A landmark model for assigning item weight for pattern mining

SONGWUT PISALPANUS

A thesis submitted to
Auckland University of Technology
In partial fulfillment of the requirements for the degree of
Master of Computer and Information Science (MCIS)

2012

School of Computing and Mathematical Sciences

Table of Contents

List of Figures

List of Tables

List of Abbreviations and Terms

- ARM: Association Rule Mining
- Conf: Confidence of rule represents the proportion of support of rule over support of rule antecedence
- Global Graph: A graph structure containing N nodes(items) where N is the entire set of items in the item universe (dataset)
- Hypercliques: A group of items that interact strongly with each other, having a group h-confidence not lesser than a user specified h-confidence threshold
- Interaction weight: The weight acquired by an item through its connections with its neighbors
- ISRF: Information System Research Framework
- KDD: Knowledge Discovery in Database
- minConf: Minimum confidence for rules that is required for it to be identified as interesting rules
- minSup: Minimum frequency of occurrence for items or itemsets that is required for it to be identified as frequent (large) items or itemsets
- Overall weight: the weight of an item that represents both its domain weight and the acquired interaction weights from its neighbors.
- PC: Proportional Confidence
- Rule Lift: The ratio between the rule's confidence and the support of the itemset in the rule confidence
- Sub-Graph: A subset of independent non-overlapping items of the partitioned global graph containing M nodes (items) where M is a group of items that interact strongly with each other. The term sub-graph is used interchangeably with the term Localized Approach in this thesis.
- Supp: Frequency of occurrence for items or itemsets
- Transmission weight: The interaction weight obtained by an item by seeding the domain weight of its neighbors to a random small weight, which is much smaller than the lowest possible domain weight taken over the entire set of items.
- WARM: Weighted Association Rule Mining
- WT: Weight Transmitter

Attestation of Authorship

“I hereby declare that this submission is my own work and that, to the best of my knowledge and belief, it contains no material previously published or written by another person nor material which to a substantial extent has been accepted for the qualification of any other degree or diploma of another university or institution of higher learning, except where due acknowledgements are made.”

Yours Sincerely,

(Songwut Pisalpanus)

Acknowledgements

I would like to sincerely thank to my supervisor Dr. Russel Pears. I acknowledge endless helps, advices and supports he gave me since I started doing this thesis. His extensive knowledge and his logical way of thinking have been of great value for me. His perceptive thinking and supervision have provided a good basis for the thesis. I also thanks for the way he kept me on track and focused on finding solutions for the issues encountered along the way.

I also sincerely extend thanks to Dr. Yun Sing Koh from the University of Auckland for her helps, advices and supports especially in the practical parts in this thesis.

Abstract

In weighted association rule mining, items are typically weighted based on background domain knowledge. However, it may not be feasible to gather domain information on every item in high dimensional datasets especially in a dynamically changing environment. Thus, it is more practical to exploit domain information to set weights for only a small subset of items and then estimate the weights of the rest through the use of a suitable interpolation mechanism. In the recent study (Koh et al., 2012), weight transmitter model was proposed. The weight transmitter model uses a subset of items, termed landmark items, whose weights are known in advance to propagate known weights to the rest of the items with unknown weights.

In this study, we seek to extend this approach by improving performance of the weight transmitter model while seeking to lower the percentage of landmark items employed in the weight estimation process. Firstly, we propose a new interestingness measure called *Proportional Confidence*, which is derived from the standard confidence measure, to use as a measure for quantifying interactions between items. Secondly, we propose a novel method to partition a global graph into a number of smaller sub-graphs called *Sub-graph generation algorithm* by utilizing divide-and-conquer approach. Thirdly, we propose a new method used in allocating landmark items by utilizing stratified random sampling approach. The results of our experiments show that our proposed landmark items assignment produces higher performance in terms of Precision, Recall, Accuracy, Lift and Execution Time compared to the original simple random sampling while our proposed sub-graph approach substantially reduces time complexity in the weight fitting process.

We also investigate the impact of our proposed weight transmitter approach compared to weighting with the domain based approach in relation to cases where sharp differences arose in the assignment of weight values to the same item. The results from the in depth study show that our proposed weight transmitter approach is in a better position to assign item weight as it takes into account interactions between items.

Chapter 1 Introduction

Computers have been extensively used in many areas for different purposes. With respect to many advantages of computers such as quicker processing time, the vast amount of primary data storage along with an ability to connect with virtually unlimited secondary data storage, computers have become the main equipment that many organizations use to extract meaningful information from the massive amount of data in their databases. The process of extraction of this meaningful information is widely known as knowledge discovery in database (KDD) (Fayyad et al., 1996).

KDD is defined as the process of extraction of information from huge volumes of data in database in order to derive hidden meaningful information and patterns which lead to the extraction of knowledge from stored data (Fayyad et al., 1996). Although there are many steps involved in the KDD process as shown in Figure 1.1, the most important of these is a data mining step. This is because it is the main analysis step in discovering patterns from data and involves the use of various algorithms, techniques and applications (Fayyad et al., 1996).

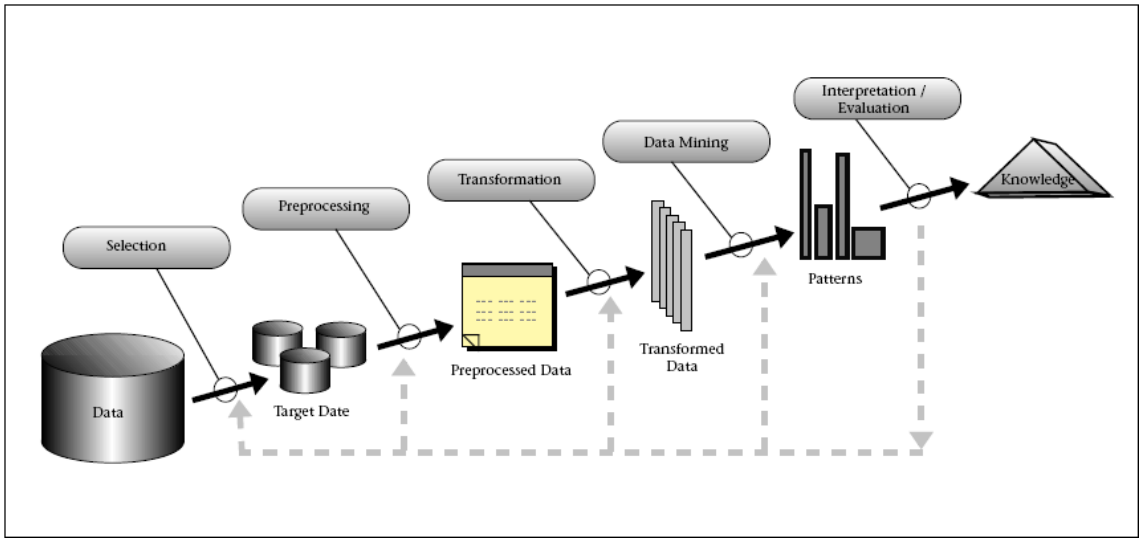


Figure 1.1: KDD Process (Fayyad et al., 1996)

Association Rule Mining (ARM) has been widely acknowledged as a powerful technique for discovering relationships or patterns between variables or attributes in transactional databases (Kantardzie & Srivastava, 2005). It has

been deployed in a wide variety of application domains including health (Gamberger et al., 2003; Kraft et al., 2003), marketing and sales (Berry & Linoff, 1997), manufacturing (Gardner & Bieker, 2000), road traffic control (Hauser & Scherer, 2001), insurance (Gerritsen, 1999; Smith et al., 2000), weather forecasting (Zhang, Wu and Huang, 2004), crime scene investigation (Warren et al., 1999), bioinformatics and in educational data mining, to name but a few from across the full spectrum of applications.

1.1 Research Background and Motivation

Association Rule Mining (ARM) was first introduced by Agrawal et al. (1993), who proposed the landmark Apriori algorithm that overcame the combinatorial explosion involved in finding significant associations between items in high dimensional data that tends to abound in real-world applications.

The search for association rules was initially motivated by the analysis of supermarket transaction data which gave rise to the well-known application known as market basket analysis. One such association rule represents the relationship between bread and jam, which can be written in terms of a rule as *{bread}* → *{jam}*. An interpretation of this rule is that jam would be expected to be bought by customers who buy bread. The main strength of association rules is their explicit capture of significant relationships between items that are implicit or hidden in large data repositories. However, a fundamental issue with the traditional association rules mining algorithm is that even a modest sized dataset can produce thousands of rules, and as datasets get larger, the number of rules produced becomes unmanageable (Koh et al., 2011).

In this context it is important to be able to isolate the most interesting or useful rules from the rest which express trivial facts or relationships that are ultimately of little or no value to a decision maker. Standard methods of managing the size of rule bases generated include the use of constraints such as rule support and rule confidence. While support based measures for filtering rules can be useful, the fundamental limitation of support based measures is that they do not attack the problem of discriminating between interesting rules and their less interesting

counterparts. This is due to the fact such measures are fundamentally statistical in their basis and thus may not capture domain specific knowledge on the utility of the rules generated from a decision maker perspective.

At a fundamental level, association rules capture correlations among sets of items based solely on the degree of co-occurrence between items, rather than taking into account the inherent properties of the items themselves. A consequence of focusing on the degree of associations, expressed through the rule support measure, is that low frequency or rare rules that capture important domain specific knowledge may be filtered out through the application of the rule support measure. Thus for example, Champagne and Caviar are expensive and high profit items that are bought with much lesser frequency than staple items such as bread or milk. However, the purchase of Champagne may act as a trigger for the purchase of Caviar, thus giving rise to a strong association rule Champagne → Caviar. However, the extraction of such a rule will require the lowering of the rule support threshold considerably, with the attendant risk of causing the rule base to explode in size, ultimately burdening the decision maker with the task of sifting the interesting rules from the ones that are not. A further negative consequence is the time required to generate the rules will increase, thus lowering the efficiency of the entire process as most of the rules generated will ultimately be discarded by the end user.

Weighted Association Rule Mining (WARM) was proposed as a new paradigm to address the issues (Tao et al., 2003; Wang et al., 2004; Yan and Li, 2006; Jian and Ming, 2008) stated above. The core concept of WARM is based on the assignment of weight to items which reflect the importance of the items from a domain perspective. Thus with an item weight scheme in place, an opportunity exists for rare but high profit items such as Champagne and Caviar to figure prominently in rules. In addition, a weight assignment scheme can also aid in ranking the rules generated, which in general will lead to a reduction in time consumed by end users in identifying the set of valuable rules (Pears et al., 2010).

With the concept of WARM described above, an important issue that then arises is the development of a suitable weight assignment scheme. Many

previous studies have relied on pre-assigned weights supplied by users on the basis of specialized domain knowledge. For instance, item profit is generally used as a basis for assigning weights to items in a retail market application (Barber and Hamilton, 2003). Similarly, page dwelling time is commonly used to assign weight to a web page in order to specify the relative importance of different web pages in a web application domain (Yan and Li, 2006).

With respect to the assignment of weights to items, two main approaches have been proposed in the literature: the domain approach where item weights are assigned solely on the basis of domain knowledge and the other in which item weights are inferred on the basis of interactions between items in a transactional database. Numerous different algorithms have been proposed that take advantage of the domain based approach; they differ mainly in the method used to combine individual item weight into weights for a set of items (henceforth referred to as itemsets, in this thesis).

A number of researchers (Sun and Bai, 2008; Koh et al., 2010; Pears et al., 2010) have pointed out the limitations of the domain based approach. Recently Koh et al. (2012) have proposed a third approach that propagates known weights from a landmark set of items to the rest of the items with unknown weights. This research seeks to extend this approach by improving the weight propagation process. These include situations where domain information required to assign weight is either not available or is impractical to collect due to the sheer number of items involved. Even when such domain information is forthcoming the aforementioned problem of imposing known beliefs (in the form of relative ranking of items) may inhibit the discovery of new, unexpected patterns.

The problems associated with the domain based approach prompted Sun and Bai (2008), Koh et al. (2010) and Pears et al. (2010) to formulate various schemes to impute item weights from relationships inherent in data. While such approaches address the issues mentioned above, they do not utilize specialized domain knowledge that may be available which may ultimately prove valuable in guiding the process of weight determination. In response to this a third approach was recently suggested by Koh et al. (2012) that uses domain

knowledge on a small subset of items and a weight propagation method to transmit the known weights to items whose weights are unknown. Their results suggest that high precision and recall among the high weight items can be obtained when only around 30% of the weights of items are specified. This research seeks to extend this approach by improving precision and recall in the extraction of high weight items, while seeking to lower the percentage of seed or landmark items employed in the weight estimation process.

1.2 Research Objective

The main objectives of the research are as follows.

- To develop a weight transmitter model that accurately estimates weights of items by utilizing a small set of items (landmark items) whose weights are known. The transmitter model propagates weights from the landmark set to the other items by utilizing the interactions between items rather than taking into account the inherent properties of items. The major benefit of this approach is that the transmitter model is then independent of the specifics of data, thus enabling it to be applied across any given application domain.

- To improve the efficiency of the weight transmitter approach. The original weight transmitter model proposed by Koh et al. (2012) used Gaussian elimination as the basic method in weight transmission. Gaussian elimination has a worst-case run time of $O(N^3)$ where N is the number of items and thus scalability is an issue with this approach. Our research utilizes a divide-and-conquer approach whereby we partition the graph representing inter-relationships amongst N items into a number of much smaller sub-graphs each of which are subjected to the weight transmitter process. This divide and conquer approach is expected to substantially reduce the run time overhead in the weight fitting process.

- To compare our proposed weight transmitter approach to weighting with the domain based approach with particular reference to cases where

sharp differences arose in the assignment of weight values to the same item. Such cases are interesting and we study such cases in depth to understand the underlying reasons why such sharp differences occur. One of the premises of this research is that the weight transmitter model is in a better position to assess weight vis-à-vis its domain based counterpart as it takes into account interactions between items. An in depth study that analyses such cases will help to determine the truth of this premise.

1.3 Thesis Structure

In this Chapter, we have provided background knowledge and motivation together with challenges associated with weighted association rule mining.

In Chapter 2, we review previous research in the area of weighted association rule mining, with particular emphasis on existing techniques and approaches in assigning item weight. We also briefly describe research on rule generation and rule evaluation metrics.

In Chapter 3, we give a formal definition of the weight assignment problem. We then go on to present the original weight transmitter model proposed by Koh et al. (2012) and the basic tools used in its implementation, which include the Gaussian elimination method for solving a linear model.

Chapter 4 presents our research methodology and a research architecture that extends Koh et al. (2012)'s weight transmitter approach. Our proposed architecture includes novel methods for assigning landmark items and for partitioning a global graph of N items into several sub-graphs.

In Chapter 5, we present our experimental design. The objectives of each experiment are presented and the performance metrics used to assess results are explained.

In Chapter 6, the experimental results are presented. We also analyze the impact generated by the methods that we propose.

In Chapter 7, we conduct two case studies with datasets obtained from the University of Auckland and the1998 World Soccer Cup competition.

In Chapter 8, we summarize our research with a discussion of the key achievements, including the impact of the novel contributions made by this research. We also outline different directions in which future research can be undertaken for further improvement.

Chapter 2 Literature Review

2.1 Introduction

The classical association rule mining method has thrived since it was introduced by Agrawal et al. (1993). The seminal Apriori algorithm introduced by Agrawal has undergone many different enhancements and new approaches altogether have been proposed, although the original conceptual basis behind association mining persists to this day. In this chapter we briefly trace through the early developments of this field before covering more contemporary research on its major variant, weighted association rule mining which has direct relevance to this research.

2.2 Association Rule Mining: its beginnings and transition to Weighted Association Rule Mining

In a landmark paper Agrawal et al. (1993) formally defined frequency based association rules and presented an efficient algorithm called Apriori that generates all association rules and meets user-defined thresholds on the itemset support and rule confidence measures. The basis of Apriori and its variants is the downward closure property that states that for any given item X that is frequent; all of its constituent items (subsets of X) must also be frequent. Frequency is defined in terms of a minimum support threshold; any item whose support (frequency) is above the specified threshold is said to be frequent. The implication of the downward closure property is that large parts of the search space can be pruned without inspection thus making the problem of finding frequent itemsets more efficient. The major bottleneck in generating association rules is identification of frequent itemsets that are the building blocks for the formation of rules. Thus, under the Apriori formulation association rule mining consisted of two major phases: frequent itemset generation, followed by rule formulation. Since the publication of Apriori numerous attempts (Park et al., 1995; Toivonen, 1996; Brin et al., 1997; Agarwal et al., 2000; Holt and Chung, 2001) have been made to optimize its performance by introducing efficient data structures such as dynamic hashing, tree hashing, etc. to efficiently scan large

datasets and compute support of candidate itemsets in order to assess whether they meet the support requirements.

Apart from optimizing performance another theme of research has been the application of user defined constraints (Srikant et al., 1997; Bayardo et al., 2000; He and Han, 2003; Yao and Hamilton, 2006) with the twin goal of engaging users and embedding semantics into the rule generation process. One commonly used type of constraint is the item constraint, by which users would explicitly state the types of items that he/she is interested in, and thus constrain the rules generated to only include such items. For example, a dairy company may be interested in only mining baskets (transactions) containing only the dairy products. Such research while taking a different approach from ours shares a common goal of embedding a degree of user defined beliefs into the rule generation process with the motive of ensuring that more relevant and useful rules would emerge as a result. Although such research addresses the issue of generating irrelevant rules to a certain extent it still fundamentally works within the Apriori frequency domain context. The problem with the frequency domain context is that interesting items that do not survive the frequent itemset generation phase will not manifest in rules, in effect incurring a loss in valuable information. This problem required a fundamental paradigm shift whereby the "interestingness" of items is considered to be on an equal footing with that of frequency, thus giving birth to the field of weighted association rule mining.

2.3 Weighted Association Rule Mining

Numerous algorithms have been proposed to overcome the limitation of classical association rule mining. Many of these algorithms replace an item's support with a weighted form of support. Each individual item is assigned a weight to reflect its importance. Items that are considered highly interesting will be assigned a higher weight. This approach is known as Weighted Association Rule Mining which was first given a formal definition by Cai et al. (1998).

In Weighted Association Rule Mining, a weight w_i is assigned to each item i, where $0 \leq w_i \leq 1$, to show the relative importance of an item over all other items.

The weighted support of an item *i* is then $w_i \cdot sup(i)$. Similar to the classical association rule mining, a weighted support threshold and a confidence threshold will be assigned to measure the strength of the association rules. An itemset, X, is considered a large itemset if the weighted support of this itemset is greater than the user-defined minimum weighted support (wminsup) threshold.

$$\left(\sum_{i \in X} Wi\right) \text{supp}(X) \geq wminsup$$

The weighted support of a rule X→Y is:

$$\left(\sum_{i \in X \cup Y} Wi\right) supp(XY)$$

An association rule X→Y is called an interesting rule if X ∪ Y is a large itemset and the *conf(X→Y)* is greater than or equal to a minimum confidence threshold where the term *conf* denotes *confidence* which follows the definition of traditional association rules (Agrawal et al., 1993).

Many algorithms have been proposed by utilizing domain information in order to assign weights to items. We will structure research on WARM into three major themes: the first containing research into algorithms that use domain knowledge as the basis for assigning item weight. Research in this theme concentrates on finding the most appropriate method of combining individual item weights into weights for sets of items. Theme 2 approaches the WARM problem from a completely different perspective: it takes the position that sufficient domain information may not be available in certain situations in order to assess item importance and hence assigns weights to items using an automated approach based on the inter-connections between items in transaction dataset. Theme 3 represents a very recent development: a hybrid approach is taken whereby the weights for a subset of items are assigned on the basis of domain knowledge and the weights of the rest of the items are inferred through a weight propagation process.

2.3.1 Domain based Weighted Association Rule Mining

Ramkumar et al. (1997) was amongst the very first attempts at researching the WARM problem. They assigned weights to items on the basis of domain knowledge and then used item weights as the basis to infer the weights of transactions that occur in a retail market application. They indicated that assigning weights to transactions provided the ability to bias the rule generation process to transactions of high importance. In their approach, rules that had weighted support greater than the user-defined wminsup threshold were generated, similar to traditional Apriori (Agrawal et al., 1993).

Tao et al. (2003) utilized item profit for setting weights for items in a retail application. They aimed to focus on strong relationships amongst highly weighted items while filtering out relationships (both strong and weak) amongst lowly weighted items. However, they pointed out that the downward closure property was violated with the use of weighted support. This is due to the fact that an itemset can be considered as large even though some of its subsets are not large due to weighting of support by item weight. Therefore, they proposed a new algorithm called weighted downward closure property that made use of weight for both items and itemsets. The itemsets whose weighted support is larger than the threshold were considered as significant itemsets. Their result showed that the selection of significant itemsets is steered to those itemsets participating in relationships with high weight items.

