Architectures Integrating Case-Based Reasoning and Bayesian Networks for Clinical Decision Support

Tore Bruland, Agnar Aamodt, and Helge Langseth

The Norwegian University of Science and Technology (NTNU)
NO-7491 Trondheim, Norway

Abstract In this paper we discuss different architectures for reasoning under uncertainty related to our ongoing research into building a medical decision support system. The uncertainty in the medical domain can be divided into a well understood part and a less understood part. This motivates the use of a hybrid decision support system, and in particular, we argue that a Bayesian network should be used for those parts of the domain that are well understood and can be explicitly modeled, whereas a case-based reasoning system should be employed to reason in parts of the domain where no such model is available. Four architectures that combine Bayesian networks and case-based reasoning are proposed, and our working hypothesis is that these hybrid systems each will perform better than either framework will do on its own.

1 Introduction

The field of knowledge-based systems has over the years become a mature field. This is characterized by the availability of a set of methods for knowledge representation, inference, and reasoning that are well understood in terms of scope, strengths, and limitations. Numerous applications have been built that are in daily use, and hence have proven the various methods' value for intelligent decision support systems and other applications. As the individual method areas get more explored and better understood, the identification of limits and strengths opens up for integration of individual methods into combined reasoning systems.

The history of knowledge-based decision support systems, e.g. expert systems, started with rule-based systems. They were followed by systems that tried to open up the "if-then" association to look for what underlying knowledge, in terms of "deeper relationships" such as causal knowledge, could explain the rule implications [1]. Cognitive theories in the form of semantic networks, frames, and scripts formed the theoretical basis for many of these model-based systems. Statistical and probabilistic theories formed another method path. As the availability of data has increased over the recent years, and methods and algorithms for probabilistic reasoning have significantly evolved, probabilistic models, and in particular those based on Bayesian theory in one way or the other, have come to dominate the model-based method field [2]. Bayesian Networks (BN) is the

most prominent among these. It is particularly interesting in that it combines a qualitative model part and a quantitative model part [3].

Both rules and deeper models represent knowledge as generalized abstractions. A good knowledge model is therefore dependent on a human domain expert to construct the model, or on methods that can generalize the model from data. In either case, details about actual observations in the world are abstracted away in the model building phase, without knowing whether some of this specific information could be useful in the problem solving phase. The third and most recent basic type of reasoning in the history of knowledge-based systems addresses this problem by representing each problem instance as a unique piece of situation-specific knowledge, to be retrieved and reused for solving similar problems later [4]. Since its birth in the early 80s, the field of case-based reasoning (CBR) has grown to become a major contributor to decision support methods in academia as well as for industrial applications [5,6,7]. Increased availability of data on electronic form has also contributed to the growth of this field.

Although some early attempts have been made to discuss possible combinations of the two, including our own [8], our current research agenda represents a much larger and more comprehensive effort. Our focus in the work presented here is on improved support for clinical decision making. We are cooperating with the Medical Faculty of our university and the St. Olavs Hospital in Trondheim. More specifically we are working with the European Research Center for Palliative Care, located in Trondheim, in order to improve the assessment, classification and treatment of pain for patients in the palliative phase [9].

Decision making in medicine is to a large degree characterized by uncertain and incomplete information. Still, clinicians are generally able to make good judgments based on the information they have. Decision making under uncertainly – in the wide sense of the term – is therefore our setting for analysing the properties of BN and CBR, aimed at how they can be integrated to achieve synergy effects.

In the following chapter, decision making under uncertainty and the essentials of BN and CBR are characterized. Related research on combined methods are summarized in chapter 3. We discuss relevant combinations of the two methods in chapter 4, by outlining four specific architectures that utilize different properties of the two methods. In chapter 5 we give an example that illustrates one of the architectures, within a simplified, non-medical "toy" domain. The last chapter summarizes the results so far and points out future work.

2 Decision-making Under Uncertainty and Incompleteness

Our motivation for integrating BN and CBR is that they both contribute to improved decision making under incomplete information and with uncertain knowledge. They are both advocated as methods that to some extent address problems in *weak theory domains*. A weak theory domain is a domain in which relationships between important concepts are uncertain [10]. Statements are more or

less plausible, and stronger or weaker supported, rather than true or false. Examples of weak theory domains are medical diagnosis, law, corporate planning, and most engineering domains. A counter-example is a mathematical domain, or a technical artifact built from well-established knowledge of electricity or mechanics.

