SVM-Based Association Rules for Knowledge Discovery and Classification

Ali Anaissi
Center of Quantum Computation and Intelligent Systems (QCIS), Faculty of Engineering and Information Technology (FEIT), University of Technology Sydney (UTS)
Broadway NSW 2007, Australia
E-mail: Ali.Anaissi@uts.edu.au

Madhu Goyal
Center of Quantum Computation and Intelligent Systems (QCIS), Faculty of Engineering and Information Technology (FEIT), University of Technology Sydney (UTS)
Broadway NSW 2007, Australia
E-mail: Madhu.Goyal-2@uts.edu.au

***Abstract*—Improving analysis of market basket data requires the development of approaches that lead to recommendation systems that are tailored to specifically benefit grocery chain. The main purpose of that is to find relationships existing among the sales of the products that can help retailer identify new opportunities for cross-selling their products to customers. This paper aims to discover knowledge patterns hidden in large data set that can yield more understanding to the data holders and identify new opportunities for imperative tasks including strategic planning and decision making. This paper delivers a strategy for the implementation of a systematic analysis framework built on the established principles used in data mining and machine learning. The primary goal of that is to form the foundation of what we envisage will be a new recommendation system in the market. Uniquely, our strategy seeks to implement data mining tools that will allow the analyst to interact with the data and address business questions such as promotions advertisement. We employ Apriori algorithm and support vector machine to implement our recommendation systems. Experiments are done using a real market dataset and the 0.632+ bootstrap method is used here in order to evaluate our framework. The obtained results suggest that the proposed framework will be able to generate benefits for grocery chain using a real-world grocery store data.**

Keywords- SVM; data mining; machine learning; Apriori algorithm; association rules.

I. Introduction

It has become apparent that improved analysis of market basket data requires the development of approaches that lead to recommendation systems that are tailored to benefit grocery chain particularly. Hence, basket market data needs to be systematically analyzed such that deriving the association rules and presented in a manner such that it will provide 'actionable knowledge' for the market's analyst. The mining of data collection has received a lot of interests in several domains such as market, financial and biomedical [**?**] [**?**] [**?**]. The aim of that is to discover knowledge patterns hidden in large data set that can yield more understanding to the data holders and identify new opportunities for imperative tasks including strategic planning and decision making. One methodology of mining complex dataset is determining the association rules which mainly used in the analysis of the market basket data [**?**] [**?**]. The main purpose of that is to find connections existing among the items that can assist retailer with distinguishing new open doors for cross-offering their items to clients. This area of data mining i.e. association rules has received a great deal of interest in the field of market basket analysis. Figuring out what items clients are liable to purchase together could be extremely helpful for products arrangement and promotion [**?**]. The rationale behind that is to find relationships between the frequent items in the presence baskets to generate association rules from these items. The association rules problem can be defined as per the following, let $I = i_1, i_2, i_3 \dots i_n$ is a set of items, and D is the dataset containing all the transactions. Each record in this dataset represents one transaction T having a set of objects such that $T \subseteq I$. Let $A,\ B$ be a set of items such that $A,\ B \subseteq I$. A suggested association rule can be written in the structure $A \Rightarrow B$, where $A \subset I, B \subset I, A \cap B = \phi$ [**?**]. A real typical example of that can be represented in the following statement

$$\{Peanutbutter,\ Jelly\} \Rightarrow \{Bread\}$$

This simple association rule says that Bread is likely to be bought if peanut butter and jelly are purchased. The items surrounded by brackets are called itemsets. With small datasets, human beings are able to find interesting connections in small datasets and can build association rules. However, this task becomes a significant challenge in the case of extremely complex and large transactional datasets with a high number of features or products. These difficulties are compounded as the number of itemsets grows exponentially with the number of products. For example, you have k items that can appear or not in a set, and then you will have 2^k possible itemsets that must be searched to find the association rules. Many research papers have been proposed to solve this problem by identifying new heuristic algorithms that can reduce the number of itemsets to search. Apriori algorithm is a dominant association rules mining techniques proposed by R. Agrawal and R. Srikant [**?**]. It is the first association rules mining algorithm that spearheaded the utilization of bolster based pruning

to control the exponential development of searching itemsets deliberately. It divides the procedure of mining association rules into two steps: The first step iteratively finds all itemsets with supports are greater than a threshold value defined by the user. These itemsets are known as the frequent itemsets. The second step uses the obtained frequent itemsets to build association rules that comply a user-defined confidence value. The derived association rules are evaluated based on two statistical measures, Support and Confidence, to see whether they deliver benefits or not. These two parameters, Support and Confidence, profoundly affect the production of association rules as they are used to limit the number of derived rules. Support is how frequently the itemset occurs in the data, and confidence is the measurement of accuracy. Table **??** list five transactions to illustrate the functionality of these two parameters. The rules in this table have the following support and confidence. $\{A \Rightarrow B\}$ has 66% confidence, with 40% support, $\{B \Rightarrow C\}$ has 75% confidence, with 60% support, and $\{AB \Rightarrow C\}$ has 50% confidence, with 20% support.