Wang et al. (2004) extended the classical association rule mining paradigm by allowing weights to be associated with items in transactions in order to reflect the interest/intensity of items in transactions. For example, 70% of people buying more than four bottles of beers will also be likely to buy more than three packs of potato chips. In their approach, they first calculated frequent itemsets without considering the weights of items and then introduced weight during the process of rule generation. In particular, they segment the domain weight space of each frequent itemset and then identify regions that contain transactions that are heavily populated with such segments in order to derive association rules. They demonstrated that their method not only improves the confidence of the rules, but also provides a mechanism for more effective targeted marketing by

categorizing customers on the basis of their level of loyalty or volume of purchases.

Yan and Li (2006) utilized time taken by a user to view a webpage to estimate its importance in a transaction in order to capture the user's interest more precisely in a web recommendation application. The main idea behind using page viewing duration to assign weights to webpages is because it reflects the relative importance of each webpage. Users generally spend greater time on a more useful page that they are interested in. In their approach, they assigned a weight to each webpage that reflected the average dwelling time a user spends on the page. Weighted association rule mining was employed to discover significant page sets. In addition, to reflect the dynamics of web applications, they allowed page weights to vary in time. As such, the weight of a particular webpage could increase as it became more popular and users spent more time viewing it. On the other hand, the opposite could happen and the weight would then decrease. Their results showed that a significant improvement in recommendation effectiveness could be obtained in comparison to using classical association rule mining.

In the study of Jian and Ming (2008), they utilized item sequence sets (ISS) for improving the efficiency of weighted association rule mining in the application area of alarm correlation in communication networks. They sought to reflect the importance of items (alarms) appearing in transactions by employing an analytic hierarchy process (AHP) that associated the equipment and the level of alarms together to compute the status of equipment in the form of a judgment matrix. They also pointed out that accurate alarm weights should be based on a combination of objective information of alarms and a subjective judgment given by domain experts. In their approach, they first selected two attributes and made use of a method based on subjective judgment to build a judgment matrix that reflects the degree of relative significance amongst different values of the selected attributes. Then, they calculated the weights of all alarms by multiplying weights of the values of different attributes through relative significance degrees of corresponding attributes by adding all of them together. With their proposed method, it enabled the judgment matrix to be more objective and the weight of alarms to be more flexible and understandable.

Many weight assignment algorithms were proposed in various application domains. However, it can be seen in previous studies that the process of weight assignment has relied on pre-assigned weight by end users based on their subjective judgments and specialized knowledge of the domain area. The major issue with relying on subjective judgment is that unexpected rules with high importance are unlikely to be discovered because rules generated from subjective input are based on previous experience that reflect known, observed patterns. In addition, weight can be pre-assigned only with respect to applications when the required information is readily available.

2.3.2 Automated Weight Association Rule Mining

Since item weight cannot be pre-assigned for some datasets, there is a need to implement a generic solution for weight assignment that takes into account the inter-relationships between items in transactional data.

Lin and Shyu (2010) proposed a new algorithm to assign weight to items (feature-value pair) in video based semantic concept detection applications. They tried to bridge the gap between low-level features and high-level feature concepts. In their approach, they first utilized Multiple Correspondence Analysis (MCA) to project the features in a new principal component space to determine relationships among feature-value pairs and categories. Then, they incorporated both correlation information from MCA and percentage of the frequency counts of positive and negative sign of each feature as the measurement to assign weight for feature-value pairs. The performance of their method was then compared against other well-known algorithms on the benchmark TRECVID dataset. The results showed that their algorithm achieved higher performance in identifying fifteen targets taken from the TRECVID dataset in terms of Recall and F1-score while giving a competitive Precision when compared against other well-known algorithms such as Decision Tree, Support Vector Machine, Naive Bayesian, Neural Network, and K-Nearest Neighbor.

Sun and Bai (2008) proposed a concept called w-support to assign weight to items without the need for domain specific input. The main idea behind their

approach is that the importance of an item is governed by the transactions in which it occurs. Their items weights were assigned from the property of the dataset based on the assumption that a good transaction, which is highly weighted, should contain many good items, and conversely, good items should be contained within many goods transactions. They utilized the HITS algorithm (Kleinberg, 1999) to rank the transactions. In their approach, the dataset was first converted into a bipartite graph containing two set of nodes representing items and transactions. They then calculated hub-weight with the adapted HITS model (Kleinberg, 1999) to rank the transaction. A w-support measure, that reflected the weight of an item, was then calculated as the proportion of the hub-weight of the transactions containing that item by the hub-weight of all transactions in the dataset. The method was shown to work well on sparse datasets in finding some significant itemsets that lead to a discovery of interesting patterns involving rare items. However, its performance was very similar to that of Apriori on dense datasets as the w-support measure and raw support values converged for dense datasets.

Pears et al. (2010) proposed a weight assignment mechanism that was based on Principal Components Analysis (PCA). Their approach was based on the concept that items should be weighted based on the degree of variation that they captured across the dataset. In their approach, the matrix containing all possible Eigen vectors was generated from the covariance matrix computed on the dataset. They then ranked items based on the Eigen vector with the largest Eigen value, as that vector is responsible for capturing the largest degree of variation across the dataset. Weights were then assigned to items on the basis of the level of expression of the items in the vector. Results from their study show that concise rules with high information content could be generated when compared to the standard Apriori approach that does not employ item weighting.

Koh et al. (2010) proposed a Valency model for the weight assignment problem. The intuition behind their approach is that items should be weighted on the basis of the strength of their connections to other items as well as the number of items that they are connected with. They modelled a transaction dataset as a bipartite graph consisting of items and links between items. The Valency model

was defined in terms of the graph and consisted of major components, connectivity and purity. The purity of a given item was represented as a function of the number of items that it was associated with, when taken over the entire transactional database, while connectivity between a given pair of items captured the strength of the connections between that pair of items. Koh et al. evaluated the rules produced from their weighting scheme based on the degree of variation captured and compared their approach against Apriori. The rules produced by the Valency model were evaluated against a measure that recorded the degree of variation captured across the dataset. The Valency model was shown to be significantly better than Apriori on this measure.

Following on from this, Koh et al. (2011) extended the Valency model to operate in a data stream environment. Results from their extended model showed that the evolving version significantly speeded up execution time while maintaining a high level of accuracy when compared to a simplistic method that simply recomputed the entire set of item weights at fixed intervals.

2.3.3 Semi-Automated Weight Association Rule Mining

Recently Koh et al (2012) proposed a semi-automatic approach to the weight assignment problem. This approach was motivated by the fact that domain information, whenever available, should be exploited in the weight assignment process. Unlike their earlier Valency model, weight assignment was not carried out solely on the basis of linkages between items but also on the basis of domain-supplied weights for a subset of items for which such weight information was available. The weights from this subset, referred to as the landmark subset were fed into a weight propagation model that transmitted weight from the landmark set to the other items whose weights were unknown. Experimentation was conducted on datasets where the ground truth about the complete set of weights was known in advance. Experimentation showed that high degrees of precision and recall were obtained with relatively low sizes of the landmark set, comprising just 20% to 30% of the total number of items that existed across the dataset as a whole.

Chapter 3 elaborates on this approach as it was used as the foundation for the current research presented in this thesis.

2.4 Label Propagation

Label propagation is one of the graph-based algorithms that have been utilized in many studies in the field of semi-supervised learning. Label propagation algorithms operate in an environment where data objects are classified into two or more categories or labels, but the labels for some of the objects are not known. Propagation algorithms based on Gaussian kernel functions propagate labels from known objects to their counterparts whose labels are unknown. Thus, some similarity with a semi automated approach to the item weight assignment problem is apparent, if we associate items to data objects and labels to item weights. In view of this, we compare our weight propagation model with the one proposed by Bengio et al. in (Bengio et al., 2006).

However, we note that labels are discrete entities and are thus not the exact equivalent of numerical weights. Despite this, we believe that sufficient similarity exists with the problem being investigated and thus a comparison with our proposed propagation model is justified.

A more detailed presentation of the model is given in Chapter 3.

2.5 Summary

In this chapter, we have provided a brief outline of past research into the item weight assignment problem within the context of weighted association rule mining. It was evident from the review that very little work exists in the semi automated approach which we will be exploring in depth in this thesis.

In the next chapter we present in detail the weight propagation model proposed by Koh et al. and highlight aspects of this approach that we will be extending in this research.

Chapter 3 Semi-Automatic Weight Assignment: Models and Methods

3.1 Introduction

In this chapter, we first present the formal definition of the weight assignment problem. Because our proposed model is based on the weight transmitter model proposed by Koh et al. (2012), we will present it in detail and discuss the basic tools and methods used in its implementation. We then highlight some aspects of this original model that we will be extending in this research.

3.2 Problem Definition

There are two distinct approaches to Weighted Association Rule Mining. The most commonly applied approach, corresponds to the situation where all weights are provided directly by the domain application expert on the basis of their subjective judgment or knowledge (Tao et al., 2003; Wang et al., 2004; Yan and Li, 2006; Jian and Ming, 2008). On the other hand, the pure automatic approach is also in existence, whereby no knowledge or subjective judgment is supplied apart from the patterns of interaction of items with each other (Sun and Bai, 2008; Koh et al., 2010; Pears et al., 2010).

However, as mentioned in Chapter 1, there is a need for a third approach that exploits domain knowledge on a small subset of items that acts in conjunction with a weight propagation method to transmit the known weights to items whose weights are unknown, as suggested by Koh et al. (2012). The weight fitting problem that they framed was to estimate the weight of items in terms of their *overall* weight. They reasoned that the weight of an item should not merely reflect its own perceived importance but also take into account its interactions with other highly weighted items. For example, retailers often reduce their profit margin on items that already have relatively low profit and market them as a package deal involving high profit items. A concrete example that was quoted in Koh et al. is that of a discount on a mobile handset that is conditional on the customer signing a long term contract with the phone company involved. In

such situations, the “low profit” item (mobile handset) is used as an incentive to entice customers into buying the high profit items (calling plan contract). Clearly, in such contexts the actual profit margin of the low profit item does not accurately reflect its importance.

To model such situations Koh et al. introduced an interaction weight in their model. The model that they proposed had two sets of weights: domain weights *dw,* which are only available for items in the landmark set *L; and* interaction weights *iw* which exist for every single item since this weight can be inferred from the pattern of co-occurrences with other items in the transaction dataset.

Given a set of items *I*, a set of landmark items *L* where $L \subset I$, and a transaction dataset *D*, the overall or acquired weight w_i of a given item *i* is determined by:

$$w_i = \frac{\sum_{l \in L} iw(i,l) \cdot (w_l + dw_l) + \sum_{m \in M} iw(i,m) \cdot (w_m)}{\sum_{k \in N} iw(i,k)}$$

Equation 3.1

where *N* represents the set of neighbors of item *i*, $dw_i \geq 0$ when *i* is the item belonging to the set of landmark items *L,* else $dw_i = 0$, *M =I-L.*

Thus an item *i* acquires a weight from its interactions with its neighbors who transmit their own weights in a quantity proportional to the degree of interaction *iw*. Neighbors that are landmarks transmit their domain weights dw_l as well as their acquired weights w_l whereas neighbors which are not landmarks (items in set *M*) only transmit their acquired weights w_m.

The accuracy of the weight estimation mechanism expressed by Equation 3.1 above is determined by the measure used to specify interaction weights. Koh et al. used the Gini Information index as it measures the degree of dependence of a given item i on any other item j in its neighbourhood. However, other measures exist that model the degree of interaction between items and Chapter 4 describes some of the measures investigated in this research.

We are now in a position to formally define the problem being investigated in this research: For a given set of landmark items L the problem of determining the set of top ranked items can now be stated formally as follows:

Return all items $i \in H$ where:

$$H = \{i | i \in I \text{ is in the top } p\% \text{ of items when ranked on acquired weight from } Eq3.1\}$$

Thus the focus of this research is to determine the best method to be used in identifying top ranked items as defined by the expression above. Once this determination is made, standard rule generation algorithms such as Apriori can be used to generate association rules.

3.3 Weight Transmitter Model Specification

A graph structure (N, E) was utilized by Koh et al. to develop the model. Nodes in the graph are represented by items while edges represent associations between pairs of items. Each node *i* is associated with the weight w_i of an item, while an edge between items *i* and *j* is represented by *G(i, j)* where G is Gini Information Index (Raileanu and Stoffel, 2004). A high value of *G(i, j)* indicates that item *j* occurs with a high degree of probability when item *i* occurs; conversely, a low value of *G(i, j)* indicates that item *j* occurs with a low degree of probability when item *i* occurs.

	B	$\overline{B}$	
A	f_{11}	f_{10}	f_{1+}
$\overline{A}$	f_{01}	f_{00}	f_{0+}
	f_{+1}	f_{+0}	N

Table 3.1: 2-way contingency table for items A and B

As shown in Table 3.1, each entry f_{ij} in this 2x2 table denotes a frequency count. For example, $f_{11}$is the number of time A and B occur together in the same transaction, while $f_{01}$is the number of transactions that contain B but not A. The row sum f_{1+} represents the support count of A, while column sum f_{+1}

represents the support count for B. Given the 2-way contingency table for items A and B above, Gini Index of items A and B is calculated by:

$$G(A,B) = \frac{f_{1+}}{N} \cdot \left[(\frac{f_{11}}{f_{1+}})^2 + (\frac{f_{10}}{f_{1+}})^2 \right] + \frac{f_{0+}}{N} \cdot \left[(\frac{f_{01}}{f_{0+}})^2 + (\frac{f_{00}}{f_{0+}})^2 \right] - (\frac{f_{+1}}{N})^2 - (\frac{f_{+0}}{N})^2$$

The Weight Transmitter model expresses the weight of a given item *k* in terms of the weights of its neighbors as:

$$Wk = \frac{\sum_{i \in S_1} G(i,k) \cdot (w_i + dw_i) + \sum_{j \in S_2} G(j,k) \cdot (w_j)}{\sum_{i \in S_1} G(i,k) + \sum_{i \in S_2} G(i,k)}$$

Equation 3.2

where S_1 represents the set of neighbours of item *i* whose domain supplied weight dw_i are known in advance, while S_2 represents the set of neighbours of item *i* whose domain weights are unknown. The term $\sum_{i \in S_1} G(i,k) + \sum_{i \in S_2} G(i,k)$ represents a known quantity c_{1k}, since all G index values can be calculated from the transaction database. The dw_i terms in the set S_1 also represent known quantities. We denote $\sum_{i \in S_1} G(i,k) \cdot dw_i$ by c_{2k}. With a substitution of the constants c_{1k}, c_{2k}, Equation 3.2 can now be re-written as:

$$c_{1k} \cdot w_k - \sum_{i \in S} G(i,k) \cdot w_i = c_{2k}$$

Equation 3.3

where $S = S_1 \cup S_2$ represent the complete neighbourhood of item k. The Equation 3.3 represents a system of *k* linear simultaneous equations in *k* unknowns which has an exact solution with the deployed Gaussian Elimination method.

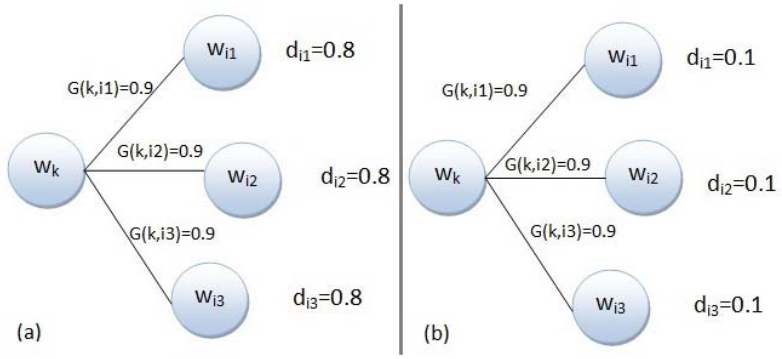


Figure 3.1: Influence of neighborhood in Weight Estimation (Koh et al., 2012)

Figure 3.1 shows two different scenarios involving four items. In scenario 1, represented by Figure 3.1 (a), an item *k* exists with unknown domain weight which interacts with items i_1, i_2 *and* i_3, each of which has known domain weights of 0.8. The item k is strongly connected to each of i_1, i_2 *and* i_3 with a G value of 0.9. In this scenario the Weight Transmitter model returns a weight value of 2.4 for each of the items, which yields a value of 0.89 being normalization to range of [0,1].

Now consider the second scenario (Figure 3.1 (b)) with the same set of items but with supplied domain weight of 0.1. The Weight Transmitter now returns a much lower value of 0.11 when compared to the first scenario. This example illustrates that when an item with unknown domain weight is strongly connected to high weight items through high G values it will acquire a high weight, whereas when the same item connects to low weight items, a low acquired weight results, regardless of the strength of the connections. Hence it is clear that the neighborhood plays an important role in determining the true weight of an item.

3.4 Methods Used

3.4.1 Gaussian Elimination Method

Gaussian Elimination method was deployed as a core method in solving linear equations generated from the weight transmitter model.

Gaussian Elimination is one of the most popular techniques for solving simultaneous linear equations of the form [A][X] = [C]. It was named after Carl Friedrich Gauss who initially developed the method. The Gaussian Elimination method consists of two steps: (1) Forward Elimination; (2) Back Substitution.

The goal of Forward Elimination is to transform the coefficient matrix into an upper triangular matrix (Gentle, 1998).

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \rightarrow \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ 0 & a_{22} & a_{23} \\ 0 & 0 & a_{33} \end{bmatrix}$$

The goal of Back Substitution is to solve each of the equations using the upper triangular matrix (Gentle, 1998).

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ 0 & a_{22} & a_{23} \\ 0 & 0 & a_{33} \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \\ X_3 \end{bmatrix} \rightarrow \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix}$$

The Back Substitution process is represented by:

$$X_i = \frac{b_i^{(i-1)} - \sum_{j=i+1}^{n} a_{ij}^{(i-1)} X_j}{a_{ii}^{(i-1)}} \qquad for\ i = n-1, n-2, \ldots, 1$$

And

$$X_n = \frac{b_n^{(n-1)}}{a_{nn}^{(n-1)}}$$

Equation 3.4

3.4.2 Label Propagation

Label propagation algorithms operate in an environment where data objects are classified into two or more categories or labels, but the labels for some of the objects are not known. In this research, we compare our weight propagation model with the one proposed by Bengio et al. (2006). Their Propagation algorithms utilized Gaussian kernel functions to generate a weight matrix to

propagate labels from known objects to their counterparts whose labels are unknown. In their algorithm, the weight matrix W: $W_{ij}$is non-zero iff X_i and X_j are neighbors and

$$W_{ij} = e^{-\frac{||x_i - x_j||^2}{2\sigma^2}} \quad ; where\ \sigma\ is\ width\ of\ Gaussian\ kernel$$

Equation 3.5

The following are the steps involved in the algorithm:

- Compute an affinity matrix W from Equation 3.5
- Forcing $W_{ii} = 0$
- Choose a parameter $\alpha \in (0,1)$ and a small $\epsilon > 0$
- $\mu \leftarrow \frac{\alpha}{1-\alpha} \in (0, +\infty)$
- Initialize $\hat{Y}^{(0)} \leftarrow (y_1, \ldots, y_l, 0,0, \ldots 0)$
- Iterate $\hat{y}_i^{(t+1)} \leftarrow \frac{\sum_j W_{ij}\hat{y}_j^{(t)} + \frac{1}{\mu}y_i}{\sum_j W_{ij} + \frac{1}{\mu} + \epsilon}$ for a labeled $(i \leq l)$ until convergence.
- Iterate $\hat{y}_i^{(t+1)} \leftarrow \frac{\sum_j W_{ij}\hat{y}_j^{(t)}}{\sum_j W_{ij} + \epsilon}$ for an unlabeled $(l+1 \leq i \leq n)$ until convergence.

However, as mentioned in Chapter 2, labels are discrete entities and are thus not the exact equivalent of numerical weights. The weight matrix W in this algorithm is then replaced by our proposed weight propagation matrix while following the rest of the steps taken of the algorithm.

3.5 Further Extensions to the Weight Transmitter Model

One of the further extensions to the basic Weight Transmitter model is the use of a partitioning scheme to reduce the run time complexity of the weight fitting process. The Gaussian elimination method used in Weight Transmitter has a worst case run time of $O(N^3)$ where N is the number of items. Therefore, scalability is an issue with this approach. To improve scalability, we utilize a divide-and-conquer method to partition the graph representing inter-relationships amongst N items into a number of smaller sub-graphs, each of

which are subjected to the weight transmitter process. The algorithm used in the partition process and an analysis of the run time of the resulting decentralized system is discussed in depth in Chapter 4.

