Retrieval of Cases by using a Bayesian Network

Torgeir Dingsøyr
Dept. of Computer and Information Science
Norwegian University of Science and Technology
7034 Trondheim, NORWAY
dingsoyr@idi.ntnu.no

Abstract

A framework for integrating methods for decision support; Case-Based Reasoning (CBR) and Data Mining (DM) is outlined. The integration approaches are divided according to which method that is considered to be *master* and which is the *slave*. A system using Bayesian networks for computing similarity metrics is implemented and compared to a traditional CBR system. Data are taken from a database from the oil industry. The retrieved cases vary greatly between the systems, especially on features that are unspecified in the "new case". If many features of the "new case" are specified, the new system performs better, according to an evaluation by a domain expert.

Introduction

Data Mining and Case-Based Reasoning are methods used for *decision support*; to organize and process information to make it available for improving the quality of decisions. It is likely that integration of the two methods will lead to a better usage of information. Here, we give a quick introduction to the methods, briefly describe existing integrated methods, outline a framework for different integration approaches. Then we describe an implemented system that uses Bayesian networks to compute similarly metrics for retrieving cases.

Data Mining (DM) has become a popular method for extracting information from large databases, in the form of patterns. The patterns can be *informative* or *predictive*, and some DM methods are classification, regression, clustering, dependency modeling and change and deviation analysis. The whole process of discovering knowledge in databases is referred to as *Knowledge Discovery in Databases* (Fayyad, Piatetsky-Shapiro, & Smyth 1996).

Case-Based Reasoning (CBR) is a method for solving problems by comparing a problem situation – a case – to previously experienced ones. The aim is to store information about earlier situations, and when new ones arrive, *retrieve* the situation that is most similar, and *reuse* it – or *revise* it to match the new problem if the most similar problem does not match sufficiently. If the new case provides new insight it should be *retained*. An introduction to CBR is given in (Aamodt & Plaza 1994).

At present, there are some integrated systems under development, like Case-Method, developed by the NEC corporation for handling corporate memory (Kitano, Shimazu, & Shibata 1993), and a system for forecasting of epidemics, developed at the University of Rostock (Bull, Kundt, & Gierl 1997). Most of the research have been to integrate Bayesian networks with CBR. Microsoft has developed two prototype systems with code-name "Aladdin" to diagnose problems in customer support (Breese & Heckerman 1995). A system named INBANCA (Integrating Bayes networks with CBR for Planning) for planning in a simulated soccer environment has been outlined in (Aha & Chang 1996). At the University of Salford, a system using Bayesian networks for indexing cases have been developed (Rodriguez, Vadera, & Sucar 1997), and at the University of Helsinki, CBR and Bayesian networks are used for classification (Tirri, Kontkanen, & Myllymäksi 1996). Tools for combining CBR and decisions trees were developed in the Esprit project INRECA (Althof *et al.* 1995b).

Integration Framework

Integration of DM and CBR can be done with either of the methods as the master and the other as the slave, depending on which method that uses information provided by the other. First we outline methods for integration with CBR as the master, then with DM as the master. We assume that there exists a casebase (database of cases) and in some scenarios also an external database from which further information can be mined.

With CBR as the master, we can:

- *Find features for a case* (from casebase) – Classify the cases in the casebase for use. This might speed up the computation of similarity metrics in the *retrieve* step of the CBR cycle, if the case(s) that will be retrieved are known to be in the same class as the new case.
- *Find features for a case* (from a database) – A database can be searched to supplement the information given in a case. For instance, Gibbs sampling can be used to fill in missing features in case-data.

- *Find domain knowledge* (from a database or a casebase) – Domain knowledge might be mined from the data in the form of functions, rules or causal graphs which can later be used by the case-based reasoner when identifying features and explaining away unimportant features in the *retrieve* step, adapting in the *reuse* step, or explaining cases in the *retain* step.
- *Construct "artificial cases"* – we can imagine that it is possible to construct cases from a database, that is not present in a casebase, "unexperienced problem situations". This would require a deep integration where the DM algorithm would search for patterns in the form of cases, which could be evaluated by a novelty function which gives high values to cases not present in the casebase.

With DM as the master:

- *Cases are the KDD process* – DM is only one part of the KDD process which can involve accessing several files, cleaning data and interpreting results. The DM search may be time-consuming. The information about the search results and the whole knowledge discovery process might be stored in a case so that extra time will not be spent on mining the same information more than once.
- *CBR provides info* – CBR can be used to provide background knowledge about features in a database, (e.g., the weight of features for a classifier can be learned from the CBR tool). In a Bayesian network, the structure of the network might be set up by the CBR tool (model construction), using its "expert knowledge" and the parameters learned using DM algorithms. CBR can also be used to provide utility, validity and novelty functions for the DM algorithm from the domain that the CBR tool is working in (model evaluation).