So, theory strength is one dimension of uncertainty characterization. Another dimension is the *knowledge completeness* of the domain. The fact that a domain has a weak theory does not always imply that there is little knowledge available. Although it may seem somewhat contradictory, weak theory domains need to compensate for lack of strong knowledge with larger amounts of knowledge, which jointly can provide a strengthening or weakening of an hypothesis being evaluated. Inferences in these domains are abductive (in the sense of "inference to the best explanation") rather than deductive, which is a characteristic of strong theories [11]. Three main knowledge types are typically combined in medical diagnosis and treatment: Physiological and pathological theories of various strengths, evidence-based clinical trials from controlled experiments, and person-centric experiences in diagnosing and treating patients [12].

General knowledge, with varying degrees of theory strength, can often be modeled by statistical distributions. The type of uncertainty that deals with assigning a probability of a particular state given a known distribution is referred to as *aleatory uncertainty*. This is a type of uncertainty that fits perfectly with the Bayesian Networks method. Another type of uncertainty, referred to as *epistemic uncertainty*, refers to a more general lack of knowledge, whether stronger or weaker, and are linked to cognitive mechanisms of processing knowledge [13]. Case-based reasoning, on the other hand, has nothing to offer for aleatory uncertainty, but is able to utilize situation-specific experiences as one type of epistemic knowledge.

For decision making under uncertainty, it is important to work with a framework that fits the domain, the available knowledge, the types of uncertainty, and the types of decisions to be made. The strongest theories in the medical sciences are often supported by randomized clinical trials, whereas weak theories lack this basis, and are just as often based on episodic knowledge and previous examples of successful and unsuccessful patient treatments. We are advocating the use of Bayesian Networks to model aleatory uncertainty and some aspects of epistemic uncertainty, and case-based reasoning to handle epistemic uncertainty related to episodic knowledge. We will achieve effects beyond what is possible with one method alone by combining them into a hybrid representation and reasoning framework, and a set of more specific architectures. This is the research hypothesis that guides our work.

Bayesian Networks constitute a modelling framework particularly made for decision making under aleatory uncertainty. Syntactically, a Bayesian network consists of a set of nodes, where each node represents a random variable in the domain, and where there are directed links between pairs of variables. Together, the nodes and arcs define a directed acyclic graph structure. Mathematically, the links and absence of links make assertions about conditional independence

statements in the domain, but for ease of modelling, it is often beneficial to consider a link as carrying information about a causal mechanism [14].

Bayesian Networks can be used for causal inferences (reasoning along the directions of the arc), and for diagnostic inference (reasoning backwards wrt. the causal influences). Recently, there has also been an increased interest in using Bayesian Networks to generate explanations of their inferences (see for instance [15] for an overview).

In case-based reasoning, a collection of past situations or events, i.e. concrete episodes that have happened, make up the knowledge. The concrete episodes are referred to as cases, and the cases - represented in some structural representation language - are stored in a case base, which is a CBR system's knowledge base. The knowledge in a CBR system is therefore situation-specific, as opposed to the generalized knowledge in a BN. A case has two main parts: A problem description and a problem solution. Sometimes a case also includes an outcome part, i.e. the result of having applied the solution to the problem. A CBR system also assigns numerical weights to the features, according to how important a particular feature type or feature value is for characterizing a particular case. In the four-step CBR cycle [5], the RETRIEVE step starts with a problem description and ends when a matching case has been found. REUSE takes that case and tries to build a solution of the new problem, either the easiest way by just copying the solution in the retrieved case over to the new problem, or by making some adaptations to better fit the current problem. In the REVISE step the solution proposed by the system is evaluated in some way, and possibly updated, before RETAIN decides what of this problem solving session should be learned, by updating the case base. A core principle of CBR is the notion of *partial matching*, which means that two cases match to a higher or lesser degree, rather than either match or do not match. Hence, the basic inference method in a CBR system is *similarity assessment*.