The other major extension relates to the method used in allocating landmark items. In the Weight Transmitter formulation Koh et al. utilized a simple random sampling technique in order to assign landmarks items. Simple random sampling has the important advantage of being efficient to implement. However, we believe that landmarks should be assigned with care and not at random, as they play a central role in guiding the weight fitting process. This is due to the fact that item neighbourhood determines the weight of an item. As such, neighbourhoods need to be representative of the ground situation; if the majority of items surrounding a given item belong to the high weight category, then the items chosen as landmarks in its neighbourhood need to reflect this fact; allocating a majority of low weight items as landmarks in this situation will introduce severe error to the weight fitting process. To reflect the ground truth a bias needs to be introduced into the landmark allocation process; that bias is towards allocating high weight items in some cases and in other cases the opposite bias in favour of low weight items. This bias simply cannot be achieved with an inherently unbiased process such as random sampling. Chapter 4 describes in depth the landmark allocation process.

3.6 Summary

In this chapter, we have presented the formal definition of the weight assignment problem together with the original weight propagation model proposed by Koh et al. We also briefly discussed some major extensions that will be made to this model.

In the next chapter we will present our research methodology and research architecture that extends Koh et al.'s weight transmitter model.

Chapter 4 Research Methodology

4.1 Introduction

In this chapter, we present the research design framework that fundamentally governs the chosen methods used in order to achieve the objectives of this research. We then go on to present our proposed weight transmitter architecture that extends Koh et al.'s weight transmitter approach.

4.2 Research Paradigm

The three major research approaches in the field of information systems are the Positivist, Interpretivist and Critical paradigms. *Orlikowski and Baroudi (1991)* explained that there are different guidelines within these three paradigms that can be used to build a conceptual framework.

Straub, Gefen and Boudreau (2004) stated that the foundation of the positivist research paradigm is the discovery of knowledge or theory that can be verified with the use of rigorous methods. Auguste Comte, the French philosopher suggested that in positivist research, experimentation and observation are exploited in order to understand the discovered domain knowledge and theory (Dash, 2005). Orlikowski and Baroudi (1991) also suggested that formal propositions, measureable degree of variables and theory testing procedures have to be provided for the discovered theory in positivist research.

The field of data mining research is more closely aligned with a positivist research paradigm rather than the Interpretivist or Critical research paradigms. This is because the main objective of data mining is to optimize the value of data and explore and extract meaningful information and knowledge from the data in databases through the process of experimentation and observation (Dash, 2005). Therefore, this thesis will be based on the positivist research paradigm.

4.3 Research Framework

Klabbers (2006) described that design science, unlike natural science, aims to implement and evaluate artefacts, which is consistent with the thesis objectives that aim to develop and assess a weight transmitter model. March and Smith (1995) also explained that design science seeks to develop valuable artefacts not for the purpose of understanding reality but rather to change reality. In addition, in design science, the implemented artefacts can be assessed by various rigorous methods and thus it aligns with the positivist research paradigm. Therefore, a suitable framework for implementing and evaluating weight transmitter model is design science.

The Information System Research Framework (ISRF) was first introduced by Hevner et al. (2004) as a conceptual research framework that offers key solutions to problems based on the design science concept. This framework considered research in the field of information systems as a problem solving exercise that solves problems in a certain environment by applying existing information and knowledge. Originally, Hevner et al. (2004) divided ISRF into seven processes. These seven processes consist of (1) design as an artifact; (2) problem relevance; (3) design evaluation; (4) research contribution; (5) research rigor; (6) design as a search process; (7) research communication. We have adopted the ISRF introduced by Hevner et al. (2004) to guide the overall research design. The ISRF specifies four distinct processes as shown in Figure 4.1.

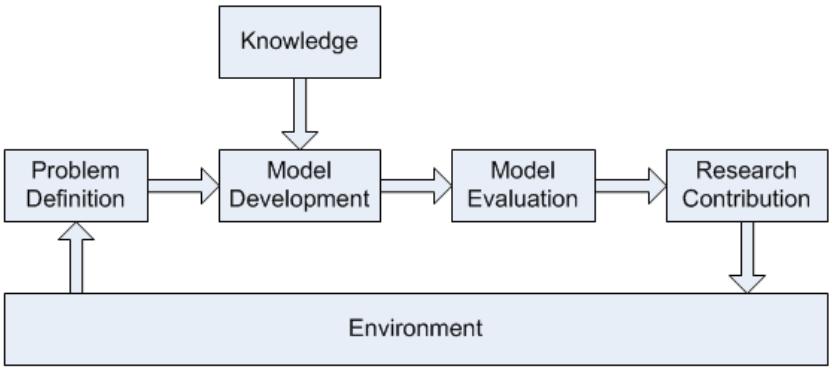


Figure 4.1: Research framework (adapted from Hevner et al., 2004)

4.4 Proposed Weight Transmitter Architecture

The proposed weight transmitter architecture which guides the model development step referred to in Figure 4.1 is illustrated in Figure 4.2. Our model consists of two phases:

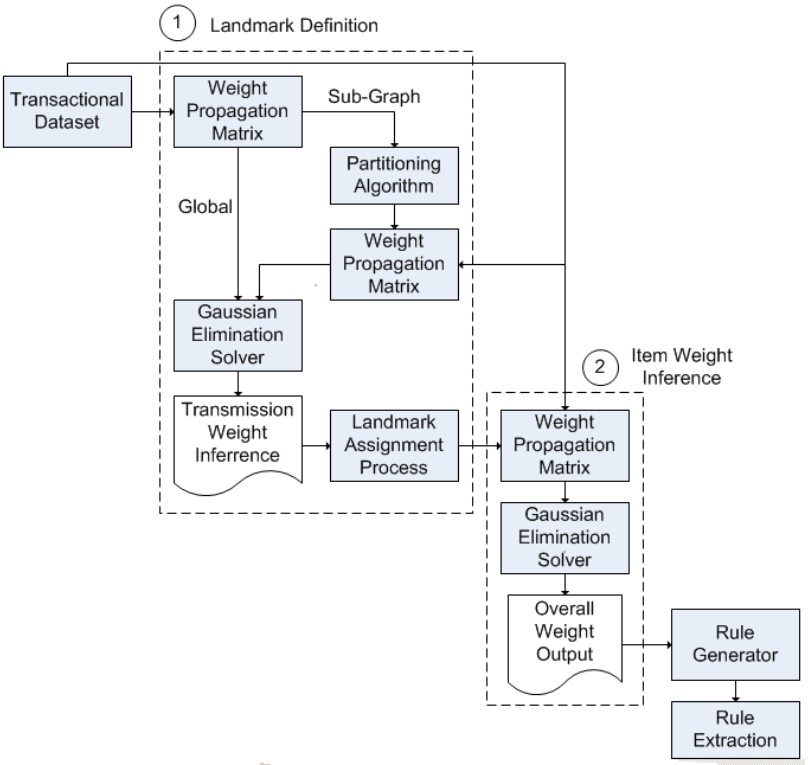


Figure 4.2: Proposed Architecture

Phase:1 (*called Landmark Definition*) is an extension to the Weight Transmitter model proposed by Koh et al. This phase features two novel methods that we propose: (1) a novel method of sub-graph generation (2) a novel method of landmark assignment, both of which will be described in detail later in this chapter.

Phase:2 (*called Item Weight Inference*) follows basically the same process as the original Weight Transmitter model proposed by Koh et al. However, we replaced the Gini Index measure by a measure called *Proportional Confidence* to specify interaction weight between items. The definition of this measure will be presented in the next section.

4.5 Enhancements to Original Weight Transmitter Model

4.5.1 Proportional Confidence: A new measure for quantifying interactions between items

In our proposed model, we utilized a novel interestingness measure called *Proportional Confidence* to represent the degree of association between pairs of items. Proportional Confidence is derived from the standard confidence measure.

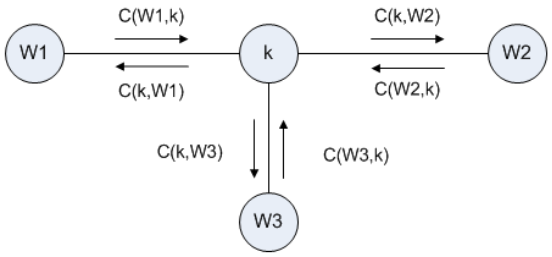


Figure 4.3: Connection between items

Consider Figure 4.3, with the use of the standard confidence measure, the interaction between items *W1* and *k* can be captured by *C(W1,k)*; similarly *C(W2,k)* quantifies the interaction between items *W2* and *k*. In the reverse direction, item *k* simultaneously transmits its weight to item *W1* with *C(k,W1)* and to item *W2* with *C(k,W2)*.

The proportional confidence between a given pair of items, say (*i,j*) is given by:

$$\frac{C(j,i)}{\sum_{l \in N(i)-\{j\}} C(l,i)}$$

Equation 4.1

where the set *N(k)* is the set of neighbors of item *k*.

The confidence term *C(j,i)* reflects the empirical conditional probability of seeing item *j* when item *i* occurs, and is measured as *supp(j,i)/supp(i)*, where supp denotes the support measure. Thus the Proportional Confidence measure,

PC(i,j) is the confidence of observing item *j* expressed as a fraction of the aggregate individual confidences of seeing the neighbors of *i* (with the exclusion of item *j*) whenever item *i* occurs.

Thus, with proportional confidence the contribution of a given item *j* to the weight of *i* is in direct proportional to its confidence with item *i* relative to the confidences of the other neighbors of item *i*.

Proportional Confidence has the effect of amplifying the difference between the strong and weak connections when compared to the original confidence measure. This implies that items with strong connections to a given item, say *k,* will tend to dominate the determination of the overall weight of item *k*. For example, consider the scenario where *C(W1,k)*=0.8, *C(W2,k)*=0.1 and *C(W3,k)*=0.1. With the use of the original confidence measure the relative contribution of *W1* over *W2* (as well as *W3*) to the weight of *k* is 0.8/0.1=8. However, with the use of proportional confidence, the contribution of *W1* over *W2* increases significantly to (0.8/0.2)/(0.1/0.9)=36 which represents an increase by a factor of 4.5. This bias towards stronger connections increases when the gap between the stronger and weaker connections widens. In the context of the weight fitting problem a bias towards stronger connections is necessary to increase the precision of the weight fitting process as our results in Chapter 6 demonstrate. By boosting the contribution made by items with stronger connections, our modified weight transmitter model is better able to discriminate between neighbors that are highly weighted from those that are lowly weighted. Thus, in determining the weight of an item, say *k*, its neighbors that simultaneously have high weight as well as high connectivity to item k will dominate the determination of the weight value of item *k*.

With the proportional confidence measure in place, we now present our proposed model for weight propagation as:

$$Wk = \sum_{i \in S1 \cup S2} \Big(\sum_{j \in S1 - \{k\}} C(j,i) \cdot (Wj + dWj) \cdot \sum_{l \in N(j)} \Big(\frac{1}{C(j,i)} \Big) + \sum_{j \in S2 - \{k\}} C(j,i) \cdot (Wj) \cdot \sum_{l \in N(j)} \big(\frac{1}{C(j,i)} \big) \Big)$$

$$where\ dWi \geq 0\ if\ i\ is\ a\ landmark\ item$$

Equation 4.2

The set *S1* represents the set of neighbours of item *k* whose domain supplied weight are known in advance, while set *S2* represents the set of neighbours of item *k* whose domain supplied weight are not known. It can be seen from Equation 4.2 that the confidence terms within each of the inner summations can be expressed in terms of proportional confidence as these terms represent a ratio of confidences in exactly the same manner as defined by Equation 4.1.

Equation 4.2 represents a system of *k* linear equations in *k* unknowns which has an exact solution with the Gaussian Elimination method which we employ.

4.5.2 Global vs. Local Approach to Weight Propagation

Koh et al.'s model of weight propagation was global in nature whereby the entire set of items in the item universe was subjected to the weight fitting process. However, in principle a local approach to weight propagation can also be applied, whereby the global set of items is partitioned into non-overlapping subsets of items and each subset of items is independently subjected to the weight fitting process.

However, the challenge with the local approach is to efficiently determine local subsets in such a way that items that have significant relationships with each other manifest in the same partition. This suggests a clustering approach. However, the strategy that we employ is more efficient than what could be obtained with a standard clustering approach, although we do preserve the spirit of clustering in the partitioning process.

This section will first present a global approach to weight transmission and then go on to describe the novel local approach.

4.5.2.1 The Global Weight Transmission Model

With the global approach to weight propagation, N linear equations are derived from the Weight Transmitter model where N is the number of items which later

are solved by the Gaussian Elimination method. The following is the algorithm that is used for global weight transmission.

Weight transmission algorithm

Parameters:

- Let 'T' be the transaction dataset
- Let 'PC' be the proportional confidence between pairs of items
- Let 'M' be the weight propagation matrix containing the proportional confidence (PC) between each pair of connected items
- Let 'ItemList' be the list of items in the dataset
- Let 'LW' be the map of weights of all items assigned as landmarks
- Let 'ADJ' be the list of neighbors of each item
- Let 'X' be the map of output weights of each item
- Let 'W' be the map of normalization output weights of each item

Method:

```
1: Read user supplied landmark weight of each item and store in LW
2: n = len(ItemList)
3: Generate matrix M[n][n+1] and populate with 0
4: For every item k in ItemList do the following
5:     Go through all transactions in T and obtain list of neighbors of k and store in ADJ
6:     For every item a in ADJ do the following
7:         Calculate PC(a,k)
8:         Store PC(a,k) at M[IndexOf(k)][IndexOf(a)]
9:         If LW[a] <> 0 then
10:            dw = -1 * LW[a] * PC(a,k)
11:            Store dw at M[IndexOf(k)][IndexOf(k)+1]
12:        End if
13:    End For
14: End For
15: Solve linear equations stored in M using standard Guassian Elimination and store
      the output weight of each item in X
16: Calculate Wmax as maximum weight in X
17: Calculate Wmin as minimum weight in X
18: For every item x in X do the following
19:    W[x] = (X[x] – Wmin) / (Wmax – Wmin)
20: End For
```

Description:

The algorithm first starts by obtaining a list of all items from the dataset. A zero valued matrix of size N by N+1 (where N the number of items) is then created for use as a weight propagation matrix. For each item, a list of its neighbors is compiled from the transactions occurring in the dataset. The proportional confidence between an item and each of its neighbors are then computed and stored in the matrix. If the neighbor happens to be a landmark item, its landmark weight is multiplied by its PC and the result is stored in the in the appropriate row and column of the matrix, thus populating the propagation matrix. The system of linear equations is then solved with the Gaussian Elimination method which returns a list of overall weights of all items. These overall weights are later normalized into a range of [0, 1].

4.5.2.2 Proposed Local Approach to Weight Transmission

A set of linear equations generated from the Global Weight Transmitter Model can be solved with standard Gaussian Elimination. Unfortunately, Gaussian Elimination has a worst case run time complexity of $O(N^3)$ time, thus severely impacting its scalability to high dimensional datasets. With this in mind we designed a novel method to partition a global set of items and generate subsets of items with the *Sub-graph generation algorithm*. We utilized a divide-and-conquer approach to partition the global graph into a number of smaller size sub-graphs. Our proposed algorithm is expected to substantially reduce the run time overhead in the weight fitting process. The following is our proposed algorithm for sub-graph generation.

Sub-graph generation algorithm

Parameters:

- Let 'ItemList' be the list of items in the dataset
- Let 'M' be the matrix containing the Proportional Confidence (PC) between each pair of connected items
- Let 'P' be the matrix after pre-processing
- Let 'PCList' be the list of all PC values stored in matrix M
- Let 'MaxDiffPC' be the maximum difference between PC
- Let 'MinSGSize' be the minimum size of sub-graph after merge

- Let ‘ItmSGNo’ be the map of items and their sub-graph no.
- Let ‘ItmIdx’ be the map of items and their index in the matrix
- Let ‘SubGraphNo’ be the running number used when generating sub-graph

Method:

1: Perform steps 1-5 of weight transmission algorithm with all items in the dataset assigned as landmarks. Each item was assigned the very small value of domain weight in order to acquire interaction weights between items without the effect of domain weight when passing through weight transmitter model
2: P = M
3: n = len(ItemList)
4: // Starting pre-processing matrix
5: // Calculate threshold value
6: Read all PC values in M and store in PCList
7: Sort PCList in descending order
8: MaxDiffPC = 0
9: **For** index k in PCList do the following
10: **If** PCList[k] – PCList[k+1] > MaxDiffPC **then**
11: MaxDiffPC = PCList[k] – PCList[k+1]
12: **End if**
13: **End For**
14: **For** every row index r and columns index c in matrix do the following
15: **If** P[r][c] > MaxDiffPC **then**
16: Update value at P[r][c] to 1
17: **Else**
18: Update value at P[r][c] to 0
19: **End If**
20: **End For**
22: // Transform from asymmetric to symmetric matrix
23: **For** every row index r and columns index c in matrix do the following
24: **If** P[r][c] = 1 OR P[c][r] = 1 **then**
25: Update value at P[r][c] and P[c][r] to 1
26: **End If**
27: **End For**
28: // Sub-graph generation
29: Populate ItmSGNo with 0 for all items
30: SubGraphNo = 0
31: **For** columns index c = 0 to n-1 do the following
32: **For** rows index r = c+1 to n-1 do the following

```
33:       If ItmSGNo[ItmIdx[c]] <> 0 then
34:          If ItmSGNo[ItmIdx[r]] = 0 then
35:             ItmSGNo[ItmIdx[r]] = ItmSGNo[ItmIdx[c]]
36:          Else
37:             If ItmSGNo[ItmIdx[r]] > ItmSGNo[ItmIdx[c]] then
38:                Update every item that have the same sub-graph group as sub-graph
                   group of ItmIdx[r] to sub-graph group of ItmIdx[c]
39:             Else If ItmSGNo[ItmIdx[r]] < ItmSGNo[ItmIdx[c]] then
40:                Update every item that have the same sub-graph group as sub-graph
                   group of ItmIdx[c] to sub-graph group of ItmIdx[r]
41:             End If
42:          End if
43:       Else
44:          If ItmSGNo[ItmIdx[r]] <> 0 then
45:             ItmSGNo[ItmIdx[c]] = ItmSGNo[ItmIdx[r]]
46:          Else
47:             SubGraphNo += 1
48:             ItmSGNo[ItmIdx[r]] = SubGraphNo
49:             ItmSGNo[ItmIdx[c]] = SubGraphNo
50:          End if
51:       End if
52:    End For
53: End For
```

Description:

This algorithm starts by performing steps 1-5 of the global weight transmitter in order to obtain a global weight propagation matrix.

The main idea utilized with this method is to separate items that are strongly connected together into several groups or sub-graphs. In order to determine which items should be grouped together, the matrix is first transformed into binary form. A value of 1 is assigned to element (i,j) of the matrix when an item i interacts with an item j with a proportional confidence higher than a computed threshold, otherwise a value of 0 is assigned.

The pre-processing phase first starts by setting a threshold value. We collapse the matrix by generating a vector that contains the full list of all actual

interaction weights expressed in terms of proportional confidence. This list is then sorted in descending order. We then calculate the maximum difference between consecutive proportional confidence values in this list. The time complexity of this operation is $O(N^2)$ as the full matrix needs to be scanned prior to the sorting step.

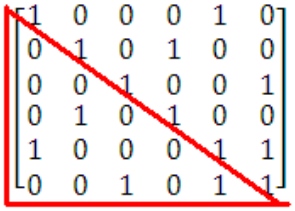
Next, each position in the matrix is compared against the threshold value; the positions that contain a proportional confidence value greater than the threshold value will be assigned the value 1; otherwise value 0 is assigned. The time complexity of this operation is also $O(N^2)$ as the full matrix needs to be rescanned.

The matrix as obtained after thresholding has to be further pre-processed as it is potentially in asymmetric form as the proportional confidence measure by itself is asymmetric in nature. The asymmetry of proportional confidence in its raw numerical form does not cause a problem but when discretized into binary form presents an interesting conflict problem.

For example, if the value obtained after pre-processing at position (1,5) is 0, it is possible that the value at position (5,1) is 1 due to the asymmetry of the underlying proportional confidence values. Both positions (1,5) and (5,1) represent a connection between the same pair of items, 1 and 5. Thus to resolve this conflict, the matrix needs to be further pre-processed in order to transform to a symmetric matrix. To further pre-process the matrix, each symmetric position will be compared against each other; if the value is equal to 1 in at least one of these two positions, both positions will acquire the value of 1. The rationale behind this assignment is that, when at least one position has value 1, it represents the fact that both items are strongly connected, and thus a strong connection should be signaled irrespective of the status of the reverse connection. Again, this step requires a complete scan of the matrix and so the time complexity is $O(N^2)$.

