Experiences in Building a Tool for Navigating Association Rule Result Sets

Peter Fule and John F. Roddick

School of Informatics and Engineering,
Flinders University, GPO Box 2100,
Adelaide, South Australia
{peterf, roddick}@infoeng.flinders.edu.au

Abstract

Practical knowledge discovery is an iterative process. First, the experiences gained from one mining run are used to inform the parameter setting and the dataset and attribute selection for subsequent runs. Second, additional data, either incremental additions to existing datasets or the inclusion of additional attributes means that the mining process is reinvoked, perhaps numerous times. Reducing the number of iterations, improving the accuracy of parameter setting and making the results of the mining run more clearly understandable can thus significantly speed up the discovery process.

In this paper we discuss our experiences in this area and present a system that helps the user to navigate through association rule result sets in a way that makes it easier to find useful results from a large result set. We present several techniques that experience has shown us to be useful. The prototype system – IRSetNav – is discussed, which has capabilities in redundant rule reduction, subjective interestingness evaluation, item and itemset pruning, related information searching, text-based itemset and rule visualisation, hierarchy based searching and tracking changes between data sets using a knowledge base. Techniques also discussed in the paper, but not yet accommodated into IRSetNav, include input schema selection, longitudinal ruleset analysis and graphical visualisation techniques.

Keywords: Data mining, Knowledge Discovery, Association Rules, Itemsets, Navigation of Results.

1 Introduction

Most knowledge discovery frameworks view rule discovery as an iterative process. This is reflected in frameworks such as CRISP (Chapman, Kerber, Clinton, Khabaza, Reinartz & Wirth 1999) which includes components for data extraction, data preparation, data mining, visualisation and interpretation, and action. As a result, incremental improvements in the speed of data mining algorithms mean less if the mining routine needs to be invoked a large number of times during the knowledge discovery process or if the consideration of the results by the human operators is relatively long. The system described here aims to reduce the number of iterations between the data preparation and the interpretation stages by providing tools that have proven to be useful for processing the results.

The motivation for this work has come from a collaborative research effort aimed at applying association mining techniques to medical data (Roddick, Fule & Graco 2003). These experiences showed that the practical use of many algorithms was deficient and indicated that both improvements to the mining algorithms themselves as well as the ability to intelligently post-process the results were required. In response to the second of these, the IRSetNav tool was developed to navigate around large result (item and rule) sets. One of the major goals of the system is to provide the tool to knowledgeable end-users. We thus placed a high value on the construction of an intuitive user interface and rejected many ideas as they introduced inconsistency in the interpretation of what was being displayed.

The remainder of the paper is structured as follows. Section 2 provides our motivation for this work in terms of the problems commonly experienced in practical data mining situations. Section 3 looks at associated research and systems while Section 4 discusses the particular example of the IRSetNav software. Section 5 then discusses future work and concludes the paper.

2 Motivation

The organisational acceptance of data mining is commonly a trade-off between the benefits promised by the technology and the inherent risk that there may be no useful knowledge to find. In some areas, the propensity to invest in what is sometimes considered a speculative activity is low and thus the resources allocated to pilot projects is correspondingly low, despite the possible benefits. At the same time, data mining is moving from domains in which the average computing professional is able to possess at least a passing understanding of the rules produced, such as commerce, to those in which a strong relationship between the two domains is necessary, such as medicine. Our prime domain of investigation, epidemiology and population health data, is one area that fits both of these criteria and it is for this domain that the IRSetNav system was first developed, although it has subsequently found applicability in other domains.

The general problem of generating excessive results (in terms of being able to easily process and understand) is well known to the knowledge discovery community and a number of researchers are addressing this issue. A variety of solutions have been proposed which can be categorised into the following groups:

- Preselection of the data likely to produce interesting results. In practice this means not including some attributes or attribute values in the

mining process.

