A Data Mining Case Study

Mark-André Krogel
Otto-von-Guericke-Universität
PF 41 20, D-39016 Magdeburg, Germany
Phone: +49-391-67 113 99, Fax: +49-391-67 120 18
Email: krogel@iws.cs.uni-magdeburg.de

ABSTRACT: The CoIL Challenge 2000 offered the opportunity to apply data mining methods to a real-world data set. This paper describes an approach to solve the tasks of the Challenge. It also points the reader to the theoretical background of the subgroup analysis algorithm Midos. Its implementation for the data mining system Kepler was most intensively used for the investigations outlined in the text. The results section contains a solution for the description task of the Challenge, which was only slightly modified for this document. This solution received a “special note” from the jury. Moreover, some practical experience gained during data mining for the Challenge is included in the paper.

KEYWORDS: CoIL Challenge 2000, subgroup analysis algorithm Midos, data mining system Kepler

1 INTRODUCTION

Data mining or Knowledge Discovery in Databases (KDD) aims at finding novel, interesting, and useful information in real-world data sets [1]. The CoIL Challenge 2000 offered the chance to work on two tasks based on an insurance domain data set. The tasks were given in [2] as:

1. Predict which customers are potentially interested in a caravan insurance policy.
2. Describe the actual or potential customers; and possibly explain why these customers buy a caravan policy.

5822 records were provided as training data and 4000 records as test data, each record being a description of a customer in terms of 86 attribute values. One half of the attributes were sociodemographic data of zip code areas where the customers live, while the other half of the attributes contained information about the insurance product situation specific to the customers described. An essential property of the data was that there were no missing values for any attribute.

Task 1 was formulated more precisely as the demand for 20 percent of the 4000 test data record indexes. These 800 customers should be likely to be interested in a caravan insurance policy such that a virtual mail campaign for these addressees would be as successful as possible.

The following sections outline theoretical and practical aspects of an approach to solve the tasks of the Challenge, the results, and some experience gained during the data mining process.

2 BACKGROUND

In our research group, we are especially interested in multi-relational learning methods, e.g. for subgroup discovery, and aspects of data mining systems, e.g. their architecture [3].

For our work on the Challenge tasks, an algorithm for **m**ult**i**-relational **d**iscovery **o**f **s**ubgroups named Midos was most intensively used. This algorithm finds statistically unusual subgroups in a database. Midos uses optimistic estimate and minimal support pruning, and an optimal refinement operator. For our purposes, the evaluation function used within Midos is of special interest. It takes not only the proportion of positive examples in a subgroup into account but also the generality of a hypothesis, i.e. the size of the subgroup. A detailed description of the algorithm can be found in [4].

The database to be analysed need not be multi-relational, as it is the case for the Challenge where analysis concerns a single relation. If the Challenge data would have been provided in the form of two tables with one containing only the zip code area descriptions including an area key, e.g. the zip code itself, and the other one containing only the data specific to customers and a foreign key attribute pointing to the zip code area description in the other table, Midos could have handled this format as well, being a multi-relational algorithm.

An advantage of the split into two tables would have been the smaller size of the less redundant data set. 1734 different zip code area descriptions could be identified in the data, which means potential savings of about a third of the original number of values in the data set, cf. figure 1.
A further advantage could have been that the algorithm Midos should have been faster on that smaller and differently structured data set.

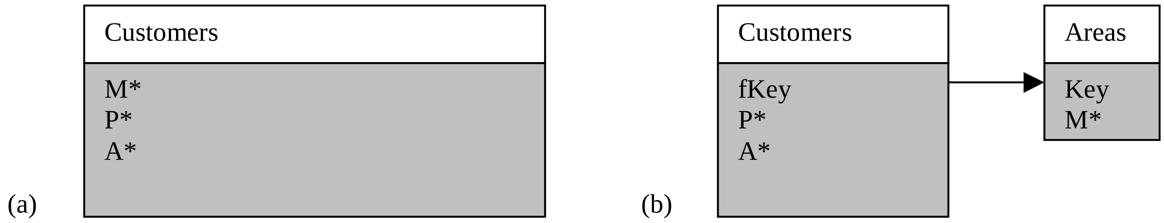

Figure 1: Schemas with table names and attribute set symbols. The sizes of the grey rectangles are proportional to the amount of data contained in the corresponding table. (a) The training data set as given for the Challenge. (b) The training data with a separation of customer specific data (attributes P* and A*) and zip code area data (attributes M*).

