A Hybrid CBR and BN Architecture Refined through Data Analysis

Tore Bruland
Norwegian University of Science and Technology
Department of Cancer Research and Molecular Medicine
Trondheim, Norway
torebrul@idi.ntnu.no

Agnar Aamodt*, Helge Langseth†
Norwegian University of Science and Technology
Department of Computer and Information Science
Trondheim, Norway
**agnar@idi.ntnu.no*, †*helgel@idi.ntnu.no*

***Abstract*—The overall goal of this research is to study reasoning under uncertainty by combining Bayesian Networks and Case-Based Reasoning through constructing an experimental decision support system for classification of cancer pain. We have experimentally analysed a medical dataset in order to reveal properties of the data with respect to properties of the two reasoning methods. We also preprocessed our medical data with help from a clinical expert, which resulted in four data sets with different characteristics. This culminates in a hybrid system architecture, where CBR handles the exceptions or outliers with respect to the distribution of the data and the target class, while BN handles the more common situations. Through a set of experiments under varying conditions we show that a hybrid BN+CBR system is favorable over each single method.**

***Keywords*-Bayesian Networks, Case-Based Reasoning, Machine Learning, Hybrid Systems, Decision Support**

I. Introduction

Types of knowledge can be characterized along the two dimensions *strength* of a theory and *completeness* of a theory. A strong theory domain contains statements that are universally true or false. In a strongest possible theory, a perfect theory, all statements are universally true or false. A weak theory domain, on the other hand, contains statements that are more or less plausible, with stronger or weaker support. Lack of theory strength has to be compensated by the utilization of all the relevant knowledge and data in order to build multiple explanations in support of an hypothesis, while a strong theory may need only a single explanation, i.e. a proof. Mathematics is an example of a strong theory domain, as are many other domains artificially created, bounded and controlled by humans. Domains which involve understanding of natural phenomena, and interactions between technologies and nature, are typically weak theory domains (although strong sub-theories often exist). Medical diagnosis and treatment, our target domain in the study reported here, is one such example.

A weak theory domain may be expressed by a set of general statements that are typically true, together with a set of "exceptions", often referred to as *outliers*. An example is a patient not responding to treatment as expected from the general theory. This indicates that the theory is incomplete, i.e. it can not explain all observed instances.

In weak theory domains it is more common to talk about *models* than theories. A model of a part of reality may be *global* or *local*. Global models are generalizations, either acquired manually through top down model construction, or in a bottom-up fashion through automated learning from a set of examples. Local models, on the other hand, are specific to one or a few situations.

Medical knowledge is to a large extent available through guidelines for diagnosis and treatment ("best practice"), but also as clinicians' specific experiences that may or may not conform with the general guidelines. A substantial body of research has shown that it is extremely difficult to build a strong computational model in medicine based on generalized knowledge only (e.g. [1], [2]). Another important source of information that clinicians make use of in their daily practice is the set of personal specific experiences gained through daily work. There is ample evidence that clinicians partly reason from theoretical knowledge, and partly from case-specific or prototype-based experience, depending on how strong causal theories the particular medical area is supported with [3]. It has been argued that computerized medical decision support should to a larger degree concentrate on the rare but difficult patient cases, instead of the more frequent routine ones [4], [5]. Past patient cases provide a level of specificity that is focused on single patients rather than generalized principles. Useful knowledge can be learned from one case only. Generalized and situation-specific knowledge therefore have a strong potential for effectively complementing each other in a decision-support setting.

We are exploring the combination of the two knowledge types by studying architectures for integration of Bayesian Networks (BN) and Case-Based Reasoning (CBR) at two different levels: One is the level of functional modules, where CBR and BN modules cooperate according to the relative strengths of the two knowledge types for the particular tasks addressed. The other is the level of data characteristics, where an analysis of existing data is used to guide how problem solving should be split between BN and CBR. An architecture of functional modules was described in an earlier paper [6], while the level of data characterization is the focus of the present paper.

Our target application is the classification of pain for patients in palliative care. The target class is a patient's pain level after two weeks from the current date. The pain level can take one of three values: mild, moderate, severe. The three values are abstractions over a scale from 0 to 10. The problem is a prediction problem casted as a classification problem. Based on a data set of approximately 1800 patients, we have used machine learning methods to study the dependencies between the target class and the various patient features. The purpose has been to get an understanding of the overall data landscape, to look for combinations of features and feature values that are more predictive of the target class than others, and to compare a BN and a CBR method for prediction accuracy and model transparency. We present the experiments and discuss the results. Initially, in order to reveal characteristic properties of BN and CBR from data under our own control, a synthetic data set was generated. Results from that analysis then guided our experiments on the medical data.

We give a brief background in the next section, in terms of core properties of BN and CBR, and a summary of the high-level functional architecture. This is followed in Section III by an overview of the hybrid data-level architeture. In Section IV our experiments with a synthetic dataset is presented, followed by an analysis of CBR and BN properties on the medical data in Section V. In section VI the hybrid CBR+BN system is tested. Finally, we conclude and give directions for future research.