- The selection of alternative quality metrics to maximise the useability of the rules generated. For monotonic metrics, the *Apriori* principle (Agrawal & Srikant 1994) can be used to prune during itemset generation. These measures can be of two kinds:
  - Objective interestingness measures, which replace the use of the commonly adopted support and confidence metrics. See for example (Brin, Motwani & Silverstein 1997, Meo 2000, Freitas 1999, Hilderman & Hamilton 1999).
  - Subjective interestingness measures, which explicitly include domain knowledge from the user to aid with the mining process (Liu, Hsu, Chen & Ma 2000, Liu, Hsu & Chen 1997, Fule & Roddick 2003, Silberschatz & Tuzhilin 1996).
- Rule redundancy reduction and pruning mechanisms to eliminate useless or redundant rules. For example, the generation of closed itemsets from which all frequent itemsets can be derived without further mining (Pasquier, Bastide, Taouil & Lakhal 1999, Pei, Han & Mao 2000, Zaki & Hsiao 2002, Toivonen, Klemettinen, Ronkainen, Hatonen & Mannila 1995, Cristofor & Simovici 2002). An allied area is ruleset summarisation (Liu, Hsu & Ma 1999).
- Rule visualisation, in which rules are presented in a way that better uses the power of human image processing to aid in understanding the results. See Ceglar *et al.* (2003) for a survey of visualisation techniques for association rule mining.

In practice, preselection of data is used frequently to eliminate unwanted frequent items. For example, if we use the commonly used quality metric of support, an item occurring for a majority of objects in a dataset it is going to be present in many of the rules that are created purely because of its dominant presence. Removing such items from the input data set is a simple solution although there may be a potential loss of important information. The application of alternative measures of quality can be used which apply an expected frequency of occurrence with the observed frequency. Moreover, a frequent k-item itemset can also cause a cascade of frequent $k + n$-item itemsets due to the dominance of a set of strongly correlated items. In this case, closed itemsets can be useful in reducing this overhead.

Moving between similar data sets can also present the repetitious task of needing to remove commonly known rules. For example, in our medical framework, when working with data from the same hospital but collected for different time periods, it was not surprising that many of the same results were found – indeed, in some situations we were only interested in previously known results that had changed significantly. The concept of a knowledge base (á la (Liu & Hsu 1996)) was therefore investigated to remove already known results.

Similarly, there are cases when a comparison of two or more result sets is required to discover similarities and differences. For example, comparing the result sets for two different hospitals, or the same hospital over different time periods, could provide useful pointers to differences in procedure or catchment. To handle this situation a difference engine was developed.

Often there is a need to constrain the visualisation in terms of the attributes and attribute values being shown. Where possible this should be able to be achieved without having to re-cook the input data[1]. For example, consider the situation in which we have the results applicable to all in-patients but, as a short term goal, we may be interested only in patients presenting with a particular complaint or being prescribed a certain medication. Alternatively, we may find a particular result interesting and wish to find other similar results.

From these experiences and circumstances we have developed a number of techniques which have been combined into a single tool called IRSetNav (Item and Rule SET NAVigation). A description of this system will be given in Section 4.

3 Related Research and Systems

There are a number of papers and prototype systems that address some of the issues involved in handling a large result set. Some of the principal systems that possess similarities to IRSetNav are discussed below.

The KEFIR system (Matheus, Piatetsky-Shapiro & McNeill 1995, Piatetsky-Shapiro & Matheus 1994) focusses on medical health costs. The main feature of KEFIR is that it finds groups of deviations in the data and determines interestingness based on the estimated benefits. KEFIR is similar to this work in that it presents the results to the user in an organised fashion, rather than solely as text output from a mining algorithm.

Liu and Hsu (1996) focus on classification systems, incorporating user descriptions of existing knowledge, in particular, rule interestingness through fuzzy matching to user expectations. Pertinent to IRSetNav is the discussion on the importance of post processing systems and of observing the change between mining results at different times, which is similar to the concept of the knowledge base.

Sahar (1999) presents a system that uses a knowledge base in a similar manner to IRSetNav. The main feature covered is the removal of descendants (as in IRSetNav) which can be done in four ways, each indicating a belief and an interest in the rule (ie. whether the user believes the rule to be true/false and whether the user has a positive or negative interest in the rule).

Tuzhilin and Gediminas (2002) describe a system with some similarities to the work presented here, except that they specialise their work in the micro array domain. Their system applies existing techniques, such as template based rule filtering, and domain specific solutions. Their solution appears tightly connected to specific properties from the micro array domain which does not render it immediately applicable to a general domain, however, some of their general principles are useful.

While the IRSetNav system takes into account some aspects of longitudinal rule processing it should not be compared to the large body of work on trend detection and emerging pattern mining (Liu, Hsu & Ma 2001). In general, this research aims to detect changes in a data set to find rules that may be interesting to the user, whereas our aim is to track user interests from one data set to the next to make it easier for the user to process the next data set that they encounter.

