if (X is A) then (Y is B)

  where X and Y are attributes in a database and A and B are the discretised values of these attributes.
  For example:

    if <Age is Young> then <Salary is Low>

Downward Closure Property (DCP)

- **DCP**
  - In a classical Apriori algorithm it is assumed that if the itemset is large, then all its subsets should also be large and is called Downward Closure Property (DCP).
  - e.g. if {A,B,C} is a frequent set then {A,B}, {A,C} and {B,C} will also be frequent.
- **Application**
  - DCP helps to generate large itemsets of increasing size by adding items to itemsets that are already large.
  - For example, if itemset {A, B} and {B, C} are not frequent, then {A, B, C} and {B, C, D} cannot be frequent so we don't consider generating the supersets that contain non-frequent subsets.
  - Thus, DCP plays important role in efficient ARM algorithm development.

Weighted ARM

- Weighted association rule mining process deals with the significance/importance of individual items in a database
- Items are assigned weights (w) according to their significance as shown in table below.
- These weights are set according to items profit margins

Table1: Weighted Items Database

ID	Item	Profit	Weight	
1	Scanner	10	0.1	…
2	Printer	30	0.3	…
3	Monitor	60	0.6	…
4	Computer	90	0.9	…

Table2: Transactions

TID	Items
1	1, 2, 4
2	2, 3
3	1, 2, 3, 4
4	2, 3, 4

Weighted ARM

- Classical association rule mining (ARM) model assumes that all items have the same significance without taking into account their weight within a transaction or record.

For example rules:

A: [wine → salmon, 1%, 80%] may be more important than
B: [bread → milk, 3%, 80%]

In classical ARM rule ***B*** is more important than rule ***A*** because rule ***B*** has higher support than rule ***A***.

But in weighted ARM with weighted settings rule ***A*** may be more important than rule ***B***, 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.

Classical WARM Types

Post Processing: where all frequent sets are generated using any classical ARM algorithm and later aggregated weights of itemsets are multiplied with their supports in order to find the weighted support.

Pre Processing: In pre processing weighted support is calculated after each step (database scan) using the same formula used in post processing. The only obvious reason for pre processing seems to do early pruning and avoid un-necessary generation of frequent sets.

Comment: post processing may be inefficient (generates all sets) pre processing may be more efficient, may also miss out interesting rules (under DCP in weighted settings, this is solved) i.e. Lemma given in paper: If an itemset is not frequent, then its superset can not be frequent

Classical WARM Issues

- Many potential itemsets are not considered (considering their weights) during the mining step because both approaches (pre-, post) use classical ARM algorithms which normally follows ***DCP*** and the only items considered for weighted support are the already generated frequent sets.
- Itemsets are first generated using their occurrences in the database and later their weights are considered for weighted support. This approach leads to loose many potential itemsets which could be important if their weights are considered as well as occurrence.
- After calculating weighted support, the weighted frequent sets do not hold ***DCP***.
- Exhaustive search is not possible with high number of items due to computational limitation 2^n-1, n is the number of items

Our Contribution: **Weighted Support & Confidence framework**

- We address the Weighted Association Rule Mining issues present in previous approaches and have proposed a **Weighted Support & Confidence framework** for databases with boolean and quantitative/fuzzy attributes.
- In our framework we consider items occurrences and weights together instead of just their occurrences (pruning process) in the database to calculate their support and confidence.
- Thus, our proposed framework reflects not only number of records supporting the itemsets, but also their degree of significance in the dataset.
- Frequent itemset generated using our approach holds valid **DCP**.

Problem Definition 1

Let the input data ***D*** have transactions $T=\{t_1, t_2, t_3 \dots t_n\}$ with a set of items $I=\{i_1, i_2, i_3 \dots i_m\}$ and a set of non-negative, real number weights $W=\{w_1, w_2, w_3 \dots w_m\}$ associated with each item. Each i^{th} transaction t_i is some subset of ***I*** and a weight ***w*** is attached to each item $t_i[i_j]$ (the "j^{th}" item in the "i^{th}" transaction).

Thus each item i_j will have associated with it a weight corresponding to the set ***W***, i.e. a pair ***(i, w)*** is called a weighted item where $i \in I$. Weight for the "j^{th}" item in the "i^{th}" transaction is given by $t_i[i_j[w]]$.

Item Weight ***IW*** is a real value given to each item i_j ranging [0..1] with some degree of importance, a weight $i_j[w]$.

Itemset Transaction Weight ***ITW*** is the aggregated weights of all the items in the itemset present in a single transaction. Itemset transaction weight for an itemset ***X*** can calculated as:

$$\text{vote for } t_i \text{ satisfying } X = \prod_{k=1}^{|X|} (\forall [i[w]] \in X)\, t_i[i_k[w]]$$

Binary Weighted Support & Confidence framework

Binary Weighted Support is the aggregated sum of itemset transaction weight of all the transactions in which itemset is present, divided by the total number of transactions. It is denoted as:

$$\mathrm{WS}(X) = \frac{\sum_{i=1}^{n} \prod_{k=1}^{|X|} (\forall [i[w]] \in X)\, t_i[i_k[w]]}{n}$$