We have now given the status of research on integration of DM and CBR, outlined a framework for different methods of integration, which should be areas for further research. To demonstrate that integration of the two methods can lead to better usage of information, and to better results, we now describe an implemented prototype system which combines CBR and domain knowledge mined from a database in the form of a Bayesian network. The network is used for for doing causal inference.

Specification of an Integrated System

To demonstrate an integrated system, we propose the following algorithm which we will denote CBRDM:

- A new case description is taken from the user.
- The CBR engine does a search in the casebase for similar cases. The similarity metric uses a Bayesian network computed from the casebase for inference such that: If no exact syntactic match is found, the most similar case is assumed to be the case which for the features that are different, has the highest probability of having the feature values.

The similarity metric is then:

$$Similarity(x,y) = -\sqrt{\sum_{i=1}^{p} f(x_i, y_i)}$$

Where x and y are two cases, p the number of features and f is defined in Figure 1.

$$f(x_i,y_i) = \begin{cases} (x_i - y_i)^2 & \text{if } x_i,\ y_i \text{ are numeric} \\ 0 & \text{if } (x_i = y_i) \text{ and } x_i,\ y_i \text{ are symbolic} \\ 1-p & \text{if } x_i,\ y_i \text{ are symbolic, } x_i = X, y_i = ? \text{ and} \\ & P(y_i = X \mid y_1, \ldots, y_{i-1}, y_{i+1}, \ldots, y_p) = p \\ 1-q & \text{if } x_i,\ y_i \text{ are symbolic, } x_i = ?, y_i = Y \text{ and} \\ & P(x_i = Y \mid x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_p) = q \\ 1 & \text{Otherwise} \end{cases}$$

Figure 1: The Function used in the Similarity Metric.

The procedure of constructing the Bayesian network and casebase, and an overview of the system is given in Figure 2. The system is described in further detail in (Dingsøyr 1998).

To test the new system, we used a database from the oil industry, OREDA, which contains data about inventory items, failures and maintenance on platform equipment. We focused on data on *compressors*, which gave 4646 *cases* which was described by nine *features*, where some were missing. Then we used Bayesian Knowledge Discoverer (Ramoni & Sebastini 1997) to construct the Bayesian network, and the case-based reasoning tool KATE[1], to compare with the results obtained from the new system (see (Althof *et al.* 1995a) for a description of this and other industrial CBR tools).

We constructed 45 queries ("new cases") which we divided into five series. Each series thus consisted of five queries which had a similar number of feature values given, and where certain feature values were varies in each series. Other feature values were set to *missing*. The number of given feature values were increased for the later series.

An example part of a query is (for the five first features):

	1	*2*	*3*	*4*	*5*
Query1	NA	NNF	MATFAIL	GASLEAK	NA
KATE	2	NNF	MATFAIL	NA	MINOR
CBRDM	2	NFI	MATFAIL	NA	MINOR

Which retrieved the indicated best cases, where "NA" is a feature value which is not available.

Results

The differences in the returned cases are shown in Table 1, where the number of features that have different values when KATE and CBRDM is compared to the query is shown. It also shows the difference between KATE and CBRDM. For instance, the first line in the table states that there is one case in Series 1 that differs

[1]KATE from Acknosoft was selected because it is an industrial CBR tool, and was easy available through the project NOEMIE, where this work was conducted.

Cases	*0*	*1*	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*Series*
CBRDM-KATE		1			2	2					1
KATE-Query	2	3									
CBRDM-Query	3	2									
CBRDM-KATE			1	1	3						2
KATE-Query		5									
CBRDM-Query		3	2								
CBRDM-KATE				3	2						3
KATE-Query		5									
CBRDM-Query		5									
CBRDM-KATE	1			1	2		1				4
KATE-Query		5									
CBRDM-Query		5									
CBRDM-KATE				2	2		1				5
KATE-Query		5									
CBRDM-Query		5									
CBRDM-KATE				3	1			1			6
KATE-Query		5									
CBRDM-Query		5									
CBRDM-KATE						2	2			1	7
KATE-Query	1	3	1								
CBRDM-Query	3	2									
CBRDM-KATE					1	1	2			2	8
KATE-Query	3		2								
CBRDM-Query	3	2									
CBRDM-KATE				1		1	1	2			9
KATE-Query		1	1	2	1						
CBRDM-Query	1	1	2	1							

Table 1: Number of Differences Between Cases, for All Series.