Once the matrix is pre-processed, the partitioning phase can be started. Our proposed algorithm only processes the bottom triangular portion of the matrix as illustrated in Figure 4.4. With the pre-processing completed, sub-graph

generation can now be performed by scanning the bottom triangular matrix. If the value at any given position in the bottom triangular region of the matrix is 1, items that represent the index of the matrix of that position will be grouped together.



	sub-graph number					
	start	through column 1	through column 2	through column 3	through column 4	through column 5
k1	0	1	1	1	1	1
k2	0	0	2	2	2	2
k3	0	0	0	3	3	1
k4	0	0	2	2	2	2
k5	0	1	1	1	1	1
k6	0	0	0	3	3	1

Figure 4.4: Sub-Graph generation

Consider Figure 4.4. Suppose there are six items k1, k2, k3, k4, k5 and k6. Item k1 represents index position 1 in the bottom triangular matrix, item k2 represents index position 2, and so on.

First we set the sub-graph number for each item to 0. We observe that in column 1, rows 1 and 5 have value; thus a sub-graph1 is created, consisting of items k1 and k5. Likewise in column 2, only rows 2 and 4 have value 1, thus giving rise to sub-graph 2 with items k2 and k4. In column 3, rows 3 and 6 have value 1, giving rise to sub-graph 3 with items k3 and k6. With column 4, there are no rows with value 1, thus no new sub-graphs are created. With column 5, which is the last column, rows 5 and 6 have value 1, thus creating overlaps with already existing sub-graphs 1 and 3. Thus instead of creating a new sub-graph, we merge sub-graphs 1 and 3 to reflect the transitivity in connections between items k5 and k6 across sub-graphs 1 and 3. This results in a total of two sub-graphs, sub-graph 1 containing items k1, k3, k5, k6 while sub-graph 2 ends up with items k2 and k4. A total of N*(N-1)/2 operations are involved in this step, once again yielding a time complexity of $O(N^2)$.

The overall time complexity of the algorithm is thus $O(N^2)$. Interestingly, the most time consuming operations involve the pre-processing stage when actual proportional confidence values are discretized. This step though cannot be

avoided as our partitioning phase requires a symmetric matrix under the diagonal.

4.5.3 Novel Approach to Landmark Weight Assignment

The original Weight Transmitter model proposed by Koh et al. utilized Simple Random sampling as the technique in assigning landmark items. Although simple random sample is efficient, it is not the optimal landmark assignment strategy due to the natural skew that exists in the item space. Typically, decision makers and end users are interested in a small subset of highly weighted items, for example in the top-most 20% of items by item weight. In this context, with the use of simple random sampling, 4 times as many landmarks will be assigned to the non interesting category of items when compared to the interesting category, on the average. This will inevitably reduce the accuracy of the weight fitting process as item neighborhoods will be deprived of the high weight landmarks, thus causing items which should be labeled as high weight items to be wrongly classified into medium or even low weight categories, particularly when the number of items employed as landmarks is a small fraction, say 10% of the total number of items that are available. This is a direct consequence of the weight propagation model that we employ whereby an item's weight is determined by the weight of items in its immediate neighborhood.

In order to address the natural skew in the data, we employ stratified random sampling in place of simple random sampling. Thus we allocate an equal number of landmarks to low, medium and high weight categories. However, the all-important question that now arises is how do we divide the items into low, medium and high strata? Domain specified weights cannot be used in this process, as we know only a small fraction of this, hereinafter referred to as the sampling percentage in this thesis. To reflect realistic scenarios in this thesis, we experiment with values of 10%, 20% and 30% as landmark percentages. Knowing at most 30% of the domain weights immediately excludes the possibility of assigning landmarks on the basis of domain weight. Thus another approach is needed to guide the process of landmark assignment.

Basically, our weight propagation model consists of two components or weights: domain weights and interaction weights, the latter of which is measured in terms of the proportional confidence metric. This implies that the only possible solution to assigning landmarks is through an acquired weight for each item which is calculated on the basis of the interaction of that item with items in its neighborhood. To model these interactions we have at our disposal the proportional confidence measure, but this alone is insufficient as the acquired weight requires both a seed weight value for each item to transmit to its neighborhood. Seed values for items need to be assigned with care in order to avoid attributing importance to items that are at variance with the ground truth. In order to resolve this issue we assigned seed values at random from a pool of values, the highest of which were one order of magnitude smaller than the known set of domain weights in the landmark set. We were able to do this with confidence as we knew the entire ground truth. In practice the entire ground truth is unknown and guidance from domain experts is required on a lower bound for domain weight for the set of items. This is not unrealistic as knowledge of the application area and environment are sufficient to deduce a lower bound, unlike the precise determination of the domain weight for each item which may be infeasible.

We are now in a position to present our algorithm for landmark item assignment.

Landmark assignment algorithm

Parameters:

- Let 'LandMarkList' be the list of items assigned as landmarks
- Let 'SampleSize' be the percentage of items required to be assigned as landmarks
- Let 'N' be the number of landmarks
- Let 'TWeightList' be the map of transmission weights of each item

Method:

1: Perform step 1-5 of weight transmission algorithm with all items in the dataset assigned as landmarks. Each item was assigned the very small value of domain weight in order to acquire interaction weight between items without the effect of domain weight when passing through weight transmitter model

2: Solve the model with Gaussian Elimination and store returned transmission weight from the Gaussian Elimination in TWeightList
3: Divide TWeightList into low, medium and high transmission weight bin range
4: N = SampleSize * Total number of items
5: **For** each bin range do the following
6: Randomly select N/3 items and store in LandMarkList
7: **End For**

Description:

Each item was assigned a very small seed value in order to acquire interaction weights between items without effecting the domain weight when passing through the weight transmitter model. Subsequently, bin ranges (for low, medium and high) weight bins) are set based on the sorted transmission weight of all items returned from weight transmitter model. Then we determine the number of landmark items as N, based on the landmark sampling percentage. Finally, N/3 Landmark items are then equally assigned to each bin range.

This algorithm can be applied to both global and local approaches to weight fitting. For the local approach, the total number of landmark items for each sub-graph is based on the proportion of size of sub-graphs to the total number of items in the global graph. The same method is then applied to each sub-graph in order to assign landmark items for each sub-graph.

4.6 Rule Generation and Extraction

Once the overall weights for all items are determined, the rule base will then be generated. Our rule base is generated by inputting the top p% of items ranked by overall weight produced by the weight transmitter model to a standard rule generator, such as Apriori.

We are most interested in rules of the form $X \rightarrow Y$ where X is an item that is low weighted on the basis of domain knowledge whereas Y is rated high on domain knowledge. This form of pattern signifies that items in set X should be weighted much more heavily than is suggested on the basis of domain knowledge alone. This is due to the fact that items such as Y can be observed with a high degree

of confidence whenever items such as X occur, suggesting that the value of X ascertained on the basis of domain knowledge alone underestimated its true value.

4.7 Hyperclique Pattern

In this research, we are also interested in finding a clique among all rules that we extracted. In order to find a clique the Hyperclique pattern algorithm is applied on all the extracted rules mentioned in the previous section.

Hyperclique pattern is a framework that is used to mine highly-correlated association patterns with the use of an objective measure called h-confidence to explore hyper clique patterns. Xiong et al. (2003) explained that hyperclique is originally derived from the idea of hypergraph. A hypergraph H = {V, E} consists of a set of vertices {V} and a set of hyperedges {E}. Every itemset is treated as a hypergraph in which each item is represented as a vertex and every vertex has a hyperedge to all other vertices (Xiong et al., 2003).

Xiong et al. (2006) defined a metric called h-confidence as a measure of association for an itemset P = $\{i_1, i_2, \ldots, i_m\}$ as follows

$$hconf(P) = min\ \{\ conf\ \{i_1 \rightarrow i_2, \ldots, i_m\},\ conf\ \{i_2 \rightarrow i_1, i_3, \ldots, i_m\},\ \ldots,\ conf\ \{i_m \rightarrow i_1, \ldots, i_{m-1}\}\ \}$$

The term *conf* follows the definition of traditional association rule confidence (Agrawal et al., 1993).

Consider this example, let *itemset P = {X, Y, Z}*. Assume that *supp({X})* = 0.1, *supp({Y})* = 0.1, *supp({Z})* = 0.5 and *supp({X, Y, Z})* = 0.05, where supp denotes the support of an itemset as defined in Agrawal et al. (1993). Since

$$Conf\{X \rightarrow Y, Z\} = supp(\{X, Y, Z\}) / supp(\{X\}) = 0.5,$$
$$Conf\{Y \rightarrow X, Z\} = supp(\{X, Y, Z\}) / supp(\{Y\}) = 0.5,$$
$$Conf\{Z \rightarrow X, Y\} = supp(\{X, Y, Z\}) / supp(\{Z\}) = 1,$$

Therefore, *hconf(P)* = min{0.5, 0.5, 1} = 0.5.

Given a set of items $I = \{, i_2, \ldots, i_n\}$ and minimum h-confidence threshold h_c, an itemset $P \subseteq I$ is a hyperclique pattern if $hconf(P) \geq h_c$.

With the above statement, Xiong et al. (2006) explained that the occurrence of item $i \in P$ in a transaction indicates the occurrence of all other items $P - \{i\}$ in the same transaction with a minimum probability of h_c. This definition captures the strength of inter-relationships between groups of items.

4.8 Performance Metric

In order to examine how the proposed model performs, a set of performance metrics needs to be defined. The weight transmitter model will be evaluated against major standard performance metrics which will be described in the following sub-sections.

4.8.1 Accuracy

In our research, we tracked all Accuracy metrics (Precision, Recall, Percentage of Accuracy) on high weight items returned by our Weight Transmitter model.

Precision refers to the accuracy proportion of high weight items returned from the model. It can be illustrated as:

$$Precision = \frac{Actual\ high\ weight\ items\ \cap Returned\ high\ weight\ items\ from\ the\ model}{Returned\ high\ weight\ items\ from\ the\ model}$$

Recall refers to the proportion of high weight items returned from the model against the actual high weight items. It can be illustrated as:

$$Recall = \frac{Actual\ high\ weight\ items\ \cap Returned\ high\ weight\ items\ from\ the\ model}{Actual\ high\ weight\ items}$$

Accuracy refers to the proportion of high weight items returned from the model against the total number of items. It can be illustrated as:

$$Accuracy = \frac{Actual\ high\ weight\ items\ \cap Returned\ high\ weight\ items\ from\ the\ model}{Total\ number\ of\ items}$$

4.8.2 Effectiveness

Lift is a measure commonly used in targeted marketing to assess the boost in performance gained over a simplistic mass marketing strategy (which essentially generates sales outcomes at random). In the context of weight estimation, the Lift demonstrates the effectiveness of the Weight Propagation model over a naive approach of labeling items as high, medium and low at random. Lift is defined by Equation 4.3

$$Lift = \frac{L_{11} \cdot N}{(L_{11} + L_{12}) \cdot (L_{11} + L_{21})}$$

Equation 4.3

where L_{11} is the set of true positive which represents the number of items that are returned as high weight from the model by supplying the landmark set of weights and still remain high when supplying with the complete set of weights; L_{12} is the set of false negatives which represents the number of items that are returned as high weight from the model only when supplying with the complete set of weights but not the landmark set of weights; L_{21} is the set of false positive which represents the number of items that are returned as high weight from the model only when supplying with the landmark set of weight but not the complete set of weight, while N is the total number of items. The higher the Lift value is than 1 the higher the effectiveness of the Weight Transmitter model.

Note that the Lift measure defined above is distinct from Rule Lift which measures the confidence of a rule relative to the support of the consequent term of that rule, as defined by Agrawal et al., (1993).

4.8.3 Computational performance

Execution time refers to the amount of time for estimating all item weights from a set of landmark items. It represents the elapsed time in solving system of linear equations with the Gaussian Elimination method. Execution time is measured in milliseconds.

4.8.4 Profit Analysis

We are also interested in measuring profit generated from items that interacted strongly with items that are known to have high weight. In order to measure profit, we track the set of items (H') where $H' = \{i | i \in I$ where i is in the top p percentile on the basis of overall weight but not on the basis of domain weight $\}$. For all items belonging to H' we defined a profit measure (P) that takes into account the amount of indirect profit that such items generated. The profit measure for a given item $i \in H'$ is computed by taking the total profit (P_1) over all the transactions (T_1) in which item i occurs and then subtracting the total profit (P_2) over all transactions (T_2) in which item i does not occur.

In order to isolate the confounding effects of transactions in T_2 having more items than T_1 we restrict each of the transactions involved in T_2 having only the neighbors of the item i under consideration. In addition, we also compensated for the differences in the size of T_1 and T_2 thus profit of an item is calculated by

$$P(i) = \frac{|T_1|}{|T_2|} \cdot \sum_{i \in t_1} \sum_{t_1 \in T_1} w(i) - \sum_{i \in t_2} \sum_{t_2 \in T_2} w(i), for\ all\ t_1, t_2\ \in T$$

Equation 4.4

where $w(k)$ represents the weight of item k that is connected to item i where T is the set of all transactions in the transaction database. However, the profit measure P by itself has little meaning unless it is compared with the profit generated by the set of items *NH* that remain low in weight without making the transition to high weight category in terms of overall weight. We expect that, the

P value of items in the set H' will be substantially higher than the profit P value of items in the set *NH*.

In addition, since some items in a transaction dataset can co-occur together by chance, we apply the algorithm adapted from Fisher's exact test explained in the study of Koh and Pears in order to eliminate items that are neighbors of items in the set H' and NH that occur together by chance with items in the set H' and NH thus enabling the true profit generated to be captured.

Their algorithm examines the probability of seeing two items occurring together. For N transactions in which item A occurs in a transactions and item B occurs in b transactions, the probability that A and B will occur together exactly c time is:

$$Pcc(c|N,a,b) = \frac{\binom{a}{c}\binom{N-a}{b-c}}{\binom{N}{b}}$$

Equation 4.5

The chance threshold is calculated independently for each pair of items. Given Pcc in Equation 4.5, we calculate the least value of chance collision s which Pcc is larger than a threshold value $p = 0.9999$.

$$Chance(N,a,b,p) = min\left\{ s| \sum_{i=0}^{i=s} Pcc(i|N,a,b) \geq p \right\}$$

Equation 4.6

If the value of chance collision s is greater than the value of the actual co-occurrence time c between item A and B, then item A and B occur together by chance.

4.9 Summary

In this chapter, we have discussed and outlined the research paradigm and methods that were employed in order to achieve the objectives of this research. We have also presented our proposed architecture for this research together

with the details of algorithms that were applied in our proposed methods. The next chapter will provide the details of the experimental design for this research.

Chapter 5 Experimental Design

5.1 Introduction

This chapter will focus on describing the experiments designed to evaluate the performance of our proposed weight transmitter model in terms of different performance indicators presented in Chapter 4. We also describe further experiments relating to rule generation and extraction. We start with a brief description of the datasets experimented with, together with relevant summary statistics.

5.2 Datasets used for experimentation

We experimented with a total of six real-world datasets. The following are the descriptions:

- **Retail dataset:** This is an industry benchmark retail market transactional dataset, which is supplied by a Belgian supermarket store. The store provided a separate file that contains the list of items and their unit price value, which we used as a substitute for unit profit, on account of the unavailability of the latter measure. There are 5599 transactions in the dataset covering a total of 1320 items.

- **Nasa web log datasets:** We also used two different web logs, which were supplied by the Nasa Kennedy Space Center WWW in Florida USA. The first dataset was collected over the month of July 1995, which we call the "Nasa" dataset. The second dataset was collected over the month of August 1995, which we call "NasaAug". We preprocessed these datasets by considering pages as items and a sequence of clicks on a set of web pages that occurred through a session as transactions. For the purposes of recording dwelling time on a page we set a session timeout to 15 minutes. Thus, for a given page, only activity that takes place within contiguous 15 minute time windows is taken into consideration when evaluating the average page dwelling time. We took

average dwelling time on a web page as a proxy for each item weight, in line with previous research (Cooley et al., 1997; Srivastava et al., 2000; Yan and Li, 2006). After preprocessing, there were 1705 transactions in Nasa, covering 844 items while 2028 transactions resulted from NasaAug, yielding 805 items.

- **Computer Science Lab datasets:** We also used three different computer lab web log request files, which were supplied by the University of Auckland. The first dataset was collected over the period of December 2007-February 2008, which we call the "Access" dataset. The second was collected over a period of February 2008-December 2008, which we denote as the "Access2" dataset and the third dataset called Uaccess. We applied the same data pre-processing techniques and set the same maximum time for each session as the Nasa and NasaAug dataset. Also, we applied the same proxy for item weight. After preprocessing, there were 5414 transactions were obtained for Access, consisting of 990 items, 5607 transactions resulted from Access2, covering 2315 items and 4843 transactions were obtained for Uaccess, consisting of 538 items.

Table 5.1 summarizes the details mentioned above.

Dataset	Type	No. of Items	No. of transaction
Access	Web log	990	5414
Access2	Web log	2315	5607
Nasa	Web log	844	1705
NasaAug	Web log	805	2028
Retail	Retail transaction	1320	5599
Uaccess	Web log	538	4843

Table 5.1: Summary of datasets details

5.3 Tools

This section highlights the programming tools that were used to develop the weight transmitter model. C++ was used to develop the model while Python was used to generate scripts for performance evaluation and rule extraction.

5.4 Experimental Plan and Execution

The experiments were conducted on a Windows XP laptop with an Intel® Core™ 2Duo CPU @ 2.10 GHz. Processer, 3 GB of Ram memory and 250 GB of hard disk space. The weight transmitter was implemented in the C++ programming language. Also, various scripts in evaluation processes were developed in Python programming language. We divided our experimentation into three separate parts which are described in detail in the following sub-sections.

5.4.1 Experiment 1: Global Approach to Item Weight Estimation

This experiment was designed to assess performance of the global approach to weight propagation. In this experiment, we investigate two main issues: the effect of replacing the Gini Index measure with our proposed Proportional Confidence measure; and the effect of replacing simple random sampling with stratified sampling.

We assessed the sensitivity of the key sampling percentage parameter on performance by varying it in a small range from 10% to 30%. At each of the sampling levels, 30 different runs were applied to select different sets of landmark items. The performance measures presented represent an average of the measure taken across the 30 different runs.

We also contrasted our approach with the Label Propagation approach, thus resulting in a three-way comparison between it, our proposed model and Koh's model. The label propagation approach, as described in Chapter 3 while being broadly similar to the other two weight propagation approaches differs in the

methods used to measure interactions between items and the weight inference strategy. As a result, it has specific parameters such as α, and convergence parameters. The parameter α is used to calculate parameter μ, which is the factor that represents weight of initial label. It ranged from (0, 1). Convergence parameter is used to determine the stopping condition for the iteration of the algorithm. We set two different values for parameter α. One is close to 0 while another close to 1. The reason behind this is to observe the effect of the parameter α on the algorithm. Also, two different values of convergence parameters (0.01 and 0.001) were set for the same reason which is to observe the effect of the convergence parameter on the algorithm. Then we select the set of parameters setting that produces the best results among them to use as a benchmark performance.

The following steps were executed for the benchmark Label Propagation approach:

1. Select a set of landmark items at each percentage sampling level with the use of simple random sampling.
2. Execute the Label Propagation algorithm with α set to 0.04, ∈ set to 0.0001 and convergence set to 0.01.
3. Collect all returned items weights.
4. Execute python script to measure the performance.
5. Repeat steps 1) to 4) for 30 runs.
6. Repeat steps 1) and 5) with α set to 0.04, ∈ set to 0.0001 and convergence set to 0.001.
7. Repeat steps 1) and 5) with α set to 0.95, ∈ set to 0.0001 and convergence set to 0.01.
8. Record the results of the set of the parameters that produced the best performance to use as benchmarking performance.

The following steps were executed for the proposed model (Weight Transmitter).

// simple random sampling

1. Select a set of landmark items at each percentage sampling from 10% to 30% with the use of simple random sampling.

2. Execute the Weight Transmitter algorithm.
3. Collect all returned items weights
4. Execute python script to measure the performance.
5. Repeat steps 1) to 4) for 30 runs.

// stratified random sampling

1. Select a set of landmark items at each percentage sampling level from 10% to 30% using stratified random sampling.
2. Execute the Weight Transmitter algorithm.
3. Collect all returned items weights
4. Execute python script to measure the performance.
5. Repeat steps 1) to 4) for 30 runs.