On this basis, CBR should be viewed as a method to deal with uncertainty along two dimensions. First, the capturing of domain knowledge as a set of specific experienced situations, rather than general associations, implicitly reflects a degree of uncertainty and incompleteness in the general theories or models of the domain. Second, the similarity assessment procedure that enables the partial matching is a method for reasoning with uncertainty. Uncertainty is captured in the individual feature weights as well as in the computation of total similarity between two cases.

3 Related Research

There is not a large volume of research that describes a combination of BN and CBR methods. Below we have identified five articles of relevance to our architectures, which are presented with a brief description of how they combine BN and CBR.

The earlier Creek system, in which general domain knowledge was represented as a semantic network [16], was extended with a Bayesian component

and exemplified by finding the cause of a "car does not start" problem [8]. The semantic network contains causal links with uncertainty and the probabilistic reasoning is performed by a Bayesian Network. The nodes and causal relations are shared between the semantic network and the BN. The cases are present as variables (on/off) in the BN, and Creek uses the BN to choose relevant cases to apply the similarity measure on (Bayesian case retrieval). The BN is a preprocessing step in the RETRIEVE phase. The Bayesian Network can also calculate causal relations that are used in the adapt method in the REUSE phase.

Tran and Schönwälder [17] describe a distributed CBR system used to find solutions in the communication system fault domain. The problem description is a set of symptoms, S, and the problem solution contains a fault hypothesis, H. Their reasoning process contains two steps: ranking and selection. The ranking step (RETRIEVE phase) finds the most similar cases with their BN relations $S_i|H_j$. The selection step (REUSE phase), use the BN relations $S_i|H_j$ from the cases to build a Bayesian Network. The most probable hypothesis from the BN is chosen.

Gomes [18] presents a computer aided software engineering tool that helps software engineers reuse previous designs. The system combines CBR, BN and WordNet. The cases in the system have a problem description part that contains a number of WordNet synonym sets. The cases are nodes in the Bayesian Network, as are also the synonym sets from the problem description. The synonym sets from the problem description are used to find all the parents from WordNet's *hypernym* relation (is-a), and all the parents are inserted into the BN. The conditional probability tables are built with formulas depending on how many parents the node has. The RETRIEVE phase is performed in three steps as follows: a) the query case description is used to activate (turn on) the synonym sets in the Bayesian Net, b) the BN nodes are calculated and the most relevant cases are found, and c) their probabilities are used to rank the cases.

Bayesian Case Reconstruction (BCR) [19] is a system for design of screening experiments for Macromolecular Crystallization. The domain knowledge is incomplete, the case base is incomplete, there are a large number of variables with a large number of values, and there are limitation on time, material and human resources. BCR is used to expand the coverage of the case base library. The BN is constructed from the domain experts and the content of the case library. The RETRIEVE phase selects the most probable cases, and they are disassembled in order to form new solutions. The Bayesian network contains the causal relations from the domain model that are well understood. In the REUSE phase, the BN is used to find the most probable new solutions. The result is a plausible solution, only.

Another system that combine BN and CBR is used to choose the optimal parameters for an algorithm used in different domains [20]. The case description contains features for an algorithm used on a domain and the case solution is a Bayesian Net. The BN is learned and evaluated through experiments in the domains with the algorithms using different parameter settings. The RETRIEVE phase selects the most similar cases. The REUSE phase is used to calculate a

reliability measure in addition to calculate the most probable arguments from the BN. The most reliable cases are those who have a high number of experiments and a large number of variations in the parameters used.

4 Different CBR and BN Architectures

Case-based reasoning and Bayesian Networks can be combined in the following ways:

- In parallel
- In the sequence BN-CBR
- In the sequence CBR-BN

In the parallel way, both methods use all of the input variables and then produce a classification independently. The results are compared by a "select the best result" algorithm, and the best classification is chosen. Our focus is on integrated approaches, represented by the two sequential combinations. BN and CBR are connected in such a way that the first system in the sequence computes something that the second system needs. The variable types used in the problem domain are as follows:

- I_i is input variable number i. For illustration purpose, see figures 1-4, the variables I_1, I_2, and I_3, are used by the BN only and the variables I_5, I_6 are used by the CBR system only. Input variable I_4 is used by both systems.
- A_j is mediating variable number j. The mediating variables represent concepts in the domain model. An expert of the domain can also be a part of the classification process and he can set evidence on a mediating variable.
- D is a variable that is derived by inference from domain knowledge. It is the main output from the BN in the BN-CBR sequence. It can be the solution of a case, as an intermediate result in the CBR-BN sequence architecture.
- C is a classification variable and it can be calculated by a BN or be the final solution of a case.