Graphical visualisation techniques are a powerful way of displaying results in a manner which is more accessible to the user. At this stage in the development of IRSetNav our aim is not to provide a graphical view of the mined results, but to give the user a way

[1] In some cases the source input data may not be readily available.

of processing the results more efficiently. However, IRSetNav does facilitate the integration of results with third party graphical and textual visualisation tools, such as rule visualisation software or more standard text and word processing packages.

4 System Description

The prototype IRSetNav system has been written in Java to allow easy cross platform deployment. The prototype has been successfully tested with an input file with around 100,000 rules. The ability to handle larger files will vary according to implementation details that are not relevant to this paper.

The main aim of the IRSetNav system is to provide an intuitive user interface to the result set. Figure 1 displays a screenshot of the user interface for the IRSetNav system. Itemsets and rules are displayed in a table based layout to give the user a clear, consistent view of the results.

A known limit of human cognition is that we are able to process a maximum of approximately seven pieces of textual information at one time. We have therefore imposed a limit on the number of rows displayed (at least by default) at any time[2]. This helps prevent the user feeling overwhelmed by the size of the result set and focus on what is currently being shown to them. This design decision was pivotal in that it led to the inclusion of other features to help maximise the navigational ability and the information content within this limited set of items.

The interface separates the items, itemsets, rules and knowledge/filter base by placing them in a tabbed panel interface, which also helps to further focus the user on one area at a time. To utilise the information contained to highlight interesting results, it is possible to sort on each of the rule quality measures.

In addition to this, the system provides tools as follows:

- Filters, to restrict the displayed itemsets and rules;
- Search facilities, to find both more generalised and more precise itemsets and rules;
- Knowledge base, to store itemsets and rules that have been filtered or marked as uninteresting to the user;
- Filter storage, to store the set of filters that the user has applied to a result set;
- The accommodation of attribute value hierarchies;
- Result set comparison techniques and difference engines;
- The ability to dynamically invoke third-party graphical or other software or recursively call IRSetNav with (subsets of) the working ruleset.

Our experience has shown that using these tools individually or in combination can address many of the issues discussed earlier. These features are covered in the following subsections.

In the rest of the paper, the notation is as follows. Items are denoted by upper case alphabetic characters while itemsets are presented as a set of items $I = \{A, B, ..\}$. An itemset I_1 that contains all of the same items as another itemset I_2 plus any additional items is defined to be a superset of the original.

[2]The system defaults to 10.

4.1 Multiple Rule Quality Measure

Itemsets and rules are usually graded according to measures such as their quality (for example, interestingness or usefulness). The most common such measures are support and confidence. If an itemset has a given interestingness value and all supersets of the itemset always have a lower or equal interestingness value, then the interestingness measure is said to have the closure property. Support is an example of an interestingness measure that has the closure property, while confidence is an example of one that does not.

Since there are several useful interestingness (or rule quality) measures besides support and confidence, it is therefore possible to prune or mask based on these alternative metrics. To this end, we designed IRSetNav to accommodate any arbitrary interestingness measure from the result set. For example, provided with rules in the format shown below:

$$antecedents \rightarrow consequents \;\; s(\alpha)\; c(\beta)\; e(\gamma)\; p(\delta)$$

we would ideally allow navigation based on any of the four quality metrics, s, c, e or p.

4.2 Filters

One of the simplest ways of maximising the usefulness of a table display and dealing with excessive numbers of rules is to filter out unwanted results using the rule's quality value (such as support or confidence) with respect to some specified thresholds. Moreover, as a result set may contain multiple rule quality metrics IRSetNav handles an arbitrary number of these, each of which can be given an upper and lower bound. This can remove many results that are obvious and quickly reveal less obvious and possibly more interesting results.

To deal with the situations where there are items which have very high support or that are highly correlated we incorporated alternative filters. The simplest of these hides or only displays itemsets and rules containing the specified item, or itemset. This can be further refined to only display rules if the specified item or itemset is part of an antecedent or consequent of a rule.

By providing filters that mask everything but the selected itemsets it is possible to focus on parts of the result set without needing to re-mine the data set. For example, if a user is interested in rules containing $\{A\}$ or $\{B\}$ they can select the items individually and filter the result set to only display rules that contain them.

The sub/superset filter allows the removal of all sub or super sets of the selected item or itemset. This allows the user to remove all instances of an unwanted relationship. For example, if the superset filter is applied to the itemset $\{A, B\}$ then itemsets such as $\{A, B, C\}$ and $\{A, B, D, E\}$ would be removed. As long as the interestingness measure has the closure property, no useful information will be removed by this filter. For example if the itemset $\{A, B, C\}$ is interesting only because it shows a relationship between $\{B, C\}$, then $\{A, B, C\}$ can safely be removed. This is because for $\{A, B, C\}$ to exist as an itemset, $\{A, B\}$, $\{B, C\}$ and $\{A, C\}$ must all meet the threshold minimum interestingness requirements as well. In this case the interesting relationship, $\{B, C\}$, would still be in the results set.