5.4.2 Experiment 2: Item Weight Estimation in a Local Viewpoint

This experiment was designed to compare the performance achieved by the proposed local approach with that of the best performer obtained from Experiment 1. In this experiment, we assigned landmark items to each of the localized partitions obtained from the partitioning process. We follow the same basic experimental procedure as described for Experiment 1. These steps are:

1. Select a set of landmark items at each percentage sampling level from 10% to 30% with the use of stratified random sampling for each of the partitions.
2. Execute with Weight Transmitter algorithm.
3. Collect all returned item weights.
4. Execute python script to measure the performance.
5. Repeat steps 1) to 4) for 30 runs.

In this experiment, we were also interested in tracking the effect of the weighting scheme on items that interacted strongly with items that were known to have high weight on a basis of overall weight but not on the basis of domain weight. This will enable us to test our research premise that items whose weights have transited from the low weight to high weight category would

generate higher indirect profit than items which do not make such a transition. The following are the steps executed:

1. Select items in top P percentile = 40% on the basis of overall weight for each percentage sampling level.
2. Execute Profit Analysis algorithm.
3. Collect all returned profit generated.

Items that transit from the low weight to high weight category do so on the basis of their interactions with high weight items. In order to eliminate the effect of chance interactions distorting results we utilized Fisher's exact test. The following steps were executed:

1. Select items in top P percentile = 40% on the basis of overall weights for each sampling level.
2. Execute the Profit Analysis algorithm and apply Fisher's exact test for chance collision threshold algorithm with a threshold value of 0.9999.
3. Collect all returned profit generated.

5.4.3 Experiment 3: Rule Generation and Extraction

This experiment was designed to track the rule base that contains rules of the form X → Y where X represents a low weight item and Y a high weight item. As explained in Chapter 3, X represents items that were rated lowly on the basis of domain weight but acquired a high (overall) weight on the basis of interactions with items such as Y that were rated highly on the basis of domain weight. This experiment will enable us to identify the set X of items appearing in rule antecedents that should be weighted much higher than is set on the basis of domain knowledge alone.

The following steps were executed for this experiment.

1. Input items in top P percentile on the basis of overall weights to a standard rule generator with the parameter settings shown in Table 5.2 where *minSup* denotes the minimum support threshold, *minConf* denotes

the minimum confidence threshold and *minLift* denotes the minimum Lift threshold. The term *support*, *confidence and Lift* follow the definition of traditional association rule (Agrawal et al., 1993).

Dataset	P%	minSup	minConf	minLift
Access	40	0.03	0.7	1.0
Access2	40	0.4	0.7	1.0
Nasa	40	0.1	0.7	1.0
NasaAug	40	0.015	0.7	1.0
Retail	40	0.01	0.7	1.0
Uaccess	40	0.4	0.7	1.0

Table 5.2: Rule generation parameters

2. Collect all rules generated
3. Execute script to extract all rules of the form X → Y as described above.
4. Rank the extracted rules based on average profit of items appearing in the rule extracted.

5.5 Summary

In this chapter, we have explained in detail the reasoning and the structure of the experiments that will be used to assess the performance of the proposed methods together with a description of the datasets that were used. The next chapter will present the empirical findings of this research.

Chapter 6 Empirical Study

6.1 Introduction

In the previous chapter, we presented the experimental design and the experimental configuration that was used in the study. In this chapter, we will focus on presenting the empirical results and then discuss the insights gained from the study.

6.2 Experiment 1: Performance of New Weight Transmitter Model

This experiment was designed to compare the performance achieved by the proposed Weight Transmitter model against that of the original Weight Transmitter model by Koh et al. This comparison focuses on two aspects; the effect of replacing the Gini index measure with our novel Proportion Confidence measure, and secondly, the effect of stratified random sampling in assigning landmark items. In addition, the Weight Transmitter model was also compared against the well established Label Propagation method which was used as the baseline method.

We varied the percentage of landmark items in a small range from 10% to 30% and tracked four different performance metrics achieved by each of the three different methods. At each of the sampling levels, 30 different trials were executed in view of the nature of the random process for selecting landmark items. The average value for each of the metrics across the 30 runs was then computed.

In order to set a fair level of benchmarking performance for the Label Propagation method, several trials were executed with different combinations of values for its parameters and we used the best combination which yielded: α=0.95 and convergence = 0.01 (These trials are presented in Appendix A).

The results for each of the performance metrics is presented in the following sub-sections.

6.2.1 Precision

Dataset	Method	10%	20%	30%
Access	Label Propagation	18.92	23.61	30.85
	Original WT	75.71	84.32	88.58
	Proposed WT- Simple random	91.32	95.38	97.13
	Proposed WT- Stratified random	95.68	97.21	97.92
Access2	Label Propagation	13.94	20.50	30.43
	Original WT	63.52	72.84	78.03
	Proposed WT- Simple random	61.15	79.34	81.98
	Proposed WT- Stratified random	83.36	83.62	83.52
Nasa	Label Propagation	42.77	56.90	66.50
	Original WT	61.67	75.82	80.22
	Proposed WT- Simple random	63.19	69.61	75.82
	Proposed WT- Stratified random	73.82	82.65	84.83
NasaAug	Label Propagation	32.31	47.27	57.34
	Original WT	72.73	81.08	86.66
	Proposed WT- Simple random	66.97	80.76	86.35
	Proposed WT- Stratified random	89.65	92.90	96.10
Retail	Label Propagation	20.64	31.14	41.07
	Original WT	39.53	48.96	54.19
	Proposed WT- Simple random	49.20	61.19	68.51
	Proposed WT- Stratified random	61.12	73.12	78.71
Uaccess	Label Propagation	18.57	28.81	39.05
	Original WT	57.44	61.31	68.81
	Proposed WT- Simple random	60.83	67.26	84.29
	Proposed WT- Stratified random	74.86	81.61	84.46

Table 6.1: Precision Analysis

Several trends are evident from Table 6.1, the first of which is that the Label Propagation algorithm returns the lowest precision across all datasets experimented with. Secondly, the original WT model had a reasonable level of precision across all datasets. Its performance improved progressively with higher levels of landmark sampling, consistent with the results reported in [Koh et al., 2012]. Thirdly we observe substantial levels of improvement in precision with the introduction of each of the two refinements that we proposed. The introduction of the proportional confidence measure generally resulted in a substantial gain in precision and a further improvement can be observed with the subsequent injection of stratified sampling. Out of the two model refinements, it is evident, that on average, stratified sampling had a bigger effect in lifting Precision than the use of proportional confidence.

Table 6.1 also shows that the new weight transmitter incorporating both proportional confidence and stratified sampling performed very well at the 20% landmark sampling level, with every dataset returning over 80% levels of precision, with the exception of the Retail dataset.

6.2.2 Recall

Dataset	Method	10%	20%	30%
Access	Label Propagation	12.09	14.77	20.14
	Original WT	49.62	61.18	67.58
	Proposed WT- Simple random	70.97	77.21	81.69
	Proposed WT- Stratified random	71.73	71.21	71.87
Access2	Label Propagation	10.67	15.34	22.15
	Original WT	36.09	47.66	54.35
	Proposed WT- Simple random	54.72	63.66	67.61
	Proposed WT- Stratified random	65.99	67.43	67.63
Nasa	Label Propagation	17.74	23.95	30.30
	Original WT	41.28	54.16	60.69
	Proposed WT- Simple random	42.64	49.98	57.38
	Proposed WT- Stratified random	49.06	59.25	63.43
NasaAug	Label Propagation	14.17	21.90	27.75
	Original WT	56.76	67.33	74.39
	Proposed WT- Simple random	45.97	59.96	68.11
	Proposed WT- Stratified random	51.61	56.60	59.04
Retail	Label Propagation	11.13	16.67	22.20
	Original WT	23.78	31.07	34.45
	Proposed WT- Simple random	39.90	51.00	58.43
	Proposed WT- Stratified random	49.20	60.40	67.15
Uaccess	Label Propagation	8.12	13.75	19.99
	Original WT	35.24	40.54	45.22
	Proposed WT- Simple random	39.75	45.13	61.63
	Proposed WT- Stratified random	34.47	47.30	58.35

Table 6.2: Recall Analysis

Table 6.2 shows that the same trends as observed with Precision are exhibited with the use of the Recall measure. Label propagation once again was the worst performer followed by original WT and the modified WT models. With the exception of the NasaAug dataset, both versions of the extended WT model outperformed the original model. As with the Precision experiments, stratified sampling was the major factor in lifting the Recall rate.

Overall, the Recall rates achieved were less than the Precision rates at each of the landmark sampling levels that were employed. This is due to the fact that

achieving a high Recall rate with a weight transmission approach requires the landmarks to cover a high proportion of high weight items in any given neighborhood, which cannot be guaranteed at the modest sampling levels that must necessarily be employed for the estimation method to be useful in practice.

Nevertheless, we observe that reasonable Recall rates of 60% or more were achieved with one or more variants of the weight transmitter models at the 30% sampling level. Furthermore, as our results in section 6.3.2 show, it is possible to improve the Recall rate with the use of the localized modeling approach.

6.2.3 Percentage Accuracy

Dataset	Method	10%	20%	30%
Access	Label Propagation	77.32	78.18	80.38
	Original WT	88.97	92.79	94.62
	Proposed WT- Simple random	97.05	98.26	99.00
	Proposed WT- Stratified random	95.64	95.74	95.94
Access2	Label Propagation	79.60	80.60	82.22
	Original WT	84.72	89.00	90.98
	Proposed WT- Simple random	91.07	95.09	95.85
	Proposed WT- Stratified random	94.00	94.31	94.35
Nasa	Label Propagation	73.83	77.26	80.97
	Original WT	86.68	90.79	92.61
	Proposed WT- Simple random	87.58	89.89	91.95
	Proposed WT- Stratified random	89.59	92.52	93.57
NasaAug	Label Propagation	73.26	77.50	80.31
	Original WT	90.96	94.11	95.69
	Proposed WT- Simple random	88.64	92.64	94.63
	Proposed WT- Stratified random	90.27	91.97	92.85
Retail	Label Propagation	75.36	77.33	79.54
	Original WT	80.21	83.29	84.26
	Proposed WT- Simple random	87.46	90.25	92.00
	Proposed WT- Stratified random	89.78	92.55	94.08
Uaccess	Label Propagation	69.28	73.06	77.58
	Original WT	85.01	86.61	87.86
	Proposed WT- Simple random	85.10	87.35	92.49
	Proposed WT- Stratified random	82.01	86.53	90.36

Table 6.3: Accuracy Analysis

Percentage accuracy, unlike Precision and Recall which are aimed exclusively at the high weight items, measures the overall level of correctness achieved in

identifying the three categories of items which happen to be low, medium and high.

Table 6.3 shows that the percentage accuracy produced by the Label Propagation method is the lowest compared to the other three methods. However, the gap in difference with other methods is very much smaller than with Precision or Recall.

In terms of the comparison between original WT and proposed WT with simple random sampling, it again followed the trends from the previous two experiments, whereby extended WT outperformed the original WT on all datasets, except for Nasa and NasaAug.

6.2.4 Lift

Dataset	Method	10%	20%	30%
Access	Label Propagation	1.8729	2.3374	3.0543
	Original WT	7.4952	8.3480	8.7691
	Proposed WT- Simple random	9.0407	9.4426	9.6157
	Proposed WT- Stratified random	9.4720	9.6234	9.6942
Access2	Label Propagation	1.3849	2.0367	3.0231
	Original WT	6.3115	7.2370	7.7532
	Proposed WT- Simple random	6.0752	7.8833	8.1450
	Proposed WT- Stratified random	8.2820	8.3087	8.2985
Nasa	Label Propagation	4.1973	5.5845	6.5260
	Original WT	6.0522	7.4413	7.8728
	Proposed WT- Simple random	6.2018	6.8317	7.4413
	Proposed WT- Stratified random	7.2448	8.1112	8.3252
NasaAug	Label Propagation	3.1718	4.6401	5.6290
	Original WT	7.1398	7.9594	8.5071
	Proposed WT- Simple random	6.5745	7.9286	8.4766
	Proposed WT- Stratified random	8.8010	9.1205	9.4346
Retail	Label Propagation	2.0487	3.0910	4.0761
	Original WT	3.9229	4.8587	5.3784
	Proposed WT- Simple random	4.8834	6.0734	6.7992
	Proposed WT- Stratified random	6.0660	7.2566	7.8115
Uaccess	Label Propagation	1.8166	2.8181	3.8196
	Original WT	5.6186	5.9972	6.7308
	Proposed WT- Simple random	5.9506	6.5794	8.2447
	Proposed WT- Stratified random	7.3230	7.9827	8.2621

Table 6.4: Lift Analysis

As mentioned in Chapter 3, the Lift measure is used to track the gain in performance in classifying or categorizing items when compared to a decision strategy that is purely random in nature. Given the good performance of the weight transmitter models on Accuracy it comes as no surprise that Label Propagation is once again the worst performer, as shown by Table 6.4.

In terms of the weight transmitter models, the extended version was the clear winner outperforming the original WT on all datasets, with stratified sampling being the major cause for the superior performance.

6.3 Experiment 2: Performance of the Local Approach

This experiment was designed to assess the local approach to weight estimation. The local approach, to the best of our knowledge has never been explored in the context of weight estimation for pattern mining and thus it would be of interest to ascertain where it stands with respect to the global approach. With this in mind, we compared the performance of the local approach with the best performer from Experiment 1, [which happened to be] weight transmitter enhanced with proportional confidence and stratified sampling.

The experimentation followed the same basic procedure described in Experiment 1. The percentage of landmark items was varied from 10% to 30% in steps of 10 and 30 different runs were executed at each of the sampling levels.

The results do not include the Access dataset as there were no natural partitions in this dataset, arising from the fact that each item was strongly connected to its neighbors. Any partition imposed would thus be artificial and would result in poor performance as neighborhoods would be split across multiple sub graphs, thus severely limiting the ability of weight transmitter to estimate items weights with any reasonable level of accuracy.

Each of the remaining datasets was partitioned into multiple sub-graphs with the number of items in each sub-graph varying from 30 to 368 items. The

performance values quoted for each of the four metrics for any given dataset were the average of that metric when taken across all the sub-graphs for that dataset.

6.3.1 Precision

Dataset	Method	10%	20%	30%
Access2	Global	83.36	83.62	83.52
	Sub-graph	92.94	93.40	95.20
Nasa	Global	73.82	82.65	84.83
	Sub-graph	69.71	80.93	86.28
NasaAug	Global	89.65	92.90	96.10
	Sub-graph	78.25	86.88	92.65
Retail	Global	61.12	73.12	78.71
	Sub-graph	58.28	68.38	74.88
Uaccess	Global	74.86	81.61	84.46
	Sub-graph	39.11	60.60	92.20

Table 6.5: Precision Analysis

Table 6.5 shows that the localized approach had mixed results with respect to Precision. At the 30% sampling level the localized approach outperformed the global approach in 3 out of the 5 datasets, while the global approach fared better in the remaining two which happened to be the NasaAug and Retail datasets.

6.3.2 Recall

Dataset	Method	10%	20%	30%
Access2	Global	65.99	67.43	67.63
	Sub-graph	83.30	83.84	87.19
Nasa	Global	49.06	59.25	63.43
	Sub-graph	47.38	58.84	64.87
NasaAug	Global	51.61	56.60	59.04
	Sub-graph	53.87	64.93	73.57
Retail	Global	49.20	60.40	67.15
	Sub-graph	47.01	56.38	63.10
Uaccess	Global	34.47	37.30	38.35
	Sub-graph	25.42	46.25	72.03

Table 6.6: Recall Analysis

In terms of Recall, it can be observed from Table 6.6 that the localized approach outperformed the global approach at all sampling levels with the

exception of the Retail dataset. The gains in Recall with the localized approach are around 20%, 14% and 13.6% respectively for the Access 2, NasaAug and Uaccess datasets which can be considered to be significant. These gains were responsible for lifting the Recall rate to well over the 70% mark for these datasets at the 30% sampling level.

The identification of high weight items via any form of the weight transmitter model is strongly dependent on the presence of high weight items amongst the landmark items as these items guide the weight fitting process. With the use of stratified random sampling, the expected percentage of high weight items present as landmarks is just 10% of the total number of high weight items taken across the dataset. Thus the performance of the localized approach which returns a minimum Recall rate of 63% (registered for the Retail dataset) and rates of over 70% for 3 other datasets can be considered to be very satisfactory.

6.3.3 F-Measure

Dataset	Method	10%	20%	30%
Access2	Global	73.66	74.66	74.74
	Sub-graph	87.86	88.36	91.02
Nasa	Global	58.95	69.02	72.59
	Sub-graph	56.42	68.14	74.06
NasaAug	Global	65.51	70.34	73.14
	Sub-graph	63.81	74.32	82.01
Retail	Global	54.52	66.15	72.47
	Sub-graph	52.04	61.80	68.49
Uaccess	Global	47.20	51.20	52.75
	Sub-graph	30.81	52.46	80.88

Table 6.7: F-Measure

The F-Measure was utilized to balance both Precision and Recall by evenly weighting them against each other. It can be seen from Table 6.7 that the localized approach outperformed the global approach at all sampling levels in the Access 2 dataset while outperforming the global approach at the 30% sampling level in the other datasets with the exception of the Retail dataset. The gains in F-Measure with the localized approach are around 17%, 9% and 11.8% respectively for the Access 2, NasaAug and Uaccess datasets which can be

considered to be significant. With these gains, the F-Measure was lifted to above the 80% mark for NasaAug and Uaccess datasets while reaching the 90% mark for the Access 2 dataset at the 30% sampling level.

6.3.4 Percentage Accuracy

Dataset	Method	10%	20%	30%
Access2	Global	94.00	94.31	94.35
	Sub-graph	97.47	97.59	98.18
Nasa	Global	89.59	92.52	93.57
	Sub-graph	89.11	92.36	93.98
NasaAug	Global	90.27	91.97	92.85
	Sub-graph	91.01	93.96	96.03
Retail	Global	89.78	92.55	94.08
	Sub-graph	89.23	91.56	93.15
Uaccess	Global	82.01	83.53	84.36
	Sub-graph	81.16	88.43	95.72

Table 6.8: Accuracy Analysis

In terms of percentage accuracy, it can be seen from Table 6.8 that the localized approach has better accuracy than the global approach on all datasets, except for Retail, where it did marginally worse than the global approach. The accuracy returned in all cases was well above the 90% mark at the 30% sampling level, thus indicating the robustness of the localized approach as a weight fitting mechanism.

6.3.5 Lift

Dataset	Method	10%	20%	30%
Access2	Global	8.2820	8.3087	8.2985
	Sub-graph	9.2339	9.2803	9.4588
Nasa	Global	7.2448	8.1112	8.3252
	Sub-graph	6.8411	7.9423	8.4678
NasaAug	Global	8.8010	9.1205	9.4346
	Sub-graph	7.6820	8.5290	9.0956
Retail	Global	6.0660	7.2566	7.8115
	Sub-graph	5.7838	6.7869	7.4313
Uaccess	Global	7.3230	7.9827	8.2621
	Sub-graph	3.8254	5.9273	9.0191

Table 6.9: Lift Analysis

In terms of Lift, it can be observed from Table 6.9 that the localized approach outperformed the global approach in 3 of the 5 datasets, while doing marginally worse for the NasaAug and Retail datasets.

6.3.6 Statistical Significant t-Test

We also perform a t-Test in order to measure the significance of performance between the localized approach and the global approach in terms of F-Measure, Percentage Accuracy and Lift at 30% sampling level with a p-value set to 0.05.

The result showed that the localized approach significantly outperformed the global approach in Access 2, NasaAug and Uaccess dataset, while the global approach performed better in Retail dataset. In addition, there was no significant difference in performance for the Nasa dataset. From this result, we can conclude that the localized approach never did worse and was better than the global approach on the majority of datasets, with Retail being the exception. (The t-Test results are presented in Appendix B).

6.3.7 Execution Time

In addition to the gains in accuracy, the localized approach has the capability to drastically reduce the execution time of the weight fitting process. This is due to the fact that the Gaussian elimination procedure used in the weight fitting process has a worst case time complexity of $O(N^3)$ where N is the total number of items. The local approach that employs a divide and conquer strategy has a much smaller worst case time complexity of $O(N^2) + \sum_{i=1}^{m} O(S_i^3)$ where m is the number of sub graphs obtained by the partitioning process and S_i is the size of sub graph i. This is due to the fact that the worst case time complexity of the partitioning process is $O(N^2)$ and the second term can be shown to have a worst case time complexity less than $O(N^3) - O(N^2)$ (see Appendix C).

The results in Table 6.10 are consistent with this run time analysis as the execution time of the localized approach is very much smaller than the global approach. The execution time for the sub-graph approach is broken down into

four components: pre-processing that is required to establish a threshold value above which connections weights are considered to be significant, the pre-processing of the matrix into binary form that captures whether a pair of items are connected together significantly or not, the separation of the global graphs into sub-graphs, and finally the weight fitting process.