Binary Weighted Confidence is the ratio of sum of votes satisfying both XuY to the sum of votes satisfying X (with Z=(XuY)). It is formulated as:

$$\mathrm{WC}(X \rightarrow Y) = \sum_{i=1}^{n} \frac{\prod_{k=1}^{|Z|} (\forall [z[w]] \in Z)\, t_i[z_k[w]]}{\prod_{k=1}^{|X|} (\forall [i[w]] \in X)\, t_i[x_k[w]]}$$

Fuzzy Weighted Support & Confidence framework

Fuzzy Weighted Support*FWS* is the aggregated sum of ***FITW*** of all the transactions in which itemset is present, divided by the total number of transactions. It is calculated as:

$$\mathrm{FWS}(X) = \frac{\sum_{i=1}^{n} \prod_{k=1}^{|L|} (\forall [i[l[w]]] \in X)\, t_i'[i_j[l_k[w]]]}{n}$$

Fuzzy Weighted Confidence *FWC* is the ratio of sum of votes satisfying both **XuY** to the sum of votes satisfying **X** (with **Z=(XuY)**). It is formulated as:

$$\mathrm{FWC}(X \rightarrow Y) = \sum_{i=1}^{n} \frac{\prod_{k=1}^{|Z|} (\forall [z[w]] \in Z)\, t_i'[z_k[w]]}{\prod_{k=1}^{|X|} (\forall [i[w]] \in X)\, t_i'[x_k[w]]}$$

Lemma

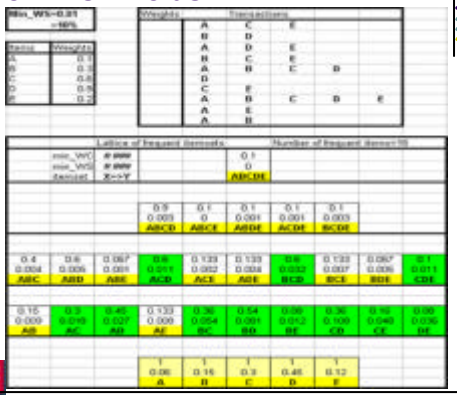
If an itemset is not frequent then its superset cannot be frequent and is always true

$WS(subset) \geq WS(sueprset)$

Proof

Note: WS is greater with fewer items because of product operator e.g in WS (tx) against WS (ty), where tx is a subset of ty

Proof - DCP Holds



FWARM Algorithm

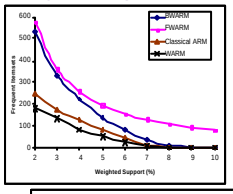
- The proposed Fuzzy Weighted ARM (FWARM) algorithm is required to be efficient as processing of fuzzy weights
- The algorithm uses tree data structures [13] and similar in fashion to Apriori algorithm (with valid DCP).
- In Algorithm:
  - C_k is the set of candidate itemsets of cardinality ***k***.
  - ***w*** is the set of weights associated to items ***I***.
  - ***F*** is the set of frequent item sets.
  - ***R*** is the set of potential rules.
  - R^f is the final set of generated fuzzy ARs

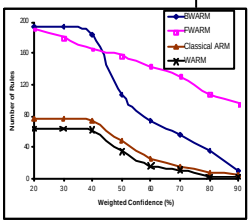
Experimental Results

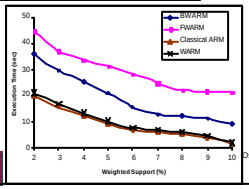
Experimental data is a transactional database containing 100K records and 1K (1000) items. Two sets of experiments were undertaken:

- First experiment show that the output behaviour of our proposed framework is quite similar to classical ARM because we use the Apriori approach in our algorithm but results are better than WARM. Experiments show
  - (i) the number of frequent sets generated (using weighted support measure) and
  - (ii) the number of rules generated (using weighted confidence measure)
- Experiment two shows comparison of execution times using different weighted supports and data sizes.

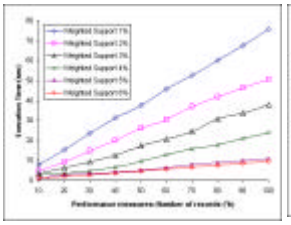
1. Quality Measures

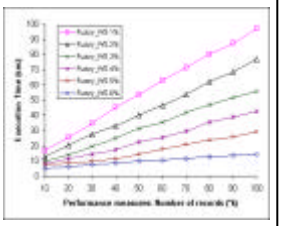






2. Performance Measures





Exceutiion time, vay threshold 0.01 to 0.06, conf =0.5
Algorithm scales linearly with FWS, similar to classical ARM

Conclusion & Further Work

- We have presented novel approach for extracting hidden informatıon from Weighted items. our proposed framework reflects not only number of records supporting the itemsets, but also their degree of significance in the dataset.
- The problem of invalidation of downward closure property (**DCP**) is solved using improved model of weighted support and confidence framework for binary and fuzzy association rule mining
- We showed the application of our method on different datasets.
- Overall, the approach presented here is effective and efficient for analysing databases with Weighted items (boolean or quantitative/fuzzy).

Future Work

- There is potential to apply FWARM to other applications with Composite attributes even with varying fuzzy sets between attributes.
- Different measures for validating DCP, normalisation of values e t c .
- We intend to extend our work byintegrating Weighted ARM with Utility ARM in transactional and relational databases containing weights and utilities of items . It is both a new approach and algorithmically challenging.
- Mechanisms for tuning membership degrees of itemsets without bias or human expert involvement e.g. genetic algorithms