Feature Selection, Association Rules Network and Theory Building

Sanjay Chawla SANJAY.CHAWLA@SYDNEY.EDU.AU
School of IT, University of Sydney
NSW 2006, Australia

Editor: Huan Liu, Hiroshi Motoda, Rudy Setiono, and Zheng Zhao

Abstract

As the size and dimensionality of data sets increase, the task of feature selection has become increasingly important. In this paper we demonstrate how association rules can be used to build a network of features, which we refer to as an association rules network, to extract features from large data sets. Association rules network can play a fundamental role in *theory building* - which is a task common to all data sciences- statistics, machine learning and data mining.

The process of carrying out research is undergoing a dramatic shift in the twenty first century. The cause of the shift is due to the preponderance of data available in all almost all research disciplines. From anthropology to zoology, manufacturing to surveillance, all domains are witnessing an explosion of data. The availability of massive and cheap data has opened up the possibility of carrying out data-driven research and data mining is the discipline which provides tools and techniques for carrying out this endeavour.

However much of these vast repositories of data generated are observational as opposed to experimental. Observational data is undirected and is often collected without any specific task in mind. For example, web servers generate a log of client activity. The web log can then be used for a myriad of tasks ranging from tracking search engine spiders to personalization of web sites. Experimental data, on the other hand, is directed and is generated to test a specific hypothesis. For example, to test the efficacy of a new drug, randomized trials are conducted and specific data is collected to answer very specific questions.

1. Feature Selection and Experimental Data

In order to appreciate the role of feature selection we first have to understand the role of experimental data in a scientific discovery process.

Taking a reductionist viewpoint, much of scientific discovery reduces to identifying relationship(s) between variables in a domain. For example, Einstein postulated that the relationship between energy and mass is governed by the equation $e = mc^2$. In order to validate the relationship, scientist will carry out experiments to test if the relationship is indeed true. The resulting data is called experimental data.

Scientist often also postulate relationship between variables which are not necessarily governed by a mathematical equation. For example, research has shown that there is a smoking is a leading cause of lung cancer. Trials are conducted to test the validity of the relationship between the variable smoking and cancer. Experimental data does not have to be large and because there is an underlying theory which leads to an experiment the number of variables is also typically small. Thus feature selection, or the process of selecting variables which maybe related to a target variable is generally not necessary.

2. Feature Selection and Observational Data

As noted above, observational data is often collected with no specific purpose in mind. For example, a biologist maybe interested in determining which gene or a set of genes control certain physiological process P. Now modern technology allows the ability to collect the expression levels of all genes in a genome. In this setting a feature selection exercise is often carried out to filter the candidate variables which correlate with the process P. The reason that feature selection is generally hard and complex is because it is possible that complex relationships may exist between a set of features and the target P. For example two features f_1 and f_2 maybe individually correlated with P but together they may not be. Or two features may not be related with P but together they may be related.

From a structural perspective, observational data tends to be large and high dimensional and experimental data is relatively small and low dimensional. An objective of feature selection is to shape observational data in order to extract potential relationship that may exist in the data.

However the ultimate objective of feature selection in data mining is for theory building. A theory is a set of postulates which explains a phenomenon. Whether we can learn or even begin to learn a phenomenon from data is a controversial idea.

However, as data is now being collected at unprecedented rates, data mining provides new opportunities to *faciliate the learning of theories from data.* This is an ambitious task because the existence of large (and high dimensional) data is neither necessary nor sufficient to explain or postulate a theory. Still, examples abound where an unexpected manifestation in raw or transformed data triggered an explanation of the underlying phenomenon of interest. Data is known to throw up "suprises" whether these can be systematically harnessed to explain the data generating process is to be seen.

3. Association Rule Mining

Association rule mining is a data mining task to find candidate correlation patterns in large and high dimensional (but sparse) observational data (Agrawal and Srikant, 1994).