Kepler is a data mining system that supports users during most steps of knowledge discovery, from data import and initial inspections, via preprocessing the data, to a number of possible analyses and the interpretation of their results. All these steps can be made using a uniform interface. Also, the system helps to manage data analysis projects.
Kepler was used in our group in different ways in the past, e.g. as a means for examining software architectures [4] and in education.

3 APPROACH

Given the time constraints of the Challenge, I decided to make use of the appropriate parts of Kepler in order to quickly arrive at solutions for the prediction tasks on which the solution for the description task should be constructed.
At first, Kepler's kernel functionality for data import into the system made the relation quickly available for an initial inspection, e.g. of distributions. However, the means for statistical analyses such as distribution type or significance tests are limited within Kepler, so I relied on information from the Challenge discussion forum, e.g. those concerning very similar distributions in the training and test data. Also, my ideas for preprocessing the data, e.g. the construction of partitions or aggregations, and the application of self-developed algorithms were sacrificed due to a lack of time.
Still at an early stage of the investigations and indepently of Kepler, I formulated an aim for the prediction model as follows: Find a small set of short classification rules as a prediction model. Here, brevity of a rule means the number of its conditions. This aim was based on the assumption that an explanation, i.e. a solution for the description task, could be easily constructed from such rules.
The Kepler online documentation states that using this system, the task of an analyst shifts from the usual figuring out *how* to apply an algorithm to the data, to *experimenting* with a large variety of tools and *deciding* which one, or which subset, provides the most useful results for the task at hand. I tried exactly this and found some of the standard algorithms of Machine Learning such as C4.5 or kNN not to provide results supporting my aim for the prediction model. For example, although C4.5 produced a small set of rules based on a pruned tree and Kepler's default parameters for this method, these rules covered only few of the positive examples, i.e. customers with caravan insurance policies. Especially, the large number of parameters for the application of C4.5 prevented me from following this search path for solutions.
Furthermore, the Kepler documentation points at an example application for Midos, which is the discovery of customer groups that are more likely to answer to a mailing campaign, i.e. exactly the task at hand. Indeed, the very first results of Midos showed promising results. A small sample of one percent of the data was produced with the help of an appropriate Kepler operator. Then, the analysis algorithm was applied to that data, using its few default parameters.
The description of subgroups provided by Midos could easily be interpreted as the rules aimed at and be used as filters on the data set, cf. figure 2. Also, the proportion of positive examples in subgroups derived from the one-percent-sample was encouragingly high. However, these results did not generalize well. Applying those filters to the whole data set, the proportion of positive examples was low in the resulting subgroups.
With the application of Midos to the whole data set, a new problem arose: it was time consuming. I assume that one reason for this behaviour is the circumstance that the current Midos implementation can only handle nominal attributes. Again, converting all attributes to an appropriate type was simple using Kepler. Nonetheless, if an implementation of the algorithm could handle numeric attributes, this would be of advantage with respect to the possible results and probably with respect to efficiency, as well.

Working with the default value of 2 for the parameter “Search depth”, which means the number of conditions that can maximally be used to define a subgroup, an analysis of the whole training data set took about 2 hours on a Pentium III with 500 MHz. Changing that value to 3 made the analysis last for about 28 hours. Since the proportion of positive examples was not very much higher in the latter case than it was in the first one, the default value was taken for further analyses.