II. Background

A Bayesian network consists of a directed acyclic graph and quantitative probability information [7], [8]. The graph contains nodes that represent the random variables in the domain, and use directed links to assert conditional independence statements (the links are often interpreted as carrying causal information [9]). For each random variable, one has to define the conditional distribution of that variable given the variable's *parents* (the nodes in the graph having links pointing directly into that node). The joint distribution over the variables $\{x_1, \ldots, x_n\}$ can then be calculated as

$$P(x_1, \ldots, x_n) = \prod_{i=1}^{n} P\left(x_i | \text{parents}(x_i)\right). \qquad (1)$$

A BN can either be constructed by domain experts or be learned from data (or by a combination of these approaches, [10]). The resulting model will be the BN with the best ability for predicting new instances, and typically a BN is therefore optimized for *density estimation* and not *classification*. This can be alleviated using one of the numerous Bayesian classifier schemes, instead of a general-purpose learning scheme [11].

Case-Based Reasoning [12] is a method that solves new problems from previous solved reasoning tasks. CBR has a vocabulary to describe a case as a problem statement, a solution, and possibly an outcome. A case base holds the previously solved cases, and the reasoning process can be described as a cycle with the following four steps [13]: First, a query case is described and the RETRIEVE step finds all similar cases from the case base. The best matching case is selected and the REUSE step takes this case and adapts it to fit the best solution. The REVISE step takes the solution and evaluates its quality. The final step is RETAIN, which learns from the problem solving experience by updating the case base.

CBR systems are lazy learners, which delay the inductive learning step until a new instance arrives. Similarity assessment is a core problem in CBR, and the methods range from simple, global similarity metrics to complex algorithms for local similarity that also take situational context and general domain knowledge into account. An example of a global similarity function, which calculates the similarity between a query case and a target case, is shown in Equation (2).

$$\text{Sim}(T, Q) = \sum_{i=1}^{n} w_i \cdot f(t_i, q_i), \qquad (2)$$

where T is a target case, Q is the query case, t_i is feature i from the target case, q_i is feature i from the query case, n is the number of features, i indicates the individual feature, ranging from 1 to n, f is the feature similarity function, and w_i is the weight for feature i.

Both BN and CBR are methods that handle uncertainty, although in different ways. The uncertainty that a decision maker is faced with can be divided into *aleatory* and *epistemic* uncertainty, where the aleatory uncertainty is the general randomness that can be characterized by probability distributions, and epistemic uncertainty is insufficient knowledge. The different types of uncertainty requires different reasoning processes, and we have previously argued for a BN and CBR hybrid, and a corresponding architecture [6], to handle the two types of uncertainty.

The architecture has four sub-architectures that represent different ways of combining CBR and BN:

- In BN-CBR-1, a BN is used to pre-process the case base, and only the cases that are found relevant by the BN model are used by CBR. This architecture is similar to the one used by Gomes [14].
- In BN-CBR-2, the BN model infers a set of variables used by the CBR system. The CBR query is constructed from that variable and the input variables. A similar set up was also used by our research group in earlier work [15].
- In CBR-BN-1, a CBR solution is used to update a node in a BN model. A somewhat similar architecture is used by Tran et al. [16]. Frank et. al. selects a large number of nearest neighbours from a query, and train a Naive Bayes model before classification [17].
- In CBR-BN-2, a CBR solution contains a local BN

model. This architecture has similarities with the one used by Pavón et al. [18].

The four sub-architectures are meant to be basic building blocks in a larger reasoning system. For example, if a step in the CBR cycle needs to reason with uncertainty, then some variables from the CBR system can be updated in the BN model, where inference subsequently takes place. The result is returned and CBR can continue its process. It is also possible to go the other way: a BN model can have a need for a more detailed local model and this is possible with CBR. A query case is created from some nodes in the BN model and after a similar case is found, its solution is used to update the BN model and the reasoning process can continue.

III. The Data Level Architecture

The role of CBR in our architecture is to provide decision support for the few but difficult patient cases, i.e. the patients that do not conform with generalized patters and routine practice. These are the most time consuming and resource demanding patients, for which intelligent decision support is particularly called for. Our approach is in line with other hybrid methods that use CBR to handle outliers, exceptions, and non-compliances, while generalization-based methods take care of the more common type of situations [4].

Our understanding of the strengths and weaknesses of the reasoning processes leads us towards a system in which the "typical" patients are treated as information that can be utilized by the the BN, and outliers are utilized by the CBR engine. The hybrid system, shown in Fig.1, can be put into effect when we are able to specify the criteria for splitting the data between BN and CBR. In the preprocessing phase, the medical data is collected from each incoming patient and the result of the split is stored in an *outliers* data store and in a *commons* data store. The *commons* data store is the basis of the BN model created with a data mining tool, and the model is accessible by the BN software used in the hybrid. The architecture can be instantiated in two different ways, indicated by the arrows "A" and "B" in the figure:

A) When a new patient arrives, the CBR system checks whether it sufficiently matches one or more past outliers. If that is the case, the solutions of the similar cases are suggested to the physician. If no matching outlier is found, the BN system is used to classify the patient according to the generalized knowledge.