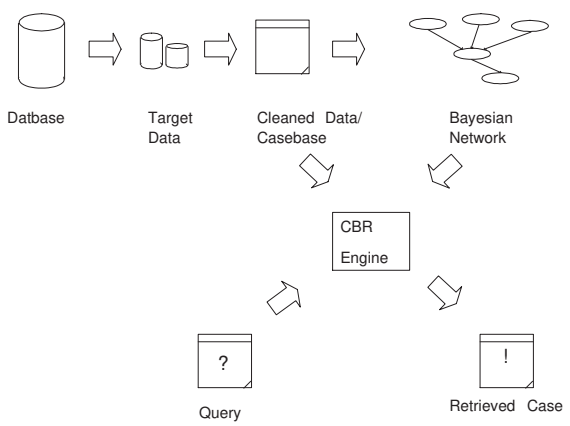

Figure 2: Architecture of the Proposed Integrated System (CBRDM).

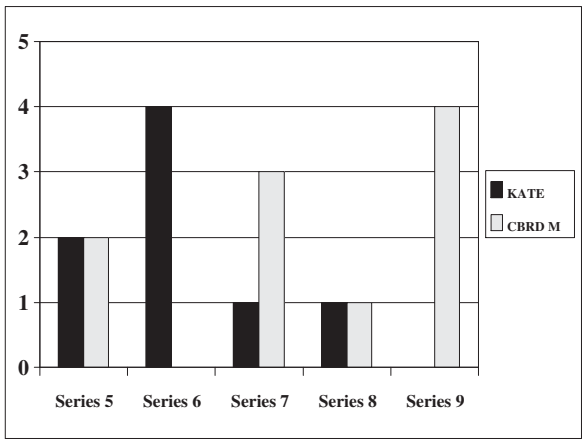

Figure 3: KATE and CBRDM was Given a Point by a Domain Expert if the Result was Better for one Method.

by one feature for CBRDM and KATE, and two cases that differ by four and five features.

An expert on the OREDA database was given a list of the processed queries, where the information on which tool had produced which result was removed. The order of the presentation of the results was picked at random. The expert was told to choose which of the two retrieved cases was most similar to the new case. The results from the expert is given in Figure 3, where a point was awarded if the result could be said to be better for one system. The expert did not evaluate the first four series.

Discussion

If we define the difference between the query and the retrieved case as the number of features that does not match and does not have the value *not available* in the query, the output cases from the two methods are only similar for one query. In Figure 4 the number of differences in feature values between the cases retrieved with the two methods are shown. 43 of the pairs of cases retrieved differ by three or more feature values. On average, every case differs by 4.5 feature values of a total of 9. So, it seems that the feature values retrieved by the systems are different.

Why Differences Occur

The difference between the query and CBRDM, and the query and KATE are never higher than 4, the average is

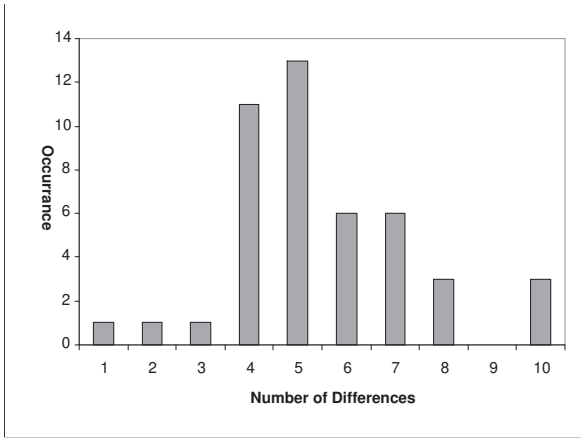

Figure 4: Accumulated Number of Differences Between Output from KATE and CBRDM.

1.02, and 17 out of 45 are exactly similar. The difference between the results from CBRDM and KATE is always larger. This indicates that the differences experienced come mostly from other feature values than the ones that are given by the user in the "new" case.

Why does this occur? In the similarity metric for CBRDM we add 1 for each feature value that is similar. If the feature value is not similar, we add the probability (< 1) for having the value that is similar, given the rest of the feature values of the case. In that way, if we cannot find an exact hit, we choose the one that has the highest probability of occurring in the future (if we assume that the frequency in the casebase approximates the probability when the number of cases is large).