Association rules have been traditionally defined in the framework of market basket analysis. Given a set of items I and a set of transactions T consisting of subsets of I, an Association Rule is a relationship of the form $A \overset{s,c}{\to} B$ where A and B are subsets of I while s and c are the minimum support and confidence of the rule. A is called the *antecedent* and B the *consequent* of the rule. The support $\sigma(A)$ of a subset A of I is defined as the percentage of transactions which contain A and the confidence of a rule $A \to B$ is $\frac{\sigma(A \cup B)}{\sigma(A)}$. Most algorithms for association rule discovery take advantage of the anti-monotonicity property exhibited by the *support* level: If $A \subset B$ then $\sigma(A) \geq \sigma(B)$.

Our focus is to discover association rules in a more structured and dense relational table. For example suppose we are given a relation $R(A_1, A_2, \ldots, A_n)$ where the domain of

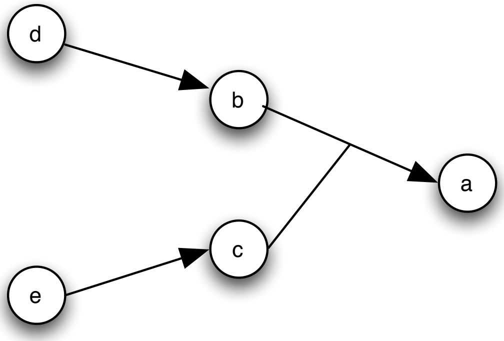


Figure 1: An example how a set of association rules with a singleton consequents can be represented as a backward hypergraph (B-graph)

A, $dom(A_i) = \{a_1, \ldots, a_{n_i}\}$, is discrete-valued. Then an *item* is an attribute-value pair $\{A_i = a\}$. The ARN will be constructed using rules of the form

$$\{A_{m_1} = a_{m_1}, \ldots, A_{m_k} = a_{m_k}\} \rightarrow \{A_j = a_j\} \text{ where } j \notin \{m_1, \ldots, m_k\}$$

4. Directed Hypergraphs

In this section we briefly describe directed hypergraphs and there relationship with association rules.

A hypergraph consists of a pair $H = (N, E)$ where N is the set of nodes and the set E is a subset of the power set 2^N. Each element of E is called a hyperedge.

In a directed hypergraph the nodes spanned by a hyperedge (e) are partitioned into the head $H(e)$ and the tail $T(e)$. The hypergraph is called *backward* if $|H(e)| = 1$ for all $e \in E$ (Gallo et al., 1993; G. Ausiello and Nanni, 1990). We will only consider association rules whose consequent are singletons and therefore the set of single consequent association rules can be identified by a B-graph (i.e., a backward hypergraph).

Example: Consider the following set of association rules:

$$\begin{aligned} r_1 : &\quad \{b, c\} \rightarrow a \\ r_2 : &\quad d \rightarrow b \\ r_3 : &\quad c \rightarrow d \\ r_4 : &\quad e \rightarrow c \end{aligned}$$

These set of rules constitutes a B-graph and can be graphically represented as shown in Figure 1.

5. Association Rules Network

In this section we formally define an Association Rules Network(ARN). Details about the ARN, the algorithm to generate them, ARN properties and examples are given in (Pandey et al., 2009).

Definition 1 *Given a set of association rules R and a frequent goal item z which appears as singleton in a consequent of a rule $r \in R$. An association rule network, $ARN(R, z)$, is a weighted B-graph such that*

1. *There is a hyperedge which corresponds to a rule r_0 whose consequent is the singleton item z.*
2. *Each hyperedge in $ARN(R, z)$ corresponds to a rule in R whose consequent is a singleton. The weight on the hyperedge is the confidence of the rule.*
3. *Any node $p \neq z$ in the ARN is not reachable from z.*

6. Association Rules Network Process

We can use ARN as a systematic tool for feature selection. The steps involved are:

1. Prepare the data for association rule mining. This entails transforming the data into transactions where each transaction is an itemset. Data where variables are continuous-valued, will have to be discretized.
2. Select and appropriate support and confidence threshold and apply an association rule mining algorithm to generate the association rules. Note that ARNs are target driven so only those association rules are of interest which are directly or indirectly related to the target node. An association rule algorithm can be customized to generate only the relevant rules. Selecting the right support and confidence threshold is non-trivial. However, since our objective is to model the norm (rather than the exception), higher values of the threshold are perhaps more suitable.
3. Build the Association Rule Network. Details are provided in (Pandey et al., 2009). This step has several exceptions which need to be handled systematically. For example, what happens if for the given support and confidence there is no association rule generated with the target node as the consequent? In which case either the support or the confidence threshold or both have to be lowered. We may also choose to select the top-k rules (by confidence) for the given target node. The advantage here is that we don't have to specify the confidence (or sometimes even the support) but now we have to specify the "top-k." Another advantage is generally we can also use the "top-k" approach to find rules in higher levels of the ARN.
4. Apply a clustering algorithm on the ARN to extract the relevant features (in the context of the target domain). The ARN is essentially a directed hypergraph. The intuition is that first level nodes have an immediate effect on the target node while higher level nodes have an indirect influence. We can use a hypergraph clustering algorithm as illustrated in (Han et al., 1997).

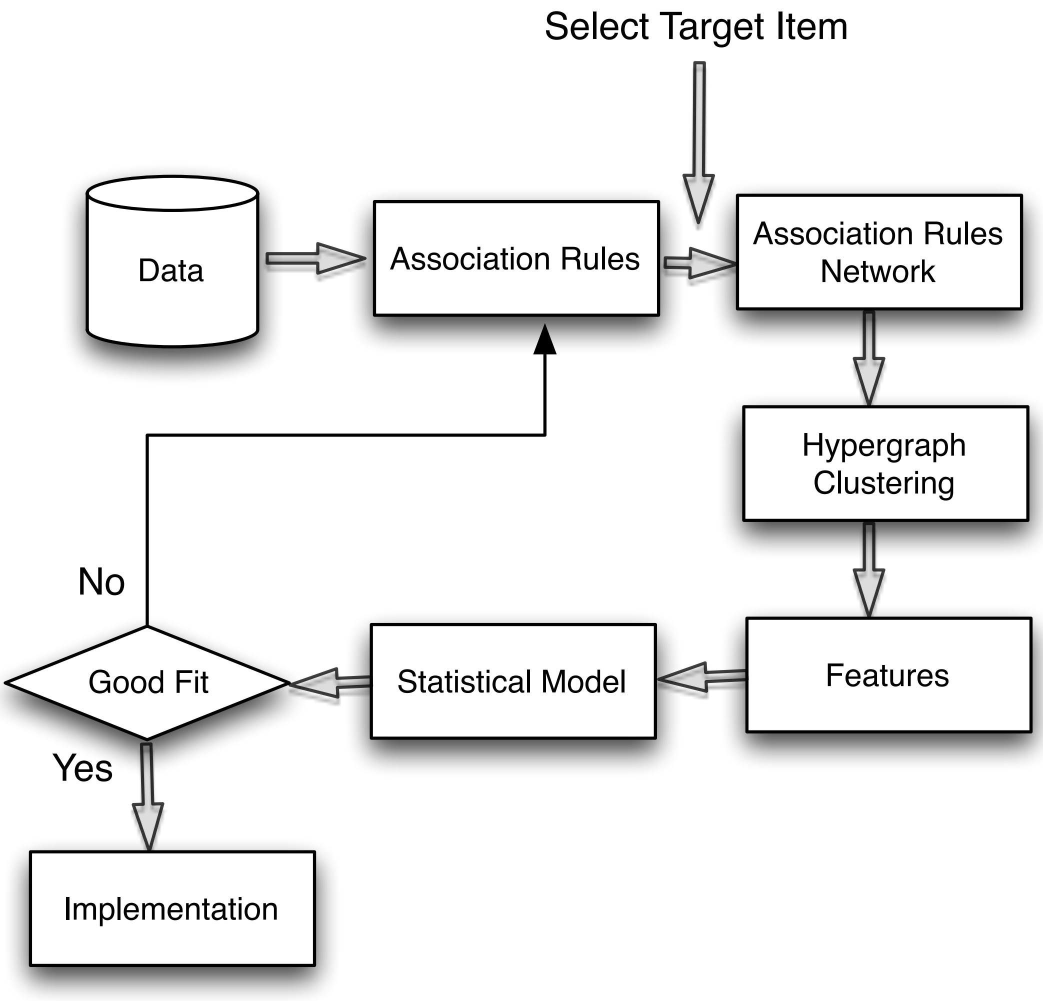
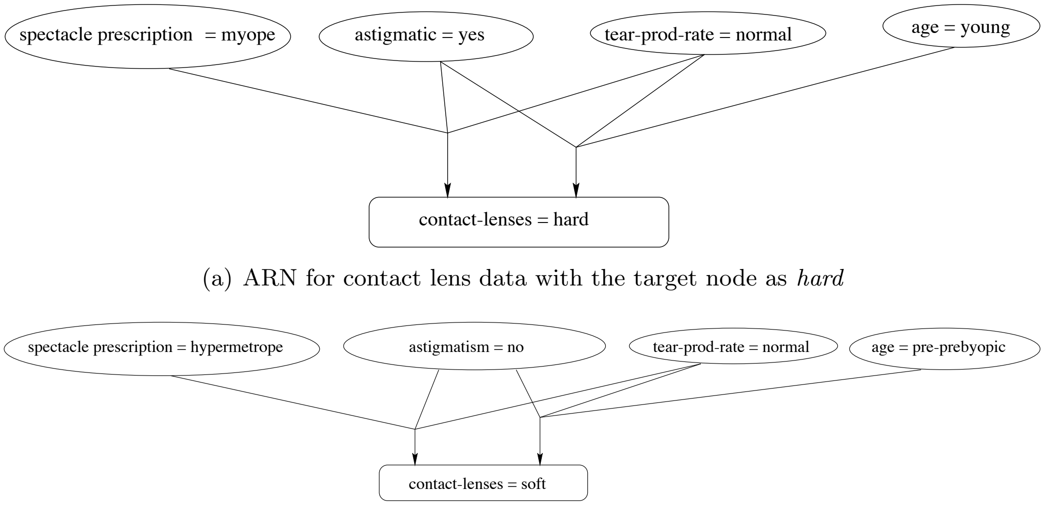