The derived metric demonstrates how frequently an itemset shows up in the data. Hence, if we realize that $\{A\}$ doesnt comply a desired support threshold, it is unrealistic of having the itemset $\{A, B\}$ or any itemset contains $\{A\}$ even though $\{A, B\}$ is regular. Otherwise, both $\{A\}$ and $\{B\}$ must be regular. Consequently, the procedure of producing rules is done in two stages:

- Determining all itemsets that comply a minimum support threshold.
- Deriving rules using the determined itemsets that comply a minimum confidence threshold.

The rest of this paper is organized as follows. Section II introduces the related work of association rules. Section III presents the methods and algorithms used in this work to achieve the desired results. Algorithms of clusters generation, Apriori and SVM are discussed in detail showing that how the association rules are generated in Apriori algorithm and how the classification model is built in SVM. Section IV presents the experiments and discuses the obtained results and calculates the accuracy of classification performance. Section V draws a conclusion about the methods we applied and the results we achieved by our proposed framework.

II. Related Work

Association rule mining is an important task and a key issue in knowledge discovery and data mining [**?**] [**?**] [**?**]. Tremendous research has been established in data mining such as correlation mining [**?**], associative classification [**?**] [**?**], and frequent pattern-based clustering [**?**] [**?**]. It has proven to be quite necessary for handling product layout based business problems, such as goods promotion strategy and correlation product recommendation. For example, association rule mining is widely employed in retail industry to discover interesting association rules to help with better decision making. The Apriori algorithm was the first algorithm proposed for mining association rule that uses the support based pruning to control the exponential growth of candidate item sets systematically [**?**]. It uses a breadth-first search strategy to counting the support of item sets and uses a candidate generation function that exploits the downward closure property of support. Apriori uses a "bottom-up" approach, where frequent subsets are extended one item at a time, and groups of candidates have experimented with the data. Han et al. introduced new algorithm called FP-tree [**?**], which is the n order of magnitude faster than the Apriori algorithm as it avoids the candidate generation process and fewer passes over the data base. It uses a model fragment growth method to avoid the costly process of candidate generation and testing used by Apriori. FP-tree utilizes a divide-and-conquer approach as follows. First, it compacts the database representing frequent items into a frequent-pattern tree or FP-tree, which preserves the item sets linking information. It then divides the compacted database into a set of restricted databases. Each associated with one frequent item and mines each such database separately. Rapid Association Rule Mining (RARM) proposed in [**?**] is claimed to be much faster than FP tree algorithm with the use of the tree structure to represent the original database and avoids the candidate generation process. But these traditional associations rule mining algorithms or frameworks produce many redundant rules. Therefore, there is the requirement of performance analysis or an investigation of association rules generation by some novel approaches. Association rule mining can be integrated with SVM [**?**] [**?**] to take advantage of knowledge represented by associated rules and use the power of SVM algorithm to create an efficient and accurate classifier model.

III. Methods

This paper delivers a strategy for the implementation of a systematic analysis framework built on the established principles used in data mining and machine learning. The aim of that is to form the foundation of what we envisage will be a new recommendation systems on the market. Uniquely, our strategy seeks to implement data mining tools that will allow the analyst to interact with the data and address business questions such as promotions advertisement. Furthermore, to bolster the recommendation systems concept, 'participatory design' will be an essential element in how our project will be managed, thus ensuring what is developed will have actual market utility and application. The concept will be tested on whether it will be able to generate benefit for grocery chain using a real-world grocery store data. This paper proposes an SVM-Based Association Rules (SVM-BAR) framework that is composed of three main steps. Step one concerns generating the association rules, step two concerns creating clusters and step three concerns building the SVM model classifier based on the generated clusters and association rules. Detailed descriptions of the training process in the proposed framework are given in the following.