The gap in execution time between the global and localized approaches widens when the number of items to be fitted is larger, as is the case with Retail dataset where the drop in execution time for the localized approach exceeds one order of magnitude. This result reinforces the utility of the localized approach over the global approach.

Dataset	Global	Sub-Graph			
		Calculating Threshold Value	Pre-Process Matrix	Partitioning	Weight fitting
Access	5,484	-	-	-	-
Access2	70,421	641	109	2,844	751
Nasa	2,610	16	15	350	78
NasaAug	2,172	16	16	312	79
Retail	11,031	16	31	1,016	126
Uaccess	532	16	2	92	93

Table 6.10: Model Execution Time (millisecond)

6.3.8 Profit Analysis

This section presents the result in measuring profit generated from items that interacted strongly with items that are known to have high weight. Profit analysis was calculated based on the winner approach which was the localized approach. As explained in Chapter 4 we use the term “profit” in a generic sense and not in the narrow sense of monetary gain obtained by selling an item at a higher price than what it cost to purchase or produce the item. Thus in a web click stream environment profit reflects the importance of individual pages, measured by the average dwelling time that a user spends on that page.

As discussed in Chapter 4, we were interested in tracking the profit generated by items in the set H' where $H' = \{i | i \in I$ where i is in the top p percentile on

the basis of overall weight but not on the basis of domain weight }. The profit in set H' was compared with the profit generated by items in the set *NH* that contains items that remain low in weight without making the transition to high weight category in terms of overall weight.

Table 6.11 shows that profit generated by the items in set H' is substantially higher than the profit generated by the items in set *NH* across all datasets, except for the Retail dataset. The implication of this result is that weight transmitter was able to ascertain the true weight of certain items by assigning higher weights to items that interacted strongly with high profit items, rather than simply relying on individual item profit ascribed purely on the basis of domain knowledge. Without the benefit of the weight fitting process, items in set H would not have survived the weight thresholding phase prior to rule generation, thus inhibiting rules containing these items from being generated. Such rules of the form: X→Y embodies potentially significant knowledge as they show that items X that have low domain weights are strongly associated with items Y that have high weight (profit). If it can be determined that items X and Y do not occur together by chance through rigorous statistical analysis then such rules are indeed valuable as it shows that low profit items X can be promoted at a discount rate when the customer makes a commitment to items X and Y which are offered as a single package as part of a promotion campaign.

Of particular interest is the big rate of transition of low weight category (on the basis of domain weight) to the high weight category (on the basis of overall weight) across all datasets, ranging from 21% to 47%.

There are two possible reasons why the Retail dataset that we used was the exception in virtually all experiments that we conducted. The first and foremost reason is the proxy that we used for item weight. Unfortunately true profit was unavailable and we had to rely on unit selling price as an alternative to profit. Secondly, the version of the dataset that we used was very sparse indeed, with an average transaction size of only 2, while the average items per transaction of others datasets are greater than 10.

Dataset	10% sampling			20% sampling			30% sampling		
	% change	H'	NH	% change	H'	NH	% change	H'	NH
Access	39	86513.81	55149.44	37	95357.44	50702.46	38	91820.20	52252.62
Access2	47	1769.99	1221.00	46	1809.56	1199.49	46	1825.01	1187.15
Nasa	40	2229.21	899.29	41	2142.54	931.18	42	2016.34	1017.39
NasaAug	23	4292.68	905.83	21	4375.55	977.51	21	4401.35	958.20
Retail	43	78.86	93.45	42	78.91	92.83	42	81.98	88.90
Uaccess	42	6618.59	2597.40	41	6987.31	2526.28	41	6730.01	2727.78

Table 6.11: Profit Analysis

In addition, since some items in a transaction dataset can co-occur by chance, we applied an algorithm adapted from Fisher's exact test in order to eliminate items that occur together by chance with items in the set H' and NH. The result is shown in Table 6.12.

Dataset	10% sampling			20% sampling			30% sampling		
	% change	H'	NH	% change	H'	NH	% change	H'	NH
Access	39	82771.56	42870.80	37	90193.74	39370.58	38	87539.11	40375.99
Access2	47	1630.73	993.52	46	1668.32	972.64	46	1683.30	960.81
Nasa	40	1677.56	700.80	41	1570.39	738.13	42	1505.21	797.89
NasaAug	23	3447.31	792.93	21	3439.17	862.39	21	3504.36	838.21
Retail	43	56.78	76.02	42	58.03	74.94	42	60.57	71.26
Uaccess	42	6413.79	2928.31	41	6348.31	3074.36	41	6090.60	3266.48

Table 6.12: Profit Analysis with Chance Collision

It can be seen that there is a reduction in the profit generated because some items are excluded when calculating the profit since they occurred by chance. However, it still shows that profit generated by the items in set H' is substantially higher than the profit generated by the items in set *NH*.

6.4 Experiment 3: Rules Extraction

This experiment was designed to track the rule base that contain rules of the form X → Y where X represents low weight items on the basis of domain weight

that transited to high weight on the basis of overall weight, whereas Y contains at least one high weight item on the basis of domain weight.

We supplied top ranked items (top p%) in terms of overall weight generated from the winner approach (sub-graph) to the standard rule generator with the setting mentioned in Chapter 5 then extracted all rules with the condition mentioned above. Table 6.13 displays the result for the rules extraction for each dataset.

Dataset	No of rules extracted					
	10% sampling		20% sampling		30% sampling	
Access	5	(0.25%)	27	(0.77%)	214	(0.63%)
Access2	42	(1.55%)	42	(1.55%)	42	(1.55%)
Nasa	156	(1.7%)	156	(1.7%)	156	(1.7%)
NasaAug	8	(0.1%)	8	(0.1%)	8	(0.1%)
Retail	46	(8.36%)	78	(8.12%)	42	(6.54%)
Uaccess	0	(0%)	264	(7.89%)	264	(7.89%)

Table 6.13: Rules Generation Summary

It can be seen that there are a number of rules of the form X → Y where X represents low weight items on the basis of domain weight that transited to high weight on the basis of overall weight, whereas Y contains at least one high weight item on the basis of domain weight. The numbers in the bracket showed the percentage of rules extracted out of the total number of rules produced from the rule generator.

These extracted rules are consistent across all sampling levels as we went through and compared all the extracted rules at each sampling level. In terms of Access dataset, rules extracted at 10% sampling were also found at 20% and 30% sampling likewise rules extracted at 20% were found at 30%. Rules extracted at 20% sampling in Retail dataset covered all extracted rules at 10% and 30% sampling. Additionally, Rules extracted in Access2, Nasa, NasaAug and Uaccess are identical across all sampling levels.

It is important to note that these rules would have not been generated if the items were weighted merely on the basis of their domain weight alone as they

would have not met the top p% threshold and would thus not have participated in the rule generation phase. Thus this result represents the key contribution that weight of an item should not be assigned on the basis on domain knowledge alone as it has strong interaction with other items.

6.5 Discussion and Analysis

In Experiment 1, it can be clearly seen from the results that the Weight Transmitter model is far superior to Label Propagation with the use of Gaussian Kernel as the propagation mechanism. A possible reason is that the Label Propagation method was designed to propagate discrete values in the form of class labels rather than numeric values.

With the employment of the Proportional Confidence instead of Gini index, a big improvement in performance was observed in most datasets while others presented slightly lower performance. A possible reason for this improvement is that the Gini index only considers the interaction between an item and its directly connected neighbors while Proportional Confidence expands the field of interaction by taking into account the effect of the interaction between the neighbors of directly connected neighbors.

In addition, it can be observed that, our proposed landmark assignment method with the used of stratified random sampling is more efficient than the original landmark assignment method with the use of simple random sampling since it produced better performance in all performance metrics on almost all of the datasets.

In order to clearly illustrate the impact of our proposed landmark item assignment method, we provide an example of item weight distribution for the Nasa dataset. Figure 6.1 shows the domain weight distribution. It can be seen that most of the items have domain weight ranging between 0.1 and 0.35. Thus when landmark items are assigned on the basis of domain weight alone, it is likely that a large proportion of low weight items will be assigned as landmarks.

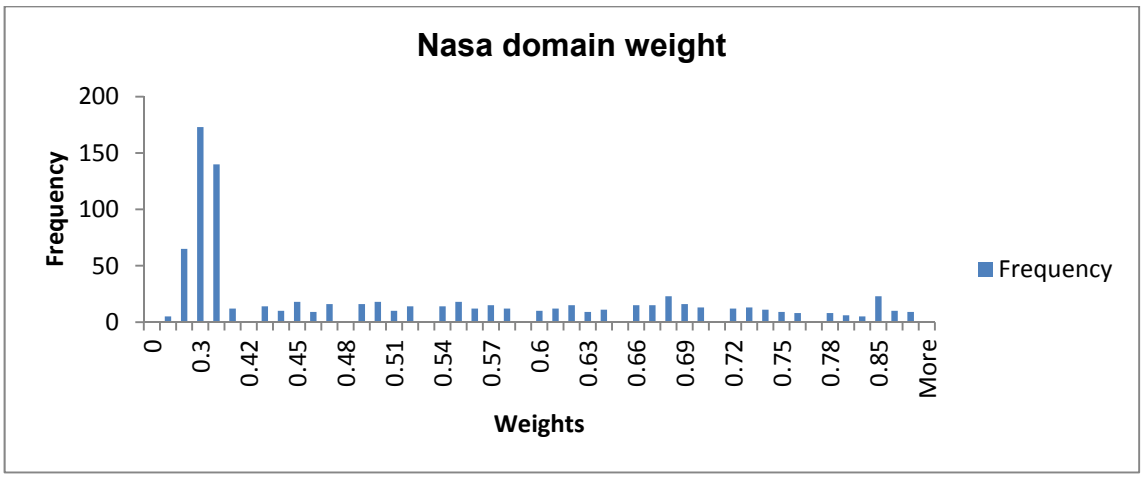


Figure 6.1: Domain Weight Distribution of Nasa Dataset

However, with our proposed landmark items assignment, we assign landmark items on the basis of transmission weight instead. Transmission weight of item was calculated on the basis of the interaction of that item with items in its neighborhood by supplying a seed weight value for each item to transmit to its neighborhood. Seed weight values were assigned at random from a pool of values where the highest value was much smaller than the lowest value of known set of domain weights in the landmark set. With this approach, we avoid introducing bias into the weight fitting process as the seed values supplied are numerically too small.

Ideally, the seed weight should be set to zero for each item so that the weight obtained from solving the linear model as given by Equation4.2 reflected the true transmission weight. However, Equation4.2 does not yield a solution with the seed vector set to zero. Hence we adopted the pragmatic solution of the introduction of a very small random seed vector that is guaranteed to be outside the range of the domain weight vector and will not bias the computation of the transmission weight. In our case, the ground truth was available and hence the domain weight vector was available for all datasets. In a real world scenario domain experts should be able to give an indication of the maximum possible domain weight value for a given dataset even if they are unable to specify with high precision individual weight values.

Figure 6.2 displays the transmission weight distribution. It can be seen that now the distribution is shifted by expanding the range from 0.1 to 0.6, which covers a

wider range than the pure domain weight. The bin ranges were then set based on this transmission weight and landmarks were then assigned in the same proportion to each bin range. This is due to the fact that item neighbourhood should determine the weight of an item. With this approach, items that are assigned as landmark items will reflect the impact of neighborhoods that are representative of the weight category (bin range) with the same proportion of landmark items.

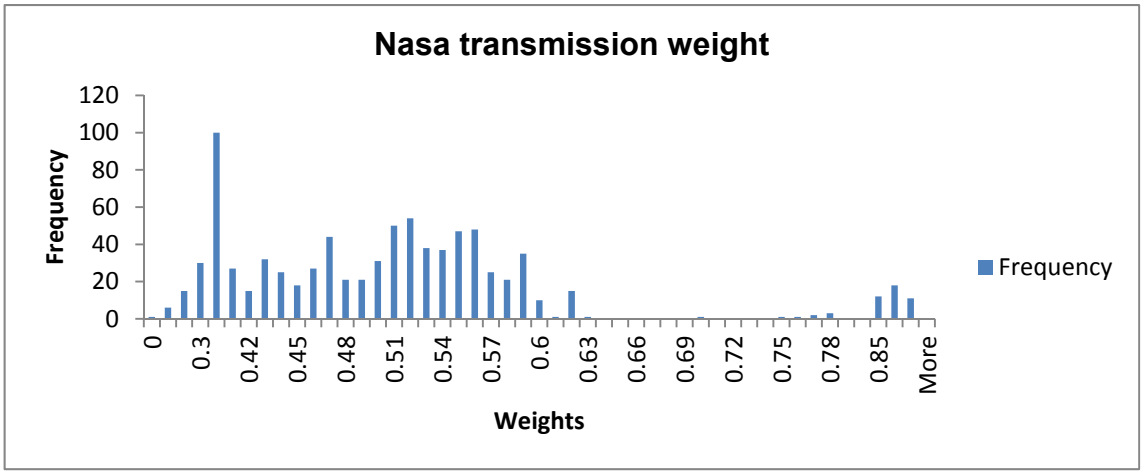


Figure 6.2: Transmission Weight Distribution of Nasa Dataset

Figure 6.3 illustrates overall weight distribution which can be clearly seen that the pattern of the distribution of overall weight follows more closely the pattern of distribution of transmission weight rather than domain weight.

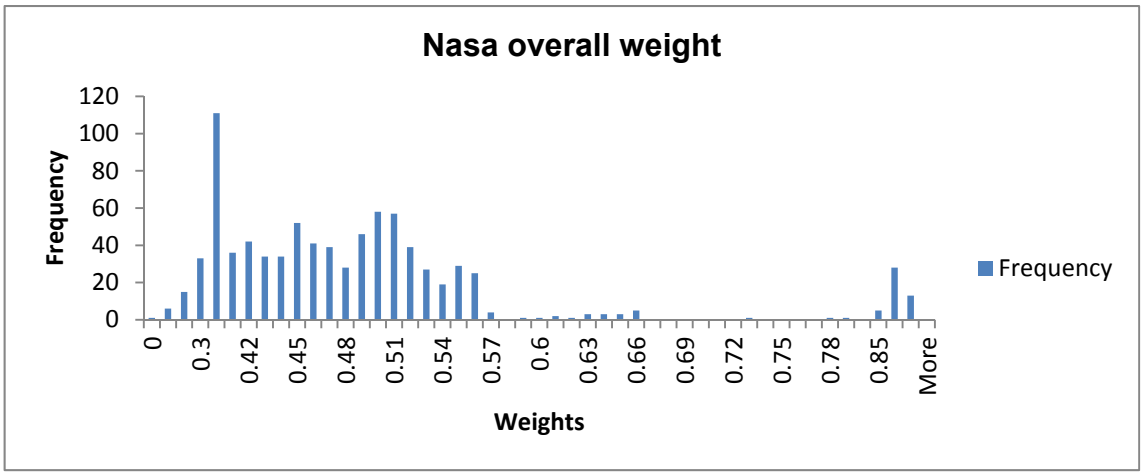


Figure 6.3: Overall Weight Distribution of Nasa Dataset

In Experiment 2, it can be observed that the sub-graph approach produced a competitive performance to global approach. However, the global approach produced a better performance at the smaller sampling levels of 10% and 20%, while the sub-graph approach performed better at a higher sampling level. This is due to the fact that utilizing stratified random sampling on a smaller size of graph results in a smaller number of landmark items being assigned. Even though in proportional terms the landmark assignment may be very similar, absolute number is important as a "critical mass" is required in order to capture neighborhoods accurately. Thus it would be harder to reflect the impact of neighborhoods that are representative of the weight category (bin range).

However, since the main idea is to reduce the time complexity, we would rate sub-graph approach to be the winner since the execution time is very much smaller while giving a competitive performance.

In terms of profit analysis, the results supported our research premise that items whose weights have transited from the low weight to high weight category would generate higher indirect profit than items which do not make such a transition.

6.6 Summary

In this Chapter, we have presented empirical findings of the research together with the analysis of the experimental results. The results showed that the Weight Transmitter model is far superior to the Label Propagation method. Additionally, a big improvement was presented when utilizing Proportional Confidence instead of Gini index in Weight Transmitter model. Also, our proposed landmark item assignment has produced better performance compared to the original simple random sampling. In addition, the proposed sub-graph approach has reduced a vast amount of time complexity in fitting weight while giving a competitive accuracy to the global approach.

In the next Chapter, we will conduct two case studies with datasets obtained from the University of Auckland which is the Access 2 dataset and the 1998

Soccer World Cup competition dataset in order to provide a greater depth of analysis.

Chapter 7 Case Study

7.1 Introduction

In the previous chapter, we presented empirical findings together with an analysis of the results. In this chapter, we will conduct two case studies on datasets obtained from the University of Auckland which is named the Access2 dataset, and secondly with the 1998 Soccer World Cup competition dataset. There are two main reasons for conducting such case studies.

The first reason relates to testing the consistency of the results obtained on two further datasets. Secondly, and more importantly, we would like to provide a greater depth of analysis by analyzing the impact of cases where major discrepancies exist between weight assignments made from the domain based approach and those from the weight transmitter model. In such cases, we perform a rigorous analysis on the rules containing items that exhibited such discrepancies with a view to determining the significance, if any, of such differences.

7.2 Dataset and Data Pre-Processing

The World Cup 1998 site logs dataset was contributed to the Internet Traffic Archive (ITA) by Martin Arlitt and is available at *http://www.acm.org/sigcomm/ITA/*. It consists of all the requests made to the 1998 World Cup Web site between April 30, 1998 and July 26, 1998. A total of 33 different World Cup HTTP servers were used for the collection at four geographically dispersed sites: Paris, France; Plano, Texas; Herndon, Virginia; and Santa Clara, California. The total numbers of requests received by the World Cup site throughout this period of time was 1,352,804,107 requests.

In order to pre-process the data, two main factors, which are webpage and user, needed to be considered since these two factors were used to generate the transaction dataset and to assign the domain weight for each webpage. The first step of the pre-processing starts with obtaining a list of all requested web

pages and ranking them by total time that they were accessed. Then we selected only the top 350 requested web pages for consideration. Next, we obtained a list of all users that made a request on those selected web pages. Then three different strategies for data selection and pre-processing were performed:

- Strategy1 - select one day from 6 periods which correspond to the group stage of the competition, the round of 16, the quarter finals stage, semifinals stage, third place play-off, finally culminating in the last game between the winners of the two semi-final games. In order to reduce the mining time to a manageable amount, the data obtained was pre-processed by randomly selecting 100 users out of 399,183 users from the user list extracted.

- Strategy 2 – select all dates in the knockout period from round 16 to the final game. Then pre-process by randomly selecting 100 users out of 970,804 users from the user list to filter data.

- Strategy 3 – select all dates in the knockout period from round 16 to final game. Then pre-process by selecting top 100 users out of 970,804 users from the user list ranked by the total number of requests they made.

For each set, we considered pages as items and a sequence of clicks on a set of web pages that occurred through a session as transactions. The maximum time was set to 15 minutes for each session. Also, we applied average dwelling time on a web page, which was taken across all transactions to use as a proxy for item weight.

We then input each set to the Weight Transmitter model in order to identify the best dataset, which turned out to be that resulting from Strategy 3 (The results of Strategy 1 and Strategy 2 are presented in Appendix D).

Once we had the World Cup dataset ready, the experiments were then performed by following the same experimental setup described in Chapter 5.

7.3 Results

It can be seen from Table 7.1 that our proposed sub-graph approach produced the best performance for all metrics across all sampling levels. This result clearly highlights the consistency of our proposed sub-graph approach.

	Method	10%	20%	30%
Precision	Global-Simple Random	51.43	58.71	65.50
	Global-Stratified Random	56.93	64.30	72.47
	Sub-graph Stratified Random	62.39	67.10	76.99
Recall	Global-Simple Random	38.82	44.39	49.77
	Global-Stratified Random	42.21	47.71	54.41
	Sub-graph Stratified Random	46.07	48.15	56.70
Percentage Accuracy	Global-Simple Random	86.25	87.86	89.62
	Global-Stratified Random	87.43	88.99	90.93
	Sub-graph Stratified Random	88.49	89.06	91.59
Lift	Global-Simple Random	4.8170	5.4991	6.0934
	Global-Stratified Random	5.2541	6.0229	6.7883
	Sub-graph Stratified Random	5.8443	6.2847	7.2113

Table 7.1: Performance Analysis (High Weights)

In terms of the execution time, it can be clearly seen from Table 7.2 that the execution time in terms of sub-graph approach is very much smaller than the global approach. This result also supports our intuition that partitioning the graph into several sub-graphs results in execution time savings in the weight fitting process.