The user creates a query description of the problem with the input variables.

Two specializations of each sequence type have been developed. The BN-CBR-1 architecture is shown in Figure 1. The case identifiers are present as variables in the Bayesian network and they have the binary values on/off that indicates if the case is activated or not. The derived variable in the variable set D is causing the cases to be activated. These Ds are *derived* features that are obtained from the input variables by inference based on domain knowledge. Hence, the BN has a filtering role in the RETRIEVE phase of the CBR system. The similarity measures are only applied on the filtered cases. The systems are loosely coupled in this architecture, because the information used in the BN is hidden from the CBR system. An example from the BN-CBR-1 architecture is given in Section 5. The BN-CBR-1 architecture was found in two of the related research articles [8,18].

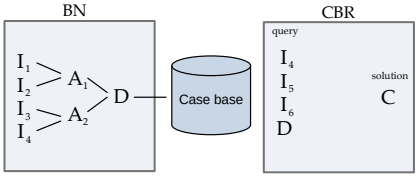

Figure 1. BN-CBR-1 Architecture

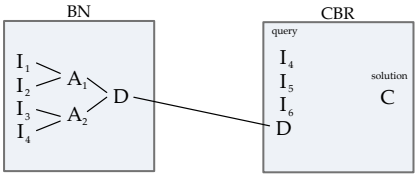

Figure 2. BN-CBR-2 Architecture

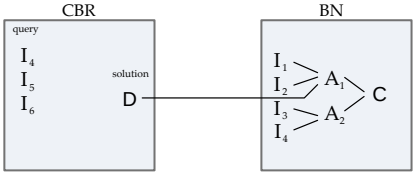

Figure 3. CBR-BN-1 Architecture

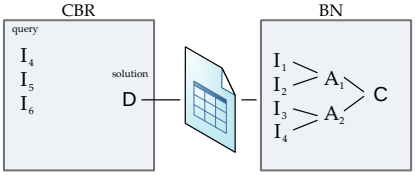

Figure 4. CBR-BN-2 Architecture

The BN-CBR-2 architecture is shown in Figure 2 and here the systems are tightly coupled. The user states all the input variables, and variable [1..4] are set as evidence in the BN, and the expert set his evidence in the BN. The probabilities of the network are calculated and the D is placed in the case description together with I_1, I_2, I_3, I_4. The RETRIEVE phase in CBR is performed, resulting in a ranked list of similar case solutions (classification variable). A variant of the BN-CBR-2 architecture was found among the related research articles [8]. In that work, as interpreted by our current framework, the CBR system is the master and the BN the slave. The domain knowledge is represented by the Bayesian network and a semantic net where the variables in the BN are shared with the semantic net. In our approach, the CBR system uses the Bayesian Network in several steps of the reasoning process. For example, the activate step in RETRIEVE (the steps are from the explanation engine [16]) sets some evidence in the BN that activates the relevant cases (BN-CBR-1 architecture). The explain step in RETRIEVE finds the most similar cases. The focus step in RETRIEVE sets new evidence from the case in the BN and finds new casual probabilities that can strengthen information in the semantic net. The BN can also be used in the REUSE phase.