B) When a new patient arrives, the BN system checks whether it is a patient of a common type by calculating the target class probabilities in the BN. If the observation is sufficiently well covered by the model, and one class has a sufficient "degree of belief" in the Bayesian network, that class is suggested to the clinician. If not, a check of whether the patient matches a past outliers is made, and if so the solution of the similar cases are suggested to the clinician.

In addition to the fixed and sequential strategies A and B the architecture opens up for iterative control loops. For example, instead of only one threshold for each of the CBR or the BN components, representing a similarity degree and a probabilistic degree of belief, respectively, a set of thresholds of decreasing values can be defined. After first the CBR component and then the BN have failed (strategy A), a lower level of thresholds is activated and the control is given back to the initial method (here: CBR). The loop iterates until the lowest levels have been reached, and the system has to give up. It should be noted that to display a few of the closest matched outlier cases, and/or the set of most likely target classes with probabilities, even if the thresholds are not high, can be of valuable help to the clinician.

This data level architecture raises some essential questions:

1) Are outliers better handled separately (e.g., by a CBR component in a hybrid system), or as an integral part of one system?
   - Does the incorporation of local models give improvements over a global model? We would have to consider not only prediction accuracy in this situation, but also include other important features, like model transparency.
   - To utilize a hybrid model, we must be able to *detect* that a case is an outlier. Can this be easily and reliably done?
2) Our medical data is marred with what we call *duplicates*; separate objects that share the same description (but not necessarily have the same class label). What effect do duplicates have on the classification results?
3) Given that a hybrid system is used: Should we use the same data representation for BN and CBR, or should we adapt the representation of new objects to the reasoning system we employ for that object?

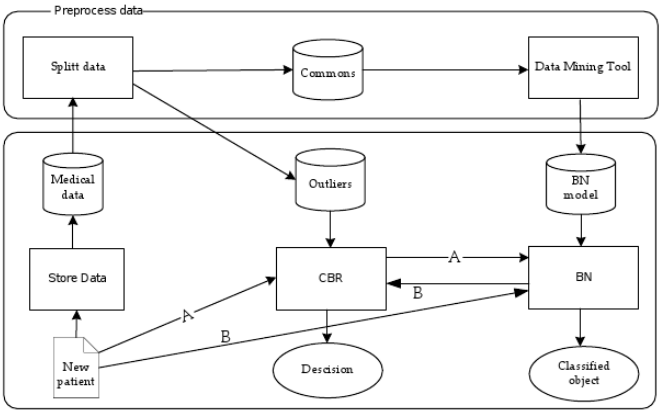

Figure 1. The Hybrid System and the Preprocessing Step

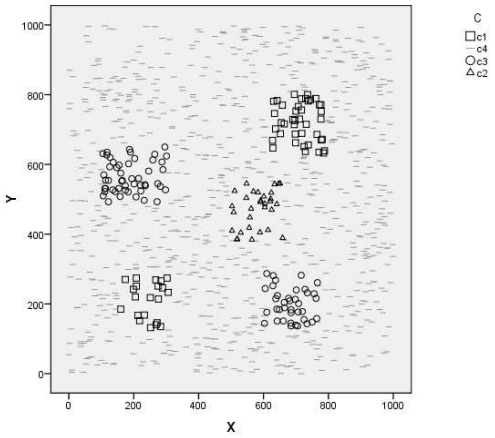

Figure 2. 1000 Instances of the local/global dataset

IV. Experimenting with synthetic data

The aim of this pre-study is to investigate how the components of the hybrid system behave in a controlled setting. We will produce datasets with given characteristics to see how these properties influence the behaviour of the BN and CBR models, with a view towards better understanding the two reasoning models' abilities. We will attempt to answer the first two essential questions raised in the previous section. The third will be addressed through experiments with the medical data set.

We use a knowledge poor instance-based learner (IBL) as the representative of a CBR classifier in the experiments. BN and IBL address different aspects of machine learning. For instance, BN is good at building a joint probability distribution (relating to the idea of a "global" model), where IBL utilizes the notion of similar cases ("local" model). Whenever a "global" model is appropriate, we would expect BN to be able to learn this model from data, and to represent the model such that it can be inspected and understood by a user. If "local" models are appropriate, we expect a BN model to perform poorly, in terms of predictive accuracy or model transparency (or both).

Our first experiment is designed to investigate the notion of local and global models, and to do so, we generated a synthetic dataset. The dataset contains the features X and Y, and the class variable C. The dataset is generated using the algorithm shown in Listing 1, and we generate datsets with 200, 500, 1000, 5000 and 9000 instances, see Fig.2, where the class values are divided into groups in attribute space.