Global	Sub-Graph			
	Setting Threshold Value	Pre-Process Matrix	Partitioning	Weight fitting
78	3	2	20	11

Table 7.2: Model Execution Time (millisecond)

In the case of profit analysis, Table 7.3 shows that profit generated by the items in set H' is substantially higher than the profit generated by the items in set *NH* where $H' = \{i | i \in I$ where i is in the top p percentile on the basis of overall weight but not on the basis of domain weight $\}$ while *NH* contains items that remain low in weight without making the transition to high weight category in

terms of overall weight as stated earlier in Chapter 4. The top p percentile was set at 40%. In addition, the results also show a big transition of low weight items on the basis of domain weight to high weight items on the basis of overall weight at about nearly 50%.

10% sampling			20% sampling			30% sampling		
% change	H'	NH	% change	H'	NH	% change	H'	NH
47	9287.00	3738.16	46	9843.85	3227.33	47	9774.90	3209.81

Table 7.3: Profit Analysis

In addition, after we utilized the algorithm adapted from Fisher's exact test, the result still shows that profit generated by the items in set H' is substantially higher than the profit generated by the items in set *NH* even though there was a reduction in the profit generated because some items are excluded when calculating the profit since they occurred by chance.

10% sampling			20% sampling			30% sampling		
% change	H'	NH	% change	H'	NH	% change	H'	NH
47	9157.93	3195.87	46	9737.55	2667.71	47	9662.20	2653.18

Table 7.4: Profit Analysis with Chance Collision

In terms of rules extraction, we followed the same experimental setup as in Chapter 5 by supplying the top p% of items by overall weight to a standard rule generator and then extracted the rule base that contains rules of the form X → Y where X represents low weight item on the basis of domain weight that transited to high weight on the basis of overall weight and Y contains at least one high weight item on the basis of domain weight.

Table 7.5 displays summary statistics on the rules extracted where *minSup* denotes the minimum support threshold, *minConf* denotes the minimum confidence threshold and *minLift* denotes the minimum Lift threshold. The terms *support*, *confidence and Lift* follow the standard definitions associated with rule mining (Agrawal et sl., 1993).

P%	minSup	minConf	minLift	No of rules extracted		
				10% sampling	20% sampling	30% sampling
40	0.04	0.7	1.0	20 (0.23%)	20 (0.23%)	20 (0.23%)

Table 7.5: Rules Generation Summary

The results show that these extracted rules are consistent across all sampling levels as the rules extracted were identical and equal in number across the levels.

All in all, with the results from the experiment with World Cup dataset, it can be observed that our proposed sub-graph approach performed better than the global approach, which illustrated the consistency of our proposed method.

7.4 Rules Analysis

In this section we analyze rules generated from the Access2 and World Cup datasets. We select 2 rules from the World Cup dataset and 4 rules from Access2 for in-depth analysis. All of the selected rules are of the form: $X \rightarrow Y$ where X is a low weight item on the basis of domain weight whereas Y is a high weight item on the basis of domain weight. Note that both X and Y are high weight items when measured by overall weight.

Although all selected rules have rule confidence greater than the confidence threshold, it cannot be conclusively concluded that the low weight item (X) boosts the appearance of high weight item (Y). This is due to the fact that these rules may contain spurious patterns involving items with substantially different support levels which Xiong et al. (2006) defined as cross-support patterns. This type of pattern severely degrades the effectiveness of the standard confidence measure. For example, the rule *caviar* → *milk* is a possible cross-support pattern since the support of caviar is expected to be much lower than the support of milk. The problem is that even when conf(*caviar* → *milk*) is high the rule is of dubious value as in general, milk occurs with very much greater frequency in relation to caviar. This means that attributing caviar to a boost in milk sales is fraught with difficulty, simply because milk occurs on its own with

high frequency and hence is unlikely to be affected by the presence (or otherwise) of caviar in the same transaction.

In order to confidently say that a low weight item (X) boosts the appearance of the high weight item (Y), we identify rule terms C for all rules of the form: $C \cap \sim X \rightarrow Y$ where C is not a low weight item. Then for each rule, we utilize Fisher's exact test to check whether C and Y occur by chance. We then collect all rule terms C that survive the Fisher test into a set U, whenever the confidence of a rule $C \cap \sim X \rightarrow Y$ is greater than 0. Next, we track all the rule statistics for the rule $U \rightarrow Y$.

The rule $U \rightarrow Y$ represents the boost of the appearance of the high weight item (Y) by a collection of non-low weight items (U) while the rule $X \rightarrow Y$ represents the boost of the appearance of the high weight item (Y) by the single low weight item (X). Now if the confidence $X \rightarrow Y$ is higher than the confidence of the rule $U \rightarrow Y$ then we can infer that the appearance of the low weight item (X) has a greater influence on the appearance of the high weight item (Y) than the collective appearance of non-low weight items U.

However, typically, the confidence of the rule $U \rightarrow Y$ is expected to be large and may contain cross-support patterns and thus it would not be appropriate to directly compare its confidence with that of the rule $X \rightarrow Y$. Alternatively, we can compare the confidence of the rule $Y \rightarrow U$ with the confidence of the rule $Y \rightarrow X$. If the confidence of the rule $Y \rightarrow U$ is less than the confidence of $Y \rightarrow X$ then we can not only infer that X and Y do not occur by chance, but we can make the even stronger inference that the appearance of Y receives a greater boost from a single lowly ranked item X than the collective appearance of higher ranked items, when that ranking is made on the basis of domain knowledge alone.

Dataset	Rule X → Y	conf (X → Y)	*conf (Y → X)*	No. of items in U	conf (U → Y)	*conf (Y → U)*
Access2	7902 → 7903	0.9655	*0.9825*	3	1	*0.9123*
	7909 → 7903	0.9818	*0.9474*	15	1	*0.7368*
	7911 → 7903	0.9310	*0.9474*	9	1	*0.7544*
	7911 → 7914	0.9483	*0.9322*	11	1	*0.7119*
World Cup	157 → 204	0.8333	*0.3750*	10	1	*0.0500*
	272 → 167	0.7083	*0.2267*	9	1	*0.0533*

Table 7.6: Rules Analysis

It can be clearly seen from Table 7.6 that the confidence of the rule *Y→ U* is less than the confidence of *Y → X* in all of the above cases considered. This result supports our research premise that the importance or weight of an item cannot be deduced on the basis of domain knowledge alone and thus reinforces the need for weight propagation models such as Weight Transmitter that we have proposed in this research.

In addition, it should be noted that interesting rules of the type X → *Y* presented above would not be discovered with the sole use of domain information in assigning weights. This is due to the fact that items such as X appearing in the antecedent of rules of the type: X → *Y* would never be included in the initial set of items submitted to the rule generator since by definition such items are too low in overall weight to be included in the top ranked list that is fed to the rule generator.

Apart from assessing statistical significance, we were also interested in assessing the real world significance or impact of such rules. In this respect, we traced back to the actual web pages involved in order to get a better understanding of the interaction between web pages. We selected rule 157 → 204 from World Cup Dataset as an example. The item 157 represents page */english/playing/trivia.html* while item 204 represents page */english/tickets/tickets_fr.html*. Unfortunately the actual links had been removed and therefore we cannot describe what information was shown on each web page. However from the name of the link, it can be inferred that page */english/playing/trivia.html* contained information about fixtures of all matches, including date and time of playing. Once soccer fans open this page and find

matches of interest, this leads to a visit to page */english/tickets/tickets_fr.html* to get information on ticket prices and on line payment of tickets. The domain weight of */english/playing/trivia.html* is relatively low when compared to */english/tickets/tickets_fr.html* due to the fact that the former web page is used merely to obtain fixture information requiring less dwelling time than deciding whether to buy the ticket and the subsequent processing time needed to complete on line payment.

7.5 Hyperclique Pattern Discovery

As mentioned in Chapter 4, we are also interested in checking for the existence of cliques of items. Hypercliques, when they exist, are potentially of great interest as they denote strong inter-relationships between all possible combinations of items within a sub group of items and as such are a succinct description of strong patterns within a dataset.

In order to find a clique a simple hyperclique pattern algorithm is utilized. More efficient algorithms are available for hyperclique discovery (Huang et al., 2004; Yamamoto et al., 2008; Ozaki and Ohkawa, 2009) but the author did not have access to the executable versions of these and implementation of such algorithms was considered to be out of scope of the current research.

To search for hypercliques, the extracted rules were scanned and the h-confidence of each rule was calculated. The items in the rules that had h-confidence above the threshold were accumulated into a clique membership set.

To illustrate how the h-confidence was calculated, we provide an example of rule {7898 7899 7908} -> {7910 7911 7913 7914}. First we calculated support of itemset P = {7898, 7899, 7908, 7910, 7911, 7913, 7914} and stored it in a variable called *ItemsetSupport*. Next, we calculated support of each item in the rule taken across all items in P and stored the minimum support value in a variable called *MinSupport*. Then we divided ItemsetSupport by MinSupport. If

the result of the division was greater than the given h-confidence threshold, all items in the rules were accumulated into a clique membership set.

The results showed that a clique of size 7 existed in the Access2 dataset whereas no such clique was discovered in World Cup dataset with the clique membership threshold set at 0.7. Unfortunately, we could not map items to actual web pages due to privacy issues involved in the Access 2 dataset but from the donor we understand that the items correspond to web pages accessed by students at the University of Auckland during the course of a tutorial in Java Programming. The tutorial content dictated that the web pages represented in the clique be accessed together as they were highly inter-related to the completion of the tutorial's stated objectives and this led to the pattern of access encapsulated by the hyperclique.

While it is relatively easy to explain the existence of specific hypercliques after the fact (i.e. after they were discovered using an automated tool) it is a different matter altogether to predict and identify the existence of such hypercliques given a dataset exhibiting large dimensionality (number of items) and a large number of instances. No amount of domain knowledge can be guaranteed to be sufficient to predict such hypercliques without the use of automated support.

It is also worth noting that hypercliques extracted were defined on high weight items and are thus potentially of greater value than hypercliques extracted on items that merely have strong inter-relationships with each other. Table 7.7 shows the items defining the hyperclique in the Access 2 dataset and the rule from which such a hyperclique was extracted.

7898 7899 7908 7910 7911 -> 7913 7914
7898 7899 7908 7910 7913 -> 7911 7914
7898 7899 7908 7910 -> 7911 7913 7914
7898 7899 7908 7911 -> 7910 7913 7914
7898 7899 7908 7913 -> 7910 7911 7914
7898 7899 7908 7914 -> 7910 7911 7913
7898 7899 7908 -> 7910 7911 7913 7914
7898 7899 7910 -> 7908 7911 7913 7914
7898 7899 7911 -> 7908 7910 7913 7914
7898 7899 7913 -> 7908 7910 7911 7914
7898 7899 -> 7908 7910 7911 7913 7914
7898 7913 -> 7899 7908 7910 7911 7914
7908 7914 -> 7898 7899 7910 7911 7913
7898 7910 -> 7899 7908 7911 7913 7914

Table 7.7: Example of a clique of 7 items in Access2 Dataset

7.6 Summary

In this Chapter, we conducted two cases studies on the Access 2 and 1998 Soccer World Cup datasets. The results are consistent with those produced with our experimental datasets. The superiority of the local sub graph approach vis-à-vis the global approach was reinforced by the conduct of these two case studies.

Overall, the case studies also illustrated the important role that weight propagation plays in assessing the true worth of items in terms of the assigned weight obtained through transmission.

Chapter 8 Conclusion

This chapter consists of two sections. In the first section, we provide a research summary and emphasize what we have achieved in this research. In the second section, we present some possible directions for future work in this research area.

8.1 Research Achievements

In this research, we presented a novel approach for weight propagation which utilizes the interactions between items rather than taking into account the inherent properties of items. We have implemented an efficient weight transmitter model that accurately estimates weights of items by utilizing a small set of items (landmark items) whose weights are known. Moreover, we have developed a novel interestingness measure call *Proportional Confidence* which is derived from the standard confidence measure.

To improve the efficiency of the original weight transmitter model proposed by Koh et al. (2012), we proposed a novel method of graph partitioning called *Sub-graph generation algorithm* that partitions a single global graph representing inter-relationships amongst N items into a number of smaller sub-graphs. In addition, we proposed a novel method of landmark items assignment that utilizes stratified random sampling approach, which enables us to better reflect the ground situation of items' neighbourhoods.

Our experimentation demonstrated a significant improvement in accuracy when utilizing *Proportional Confidence* in the proposed weight transmitter model. Also, our proposed landmark item assignment scheme produced better performance compared to the original simple random sampling scheme. In addition, the run time overhead of the weight fitting process was reduced substantially with the use of the localized approach, while maintaining a competitive level of accuracy to the global approach.

The main contributions of this research are summarized as below:

- A new measure for quantifying interaction between items called *Proportional Confidence* has been proposed. It takes into account not just the interaction between an item and its directly connected neighbours but also the effect of the interaction between neighbors of directly connected neighbors.

- A novel method to partition a global set of items and generate subsets of items called *Sub-graph generation algorithm* has been proposed. This algorithm utilizes a divide-and-conquer approach to partition a global graph into a number of smaller size of sub-graphs. This approach substantially reduces the run time overhead in the weight fitting process thus resulting in improving the efficiency of the weight transmitter model.

- A new approach to landmark item weight assignment has been proposed. This approach utilizes stratified random sampling to allocate equal numbers of landmark items to low, medium and high weight categories which are set based on the transmission weight of each item which is calculated on the basis of the interaction of that item with items in its neighborhood. With this approach, items that are assigned as landmark items will reflect the impact of neighborhoods that are representative of the weight category under consideration.

Overall, we showed in this research that the weight transmitter model is in a better position to assess item weight rather than a pure domain weight based approach as it takes into account interactions between items. This is supported by the result of profit analysis conducted on ground truth data that a substantial percentage of items transited from the low weight category to the high weight category and that these items contributed higher profit on the average than items that were weighted highly on domain weight alone.

In this research we set out to examine the hypothesis that interaction between a given item and other items in its neighbourhood plays a significant role in determining its weight. Based on the results produced through extensive

experimentation we conclude that the novel methods that we proposed demonstrated the truth of our research premise.

8.2 Future Work

One possible area for future work would explore the possibility to further optimize the weight fitting process. In this research, we utilized Gaussian elimination method to solve the linear equations in our proposed Weight Transmitter model. Gaussian elimination method is known to be a direct method in solving linear equations which is generally employed to solve the system that is not large (a matrix of the order of 1000) (Kalambi, 2008). However, in a real world scenario datasets can contain a large number of items sometimes of the order of hundred-thousand or even a million. Due to the non linear scalability of Gaussian elimination with respect to run time, it will not be feasible to solve equations of this magnitude. In addition, Gaussian elimination is known to be prone to rounding error which, in general will increase with the size of the matrix manipulated (Kalambi, 2008). Thus for very large datasets, an alternative approach to solving the linear model is highly desirable.

One such alternative approach is to utilize an iterative method such as Jacobi method (the method is named after Carl Gustav Jakob Jacobi). Unlike the direct methods, Jacobi iterative method starts with making an initial approximation and then refining the solutions progressively through a recursive process until convergence is achieved (Young, 2003). The main time complexity in solving the equations is the matrix vector product which requires $O(N^2)$ per iteration thus it requires $k \cdot O(N^2)$ in total where k is the number of iterations required to meet convergence criterion, with N being the total number of items.

Consider a scenario where the dataset contains a million items; utilizing an iterative approach such as Jacobi is very likely be more efficient than Gaussian elimination as the number of iterations k is very small in practice, when compared to the number of items N. As a consequence, the future research should investigate the optimization of the weight fitting process by utilizing approximate approaches such as the Jacobi iterative method.

Future research should also be able to deploy the Weight Transmitter model in a data stream environment. The proposed Weight Transmitter model has proven to be efficient in a static data environment. However, in a data stream environment, interaction weights between items change over time as new transactions continuously arrive. These new incoming transactions may include new items, which imply that a process needs to be put in place to assign weights for the new incoming items. Furthermore, connections between existing items may also change over time, thus requiring a reassessment of the overall weight assigned to the items involved. This is especially true in highly dynamic environments such as web click stream applications.

Thus there is a need for a mechanism that interfaces with the Weight Transmitter model. This mechanism would need to keep track of changes in the stream and adjust weights by identifying items that had significant changes in their interaction with other items. The main components of such a mechanism should include a buffer to store transactions and a sliding window to keep track of changes in the stream. Then, the domain weights for the new items and the transactions in each window can be supplied to the Weight Transmitter model.

Essentially, an incremental version of Weight Transmitter needs to be developed to efficiently update weights of items with the change in access patterns for items when significant changes occur. Two types of changes will need to be monitored. The first type involves changes in the domain weight of an item over time. For example in retail application, profitability of items changes over time depending on marketing and pricing policies, amongst other factors. Similarly, in a web click stream environment, dwelling time is subject to change, depending on external factors such as changing demand over time for services or content offered by the page in question. The second type of change involves changes in interaction patterns between items and corresponds to changes in interaction weights. Again the causative factors for such changes are similar to those that occur in type 1 change.

References

Agarwal, R., Aggarwal, C., & Prasad, V. V. V. (2000). A tree projection algorithm for generation of frequent itemsets. *Journal of Parallel and Distributed Computing, 61(3),* 350-371.

Agrawal, R., Imielinski, T., & Swami, A.N. (1993). Mining association rules between sets of items in large databases. *In: Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data,* 207-216.

Barber, B. & Hamilton, H. J. (2003). Extracting Share Frequent Itemsets with Infrequent Subsets. *Data Mining and Knowledge Discovery, 7(2)*, 153-185.

Bayardo, R. J. Agrawal, R., & Gunopulos, D. (2000). Constraint-Based Rule Mining in Large, Dense Databases. *Data Mining and Knowledge Discovery, 4(2-3)*, 217-240.

Bengio, Y., Delalleau, O., & Le Roux, N. (2006). Label Propagation and Quadratic Criterion. Semi-Supervised Learning, *MIT Press*, 193-216.

Berry M., & Linoff G. (1997). Data Mining Techniques: For Marketing, Sales, and Customer Support, New York, NY: John Wiley and Sons.

Brin, S. Motwani, R., Ullman, J. D., & Tsur, S. (1997). Dynamic itemset counting and implication rules for market basket data. *In: Proceedings of the 1997 ACM SIGMOD international conference on Management of data, 26(2)*, 255-264.

Cai, C.H., Fu, A.W.C., Cheng, C.H., & Kwong, W.W. (1998). Mining association rules with weighted items. *In: Proceedings of the 1998 International Symposium on Database Engineering & Applications,* 68-77.

Cochran, G. W. (1997). *Sampling Techniques, Third Edition*. Hoboken, NJ: John Wiley and Sons.

Cooley, R., Mobasher, B., & Srivastava, J. (1997). Web mining: information and pattern discovery on the World Wide Web. *In: Proceeding of the Ninth IEEE International Conference on Tools with Artificial Intelligence,* 558-567.

Dash, N. (2005). Selection of the Research Paradigm and Methodology. Online Research Methods Resource. Retrieved April 1, 2012, from http://www.celt.mmu.ac.uk/researchmethods/Modules/Selection_of_methodology/index.php

Fayyad, U., Piatetsky-Shapiro, G., & Smyth, P. (1996). From data mining to knowledge discovery in databases. *American Association for Artificial Intelligence, 17(3),* 37-54.

Gamberger, D., Lavrac, N., & Krstacic, G. (2003). Active subgroup mining: a case study in coronary heart disease risk group detection. *Artificial Intelligence in Medicine, 28(1),* 27-57.

Gardner, M., & Bieker, J. (2000). Data mining solves tough semiconductor manufacturing problems. *In: Proceeding of the Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining,* 376-383.

Gentle, J. E. (1998). *Numerical Linear Algebra for Applications in Statistics*. AAAI Press/MIT Press.

Gerritsen, R. (1999). Assessing loan risks: a data mining case study. *IT Professional, 1(6),* 16-21.

Hauser, T. A., & Scherer, W. T. (2001). Data Mining Tools for Real Time Traffic Signal Decision Support and Maintenance. *In: Proceeding of the IEEE International Conference on Systems, Man, and Cybernetics, 3,* 1471-1477.

He, Y., & Han, J. (2003). Pushing support constraints into association rules mining. *Knowledge and Data Engineering, 15(3),* 642-658.

Hevner, A. R., March, S. T., Park, J., & Ram, S. (2004). Design science in information systems research1. *MIS Quarterly, 28(1),* 75-106.

Holt, J. D., Chung, S. M. (2001). Mining association rules using inverted hashing and pruning. *Information Processing Letters, 83(4),* 211-220.

Huang, Y., Xiong, H., Wu, W., & Zhang, Z. (2004). A Hybrid Approach for Mining Maximal Hyperclique Patterns. *In: Proceeding of the 16th IEEE International Conference on Tools with Artificial Intelligence,* 354-361.