The difference between the query and the retrieved cases are relatively stable over the different series. But the difference between CBRDM and KATE results vary greatly with the series. This is not surprising, as there are nodes in the Bayesian network (not shown here) that are not connected, and other nodes that have several connections. The similarity metrics with features that have connections will be higher than those without connections.

Evaluation of Results

Series 5 to 8 have approximately the same number of features given in the query, while more feature values are given in the queries in Series 9. It seems that CBRDM produces better results when more feature values are given in the query, and when the feature values that are given are a sub-node in the Bayesian network.

Conclusion

From the discussion, we can draw the following conclusions:

- The results from the CBRDM tool developed and KATE differ greatly in the features that are left blank in the "new case" under retrieval.
- The integration of Bayesian networks and Case-Based Reasoning can lead to better results if the networks represents sound knowledge. This improvement is better when a large number of features are given in the input query, and when the given features are sub-nodes in the Bayesian network.

Acknowledgments

The work reported here was done in the Esprit NOEMIE project, with professor Agnar Aamodt at the Norwegian University of Science and Technology (NTNU) and professor Edwin Diday at Universite Dauphine as supervisors. I would like to thank Helge Langseth at Sintef, Moufida Massrali at Universite Dauphine and M. Letizia Jaccheri at NTNU for comments.

References

Aamodt, A., and Plaza, E. 1994. Case-based reasoning: Foundational issues, methodological variations, and system approaches. *AI Communications* 7(1):39–59.

Aha, D. W., and Chang, L. W. 1996. Cooperative bayesian and case-based reasoning for solving multiagent planning tasks. Technical report, Navy Center for Applied Research in AI, Naval Research Laboratory, Washington, DC, USA.

Althof, K.-D.; Auriol, E.; Barlette, R.; and Manago, M. 1995a. *A Review of Industrial Case-Based Reasoning Tools*. AI Intelligence.

Althof, K.-D.; Auriol, E.; Traphöner, R.; and Wess, S. 1995b. Inreca – a seamlessly integrated system based on inductive inference and case-based reasoning. In Aamodt, A., and Veloso, M., eds., *Case-Based Reasoning Research and Development, ICCBR-95*, 371–380.

Breese, J. S., and Heckerman, D. 1995. Decision-theoretic case-based reasoning. In *Proceedings of the Fifth International Workshop on Artificial Intelligence and Statistics*, 56–63.

Bull, M.; Kundt, G.; and Gierl, L. 1997. Discovering of health risks and case-based forecasting of epidemics in a health surveillance system. In Komorowski, J., and Zytkow, J., eds., *Principles of Data Mining and Knowledge Discovery. Proceedings*, 68–77.

Dingsøyr, T. 1998. Integration of data mining and case-based reasoning. Technical report, Norwegian University of Science and Technology, 7034 Trondheim, NORWAY.

Fayyad, U. M.; Piatetsky-Shapiro, G.; and Smyth, P. 1996. *Advances in Knowledge Discovery and Data Mining*. AAAI/MIT Press. chapter From Data Mining to Knowledge Discovery: An overview.

Kitano, H.; Shimazu, H.; and Shibata, A. 1993. Case-method: A methodology for building large-scale case-based systems. In *Proceedings of the AAAI*, 303–308.

Ramoni, M., and Sebastini, P. 1997. Discovering bayesian networks in incomplete databases. Technical report, Knowledge Media Institute, The Open University.

Rodriguez, A. F.; Vadera, S.; and Sucar, L. E. 1997. A probabilistic model for case-based reasoning. In Smith, I., and Faltings, B., eds., *Case-Based Reasoning Research and Development, ICCBR-97. Proceedings*, 623–632.

Tirri, H.; Kontkanen, P.; and Myllymäksi, P. 1996. A bayesian framework for case-based reasoning. In Smith, I., and Faltings, B., eds., *Advances in Case-Based Reasoning, EWCBR-96*, 413–427.

Appendix

1. **Integration name/category**: CBRDM/ Integration of CBR and Data Mining.
2. **Performance Task**: Retrieval.
3. **Integration Objective**: Solution quality. Select cases that evaluate better by a domain expert.
4. **Reasoning Components**: Bayesian Network.
5. **Control Architecture**: CBR as master.
6. **CBR Cycle Step(s) Supported**: Retrieval.
7. **Representations**: Cases.
8. **Additional Reasoning Components**: -
9. **Integration Status**: Proposed.
10. **Priority future work**: Application of the technique on experience databases for software process improvement.