The CBR-BN-1 architecture is shown in Figure 3 and it shows two tightly coupled systems. The CBR system finds a n-best list with the input variables I_4, I_5, I_6, and the case solution contains the derived variable D. The variable D, the input variables I_1, I_2, I_3, I_4 are set as evidence in the BN, before the posterior probabilities of the classification variable C are calculated. There are two ways to look at the CBR-BN-1 architecture. The first is where the CBR system is a preprocessing step for the BN. The second is where the BN is used in the REUSE phase of CBR. In the first approach, the preprocessing step can be used on a part of the BN model that is unknown. Here an expert can create cases that replace this unknown BN model. The cases must contain D variables

with probabilistic values. Some can also be given by an expert of the domain that is present in the classification process. After the variables are inserted as evidence in the BN, the classification variable is calculated. If the C values are in range of each other there is a possibility of more than one probable class. If the best C value is under a threshold there is no probable class. In the second way of the CBR-BN-1 architecture, the REUSE phase can contain reasoning under uncertainty. The CBR system finds the most similar cases and the REUSE phase can use the BN in order to adapt the case. The classification variable is available in the BN. The CBR-BN-1 architecture was found in one of the related research articles [17].

The CBR-BN-2 architecture is shown in Figure 4. The CBR system uses the input variables I_4, I_5, I_6 to find a solution that contains the most suitable BN model. The BN model is loaded and the input variables I_1, I_2, I_3, I_4 are set as evidence. Afterwards, the classification variable C is calculated. The different BN models has common evidence and classification nodes, but other nodes, causal links, and the conditional probability tables can be different in each model. The information used in the CBR system is hidden from the BN system, and therefore the systems are loosely coupled in this architecture. The CBR-BN-2 architecture was found in one of the related research articles [20].

5 Implementation and example system

We are currently in the process of analyzing previous recorded clinical data, but this is a time-consuming process, and we do not yet have sufficient results for experimentation with the above architectures. Instead of a medical example we are studying the architectures through a simple movie recommendation system. The sole purpose of the experiment is therefore to study the feasibility of a cooperation between BN and CB methods as suggested by one of the architectures. In the following example the BN-CBR-1 architecture is illustrated.

The system evaluates movies based on comparing the user of the system to a set of previous users, and recommends the favorite movie of the previous user that is most similar to the current user. To make sure that any recommendation is suitable for the current user, a filtering process that removes films that are either unsuitable due to age constraints or excessive violence/nudity must be undertaken. The task of generating a recommendation therefore consists of two subtasks: *i*) Finding the movies that are appropriate for the current user. For this task we have a good understanding of the mechanisms, which makes it suitable for the BN method. *ii*) Among those movies, choose the one that she will most probably enjoy. For this task we have no explicit domain model representing what users like to type of movies, making it fit for a CBR method.

A case consists of a description of a user, in the problem description part, and her favorite movie, in the problem solution part. The user description contains personal features (like `Gender`, `Age`, `Occupation`, and `Favorite Movie Genre`). We have no guarantee that the favorite movies of previous users are suitable for the current user; still only the appropriate movies in the system should be

made available for her. This is ensured by letting the BN take charge of the case activation. The input to the BN is each film's `Age Limit`, `Nudity Level`, and `Violence level` together with the current user's `Age`. The output from the BN is the variable `Suitable` with the values `yes` and `no`. The age categories in the BN are `kid`, `small youth`, `big youth`, and `adult`. The first task for the BN is to let the age groups see movies with the correct age limit only. The second task is to restrict the age groups access to movies with nudity and violence. The kids are not allowed to see any movie with nudity and violence. The small youth is allowed to see a little violence, the big youth can see a little violence and nudity. The adult has no restrictions. Assume, for instance, that a 12 year old girl with drama as her favorite genre approaches the recommender system. Only cases recommending movies appropriate for that age, nudity, and violence level are activated. Next, the CBR system uses a local similarity measure that uses a taxonomy, and the result is a list of drama movies free of unsuitable age limits, nudity, and violence.

Our integrated system is implemented with the software components Smile, jColibri, and MyCBR. The CBR development environment jColibri (from the University of Madrid) integrates the Bayesian network software Smile (from the University of Pittsburgh) and local similarity measure functions from MyCBR (developed at DFKI in Kaiserslautern).

Our small experimental study has shown that an integration of CBR and BN according to the properties of each individual method, as captured by the BN-CBR1 architecture is feasible. The system has been implemented to include all the four architectures, and the detailed study of the other three are now in process.