Listing 1. Algorithm for Calculating the Class for Dataset 1

```
if ( x in [152,319] and y in [127,286] )
  class = c1;
else if ( x in [494,660] and y in [383,549] )
  class = c2;
else if ( x in [602,767] and y in [126,288] )
  class = c3;
else if ( x in [104,307] and y in [489,655] )
  class = c3;
else if ( x in [610,794] and y in [625,808] )
  class = c1;
else
  class = c4;
```

From the scatter plot in Fig.2 we can see that the data resembles the number five of a dice, where the first class state is divided between the pip up to the left and down to the right, the second is the pip up to the right and down to the left, the third class state is the pip in the center, and the forth class state is the rest. One would think that a representation utilizing local models would suit this dataset well. It is, for instance, clear from Listing 1 that as long as $X > 794$, the class is $c = c_4$, independently of the value of Y, hence in the context defined by $X > 794$ we would not want to involve Y in the classification process. On the other hand, for the case $X \in [610, 660]$, it is y that defines the class, and a different model structure is required.

Table I
Accuracy results for the local/global dataset

Num	200	500	1000	5000	9000
BN	80.0 (0.0)	94.3 (2.8)	96.3 (1.9)	99.4 (0.3)	99.7 (0.1)
IB1	89.8 (6.4)	93.5 (3.2)	95.8 (2.1)	98.3 (0.5)	98.8 (0.3)
IB5	92.2 (6.4)	92.2 (3.7)	95.5 (2.0)	98.2 (0.5)	98.7 (0.3)
IB9	89.7 (6.2)	93.4 (3.3)	95.5 (1.9)	98.1 (0.6)	98.5 (0.3)

The data mining tool Weka [19] is used to run the experiments, and we used the following algorithms: i) BN classifier [20] with the *simulated annealing* and *tabu search* [21] options. Note that the BN implementation in weka assumes discrete data, hence a discretisation procedure must also be run [22]. ii) For IBL, the k-nearest neighbors classifier IBk [23], mostly run with $k = 1$. The results are shown in Table I, where the classification accuracies and related standard deviations are shown as a function of the size of the dataset. Not surprisingly, IBk is better than BN for the smallest dataset (200 cases). The BN algorithm does not have a sufficient amount of data to choose the rather complex gold standard model, and also fails to discretise the data with sufficient granularity. However, the BN algorithm surpasses the IBk algorithm in classification accuracy, and for the larger datasets ($N > 1000$), the BN algorithm performs slightly better than IBk in terms of accuracy, and also obtaining more robust results (lower standard deviation). Changing the number of neighbors (option k) in IBk does not seem to affect the classification accuracy significantly. Just looking at the predictive accuracy, one would then conclude that the BN is superior to IBk, and we have no argument for the proposed data-level architecture: As long as sufficient data is available, a BN can learn any distribution arbitrarily well, and therefore outperform "local" models. However, model transparency is also crucial in our domain, and we therefore investigate the learned BN, see Fig.3. Although the model captures the data generation process with high accuracy, it is very difficult to understand the

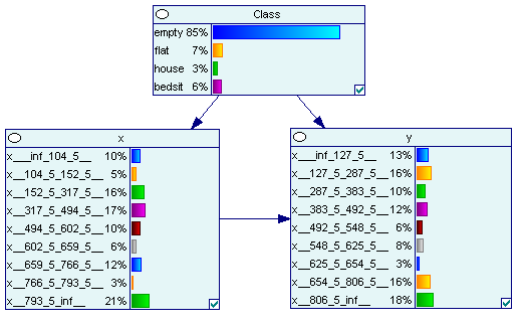

Figure 3. Learned BN Model, $N = 9000$.

classification rule it implements as it is hidden inside the conditional probability tables of the model. By examining the graphical structure alone we can, for instance, not see the context-specific independence [24] of Y and C when $X > 794$, nor understand that the classification function is particularly reliant on Y when $X \in [610, 660]$. The BN structure's representation is a global model, and the local models of context-specific independence is somewhat lost.[1] The answer to our first question is therefore that a single global model is infeasible when the domain can be more naturally described by a set of local models.

We can detect "outliers" either using the BN or the CBR sub-system. The BN can simply use the match between the model and the query (measured by, e.g., the *conflict measure*, [8]) to determine that the query fits poorly with the model. Alternatively, the CBR sub-system can compare a new query to its existing case-base of outliers. Either way, our system is able to detect outliers effectively. We therefore conclude that a hybrid data-level architecture is both useful and realizable.

Our next fundamental question relates to duplicates (also known as "class ambiguities"), and we will now proceed by investigating the effect duplicates have on the performance of BN and IBL using a synthetic dataset. The dataset contains the features X, Y and C. The variables are nominal (categorical scale), where X can take any of the four states x_1, x_2, x_3 or x_4, $Y \in \{y_1, y_2, y_3\}$, and the possible classes are $\{c_1, c_2, c_3, c_4, c_5\}$. A random number generator assign states to X and Y (each configuration being equally probable), and the class was defined using the algorithm in Listing 2.