Using this filter we can also focus our result set for rules that contain $\{A\}$ and $\{B\}$ by selecting the itemset $\{A, B\}$ and filtering to only display supersets of it.

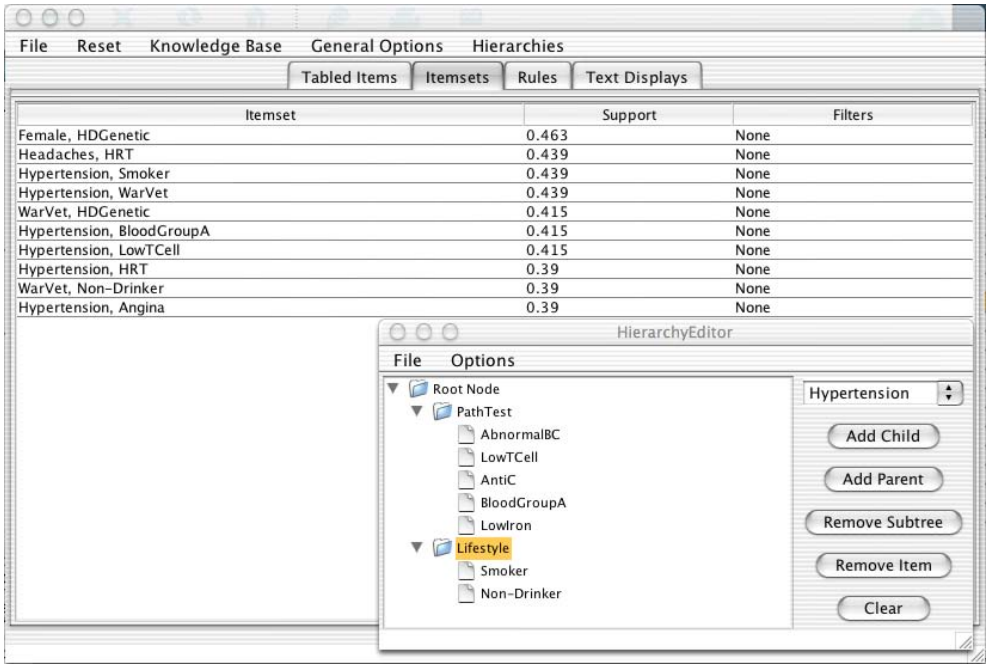

Figure 1: IRSetNav User Interface

4.3 Searching

Searching for similar interesting rules in the result set can be useful for focusing on related information within a result set. The user can select a target and search for items that are related to it. Results from a search can be displayed in a separate table in the interface, or as an input set for a new IRSetNav window from which a new investigation can begin.

Searching can be focused for entries that are more general or more specialised than the specified target. In each of these cases there are two ways of searching: looking at the items that the itemset or rule contains, or employing a concept hierarchy. Using the itemset's or rule's structure to find more general or specialised results involves looking for supersets or subsets of the entry respectively. The hierarchy based method involves creating a hierarchy of the items. IRSetNav provides a tool for generating the hierarchy with a graphical display.

4.4 Knowledge Base and Filter Store

To support long-term analysis of a domain area, a knowledge base and filter store were implemented. When the user filters out any itemsets or rules from the result set they can be stored in the knowledge base. The filters applied are stored in the filter store. These storage sets can be saved, reloaded and applied to filter another result set. For example, they can be used to remove all rules stored in the knowledge base from the results set. Apart from allowing known rules to be masked it also allows mining sessions to be interrupted and resumed.

These tools were created to be used in several situations, particularly when working in domains where the knowledge base is substantial. When working with multiple data sets that have similar results, we aimed to avoid having the user need to remove itemsets and rules that were already encountered and dismissed as uninteresting from a previous result set.

The aim of the filter store is to increase the coverage of the known results compared to the knowledge base. Where the knowledge base stores only entries that have been directly marked by the users, the filter base stores the relationships that have been marked as uninteresting. This gives IRSetNav the ability to filter out results that the user will find uninteresting without the need to directly review them.