The first step of SVM-BAR employs Apriori algorithm with the aim to extract the association rules form the training dataset according to the defined threshold values of the parameters Support and Confidence. Multiple executions of Apriori al-

TABLE I
ILLUSTRATION OF PARAMETER SUPPORT AND CONFIDENCE

TID	Items	$support = \frac{Occurrence}{total\ support}$	Given $X \Rightarrow Y$, $Confidence = \frac{Occurrence\{Y\}}{ccurrence\{X\}}$
1	ABC	Total support = 5	Confidence$\{A \Rightarrow B\}$ = 2/3 = 66%
2	ABD	Support AB = 2/5 = 40%	Confidence$\{B \Rightarrow C\}$ = 3/4 = 75%
3	BC	Support BC = 3/5 = 60%	Confidence$\{AB \Rightarrow C\}$ = 1/2 =50%
4	AC	Support ABC = 1/5 = 20%	
5	BCD		

gorithm is taken at this step each with different threshold values of the parameters that will generate a different number of clusters. Groups are created based on the customers that bought similar products in the generated rules. These groups of customers form a cluster and are subsequently removed from the original datasets. This process is recursively repeated until the generated rules don't support the minimum defined support threshold. Each iteration of this process will create a new cluster according to the customers and products. Following this approach, the clusters generated lastly might be weak clusters and doesn't add value to the association rules in contrast to the initial clusters that might be more informative. Consequently, we have measured the quality of each group based on the average distance between its points and based on the confidence value generates in the current association rules. The clusters that produce maximum distance between its points, as well as maximum confidence, will be selected as the appropriate cluster. The algorithm is illustrated below: The algorithm receives two parameters, data, all transactions

Algorithm 1 Clustering (Data , support)

Require: $ctr = 1$
Require: $Clusters = 0$
Require: $DB = data$
Require: $sup = support$
  while *not termination condition* **do**
    $Rules = Apriori(sup)$
    $cluster[ctr] = transactions\ containing\ products\ in\ Rules$
    $DB = DB \setminus cluster[ctr]$
    $Clusters = Cluster \cup cluster[ctr]$
  end while
  $return\ Clusters$

in the dataset, and support, is the minimum acceptable support. The algorithm starts with the entire set of transactions in the dataset, and at every iteration, we generate association rules from the original training dataset. The generated association rules are stored in $Rules$. The cluster is then created using the rules we stored in $Rules$ and added them to the variable $Clusters$. Subsequently, we eliminate the transactions appeared in the cluster, and we generate other association rules but restricted to the remaining set of the transactions. We repeat this process iteratively until we reach a threshold value of $confidence$, where $confidence$ is a user defined variable. As the results of the experiments will show, the sequence of clusters has the property that the first clusters that are built are of good quality while clusters that are generated later may become less informative. The next step involves developing a classification model to classify new customers based on the derived grouped data. In the current study, we use support vector machine (SVM) [**?**] classifiers as the base learning model. The following sections describe the two algorithms, Apriori and SVM.

A. Apriori

Apriori algorithm is initially applied on the data in order to generate association rules. This will be done through several executions each with different values of the parameter Support. Once the association rules have been generated, the rules with the highest confidence are selected. Each execution will generate different association rules which will subsequently clustered into categories based on the item appears in the right hand side of the association rule. The description of Apriori Pseudo code algorithm is presented in the following section.

Algorithm 2 Apriori

Input:
DB: transaction database;
sup: the minimum support threshold
Output: frequent itemsets
Description:

1: L_1= find_frequent_1-itemsets(DB);
2: **for** (k=2; $L_{k-1} = \varphi; k++$) {
3: C_k= Apriori_gen(L_{k-1});
4: **for each** transaction $t \in DB$ {
5: C_t = subset(C_k, t);
6: **for each** candidate $c \in C_t$
7: $c.count++$;
8: }
9: $L_k = \{c \in C_k | c.count \geq sup\}$
10: }
11: return $L = \bigcup_k L_k$;
12: Procedure Apriori gen(L_{k-1}: frequent$(k-1)$-itemsets)

B. Support Vector Machine

Support vector machine algorithm classifier is also used in this paper in order to evaluate the classification accuracy for a new transaction not exists in the data used in Apriori algorithm. SVM is proposed to retrieve the most similar association rules from the knowledge database that best match a new query transaction. Support vector machine is a classifier

using a decision boundary to separate two classes defined by solving a quadratic optimization problem. SVM finds an optimal solution that maximizes the distance between the hyperplane and the most critical training samples. The decision boundary is then specified by a subset of critical training samples named support vectors that lie on the edge. SVM extends to multi-class classification using several methods [**?**] [**?**] [**?**]. SVM has been extensively and effectively used in many applications because its design is well suited complex large datasets. SVM is considered as one of the best performers for a number of classification tasks ranging from text to microarray data [**?**] [**?**] [**?**].