Listing 2. Gold standard model, "Duplicates" dataset

```
if ( x1 and y1 ) class=c1;   if ( x1 and y2 ) class=c2;
if ( x1 and y3 ) class=c3;   if ( x2 and y1 ) class=c4;
if ( x2 and y2 ) class=c5;   if ( x2 and y3 ) class=c1;
if ( x3 and y1 ) class=c2;   if ( x3 and y2 ) class=c3;
if ( x3 and y3 ) class=c4;   if ( x4 and y1 ) class=c5;
if ( x4 and y2 ) class=c1;   if ( x4 and y3 ) class=c2;
```

[1] Alternative methods of learning BNs from data, including [25], attempt to learn a representation of the conditional probability tables that capture this aspect, but this only partly alleviates our problem, as one still will have a global BN structure, which does not capture the context-specific independences.

We say that two observations $\boldsymbol{o}_1$ and $\boldsymbol{o}_2$ are *duplicates* if their attribute descriptions are identical (i.e., equality for both x and y in the synthetic dataset). Further, duplicates are of Type 1 (also denoted *copies*) when two objects are equal (including the class belonging), duplicates of Type 2 (or *class ambiguity*, also known as *class noise* [26] and *label misclassification* [27]) when the objects have different values for the class. For example, the two observations $\boldsymbol{o}_1 = (x_1, y_1, c_1)$, and $\boldsymbol{o}_2 = (x_1, y_1, c_2)$ are duplicates of Type 2.

The first version of the data set contains Type 1-duplicates only, giving the total size $N = 500$ observations. The second version of the dataset starts from the first, but has 10% of the instances switched from duplicate of Type 1 to Type 2. For example, the dataset has 42 duplicates of Type 1 for the object (x_1, y_1) , and 3 duplicates of Type 2. Gradually, the dataset is more and more corrupted with class ambiguities, giving a total of four different datasets. The results are presented in Table II. We notice that the classification accuracies for BN and IBk are similar, and that the results are as expected. Obviously, the target concept (Listing 2) is very simple, and $N = 500$ noise-free observations is sufficient for the methods to detect the gold standard model (Version 1 of the dataset). Next, duplicates of Type 2 seem to make both methods perform poorer. This is not surprising, as increasing the frequency of class ambiguities is essentially the same as increasing the theoretically achievable error of the Bayes optimal classifier [28], see also [26], [27].

The answer to the second topic of the previous section is thus divided into two parts: i) Copies (duplicates of Type 1) are helpful when learning a BN from data, as more observations give a better picture of the underlying distribution that generates the dataset. Similarly, copies lengthen the list of similar cases in IBL and they have a positive impact when $k > 1$. ii) Class ambiguity (Type 2 duplicates) confuse both algorithms, and the end result is a lower classification accuracy.

V. Experimenting with the Medical Dataset

Based on the discussion of the previous section, we conclude that a hybrid model can be useful when we want to optimize predictive accuracy and model transparency together. In this section we will continue our investigations with a view towards deciding on the representation that the different reasoning systems should work with. It is well known that where BN is a probabilistic model, which

Table II
Results for the Duplicates dataset

Ver	Amb	BayesNet	IB1	IB5	IB17
1	0	100.0 (0.0)	100.0 (0.0)	100.0 (0.0)	100.0 (0.0)
2	10	90.4 (3.6)	90.4 (3.6)	90.4 (3.6)	90.4 (3.6)
3	20	82.0 (4.3)	82.2 (4.3)	82.2 (4.3)	82.2 (4.3)
4	30	76.0 (5.4)	76.0 (5.4)	76.0 (5.4)	76.0 (5.4)

gracefully handles noisy or irrelevant attributes in the object description, IBL can be harmed by this type of data. This motivates that different representations can be optimal for the two systems, something we will investigate using our medical dataset.

The hybrid system used for these experiments is the CBR-BN-1 architecture introduced in Section III. It is implemented within jColibri (CBR software from the University of Madrid) and Smile (Bayesian network software from the University of Pittsburgh).

The dataset for these experiments is anonymized and contains information about 1800 patients with cancer pain. The dataset is basically a time series, where a patient is examined at point of entry, referred to as the baseline, then measurements are repeated after one week, two weeks, three, four, and twelve weeks. The target class we want to predict is the patient's average pain intensity in week 2. In our first experiment only baseline data are given as input, while in a second experiment week 1 data are included as well.

We want to test the effect of feature relevance on the methods, so a cancer pain specialist has selected the most important features and placed them into the four groups: $\mathcal{A}$, $\mathcal{B}$, $\mathcal{C}$, and $\mathcal{D}$. Group $\mathcal{A}$ contains the 5 most important features, group $\mathcal{B}$ adds 5 slightly less relevant features, etc. The full dataset (denoted $\mathcal{A} \cup \mathcal{B} \cup \mathcal{C} \cup \mathcal{D}$ or $\mathcal{ABCD}$ for short) consists of 22 attributes. A brief statistical analysis is run and frequency diagrams are created for numeric features and frequency tables for categorical features. Only a few numeric features were present; the features with a few integer values are converted into categorical features. After removing instances with missing data in important features, the remaining dataset contains 1569 instances. Three features about pain intensity are scaled down from a range of 0-10 to three states (mild, moderate and severe). We then analyzed the data by learning BN and IBk classifiers, and calculating the predictive ability of these models.