Jian, W., & Ming, L.X. (2008). An effective mining algorithm for weighted association rules in communication networks. *Journal of Computers, 3(10),* 20-27.

Jolliffe, I. T. (2002). Principal Component Analysis, Second Edition. Springer, Heidelberg.

Kalambi, B. I. (2008). A Comparison of three Iterative Methods for the Solution of Linear Equations. *Journal of Applied Sciences and Environmental Management, 14 (2)*, 53-55.

Kantardzie, M. & Srivastava, A. N. (2005). Data Mining: Concepts, Models, Methods, and Algorithms. *Journal of Computing and Information Science in Engineering, 5(4)*, 394.

Klabbers, J. H. G. (2006). A framework for artifact assessment and theory testing. *Simulation & Gaming*, 37(2), 155-173.

Kleinberg, J. M. (1999). Authoritative sources in a hyperlinked environment. *Journal of the ACM, 46(5)*, 604-632.

Koh, Y. S., Pears, R., & Dobbie, G. (2011). Automatic assignment of item weights for pattern mining on data stream. *In: Advances in Knowledge Discovery and Data Mining. Lecture Notes in Computer Science, 6634,* 387–398

Koh, Y. S., Pears, R., & Dobbie, G. (2012). WeightTransmitter: Weighted Association Rule Mining Using Landmark Weights. *In: Advances in Knowledge Discovery and Data Mining. Lecture Notes in Computer Science, 7302,* 37–48.

Koh, Y. S., Pears, R., & Yeap, W. (2010). Valency based weighted association rule mining. *In: Advances in Knowledge Discovery and Data Mining. Lecture Notes in Computer Science, 6118,* 274–285.

Koh, Y. S., & Rountree, N. (2005). Finding sporadic rules using apriori-inverse. *In: Advances in Knowledge Discovery and Data Mining. Lecture Notes in Computer Science, 3518,* 97–106.

Kotsiantis, S., & Kanellopoulos, D. (2006). Association Rules Mining: A Recent Overview. *GESTS International Transactions on Computer Science and Engineering, 32(1),* 71-82.

Kraft, M. R., Desouza, K. C., & Androwich, I. (2003). Data mining in healthcare information systems: case study of a veterans' administration spinal cord injury population. *In: Proceedings of the 36th Hawaii International Conference on System Sciences,* 159–167.

Lin, L., & Shyu, M. L. (2010). Weighted association rule Mining for Video Semantic Detection. *International Journal of Multimedia Data Engineering and Management, 1(1),* 37-54.

Orlikowski, W. J., & Baroudi, J. J. (1991). Studying information technology in organizations: Research approaches and assumptions. *Information Systems Research, 2(1),* 1-28.

Ozaki, T., & Ohkawa, T. (2009). Efficient Discovery of Closed Hyperclique Patterns in Multidimensional Structured Databases. *IEEE International Conference on Data Mining Workshops,* 533-538.

Park, J. S., Chen, M. S. & Yu, P. S. (1995). An effective hash-based algorithm for mining association rules. *In: Proceedings of the 1995 ACM SIGMOD International Conference on Management of Data, 24(2),* 175-186.

Pears, R., Koh, Y.S., & Dobbie, G. (2010). Ewgen: Automatic generation of item weights for weighted association rule mining. *In: Advance Data Mining and Application. Lecture Notes in Computer Science, 6440*, 36-47.

Raileanu, L.E., & Stoffel, K. (2004). Theoretical comparison between the gini index and information gain criteria. *Annals of Mathematics and Artificial Intelligence, 41(1),* 77–93.

Ramkumar, G. D., Ranka, S., Tsur, S. (1997). Weighted association rules: Model and algorithm. *In: Proceeding of the Forth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining.*

Smith, K. A., Willis, R. J. & Brooks, M. (2000). An Analysis of Customer Retention and Insurance Claim Patterns Using Data Mining: A Case Study. *The Journal of the Operational Research Society, 51(5),* 532–541.

Srikant, R., Vu, Q., & Agrawal, R. (1997). Mining Association Rules with Item Constraints. *The Journal of the Operational Research Society, 51(5),* 67-73.

Srivastava, J., Cooley, R., Deshpande, M., & Tan, P. N. (2000). Web usage mining: discovery and applications of usage patterns from Web data. *ACM SIGKDD Explorations Newsletter, 1(2),* 12-23.

Straub, D., Gefen, D., & Boudreau, M. C. (2004). The quantitative, positivist research methods website. Electronic Source.

Sun, K., & Bai, F. (2008). Mining weighted association rules without preassigned weights. *IEEE Transaction on Knowledge and Data Engineering, 20(4),* 489–495.

Tao, F., Murtagh, F., & Farid, M. (2003). Weighted association rule mining using weighted support and significance framework. *In: Proceedings of The Ninth International Conference on Knowledge Discovery and Data Mining*, 661–666.

Toivonen, H. (1996). Sampling large database for association rules. *In: Proceeding of the 22nd International Conference on Very Large Data Bases*, 134–145.

Wang, W., Yang, J., & Yu, P.S. (2004). WAR: Weighted Association Rules for Item Intensities. *Knowledge and Information Systems, 6(2),* 203–229.

Warren, J., Reboussin, R., Hazelwood, R. R., Gibbs, N. A., Trumbetta, S. L., & Cummings, A. (1999). Crime scene analysis and the escalation of violence in serial rape. *Forensic Science International, 100(1-2),* 37–56.

Xiong, X., Tan, T.N. ,& Kumar, V. (2003). Mining Hyperclique Patterns with Confidence Pruning. *In: Technical Report 03-006, January, Department of computer science, University of Minnesota - Twin Cities*.

Xiong, X., Tan, T.N. ,& Kumar, V. (2006). Hyperclique pattern discovery. *Data mining and knowledge discovery, 13(2),* 219–242.

Yamamoto, T., Ozaki, T., & Ohkawa, T. (2008). Discovery of Internal and External Hyperclique Patterns in Complex Graph Databases. *IEEE International Conference on Data Mining Workshops,* 301-309.

Yan, L., & Li, C. (2006). Incorporating pageview weight into an association-rule-based web recommendation system. *In: Australian Conference on Artificial Intelligence. Lecture Notes in Computer Science, 4304,* 577–586.

Yao, H., & Hamilton, H. J. (2006). Mining itemset utilities from transaction databases. *Data & Knowledge Engineering, 59,* 603-626.

Young, D. M. (2003). *Iterative Solution of Large Linear Systems*. New York, NY: Academic Press, Inc.

Zhang, Z., Wu, W., & Huang, Y. (2004). Mining dynamic inter-dimension association rules for local-scale weather prediction. *In: Proceedings of the 28th Annual International Computer Software and Applications Conference, 2*, 146-149.

Appendix A: Benchmarking performance of the Label Propagation method

Parameter	Dataset	10%	20%	30%
Alpha = 0.043 Convergence = 0.01	Access	19.31	24.98	31.88
	Access2	19.90	22.33	30.52
	Nasa	27.57	32.55	38.91
	NasaAug	20.54	27.38	35.42
	Retail	22.21	31.12	40.86
	Uaccess	20.60	29.54	35.24
Alpha = 0.043 Convergence = 0.001	Access	19.38	25.21	31.88
	Access2	20.31	22.66	30.52
	Nasa	27.96	32.66	39.00
	NasaAug	20.47	27.39	35.47
	Retail	22.44	31.24	40.97
	Uaccess	20.61	29.56	35.34
Alpha = 0.95 Convergence = 0.01	Access	18.92	23.61	30.85
	Access2	13.94	20.50	30.43
	Nasa	42.77	56.90	66.50
	NasaAug	32.31	47.27	57.34
	Retail	20.64	31.14	41.07
	Uaccess	18.57	28.81	39.05

Table 1: Precision Analysis of Label Propagation method

Parameter	Dataset	10%	20%	30%
Alpha = 0.043 Convergence = 0.01	Access	8.49	11.21	14.74
	Access2	9.74	9.86	13.19
	Nasa	11.10	13.33	16.56
	NasaAug	8.52	11.32	15.28
	Retail	8.77	12.82	18.13
	Uaccess	8.58	12.91	15.97
Alpha = 0.043 Convergence = 0.001	Access	8.52	11.30	14.73
	Access2	9.87	9.88	13.17
	Nasa	11.26	13.36	16.59
	NasaAug	8.42	11.24	15.16
	Retail	8.86	12.86	18.16
	Uaccess	8.54	12.89	16.02
Alpha = 0.95 Convergence = 0.01	Access	12.09	14.77	20.14
	Access2	10.67	15.34	22.15
	Nasa	17.74	23.95	30.30
	NasaAug	14.17	21.90	27.75
	Retail	11.13	16.67	22.20
	Uaccess	8.12	13.75	19.99

Table 2: Recall Analysis of Label Propagation method

Parameter	Dataset	10%	20%	30%
Alpha = 0.043 Convergence = 0.01	Access	70.74	72.34	74.39
	Access2	73.38	71.70	72.84
	Nasa	69.92	71.43	73.71
	NasaAug	69.18	70.70	73.33
	Retail	68.80	71.67	75.40
	Uaccess	69.22	72.26	74.19
Alpha = 0.043 Convergence = 0.001	Access	70.75	72.37	74.38
	Access2	73.30	71.47	72.80
	Nasa	70.00	71.41	73.73
	NasaAug	68.87	70.45	73.11
	Retail	68.84	71.67	75.41
	Uaccess	69.11	72.16	74.20
Alpha = 0.95 Convergence = 0.01	Access	77.32	78.18	80.38
	Access2	79.60	80.60	82.22
	Nasa	73.83	77.26	80.97
	NasaAug	73.26	77.50	80.31
	Retail	75.36	77.33	79.54
	Uaccess	69.28	73.06	77.58

Table 3: Percentage Accuracy Analysis of Label Propagation method

Parameter	Dataset	10%	20%	30%
Alpha = 0.043 Convergence = 0.01	Access	1.91	2.47	3.16
	Access2	1.98	2.22	3.03
	Nasa	2.70	3.19	3.82
	NasaAug	2.02	2.69	3.48
	Retail	2.22	3.11	4.09
	Uaccess	2.01	2.89	3.45
Alpha = 0.043 Convergence = 0.001	Access	1.92	2.50	3.16
	Access2	2.02	2.25	3.03
	Nasa	2.74	3.20	3.83
	NasaAug	2.01	2.69	3.48
	Retail	2.24	3.12	4.10
	Uaccess	2.02	2.89	3.46
Alpha = 0.95 Convergence = 0.01	Access	1.87	2.34	3.05
	Access2	1.38	2.04	3.02
	Nasa	4.20	5.58	6.53
	NasaAug	3.17	4.64	5.63
	Retail	2.05	3.09	4.08
	Uaccess	1.82	2.82	3.82

Table 4: Lift Analysis of Label Propagation method

Appendix B: t-Test comparison between global approach and localized approaches

F-Measure	*Global*	*Sub-Graph*
Mean	0.747084891	0.910067168
Variance	0.000742102	0.000916397
Observations	30	30
Hypothesized Mean Difference	0	
df	57	
	-	
t Stat	21.92014505	
P(T<=t) one-tail	9.36172E-30	
t Critical one-tail	1.672028889	
P(T<=t) two-tail	1.87234E-29	
t Critical two-tail	2.002465444	

Table 1: F-Measure t-Test for Access2 dataset

Accuracy	*Global*	*Sub-Graph*
Mean	0.943527718	0.981771058
Variance	5.32106E-05	4.38355E-05
Observations	30	30
Hypothesized Mean Difference	0	
Df	57	
	-	
t Stat	21.26313888	
P(T<=t) one-tail	4.40515E-29	
t Critical one-tail	1.672028889	
P(T<=t) two-tail	8.8103E-29	
t Critical two-tail	2.002465444	

Table 2: Percentage Accuracy t-Test for Access2 dataset

Lift	*Global*	*Sub-Graph*
Mean	8.298536267	9.458770264
Variance	0.028457047	0.054714306
Observations	30	30
Hypothesized Mean Difference	0	
df	53	
	-	
t Stat	22.03531842	
P(T<=t) one-tail	1.18596E-28	
t Critical one-tail	1.674116237	
P(T<=t) two-tail	2.37192E-28	
t Critical two-tail	2.005745949	

Table 3: Lift t-Test for Access2 dataset

F-Measure	*Global*	*Sub-Graph*
Mean	0.725168448	0.740129626
Variance	0.002832758	0.00155679
Observations	30	30
Hypothesized Mean Difference	0	
Df	53	
	-	
t Stat	1.236848504	
P(T<=t) one-tail	0.110797983	
t Critical one-tail	1.674116237	
P(T<=t) two-tail	0.221595967	
t Critical two-tail	2.005745949	

Table 4: F-Measure t-Test for Nasa dataset

Accuracy	*Global*	*Sub-Graph*
Mean	0.935703002	0.939770932
Variance	0.000201177	0.000105538
Observations	30	30
Hypothesized Mean Difference	0	
Df	53	
	-	
t Stat	1.272233625	
P(T<=t) one-tail	0.104421877	
t Critical one-tail	1.674116237	
P(T<=t) two-tail	0.208843754	
t Critical two-tail	2.005745949	

Table 5: Percentage Accuracy t-Test for Nasa dataset

Lift	*Global*	*Sub-Graph*
Mean	8.325207623	8.467825002
Variance	0.306290368	0.180720797
Observations	30	30
Hypothesized Mean Difference	0	
df	54	
	-	
t Stat	1.119344116	
P(T<=t) one-tail	0.133973836	
t Critical one-tail	1.673564907	
P(T<=t) two-tail	0.267947672	
t Critical two-tail	2.004879275	

Table 6: Lift t-Test for Nasa dataset

F-Measure	*Global*	*Sub-Graph*
Mean	0.730307993	0.819507211
Variance	0.001992352	0.001999964
Observations	30	30
Hypothesized Mean Difference	0	
df	58	
t Stat	-7.732309458	
P(T<=t) one-tail	8.61421E-11	
t Critical one-tail	1.671552763	
P(T<=t) two-tail	1.72284E-10	
t Critical two-tail	2.001717468	

Table 7: F-Measure t-Test for NasaAug dataset

Accuracy	*Global*	*Sub-Graph*
Mean	0.928530021	0.960289855
Variance	0.000250175	0.000116267
Observations	30	30
Hypothesized Mean Difference	0	
df	51	
t Stat	-9.08733549	
P(T<=t) one-tail	1.51762E-12	
t Critical one-tail	1.675284951	
P(T<=t) two-tail	3.03524E-12	
t Critical two-tail	2.007583728	

Table 8: Percentage Accuracy t-Test for NasaAug dataset

Lift	*Global*	*Sub-Graph*
Mean	9.434641003	9.095577432
Variance	0.053723598	0.238644695
Observations	30	30
Hypothesized Mean Difference	0	
df	41	
t Stat	3.434603621	
P(T<=t) one-tail	0.000685355	
t Critical one-tail	1.682878003	
P(T<=t) two-tail	0.00137071	
t Critical two-tail	2.019540948	

Table 9: Lift t-Test for NasaAug dataset

F-Measure	*Global*	*Sub-Graph*
Mean	0.724613238	0.684829696
Variance	0.001719089	0.001776249
Observations	30	30
Hypothesized Mean Difference	0	
Df	58	
t Stat	3.685695318	
P(T<=t) one-tail	0.000251808	
t Critical one-tail	1.671552763	
P(T<=t) two-tail	0.000503617	
t Critical two-tail	2.001717468	

Table 10: F-Measure t-Test for Retail dataset

Accuracy	*Global*	*Sub-Graph*
Mean	0.940757576	0.931515152
Variance	8.66185E-05	9.17324E-05
Observations	30	30
Hypothesized Mean Difference	0	
Df	58	
t Stat	3.790607918	
P(T<=t) one-tail	0.000180009	
t Critical one-tail	1.671552763	
P(T<=t) two-tail	0.000360019	
t Critical two-tail	2.001717468	

Table 11: Percentage Accuracy t-Test for Retail dataset

Lift	*Global*	*Sub-Graph*
Mean	7.811468971	7.431264729
Variance	0.191710745	0.194586028
Observations	30	30
Hypothesized Mean Difference	0	
df	58	
t Stat	3.350557268	
P(T<=t) one-tail	0.000711573	
t Critical one-tail	1.671552763	
P(T<=t) two-tail	0.001423145	
t Critical two-tail	2.001717468	

Table 12: Lift t-Test for Retail dataset

F-Measure	*Global*	*Sub-Graph*
Mean	0.526132097	0.807111124
Variance	0.001664184	0.002546426
Observations	30	30
Hypothesized Mean Difference	0	
Df	56	
t Stat	-23.71712421	
P(T<=t) one-tail	3.4392E-31	
t Critical one-tail	1.672522304	
P(T<=t) two-tail	6.8784E-31	
t Critical two-tail	2.003240704	

Table 13: F-Measure t-Test for Uaccess dataset

Accuracy	*Global*	*Sub-Graph*
Mean	0.84361834	0.957249071
Variance	0.000570829	0.00021611
Observations	30	30
Hypothesized Mean Difference	0	
Df	48	
t Stat	-22.18635462	
P(T<=t) one-tail	3.50919E-27	
t Critical one-tail	1.677224197	
P(T<=t) two-tail	7.01838E-27	
t Critical two-tail	2.010634722	

Table 14: Percentage Accuracy t-Test for Uaccess dataset

Lift	*Global*	*Sub-Graph*
Mean	8.262142857	9.019069264
Variance	0.01714955	0.160939232
Observations	30	30
Hypothesized Mean Difference	0	
df	35	
t Stat	-9.824173031	
P(T<=t) one-tail	6.72917E-12	
t Critical one-tail	1.68957244	
P(T<=t) two-tail	1.34583E-11	
t Critical two-tail	2.030107915	

Table 15: Lift t-Test for Uaccess dataset

Appendix C: Proof for the time complexity comparison between global approach and localized approach

Given N = total number of items in global graph. Suppose the global graph is partitioned in to m sub-graphs, thus

Now $N = S_1 + S_2 + S_3 +, \dots, +S_m$ where $S_1, S_2, S_3, \dots, S_m$ are the number of items in each sub-graph S_i, thus:

$$N^3 = (S_1 + S_2 + S_3 +, \dots, +S_m)^3$$

Since for any given real numbers a and b we have:

$$(a + b)^3 = a^3 + 3a^2b + 3ab^2 + b^3,$$

It follows from mathematical induction that:

$N^3 = (S_1 + S_2 + S_3 +, \dots, +S_m)^3 > S_1^3 + S_2^3 + S_3^3 +, \dots, +\mathrm{m} \times N^2 + \mathrm{T}$ where T is a positive real number.

Therefore it follows that:

$$O(N^3) > O(S_1^3) + O(S_2^3) + O(S_3^3) +, \dots, O(S_m^3) + O(N^2)$$

Appendix D: Performance results of strategy1 and strategy2 data pre-processing for World Cup 1998 dataset

	Method	10%	20%	30%
Strategy1	Global-Simple Random	39.22	39.78	42.19
	Global-Stratified Random	39.50	41.58	44.39
	Sub-graph Stratified Random	42.03	43.08	44.86
Strategy2	Global-Simple Random	36.94	40.89	44.00
	Global-Stratified Random	43.72	48.00	52.22
	Sub-graph Stratified Random	50.44	54.44	63.33

Table 1: Precision Analysis

	Method	10%	20%	30%
Strategy1	Global-Simple Random	51.43	51.43	54.51
	Global-Stratified Random	53.47	54.76	57.08
	Sub-graph Stratified Random	58.85	59.95	60.94
Strategy2	Global-Simple Random	26.26	30.01	32.63
	Global-Stratified Random	32.37	37.72	42.64
	Sub-graph Stratified Random	37.51	41.77	49.04

Table 2: Recall Analysis

	Method	10%	20%	30%
Strategy1	Global-Simple Random	82.11	82.42	83.49
	Global-Stratified Random	82.20	82.54	83.57
	Sub-graph Stratified Random	82.76	83.08	83.62
Strategy2	Global-Simple Random	81.59	82.96	83.94
	Global-Stratified Random	83.47	85.56	87.20
	Sub-graph Stratified Random	85.56	86.98	89.26

Table 3: Percentage Accuracy Analysis

	Method	10%	20%	30%
Strategy1	Global-Simple Random	3.16	3.28	3.47
	Global-Stratified Random	3.19	3.36	3.57
	Sub-graph Stratified Random	3.40	3.41	3.59
Strategy2	Global-Simple Random	3.33	3.68	3.96
	Global-Stratified Random	3.94	4.32	4.70
	Sub-graph Stratified Random	4.54	4.90	5.70

Table 4: Lift Analysis