Figure 2: The ARN Process for theory building. Association rules are generated and a target item is selection which serves as the goal node of the ARN. An ARN is is a weighted B-graph. A hypergraph clustering algorithm is applied to the B-graph and each cluster represents one feature. The features are used as independent variables in a statistical model where the goal node is the dependent variable.



(a) ARN for contact lens data with the target node as *hard*

(b) ARN for contact lens data with target node as *soft*

Figure 3: ARN for the Contact Lens data. Clearly *tear production rate* does not seem like a good feature

5. The elements of the cluster are a collection of items (features) which are correlated. Choose one element of the cluster as the candidate feature. The number of clusters selected is a parameter and will require carefully calibration.

6. Build and test a statistical model (e.g., regression) to formally test the relationship between the dependent and the candidate variables.

7. ARN Examples

We give two examples of ARN and show how they can be used for feature selection.

7.1 Contact Lens Example

We use a relatively simple data set from the UCI archive (Blake and Merz, 1998) to illustrate how ARNs can be used for feature selection. The ARN for the Lenses data are shown in Figure 3. The dependent variable is whether a patient should be fitted with hard contact lenses, soft contact lenses or should not be fitted with contact lenses. There are four attributes. We built an ARN where the goal attribute is the class. Support and confidence was chosen as zero. It is clear that both ARNs (for hard and soft lenses) can be used to elicit features which are important to distiguish between the two classes.