Table III
CLASSIFICATION ACCURACY FOR DIFFERENT ATTRIBUTE SETS

Dataset	BN	IB1	IB5
$\mathcal{A}$	64.1 (3.2)	60.2 (3.6)	61.2 (3.3)
$\mathcal{AB}$	64.2 (3.2)	56.0 (4.1)	60.3 (3.6)
$\mathcal{ABC}$	62.9 (3.3)	54.2 (3.6)	60.1 (3.6)
$\mathcal{ABCD}$	62.5 (3.5)	50.0 (3.2)	55.4 (3.7)

The results, see Table III, indicate that the domain expert's indications of which attributes are the most important are of high quality. There is no model that can show (significantly) improved accuracy as attributes outside dataset $\mathcal{A}$ are added. We conclude that for the present data, we can use the attributes in dataset $\mathcal{A}$ as representation for both reasoning systems, and do not need to make method-specific representations.

In the second set of experiments data from baseline are combined with data from week 1 in order to classify the average pain for week 2. The full dataset consists of 29 attributes. As Table IV shows, and not surprisingly, we get better results than without week 1 data. In this test two other machine learning methods was added for comparison, support vector machines (SVM) and Adaboost, which both are regarded among the best classifier methods for this type of data. The SVM result where computed by LibSVM (software from the National Taiwan University) with the Easy.pl Python program. The Adaboost result was computed with Weka's AdaboostM1 classifier with the default *decision stump* option.

Table IV
CLASSIFICATION ACCURACY AS FOR DIFFERENT ATTRIBUTE SETS

Dataset	BN	IB9	Adaboost	SVM
$\mathcal{A}$ mix	69.2 (3.6)	69.5 (3.3)	70.6 (3.4)	70.7
$\mathcal{AB}$ mix	70.2 (3.5)	67.6 (3.3)	70.6 (3.4)	70.9
$\mathcal{ABC}$ mix	67.6 (3.0)	66.6 (2.9)	70.6 (3.4)	70.7
$\mathcal{ABCD}$ mix	67.4 (3.2)	65.7 (3.1)	70.6 (3.4)	70,6

The goal was to investigate whether BN and IBk gave a better accuracy combined than on their own. The IBk classifier with Euclidean distance was augmented with a simple 'domain model' and a different class calculation. The domain model is very simple: It assumes that the variables *baseline average pain* and *week 1 average pain* must be equal in the query and the instance. A Bayesian model is used to calculate the class value. One calculation is performed for each instance that is picked from the ranked list of matched cases. The dataset $\mathcal{ABCD}$ mix was split into ten test sets and ten training sets. Ten Bayesian models were trained on each training set using Weka, and the models were converted into Smile format. The data was discretized in order to simplify the communication between BN and IBk. Five classifiers were used:

- c1 Bayesian Network
- c2 IBk
- c3 IBk with BN (IBk-bn)
- c4 IBk with domain model (IBk-m)
- c5 IBk with BN, and domain model (IBk-bn-m)

The parameters for the IBk classifiers where $k = 1, 3, 5, 7, 9$ and 11. The Bayesian models where learned with *tabu search* and maximum five parents. The best average accuracies are shown in Table V.

Table V
BEST AVERAGE ACCURACY

	c1	c2	c3	c4	c5
k		9	9	9	3
accuracy	67.5	64.6	68.8	68.4	69.6

The differences between Bayesian Networks and the other classifiers are shown in Table VI together with the p-values from a two tailed paired t-test. The pair BN and IBk have a p-value of 0.01619 which means that BN has a better

accuracy (statistical significant at level 0.01619). The pair BN and IBk-bn, and the pair BN IBk-m have the p-values 0.13405 and 0.50179, respectively, and statistically significant differences can thus not be claimed. The pair BN and IBk-bn-m has a p-value of 0.01093. Since the BN classifier was better than the IBk, the IBk-bn-m classifier is also better than the IBk. The results show that the combination of IBk, BN and a model is better than BN and IBk alone.