6 Conclusion and Further Plans

We have presented a framework for reasoning under uncertainty by combining BN and CBR, and described four architectures. So far, we have created a simple application using Smile, jColibri, and MyCBR. The BN-CBR-1 architecture can be a preprocessing step to CBR or a part of the similarity measure for uncertain information. BN-CBR-2 and CBR-BN-1 are tightly coupled architectures. Here the uncertain causal relations are present in BN and CBR. BN's strength is to reason under uncertainty with a well understood model, although not requiring a strong theory. CBR's strength is to reason under uncertainty with a model that is less understood. Based on past research, and the current state of our research, it is reasonable to claim that the combination of the strengths of BN and CBR perform better than BN and CBR on their own. However, we still have to provide experimental evidence for this.

In our ongoing and future work, our group will elaborate on how to combine BN and CBR in all the architectures. We will move from our toy domain into medical decision support in palliative care as soon as a sufficient amount of data and knowledge is available.

7 Acknowledgements

The reported research is funded by the TLCPC project, Norwegian Research Foundation under contract no NFR-183362.

References

1. Hamscher, W., Console, L., de Kleer, J., eds.: Readings in model-based diagnosis. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA (1992)
2. Pearl, J.: Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference. Morgan Kaufmann (1988)
3. Jensen, F.V., Nielsen, T.D.: Bayesian Networks and Decision Graphs. 2. edn. Springer Verlag (2007)
4. Kolodner, J.L.: Case-based reasoning. Morgan Kaufmann Publishers (1993)
5. Aamodt, A., Plaza., E.: Case-Based Reasoning: Foundational Issues, Methodological Variations, and System Approaches. AI Communications **7**(1) (1994) 39–59
6. Watson, I.: Applying case-based reasoning: techniques for enterprise systems. Morgan Kaufmann Publishers Inc. San Francisco, CA, USA (1998)
7. Aha, D.W., Marling, C., Watson, I.D.: Case-based reasoning; a special issue on state-of-the-art. The Knowledge Engineering Review **20**(03) (2005)
8. Aamodt, A., Langseth, H.: Integrating Bayesian Networks into Knowledge-Intensive CBR. In: AAAI Workshop on Case-Based Reasoning Integrations. (1998)
9. Hjermstad, M., Fainsinger, R., Kaasa, S., et al.: Assessment and classification of cancer pain. Current Opinion in Supportive and Palliative Care **3**(1) (2009) 24
10. Porter, B.: Similarity Assessment: computation vs. representation. In: In Procs. of DARPA CBR Workshop, Morgan Kaufmann Publishers (1989) 82
11. Patel, V., Arocha, J., Zhang, J.: Thinking and reasoning in medicine (2004)
12. Schmidt, R., Montani, S., Bellazzi, R., Portinale, L., Gierl, L.: Cased-based reasoning for medical knowledge-based systems. International Journal of Medical Informatics **64**(2-3) (2001) 355–367
13. Lindgaard, G., Pyper, C., Frize, M., Walker, R.: Does Bayes have it? Decision Support Systems in diagnostic medicine. International Journal of Industrial Ergonomics **39**(3) (2009) 524–532
14. Pearl, J.: Causality: Models, Reasoning, and Inference. Cambridge University Press (2000)
15. Lacave, C., Díez, F.: A review of explanation methods for Bayesian networks. The Knowledge Engineering Review **17**(02) (2003) 107–127
16. Aamodt, A.: Explanation-driven case-based reasoning. Topics in case-based reasoning (1994) 274–288
17. Tran, H., Schönwälder, J.: Fault Resolution in Case-Based Reasoning. In: Proceedings of the 10th Pacific Rim International Conference on Artificial Intelligence: Trends in Artificial Intelligence, Springer (2008) 429
18. Gomes, P.: Software design retrieval using Bayesian Networks and WordNet. Lecture Notes in Computer Science (2004) 184–197
19. Hennessy, D., Buchanan, B., Rosenberg, J.: Bayesian Case Reconstruction. Lecture notes in computer science (2002) 148–158
20. Pavón, R., Díaz, F., Laza, R., Luzón, V.: Automatic parameter tuning with a Bayesian case-based reasoning system. A case of study. Expert Systems With Applications **36**(2P2) (2009) 3407–3420