7.2 Open Source Software Example

We have carried out an extensive analysis of the Open Source Software domain using ARN. Details can be obtained from (S.Chawla et al., 2003; Pandey et al., 2009). The OSS data was obtained to understand why certain software products available from sourceforge.net

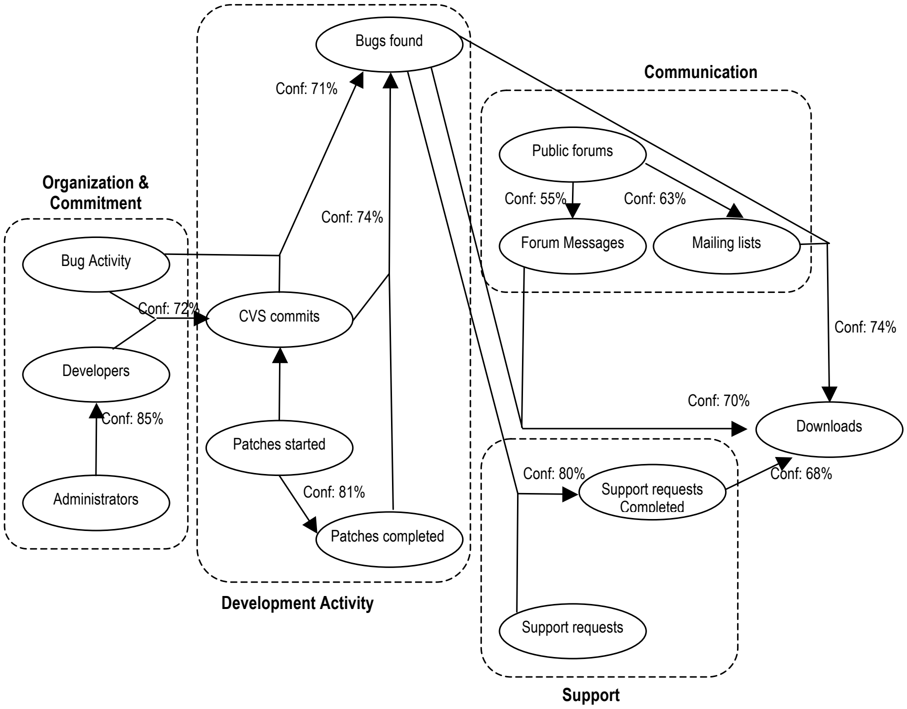


Figure 4: The ARN for Open Source data (Pandey et al. (2009)). The goal node is *Downloads=high*. After the ARN was formed, the directed hypergraph was clustered. The cluster constitute the features, and in this example, suggest meaningful features which may suggest an explanation of popular projects.

become popular. This is a cross-sectional study. Perhaps a future problem is to use ARNs for study of longtitudinal data.

8. Conclusion

In this paper we have briefly illustrated a new framework that we have introduced to elicit candidate features for theory building in a domain. Association Rules Network (ARN) are built from association rules and can be used as first step to understand the interaction amongst variables in a domain. ARNs are particular suitable in settings where the data available is observational (as opposed to experimental).

References

Rakesh Agrawal and Ramakrishnan Srikant. Fast algorithms for mining association rules. In *Proc. 20th Int. Conf. Very Large Data Bases, VLDB*, pages 487–499. Morgan Kaufmann, 12–15 1994. ISBN 1-55860-153-8.

C.L. Blake and C.J. Merz. UCI repository of machine learning databases, 1998. URL `http://www.ics.uci.edu/~mlearn/MLRepository.html`.

G.F. Italiano G. Ausiello and U. Nanni. Dynamic maintenance of directed hypergraphs. *Theoretical Computer Science*, 72(2-3):97–117, 1990.

Giorgio Gallo, Giustino Longo, and Stefano Pallottino. Directed hypergraphs and applications. *Discrete Applied Mathematics*, 42(2):177–201, 1993.

Eui-Hong Han, George Karypis, Vipin Kumar, and Bamshad Mobasher. Clustering based on association rule hypergraphs. In *Proceedings SIGMOD Workshop Research Issues on Data Mining and Knowledge Discovery(DMKD '97)*, 1997.

G. Pandey, S. Chawla, S. Poon, B. Arunasalam, and J. Davis. Association rules network: Definition and applications. *Statistical Analysis and Data Mining*, 1(4):260–279, 2009.

S.Chawla, B.Arunasalam, and J. Davis. Mining open source software(oss) data using association rules network. In *Advances in Knowledge Discovery and Data Mining, 7th Pacific-Asia Conference, PAKDD'03*, pages 461–466. Springer, 2003.