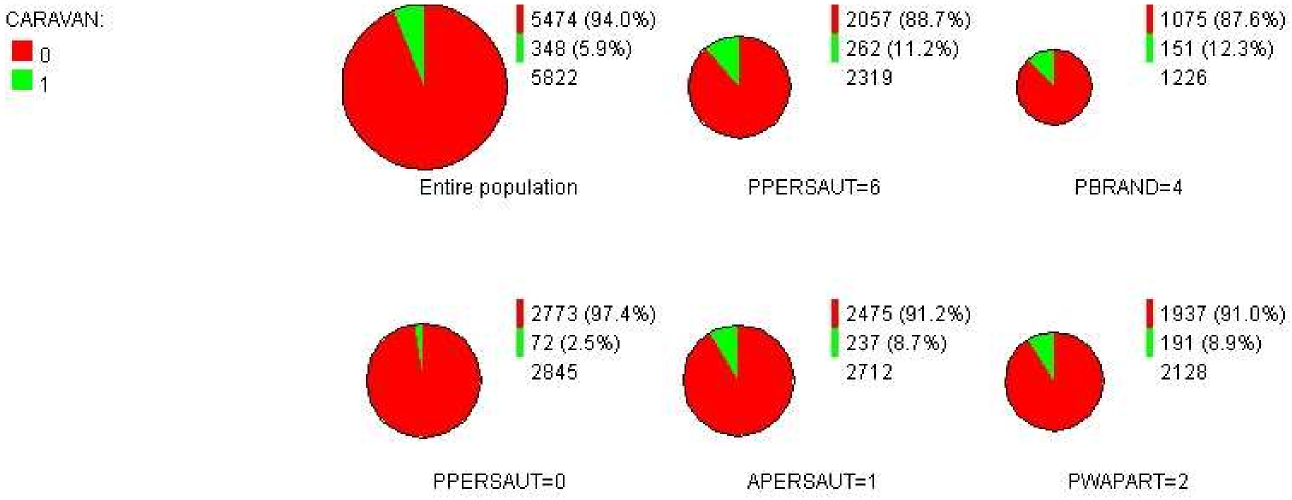

Figure 2: Example for a Midos result visualization provided by Kepler (“Solution size” = 5 and “Search depth” = 1). Circle sizes are proportional to group sizes.

The other parameters to be determined were “Solution size” (number of subgroups to be computed), “Minimal support” (minimum subgroup size in terms of a percentage of the training data set size), and “Max. Attribute Cardinality” (maximum number of different values for an attribute that should be considered). While it was easy to set the last parameter, namely to the number of values for MOSTYPE, which showed more values than any other attribute, a useful choice of the values for the first two parameters evolved only during an interactive process.
At the beginning of that process, the initial aim could be refined on the basis of experience with the data and Midos. The aim was now the following: Find a small set of short rules such that they filter disjunct subgroups with high proportions of positive examples and whose elements add up to about 20 percent of the training examples.
In this situation, I would have wished for the Midos implementation to offer a parameter like “Maximum support” or an opportunity to manipulate the evaluation function, especially for disabling those parts responsible for preferring larger subgroups over smaller ones. These features were not available such that the choice of the value for the parameter “Solution size” had to be determined empirically. While there was a default value of 10, the rule of my choice, or better, of a 5-fold crossvalidation’s choice in the first step (PPERSAUT = 6 and MOSTYPE = 8) came only at position 24. I experimented with parameter settings up to 100. Finally, I used a value of 40 because the visualization of a result set of that size allowed a good recognition of the subgroups and because I did not observe any high quality subgroups beyond that limit.
As for the parameter “Minimal support”, I chose a value of 0.025 instead of the default 0.1, because the default value would have meant to aim at only two subgroups to make up the 20 percent group mentioned above, while smaller values would have led to many very small subgroups. Handling these subgroups would have been expensive, and I had the intuition that those small group rules would not generalize very well.
Furthermore, one more parameter like “Disjunct subgroups only” or a visualization of pairs of overlapping subgroups could have been useful. Instead of using facilities like these, Midos was run several times with the parameter settings given above, each time removing the records from the training data set which were filtered out by the best rule in order to arrive at the data set for the next run.
The first seven rules determined in that way filtered about 20 percent of all training data, which contained about 50 percent of the positive examples. From the eighth rule, the number of covered positive examples decreased rapidly. In order to cover all positive examples, there were 27 rules necessary whose subgroup union made up nearly 60 percent of the test records. These outcomes seemed disappointing to me. It was only when I received the information from the Challenge discussion forum that it was hard to get more than 60 percent of the positive examples in the 20 percent extract, that I decided to hand in my results.

The construction of an explanation from the rule set found was comparatively easy, as I had hoped for. I chose to represent the rules as a WHERE-clause of a SQL query, because I assumed this to be comprehensible and operational not only for specialists but also for advanced computer users in general. Moreover, some general knowledge about the world was included in the non-formal text of the explanation as a solution for the description task.
The following section contains this text, which was only slightly modified for this paper. It is not mentioned there that I also checked the according subgroups to have an income above average, which could be confirmed.

4 RESULTS

The prediction model was constructed with the help of a subgroup discovery algorithm that was applied to the training data set. It describes the interesting customers in terms of conditions on certain attributes, cf. tables 1 and 2.

PPERSAUT = 6	and	MOSTYPE = 8	or
MSKB1 = 3	and	PBRAND = 4	or
PPERSAUT = 6	and	MKOOPKLA = 8	or
PPERSAUT = 6	and	MOSHOOFD = 1	or
PPERSAUT = 6	and	MOPLLAAG = 2	or
PPERSAUT = 6	and	MBERARBO = 1	or
PPERSAUT = 6	and	PBRAND = 3	

Table I: The prediction model.