When using the knowledge base on consecutive data sets, it is likely that the interestingness for entries will vary slightly between result sets. A change threshold is used to determine if the interestingness of the rule has changed significantly enough to bring the entry to the attention of the user. That is, for an entry in a result set to be deemed significantly different, the interestingness measure must be greater than the recorded interestingness plus the change threshold, or less than the recorded interestingness minus the threshold. The user can change the value of the change threshold for each entry to suit the situation.

To alert the user of the change the entry is not removed from the result set but is instead highlighted using colour according to the type of change. We use shades of green for a positive change and shades of blue for a negative change. The shades are light if the change is minimal and get darker as the magnitude of the change increases. The user can dismiss the rule again to update the recorded interestingness.

For example, if the user is not interested in the itemset $\{A, B\}$ with support of 50% from the initial result set, but would like to be notified if its support changes significantly in any subsequent result sets, then the change threshold for that itemset can be set to, say, 5%. If on a subsequent result set the support value for $\{A, B\}$ goes to 56% then $\{A, B\}$ is highlighted with a light blue colour.

4.5 Result Set Comparison

This tool is useful in situations where two result sets are created from different sources but from a similar set of items. When the two result sets are compared, each itemset and rule will fall into one of three categories: having an equivalent partner in the other result set, having a partner in the other result set with significantly differing interestingness value, or having no partner in the other result set.

This is displayed to the user as a table of all the entries from each data set. Colour is again used to identify which of the three groups each itemset or rule belongs to. Purple is used for entries that have a

matching pair. Shades of green and blue are used for entries that have a similar partner, with the shading being assigned in a similar manner to the knowledge base for consistency of interface. For entries that belong only to one of the sets a textual indication is given of which set it belongs to.

4.6 Visualisation Tools

Visualisation tools can be very powerful for processing result sets. The drawback with most visualisation tools is that they give only a global view of the data. By connecting the selection capabilities of IRSetNav to a visualisation tool we can gain the benefits of both.

The current level of integration in the prototype is to pass data to generic third party visualisation tools. Future work will extend this to provide real time communication between IRSetNav and specially created visualisation tools.

5 Conclusions and Future Work

Using our experiences in the mining of medical data we have developed IRSetNav which incorporates several techniques that we have found to be useful for speeding up the knowledge discovery process. Subsequent use on other projects has indicated that they are applicable to a wider range of domains. As stated earlier, it is our aim to provide this tool to knowledgeable end-users and thus an intuitive interface is considered a high priority.

We have current plans to extend the system further into longitudinal analysis beyond the use of the filter store and the knowledge base. The system is already capable of handling arbitrary interestingness measures which means that it is simple to alter the results to extend them into the longitudinal field. Following this we are interested in looking at search capabilities across multiple interestingness values of a rule or itemset. For example, we could look for any rules which have confidence at a steady level before and after a sudden spike.

Note that longitudinal in this case need not be temporally ordered datasets nor need they be just one-dimensional. Datasets ordered in time and space should be able to be accommodated forming an n-dimensional hypercube of rule quality values.

Another area we believe could improve overall productivity is a more flexible way of specifying the schema and data cleaning of the input data, particularly where this data has be derived from more or less structured databases. For example, intuitive ways of turning on and off selected attributes and of coercing data to agreed enumerated values could both improve usability and performance.

In this paper we have discussed IRSetNav which helps reduce iterations in the knowledge discovery process by reducing its iterative nature. While our extensions are pragmatic and come from real world requirements, we believe there are also useful pointers to future research in this area.

References

Agrawal, R. & Srikant, R. (1994), Fast algorithms for mining association rules, *in* 'Twentieth International Conference on Very Large Data Bases', Santiago, Chile, pp. 487–499.

Brin, S., Motwani, R. & Silverstein, C. (1997), Beyond market baskets: generalizing association rules to correlations, pp. 265–276.

Ceglar, A., Roddick, J. F. & Calder, P. (2003), Guiding knowledge discovery through interactive data mining, *in* P. C. Pendharkar, ed., 'Managing Data Mining Technologies in Organisations: Techniques and Applications', Idea Group Pub., Hershey, PA, pp. 45–87. Ch. 4.

Chapman, P., Kerber, R., Clinton, J., Khabaza, T., Reinartz, T. & Wirth, R. (1999), The crisp-dm process model, Discussion paper, CRISP-DM Consortium.

Cristofor, L. & Simovici, D. A. (2002), Generating an informative cover for association rules., *in* 'Proceedings of the 2002 IEEE International Conference on Data Mining (ICDM 2002), 9-12 December 2002, Maebashi City, Japan', pp. 597–600.