Suppose we have two features, $x1$ and $x2$, and we want to classify all these elements appeared in Fig **??**. We can see we have the class red and the class black. The goal of the SVM is to design a hyperplane that classifies all training vectors in two categories. We can define the black line as the hyperplane that classifies all the training vectors in the two classes. We can have multiple hyperplanes that can classify all the instances correctly in this feature set. However, the best choice will be the hyperplane that leaves the maximum margin from both classes. The margin is that distance between the hyperplane and the closest elements from this hyperplane. This hyperplane is defined by one equation

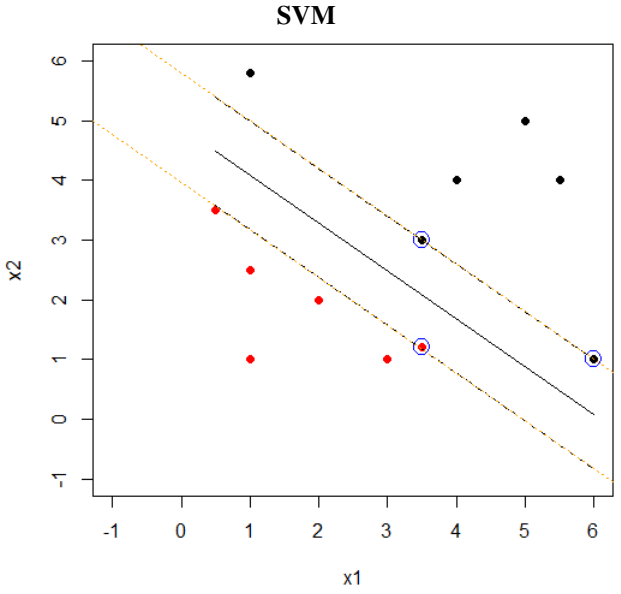

Fig. 1. SVM applied onto two features, $x1$ and $x2$

$$g(\vec{x}) = \vec{w}^T \vec{x} + b \tag{1}$$

This equation generates values greater than one for all the input vectors which belongs to the class number one. And also will deliver values less than one for the input vectors belongs to the class number two.

$$g(\vec{x}) \geq 1 \qquad \forall \vec{x} \in Class1 \tag{2}$$

$$g(\vec{x}) \leq -1 \qquad \forall \vec{x} \in Class2 \tag{3}$$

From the geometry we know that the distance between a point and a hyperplane is computed by this equation

$$z = \frac{|g(\vec{x})|}{||\vec{w}||} = \frac{1}{||\vec{w}||} \tag{4}$$

So the total margin which is composed by this distance will be computed by this equation.

$$\frac{1}{||\vec{w}||} + \frac{1}{||\vec{w}||} = \frac{2}{||\vec{w}||} \tag{5}$$

And the aim is that minimising this term will maximise the separability. When we maximise this weight vector we will have the biggest margin that will split the two classes. To minimise this vector $\vec{w}$ is a non-linear task optimisation which can be solved by this condition Karush-Kuhn-Tucker (KKT) using The Lagrange multipliers α_i. The main equation states that the value of omega will be the solution of this sum.

- Minimize

$$L_P = \frac{\|w\|^2}{2} - \sum_{i=1}^{l} \alpha_i y_i (\mathbf{x}_i \mathbf{w} + b) + \sum_{i=1}^{l} \alpha_i \tag{6}$$

- Convex quadratic programming problem with the dual: maximize

$$L_D = \sum_{i=1}^{l} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{l} \alpha_i \alpha_j y_i y_j (\mathbf{x}_i \mathbf{x}_j) \tag{7}$$

Let us assume that we have n labeled examples $(x_1, y_1), \ldots, (x_n, y_n)$ with labels $y_i \in \{1, -1\}$. We want to find the hyperplane $< w, x > +b = 0$ (i.e. with parameters (w, b)) satisfying the followin three conditions:

1) The scale of (w, b) is fixed so that the plane is in canonical position w.r.t. $\{x_1, \ldots, x_n\}$. i.e.,

$$\min_{i \leq n} | < w, x_i > +b| = 1$$

2) The plane with parameters (w, b) separates the $+1$'s from the the -1's. i.e.,

$$y_i(< w, x_i > +b) \geq 0 \text{ for all } i \leq n$$

3) The plane has maximum margin $\rho = 1/|w|$. i.e., minimum $|w|^2$.