Table VI
ANALYSIS OF STATISTICAL SIGNIFICANCE

	c1-c2	c1-c3	c1-c4	c1-c5
fold 1	-0,01000	-0,04000	-0,05000	-0,04000
fold 2	0,02000	0,03000	0,02000	0,01000
fold 3	0,10000	0,02000	0,05000	0,00000
fold 4	0,02000	-0,02000	-0,08000	0,00000
fold 5	0,06000	-0,03000	0,01000	-0,03000
fold 6	0,00000	-0,02000	-0,02000	-0,02000
fold 7	0,02000	-0,01000	-0,04000	-0,02000
fold 8	0,03000	-0,01000	0,04000	-0,03000
fold 9	0,03000	-0,05000	-0,01000	-0,06000
fold 10	0,02000	0,00000	-0,01000	-0,02000
mean	0,02900	-0,01300	-0,00900	-0,02100
var	0,00097	0,00062	0,00165	0,00043
t	2,95127	-1,64658	-0,69971	-3,19423
abs-t	2,95127	1,64658	0,69971	3,19423
p	0,01619	0,13405	0,50179	0,01093

A version of the nearest-neighbor classifier that uses a radius (IBr), instead of a fixed k, was also created. This classifier was also augmented with a simple model and a different class calculation. Some queries did not have any similar cases, because of the radius setting.

Table VII
RESULTS USING RADIUS

	IBr	IBr-bn	IBr-m	IBr-bn-r
Radius	$\sqrt{5}$	$\sqrt{6}$	$\sqrt{7}$	$\sqrt{7}$
% outliers	53.9	30.4	20.4	20.4
Accuracy	68.0	70.3	66.0	71.2

The outliers in table Table VII are the average number of outliers among the instances in the test datasets. The table contains some results from the radius classifiers. Since all variables are categorical, the radius indicates the maximum variables that can be unequal. A radius equal to $\sqrt{5}$ permits up to 5 unequal variables among 29, and the table shows that IBr got 53.9 % outliers (averaged over the ten datasets). The remaining test instances have an accuracy of 0.68.

We also investigated two other datasets (credit-g and waveform) from the UC Irvine Machine Learning Repository, in order so see if they followed the same pattern as the medical dataset. In a search for simple domain models for these data sets we did not find any suitable models that could improve the accuracy. The c3 classifier showed similar accuracy as the c1 classifier on the credit-g dataset. None of the classifiers with the nearest neighbor performed well on the waveform dataset.

VI. CONCLUSION AND FURTHER WORK

In the research reported here we have studied the combination of case-based reasoning and Bayesian networks for medical decision support. A central issue has been to represent and reason with different forms of uncertainty, and to that end utilize the individual abilities of CBR and BN, leading to a data-level architecture of our hybrid system. The combination of BN and CBR in the architecture takes advantage of an important property of both methods: The knowledge contained in a BN represents generalized knowledge in terms of statistical distributions. As such it holds information about individuals (i.e. patients) that are "averaged" over the number of patients, and hence influenced by the number of individuals with the same feature values. A case base, on the other hand, contains knowledge at the level of an individual patient, and the knowledge contribution of a single individual, represented as a case, is not necessarily dependent on whether there are many other individuals with the same feature values. What matters is that there is a strong similarity match between a query case (a new patient) and an existing case in the case base (a past patient).

The architecture raised a number of important questions, and a substantial data analysis was undertaken in order to identify characteristics of the data with respect to the properties and abilities of simple instantiations of the two methods.

Experiments with synthetic datasets showed that the BN is capable of learning complex goal models if it has sufficient training data, but potentially at the cost of model transparency. Further, we verified that both methods have problems with the training data that contains class ambiguities. It is known that the BN method is more robust to irrelevant features than the IBL-based CBR, but a domain expert's feature selection was shown to be helpful in that respect, and we concluded that both reasoning systems can utilize the same patient representation.

An IBL learner, i.e. a knowledge-poor CBR method, was used in the experiments described in this paper. This was an initial study to identify characteristics of a plain and simple CBR method as such. In our ongoing work we shift to a more knowledge-intensive CBR approach, inspired by our earlier Creek system [6], [15], but replacing the original semantic network with a Bayesian network. We are currently in the process of building a domain model using BN-based Influence Diagrams.

ACKNOWLEDGMENT

This research is partly conducted within the project TL-CPC (Transactional Research in Lung Cancer and Palliative Care), a nationally funded project lead by Stein Kaasa (contract no NFR-183362) in which we cooperate with the Medical Faculty of NTNU and the St. Olav Hospital in Trondheim. We wish to thank Cinzia Brunelli and Anne

Kari Knudsen for providing the medical data set, interpreting the data, and analyzing the relevance of the features from a clinical perspective.

REFERENCES

[1] D. R. Patel, V.L.and Kaufman and A. J. F., "Emerging Paradigms of Cognition in Medical Decision-making," *Journal of Biomedical Informatics*, vol. 35, no. 1-2, pp. 52–75, 2002.

[2] R. Schmidt, S. Montani, R. Bellazzi, L. Portinale, and L. Gierl, "Cased-based Reasoning for Medical Knowledge-based Systems," *International Journal of Medical Informatics*, vol. 64, no. 2-3, pp. 355–367, 2001.

[3] W. Ahn, N. Kim, M. Lassaline, and M. Dennis, "Causal Status as a Determinant of Feature Centrality* 1," *Cognitive Psychology*, vol. 41, no. 4, pp. 361–416, 2000.

[4] R. Schmidt and O. Vorobieva, "Explaining Medical Model Exceptions," *Computational Intelligence in Healthcare 4*, pp. 265–287, 2010.