Attribute	Value(s)	Meaning according to distributions of attributes and data set description
MBERARBO	1	Low percentage of “Unskilled labourers”
MKOOPKLA	8	High percentage of “Purchasing power class”
MOPLLAAG	2	Low percentage of “Lower level education”
MOSHOOFD	1	“Successful hedonists”
MOSTYPE	8	“Middle class family”
MSKB1	3	High percentage of “Social class B1”
PBRAND	3 or 4	High value for “Contribution to fire policies”:
PPERSAUT	6	High value for “Contribution to car policies”

Table II: The meaning of the attributes and values contained in the prediction model; attributes in alphabetical order.

The conditions can be used as a WHERE-part of an SQL-SELECT-statement, for example. This kind of filter selects a few more than 20 percent of the customers from the training data set as well as from the test data set. Some of the selected records have to be ignored in order to arrive at exactly 20 percent as demanded in the prediction task. The record set selected from the training data set contains about 50 percent of all training data set records with CARAVAN = 1 ("Number of mobile home policies").
A reformulation and interpretation of the filter conditions is the following: Customers are likely to be interested in caravan policies if they tend to pay high insurance policies for cars and/or for fire policies, and if they live in areas with a high percentage of people with a comparatively high social/educational status.
Cars are in many cases necessary for moving caravans such that the relation between car insurance and caravan insurance seems plausible. As for the fire policies, the connection to caravans is not that clear to me. There is maybe a high risk of fires in caravans which is known to many caravan owners.
The social/educational status of the customers themselves and/or in their environment could trigger the desire for a certain degree of luxury and safety, and it could often be the precondition for an income that makes that luxury and safety affordable.

5 CONCLUSION

The evaluation of the solution proposals for the CoIL Challenge 2000 showed that the prediction model described above reached a midfield rank, covering about 40 percent of the interesting customers contained in the test data. The best solution covered about 50 percent. The solution of the description task presented above received a “special note” by the marketing expert who evaluated those solutions. The conclusion may be that it is possible to achieve good results on a data mining task even for a comparatively inexperienced user of a data mining system such as Kepler.
The effort I had to take to arrive at my solutions for the tasks was about 50 hours of work, not including this report. It may be argued that this venture was more expensive than sending mail to all 4000 customers described in the test data

set. This seems even more important when looking at the 60 percent of interesting customers that were missed by my prediction model which would mean a high loss of profit in a real mailing campaign.
A thorough theoretical background as well as some data mining experience seem to be very useful for a data miner, even when working with a tool like Kepler. The appropriate background would make it easier to understand the descriptions contained in the Kepler documentation. Practical experience should make the effects of Kepler's features clearer that can occur while working with different data sets and which were not always predictable from the documentation in the case of the Challenge data from my point of view.
I also see some opportunities to improve Kepler and the methods implemented to be used within this environment. This includes support for the users such as space and time complexity forecasts and progress reports. Last not least, some guidance through the "search space" of data mining methods, their parameters, and their different ways of application, e.g. the usage of different samples from the data, seems very advantageous to me. This may also be a potential for useful research work in the future.

ACKNOWLEDGEMENTS

I would like to thank the CoIL Challenge 2000 organizers for providing that valuable opportunity for data mining. Special thanks to Peter van der Putten, who encouraged me to write up a short paper on my approach and results. Thanks to the participants who used the email discussion forum, where I could find many useful hints. Also, I thank my colleagues at Magdeburg University Susanne Hoche, Tobias Scheffer, and Stefan Wrobel, for discussions during the contest and about this paper. Last not least, thanks to Dietrich Wettschereck from dialogis GmbH for his always kind and quick Kepler system support.

REFERENCES

[1] Fayyad, W.; Piatetsky-Shapiro, G.; Smyth, P., 1996, "From data mining to knowledge discovery: An overview.", In: "Advances in Knowledge Discovery and Data Mining", W. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy (eds.), AAAI/MIT Press, Cambridge/USA, pp. 1 – 34
[2] http://www.dcs.napier.ac.uk/coil/challenge/thetasks.html
[3] http://kd.cs.uni-magdeburg.de/index_e.html
[4] Wrobel, Stefan, 1997, "An algorithm for multi-relational discovery of subgroups", Principles of Data Mining and Knowledge Discovery: First European Symposium, Proceedings of the PKDD '97, J. Komorowski and J. Zytkow (eds.), Springer-Verlag, Berlin, New York, pp. 78 – 87
[5] Wrobel, Stefan; Wettschereck, Dietrich; Sommer, Edgar; Emde, Werner, 1996, "Extensibility in Data Mining Systems", Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD-96), AAAI Press, pp. 214 – 219