Of course, there may not be a separating plane for the observed data. Let us assume, for the time being, that the data is in fact linearly separable. Apparently 1 and 2 combined into just one condition:

$$y_i(< w, x_i > +b) \geq 1 \text{ for all } i \leq n.$$

Thus, we want to solve the following optimization problem,

$$\text{minimize } \frac{1}{2}|w|^2$$

overall $w \in R^d$ and $b \in R$ subject to,

$$y_i(< w, x_i > +b) - 1 \geq 0 \text{ for all } i \leq n.$$

This is a very simple quadratic programming problem. There are readily available algorithms of complexity $O(n^3)$ that can be used for solving this problem.

IV. EXPERIMENTS

The R packages `arules` contains the algorithm Apriori, and `e1071` contains the algorithm SVM are used in this paper to implement our framework. We have validated our framework approach on a real-world dataset generated from actual shop sales who have over 5,000 products. This dataset is the focus of our study for ensuring what is developed will have actual market utility and application. The company has a large number of customers and daily transactions. Some transactions have more than 50 products each; other has less than ten products. Thus, we have decomposed the dataset into two subsets. The first one (D1) contains data for 900 customers who bought over 50 products, and the second one (D2) contains data for 600 customers who bought less than 50 products.

We have initially applied SVM-Based Association Rules (SVM-BAR) framework onto the data set D1. The first step in our framework is to use Apriori algorithm to find hidden rules in the transactions. These rules will be categorized subsequently into different clusters based on the products appeared on the left-hand side of the item set.

The result of this experiment produces a set of 10 clusters ranked in decreasing order based on the average distance between its points and based on the confidence value generated in its association rules. Groups with low confidence values are removed, and the points are allocated to appropriate cluster determined based on the classification accuracy of these points in the next step. Means that this transaction will be added to the test data points. The next experiment is to evaluate the classification accuracy of the SVM model using the test activities. The AUC accuracy is used here to estimate the classification performance of our model. The reason of that is because the number of points in each generated cluster is not evenly distributed among the clusters, so it makes the data points exist in an imbalanced form. As a result, classification will lean towards the clusters having the majority data points. It also makes the samples in the minority group difficult to be fully recognized. It results in unsatisfactory classification performance.

Accordingly, SVM is trained on balanced clusters using the training datasets employed in the Apriori algorithm. We have applied down sampling techniques that aim to alter the distribution of the clusters toward more balanced groups. Down sampling is a technique used to remove some observations in such way from the majority class. It aims to attain the sample number of the majority class as in the minority class. This technique has been extensively used for handling the problem of class imbalanced datasets [**?**] [**?**] [**?**].

SHRINK, which is an algorithm proposed by Kubat et al. (1997) [10], is used for the down sampling technique by reducing the number of sample of the majority class. The evaluation of the test dataset by SVM trained on 100 bootstrap samples using the 0.632+ bootstrap method gives an average value of AUC equal to 0.91. This procedure is similarly applied onto the second dataset (D2). Eight clusters are generated from the first step of our framework which applies Apriori algorithm with our clustering technique. We have also trained SVM on the derived clusters considering the imbalanced problem which was handled by applying Shrink algorithm on the training dataset.

TABLE II
EXPERIMENTS RESULTS ON THE FIRST DATASET D1

Clusters	Number of customers	Confidence
1	52	85%
2	40	83%
3	74	65%
4	64	50%
5	105	48%
6	83	52%
7	87	45%
8	110	43%

TABLE III
EXPERIMENTS RESULTS ON THE SECOND DATASET D2

Clusters	Number of customers	Confidence
1	48	93%
2	56	88%
3	74	78%
4	123	45%
5	205	38%
6	83	52%

The classification performance of the test dataset by SVM trained on 100 bootstrap samples using the 0.632+ bootstrap method gives an average value of AUC accuracy equal to 0.96.

V. CONCLUSION

This paper introduces a simple SVM-BAR framework for deriving association rules, mining clusters and classification for large datasets of transaction contains information about the purchased products by the customers. This framework generates a list of association rules that subsequently grouped into different categories based on the product outcome of the rules. SVM used to test whether this cluster can capture the variation based on the different type of rules. According to the feedback received by the clients, this framework provides an actionable knowledge for the market's analyst. A significant classification accuracy is achieved by SVM, which allows the client to target customer with their needs and proposing useful promotions. Having support value equal to 25%, Apriori results in a significant clusters in the data with 91 to 96% classification results using SVM classifier.

Our future work is to follow our constructionist data analysis approach to thoroughly assess a range of market datasets.

ACKNOWLEDGMENT

The authors thank FEIT for supporting this research work through the FEIT Industry Grant 2015.