[5] S. Montani, "Case-based Reasoning for Managing Non-compliance with Clinical Guidelines," *Computational Intelligence*, vol. 25, no. 3, pp. 196–213, 2009.

[6] T. Bruland, A. Aamodt, and H. Langseth, "Architectures Integrating Case-Based Reasoning and Bayesian Networks for Clinical Decision Support," in *Intelligent Information Processing V*, Z. Shi, S. Vadera, A. Aamodt, and D. Leake, Eds. Springer, 2010, pp. 82–91.

[7] J. Pearl, *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann, 1988.

[8] F. V. Jensen and T. D. Nielsen, *Bayesian Networks and Decision Graphs*, 2nd ed. Springer Verlag, 2007.

[9] J. Pearl, *Causality – Models, Reasoning, and Inference*. Cambridge, UK: Cambridge University Press, 2000.

[10] D. Heckerman, D. Geiger, and D. M. Chickering, "Learning Bayesian Networks: The Combination of Knowledge and Statistical Data," *Machine Learning*, vol. 20, pp. 197–243, 1995.

[11] R. Greiner, A. J. Grove, and D. Schuurmans, "Learning Bayesian Nets that Perform Well," in *Proceedings of the Thirteenth Conference on Uncertainty in Artificial Intelligence*. San Francisco, CA.: Morgan Kaufmann Publishers, 1997, pp. 198–207.

[12] D. W. Aha, C. Marling, and I. D. Watson, "Case-Based Reasoning; a Special Issue on State-of-the-Art," *The Knowledge Engineering Review*, vol. 20, no. 03, 2005.

[13] A. Aamodt and E. Plaza., "Case-Based Reasoning: Foundational Issues, Methodological Variations, and System Approaches," *AI Communications*, vol. 7, no. 1, pp. 39–59, 1994.

[14] P. Gomes, "Software Design Retrieval Using Bayesian Networks and WordNet," *Lecture Notes in Computer Science*, pp. 184–197, 2004.

[15] A. Aamodt and H. Langseth, "Integrating Bayesian Networks into Knowledge-Intensive CBR," in *AAAI Workshop on Case-Based Reasoning Integrations*, 1998.

[16] H. Tran and J. Schönwälder, "Fault Resolution in Case-Based Reasoning," in *Proceedings of the 10th Pacific Rim International Conference on Artificial Intelligence: Trends in Artificial Intelligence*. Springer, 2008, p. 429.

[17] E. Frank, M. Hall, and B. Pfahringer, "Locally Weighted Naive Bayes," in *Proceedings of the 19th Conference on Uncertainty in Artificial Intelligence*, vol. 256, 2003, pp. 249–256.

[18] R. Pavón, F. Díaz, R. Laza, and V. Luzón, "Automatic Parameter Tuning with a Bayesian Case-Based Reasoning System. A case of study," *Expert Systems With Applications*, vol. 36, no. 2P2, pp. 3407–3420, 2009.

[19] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. Witten, "The WEKA Data Mining Software: An update," *ACM SIGKDD Explorations Newsletter*, vol. 11, no. 1, pp. 10–18, 2009.

[20] R. R. Bouckaert, "Bayesian Network Classifiers in Weka ," Weka documentation, 2008.

[21] ——, "Bayesian Belief Networks: from Construction to Inference," Ph.D Thesis, The University of Utrecht, 1995.

[22] U. Fayyad and K. Irani, "Multi-interval Discretization of Continuous-valued Attributes for Classification Learning," in *Proceedings of the Thirteenth International Joint Conferance on Artificial Intelligence*. Chambury, France: Morgan Kaufmann, 1993, pp. 1022–1027.

[23] D. Aha, D. Kibler, and M. Albert, "Instance-Based Learning Algorithms," *Machine learning*, vol. 6, no. 1, pp. 37–66, 1991.

[24] C. Boutilier, N. Friedman, M. Goldszmidt, and D. Koller, "Context-Specific Independence in Bayesian Networks," in *Proceedings of the Twelfth Annual Conference on Uncertainty in Artificial Intelligence*. Morgan Kaufmann, 1996, pp. 115–123.

[25] N. Friedman and M. Goldszmidt, "Learning Bayesian networks with local structure," in *Proceedings of the Twelfth Annual Conference on Uncertainty in Artificial Intelligence*. Morgan Kaufmann, 1996, pp. 252–262.

[26] X. Zhu and X. Wu, "Class Noise vs. Attribute Noise: A Quantitative Study of Their Impacts," *Artificial Intelligence Review*, vol. 22, no. 3, pp. 177–210, 2004.

[27] C. Brodley and M. Friedl, "Identifying Mislabeled Training Data," in *Journal of Artificial Intelligence Research*. AI Access Foundation and Morgan Kaufmann Publishers, 1999, pp. 131–167.

[28] T. M. Mitchell, *Machine Learning*. Boston, MA.: McGraw Hill, 1997.