Freitas, A. (1999), 'On rule interestingness measures', *Knowledge Based Systems* **12**(5-6), 309–315.

Fule, P. & Roddick, J. F. (2003), Detecting privacy and ethical sensitivity in data mining results, *in* V. Estivill-Castro, ed., '27th Australasian Computer Science Conference (ACSC2004)', Vol. 27 of *CRPIT*, ACS, Dunedin, New Zealand.

Hilderman, R. J. & Hamilton, H. J. (1999), Heuristic measures of interestingness, *in* J. Zytkow & J. Rauch, eds, '3rd European Conference on Principles and Practice of Knowledge Discovery in Databases (PKDD'99)', Vol. 1704 of *Lecture Notes in Artificial Intelligence*, Springer, Prague, pp. 232–241.

Liu, B. & Hsu, W. (1996), Post-analysis of learned rules, *in* 'AAAI/IAAI, Vol. 1', pp. 828–834.

Liu, B., Hsu, W. & Chen, S. (1997), Using general impressions to analyze discovered classification rules, *in* 'Knowledge Discovery and Data Mining', pp. 31–36.

Liu, B., Hsu, W., Chen, S. & Ma, Y. (2000), 'Analyzing the subjective interestingness of association rules', *IEEE Intelligent Systems* **15**(5), 47–55.

Liu, B., Hsu, W. & Ma, Y. (1999), Pruning and summarizing the discovered associations, *in* 'Knowledge Discovery and Data Mining', pp. 125–134.

Liu, B., Hsu, W. & Ma, Y. (2001), Discovering the set of fundamental rule changes, *in* 'Seventh ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD-2001)', ACM, San Francisco, CA.

Matheus, C. J., Piatetsky-Shapiro, G. & McNeill, D. (1995), Key findings reporter for analysis of health-care information, *in* U. M. Fayyad & R. Uthurusamy, eds, 'First International Conference on Knowledge Discovery and Data Mining (KDD-95)', AAAI Press, Menlo Park, CA, USA, Montreal, Canada, p. Demonstration.

Meo, R. (2000), 'Theory of dependence values', *ACM Transactions on Database Systems* **25**(3), 380–406.

Pasquier, N., Bastide, Y., Taouil, R. & Lakhal, L. (1999), Discovering frequent closed itemsets for association rules, *in* 'Proceedings of the 7th International Conference on Database Theory (ICDT99)', Springer, Jerusalem, Israel, pp. 398–416.

Pei, J., Han, J. & Mao, R. (2000), Closet: An efficient algorithm for mining frequent closed itemsets, *in* 'ACM SIGMOD International Workshop on Data Mining', ACM Press, Dallas, Texas, pp. 21–30.

Piatetsky-Shapiro, G. & Matheus, C. (1994), The interestingness of deviations, *in* U. M. Fayyad & R. Uthurusamy, eds, 'AAAI-94 Workshop on Knowledge Discovery in Databases', IEEE Press, Seattle, Washington, USA, pp. 25–36.

Roddick, J. F., Fule, P. & Graco, W. J. (2003), 'Exploratory medical knowledge discovery : Experiences and issues', *SigKDD Explorations* **5**(1), 94–99.

Sahar, S. (1999), Interestingness via what is not interesting, *in* S. Chaudhuri & D. Madigan, eds, 'Fifth International Conference on Knowledge Discovery and Data Mining', ACM Press, San Diego, CA, USA, pp. 332–336.

Silberschatz, A. & Tuzhilin, A. (1996), 'What makes patterns interesting in knowledge discovery systems?', *IEEE Transactions on Knowledge and Data Engineering* **8**(6), 970–974.

Toivonen, H., Klemettinen, M., Ronkainen, P., Hatonen, K. & Mannila, H. (1995), Pruning and grouping of discovered association rules, *in* 'ECML-95 Workshop on Statistics, Machine Learning, and Knowledge Discovery in Databases', Heraklion, Greece, pp. 47–52.

Tuzhilin, A. & Gediminas, A. (2002), Handling very large numbers of association rules in the analysis of microarray data, *in* 'Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining', ACM, Edmonton, Alberta, Canada, pp. 396–404.

Zaki, M. J. & Hsiao, C.-J. (2002), Charm: An efficient algorithm for closed itemset mining, *in* 'Second SIAM International Conference on Data Mining', SIAM, Arlington, Vancouver, pp. 457–473.